applied sciences

Special Issue Reprint

New Advances in Audio Signal Processing

Edited by
Giovanni Costantini and Daniele Casali

mdpi.com/journal/applsci

MDPI

New Advances in Audio Signal Processing

New Advances in Audio Signal Processing

Editors

Giovanni Costantini
Daniele Casali

MDPI

Basel • Beijing • Wuhan • Barcelona • Belgrade • Novi Sad • Cluj • Manchester

Editors

Giovanni Costantini
Electronic Engineering
University of Rome
Tor Vergata
Rome
Italy

Daniele Casali
Electronic Engineering
University of Rome
Tor Vergata
Rome
Italy

Editorial Office
MDPI
St. Alban-Anlage 66
4052 Basel, Switzerland

This is a reprint of articles from the Special Issue published online in the open access journal *Applied Sciences* (ISSN 2076-3417) (available at: https://www.mdpi.com/journal/applsci/special_issues/Audio_SP).

For citation purposes, cite each article independently as indicated on the article page online and as indicated below:

Lastname, A.A.; Lastname, B.B. Article Title. *Journal Name* **Year**, *Volume Number*, Page Range.

ISBN 978-3-7258-1059-8 (Hbk)
ISBN 978-3-7258-1060-4 (PDF)
doi.org/10.3390/books978-3-7258-1060-4

Cover image courtesy of Valerio Cesarini

Contents

About the Editors

Giovanni Costantini

Giovanni Costantini received his master's degree in Electronic Engineering from Sapienza University of Rome and his Ph.D. in Telecommunications and Microelectronics Engineering from the University of Rome Tor Vergata.

He is an associate professor at the Department of Electronic Engineering of the University of Rome Tor Vergata, where he is the Director of the Master in Sonic Arts and chairs committees on circuit theory, digital sound processing, biological and multimedia signal processing, and electronic music.

His research interests include artificial intelligence, machine learning, and sound analysis, with a particular focus on the processing of biomedical and speech signals.

Daniele Casali

Daniele Casali received his master's degree in Electronic Engineering and his Ph.D. in Engineering of Sensors and Learning Systems from the University of Rome Tor Vergata.

He held fellowships with the Department of Electronic Engineering of the University of Rome Tor Vergata and chaired committees on digital sound processing and circuit theory.

His research interests include neural networks, machine learning, and sound analysis, with a particular focus on the processing of biomedical and speech signals.

Preface

Audio signal processing is an ever-growing field that is seeing a relevant improvement due to its potential for automation and remote analysis. Simultaneously, new advancements in AI and new requirements born from the diffusion of concepts such as the IoT make audio signals a crucial vector of information.

Authors from various fields, especially those centered on computer science, AI, acoustics, and electronic engineering, have contributed to a diverse and comprehensive overview of the current developments in audio analysis, which include acoustic measurement techniques, pitch detection algorithms, and deep learning architectures for the extraction of information from audio signals, especially speech.

The Guest Editors would like to thank Dr. Valerio Cesarini for his expertise in the field and his assistance in managing this Reprint and the associated Special Issue.

Giovanni Costantini and Daniele Casali
Editors

applied sciences

MDPI

Article

Experimental Assessment of the Acoustic Performance of Nozzles Designed for Clean Agent Fire Suppression

Marco Strianese [1], Nicolò Torricelli [1,2], Luca Tarozzi [2] and Paolo E. Santangelo [1,3,*]

[1] Dipartimento di Scienze e Metodi dell'Ingegneria, Università degli Studi di Modena e Reggio Emilia, 42122 Reggio Emilia, Italy
[2] Bettati Antincendio S.r.l., 42124 Reggio Emilia, Italy
[3] Centro Interdipartimentale per la Ricerca InterMech—MO.RE., 41125 Modena, Italy
* Correspondence: paoloemilio.santangelo@unimore.it; Tel.: +39-0522-52-2223

Abstract: Discharge through nozzles used in gas-based fire protection of data centers may generate noise that causes the performance of hard drives to decay considerably; silent nozzles are employed to limit this harmful effect. This work focuses on proposing an experimental methodology to assess the impact of sound emitted by gaseous jets by comparing various nozzles under several operating conditions, together with relating that impact to design parameters. A setup was developed and repeatability of the experiments was evaluated; standard and silent nozzles were tested regarding the discharge of inert gases and halocarbon compounds. The ability of silent nozzles to contain the emitted noise—generally below the 110 dB reference threshold—was proven effective; a relationship between Reynolds number and peak noise level is suggested to support the reported increase in noise maxima as released flow rate increases. Hard drives with lower speed were the most affected. Spectral analysis was conducted, with sound at the higher frequency range causing performance decay even if lower than the acknowledged threshold. Independence of emitted noise from the selected clean agent was also observed in terms of released volumetric flow rate, yet the denser the fluid, the lower the generated noise under the same released mass flow rate.

Keywords: fire protection equipment; acoustic nozzle; inert gas; halocarbon compound; sound pressure level

Citation: Strianese, M.; Torricelli, N.; Tarozzi, L.; Santangelo, P.E. Experimental Assessment of the Acoustic Performance of Nozzles Designed for Clean Agent Fire Suppression. *Appl. Sci.* **2023**, *13*, 186. https://doi.org/10.3390/app13010186

Academic Editors: Giovanni Costantini and Daniele Casali

Received: 29 November 2022
Revised: 18 December 2022
Accepted: 20 December 2022
Published: 23 December 2022

1. Introduction

Several applications require gas-based fire protection systems, as the systems involving the discharge of liquid water—mainly those consisting of sprinklers—may cause damage to items that could even exceed the loss induced by the fire itself. One of the main examples, and also the reference case for the present work, is embodied by the protection of electrical and electronic equipment: any scenarios where an electrical voltage is applied (e.g., data centers) discourages the use of liquid water to perform fire suppression and extinction, since water exhibits high electrical conductivity [1,2]. Other typical scenarios consist of archives, libraries and generally all locations where the mentioned materials (e.g., paper) may be damaged by liquid water or where rapid cleanup after operating the system is recommended [1]. Prior to their 1994 ban as a result of the Montreal Protocol to protect the ozone layer [1], followed by a phase-out stage, halocarbon compounds, typically known as halons, were employed as gaseous agents, with Halon 1301 arguably being the most common. Their extinguishing action is primarily based on inhibiting the combustion reaction: the halogen atoms (i.e., bromine, chlorine or fluorine that substitute hydrogen within a compound derived from a hydrocarbon) react with chemicals involved in the combustion process, thus breaking its chain reaction [1]. Several substances are currently being used as halon replacements and are overall defined as clean agents [2]; they consist of either halocarbon compounds (e.g., the fluorinated ketone FK-5-1-12, also known as Novec, and employed in the present work) or inert gases (e.g., argon and nitrogen, also tested in

the present research). The involved systems are total flooding: the whole compartment that is on fire is filled with the released gaseous agents. So, both categories mainly rely on oxygen depletion as their main extinguishing mechanism [2]; however, the former also tends to combine chemical action (i.e., inhibition of combustion) with heat extraction and, consequently, flame cooling towards extinction.

In spite of the numerous advantages exhibited by clean agents, the main form of which is characterized as being electrically non-conductive, leaving no residue after discharge and having virtually null Ozone Depletion Potential (ODP), the whole related system may be somewhat complex, since they require a storage capacity larger than that of halon-based systems as a result of poorer extinction effectiveness [2]. Most clean agent fire suppression systems include a storage tank, where the agent is superpressurized—usually by nitrogen—if in the form of a liquified gas, as is the case for most halocarbon compounds, or is simply contained at high pressure (usually in the range of 150–300 bar) in the case of an inert gas system [2,3]; valves, piping, nozzles and controllers (e.g., flow rate and pressure) are also part of the design. While the mechanical strength of pipes and nozzles to withstand the gas pressure does not currently appear a major concern for designers, the noise generated by the agent, mostly at the nozzle exit and, to a lesser extent, through valves and channels, represents a potentially relevant cause of damage to the equipment in the compartment, as well as first responders (e.g., firefighters) if present within the compartment as the discharge occurs [4]. A well-known problem and subject of extensive research efforts in aeronautics [5], the high-speed jet noise yielded by turbulence intensity at the nozzle outlet, and often emphasized by transitioning from subsonic to supersonic flow [6,7], has been studied for decades, mainly with the aim of reducing its intensity. In the case of clean agent gaseous jets released in a generic compartment undergoing a fire event, the sensitivity of Hard Disk Drives (HDDs) to noise can result in a remarkable performance loss [4,8]. As a quantitative experimental result, Sound Pressure Level (SPL) in the range of 110–130 dB is commonly accepted as the threshold beyond which HDD read/write performance becomes dramatically penalized [4,9,10]. Interestingly, HDD performance loss due to the overpressure and the steep pressure gradient caused by rapid discharge within the compartment appears somewhat irrelevant [4]. In spite of the increasing popularity of new technologies (e.g., Solid-State Disks—SDDs), which are not noise-sensitive, this problem may have a significant extent: in 2021, there were about 8000 data centers in the 110 countries providing information, with a predicted 150-fold increase in the generation of new data over the 2010–2025 timespan [11]. Moreover, a similar detrimental effect of noise yielded by the same gaseous jets can likely occur in some medical devices featuring an architecture similar to that of HDDs, an issue worth considering in fire protection systems designed for the healthcare industry.

As a technical solution to reduce the noise generated by clean agent discharge, silent nozzles—often also referred to as acoustic nozzles—have been developed. Since modifying the nozzle outlet does not appear to fully address the challenge, given that SPL was proven to be largely independent of nozzle shape [7], adding sound absorptive layers to the nozzle outer surface has become the most employed approach [3,12,13], with the insertion of horizontal plates also being recommended to make the released flow rate and pressure as balanced as possible in multiorifice nozzles [12,14]. Currently, the open literature presents relatively few studies that focus on the design of such nozzles and on assessing their acoustic performance, especially when compared with that exhibited by standard nozzles under the same discharge conditions. In that regard, it is worth mentioning that Koushik et al. [15] proposed an experimental approach to evaluate noise emitted by standard fire suppression nozzles releasing inert gases: an array of microphones were employed to measure SPL at various azimuthal locations with a constant distance (1 m) from the nozzle. Some degree of directionality (i.e., dependence on the angular coordinate) was found, together with potential impact of walls on the acoustic path (i.e., reflection and direct path). The extensive use of numerical modeling generally permeates the works on silent nozzles, most of which [12,14] are focused on determining the most effective nozzle

selection and configuration within an enclosure towards making SPL at the HDD locations lower than the previously mentioned threshold (110–120 dB, in those studies). On the other hand, the more recent contributions by Kim et al. [3,13] present a computational approach—remarkably validated through dedicated experiments—to optimize some design parameters of silent nozzles. Guided by DOE (Design of Experiments), their research effort led to them identifying the diameter of the external sound-absorbing shell as the most impactful variable. It is worth noting that some of the reviewed works suggest a certain dependence of the emitted noise on the orientation of the nozzle and the location at which SPL is evaluated [14,15], whether the nozzle is a standard or a silent one; on the other hand, other studies appear to practically consider the nozzle as a point source [3,12]. This aspect may be mostly related to the nozzle geometry.

Even though some comparison between standard and silent nozzles is presented in the work by Loureiro et al. [12], it appears that a quantitative and comprehensive approach to the purpose represents a challenge still to be addressed, at least in the open literature. More specifically, an experimental methodology allowing investigators to carry out such a comparison in terms of variables of interest for fire protection system designers (e.g., type of clean agent, released flow rate, orifice diameter) is still in demand. The present research is aimed at embarking on this quest, with the experimental datasets and facilities provided and described in the studies by Kim et al. [3] and Loureiro et al. [12] serving as a foundation. Arguably an unprecedented effort, this work may set a standard methodology to quantitatively compare the acoustic performance of various standard and silent nozzles, also leading to discussion in reference to the relevant physical quantities. Both the proposed approach and the obtained results could be appealing to designers and researchers focusing on clean agent fire suppression.

2. Materials and Methods

As remarked in the introduction (Section 1), no standard procedure is currently available for testing the acoustic performance of nozzles used in gas-based fire suppression. However, some previous works [3,12] present the description of experimental rigs; moreover, some technical standards are also available on fire protection of data centers and electrical devices [16], together with technical reports about tests to evaluate the harmful effect of noise—notably that emitted by gaseous jets released through a nozzle—on HDDs [17–20]. Thus, both the setup and procedure developed and proposed here were inspired by these sources of information.

2.1. Experimental Setup and Procedure

The experimental setup was installed in a large enclosure (25 $\times$ 10 $\times$ 5 m, length $\times$ width $\times$ height); a sketch of the whole assembly is presented in Figure 1. Aiming at evaluating acoustic performance of the nozzles against both categories of clean agents, the rig was designed to accommodate the storage and the supply system of both nitrogen IG100, employed as representative of the inert gases, and FK-5-1-12, employed as representative of halocarbon compounds. In fact, this selection required us to assemble two facilities, due to the different nature and properties of the two involved substances: the first rig was used to discharge a plain inert gas (i.e., nitrogen IG100), as shown in Figure 1a; the second rig was devised to discharge a multiphase, bicomponent mixture (i.e., FK-5-1-12 and nitrogen), as shown in Figure 1b. Since FK-5-1-12 is stored as a liquified gas, nitrogen was required to pressurize it within its storage tank (Section 1).

As demonstrated in Figure 1, the difference between the two testing facilities lies in the supply system (i.e., storage and piping), which required setting different lengths of the pipes and selecting components made of different materials.

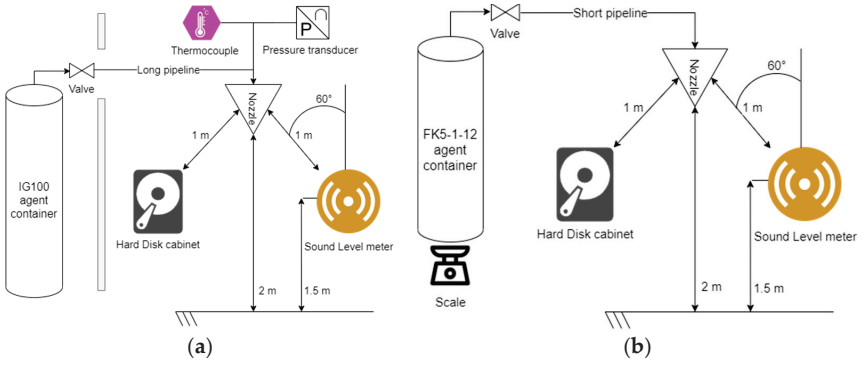


Figure 1. Schematic of the realized setup to evaluate acoustic performance by supplying (**a**) nitrogen IG100 or (**b**) FK-5-1-12.

Notably, the first setup (Figure 1a, used for the plain inert gas) featured a longer pipeline (10 m) and a constant flow valve manufactured by Bettati Antincendio S.r.l. (Reggio Emilia, Italy) to allow the regulation of the supplied flow rate. The longer length of the pipes were designed to accommodate the storage container and the related valve in a separate room, hence the wall sketched in Figure 1a, since the employed valve represents an additional source of noise, which would have biased the comparison between the inert gas and the halocarbon compound configurations, if not conveniently offset. As also included in Figure 1a, the facility for inert gas discharge was provided with a thermocouple and a pressure transducer, inserted within the pipeline at 200 mm distance from the nozzle (i.e., about 5–10 times the maximum diameter of the orifices used throughout the experiments [21]). These instruments were required to measure temperature and pressure of the gas stream as an input to ultimately calculate the released mass flow rate through the relationships reported in Section 2.2. On the other hand, these probes were not included within the facility for FK-5-1-12 discharge (Figure 1b), since their multiphase nature does not allow an evaluation of the supplied flow rate by any suitable model. Therefore, mass was recorded using a scale directly placed under the storage cylinder, then reconstructing mass flow rate through a finite difference approximation, also described in Section 2.2. The facility for FK-5-1-12 discharge featured a shorter pipeline (1.5 m) connecting the container to the nozzle, as the valve employed in this case, also manufactured by Bettati Antincendio S.r.l., did not contribute significantly to noise generation.

It is worth clarifying that the temperature and pressure of the substances contained in the storage cylinders were monitored in both facilities, as well as directly downstream of the valve: in the former case, the tank included a pressure gauge and a thermocouple, whereas in the latter, another pressure transducer and thermocouple were inserted. The tank used for hosting nitrogen IG100 featured 80 L capacity, with gas being stored at 300 bar pressure; the valve allowed regulating outlet pressure to impose 70 bar maximum pressure in the manifold. FK-5-1-12 was stored in a 14 L tank instead, with a 1 kg/L filling ratio (i.e., 1 kg of clean agent per 1 L storage capacity). As typical in the storage of liquified gases, the agent was superpressurized at 70 bar by nitrogen; the employed valve was specifically developed to issue a flow at 42–70 bar. A full view of the setup is shown in the photo of Figure 2, which combines and includes all the items of both facilities.

The nozzle was placed at the center of the base of the chamber, as far away from the walls as possible. This configuration—applied within a large enclosure—was aimed at reducing the effect of reverberation by walls, which was assessed as potentially significant [15], since the tests were not conducted in an anechoic chamber.

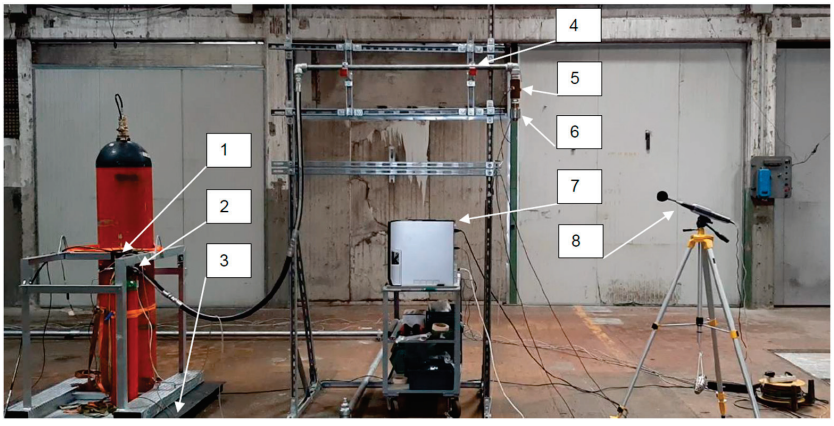
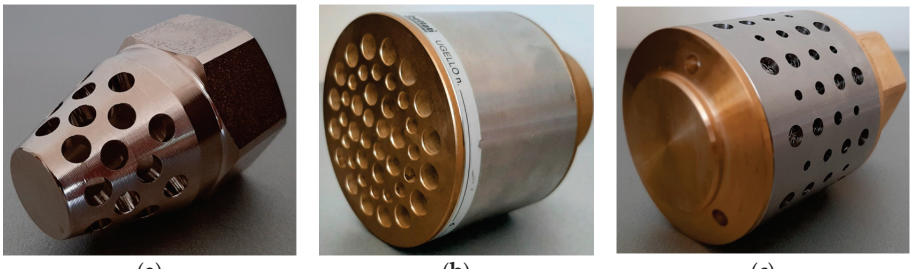


Figure 2. Photo of the experimental setup: (**1**) storage cylinder; (**2**) pressure transducer and thermocouple (downstream of the valve); (**3**) scale; (**4**) pipeline; (**5**) pressure transducer and thermocouple (upstream of the nozzle, only for inert gas discharge system); (**6**) nozzle; (**7**) servers, including one SSD and two HDD; (**8**) sound level meter.

The nozzle was also placed at 2 m height from the floor (Figure 1), a common value in installations within data centers [16] and suitable for accommodating all the components surrounding it. As for nozzle types, the tested ones are all manufactured by Bettati Antincendio S.r.l.: Figure 3 presents photos of both the employed standard nozzle (Figure 3a) and silent ones designed for each category of clean agents (Figure 3b,c).



(**a**) (**b**) (**c**)

Figure 3. Photos of the nozzles employed in the experiments: (**a**) standard; (**b**) silent nozzle for the discharge of inert gases; (**c**) silent nozzle for the discharge of halocarbon compounds (low ceiling height).

Interestingly, some preliminary tests made select a silent nozzle for low ceiling installations (Figure 3b) when the discharge of halocarbon compounds (i.e., the FK-5-1-12) was involved, since it proved more effective, especially if combined with the chosen valve. On the other hand, silent nozzles were employed in the discharge of nitrogen IG100. It is worth clarifying that the tested silent nozzles present a sound-absorbent layer included within their frame, which makes them similar to those investigated in the works by Kim et al. [3,13]. Figure 4 presents simplified technical sketches of the employed nozzles, highlighting the orifice diameter.

As shown in Figure 4b,c, silent nozzles are endowed with the mentioned sound-absorbing shell. Orifice diameter varied over the following values in the whole series of tests and for both standard and silent nozzles: 5, 6, 10, 14, 17 and 23 mm. It is worth clarifying that the experimental setup proposed in the present work is aimed at comparing standard with silent nozzles in a single-nozzle configuration. Therefore, the combined action of the

discharge by more nozzles at the same time, and its effect on noise, is not included; notably, superposition of waves and interference of sound may make the generated noise vary from that produced by a single nozzle throughout its discharge.

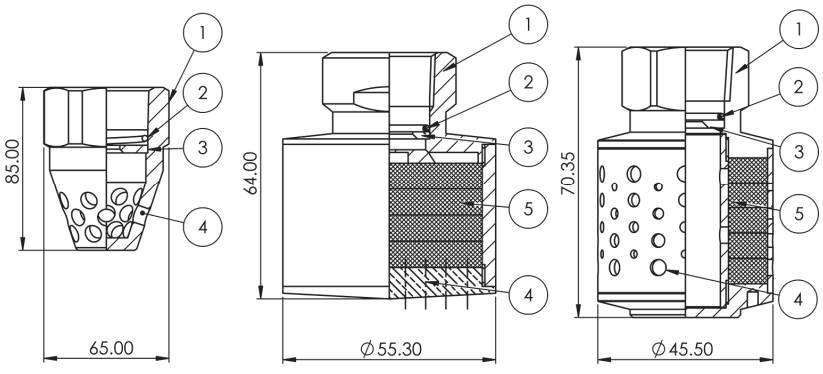


Figure 4. Technical sketch of the employed nozzles, with values provided referring to the orifice diameter: (**a**) standard; (**b**) silent; (**c**) silent for low ceiling applications; the marked components are (**1**) nozzle body; (**2**) spring; (**3**) orifice; (**4**) holes; (**5**) sound-absorbing layer.

A server rack containing an SSD and two HDDs with different characteristics was located at 1 m distance from the nozzle outlet (Figure 1) to assess the impact of the emitted noise on electronic equipment typical of data centers. The set distance is consistent with that applied in previous works [12,15] for evaluating SPL from nozzles used in gas-based fire suppression.

Both HDD and SSD performance was evaluated by recording their speed over time using a free software (*HD Speed* version 1.7, released by SteelBytes). The acoustic emissions from the tested nozzles were measured by a sound level meter, with its microphone positioned at 1 m distance from the nozzle exit and at 1.5 m from the floor (Figure 1); the instrument (specifically, its axis of symmetry) was inclined by about 60 °C with respect to the vertical axis, coincident with the axis of symmetry of the nozzle. The distance between the sound level meter and the nozzle is consistent with the value set for the server rack, and also supported by previous studies [12,15] and recommendations from test reports [17–20], since it is representative of the typical minimum distance at which obstructions (e.g., a cabinet containing electronic equipment) are located.

The measuring instruments employed in the experiments and their characteristics (e.g., acquisition frequency, accuracy) are summarized in Table 1; the characteristics of the used hard drives—both HDD and SDD—are reported in Table 2, with specific focus on rate of rotation (HDD only) and nominal speed.

Table 1. Employed measuring instruments and relevant characteristics.

Measured Parameter	Acquisition Frequency (Hz)	Instrument/ Sensor	Data Acquisition Board
Storage mass (cylinder + clean agent)	10	Scale *P1250S5* by LAUMAS Elettronica, 1500 kg F.S.	*NI 9208* board by National Instruments, 16 channels (current)
Pressure	10	Pressure transducer *21y* by Keller, 400 bar F.S. ± 0.1 bar accuracy	

Table 1. *Cont.*

Measured Parameter	Acquisition Frequency (Hz)	Instrument/ Sensor	Data Acquisition Board
Temperature	10	Thermocouple T type, 1.0 mm bead diameter, accuracy in accordance with standard IEC 60584	*NI 9212* board by National Instruments, 8 channels (thermocouple)
SPL	1	Sound level meter *HD2010UC/A* by Deltaohm, with spectral analysis by octave bands from 31.5 Hz to 8 kHz	Data stored on the instrument
Hard drive performance	2	–	Data recorded by *HD Speed* software, version 1.7

Table 2. Employed hard drives and relevant characteristics (*, assessed in a quiet room using *HDSpeed* software).

Reference Name	Manufacturer	Model	Type	Rate of Rotation (rpm)	Nominal Speed * (MB/s)
SSD	Toshiba	*OCZ-VERT EX3 MI SCSI*	SSD	–	229
HDD1	Hitachi	*DeskStar 7K160*	HDD	7200	77
HDD2	Western Digital	*WD36 ADFD*	HDD	10,000	89

The tested configurations are reported in Table 3. Notably, the performed experiments can be subdivided into three types, all of which were conducted on both standard and silent nozzles:

- The first set was aimed at assessing the reliability of the experimental setup and measurements—mostly through statistical analysis—together with evaluating the potential directionality of SPL; these tests were carried out employing an inert gas mixture (i.e., IG55, 50% argon and 50% nitrogen) with a fixed nozzle diameter (5 mm) and pressure at the nozzle outlet; measurements were taken at various values of the distance between the nozzle and the sound level meter, also varying the angle of inclination of the sound level meter with respect to the nozzle.
- The second set was aimed at investigating the discharge of inert gases (nitrogen IG100, in the present case); these tests were carried out at varied orifice diameters, between 5 and 23 mm, while keeping the distance between the nozzle and the sound level meter constant and equal to 1 m (Figure 1).
- The third set was aimed at investigating the discharge of halocarbon compounds (FK-5-1-12, in the present case); these tests were carried out only using nozzles with 6 mm orifice diameter, given the higher cost of the involved substance; the distance between the nozzle and the sound level meter was also kept constant and equal to 1 m (Figure 1).

At least three repeated tests were conducted for each configuration listed in Table 3 to acquire a statistically significant dataset.

Table 3. Detailed list of the investigated configurations (*, multiple data series recorded for each tested distance at various angular locations).

Set	Nozzle Type	Clean Agent	Nozzle-to-Sound Meter Level Distance (m)	Nozzle Diameter (mm)
I	Silent	IG55	2.5	5
	Silent	IG55	1.25	5
	Silent	IG55	5	5
	Silent	IG55	1–2–4–8 *	5
	Standard	IG55	1–2–4–8 *	5
II	Silent	IG100	1	5
	Standard	IG100	1	5
	Silent	IG100	1	10
	Standard	IG100	1	10
	Silent	IG100	1	14
	Standard	IG100	1	14
	Silent	IG100	1	17
	Standard	IG100	1	17
	Silent (low ceiling applications)	IG100	1	17
	Silent	IG100	1	23
	Standard	IG100	1	23
III	Standard	FK-5-1-12	1	6
	Silent (low ceiling applications)	FK-5-1-12	1	6

2.2. Data Processing and Analysis

The analysis of the acquired experimental dataset pursued two main purposes through dedicated approaches:

- Determining and comparing the acoustic performance of standard and silent nozzles; to this end, SPL is presented as a function of released flow rate and time for both tested clean agents and for the whole set of investigated orifice diameters; the results from spectral analysis by octave bands are also included.
- Evaluating the effect of SPL on the different tested hard drives and highlighting quantitatively the performance loss in the same configuration towards a comparison between standard and silent nozzles; to this end, an index of the disk behavior and performance through the discharge is proposed as inspired by a recent report [20], which produces measurements of read/write speed and includes that under the SPL due to white noise (i.e., baseline speed).

$$Disk\ performance\ index = \frac{Disk\ speed}{Disk\ baseline\ speed}. \quad (1)$$

Moreover, the reliability of the acquired dataset and the overall experimental approach was challenged by the following:

- Checking the applicability of the point source model [22], which supports the proposed design of the experimental setup, where SPL was measured at the location described in Section 2.1 and shown in Figure 1, without resorting to an array of microphones, as in [15]; this preliminary evaluation is included in Section 2.3.
- Evaluating standard deviation [23] of the acquired data—mainly the SPL dataset as a function of time or released flow rate over repeated tests under the same configuration—in order to assess repeatability.

As already mentioned in Section 2.1, the released flow rate was calculated for both the inert gas and the halocarbon compound discharge. Notably, a classic model was employed

to calculate the former, with the flow rate of a jet (compressible flow, assuming an ideal gas) issued through an orifice [24] being expressed by Equation (2):

$$\begin{aligned} \dot{Q}_m &= \gamma \cdot A_m \cdot \rho_{m,CR} \cdot w_{m,CR}, \text{ if } \beta_{noz} \leq \beta_{CR}, \\ \dot{Q}_m &= \gamma \cdot A_m \cdot \sqrt{\frac{2k}{k-1} \cdot p_0 \cdot \rho_0 \left(\beta_{noz}{}^{\frac{2}{k}} - \beta_{noz}{}^{\frac{k+1}{k}} \right)}, \text{ if } \beta_{noz} > \beta_{CR}, \end{aligned} \tag{2}$$

where $\dot{Q}_m$ is mass flow rate, γ is the discharge coefficient of the nozzle, A_m is the surface area of the outlet section, ρ is density, w is the average jet velocity, β is defined as the ratio between pressure downstream of the nozzle (i.e., atmospheric pressure) and upstream of the nozzle, $w_{m,CR}$ refers to the gas velocity at nozzle orifice, CR refers to critical value of air at 20 °C temperature and noz refers to nozzle. The following set of relationships allows the quantification of discharge coefficient, density and velocity, through the jet temperature:

$$\gamma = e^{k1} \cdot p_0^{k2} \cdot D_m^{k3}, \tag{3}$$

$$\rho_{m,CR} = \rho_0 \cdot \left(\frac{2}{k+1} \right)^{\frac{k}{k-1}}, \tag{4}$$

$$w_{m,CR} = \sqrt{k \cdot R_{gas} \cdot T_{m,CR}}, \tag{5}$$

$$T_{m,CR} = T_0 \frac{2}{k+1}, \tag{6}$$

where e is the Euler's number, p is gas pressure, D_m is the orifice outlet diameter, 0 refers to the conditions upstream of the nozzle, k is defined as the ratio between specific heat capacity at constant pressure and specific heat capacity at constant volume of the involved gas, R_{gas} is the specific gas constant, T_m is jet temperature, T is temperature and $k1$, $k2$ and $k3$ are coefficients specific to the nozzle. Density values were taken from the *CoolProp* free library [25] by selecting that of nitrogen corresponding to temperature and pressure measured upstream of the nozzle (Section 2.1, Figure 1a). The $k1$, $k2$ and $k3$ coefficients were experimentally quantified by VdS Schadenverhütung GmbH for the selected nozzles, in order to also evaluate their outflow area.

The evaluation of mass flow rate in the case of the FK-5-1-12 discharge was performed by processing the readings of the employed scale (Section 2.1, Figure 1b). As previously mentioned, the derivative of mass with respect to time was calculated by the approximation of the differential to finite difference:

$$\dot{Q}_m = \frac{dm}{dt} \approx \frac{\Delta m}{\Delta t} = \frac{m_i - m_{i+1}}{\frac{1}{f}}, \tag{7}$$

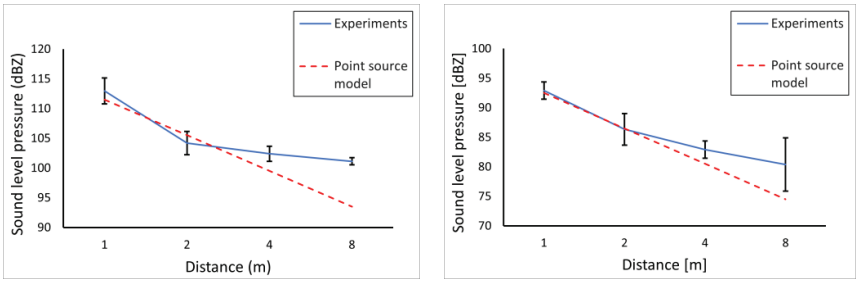
where m is mass, t is time, f is acquisition frequency and i is the index referring to the generic ith acquisition ($i = 0, \ldots\ n$, where n is the last but one acquisition). This approach is relatively common to extrapolate mass flow rate from mass readings [26–28]; no moving average was applied in this calculation, since the acquired mass trend proved relatively bias-free in terms of statistical noise.

2.3. Applicability of Point Source Model and Statistical Analysis

The point source model consists of assuming that sound radiates equally and symmetrically in all directions from the emitting source; as the underlying hypothesis, the source is far smaller than the wavelength associated with the emitted sound [22]. This spherical symmetry implies that sound intensity $I = W/4\pi r^2$, where W is the power in the wave and r is the radius of the generic sphere, decreases according to the classic inverse square law (i.e., $\sim 1/r^2$) as distance from the source increases. Sound pressure follows an inversely proportional law (i.e., $\sim 1/r$) instead; SPL expresses sound pressure relative to a reference value through a logarithmic function (i.e., $SPL = 20 \log(p/p_0)$, where p is

sound pressure, 0 refers to the reference value, and the unit for SPL is dB). Combining the previously reviewed relationships, a quantitative result from the point source model consists of SPL decreasing by about 6 dB each time the distance from the emitting source is doubled.

The applicability of the point source model to the present case was challenged by a preliminary set of tests (i.e., the first set described in Section 2.1 and reported in Table 3). The scope was to evaluate the agreement between theory and experimental results, mainly by assessing the distance from the nozzle to the sound level meter at which sound absorption and reflection from walls make the approximation inevitably fail, and the degree of directionality. This latter parameter was investigated by varying the angle of inclination of the sound level meter with respect to the nozzle (i.e., the source) over several values in the 0–90 °C range, thus following the approach by Koushik et al. [15]. As reported in Table 3, the distance between source and instrument was also varied at each testing location over the angular coordinate. Figure 5 shows the SPL dataset acquired at various angular locations at the selected values of the distance (1, 2, 4 and 8 m from the nozzle).



(**a**) (**b**)

Figure 5. Comparison experimental results and point source model at various values of nozzle-to-sound level meter distance for (**a**) standard and (**b**) silent nozzles; data points represent mean SPL value over the dataset acquired at the same distance and varying the angle of inclination; the error bars represent doubled standard deviation.

Notably, the data points represent the mean SPL value over the dataset collected at various angular locations at the same distance, while the error bar represents the doubled standard deviation calculated over the same population [23].

The SPL trend predicted by the point source model is also included. As demonstrated by the comparison between predicted and experimental profiles, the applicability of the model to the proposed experiment appears firmly supported within a 2 m diameter sphere centered at the nozzle as the source, for both standard (Figure 5a) and silent (Figure 5b) nozzles. On the other hand, as the distance is further increased, reverberation and absorption by walls appear to make the assumption of point source increasingly fail. As previously noted, point source theory relies on the complete absence of directionality, which is in fact somewhat present for both the standard (Figure 5a) and the silent nozzle (Figure 5b). However, the variability over the investigated angular locations at 1 and 2 m distance from the nozzle (i.e., those relatively free from the wall effect) is limited to $\pm$2.5 dB: even though quantitatively not negligible, it was deemed small enough to allow the point source model to be reasonably applicable, even in light of a threshold distinguishing harmless from harmful noise to HDDs with a 20 dB span (110–130 dB, Section 1). Thus, the proposed experimental approach and setup (Section 2.1, Figure 1), where the sound pressure was evaluated at 1 m distance from the nozzle and hard drives were placed at the same distance to assess the impact of noise on their performance, appears overall supported for the nozzles tested in the present study.

As reported in Section 2.2, error analysis was also carried out to assess the repeatability of the experiments. This evaluation was also conducted through the first set of tests (Table 3); notably, specific experiments were performed at fixed pressure against both standard and silent nozzles, where 10 s sampling time was applied to SPL measurements, varying the distance from the nozzle. Figure 6 shows the dataset acquired at 1, 2 and 4 m distance from the nozzle over the 10 s acquisition (overall, five series of acquisitions); as a representative example, the tests conducted with the standard nozzle are considered in the graphic.

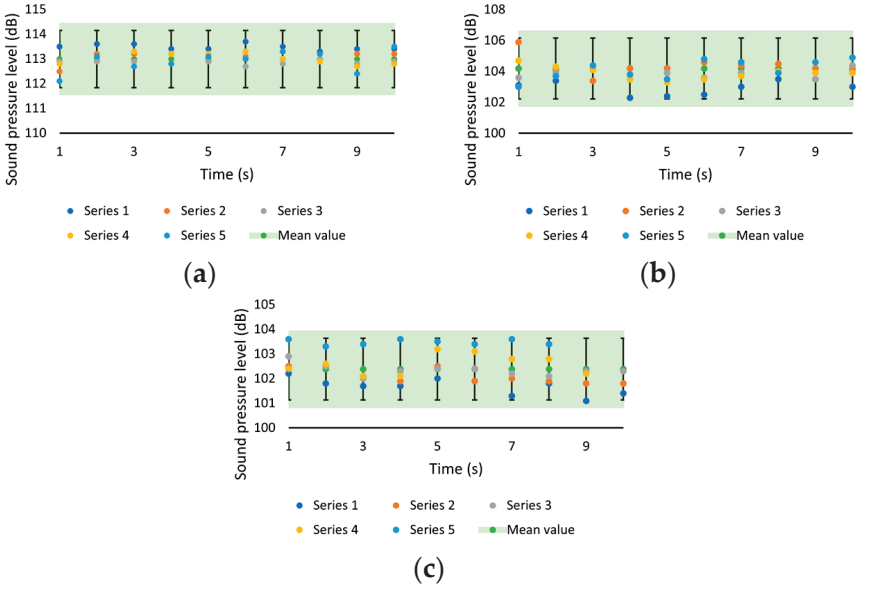


Figure 6. SPL dataset to assess the repeatability of the experiments for the standard nozzle, with data points collected at (**a**) 1, (**b**) 2 and (**c**) 4 m distance from the nozzle to the sound level meter.

The plot includes individual data points and mean values calculated at each acquisition, together with a band, the size of which is three times the maximum standard deviation calculated over the whole set of tests. All the data points lie within the highlighted band, which exhibits relatively high repeatability and low random error (i.e., high precision). From a quantitative standpoint, the standard deviation calculated for the dataset shown in Figure 6 is in the order of 0.5–1 dB, which amounts to less than 1% of the reading. As mentioned in Section 2.1, the first set of tests specifically served as a source of data for statistical analysis. However, the assessment of variance should be taken as representative of repeatability for the whole set of experiments, since no significant difference from the value previously reported arose when performing similar analysis against the dataset from the second and the third set of tests.

3. Results and Discussion

In this section, the outcomes from the experiments conducted through the second and the third set of tests (Table 3) are presented and discussed, mostly focusing on the comparison between standard and silent nozzle in terms of acoustic performance and detrimental effect on the read/write speed of the selected hard drives. An evaluation of the impact of design parameters such as orifice diameter, released clean agent and released mass flow rate also stemmed from this comparison.

3.1. Discharge of Inert Gases: Effect of Generated Noise

The diagrams included in Figure 7 show the effect of the noise generated by standard and silent nozzles with various orifice diameters on both HDDs and SDDs, throughout a full discharge of nitrogen IG100. Notably, selected experiments using nozzles of 5, 10, 14, 17 and 23 mm diameter are presented; the results from the tests with the silent nozzles for low ceiling applications are omitted, as an approximately similar behavior was found between them and the other silent nozzles with the same orifice diameter. Overall, the experimental results demonstrate the ability of silent nozzles to bring SPL down to relatively harmless values through the discharge: standard nozzles present SPLs greater than 110 dB—the lower limit of the damaging threshold (Section 1)—over most of the gas release, and in most cases (i.e., for orifice diameter greater than 5 mm), peaks exceeding 120 dB occur; on the other hand, the SPL of silent nozzles exceeds the threshold only for about 25 s even with the larger orifice diameter (larger than 10 mm), also never reaching 120 dB. As a finding of general validity for both types of nozzle, SPL exhibits a sudden increase up to a local maximum upon starting the discharge, which is consistent with the theory associated with noise emitted by gaseous jets [5–7], reviewed in Section 1: since noise is largely due to turbulence effects, it may be interpreted as function of Reynolds number ($Re = w_{m,CR} D_m / v$, where v is kinematic viscosity of the involved substance); as *Re* increases, turbulence develops until critical *Re* is reached (i.e., fully developed turbulence) [29], to which SPL local peaks arguably correspond.

Instances of additional maxima occurring after the first peak appear mostly for the discharge through the standard nozzle (Figure 7a,c,d,e), and hint at some slight pressure variability imposed by the constant flow valve: constant flow valves are developed and employed to keep pressure and, consequently, flow rate as constant as possible throughout the discharge, which would virtually result in constant Re and SPL until pressure decays in the storage tank towards the end of the process. This practically translates into a plateau-like behavior, as apparent in Figure 7f,g,h, with the potential singularities previously mentioned occurring in some cases. The proposed discussion also substantiates another observation holding for both standard and silent nozzles (Figure 7): the larger orifice diameter, the higher the SPL maxima and overall SPL values throughout its trend.

Assuming constant pressure (i.e., 70 bar, as mentioned in Section 2.1), at least throughout the controlled phase of the discharge, results in almost constant average velocity, even with a Bernoulli inviscid model [30]; as orifice diameter increases, both *Re* and released mass flow rate increase. Therefore, it can be inferred that the discharge time obviously decreases, while SPL increases, as higher flow rate is issued, providing that supply pressure is kept constant.

An insight into the effect of the sound frequency stems from maps presenting SPL as a function of time and frequency throughout the discharge; Figure 8 shows experiments conducted for both standard and silent nozzles with 10, 17 and 23 mm orifice diameter as the most representative. It is interesting to note that SPL reaches higher values (i.e., greater than 110 dB) within the 1–8 kHz range, which is somewhat lower than the findings from Koushik et al. [15] on standard nozzles (i.e., more in the 10–20 kHz range). This suggests that the employed nozzles emit noise at lower frequency, hence higher wavelength, which ultimately yields more reliable applicability of the point source model (Section 2.3) as a result of the less remarkable directionality found with respect to the noise emitted by the nozzle used in [15]. Moreover, SPL appears higher than 100 dB over periods of time when frequency is greater than 4 kHz. According to Dutta [31], noise can affect hard drive performance even beyond a 105 dB threshold at a frequency between 4 and 10 kHz.

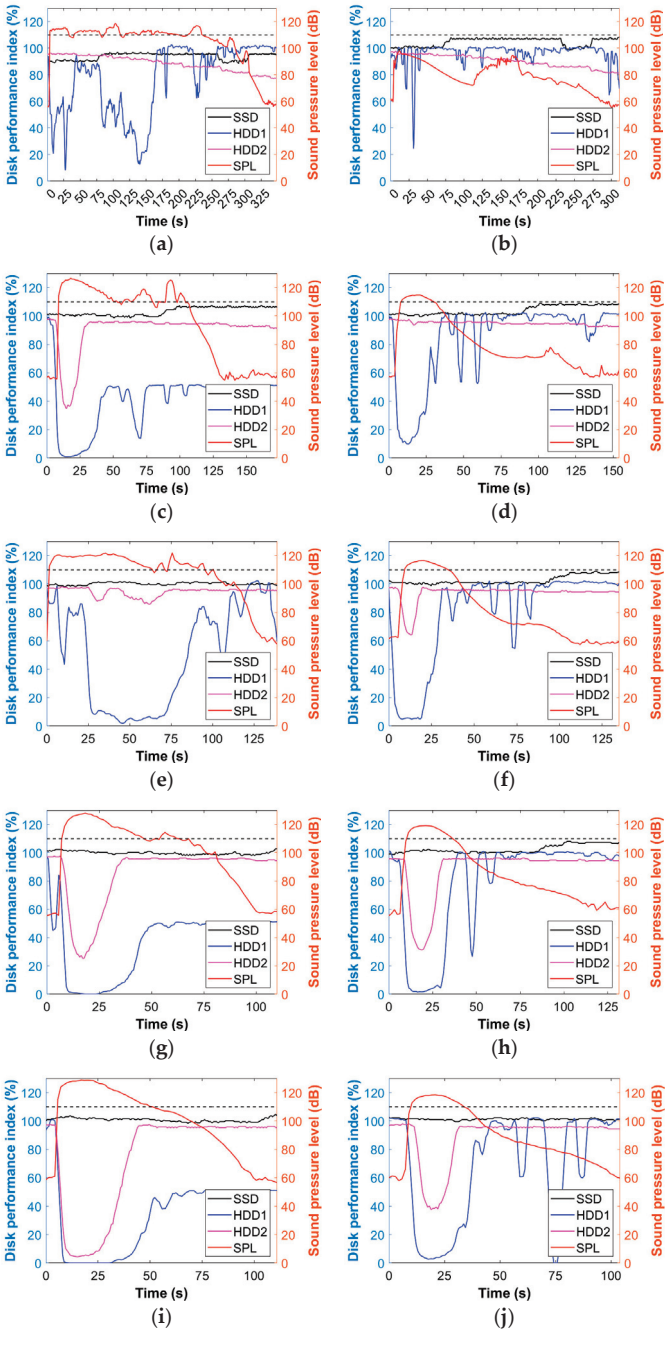


Figure 7. SPL and disk performance index as a function of time throughout a nitrogen IG100 discharge by (**a**) standard and (**b**) silent nozzle with 5 mm orifice diameter; (**c**) standard and (**d**) silent nozzle with 10 mm orifice diameter; (**e**) standard and (**f**) silent nozzle with 14 mm orifice diameter; (**g**) standard and (**h**) silent nozzle with 17 mm orifice diameter; (**i**) standard and (**j**) silent nozzle with 23 mm orifice diameter.

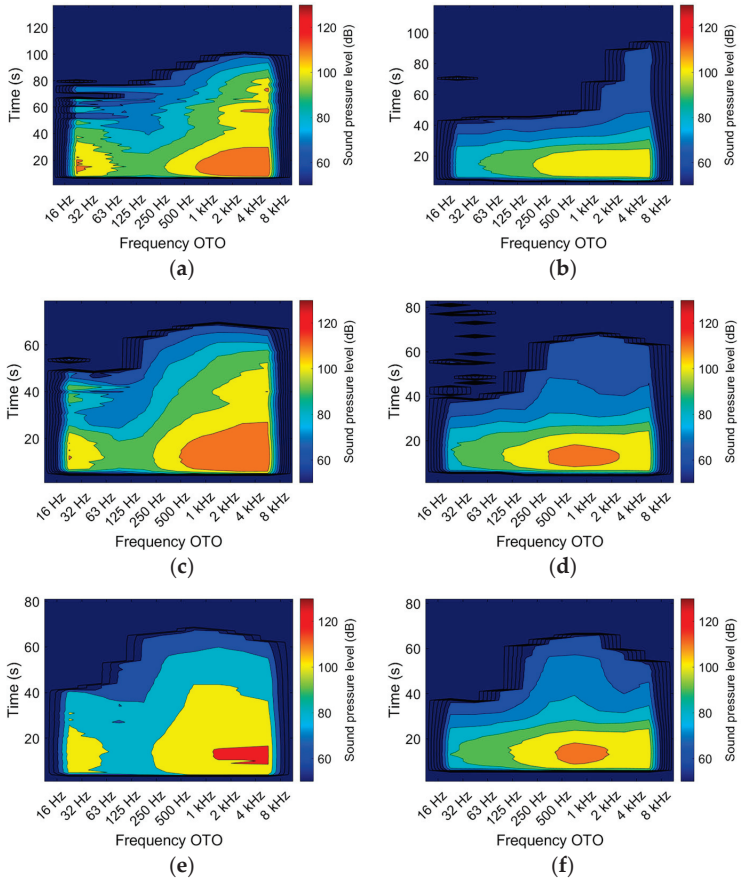


Figure 8. SPL as a function of time and OTO (One-Third Octave bands) frequency throughout a nitrogen IG100 discharge by (**a**) standard and (**b**) silent nozzle with 10 mm orifice diameter; (**c**) standard and (**d**) silent nozzle with 17 mm orifice diameter; (**e**) standard and (**f**) silent nozzle with 23 mm orifice diameter.

When comparing hard drive performance, the difference between configurations with the standard nozzle and those with the silent ones is highlighted by HDD performance index trends. As expected, solid-state storage devices (i.e., SDD) are almost not affected by external noise, hence SDD performance was close to 100% throughout each conducted experiment (Figure 7). The exhibited mild oscillations—in some instances even leading to performance higher than the nominal one (i.e., greater than 100%)—appear unrelated to the actual experimental conditions. On the other hand, HDD performance decay is highly consistent with SPL exceeding the 110 dB threshold; moreover, the higher the SPL, the greater the performance decay (Figure 7). This observation is particularly emphasized by HDD1, which is the hard drive featuring the smaller rate of rotation, and consequently, the lower nominal speed (Table 2); HDD2 seemed less affected by noise, to the point of having its performance almost unaffected by SPL lower than 120 dB and also by noise regrowth for a short timespan (Figure 7c,g). It is also worth noting that HDD1 did not recover its performance level under undisturbed conditions as noise fell below the threshold, due to the discharge by the standard nozzle approaching the end of the process (Figure 7c,g,i); this phenomenon never appeared in tests run with silent nozzles instead. This prolonged performance loss occurring with the discharge produced by standard nozzles with large orifice diameters is arguably due to SPL being above 100 dB at frequencies greater than

4 kHz over a wide portion of the discharge event (Figure 8a,c,e). As previously reported, Dutta suggests that hard drives are affected by SPLs as low as 105 dB in the 4–10 kHz band [31], which may explain the result of only partial recovery even when noise fell below 110 dB. The diagram of Figure 9 serves as an overview of the impact of noise from the tested nozzles on hard drives: (maximum) performance loss is plotted against maximum SPL reached for each tested condition, including both standard and silent nozzle experiments. An apparent correlation between HDD read/write speed and noise appears from the performance dataset, with HDD performance becoming lower as SPL increases. The linear best-fitting curve, applied to each database related to the three tested hard drives, emphasizes that noise is more impactful against the HDD featuring lower nominal speed with respect to those with higher rates of rotation.

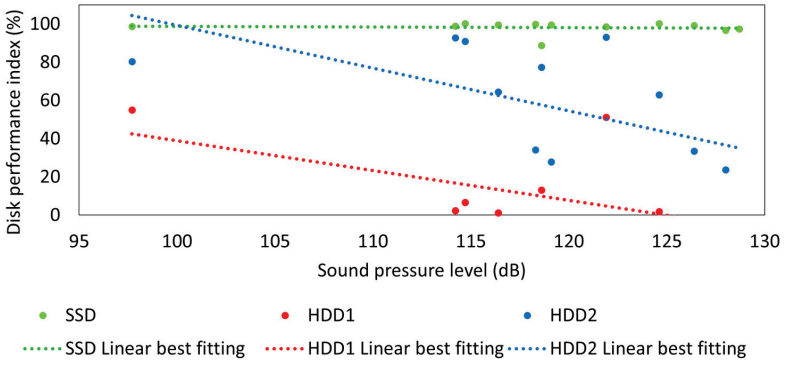


Figure 9. Performance index as a function of maximum SPL reached in every tested condition with nitrogen IG100.

3.2. Discharge of Halocarbon Compounds: Effect of Generated Noise

As reported in Sections 1 and 2.1, one of the scopes of the present work was comparing the discharge of the two types of clean agents in terms of emitted noise, which prompted us to test FK-5-1-12 as a representative of halocarbon compounds. The higher cost associated with these agents—also noted as a drawback by DiNenno [2]—led us to run experiments only against one configuration (Table 3), obviously through both standard and silent nozzles (for low ceiling applications). The trends of SPL and disk performance index are shown for representative tests in Figure 10. The overall consistency of these results with the dataset for the nitrogen IG100 discharge through nozzles of similar orifice size (5 mm, Figure 7a,b) seems to somewhat substantiate a statement provided in a well-recognized white paper [4]: *"The agent used in an extinguishing solution doesn't define per-se the noise level of the system overall"*. However, it is also worth noting that FK-5-1–12 mass flow rate released through a 6 mm diameter nozzle is remarkably larger than that of nitrogen IG100 through a 5 mm diameter nozzle; in fact, the former is close to nitrogen IG100 mass flow rate issued by a 23 mm diameter nozzle (Figure 7i,j), due to its higher density. Even though the released volumetric flow rate can be assumed to be very similar, it appears that almost the same noise is generated by a larger mass flow rate of FK-5-1-12 as that generated by a smaller mass flow rate of nitrogen IG100, when the discharge is operated at the same pressure.

The ability of silent nozzles to reduce noise down to harmless levels is also proven by both the profile of disk performance index (Figure 10) and by SPL presented as a function of frequency, as in the maps of Figure 11. Solid-state drives are obviously not affected, and neither are HDDs with high rates of rotation (i.e., HDD2) if orifice diameter is smaller than 10 mm, yet HDDs with lower speed (i.e., HDD1) exhibit a considerable decay in their performance if standard nozzles are used to release halocarbon compounds, as opposed to the discharge produced by silent ones. However, a remarkable effect can be observed in the results from tests with the silent nozzle (Figures 10b and 11b): SPL shows a sharp

decrease after the expected peak upon the onset of discharge (Section 3.1), then followed by another increase up to a plateau-like condition.

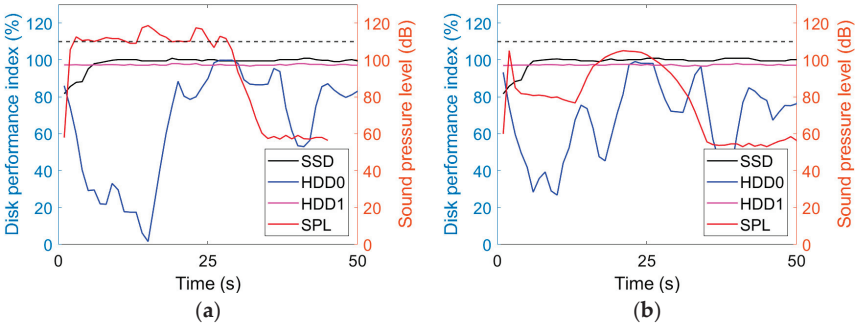


Figure 10. SPL and disk performance index as a function of time throughout FK-5-1-12 discharge by (**a**) standard and (**b**) silent nozzle for low ceiling applications with 6 mm orifice diameter.

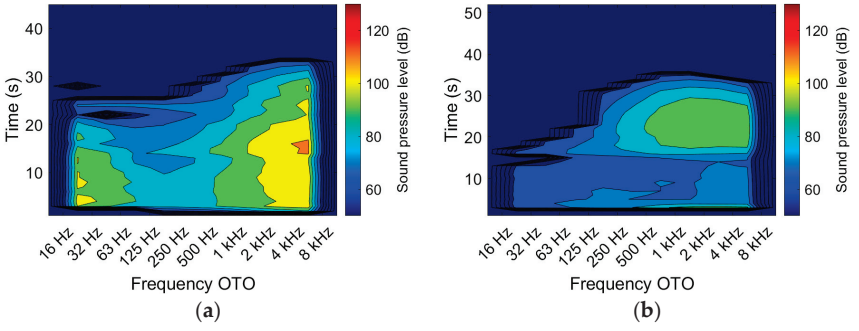


Figure 11. SPL as a function of time and OTO (One-Third Octave bands) frequency throughout FK-5-1-12 discharge by (**a**) standard and (**b**) silent nozzle for low ceiling applications with 6 mm orifice diameter.

This trend is arguably typical of the discharge of liquified gases such as FK-5-1-12, where another gas is also added to superpressurize the agent (i.e., nitrogen, Section 2.1). Firstly, the liquid phase (i.e., FK-5-1-12) is released; as the agent becomes almost exhausted, the discharge becomes nitrogen-laden. Thus, in this last part of the discharge, the Reynolds number changes as a result of a change in the released chemical substance, which reflects the involved thermophysical properties. The tested silent nozzles demonstrated a more effective damping action on the noise emitted through the actual FK-5-1-12 discharge (first phase), whereas the trend already observed in tests with nitrogen IG100 (e.g., Figure 7d,f,h,j) occurred once the gas used to superpressurize the agent was finally released (second phase). This outcome—also shown by the high-frequency (i.e., 1–4 kHz) SPL dataset of Figure 11b—suggests a certain dependence of the sound-absorbing performance on the clean agent type.

3.3. Overview of the Relationship between Flow Rate and Noise

Aiming at a deeper understanding of the effect of released flow rate on the noise emitted by nozzles used in gas-based fire suppression, the SPL dataset was related to mass flow rate calculated by the approach proposed in Section 2.2. The results are presented in Figure 12 for both standard and silent nozzles, those designed for low ceiling applications are included with regard to FK-5-1-12 discharge. As already apparent from the list of tested configurations (Table 3), most of the data are available for standard and silent nozzles with nitrogen IG100 discharge, whereas datasets for the other investigated conditions (i.e.,

all those involving FK-5-1-12 discharge) consist of a smaller population. Therefore, the degree of statistical significance varies between datasets. Data are plotted in Figure 12 with reference to the employed orifice diameter; the best-fitting curve was also generated as a power law ($y = ax^b$) and is also included in the plots. While formulating the fitting curve, SPL values below 60 dB were discarded, as they were considered almost white noise, and so were flow rate values below 0.15 kg/s, since measurements of the various involved parameters (e.g., pressure, temperature, mass) were deemed unreliable at the onset of the discharge as a result of the steep transient growth. The values of scaling factor a and exponent b are presented in Table 4 for each analyzed condition.

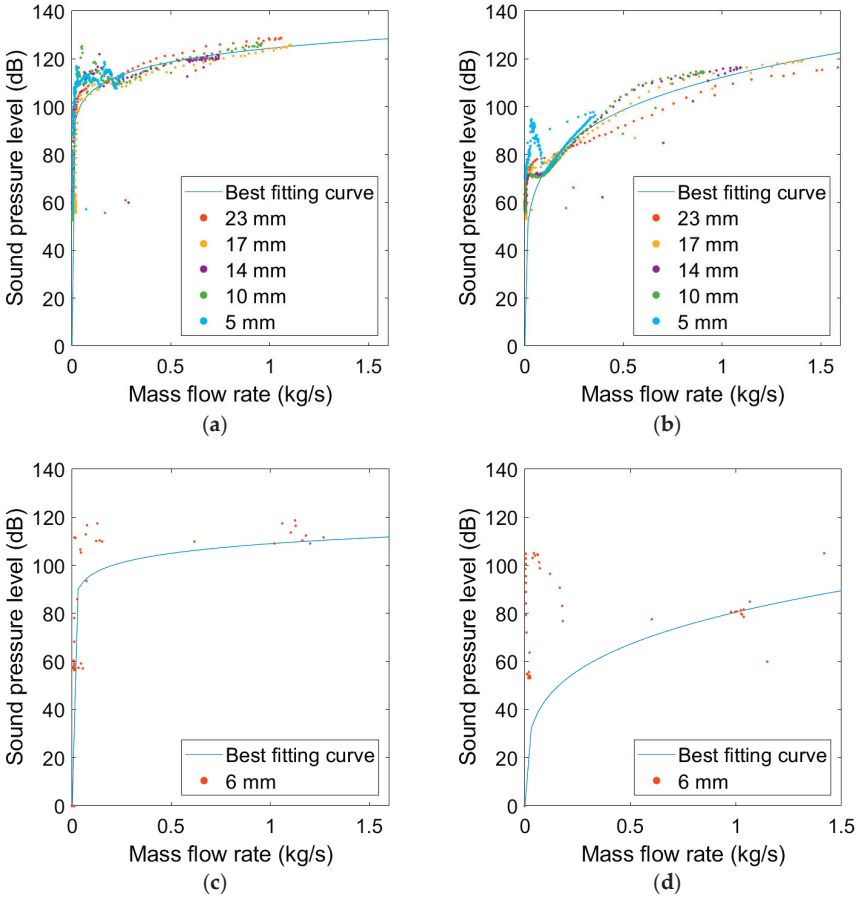


Figure 12. SPL as a function of released mass flow rate, with data points (referred to orifice diameter) and best-fitting curve: (**a**) standard nozzle and nitrogen IG100 discharge; (**b**) silent nozzle and nitrogen IG100 discharge; (**c**) standard nozzle and FK-5-1-12 discharge; (**d**) silent nozzle for low ceiling applications and FK-5-1-12 discharge.

The noise-damping effect of silent nozzles is evident in the diagrams of Figure 7. Moreover, it is worth observing that the larger the released mass flow rate, the higher the emitted noise, a result that holds for all the tested conditions and previously discussed in Section 3.1. However, and rather interestingly, the profiles suggest that the emitted noise has an ever-decreasing growth, evidence of which is also shown by the best-fitting curves and their decreasing slope as mass flow rate increases. This observation also indirectly arises from SPL trends shown in Figure 7: the peaks exhibit similar values as orifice

diameter shifts towards the larger sizes, regardless of whether standard (Figure 7g,i) or silent (Figure 7h,j) nozzles are employed. This behavior is consistent with the discussion proposed in Sections 3.1 and 3.2 with regard to the relationship between noise and Reynolds number, which hints at turbulence intensity of the issued jet: as fully developed turbulence is approached, the emitted noise tends to increase under a milder slope. The comparison of Figure 12c and Figure 12b,d suggests that FK-5-1-12 generally yields lower SPL than nitrogen IG100, which is consistent with the findings described in Section 3.2: higher mass flow rate of FK-5-1-12 generates approximately the same noise as a lower mass flow rate of nitrogen IG100, since FK-5-1-12 density is higher than nitrogen density. Even though the datasets are of different size, as previously remarked, and the gap in terms of emitted noise is perceivable, yet mild (i.e., in the order of 10–20 dB, considering standard and silent nozzles), it appears that the different values of the involved thermophysical properties (i.e., density and viscosity) make FK-5-1-12 discharge generate lower noise than nitrogen IG100 discharge, under the same released mass flow rate. On the other hand, the generated noise is rather similar under the same released volumetric flow rate.

Table 4. Constant scaling factor and exponent for the fitting curves generated through a power law formulation.

Tested Condition (Nozzle Type, Clean Agent)	Scaling Factor a	Exponent b
Standard, nitrogen IG100	124.4	0.06788
Silent, nitrogen IG100	112.3	0.1873
Standard, FK-5-1-12	109	0.05422
Silent for low ceiling applications, FK-5-1-12	80.49	0.2596

4. Conclusions

Most hard drives are subject to performance decay as noise beyond a certain threshold —usually identified as 110 dB [4,16–20]—reaches them. The problem is particularly relevant when fire protection of data centers is involved, since gas-based suppression systems are widely employed for this purpose: the noise emitted by the nozzles through a discharge of the selected clean agents (i.e., inert gases or halocarbon compounds) is well-known to damage the electronic equipment within the compartment. Hence, silent nozzles are designed and often used to address this undesired effect; however, the open literature currently offers few scientific studies on the subject. Arguably an unprecedented effort, an experimental approach was devised and developed to quantitatively assess the noise generated by both standard and silent nozzles in a comparative fashion, together with the performance decrease in the hard drives throughout several discharge conditions. To this end, a preliminary evaluation of repeatability and ability to properly capture the involved phenomena was carried out against the proposed experimental setup and methodology. Notably, the employed nozzles proved reasonably suitable for applying the point source model [22], thus allowing us to neglect the directionality of the emitted noise. The developed approach may be expanded by repeating the same measurements (i.e., sound pressure level and performance decay) at various angular locations, following the guidance from a previous work [15], which applies the here-presented setup as a reference to evaluate acoustic performance of virtually any nozzle for gas-based suppression system. A series of experiments were conducted, mostly releasing nitrogen IG100, but also FK-5-1-12, which allowed a comparison between the two types of clean agents. Constant pressure was imposed throughout a large portion of each discharge event using constant flow valves; various orifice diameters were tested, which translates into variations in the released flow rate. Moreover, three hard drives were used as targets to test their performance over a discharge event: two HDDs with different rates of rotation and one SSD, which was almost unaffected by external noise.

As expected, silent nozzles proved capable of limiting the generated noise down to values less harmful to hard drives. Notably, each discharge—whether produced by standard or silent nozzles—exhibited an increase in the emitted noise up to a maximum

from its onset, usually followed by a plateau-like trend and the final decay. Fluctuations occurred possibly as a result of pressure oscillations within the controlled phase (i.e., the release at virtually constant pressure). This behavior hints at the known dependence of sound emitted by gaseous jets flowing through an orifice on turbulence intensity, thus being ultimately correlated to Reynolds number. This relationship also supports the discovered increase in peak noise level as the released flow rate increased (i.e., as larger orifice diameter was employed in the tests). Nevertheless, the difference between the reached noise maxima became less remarkable as the released flow rate increased, which also suggests that once the Reynolds number becomes sufficiently high that fully developed turbulence be reached, noise intensity increases along a milder slope. Rather interestingly, the recorded performance decay of the target HDD qualitatively followed the noise trend, yet the HDD featuring the lower read/write speed appeared remarkably more affected than that with the higher rate of rotation. Notably, the former exhibited instances of partial performance recovery as noise decreased as the discharge event approached its end, which emphasized the need for conducting spectral analysis of the emitted sound. It was observed that noise emitted at a higher frequency (4–8 kHz) could reach values above 100 dB even in the last part of the discharge event: as found by Dutta [31], such noise intensity in this band can be detrimental to HDD performance, even though it is lower than 110 dB. With regard to the comparison between the two clean agent types, a similar behavior could be inferred by comparing results from nitrogen IG100 and FK-5-1-12 discharge. However, a noise regrowth was observed as silent nozzles were employed in the latter case, which is arguably related to the discharge of the superpressurizing agent (i.e., nitrogen) added within the tank, once the halocarbon compound is fully released. This second phase proved consistent with inert gas discharge in terms of emitted noise, yet also suggested a better noise-damping effect on FK-5-1-12 by silent nozzles than on nitrogen. The commonly accepted independence of generated noise from the released agent is also supported by the acquired dataset when referring to volumetric flow rate (i.e., orifice diameter of similar size and same discharge pressure). However, the discharge of a stronger FK-5-1-12 mass flow rate generates almost the same noise as that of a lower mass flow rate of nitrogen IG100, which is consistent with the former having higher density. The difference in the values of the involved thermophysical properties (i.e., density and viscosity) impacts on Reynolds number of the two jets and ultimately on the emitted noise as they pass through an orifice.

The present work sheds light on the acoustical performance of nozzles employed in gas-based fire protection in relation to several design parameters (i.e., selected clean agent type, flow rate, orifice diameter); moreover, it proposes an approach to compare nozzles in terms of emitted noise. Overall, both the results and the methodology may be of interest for nozzle and fire protection system designers, as well as provide a foundation for further research on specific aspects of the involved phenomena (e.g., quantitative relationship between noise and turbulence intensity, potential supersonic-to-subsonic transition through the discharge). Further research may arise out of this work towards investigating the combined action of several nozzles—either standard or silent—at the same time, in terms of generated noise (i.e., superposition of waves and interference) and impact on HDDs. Moreover, the here-proposed methodology can be employed to expand the variety of tested silent nozzles, which may include new structures such as sound-absorbing layers (e.g., honeycomb porous shells [32]).

Author Contributions: Conceptualization, L.T. and P.E.S.; methodology, N.T., L.T. and P.E.S.; formal analysis, M.S., N.T., L.T. and P.E.S.; investigation, M.S., N.T., L.T. and P.E.S.; resources, L.T.; writing—original draft preparation, P.E.S.; writing—review and editing, N.T. and P.E.S.; supervision, P.E.S.; project administration, L.T. and P.E.S.; funding acquisition, L.T. All authors have read and agreed to the published version of the manuscript.

Funding: This study was partially funded by Regione Emilia-Romagna, Italy through the POR-FESR 2014-2020 program, as a part of the FIRESOFT—*Piattaforma software di modellazione, testing e validazione in-house di impianti antincendio fissi* project, grant agreement (CUP code) no. E85F19001860007.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors wish to thank the technical staff of Bettati Antincendio S.r.l., Italy for providing assistance in assembling the experimental setup.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Grant, C.C. Halon design calculations. In *SFPE Handbook of Fire Protection Engineering*; DiNenno, P.J., Drysdale, D., Beyler, C.L., Walton, W.D., Custer, R.L.P., Hall, J.R., Jr., Watts, J.M., Jr., Eds.; National Fire Protection Association: Quincy, MA, USA, 2002; Section 4, Chapters 4–6; pp. 149–172.
2. DiNenno, P.J. Halon replacement clean agent total flooding systems. In *SFPE Handbook of Fire Protection Engineering*; DiNenno, P.J., Drysdale, D., Beyler, C.L., Walton, W.D., Custer, R.L.P., Hall, J.R., Jr., Watts, J.M., Jr., Eds.; National Fire Protection Association: Quincy, MA, USA, 2002; Section 4, Chapters 4–7; pp. 173–200.
3. Kim, Y.-H.; Lee, M.; Hwang, I.-J.; Kim, Y.-J. Noise reduction of an extinguishing nozzle using the Response Surface Method. *Energies* **2019**, *12*, 4346. [CrossRef]
4. Mann, M.T.; Coll, M. *Potential Problems with Computer Hard Disks When Fire Extinguishing Systems Are Released*; Siemens Switzerland: Zug, Switzerland, 2010.
5. Lilley, G.M. Aerodynamic noise—A review of the contributions to jet noise research at the College of Aeronautics, Cranfield 1949-1961 (together with some recent conclusions). *Aeronaut. J.* **1984**, *88*, 213–223.
6. Tam, C.K.W.; Tanna, H.K. Shock associated noise of supersonic jets from convergent-divergent nozzles. *J. Sound Vib.* **1982**, *81*, 337–358. [CrossRef]
7. Tam, C.K.W. Influence of nozzle geometry on the noise of high-speed jets. *AIAA J.* **1998**, *36*, 1396–1400. [CrossRef]
8. Dutta, T.; Barnard, A.R. Performance of hard disk drives in high noise environments. *Noise Control Eng. J.* **2017**, *65*, 386–395. [CrossRef]
9. Green, K.; Nelson, D.; Pai, N.; Nickerson, M. Hard drive performance degradation due to high level noise in data centers. In Proceedings of the 40th International Congress and Exposition on Noise Control Engineering—Inter-Noise, Osaka, Japan, 4–7 September 2011.
10. Nickerson, M.L.; Green, K.; Pai, N. Tonal noise sensitivity in hard drives. *Proc. Meet. Acoust.* **2014**, *20*, 040006.
11. Daigle, B. *Data Centers around the World: A Quick Look*; United States International Trade Commission: Washington, DC, USA, 2021.
12. Loureiro, M.A.; Elder, A.; Ahmadzadegan, A. Acoustic nozzle design for fire protection application. In Proceedings of the 2017 Suppression, Detection, and Signaling Research and Applications Conference (SUPDET 2017) and 16th International Conference on Fire Detection (AUBE'17), College Park, MD, USA, 12–14 September 2017.
13. Kim, Y.-H.; Yoo, H.-S.; Hwang, I.-J.; Kim, Y.-J. Influence of the nozzle contraction angles of gaseous extinguishing systems on discharge noise. *Fire Sci. Eng.* **2019**, *33*, 77–82. [CrossRef]
14. Mihalace, C.-M.; Bigan, C.; Panduru, V.; Tsakiris, C. Modelling of noise reduction for datacentre buildings fire protection with inert gas systems. *MATEC Web Conf.* **2019**, *290*, 12006. [CrossRef]
15. Koushik, S.; McCormick, D.; Cao, C.; Corn, M. Acoustic impact of fire suppression nozzles. In Proceedings of the 2017 Suppression, Detection, and Signaling Research and Applications Conference (SUPDET 2017) and 16th International Conference on Fire Detection (AUBE'17), College Park, MD, USA, 12–14 September 2017.
16. *NFPA 75*; Standard for the Fire Protection of Information Technology Equipment. National Fire Protection Association: Quincy, MA, USA, 2020.
17. *Silence Is Golden—Maintaining the Integrity of HDD's in a Noisy Situation*; Fire Eater: Hillerød, Denmark, 2016.
18. *Effect of Sound Waves on Data Storage Devices*; Fire Suppression Systems Association: Baltimore, MD, USA, 2018.
19. Sandahl, D.; Elder, A.; Barnard, A. *Impact of Sound on Computer Hard Disk Drives and Risk Mitigation Measures*; Form no. T-2016367-01; Johnson Controls: Milwaukee, WI, USA, 2018.
20. *Silent Extinguishing*; Siemens Switzerland: Zug, Switzerland, 2022.
21. Silva, R.A.; Buiatti, C.M.; Cruz, S.L.; Pereira, J.A.F.R. Pressure wave behaviour and leak detection in pipelines. *Comput. Chem. Eng.* **1996**, *20*, 5491–5496. [CrossRef]
22. Li, K.M.; Waters-Fuller, T.; Attenborough, K. Sound propagation from a point source over extended-reaction ground. *J. Acoust. Soc. Am.* **1998**, *104*, 679. [CrossRef]
23. Coleman, H.W.; Steele, W.G. *Experimentation and Uncertainty Analysis for Engineers*; Wiley: New York City, NY, USA, 1999.
24. Tilton, J.N. Fluid and particle dynamics. In *Perry's Chemical Engineers' Handbook*; Perry, R.H., Green, D.W., Eds.; McGraw-Hill: New York City, NY, USA, 1997; Section 6; pp. 1–54.
25. CoolProp. Available online: http://www.coolprop.org/ (accessed on 24 August 2022).

26. Santangelo, P.E.; Jacobs, B.C.; Ren, N.; Sheffel, J.A.; Corn, M.L.; Marshall, A.W. Suppression effectiveness of water-mist sprays on accelerated wood-crib fires. *Fire Saf. J.* **2014**, *70*, 98–111. [CrossRef]
27. Santangelo, P.E.; Tarozzi, L.; Tartarini, P. Full-scale experiments of fire control and suppression in enclosed car parks: A comparison between sprinkler and water-mist systems. *Fire Technol.* **2016**, *52*, 1369–1407. [CrossRef]
28. Cannio, M.; Righi, S.; Santangelo, P.E.; Romagnoli, M.; Pedicini, R.; Carbone, A.; Gatto, I. Smart catalyst deposition by 3D printing for Polymer Electrolyte Membrane Fuel Cell manufacturing. *Renew. Energy* **2021**, *163*, 414–422. [CrossRef]
29. Mi, J.; Xu, M.; Zhou, T. Reynolds number influence on statistical behaviors of turbulence in a circular free jet. *Phys. Fluids* **2013**, *25*, 075101. [CrossRef]
30. White, F.M. *Viscous Fluid Flow*; McGraw-Hill: New York City, NY, USA, 2006.
31. Dutta, T. Performance of Hard Disk Drives in High Noise Environments. Master's Thesis, Michigan Technological University, Houghton, MI, USA, 2017.
32. Thongchom, C.; Jearsiripongkul, T.; Refahati, N.; Roudgar Saffari, P.; Roodgar Saffari, P.; Sirimontree, S.; Keawsawasvong, S. Sound transmission loss of a honeycomb sandwich cylindrical shell with functionally graded porous layers. *Buildings* **2022**, *12*, 151. [CrossRef]

applied sciences

MDPI

Article

COVID-19 Detection Model with Acoustic Features from Cough Sound and Its Application

Sera Kim [1], Ji-Young Baek [1] and Seok-Pil Lee [2,*]

[1] Department of Computer Science, Graduate School, SangMyung University, Seoul 03016, Republic of Korea
[2] Department of Electronic Engineering, SangMyung University, Seoul 03016, Republic of Korea
* Correspondence: esprit@smu.ac.kr

Abstract: Contrary to expectations that the coronavirus pandemic would terminate quickly, the number of people infected with the virus did not decrease worldwide and coronavirus-related deaths continue to occur every day. The standard COVID-19 diagnostic test technique used today, PCR testing, requires professional staff and equipment, which is expensive and takes a long time to produce test results. In this paper, we propose a feature set consisting of four features: MFCC, Δ^2-MFCC, Δ-MFCC, and spectral contrast as a feature set optimized for the diagnosis of COVID-19, and apply it to a model that combines ResNet-50 and DNN. Crowdsourcing datasets from Cambridge, Coswara, and COUGHVID are used as the cough sound data for our study. Through direct listening and inspection of the dataset, audio recordings that contained only cough sounds were collected and used for training. The model was trained and tested using cough sound features extracted from crowdsourced cough data and had a sensitivity and specificity of 0.95 and 0.96, respectively.

Keywords: AI diagnostics; COVID-19 screening; deep learning; speech recognition

Citation: Kim, S.; Baek, J.-Y.; Lee, S.-P. COVID-19 Detection Model with Acoustic Features from Cough Sound and Its Application. *Appl. Sci.* **2023**, *13*, 2378. https://doi.org/10.3390/app13042378

Academic Editor: Zhi-Ting Ye

Received: 4 January 2023
Revised: 5 February 2023
Accepted: 10 February 2023
Published: 13 February 2023

1. Introduction

COVID-19 is an acute respiratory infection that develops from SARS-CoV-2, a new type of coronavirus that was first reported in November 2019. This is a pandemic that continues worldwide as of November 2022, with a cumulative number of confirmed cases of 640 million and fatalities of 6.6 million. A characteristic of coronavirus is that it spreads swiftly and readily. Consequently, studies are being actively conducted on how to analyze how the coronavirus spreads and how to prevent its spread [1–3]. The omicron mutation, which has a low fatality rate but a very high transmission rate, has become the dominant variant. As the number of confirmed cases increases quickly, so do the numbers of severely sick patients and fatalities. Additionally, even though the fatality rate is low, an infection of COVID-19 may still be fatal for the elderly or those with underlying illnesses; thus, it is crucial to stop the spread of the disease by obtaining early diagnosis and treatment. The main route of infection is known to be droplets and respiratory secretions in the air produced by infected individuals.

The most frequently used diagnostic test for COVID-19 is real-time reverse transcription polymerase chain reaction (real-time RT-PCR), which is a technique for amplifying and identifying a particular coronavirus gene [4]. Because it has the greatest sensitivity and specificity and can detect even minute amounts of virus in a sample, this test method is used as a worldwide standard. Its drawbacks include the need for specialized tools, reagents, and skilled professionals, as well as the comparatively lengthy turnaround time of roughly 24 h for diagnostic outcomes.

Worldwide studies are being performed to find ways other than genetic testing to identify those who are COVID-19 positive. Chest X-ray or chest computed tomography (CT) images have been offered as the input for deep learning models [5,6]. In a study that used the fact that COVID-19 positive individuals have a specific volatile substance that

is distinct from that of non-infected individuals, a COVID-19 detection scheme using the olfactory abilities of dogs was proposed [7]. Another study proposed using heart rate, sleep time, and activity data collected using wearable sensors to detect COVID-19 [8]. In a study that examined correlations with positive COVID-19, from findings using 42 characteristics that included fever, cough, chest CT, and body temperature, the characteristic that showed the strongest positive correlation was cough [9]. Based on this, this study investigates a method for identifying COVID-19-infected people using cough.

Many studies are being conducted to identify COVID-19 through cough sounds in order to allow low-cost and quick large-scale diagnostic testing [10–23]. Respiratory symptoms are one of the features of COVID-19 infection; hence, the data provided by the sound of coughing is used. A sound contains many features [24], and so does the sound of a cough. The deep learning model trained using these features can determine whether a cough sound is from a COVID-19-infected individual.

Looking at previous studies that used cough sounds, the amount of data is not large. Brown et al. [20] used only the Cambridge dataset [20], and Feng et al. [21] used Coswara [25] and Virufy [26] as datasets to study a COVID-19 diagnostic model. Fakhry et al. [22] used only the COUGHVID dataset [27]. In order to improve the stability and accuracy of the results, the quantity and quality of data are very important. Therefore, in this paper, all of the Cambridge, Coswara, and COUGHVID datasets were used, and a high-quality dataset was built through preprocessing.

In addition, when selecting a feature set in previous studies, the model was trained by combining several spectral-based features simply because it was a feature mainly used in speech. In this study, we propose a feature set optimized for COVID-19 diagnosis. By using the Bhattacharyya distance [28], which is a method of calculating the degree of separation between classes, a feature set was constructed by obtaining features that can discriminate well between the cough sounds of COVID-19-positive subjects and those of negative subjects. As a result, the feature set was composed of mel frequency cepstral coefficients (MFCC), Δ^2-MFCC, Δ-MFCC, and spectral contrast. With this feature set and a mel spectrogram as input, we trained a model [22] that combined ResNet-50 [29] and a deep neural network (DNN), and the model achieved a 0.95 sensitivity and a 0.96 specificity. The result showed improvement compared with previous studies.

The structure of this paper is as follows. The collection of three crowdsourcing datasets is described in Section 2, along with an earlier study on models for diagnosing COVID-19 infections using each dataset. The current study's database, database preparation, method for determining the Bhattacharyya distance and creating the feature set, and model are all covered in Section 3. The experimental results compared with previous studies are presented in Section 4. How to apply the constructed model to an application is covered in Section 5. The study's findings and future directions are covered in Section 6.

2. Related Work

2.1. Cambridge

The Cambridge dataset [20] was collected using an Android/web application and includes the participants' cough, voice, and breath sounds as well as information on their medical history and symptoms. Participants record three coughing sounds and three breathing sounds after providing their age, gender, a brief medical history, and any symptoms they may be experiencing. A total of 4352 users of the web app and 2261 users of the Android app each contributed to the dataset, resulting in 5634 samples and 4352 samples, respectively. The participants' gender distribution was as follows: 4525 males, 2056 females, 26 non-respondents, and 6 others. Through the process of directly checking all the samples of COVID-19-positive test cases, 141 samples were retrieved.

Brown et al. [20] trained a classification model by extracting two types of features—handcrafted features and features from transfer learning—from the Cambridge dataset. Gradient boosting trees and support vector machines (SVMs) were used as a classification model. Raw sound waveform data were resampled at a frequency of 22 kHz before

handcrafted characteristics were extracted. Handcrafted features were extracted at the frame level and segment level. A total of 477 handcrafted features were used, including duration, MFCC, Δ-MFCC, Δ^2-MFCC, onset, period, root mean square (RMS) energy, roll-off frequency, spectral centroid, tempo, and zero-crossing. The features from transfer learning used VGGish [30], which is a convolutional neural network designed for audio classification based on raw audio input. VGGish is a model trained using a large YouTube dataset, and the parameters of the model are public. A pre-trained VGGish model was used to extract 256 dimensional features, and the sampling rate was 16 kHz. As a result of training the SVM model by extracting two types of features using only the cough sound data, the area under the ROC curve (AUC) was 0.82 and the sensitivity was 0.72.

2.2. Coswara

A project named Coswara [25] was carried out in India to develop a tool for diagnosing COVID-19 using audio recordings such as speech, breathing, and coughing. Worldwide data were gathered with an easy-to-use, interactive user interface. Participants recorded samples using a device microphone, such as a laptop or mobile phone, and provided metadata using a web browser application. During recording, participants were instructed to keep a distance of 10 cm between their mouth and the device. Nine audio samples were recorded per person: cough sounds (deep and shallow), breathing sounds (fast and slow), sustained vowel ('eu', 'I', 'u') pronunciation sounds, and counting sounds (slow and fast). The participant's age, gender, area, history of illness, and current state of health with respect to COVID-19 were all included in the metadata. The sampling frequency of audio samples was 48 kHz. Data from 2747 individuals were made public as of 24 February 2022. Of these, 681 individuals had COVID-19-positive test results.

Feng et al. [21] used Virufy [26], a dataset collected under the supervision of medical professionals in hospitals, along with the Coswara dataset, to study a COVID-19 diagnostic model. In the Coswara dataset, only shallow cough recordings were used, and 200 samples of healthy people's data were randomly selected and used to balance the training data set. The Virufy dataset consists of 16 recordings: 7 recordings from people diagnosed with COVID-19 and 9 recordings from healthy people. The Coswara dataset was used for training and the Virufy dataset was used for testing. First, silent recordings and noise/speaking recordings were distinguished through the SVM model trained with the energy features of each audio recording, and only the parts containing sound were extracted. Features of the audio signal were then extracted: centroid, energy, entropy of energy, spectral flux, MFCCs, spectral spread, and zero-crossing rate. A k-nearest neighbors (KNN) model was used to distinguish conversational sounds from cough sounds in a recording, and a new recording was created by connecting the cough sounds detected within one recording. The next step was to train four models—the KNN, SVM, random forest, and recurrent neural network (RNN)—to classify coughs in COVID-19-positive participants and healthy individuals. As a result of testing with the RNN model, which had the best results during the training process, the accuracy was 0.81 and the AUC was 0.79.

2.3. COUGHVID

The COUGHVID dataset [27] is a large publicly available cough sound dataset. The dataset contains over 20,000 recordings and contains the labels COVID-19, symptomatic, and healthy. Data collection was conducted using a web application from 1 April 2020 to 10 September 2020. After 10 s of cough sound recording for each person, participants were asked to fill out and submit simple information, including age, gender, and current condition. To remove non-cough recordings from the database, a cough detection model was applied and the cough sound scores of all recordings were analyzed. The scores were indicated by "cough_detected" in the metadata. Audio data were in the WEBM or OGG format, and the sampling frequency was 48 kHz.

Fakhry et al. [22] proposed a multi-branch deep learning network using the COUGHVID dataset as a model for diagnosing COVID-19. Only data with a cough detection score of 0.9 or higher were used, and the filtered data were 4446 recordings from healthy people, 923 recordings from symptomatic people, and 380 recordings from people who tested positive for COVID-19, for a total of 5749 audio files. In order to rectify the imbalance in the number of data, they increased the number of COVID-19-positive recordings to 750 by adding Gaussian noise, pitch shifting, and shifting or increasing the time signal. By selecting an equal number of recordings from symptomatic and healthy people, a total of 2250 audio samples were used. MFCCs and the mel spectrogram, which are commonly used in audio classification and speech recognition, were used as acoustic features for the network training. In addition, clinical features such as respiratory status or fever were used as a one-dimensional vector of binary numbers. The MFCCs were input to a dense layer of 256 and 64 nodes, and the mel spectrograms were input to a ResNet-50 that was pre-trained on the ImageNet dataset. Clinical features were input to a dense layer of 8 nodes. The output values of all results were connected to form a combined multi-branch. The sensitivity and specificity of this model were 0.85 and 0.99, respectively.

3. Proposed Method

3.1. Data

The cough sound databases of Cambridge [20], Coswara [25], and COUGHVID [27] presented in Section 2 were used in the experimental data of this study. From the COUGHVID data, only data with a cough detection score of 0.9 or higher were extracted based on metadata that included score information in relation to the degree of cough sound detection. Because all three databases had more cough sound data from COVID-19-negative participants than from positive individuals, only a portion of the cough sound data from negative participants was used; this was conducted to balance the data. Due to the nature of crowdsourced data, which is collected in various environments, it may contain data that are difficult to use for a study. Therefore, the audio files were directly listened to and inspected. Data were deleted through inspection in the following cases.

- The cough sound is quieter than the noise.
- The recording quality is too poor.
- Background noise (conversation, road noise, music, TV/radio, etc.) is mixed with the cough sound.
- It is difficult to recognize the cough sound.

As indicated in Tables 1 and 2, 4200 audio files were inspected, and finally 2049 cough sound audio files were selected. The selected database consisted of 1106 audio files of cough sounds from COVID-19-positive participants, 530 audio files of cough sounds from healthy people, and 413 audio files of cough sounds from people with symptoms.

Table 1. Number of audio files before inspection.

	Cambridge	Coswara	COUGHVID	Total
COVID-19	299	1336	441	2076
healthy	499	400	201	1100
symptomatic	295	428	301	1024

Table 2. Number of audio files after inspection.

	Cambridge	Coswara	COUGHVID	Total
COVID-19	247	656	203	1106
healthy	299	123	108	530
symptomatic	179	92	142	413

3.2. Preprocessing

Because the data were collected in a variety of ways and in a wide range of environments, normalization was first performed to make the scale of the data uniform. Then, a process of detecting only cough sounds in the data was carried out. Through this process, unnecessary voices and other noises recorded during the data collection process were removed, and only clear cough sounds, which are useful data for the study, were obtained. The method used for cough detection was that described by Orlandic et al. [27]. In Figure 1, cough detection was performed using one audio file as the original data, and the detected cough segment is shown as an example.

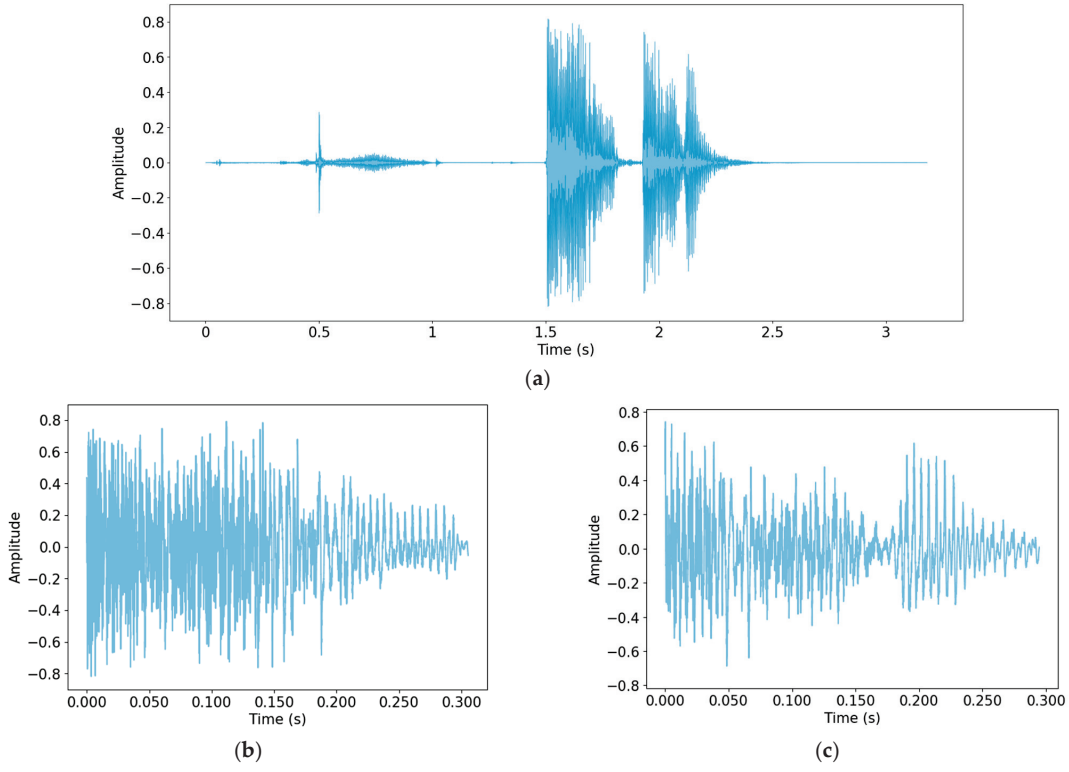


Figure 1. Performing cough detection: (**a**) Original audio file data; (**b**) detected cough segment 1; (**c**) detected cough segment 2.

As shown in Figure 1, the part of the speech sound in the front part of (a) was not detected. The cough recorded twice in succession was detected by dividing it into two cough segments. Table 3 shows the number of cough segments from each database.

Table 3. Number of cough segments detected.

	Cambridge	Coswara	COUGHVID	Total
COVID-19	687	957	483	2127
healthy	627	285	255	1167
symptomatic	622	155	346	1123

3.3. Feature Set

3.3.1. Audio Feature Vector

In this section, a feature set for this study was constructed. In addition to the spectral-based speech features mainly used in speech research, several features were added and used. Twelve features were used: chroma, onset, RMS energy, spectral bandwidth, spectral centroid, spectral contrast, spectral flatness, spectral roll-off, MFCC, Δ-MFCC, Δ^2-MFCC, and zero-crossing rate. The 13th order MFCC was used as the MFCC. All features were extracted using the librosa package [31] with a sampling frequency of 24,000 Hz. To form the final feature set, features that were effective for detecting COVID-19 were selected from among the 12 features. In order to do this, the Bhattacharyya distance [28], a method that measures the separability of classes, was used.

3.3.2. Bhattacharyya Distance

The database was divided into two classes, positive and negative, to identify the difference between COVID-19-positive data and negative data. Features were extracted from the cough segment, and the separation between the two classes was calculated for the same feature vector. Equation (1) is the formula for calculating the Bhattacharyya distance, where μ_1 and μ_2 represent the averages of each class and $\sum_1$ and $\sum_2$ represent the covariances of each class. The larger the difference between the two classes, the larger the distance.

$$D_B = \frac{1}{8}(\mu_2 - \mu_1)^T \left[\frac{\sum_1 + \sum_2}{2}\right]^{-1} (\mu_2 - \mu_1) + \frac{1}{2}\ln \frac{\left|\frac{\sum_1 + \sum_2}{2}\right|}{|\sum_1|^{1/2} |\sum_2|^{1/2}} \quad (1)$$

Table 4 presents the results for the Bhattacharyya distance sorted in descending order. The feature with the largest difference between the two classes was MFCC, with a value of 0.207171, and the feature with the smallest difference was onset with a value of 0.002387. The final feature set to be used in this study consisted of the top four features: MFCC, Δ^2-MFCC, Δ-MFCC, and spectral contrast.

Table 4. Bhattacharyya distance of each feature.

Feature	Bhattacharyya Distance
MFCC	0.207171
Δ^2-MFCC	0.149195
Δ-MFCC	0.099828
Spectral Contrast	0.090616
Chroma	0.063358
Spectral Flatness	0.057523
Spectral Bandwidth	0.046912
Spectral Roll-Off	0.032971
RMS Energy	0.018301
Spectral Centroid	0.016368
Zero-Crossing Rate	0.002629
Onset	0.002387

3.4. Model

In this study, a model combining ResNet-50 and a DNN proposed by Fakhry et al. [22] was used. ResNet-50 is a convolutional neural network composed of 50 layers that allows for stable learning while the depth of the model increases. The DNN is an artificial neural network (ANN) that consists of several hidden layers between the input layer and the output layer. The mel spectrogram image obtained from the cough segment was input to ResNet-50, the feature set was input to the DNN, and the inputs were trained.

In the first branch, ResNet-50 was trained with the mel spectrogram image of (224, 224, 3) as the input. In ResNet-50, an image of size (224, 224, 3) went through multiple

convolutional, activation, and pooling layers to reduce the size of the image while extracting features. This process was repeated multiple times, with each iteration reducing the size of the feature map while increasing the number of filters used by the network. The output of the final pooling layer was a tensor with size (7, 7, 2048), which represented a compact representation of the input image. The output went through global average Pooling and global max pooling separately. The global pooling layer is a method of replacing values of the same channel with one average or maximum value. Overfitting can be prevented because the parameters are reduced. If an input of size (height, width, channel) passes through the global pooling layer, it becomes (1, 1, channel). The two outputs obtained through this process were connected after batch normalization and dropout were performed. In batch normalization, the activations of each neuron in a layer are normalized using the mean and standard deviation of the activations in a subset of the training data. The normalized activations are then scaled and shifted using learned parameters. In dropout, neurons are randomly dropped out during each iteration of training, with a specified probability. These processes help to ensure that the network will be able to generalize new data well, while still being able to learn effectively from the training data. The second branch took a 46-dimensional feature set as input. The feature set was input to a dense layer consisting of 256 nodes, and batch normalization and dropout were performed. The output became the input to a dense layer with the same number of nodes, and batch normalization and dropout were performed once more. The above process was performed in the same way for the dense layer consisting of 64 nodes to obtain another output. The two outputs were then connected. The outputs obtained through the first branch and the second branch were connected and became the input to the dense layer, and batch normalization and dropout were performed. Finally, the sigmoid function was used to calculate the value, to distinguish whether the input was the cough sound of COVID-19-positive individual or negative individual. Figure 2 shows the flow chart of the model.

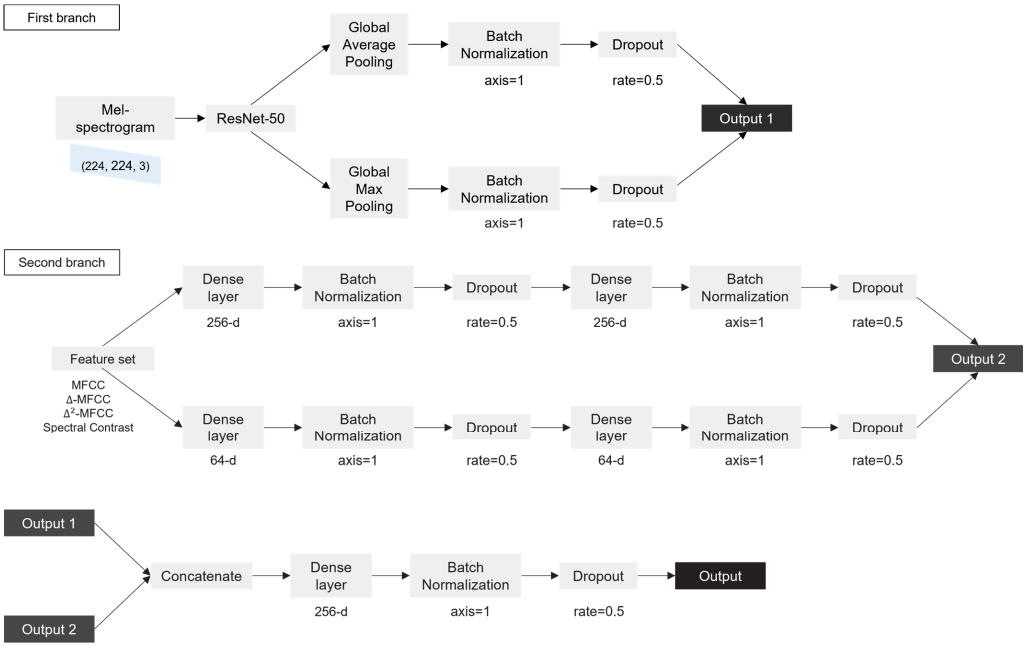


Figure 2. Flow chart of the model.

4. Experiment

In the experimental step, we carried out a procedure to verify that the feature set proposed in this study and the combined model of Resnet-50 and DNN were effective at detecting COVID-19. The experiment was conducted by using a combination of the various datasets and feature sets. For example, for the same database, the results of training using 'A feature set' and the results of training using 'B feature set' were compared. The hyperparameters used in the experiment were set to optimizer Adam, learning rate 0.001, and epoch 50. The experimental results of this study were compared with those of previous studies.

4.1. Evaluation Index

Accuracy, sensitivity, specificity, and precision, which are frequently used to evaluate the performance of classification models, were used to evaluate the training results. The focus was on sensitivity and specificity, which are primarily considered when measuring the reliability of the actual COVID-19 test diagnosis method. Figure 3 is a confusion matrix used to calculate the above indicators.

		Predicted	
		Positive	Negative
Actual	Positive	TP	FN
	Negative	FP	TN

Figure 3. Confusion matrix.

Sensitivity is the ratio of data predicted as positive (TP) to the actual positive class (TP + FN), and specificity is the ratio of data predicted as negative (TN) to the actual negative class (FP + TN). A high sensitivity means that there is a low probability of a false negative, i.e., a low probability of a positive being falsely classified as negative.

4.2. Results

The results of each study are shown in Table 5 (a) to (d) show the results using the LSTM model and the ResNet-50 model, not the model proposed in this study. (a) and (b) are the results of Son [23] using COUGHVID data, and (c) and (d) are the results using the dataset constructed in this study. (e) to (i) are the results verifying the feature set we proposed. All used a model that combined ResNet-50 and DNN. (e) is the result of study by Fakhry et al. [22], using 13 MFCCs and a mel spectrogram as features from the COUGHVID dataset only, and the accuracy, sensitivity, and specificity was 0.89, 0.93, and 0.86, respectively. (f) is the result of Son et al. [23], using seven features (13 MFCCs, spectral centroid, spectral bandwidth, spectral contrast features, spectral flatness, spectral roll-off, and chroma), and a mel spectrogram. The accuracy, sensitivity, and specificity of Son's study were 0.94, 0.93, and 0.94, respectively. (g) to (i) are the experimental results obtained in this study, and the model was trained using data from Cambridge, Coswara, and COUGHVID. (g) is an extension of only the database in Fakhry's study, with the others remaining the same. The result had an accuracy of 0.93, a sensitivity of 0.93, a specificity of 0.93, and a precision of 0.93. (h) shows the model trained with the feature set proposed by Son's study, and the result gave an accuracy of 0.92, a sensitivity of 0.90, a specificity of 0.94, and a precision of 0.90. (i) is the method proposed in this study. The model was trained using the configured feature set, MFCC, Δ-MFCC, Δ^2-MFCC, and spectral contrast. The result had an accuracy of 0.96, a sensitivity of 0.95, a specificity of 0.96, and a precision of 0.95. This performance showed a better result than the previous studies mentioned above.

Table 5. Comparison of the results.

	Database	Feature Set	Model	Performance		
				Accuracy	Sensitivity	Specificity
(a)	COUGHVID	MFCC	LSTM	0.62	0.60	0.62
(b)	COUGHVID	Spectrogram	ResNet-50	0.88	0.90	0.88
(c)	Cambridge + Coswara + COUGHVID	MFCC	LSTM	0.62	0.58	0.67
(d)	Cambridge + Coswara + COUGHVID	Spectrogram	ResNet-50	0.91	0.87	0.93
(e)	COUGHVID	[22]	ResNet-50 + DNN	0.89	0.93	0.86
(f)	COUGHVID	[23]	ResNet-50 + DNN	0.94	0.93	0.94
(g)	Cambridge + Coswara + COUGHVID	[22]	ResNet-50 + DNN	0.93	0.93	0.93
(h)	Cambridge + Coswara + COUGHVID	[23]	ResNet-50 + DNN	0.92	0.90	0.94
(i)	Cambridge + Coswara + COUGHVID	Proposed feature set + spectrogram	ResNet-50 + DNN	0.96	0.95	0.96

To statistically verify the above results, a statistical analysis method was used. For the results of (e) to (i), which are experiments using the proposed model, one-way analysis of variance (ANOVA) was used to confirm whether the differences in performances are statistically significant. Table 6 shows the ANOVA results. The p-value was 0.0205, which is less than 0.05. This indicates that the difference in performance between each experiment is statistically significant.

Table 6. Results of ANOVA.

	Degrees of Freedom	Sum of Squares	Mean Square Error	F Value	p-Value
group	4	0.006493	0.001623	4.775	0.0205
Residuals	10	0.003400	0.000340		

Thereafter, as a post hoc analysis, differences between performances were confirmed using the Bonferroni multiple comparison analysis method. It was executed using R studio, which is widely used for data analysis and statistical computing. Figure 4 shows the Bonferroni correction results. The performances of (e) to (i) are divided into groups a, ab, and b, and the difference between the performances is visualized.

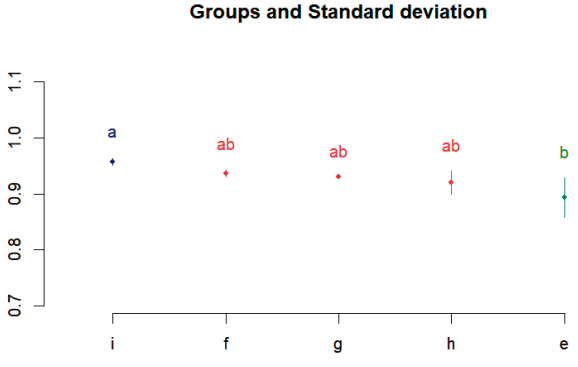


Figure 4. Post hoc test results using Bonferroni correction.

5. COVID-19 Detecting Application

We developed an application using the proposed model so that it could be used to diagnose COVID-19 in many people. An application for Android, which is a mobile operating system based on open-source software produced and released by Google, was produced. Figure 5 shows the execution process of the developed application.

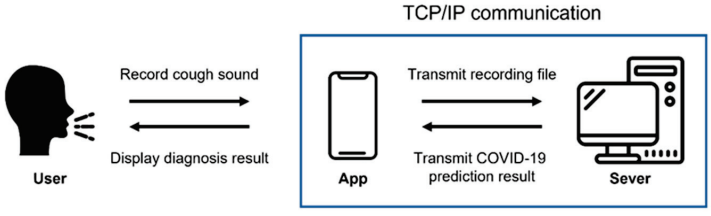
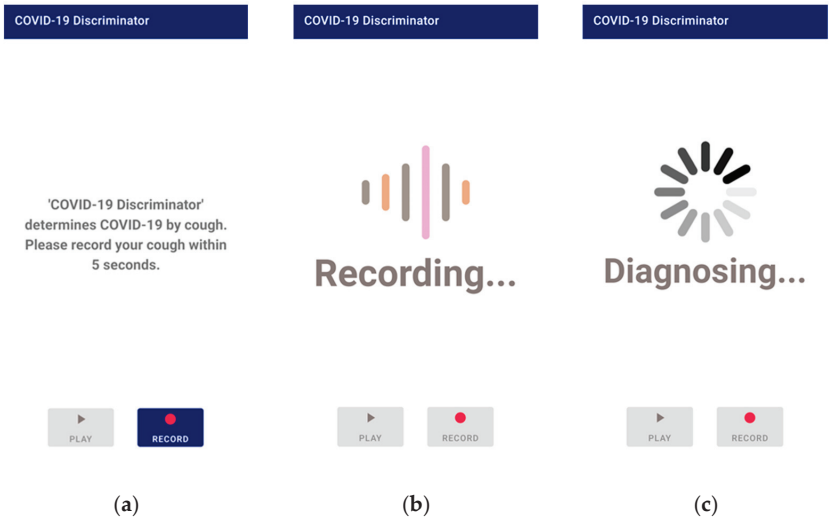


Figure 5. Execution process of the application.

When the application is executed and recorded, the recording is transmitted to the server using Transmission Control Protocol (TCP)/Internet protocol (IP) socket communication. The user's voice is recorded as a binary pulse-code modulation (PCM) file using the Android AudioRecord API [32]. In order for the user to play and listen to the recorded file in the application, the AudioTrack API [33] provided by Android was used. The recording format is designated as a sampling rate of 48 kHz, stereo channels, PCM 16 bit. The main screen of application, the screen during recording, and the screen when the recorded voice is transmitted to the server and processed are all shown in Figure 6.



(**a**) (**b**) (**c**)

Figure 6. Application screen 1: (**a**) the main screen; (**b**) the screen during recording; (**c**) the screen during processing.

After receiving the data, the server converts it to a mono-channel WAV file, which is the same format used by the database data used in this study. Then, the cough segment is extracted through the preprocessing process described in Section 3.1. The extracted cough segments are input into the trained model to measure a COVID-19 diagnosis prediction value, and the result is transmitted to the application. There are three types of results: positive, negative, and retry. A retry occurs when a cough is not detected during preprocessing. The application shows the diagnosis result screen based on the results transmitted from the server. Figure 7 shows a screen displaying diagnostic results from the application.

(**a**) (**b**) (**c**)

Figure 7. Application screen 2: (**a**) positive result; (**b**) negative result; (**c**) if no cough is detected.

6. Conclusions

In this study, we proposed a COVID-19 diagnostic model and its application based on an artificial intelligence (AI) model with optimized feature vectors using cough sounds. The Bhattacharyya distance was used to measure the separability of features from COVID-19-positive cough data and negative cough data. MFCC, Δ-MFCC, Δ^2-MFCC, spectral contrast, chroma, spectral flatness, spectral bandwidth, spectral roll-off, RMS energy, spectral centroid, zero-crossing rate, and onset showed high values, in that order. The highest-valued MFCC had a value of 0.207171. Subsequently, Δ^2-MFCC had a value of 0.149195, Δ-MFCC had a value of 0.099828, and spectral contrast had a value of 0.090616. These top four features made up the feature set that this study proposed. After training the combined ResNet-50 and DNN model, the result had an accuracy of 0.96, a sensitivity of 0.95, a specificity of 0.96, and a precision of 0.95. Using this model, an application for Android was developed so that many people could use it for COVID-19 testing. The COVID-19 test model using cough sounds, the result of this study, has a simpler procedure and lower cost than the polymerase chain reaction (PCR) test that analyzes genes. Moreover, it is expected that this application will be a useful tool for those who are unable to do a PCR test, as it is difficult to insert a cotton swab into the nasopharynx due to anatomical or medical issues. In future studies, the model can be upgraded by using not only cough sound data but also clinical information data, including information on fever, headache, and other symptoms. In addition, if more quality cough sound data are collected and utilised, improved results can be expected.

Author Contributions: Conceptualization, S.-P.L.; methodology, S.K. and J.-Y.B.; investigation, S.K. and J.-Y.B.; writing—original draft preparation, S.K.; writing—review and editing, S.-P.L.; project administration, S.-P.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Sangmyung University.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Experiments used publicly available datasets.

Acknowledgments: This work was supported by the 2022 Research Grant from Sangmyung University.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviation

AI	Artificial intelligence
ANN	Artificial neural network
ANOVA	Analysis of variance
AUC	Area under the ROC curve
CT	Computed tomography
DNN	Deep neural network
FN	False negative
FP	False positive
IP	Internet protocol
KNN	K-Nearest neighbors
MFCC	Mel frequency cepstral coefficients
PCM	Pulse code modulation
PCR	Polymerase chain reaction
RMS	Root mean square
RNN	Recurrent neural network
ROC	Receiver operating characteristic
RT-PCR	Reverse transcription polymerase chain reaction
SVM	Support vector machine
TCP	Transmission control protocol
TN	True negative
TP	True positive

References

1. IHME COVID-19 Forecasting Team. Modeling COVID-19 scenarios for the United States. *Nat. Med.* **2021**, *27*, 94–105. [CrossRef]
2. Ma, W.; Zhao, Y.; Guo, L.; Chen, Y.Q. Qualitative and quantitative analysis of the COVID-19 pandemic by a two-side fractional-order compartmental model. *ISA Trans.* **2022**, *124*, 144–156. [CrossRef] [PubMed]
3. Baleanu, D.; Mohammadi, H.; Rezapour, S. A fractional differential equation model for the COVID-19 transmission by using the Caputo–Fabrizio derivative. *Adv. Differ. Equ.* **2020**, *299*, 1–27. [CrossRef] [PubMed]
4. Tahamtan, A.; Ardebili, A. Real-time RT-PCR in COVID-19 detection: Issues affecting the results. *Expert Rev. Mol. Diagn.* **2020**, *20*, 453–454. [CrossRef]
5. Luz, E.; Silva, P.; Silva, R.; Silva, L.; Guimarães, J.; Miozzo, G.; Moreira, G.; Menotti, D. Towards an effective and efficient deep learning model for COVID-19 patterns detection in X-ray images. *Res. Biomed. Eng.* **2022**, *38*, 149–162. [CrossRef]
6. Alshazly, H.; Linse, C.; Barth, E.; Martinetz, T. Explainable COVID-19 detection using chest CT scans and deep learning. *Sensors* **2021**, *21*, 455. [CrossRef] [PubMed]
7. Sakr, R.; Ghsoub, C.; Rbeiz, C.; Lattouf, V.; Riachy, R.; Haddad, C.; Zoghbi, M. COVID-19 detection by dogs: From physiology to field application—A review article. *Postgrad. Med. J.* **2022**, *98*, 212–218. [CrossRef] [PubMed]
8. Quer, G.; Radin, J.M.; Gadaleta, M.; Baca-Motes, K.; Ariniello, L.; Ramos, E.; Kheterpal, V.; Topol, E.J.; Steinhubl, S.R. Wearable sensor data and self-reported symptoms for COVID-19 detection. *Nat. Med.* **2021**, *27*, 73–77. [CrossRef]
9. Gorji, F.; Shafiekhani, S.; Namdar, P.; Abdollahzade, S.; Rafiei, S. Machine learning-based COVID-19 diagnosis by demographic characteristics and clinical data. *Adv. Respir. Med.* **2022**, *90*, 171–183. [CrossRef]
10. Agbley, B.L.Y.; Li, J.; Haq, A.; Cobbinah, B.; Kulevome, D.; Agbefu, P.A.; Eleeza, B. Wavelet-based cough signal decomposition for multimodal classification. In Proceedings of the 2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), Chengdu, China, 18–20 December 2020; IEEE: Piscataway, NJ, USA, 2021; pp. 5–9. [CrossRef]
11. Laguarta, J.; Hueto, F.; Subirana, B. COVID-19 artificial intelligence diagnosis using only cough recordings. *IEEE Open J. Eng. Med. Biol.* **2020**, *1*, 275–281. [CrossRef]
12. Coppock, H.; Gaskell, A.; Tzirakis, P.; Baird, A.; Jones, L.; Schuller, B. End-to-end convolutional neural network enables COVID-19 detection from breath and cough audio: A pilot study. *BMJ Innov.* **2021**, *7*, 356–362. [CrossRef] [PubMed]
13. Chetupalli, S.R.; Krishnan, P.; Sharma, N.; Muguli, A.; Kumar, R.; Nanda, V.; Pinto, L.M.; Ghosh, P.K.; Ganapathy, S. Multi-modal point-of-care diagnostics for COVID-19 based on acoustics and symptoms. *arXiv* **2021**, arXiv:2106.00639. [CrossRef]
14. Mohammed, E.A.; Keyhani, M.; Sanati-Nezhad, A.; Hejazi, S.H.; Far, B.H. An ensemble learning approach to digital corona virus preliminary screening from cough sounds. *Sci. Rep.* **2021**, *11*, 15404. [CrossRef] [PubMed]
15. Tris Atmaja, B.; Sasou, A. Cross-dataset COVID-19 Transfer Learning with Cough Detection, Cough Segmentation, and Data Augmentation. *arXiv* **2022**, arXiv:2210.05843. [CrossRef]

16. Mahanta, S.K.; Kaushik, D.; Van Truong, H.; Jain, S.; Guha, K. COVID-19 diagnosis from cough acoustics using convnets and data augmentation. In Proceedings of the 2021 First International Conference on Advances in Computing and Future Communication Technologies (ICACFCT), Meerut, India, 16–17 December 2021; IEEE: Piscataway, NJ, USA, 2022; pp. 33–38. [CrossRef]
17. Sunitha, G.; Arunachalam, R.; Abd-Elnaby, M.; Eid, M.M.; Rashed, A.N.Z. A comparative analysis of deep neural network architectures for the dynamic diagnosis of COVID-19 based on acoustic cough features. *Int. J. Imaging Syst. Technol.* **2022**, *32*, 1433–1446. [CrossRef]
18. Sabet, M.; Ramezani, A.; Ghasemi, S.M. COVID-19 Detection in Cough Audio Dataset Using Deep Learning Model. In Proceedings of the 2022 8th International Conference on Control, Instrumentation and Automation (ICCIA), Tehran, Iran, 2–3 March 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1–5. [CrossRef]
19. Arif, A.; Alanazi, E.; Zeb, A.; Qureshi, W.S. Analysis of rule-based and shallow statistical models for COVID-19 cough detection for a preliminary diagnosis. In Proceedings of the 2022 13th Asian Control Conference (ASCC), Jeju, Republic of Korea, 4–7 May 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 465–469. [CrossRef]
20. Brown, C.; Chauhan, J.; Grammenos, A.; Han, J.; Hasthanasombat, A.; Spathis, D.; Xia, T.; Cicuta, P.; Mascolo, C. Exploring automatic diagnosis of COVID-19 from crowdsourced respiratory sound data. *arXiv* **2020**, arXiv:2006.05919. [CrossRef]
21. Feng, K.; He, F.; Steinmann, J.; Demirkiran, I. Deep-learning based approach to identify COVID-19. In Proceedings of the Southeast Conference 2021, Atlanta, GA, USA, 10–13 March 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–4. [CrossRef]
22. Fakhry, A.; Jiang, X.; Xiao, J.; Chaudhari, G.; Han, A. A multi-branch deep learning network for automated detection of COVID-19. In Proceedings of the 22nd Annual Conference of the International Speech Communication Association 2021, Brno, Czechi, 30 August–3 September 2021; pp. 3641–3645. [CrossRef]
23. Son, M.J.; Lee, S.P. COVID-19 Diagnosis from Crowdsourced Cough Sound Data. *Appl. Sci.* **2022**, *12*, 1795. [CrossRef]
24. Muda, L.; Begam, M.; Elamvazuthi, I. Voice recognition algorithms using mel frequency cepstral coefficient (MFCC) and dynamic time warping (DTW) techniques. *arXiv* **2010**, arXiv:1003.4083. [CrossRef]
25. Sharma, N.; Krishnan, P.; Kumar, R.; Ramoji, S.; Chetupalli, S.R.; Ghosh, P.K.; Ganapathy, S. Coswara–a database of breathing, cough, and voice sounds for COVID-19 diagnosis. *arXiv* **2020**, arXiv:2005.10548. [CrossRef]
26. Chaudhari, G.; Jiang, X.; Fakhry, A.; Han, A.; Xiao, J.; Shen, S.; Khanzada, A. Virufy: Global applicability of crowdsourced and clinical datasets for AI detection of COVID-19 from cough. *arXiv* **2020**, arXiv:2011.13320. [CrossRef]
27. Orlandic, L.; Teijeiro, T.; Atienza, D. The COUGHVID crowdsourcing dataset, a corpus for the study of large-scale cough analysis algorithms. *Sci. Data* **2021**, *8*, 156. [CrossRef] [PubMed]
28. Kailath, T. The divergence and Bhattacharyya distance measures in signal selection. *IEEE Trans. Commun. Technol.* **1967**, *15*, 52–60. [CrossRef]
29. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
30. Dan, E. Github. Available online: https://github.com/tensorflow/models/tree/master/research/audioset/vggish (accessed on 9 June 2021).
31. Librosa. Available online: https://librosa.org (accessed on 13 December 2022).
32. Android Developers. Available online: https://developer.android.com/reference/android/media/AudioRecord (accessed on 13 July 2022).
33. Android Developers. Available online: https://developer.android.com/reference/android/media/AudioTrack (accessed on 13 July 2022).

applied sciences

MDPI

Article

Investigation of Machine Learning Model Flexibility for Automatic Application of Reverberation Effect on Audio Signal

Mantas Tamulionis †, Tomyslav Sledevič † and Artūras Serackis *,†

Department of Electronic Systems, Vilnius Gediminas Technical University (VILNIUS TECH), Plytinės g. 25, LT-10105 Vilnius, Lithuania; mantas.tamulionis@vilniustech.lt (M.T.); tomyslav.sledevic@vilniustech.lt (T.S.)
* Correspondence: arturas.serackis@vilniustech.lt
† These authors contributed equally to this work.

Abstract: This paper discusses an algorithm that attempts to automatically calculate the effect of room reverberation by training a mathematical model based on a recurrent neural network on anechoic and reverberant sound samples. Modelling the room impulse response (RIR) recorded at a 44.1 kHz sampling rate using a system identification-based approach in the time domain, even with deep learning models, is prohibitively complex and it is almost impossible to automatically learn the parameters of the model for a reverberation time longer than 1 s. Therefore, this paper presents a method to model a reverberated audio signal in the frequency domain. To reduce complexity, the spectrum is analyzed on a logarithmic scale, based on the subjective characteristics of human hearing, by calculating 10 octaves in the range 20–20,000 Hz and dividing each octave by 1/3 or 1/12 of the bandwidth. This maintains equal resolution at high, mid, and low frequencies. The study examines three different recurrent network structures: LSTM, BiLSTM, and GRU, comparing the different sizes of the two hidden layers. The experimental study was carried out to compare the modelling when each octave of the spectrum is divided into a different number of bands, as well as to assess the feasibility of using a single model to predict the spectrum of a reverberated audio in adjacent frequency bands. The paper also presents and describes in detail a new RIR dataset that, although synthetic, is calibrated with recorded impulses.

Keywords: room reverberation; room impulse response; recurrent neural networks; audio signal spectrum; filter bank

Citation: Tamulionis, M.; Sledevič, T.; Serackis, A. Investigation of Machine Learning Model Flexibility for Automatic Application of Reverberation Effect on Audio Signal. *Appl. Sci.* **2023**, *13*, 5604. https://doi.org/10.3390/app13095604

Academic Editors: Giovanni Costantini and Daniele Casali

Received: 28 March 2023
Revised: 24 April 2023
Accepted: 26 April 2023
Published: 1 May 2023

1. Introduction

The room impulse response, which represents the acoustic properties of the room, is widely used in a broad range of audio signal-processing tasks. RIR can be useful for sound source localisation [1], speech recognition [2], or speech signal separation [3]. If the room being analyzed is characterized by unwanted acoustic phenomena, the measured RIR can show spectral changes [4]. These changes can be eliminated by an equalization scheme. The influence of room acoustic characteristics on the RIR spectrum can vary depending on the location of the measurement, so the equalization scheme must be adaptive. Such adaptive equalization schemes typically use an FIR filter whose attenuation coefficients are continuously updated to reduce the difference between the spectrum actually obtained at the measurement position and the desired spectrum [5]. However, the flexibility of the filter depends on the filter order and the coefficient estimation algorithms.

Updating the attenuation coefficients of the FIR filter is usually performed using the filtered-x least mean square (FxLMS) algorithm [6]. However, it was later discovered that this algorithm is not stable and can cause sudden interference in its error signal [7]. Subsequent studies have proposed the use of the maximum correntropy criterion (MCC) method for adaptive filtering, which has been shown to be more robust than previously popular methods [8]. Even later, the generalised maximum correntropy criterion (GMCC)

method was proposed and performed better than the standard MCC [9]. The RIR impulse is observed to be a sparse set of coefficients, i.e., some of its intermediate values are close to zero. On this basis, it was decided that the equalization process could be further improved if the adaptive algorithm took advantage of the sparseness of the RIR impulse [10].

To create the impression of realistic room acoustics as the listener's position varies in the virtual room, we need to continuously convolve an anechoic signal in real time with a different RIR filter from a large dataset. The dataset should consist of RIRs recorded in a real room, but this is time consuming, as each new RIR requires a new measurement when a new position is chosen for the sound source or receiver. This means that, for example, to capture a set of RIR data covering the entire area of a small room (up to 10 sq. m.), more than 1000 measurements may be required, with the position of the measuring microphone changing every 10 cm. In addition, a quiet environment is needed to ensure the quality of the RIR measurements. According to ISO 3382-1, the sound source must emit a sound pressure level at least 35 dB above the background noise in the room [11].

As an alternative to RIR measurements, RIR can be modeled using one of the geometric acoustics methods. The most commonly used one is the image source method (ISM) [12]. However, satisfactory results can only be achieved in this way by modelling an almost empty room with clear geometry (e.g., a rectangle). The ISM method is a simplified assumption that sound waves propagate in straight lines at a fixed speed, the energy is uniformly attenuated, and the waves are mirrored when they reach a surface. In the real world, the sound wave is not perfectly reflected; some of it is scattered in different directions, depending on the roughness of the surface. Only the early reflections are mirror-like, and later they become increasingly diffuse. Thus, in practice, a hybrid approach is often used, where the first reflections are modeled by an ISM and the later ones by the ray-tracing method. The ISM method also does not allow the modelling of objects in the room that interfere with the propagation of the sound wave and cause reflections. Tang et al. proposed improvements using a Monte Carlo path-tracing method that can model diffuse reflections, which means better simulation of existing obstacles [13]. However, the authors point out that this algorithm also has the disadvantage of not being able to model low frequencies and diffraction well. There have been attempts to to solve the problem with the use of artificial neural networks which, trained on existing RIRs, can predict the desired data.

The use of neural networks can be a more flexible approach and a good alternative to this task. The RIR can be estimated using its spectrogram as an image, as well as individual RIR parameters such as the geometry of the simulated room and the absorption coefficients of its surfaces. In the study by Yu and Kleijn, the RIR parameters were estimated separately, with convolutional neural networks (CNNs) used for room geometry and feedforward multilayer perceptrons (MLPs) for surface absorption coefficients [14]. The authors claim that their method works when neural networks are trained with a single RIR impulse. In fact, it should be noted that this condition is only partially fulfilled, as the algorithm is initially allowed to learn from a single simulated RIR impulse that has been generated by the ISM method using the RIR generator [15]. Afterwards, it has been shown that much better results can be achieved by increasing the number of RIRs dedicated to training. In addition, the performance of the algorithm is tested by training the networks on the recorded RIRs. The BUT ReverbDB dataset is used for this purpose [16].

Machine learning methods are applied not only to RIR generation but also to other acoustic environment analysis tasks. Classification of rooms by volume using RIR can be performed using statistical pattern recognition [17]. The authors of this paper claim that their algorithm does not require data about the distance between the sound source and the microphone. However, good results were only achieved using simulated rather than measured RIRs. Convolutional neural networks are used to perform speech recognition tasks and to build speech-to-text models. In [18], the authors used a CNN-based approach to recognise tonal speech signals. Feature extraction was performed using Mel frequency cepstral coefficients (MFCC). Machine learning can also be used to assess the competence

of psychotherapists by performing speech recognition from audio and text analysis from a report together. In [19], the possibility of determining the quality of a practitioner's performance by analysing audio recordings and transcripts of psychotherapeutic conversations and comparing the result with manual assessments of competency was explored. The best predictive performance was achieved by a Lasso regression model. In [20], the authors used time-domain features (MFCCT) in addition to MFCCs in speech emotion recognition (SER) to extract features from an audio signal. The CNN-based SER model outperformed comparable models that used non-hybrid features. Machine learning is also being applied in the field of tourism to generate additional recommendations for destinations with fewer reviews on specialised tourism portals. Missing reviews can be identified and selected from social media posts containing geolocation information. In [21], the authors used machine learning-based clustering and classification methods, namely a fine-tuned transformer neural network-based BERT model.

In this paper, we present a new dataset for RIR estimation based on the fusion of recorded and simulated RIRs. In addition, we present a study of an alternative method for modeling the spectrum of a reverberated signal. The idea of this paper is to check if a neural network can learn the effects of acoustics and replace the traditional method of using RIR filters. We train the neural network with frequency-domain data, dividing logarithmically into 1/3 or 1/12 octave. The studies test the feasibility of modeling reverberated audio for several different frequency bands by training a model for only one band, thus trying to avoid the need to train different models for each frequency band separately.

We have chosen recurrent neural networks (RNNs) for this task because they could be good for modeling reverberating audio, as they are designed to handle sequential data, allowing them to account for the time-varying nature of audio signals, and their internal memory cells can effectively capture the dependencies between successive audio samples, leading to a more accurate representation of reverberation characteristics. The bidirectional LSTM, LSTM, and GRU recurrent neural network architectures offers unique strengths and trade-offs in terms of modeling capacity, computational efficiency, and memory requirements, and a thorough evaluation can help identify the most suitable approach for capturing the complex temporal relationships present in reverberating audio signals, ultimately leading to better performance and practical applicability.

The structure of the article is as follows: Section 2 presents the dataset and the methods used in our study. The preparation of the dataset is described in detail in Section 2.1. Section 2.2 provides a detailed explanation of our method, which compared three recurrent neural network structures that attempted to predict room reverberation for each octave band. Section 3 describes our experimental setup and a comparison of the reverberation prediction results using different recurrent neural network models. Section 4 provides a discussion and concludes the results of our study.

2. Materials and Methods

2.1. Preparation of the Dataset

To train the algorithm properly, we need to create a large set of data samples, avoiding to record all RIRs as this would be time-consuming, but trying to maintain the authenticity of the RIR impulses. To achieve these goals, we decided to create a dataset of synthetic impulses, but based on the recorded RIRs. First, measurements were made in a university laboratory, choosing a small number of fixed measurement positions. Subsequently, an identical room was designed and imported into the "Odeon" acoustic design software. The acoustic parameters of the measured and modeled RIRs were compared and the absorption coefficients of the modeled room surfaces were changed accordingly. This allowed the creation of new synthetic RIRs that are authentic and correspond not only to several measured room positions but also to any selected point in the virtual room.

Measurements were taken in a small rectangular room. The main purpose of the room was to test the VR software, so it was almost empty; only three wooden tables remained after the computer screens were removed. The room has a floor area of 31.35 m^2 and a

ceiling height of 2.86 m. Three walls of the room are covered with large porous bricks, one wall is concrete and painted, and the ceiling is made up of small square plasterboards with aluminium gaps between them. The floor of the room is linoleum floored and access to the room is through a wide glass door.

Authentic room pulses were recorded according to ISO 3382-1 [11]. It is recommended to select and test at least two different sound source positions in the room (with a height of 1.5 m from the ground), as well as at least three to four microphone positions, which should be spaced at least 2 m (half the measured wavelength of the lowest frequency) apart, and at least 1 metre (a quarter of the wavelength of the lowest frequency) away from any reflecting surface. The different microphone positions should be chosen in such a way that the results take into account the reflections produced by all walls covered with different materials, and the height of the measuring microphone should be adjustable to 1.2 m, which corresponds to the typical height of the ear position of a seated listener. To maintain the distances specified in the standard, two positions of the sound source and three positions of the microphones were selected and tested, resulting in a total of six different combinations. The sound source and microphone positions are shown in Figure 1, as well as the grid of microphone positions used in the virtual version of this room. The standard also specifies that the sound source should be omnidirectional and should reproduce all frequencies uniformly between 125 Hz and 4000 Hz. However, to analyze the effect of room acoustics on human voice, these measurements were carried out using a directional loudspeaker, Genelec 8010A, whose directivity is compared to that of human speech in Figure 2 [22].

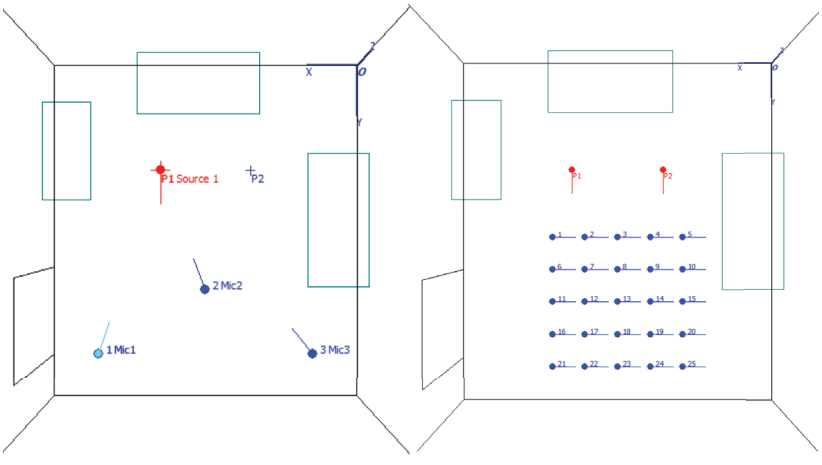


Figure 1. Left image shows the locations of the sound source and the measurement microphone in a real room, the different combinations of which were recorded as separate RIR impulses. Right image shows the selected positions of the 2 sources and 25 receivers respectively in the virtual room.

A sonarworks XREF 20 omnidirectional microphone and RME Fireface UC sound card were used as receiver and recorder. We also used the Measure impulse response tool offered by Odeon 16, which allows us to generate and transmit an exponential sweep signal and record the impulse.

The same room was then modeled in SketchUp and imported into Odeon. With the same positions for the sound sources and microphones, as well as the assumed absorption coefficients for the surfaces, the RIR simulation was performed. Odeon allows the technical characteristics of a real loudspeaker—directivity, frequency response, dynamic range, etc.—to be assigned to a virtual sound source. The user has to import and activate a CLF (common loudspeaker format) file, which can be downloaded for each specific model of almost all popular loudspeaker manufacturers.

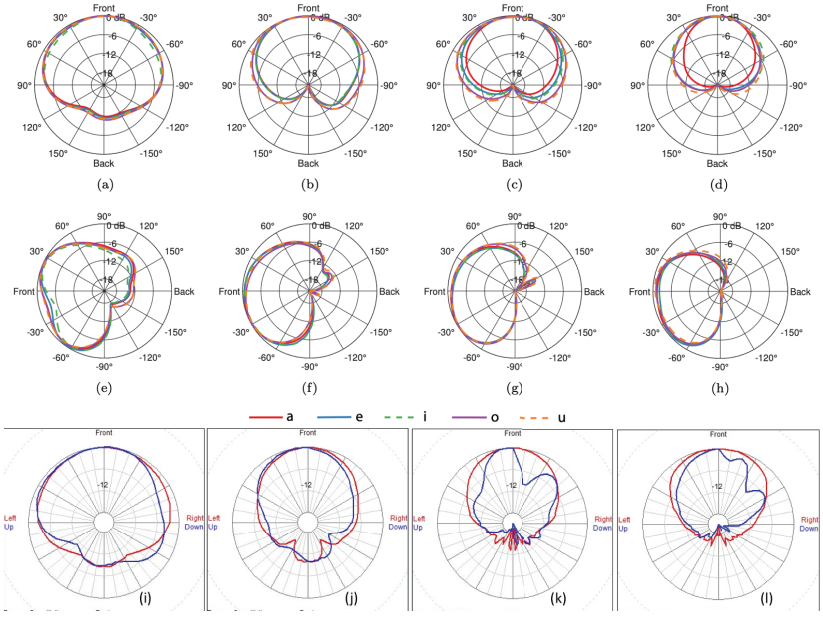


Figure 2. Comparison of the directionality of the human voice in different vowels (**a**–**h**) with the directionality of the Genelec 8010A loudspeaker used in the measurements (**i**–**l**), in the horizontal ((**a**–**d**) and red line in (**i**–**l**)) and in the vertical planes ((**e**–**h**) and blue line in (**i**–**l**)) in different bands: 1 kHz (**a**,**e**,**i**), 2 kHz (**b**,**f**,**j**), 4 kHz (**c**,**g**,**k**), and 8 kHz (**d**,**h**,**l**).

Odeon has the ability to import recorded pulses and compare them with simulated ones. The accuracy of the results depends on the precise choice of the surface absorption coefficients, and initially the results varied considerably. Another Odeon tool, "Genetic Material Optimizer", was then used [23]. It compares the characteristics of the recorded and simulated pulses and tries to recalculate the possible absorption coefficients. Before running the algorithm, it is necessary to select the permissible limits of variation of the absorption coefficient for each material. For porous bricks and plasterboard, we have set a higher modification limit. These materials cover 3 walls and the ceiling; in general, most of the room surface. We can see that the algorithm only slightly changed the absorption coefficient of the materials with a modification limit of 50%, whereas the absorption coefficient of the materials with a higher modification limit was changed in detail.

The differences between the recorded and simulated impulses are evaluated by the JND (just-noticeable difference) value [24], which is also described in the ISO standard and corresponds to 1 dB for most acoustic evaluation parameters. This means that if the difference between the impulses is less than 1 JND, it can be assumed to be negligible and can be ignored. Before the algorithm was run, this value ranged from 13 to 15 JND in the individual frequency bands; after optimization it ranged from 0.7 to 3. Only in the lower frequency bands did the differences remain larger, but the developers of Odeon warn the user that the algorithm is not able to reduce the differences to below 1 JND in the lower frequency bands. Once the absorption coefficients have been optimized and the differences between the simulated and recorded impulse parameters have been verified to be within acceptable limits, it can be said that we have simulated the acoustics of a virtual room that closely matches the acoustics of a real room. In this case, we can create RIRs not only for the 3 fixed measurement locations but also for any point in the virtual room. In Figure 3 a comparison of the measured and simulated RIRs can be seen in terms of the early decay time before and after optimisation of the absorption coefficients. Figure 4 shows

the similarity of the spectrum of the human voice when such a signal is convolved with a measured or simulated RIR impulse.

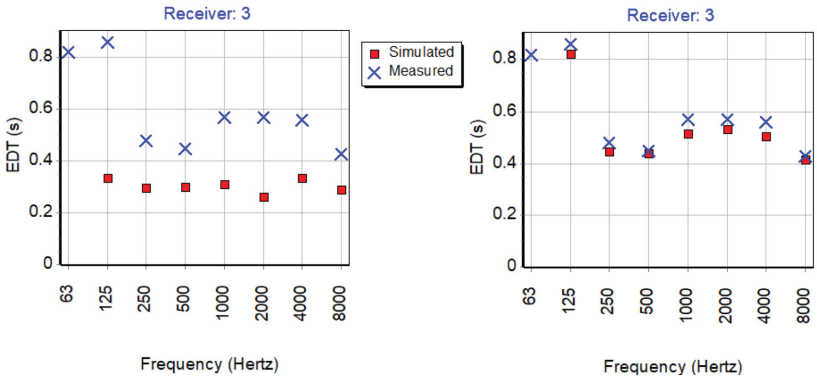


Figure 3. The recorded and simulated RIRs were compared with early decay time (EDT) values in different frequency bands before optimization (**left**) and after optimization (**right**).

Using the methodology described above, 50 RIRs were created for this study from 2 source positions and 25 receiver positions spaced 0.5 m apart. Using our calibrated virtual room model, we can create a larger dataset if necessary. Most importantly, the simulation is realistic, validated by real records. This makes any new study more valuable, as newly implemented models can be trained and tested on real acoustic behaviour, rather than on a dataset that is usually built using simplified models in an environment that will never be close to a real room. The latest version of the described dataset and more detailed technical information can be found in https://github.com/tamulionism/Room-Impulse-Response-dataset, accessed on 30 April 2023.

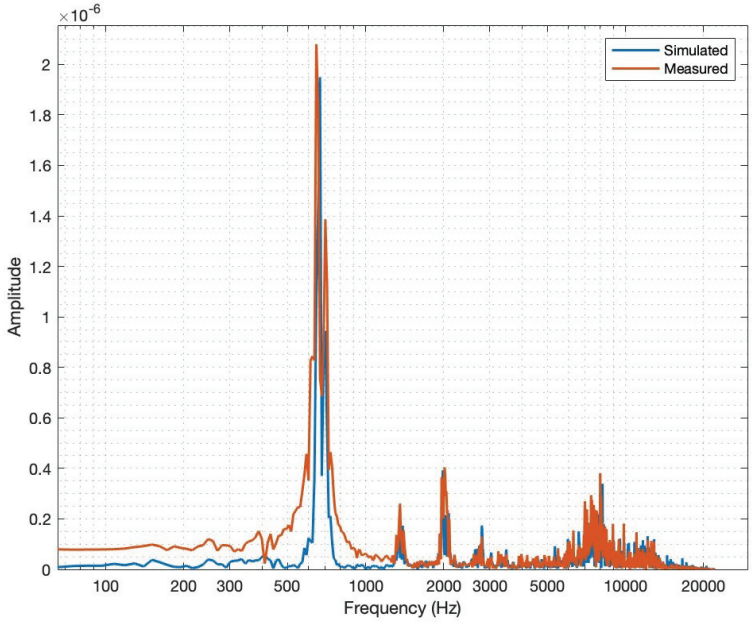


Figure 4. Comparison of the spectra of the anechoic voice signal of a singing woman convolved with an Odeon simulated RIR and an RIR impulse measured in a real room.

2.2. Deep Recurrent Neural Networks for Reverberated Signal Modeling

Three slightly different recurrent neural network structures were compared, which could be used as candidates for a reverberation prediction model:

- Long short-term memory (LSTM) [25];
- Bi-directional long short-term memory (BiLSTM) [26];
- Gated recurrent units (GRU) [27].

The architecture of the recurrent neural networks (RNN) includes feedback connections, making them more suitable for modeling acoustic effects than feed-forward network structures.

We try to predict the spectrum of the reverberated signal separately for each octave. In our study, we test all three neural network structures by training them on different frequency bands of the reverberated signal. Each prediction model consists of an input layer to which a sequence of time-varying spectral node values is sent, as well as two layers of recurrent neural network cells, one fully connected layer, and a regression layer, which generate a predicted sequence of changes in the spectral nodes over time.

To investigate the relationship between the number of RNA cells in a layer and the accuracy of the predicted spectral band, we tested the performance of networks with three different combinations of cell numbers. We first selected 10 cells in the first layer and 20 cells in the second layer, then repeated the experiments by equalling the number of cells in the two layers to 20, and finally we performed another series of experiments by increasing the number of cells in the second layer to 40.

In the experiments, we try to evaluate the ability of different network structures to predict a reverberated signal:

1. In the same frequency band used in the training, but replacing the input samples with previously unknown ones;
2. In two adjacent frequency bands, when the model was trained on the middle band and tested on adjacent bands
3. In all frequency bands when the octave is divided into 12 parts. Firstly, when a separate model was trained to predict each frequency band, and secondly, when the prediction was performed by taking input data from each frequency band separately, and the reverberated signal was predicted using a model trained on only one frequency band.

Variations of the experimental set-up were carried out to determine how flexible the prediction model can be to predict a specific band of the reverberated signal. In addition, it was necessary to see how different the model should be for neighboring frequency bands when the octave is divided into 3 or 12 parts.

The audio used for model training and experimental testing was divided into 250 ms analysis frames. This is the maximum delay time that can be accepted in real-time auralization systems [28]. The conversion from time to frequency domain was performed using a window of 512 sample width with 256 sample overlaps.The data used to train the models were divided into three parts: training (70%) validation (15%) and testing (15%). All models were trained using the same training options: the ADAM optimizer, a constant learning rate of 0.001, shuffle of the data after each epoch from 10,000, and a small batch size of 50.

3. Results

Table 1 presents the results of an experimental study where we used different RNN types (bidirectional LSTM, LSTM and GRU) architectures to simulate the reverberated signal in a single frequency band. The aim of this study was to investigate which RNN architecture can be used for reverberant signal modeling and how the size of the recurrent layers affects the results.

As can be seen from Table 1, RNN structures with more parameters, such as LSTM and BiLSTM, show a stable increase in R-squared as we increase the size of the hidden layer (the number of recurrent unit cells in the layer). The GRU-based RNN structure showed

unstable results after training, so that in some experimental studies the GRU-based model was not used at all (see Table 2).

Table 1. Comparison of different RNN structures by varying the number of cells in the layers of the recurrent neural network.

RNN	Layer Size	SSE (10^3)	RSS [min max]	R-Squared	RMSE [min max]
LSTM	10 + 20	15.85	[0.24 65.29]	−0.13	[3.36×10^{-5} 1.46]
	20 + 20	3.67	[0.23 37.39]	0.74	[2.24×10^{-5} 0.66]
	20 + 40	1.36	[0.19 26.03]	**0.90**	[1.57×10^{-5} 0.56]
BiLSTM	10 + 20	6.43	[0.15 43.44]	0.54	[1.18×10^{-4} 0.81]
	20 + 20	2.71	[0.17 22.16]	0.81	[7.09×10^{-6} 0.37]
	20 + 40	2.16	[0.17 20.38]	**0.85**	[2.06×10^{-5} 0.38]
GRU	10 + 20	5.05	[0.32 29.97]	0.64	[1.30×10^{-4} 0.60]
	20 + 20	2.45	[0.21 34.03]	**0.83**	[6.37×10^{-5} 0.60]
	20 + 40	3.31	[0.18 32.59]	0.77	[7.99×10^{-5} 0.66]

Table 2. Comparison of different RNN structures with fixed layer sizes, trained on a single bin, covering 1/12 of the octave band width. Tested on 12 neighboring bins.

RNN	Bin Number	SSE (10^4)	RSS (Mean)	R-Squared	RMSE (Mean)
LSTM	1	30.5	9.59	0.63	0.1621
	2	52.6	12.54	0.37	0.2272
	3	75.6	14.45	0.27	0.2703
	4	84.3	14.74	0.27	0.2793
	5	65.4	14.10	0.32	0.2545
	6	17.6	9.16	0.57	0.1337
	7	1.72	2.95	**0.92**	0.0191
	8	12.8	4.78	0.70	0.0523
	9	106	5.69	0.33	0.0881
	10	129	6.92	0.27	0.1177
	11	117	6.55	0.35	0.1060
	12	147	6.99	0.39	0.1091
BiLSTM	1	15.9	8.48	0.81	0.1788
	2	13.0	9.71	0.84	0.2240
	3	15.1	10.55	0.85	0.2614
	4	18.3	10.71	0.84	0.2749
	5	15.7	10.77	0.84	0.2709
	6	8.1	7.28	0.80	0.1507
	7	1.10	1.94	**0.95**	0.0145
	8	10.4	4.22	0.76	0.0489
	9	86.8	5.09	0.45	0.0821
	10	104	4.76	0.41	0.1321
	11	103	4.84	0.43	0.1218
	12	114	6.28	0.52	0.0817

To compare the flexibility of the selected RNNs in learning individual frequency bands of the reverberated signal, we trained 30 structures (each of the 10 octaves of the human audible frequency spectrum was divided into three bands). We used RNN models with 20 cells in the first hidden layer and 40 cells in the second hidden layer, which is the largest structure studied in the first experiment and which showed the best fitting results.

Tables 3 and 4 show the results of an experimental study to test whether a model trained to predict the central band of an octave divided into three parts is good enough to predict adjacent frequency bands. We compared the results for 8 different octaves, ignoring only the first and last octaves—frequencies below 40 Hz and above 10 kHz. A noticeable reduction of fit was observed. We can also see from the results that the use of the central frequency band model to predict neighboring frequency bands also depends on the octave

chosen. This is an expected result, as we cannot normally achieve a uniform distribution of sound content across all octaves in any real recording dataset.

Table 3. R-squared fitting estimate comparison of LSTM trained on a single bin, tested on neighboring ones, using resolution of 1/3 of the octave band width.

R-Squared	5 bin	8 bin	11 bin	14 bin	17 bin	20 bin	23 bin	26 bin
Bin at the Left	0.72	0.64	0.68	0.90	0.61	0.13	0.76	0.85
Bin at the Center	0.78	0.84	0.87	0.98	0.95	0.95	0.87	0.88
Bin at the Right	0.65	0.74	0.71	0.79	0.30	0.62	0.75	0.87

Table 4. RMSE fitting estimate comparison of LSTM trained on a single bin, tested on neighboring ones, using resolution of 1/3 of the octave band width.

RMSE (Mean)	5 bin	8 bin	11 bin	14 bin	17 bin	20 bin	23 bin	26 bin
Bin at the Left	0.0971	0.1375	0.1491	0.0680	0.0742	0.3782	0.0746	0.0678
Bin at the Center	0.0777	0.0680	0.0687	0.0182	0.0189	0.0260	0.0367	0.0484
Bin at the Right	0.1644	0.1100	0.1335	0.1206	0.1737	0.2157	0.0940	0.0531

By dividing each octave into 12 parts, we can analyze the half-tone pattern of the reverberated signal. In this part of the study, we first decided to compare the ability to learn from samples for each frequency band separately. The results are shown in Table 5. We again trained three RNN structures with layer sizes of 20 + 40 RNN cells in two hidden layers, respectively. The LSTM and BiLSTM-based models showed relatively stable results, but the GRU-based RNN was difficult to train to be close to matching all 12 frequency bands.

Table 5. Comparison of different RNN structures with fixed layer sizes, trained on a single bin, covering 1/12 of the octave band width. 12 trained models in total, for neighboring bins.

RNN	Bin Number	SSE (10^3)	RSS (Mean)	R-Squared	RMSE (Mean)
LSTM	1	30.42	3.59	0.96	0.0227
	2	70.52	4.41	0.91	0.0325
	3	59.89	4.89	0.94	0.0354
	4	110.75	5.62	0.90	0.0483
	5	117.16	5.53	0.88	0.0452
	6	38.58	4.55	0.91	0.0311
	7	17.16	2.95	0.92	0.0191
	8	19.62	2.93	0.95	0.0194
	9	23.79	2.18	0.98	**0.0113**
	10	245.23	1.94	0.86	0.0176
	11	315.59	8.44	0.82	0.1642
	12	16.87	2.13	**0.99**	0.0134
BiLSTM	1	38.78	3.05	0.95	0.0278
	2	62.71	3.30	0.92	0.0249
	3	29.26	3.46	0.97	0.0236
	4	36.86	3.94	0.97	0.0268
	5	47.86	3.82	0.95	0.0283
	6	19.48	3.19	0.95	0.0222
	7	10.99	1.94	0.95	0.0145
	8	10.95	1.92	0.97	0.0126
	9	10.47	1.55	0.99	0.0099
	10	36.03	1.14	0.98	0.0097
	11	39.34	1.25	0.98	0.0096
	12	4.25	1.57	**1.00**	**0.0092**

Table 5. *Cont.*

RNN	Bin Number	SSE (10^3)	RSS (Mean)	R-Squared	RMSE (Mean)
GRU	1	32.56	3.69	**0.96**	0.0241
	2	68.70	4.73	0.92	0.0365
	3	86.63	5.16	0.92	0.0430
	4	145.46	5.60	0.87	0.0527
	5	73.08	5.49	0.92	0.0409
	6	89.77	4.85	0.78	0.0405
	7	95.12	3.90	0.58	0.0418
	8	24.23	2.95	0.94	0.0216
	9	57.30	2.43	**0.96**	**0.0164**
	10	N/A	32.44	N/A	0.6615
	11	N/A	31.48	N/A	0.6381
	12	124.86	2.69	0.95	0.0210

For the last study, we chose the seventh band, which is in the middle of the twelve. The experimental results of the model trained for one frequency band and used to predict the reverberation of the other frequency bands are presented in Table 2. The GRU-based RNN model was not considered in this experimental study because the initial tests showed even worse fitting accuracy and the same trends as for the LSTM and BiLSTM-based RNNs.

4. Conclusions

This paper discusses the flexibility of a recurrent neural network to automatically compute a reverberated audio signal. The algorithm models the reverberation-affected signal in the frequency domain by analysing the spectrum on a logarithmic scale. The study examines three different recurrent network structures and compares the modeling of the reverberated signal when each octave of the spectrum is divided into a different number of bands. Using a model trained for one mid-octave band (No. 7) and tested as a model for applying the reverberation effect to the remaining 11 bands, it was found that even a half-tone change in the spectrum should be analyzed separately. To ensure a good prediction of the full spectrum of the reverberated signal, we need to train a separate one-dimensional RNN model for each band. This can be defined as a limitation of our proposed method.

The BiLSTM-based RNN has shown more stable results in part of the frequency spectrum. Considering that all models were trained using the same audio recordings, it can be concluded that this type of RNN is more flexible in adapting to frequency changes related to room reverberation. Future work may explore methods for consolidating these models or refining the architecture to achieve more efficient and scalable solutions for modeling reverberated audio across different frequency bands.

Author Contributions: Conceptualization and methodology, all authors; validation, M.T.; analysis, A.S.; writing—original draft preparation, M.T. and T.S.; supervision, A.S.; writing—review and editing, all authors. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mane, S.S.; Mali, S.G.; Mahajan, S.P. Localization of Steady Sound Source and Direction Detection of Moving Sound Source Using CNN. In Proceedings of the 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kanpur, India, 6–8 July 2019 .
2. Tang, Z.; Meng, H.Y.; Manocha, D. Low-Frequency Compensated Synthetic Impulse Responses for Improved Far-Field Speech Recognition. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 6974–6978.
3. Jenrungrot, T.; Jayaram, V.; Seitz, S.; Kemelmacher-Shlizerman, I. The Cone of Silence: Speech Separation by Localization. 2020. Available online: https://arxiv.org/abs/2010.06007 (accessed on 30 April 2023)
4. Bergner, J.; Preihs, S.; Hupke, R.; Peissig, J. A System for Room Response Equalization of Listening Areas Using Parametric Peak Filters. In Proceedings of the 2019 AES International Conference on Immersive and Interactive Audio (March 2019), York, UK, 27–29 March 2019.
5. Cecchi, S.; Carini, A.; Spors, S. Room Response Equalization—A Review. *Appl. Sci.* **2018**, *8*, 16. [CrossRef]
6. Fuster, L.; De Diego, M.; Azpicueta-Ruiz, L.A.; Ferrer, M. Adaptive Filtered-x Algorithms for Room Equalization Based on Block-Based Combination Schemes. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2016**, *24*, 1732–1745. [CrossRef]
7. Kurian, N.C.; Patel, K.; George, N.V. Robust Active Noise Control: An Information Theoretic Learning Approach. *Appl. Acoust.* **2017**, *117*, 180–184. [CrossRef]
8. He, Z.C.; Ye, H.H.; Li, E. An Efficient Algorithm for Nonlinear Active Noise Control of Impulsive Noise. *Appl. Acoust.* **2019**, *148*, 366–374. [CrossRef]
9. Zhao, J.; Zhang, H.; Wang, G. Fixed-Point Generalized Maximum Correntropy: Convergence Analysis and Convex Combination Algorithms. *Signal Process.* **2019**, *154*, 64–73. [CrossRef]
10. Kumar, K.; George, N.V. A Generalized Maximum Correntropy Criterion Based Robust Sparse Adaptive Room Equalization. *Appl. Acoust.* **2020**, *158*, 107036. [CrossRef]
11. *ISO 3382-1*; Acoustics—Measurement of Room Acoustic Parameters —Part 1: Performance Spaces. International Organization for Standardization: Geneva, Switzerland, 2009.
12. Allen, J.B.; Berkley, D.A. Image Method for Efficiently Simulating Small-Room Acoustics. *J. Acoust. Soc. Am.* **1979**, *65*, 943–950. [CrossRef]
13. Tang, Z.; Chen, L.; Wu, B.; Yu, D.; Manocha, D. Improving Reverberant Speech Training Using Diffuse Acoustic Simulation. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 6969–6973.
14. Yu, W.; Kleijn, W.B. Room Acoustical Parameter Estimation from Room Impulse Responses Using Deep Neural Networks. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2021**, *29*, 436–447. [CrossRef]
15. Habets, E. RIR Generator. 2010. Available online: https://www.audiolabs-erlangen.de/fau/professor/habets/software/rir-generator (accessed on 30 April 2023).
16. Szoke, I.; Skacel, M.; Mosner, L.; Paliesek, J.; Cernocky, J.H. Building and Evaluation of a Real Room Impulse Response Dataset. *IEEE J. Sel. Top. Signal Process.* **2019**, *13*, 863–876. [CrossRef]
17. Shabtai, N.R.; Zigel, Y.; Rafaely, B. Room Volume Classification from Room Impulse Response Using Statistical Pattern Recognition and Feature Selection. *J. Acoust. Soc. Am.* **2010**, *128*, 1155–1162. [CrossRef] [PubMed]
18. Dua, S.; Kumar, S.S.; Albagory, Y.; Ramalingam, R.; Dumka, A.; Singh, R.; Rashid, M.; Gehlot, A.; Alshamrani, S.S.; Alghamdi, A.S. Developing a Speech Recognition System for Recognizing Tonal Speech Signals Using a Convolutional Neural Network. *Appl. Sci.* **2022**, *12*, 6223. [CrossRef]
19. Attas, D.; Power, N.; Smithies, J.; Bee, C.; Aadahl, V.; Kellett, S.; Blackmore, C.; Christensen, H. Automated Detection of the Competency of Delivering Guided Self-Help for Anxiety via Speech and Language Processing. *Appl. Sci.* **2022**, *12*, 8608. [CrossRef]
20. Alluhaidan, A.S.; Saidani, O.; Jahangir, R.; Nauman, M.A. Speech Emotion Recognition through Hybrid Features and Convolutional Neural Network. *Appl. Sci.* **2023**, *13*, 4750. [CrossRef]
21. Silaa, V.; Masui, F.; Ptaszynski, M. A Method of Supplementing Reviews to Less-Known Tourist Spots Using Geotagged Tweets. *Appl. Sci.* **2022**, *12*, 2321. [CrossRef]
22. Pörschmann, C.; Arend, J.M. Analyzing the Directivity Patterns of Human Speakers. In Proceedings of the 46th DAGA, Hannover, Germany, 16–19 March 2020; pp. 1141–1144.
23. ODEON Room Acoustics Software User's Manual. Version 16. Available online: https://odeon.dk/download/Version17/OdeonManual.pdf (accessed on 30 April 2023) .
24. Bradley, J.S. Review of Objective Room Acoustics Measures and Future Needs. *Appl. Acoust.* **2011**, *72*, 713–720. [CrossRef]
25. Irie, K.; Tüske, Z.; Alkhouli, T.; Schlüter, R.; Ney, H. LSTM, GRU, Highway and a Bit of Attention: An Empirical Overview for Language Modeling in Speech Recognition. In Proceedings of the Interspeech 2016, San Francisco, CA, USA, 8–12 September 2016; pp. 3519–3523.
26. Kurata, G.; Audhkhasi, K. Improved Knowledge Distillation from Bi-Directional to Uni-Directional LSTM CTC for End-to-End Speech Recognition. In Proceedings of the 2018 IEEE Spoken Language Technology Workshop (SLT), Athens, Greece, 18–21 December 2018.

27. Alom, M.Z.; Taha, T.M.; Yakopcic, C.; Westberg, S.; Sidike, P.; Nasrin, M.S.; Hasan, M.; Van Essen, B.C.; Awwal, A.A. S.; Asari, V.K. A State-of-the-Art Survey on Deep Learning Theory and Architectures. *Electronics* **2019**, *8*, 292. [CrossRef]
28. Wenzel, E.M. Effect of increasing system latency on localization of virtual sounds. In Proceedings of the 16th International Conference: Spatial Sound Reproduction (March 1999), Arktikum, Finland, 10–12 April 1999.

applied sciences

MDPI

Article

Vacuum Cleaner Noise Annoyance: An Investigation of Psychoacoustic Parameters, Effect of Test Methodology, and Interaction Effect between Loudness and Sharpness

Serkan Atamer * and Mehmet Ercan Altinsoy

Chair of Acoustics and Haptics, Faculty of Electrical and Computer Engineering, TU Dresden, 01069 Dresden, Germany; ercan.altinsoy@tu-dresden.de
* Correspondence: serkan.atamer@tu-dresden.de

Abstract: The first aim of this paper was to determine the variability in the signal characteristics and psychoacoustic data of canister-type vacuum cleaners. Fifteen vacuum cleaners with different sound power levels, provided by the manufacturers, were selected as test units to calculate their acoustic and psychoacoustic parameters. The selection of the devices was based on an even distribution of the reported sound power levels. The investigated variability in the acoustic and psychoacoustic parameters on different vacuum cleaners was discussed to derive the common characteristics of canister-type vacuum cleaner noise. The derived common characteristics were compared with the those in the available literature on the noise generation mechanisms of vacuum cleaners. Based on these characteristics, prototypical vacuum cleaner noise was defined. The second aim of this paper was to understand the annoyance perception of vacuum cleaner noise. Annoyance assessments were obtained from two sets of listening experiments. The first listening experiment was conducted to find the correlates of annoyance evaluations. Loudness, sharpness and tonal components at lower and higher frequencies were found to be dominant correlates of vacuum cleaner noise annoyance estimations. In the second listening experiment, a possible interaction between loudness and sharpness was investigated in different listening test methods. The selected loudness and sharpness values for this experiment were consistent with the observed ranges in the first part. No significant interaction between loudness and sharpness was observed, although each separately correlated significantly positively with annoyance.

Keywords: sound quality; annoyance; perception; psychoacoustics; vacuum cleaner noise; household appliance noise

Citation: Atamer, S.; Altinsoy, M.E. Vacuum Cleaner Noise Annoyance: An Investigation of Psychoacoustic Parameters, Effect of Test Methodology, and Interaction Effect between Loudness and Sharpness. *Appl. Sci.* **2023**, *13*, 6136. https://doi.org/10.3390/app13106136

Academic Editors: Giovanni Costantini and Daniele Casali

Received: 18 April 2023
Revised: 11 May 2023
Accepted: 14 May 2023
Published: 17 May 2023

1. Introduction

Among other appliances, vacuum cleaner noise is one of the most annoying household appliance noises in our living environments. For users, vacuum cleaner noise creates an unpleasant feeling, causes fatigue and even causes anger after long usage. For passive listeners, vacuum cleaner noise makes it almost impossible to concentrate on a task or to continue verbal communication. Vacuum cleaner noise in indoor spaces can reach up to 70–80 dB(A), which makes normal speech almost inaudible. Research on vacuum cleaner noise can be divided into two main categories. The first group of studies focuses on the noise generation mechanisms of a vacuum cleaner and possible design changes for noise reduction. In contrast, the second group of studies focuses on the noise annoyance evaluation of vacuum cleaners, trying to understand the dominant correlates and developing sound quality models. At this point, these two group of studies need to be combined: it is important to understand how specific noise components affect annoyance in order to effectively reduce the total annoyance perception.

In their recent publication, Yoshido and Hatta [1] investigated the level of discomfort created by vacuum cleaner noise for active and passive listening conditions. Under

active listening conditions, participants used the vacuum cleaners themselves, whereas under passive listening conditions, participants listened to recorded noise from the same vacuum cleaners. The main point of this study was to understand the difference between robot vacuum cleaners and conventional vacuum cleaners. Under the passive listening conditions, no distracting task was given to the participants. The results showed that the levels of uncomfortableness were significantly higher under the passive listening conditions. Kumar et al. [2] focused on experimental assessments of annoyance with noise from three vacuum cleaners and the correlations between psychoacoustic parameters and annoyance evaluations. Annoyance index values for the three vacuum cleaners were calculated based on the model suggested by Altinsoy et al. [3]. They pointed out that loudness has a critical significant effect on vacuum cleaner annoyance. Altinsoy [4] investigated the main signal characteristics of vacuum cleaners, and no difference was observed between annoyance ratings from single microphone recordings and artificial head recordings. Additionally, it was found that loudness, sharpness, roughness, tonality and articulation index values can be used to model the annoyance ratings of participants. In particular, a free interview conducted with participants at the end of the listening experiments showed that most of the subjects claimed vacuum cleaner noise to be highly annoying when it disturbed communication. From this information, the articulation index values were also included in the developed model of vacuum cleaner noise annoyance.

Martin et al. [5] compared the operating noise of two vacuum cleaners (with and without silent technology) in different usage scenarios (different floor coverings and different powers) with regard to the overall user experience (UX). They found that the subjective ratings depended on both the usage scenario and the vacuum cleaner model. The vacuum cleaner without noise reduction was rated significantly worse by users than the vacuum cleaner with noise reduction in UX. Furthermore, the rating of annoyance at low power consumption was different for different floor coverings.

Companies may be reluctant to invest in better sound quality since it is difficult to quantify the profit return from a better sounding device. Takada et al. [6] investigated this issue by measuring the participants' willingness to purchase a product based on its noise. They suggested that an improvement in sound quality, especially in conditions where participants were able to listen to the product noise, increased the commercial value of the better-sounding device and increased the participants' willingness to buy the product. In another study by Takada [7], a similar approach was also applied to vehicle-door-closing sounds. In addition, two experiments on customer product selection based on acoustic performance were conducted in the same publication, relating to vacuum cleaners and hair dryers. All these experiments show that a design addressing product sound sufficiently increases the willingness to buy that product.

In another study, Ih et al. [8] focused on the annoyance estimations of vacuum cleaners and derived a prediction model for annoyance. One example of vacuum cleaner noise was recorded, and some frequency ranges were classified in terms of their importance. By increasing and decreasing the levels of these defined frequencies and using the orthogonal array technique, they designed listening tests with the aim of developing an annoyance index for vacuum cleaners. In addition to annoyance, the effects of defined frequency bands on the "performance", "loudness" and "sharpness" of the vacuum cleaners were also investigated in the listening tests. The study concluded with an artificial neural network model that was developed for the prediction of vacuum cleaner annoyance. Lyon [9] also used vacuum cleaners as an example in his work to explain the main stages of product sound quality analyses. Different components of vacuum cleaner noise were modified, and a listening test was created using a central composite design, so that a smaller number of stimuli could be used, rather than a full factorial design. The sounds of vacuum cleaner components (motor sound, suction fan noise, airflow noise and rotating brush noise) were changed, and it was found that the acceptability function of vacuum cleaner noise was dependent mainly on the airflow noise and motor noise components. It was explained

that an equal reduction of 5 dB in both noise sources was required to obtain an optimally acceptable design.

In their study, Rukat et al. [10] presented a comparison of the acoustic parameters of a vacuum cleaner on different surfaces. They performed various measurements of vacuum cleaners in different arrangements, taking into account that vacuum cleaners can be classified as devices with extensive sound sources. They found that the noise emitted by vacuum cleaners depends on the type of surface used and the arrangements of the device (canister and suction nozzle). They also concluded that it is sufficient to parameterize the acoustic performance of the device with single values, where it would be more feasible for the well-being of the end user to report the most unfavorable working conditions.

In addition to perceptual studies, other studies have focused on understanding the noise generation mechanisms of vacuum cleaners. A detailed acoustic characterization of a wet-type vacuum cleaner was conducted in the publication by Buratti et al. [11]. They explained that the total emitted noise is the sum of several contributions, such as aerodynamic noise, and mechanical and electromagnetic components. The mechanical and electromagnetic components generate rotational discrete tonal noise and the aerodynamic noise generates broadband noise.

In a series of three publications [12–14], Cudina and Prezelj explained the noise generation mechanisms of a vacuum cleaner in detail. These highly detailed publications showed the complexity of the generated noise and its components. The first publication provided an overview of the noise components that can be found in vacuum cleaner noise and how the mechanical and electromagnetic portions create tonal and broadband noise characteristics. Moreover, the consideration of the performance and noise characteristics at the same time offered insight into the inconsistency between the desired suction power and the noise level. The second publication of this series focused on the aerodynamic portion of the noise and the effect of blower geometry on different flow rates. A conclusion was made that vaned diffusers have more disadvantages than advantages and need to be omitted to reduce noise. The third and final publication of these series explained structure-borne noise. The researchers also suggested possible improvements for manufacturers to decrease structure-borne noise in vacuum cleaners.

Novakovic et al. [15] designed a new centrifugal impeller to improve the noise quality of vacuum cleaners. The aim was to increase the perceived noise quality and not only to reduce the overall noise level. The optimization process was based on two different general noise exposure models. They finalized their propeller design with triangular flow channels. In listening tests, they found that it was possible to make a user-oriented design change based on the psychoacoustic findings.

Brungart and Lauchle [16] performed sound power level measurements on a handheld vacuum cleaner to identify the main components of the noise. After analyzing the noise, they implemented modifications on the fan casing and the blade distribution, which changed the blade pass frequency. They evaluated the modifications in terms of their preference in jury testing, especially considering the magnitude of the tonal components in the overall noise. Brungart et al. [17] investigated the effect of modifications on fans and motors on an upright vacuum cleaner in another publication. They found that prominent tonal noise is created by an interaction between the electric motor cooling fan and the surrounding gussets and posts. They removed these elements in an alternating fashion such that the first blade passing frequency of the electric motor cooling fan was eliminated.

Teoh et al. [18] made modifications to a canister vacuum cleaner to reduce its noise. They pointed out that the noise of a canister vacuum cleaner consists of the blade passing noise generated by motor and the aerodynamically induced airborne noise. Two different noise reduction methods were used: the introduction of sound insulation panels made of porous expanded polypropylene and honeycomb noise filters. After these modifications, the total noise level was reduced by 7.4 dB(A), with a reduction in suction power of only 0.93%.

This study focuses on understanding the general sound characteristics of vacuum cleaners and their annoyance perception. There are many different brands and types of vacuum cleaners with different designs in the market. The differences in design result in differences in noise characteristics: some of the devices are loud, whereas some of them have higher sharpness values. Some of the devices have distinct tonal components, whereas some of the designs are free from tonality. Then, the main question is what kind of canister-type vacuum cleaners should be selected and recorded to investigate, as much as possible, the variability in noise that can be observed, so that the variability in the market can be properly represented? What is the generic vacuum cleaner noise, and how much variability can there be between different models? The goal is to select proper samples from the market such that these selected samples can represent the variability in noise.

To reach this goal, canister-type vacuum cleaners are selected such that the selected samples can represent the variability in noise. The main aim in this study is to select devices such that the selected samples can represent the variability in noise from canister vacuum cleaners.

First, the basic characteristics of vacuum cleaner noise are provided for the selected examples. Then, the ranges of calculated psychoacoustic parameters for selected vacuum cleaners are presented. Variability in the acoustic and psychoacoustic parameters on different vacuum cleaners is discussed to derive common characteristics of canister-type vacuum cleaner noise. This variability is then related to the available information on the noise generation mechanisms of vacuum cleaners in the literature. This observed variability in noise samples in the market is used to set up listening experiments and their ranges.

Afterward, two sets of listening tests are conducted in this study. The first listening test is an explanatory test to understand the main correlates of vacuum cleaner annoyance. Based on the results obtained from this test, a second set of listening tests is conducted to investigate the possible interaction effect on loudness and sharpness using a factorial design in different testing methodologies.

2. Stimuli—Signal Characteristics of Vacuum Cleaners

2.1. Selection of Vacuum Cleaners

To obtain an overview of vacuum cleaner noise, 15 vacuum cleaners were selected from the market and recorded under anechoic conditions. The selection of the vacuum cleaners was performed using the online portals of the two largest consumer electronics retail companies in Germany. Robot vacuum cleaners, upright vacuum cleaners, handheld vacuum cleaners and wet-type vacuum cleaners were not taken into account, and only canister-type vacuum cleaners were selected for this study. For canister-type vacuum cleaners, there were 155 vacuum cleaners available on both websites at the time this study was written [19,20].

For the selection of these vacuum cleaners, the maximum electrical power and sound power levels according to the manufacturers were taken into account. From the available models, the sound power levels ranged from 57 to 82 dB(A). The median value of declared sound power levels was 73.4 dB(A), where the upper and lower quartiles were 78 and 69 dB(A), respectively. The maximum electrical power of the canister-type vacuum cleaners ranged between 130 and 1700 Watts. The median value of the power was 700 Watt, and the upper and lower quartiles were 1000 and 400 Watts, respectively. The 130-Watt, 1400-Watt and 1700-Watt models were outliers. For a comparison with the given range of parameters, the parameter range of the selected 15 devices are listed in Table 1. The sound power levels of the 15 selected devices are also given in Table 2. The sound power levels of the selected devices show a good distribution over the defined market range, and no concentration on a particular sound power level was observed.

The sound power levels according to the manufacturers are provided for the highest working mode of the vacuum cleaners, as stated in [21]. The annoyance evaluations within this study were also conducted using the maximum power mode of the selected devices. However, it might be important to note that lower suction power modes, although quieter,

might emit different tonal components depending on the rotational speed of the motor, which might change the annoyance evaluations. This effect was not taken into account in this study and might be a topic of further investigation. Especially for lower broadband levels, the effect of the tonal components on annoyance might be more dominant.

Table 1. Market ranges of vacuum cleaners and the corresponding ranges selected for this study.

	Available in the Market		**Selected**	
	Minimum	**Maximum**	**Minimum**	**Maximum**
Sound Power Level (dB(A))	57	82	59	82
Maximum Power (Watt)	130	1700	600	1400

Table 2. Sound power levels for the selected vacuum cleaners according to their manufacturers (in dB(A)).

Device Number	**Sound Power Level (dB(A))**	**Device Number**	**Sound Power Level (dB(A))**	**Device Number**	**Sound Power Level (dB(A))**
1	71	6	79	11	63
2	71	7	82	12	79
3	70	8	66	13	59
4	61	9	74	14	72
5	80	10	77	15	68

2.2. Recordings

The selected vacuum cleaners were recorded in a fully anechoic environment. There are two main working conditions of a vacuum cleaner: the first one is on hard flooring and the second one is on carpet. Since carpet might affect noise emission, both conditions were taken into account in this study. Vacuum cleaners were positioned directly on a reflective, heavy surface or on a carpet placed on top of this surface. Single microphone recordings were obtained by placing the microphone directly in front of the vacuum cleaner at a distance of 0.75 m and a height of 1.5 m. For the recordings, the vacuum cleaners were positioned on top of the reflective plane, as stated in the standard, for determination of the airborne acoustical noise of vacuum cleaners [21] (Figure 1).

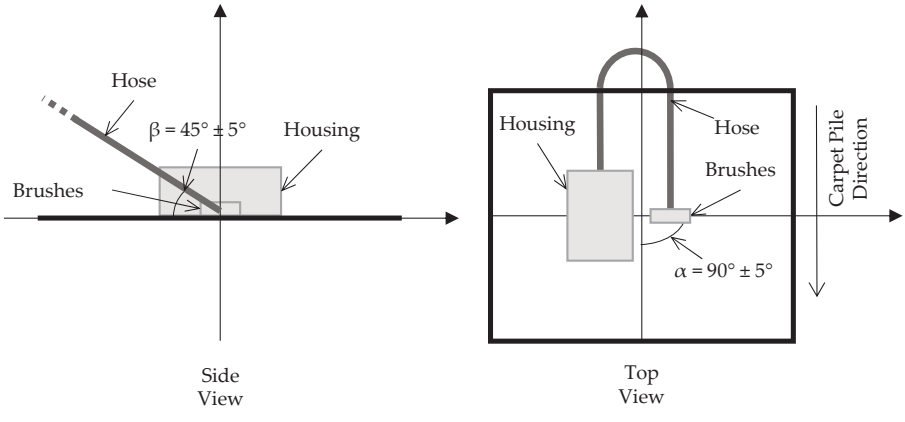


Figure 1. Positioning of the vacuum cleaner, as described in [21].

2.3. Signal Characteristics

Vacuum cleaner sounds are usually characterized by band noise with tonal components [12]. Figures 2 and 3 show the FFTs and spectrograms of all recorded vacuum cleaners on hard flooring, respectively. Figure 2 was plotted with 1/24 octave smoothing so that

the differences between different vacuum cleaner recordings are easier to observe. For hard flooring, considering the threshold of hearing, the overall frequency range of vacuum cleaner noise is from 70 Hz up to 10 kHz. In most cases, there is a tonal component at 100 Hz, with different intensities for different brands and types. Usually, at approximately 500 Hz and 5000 Hz, vacuum cleaner noise reaches its maximum A-weighted level. On top of the 100 Hz tone, there are usually other tonal components observed with respect to vacuum cleaner sounds. At approximately 500–750 Hz, a single tone component exists for some vacuum cleaners, and some other tonal components are present in the 3000–5000 Hz range. Additionally, for some models, it is possible to observe tones at approximately 10 kHz. Finally, the variation in the levels between 500 Hz and 10 kHz can be up to 15 dB.

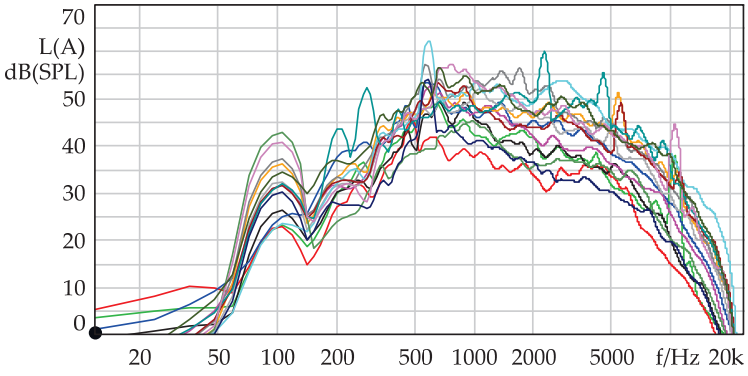


Figure 2. FFTs of all 15 vacuum cleaner noise for hard flooring case (1/24 octave intensity averaging smoothing, A-weighted; spectrum size: 4096).

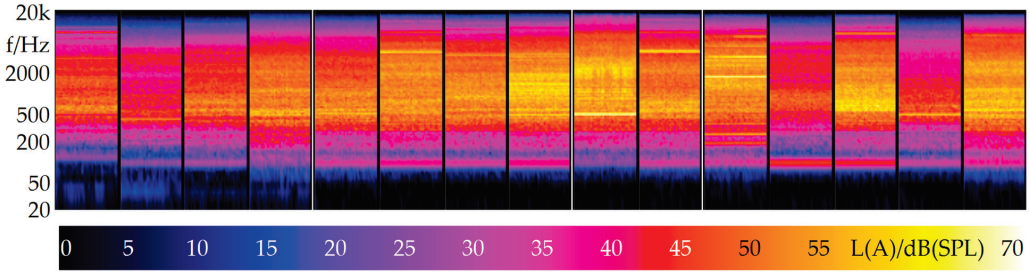


Figure 3. Spectrograms of 15 selected vacuum cleaner noise recordings on hard flooring (A-weighted, spectrum size: 4096).

Vacuum cleaner sound is stationary. It can be seen in Figure 3 that all of the example vacuum cleaner sounds show no significant fluctuation over time. Tonal components (for example, approximately 500 Hz for the ninth vacuum cleaner) stay constant as the device keeps running.

The range of acoustic parameters for the 15 selected vacuum cleaner sounds for the hard flooring and carpet cases are given in Table 3. The loudness values were calculated according to the ISO 532-1 standard [22]. The DIN 45631 [23] and ISO 532-2 [24] standards were omitted in this paper since, for broadband noises such as vacuum cleaners, the three standards delivered similar values. The sharpness values were calculated according to the publications of Aures and Bismarck, as well as the German Standard DIN 45692 [25–27]. It is important to note that the results from these different models differ in one important aspect: the Bismarck and DIN 45692 models do not take into account the influence of intensity of the signal on sharpness perception; hence, these models are usually used for sounds with similar loudness values. The Aures model, on the other

hand, takes into account the influence of loudness of the signal on sharpness perception. However, one of the focuses of this paper is to investigate the possible interaction between loudness and sharpness. Hence, these three different sharpness models were taken into account since they have different methods of including the effect of loudness on sharpness perception. Eventually, three different sharpness values were calculated: the first one is Aures sharpness, with the loudness values calculated according to ISO 532-1; the second one is the Bismarck sharpness, with the loudness values calculated according to ISO 532-1; and the last one is DIN 45692 sharpness, with the loudness values calculated according to DIN 45631. Finally, single value tonality values were calculated based on the publications of Aures and Terhard [25,28] and on the hearing model of Sottek [29]. Both models have a psychoacoustic basis, but there are also clear differences. The Aures model starts from Zwicker loudness and extracts the tonal components from a FFT spectrum. The degree of tonality is calculated based on the ratio of tonal to non-tonal loudness as a function of time. Spectral information is not included in this model [30]. On the other hand, the Sottek model includes a hearing model approach in which the signal is first filtered through the outer and middle ear filtering and the partial loudness of tonal to non-tonal content in critical bands is calculated to determine the tonal loudness. In addition, recent studies have found [31–38] that the perception of tonal content is frequency-dependent, so the final decision on the strength of the tonal content takes into account the frequency of the tone. Additionally, the distribution of the aforementioned psychoacoustical parameters over 15 vacuum cleaners can be found in Figures 4–6. From the calculated values, it can be observed that the variations over loudness and sharpness show a fine distribution over the defined range. The tonality values also show a degree of distribution for tuHMS values between 0 and 1.3, with one outlier with strong tonality.

Table 3. Calculated minimum and maximum acoustic and psychoacoustic measures among different vacuum cleaners for the hard flooring and carpet cases.

Case	Parameter	Min	Max	Unit
Hard Flooring	Level	64.3	76.7	dB
	A-weighted level	61.4	76.4	dB(A)
	Loudness (ISO 532)	15.5	38.5	sone
	Sharpness (DIN 45631 and ISO 532 + Aures)	2.82	4.5	acum
	Sharpness (DIN 45692 − DIN 45631)	1.76	2.15	acum
	Sharpness (DIN 45631 and ISO 532 + Bismarck)	1.62	1.95	acum
	Tonality (Aures)	0.0505	0.2470	tu
	Tonality (Hearing Model)	0.06	2.51	tuHMS
Carpet	Level	59.7	76.1	dB
	A-weighted level	52.1	75.4	dB(A)
	Loudness (ISO 532)	8.6	36.4	sone
	Sharpness (DIN 45631 and ISO 532 + Aures)	2.47	4.43	acum
	Sharpness (DIN 45692 − DIN 45631)	1.63	2.15	acum
	Sharpness (DIN 45631 and ISO 532 + Bismarck)	1.5	1.97	acum
	Tonality (Aures)	0.0598	0.3230	tu
	Tonality (Hearing Model)	0.05	1.95	tuHMS

The frequency content of the emitted noise strongly depends on the positioning of the vacuum cleaner (hard flooring or carpet). Since it absorbs some of the emitted energy in the mid- to high-frequency range, the carpet condition changes the noise. To illustrate this effect, Figure 7 shows the spectra of all recorded vacuum cleaners with and without a carpet. The left panel shows the spectra for the recordings without a carpet, and the right panel shows those with a carpet. The figure colors were intentionally kept in greyscale so that the differences between the hard flooring and carpet cases could be easily compared. In this figure, it can be seen that the levels can be lowered by up to 10 dB in the regions above 500 Hz. This shows that the variation in the recordings and stimuli pool can be

even broader if the carpet case and the hard flooring case are used simultaneously for listening experiments.

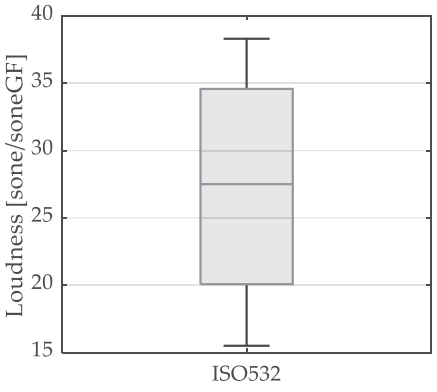


Figure 4. Calculated loudness values for all 15 vacuum cleaners based on ISO 532-1 (recordings on hard flooring).

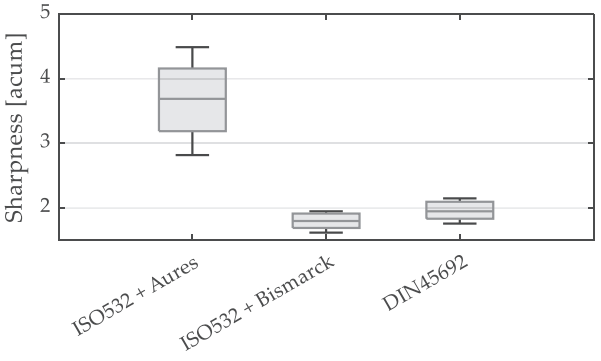


Figure 5. Calculated sharpness values for all 15 vacuum cleaners based on three different sharpness models (recordings on hard flooring).

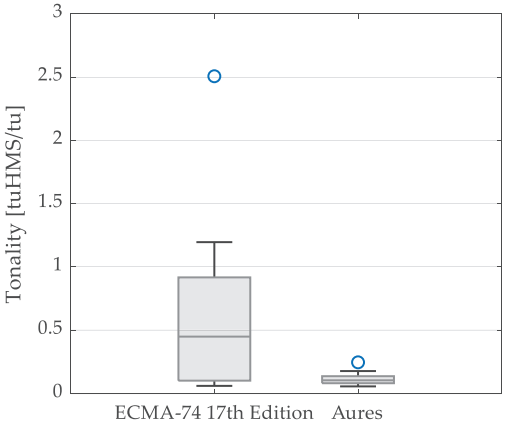


Figure 6. Calculated tonality values for all 15 vacuum cleaners based on two different tonality models (recordings on hard flooring).

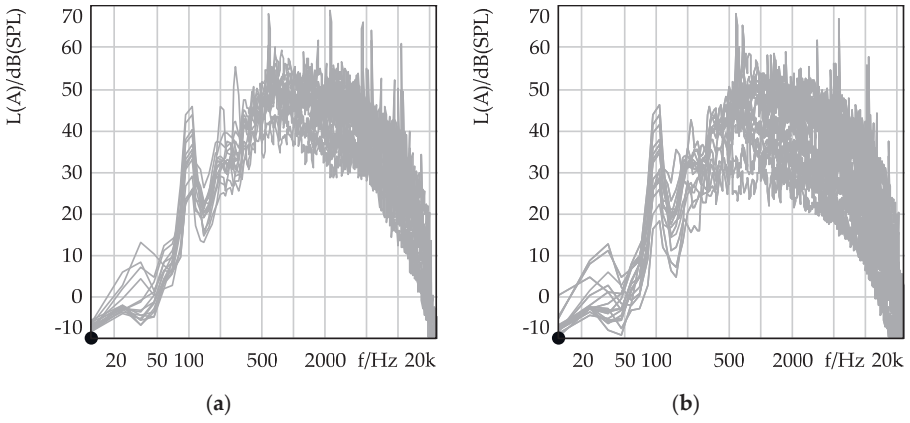


Figure 7. Frequency content of every vacuum cleaner recorded: (**a**) hard flooring; (**b**) carpet (A-weighted, spectrum size: 4096).

Considering the aforementioned observations, typical vacuum cleaner noise can be described and visualized as in Figure 8. The A-weighted level increases up to 500 Hz; then, a slight decrease is observed up to the 5 kHz range. From the 5 kHz range, a steep decrease in the overall A-weighted level can be observed. There are different ranges of these broadband noise characteristics for different vacuum cleaners, as shown by the dotted lines in Figure 8. In addition to these frequencies, a 100 Hz tone is observed for almost all vacuum cleaners. The intensity of this tonal component also varies between different models. Finally, above 500 Hz, tonal components can be observed in various vacuum cleaners, and their frequency, number and intensity change between different brands/models.

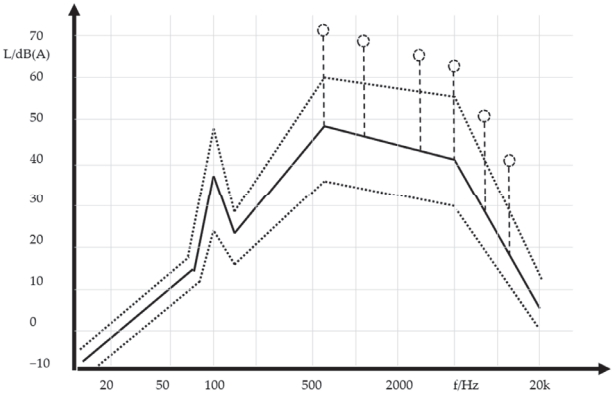


Figure 8. Average vacuum cleaner noise and possible variations. The dotted lines show the variation between different brands and models. Possible tones were also represented (for spectrum size: 4096).

Furthermore, in addition to the single value tonality calculations based on the Aures model, tonal components were calculated based on DIN 45681 [39] and the hearing model of Sottek [29]. The calculated tones where the penalty values are equal to or greater than 2 dB, for both the hard flooring and carpet cases, are given in Figures 9 and 10. It can be seen that the penalty values calculated for 100 Hz tones are higher for DIN 45681; however, since the hearing model tonality includes the frequency-dependent perceptual characteristics of tonal components, the calculated tonality values for higher frequencies dominate in Figure 10. For both figures, tonal content was divided into three main regions

shown in different colors: blue color represent the tonality around 100 Hz (Tonality LOW), orange color represent the tonality around 200–800 (Tonality MID) and lastly, yellow color represent the tonality around 1000–10,000 Hz (Tonality HIGH).

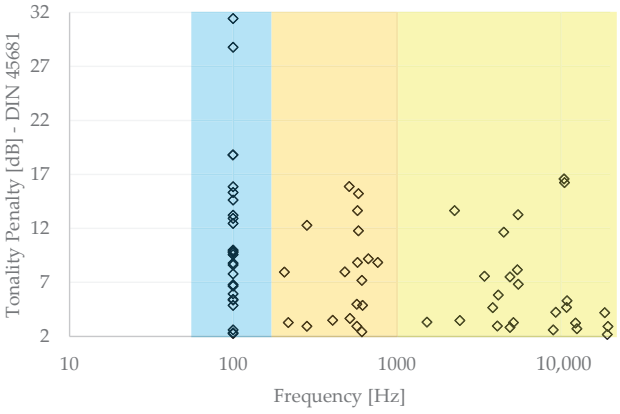


Figure 9. Calculated tonal components of which the penalty value is more than 2 dB according to DIN 45681, for both the hard flooring and carpet cases. These components are grouped into LOW–MID–HIGH regions.

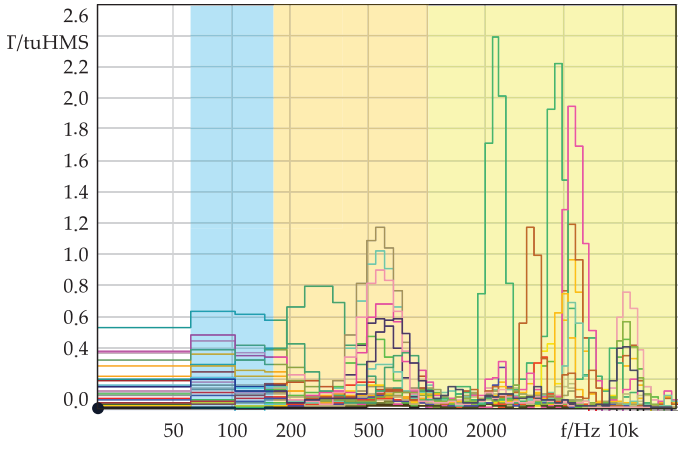


Figure 10. Calculated specific tonality based on hearing model of Sottek, for both the hard flooring and carpet cases. Calculated values are grouped into LOW–MID–HIGH regions.

3. Listening Test 1

The first listening test was conducted to understand the main correlates of annoyance due to vacuum cleaner noise. Participants were asked to rate their perceived annoyance on a rating scale with verbal anchors in the form of a slider. Twenty-one people participated in the listening test, which was conducted in a soundproof audiometric booth. The listening test was performed with both original and synthesized recordings. These additional vacuum cleaner samples were created to increase the variation in the data. The correlation between psychoacoustical parameters and the annoyance estimations was calculated at the end of the listening test.

3.1. Stimuli, Subjects and Test Method

For the first listening test, in addition to the original recordings, new synthesized recordings were obtained by parameterizing the main signal characteristics to increase the

variability in the data. Finally, 92 stationary 5 s stimuli were obtained from 15 different vacuum cleaners. The following methods were used to modify the original signals and to obtain new stimuli for the listening tests:

- Original stimulus (hard flooring and carpet conditions);
- Recording of only the housing of the vacuum cleaners (without brushes or the suction hose);
- Increasing/decreasing the overall level;
- Filtering out the dominant tonal components (depending on the frequency of the existing tonal component);
- Low-pass filtering at 2 kHz and 4 kHz to increase the variability in the bass–treble ratio.

These new stimuli were created to increase the possible variation and coverage. Although the vacuum cleaner selection was carried out with justification (using maximum electrical power and sound power levels as descriptors), it is still a sample and might not fully represent every vacuum cleaner in the market. With these additional stimuli, we aimed to cover any other vacuum cleaner in the market not directly included in the first selection and any possible future developments that might be introduced in vacuum cleaner design with technological developments.

An example signal manipulation is shown in Figure 11: The original signal was found to have tonal components at 200 Hz, 300 Hz and 500 Hz. In the second step, the 200 and 300 Hz components were taken out, and in another step, the 500 Hz component was taken out. Afterwards, the overall signal levels were changed by +6 dB, −6 dB and −12 dB. Other signals were also manipulated in this way to obtain more variation in the data. The calculated acoustical and psychoacoustical parameters and the standard deviations after the signal manipulations are shown in Table 4.

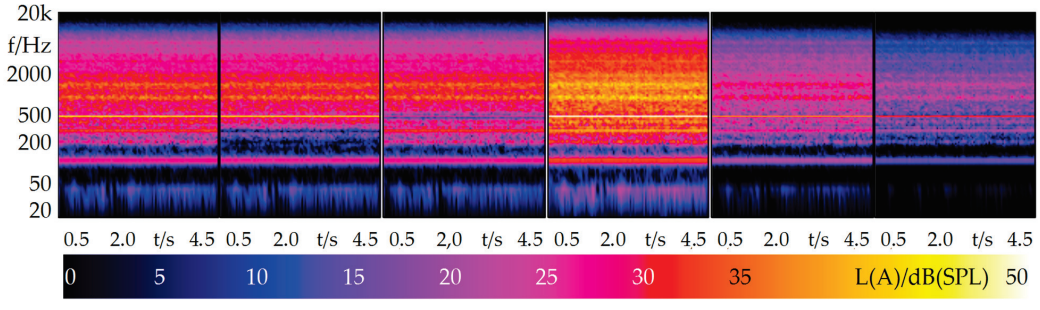


Figure 11. Synthesizing new stimuli: one example case, from left to right: original case, bandstop 200 Hz and 300 Hz, bandstop 500 Hz, 6 dB increase, 6 dB decrease, and 12 dB decrease (spectrum size: 4096).

Twenty-one subjects participated in the test, which was conducted in a soundproof audiometric booth. Eight participants were female, and thirteen participants were male. The age of the participants ranged from 23 to 63 years, with a mean of 37.7 and a standard deviation of 12.7. None of the participants reported having a known hearing problem, rather than age-related hearing loss. The overall variability in the loudness, sharpness and tonality values were assessed before the test, and 20 stimuli were selected for training. The training stimuli were selected to contain a representative range of sound levels and loudness, sharpness and tonality values. All subjects voluntarily participated in the experiment. At the beginning of the test, participants were informed about the contents of the test (vacuum cleaner noise assessments) and the test procedure. The training session and the test session were described. Participants were told that they had to familiarize themselves with the training session information for the real test. The graphical user interface was explained to the participants together in the experiment room. The first signal playback was conducted together with the participant to ensure that the sound reproduction system was working

properly and that the participants were comfortable with the signals and the headphones. For the listening experiment, a slider scale was used, where the participants were asked to evaluate the annoyance of the sounds ("How do you evaluate the annoyance?") on a quasi-continuous scale (from 0 to 100 with a step size of 1) with equidistant neighboring categories (not at all, slightly, moderately, very or extremely) (Figure 12). Stimuli were played in a randomized order for each participant.

Table 4. Calculated minimum and maximum acoustic and psychoacoustic measures, as well as standard deviations after modification of the stimuli.

Parameter	Min	Max	STD	Unit
Level	44.1	76.7	6.1	dB
A-weighted level	40.1	76.4	7.7	dB(A)
Loudness (ISO 532)	3.3	38.5	8.5	sone
Sharpness (ISO 532 + Aures)	1.54	4.49	0.65	acum
Sharpness (DIN 45692 − DIN 45631)	1.16	3.27	0.28	acum
Sharpness (ISO 532 + Bismarck)	1.12	2.93	0.24	acum
Tonality (Aures)	0.0433	0.3230	0.0536	tu
Tonality (Hearing Model)	0.04	2.51	0.42	tuHMS

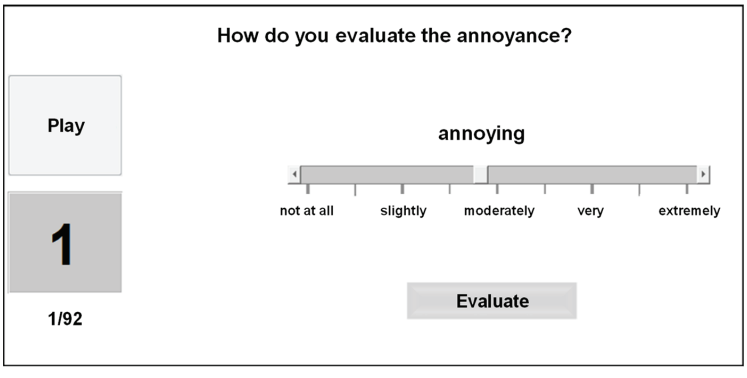


Figure 12. Graphical user interface used for slider scale experiment.

3.2. Results

The distributions of the mean annoyance evaluations showed that the participants used most of the available surface range for the evaluations. For the first listening test, the minimum and maximum of the mean annoyance evaluations were 5.6 and 96.7, respectively, and the mean average annoyance estimations and the median average annoyance estimations were 49.3 and 52.1, respectively. The quartiles of this distribution were 69.2 and 37.8.

The correlations between the calculated parameters and annoyance estimations can be found in Table 5. The sample size in these calculations was large, with 92 stimuli. Since the significance test for the correlation also depends on the sample size, it was possible to obtain significant or highly significant correlation values, even though the calculated correlation coefficient was only 0.25. At this point, it is important to focus on the interpretation of strong or weak correlations in correlation coefficients. Weak but significant correlations are not meaningful from a psychoacoustic point of view, as this effect is rather small but could be demonstrated due to the large sample.

The first explanatory investigations show that overall levels (dB(A) values, as well as loudness) play a crucial role in annoyance estimations (correlation 0.966 for dB(A) and 0.963 for loudness), which is consistent with many other publications on sound quality, as well as the cited publications for vacuum cleaners. The effect of sharpness was also found to be large with high significance. It is important to note that the correlations

change significantly depending on the applied sharpness calculation method. Sharpness calculations based on the method of Aures show higher correlations (0.763 with ISO 532-1 loudness calculations) than the model of Bismarck (0.261 with ISO 532-1 loudness calculations) and DIN 45692 (0.261 with DIN 45631 loudness calculations). This result is rather expected, since the Aures sharpness model includes the effect of loudness variations in comparison with the Bismarck model and the DIN 45692 standard. This effect can be described as a possible multicollinearity between these two parameters.

Table 5. Correlations between annoyance estimations and psychoacoustic parameters.

	Annoyance	dB (A)	Loudness (ISO)	Sharpness (ISO + Aures)	Sharpness (ISO + Bismarck)	Sharpness (DIN)	Tonality (Aures)	Tonality (HMS)
Annoyance	1	0.966 **	0.963 **	0.763 **	0.257 *	0.261 *	−0.140	0.439 **
dB(A)		1	0.962 **	0.689 **	0.148	0.149	−0.187	0.408 **
Loudness (ISO)			1	0.778 **	0.244 *	0.245 *	−0.056	0.493 **
Sharpness (ISO + Aures)				1	0.794 **	0.793 **	−0.86	0.439 **
Sharpness (ISO + Bismarck)					1	0.999 *	−0.105	0.157
Sharpness (DIN)						1	−0.096	0.157
Tonality (Aures)							1	0.512 **
Tonality (HMS)								1

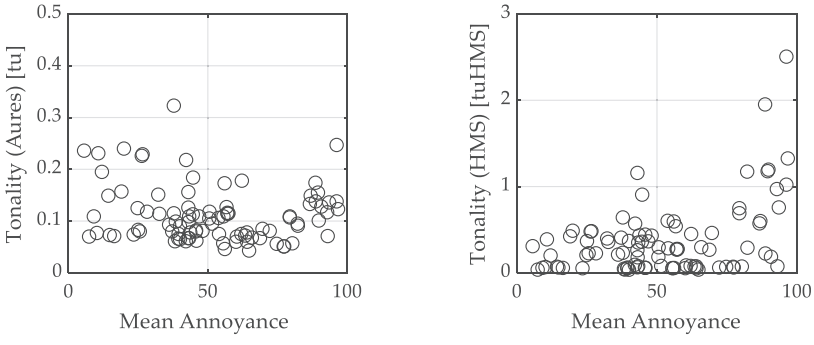
* Correlation is significant at the 0.05 level (2-tailed) ** Correlation is significant at the 0.01 level (2-tailed).

Finally, single value tonality calculations based on the model of Aures model were not correlated with mean annoyance evaluations (−0.140), whereas the calculations based on the hearing model have significant moderate correlations with mean annoyance evaluations (0.439). On the other hand, there is also an almost equally high correlation between the loudness of the stimuli and the single value tonality calculations based on the hearing model (0.493). The effects of overall loudness of the signals on tonality perception, as well as the frequency-dependent characteristics of tonality perception, are already included within the hearing model of tonality [37]. The expert panel of listeners usually complained about the noise samples after the experiment when there was a dominant salient tonal component. Almost every participant mentioned this tonality problem, although the calculated correlation was moderate. This phenomenon can be better explained in Figure 13. Correlation looks for a linear relationship between the input and output parameters; however, the tonality calculations show a usual stepwise behavior. Some of the stimuli have no or almost negligible tonality, while some of the stimuli have higher tonal components. In particular, the right panel of Figure 13 shows that, whenever a stimulus has a strong tonality based on the hearing model, the mean annoyance evaluations for this stimulus was usually 80 or above.

Since it is known from the recent literature that the effect of single tones in annoyance depends on the frequency of the tone (see, for example, References [34,36,37]), another method was used to calculate the correlations between the tonality and annoyance estimations. The tonal components in the vacuum cleaner noise are mainly clustered in three distinctive regions. These regions can be seen in Figures 9 and 10. The first region includes the tones at approximately 100 Hz, the second region includes the tones between 200 Hz and 1000 Hz, and the last region includes the tones above 1000 Hz. These three regions were defined as Tonality LOW, Tonality MID and Tonality HIGH. Tonality values were calculated for these defined frequency ranges based on DIN 45681 [39] and the hearing model of Sottek [29], and if more than one tonal component was present in these ranges, the maximum tonal penalty value was calculated.

The correlation values were calculated for the newly defined single values and were presented in Table 6. For readability reasons, redundant or repetitive parameters were removed from Table 5. Here, it can be seen that, based on hearing model tonality, the low-frequency region (100 Hz) and the high-frequency region (over 1000 Hz) show moderate

correlation between mean annoyance evaluations. Similar observations cannot be made for the DIN 45681 tonality calculations for the low, mid and high regions. Finally, similar investigations can also be found in Figure 14, where the stimuli with higher tonality values in the low and high ranges (hearing model) tend to have higher mean annoyance ratings (lower left and lower right panel of this figure).



(**a**) (**b**)

Figure 13. Mean annoyance evaluations vs. single value tonality calculations, based on (**a**) Aures model; (**b**) hearing model.

Table 6. Correlations between annoyance estimations and selected psychoacoustic parameters with defined tonality regions.

Pearson Correlation—Listening Test 1 (Tonality)									
	Annoyance	**Loudness (ISO)**	**Sharpness (ISO + Aures)**	**Tonality LOW (DIN)**	**Tonality MID (DIN)**	**Tonality HIGH (DIN)**	**Tonality LOW (HMS)**	**Tonality MID (HMS)**	**Tonality HIGH (HMS)**
Annoyance	1	0.963 **	0.763 **	−0.200	−0.260 *	0.249 *	0.354 **	0.225 *	0.454 **
Loudness (ISO)		1	0.778 **	−0.133	−0.155	0.247 *	0.386 **	0.324 **	0.469 **
Sharpness (ISO + Aures)			1	−0.091	0.031	0.046	0.272 **	0.247 *	0.439 **
Tonality LOW (DIN)				1	0.349 **	−0.019	0.692 **	0.321 **	0.018
Tonality MID (DIN)					1	−0.238	0.184	0.696	0.034
Tonality HIGH (DIN)						1	0.342 **	0.047	0.572 **
Tonality LOW (HMS)							1	0.543 **	0.539 **
Tonality MID (HMS)								1	0.322 **
Tonality HIGH (HMS)									1

* Correlation is significant at the 0.05 level (2-tailed) ** Correlation is significant at the 0.01 level (2-tailed).

Ultimately, it was found that the annoyance estimations of vacuum cleaners depend mainly on the overall loudness of the signal, the degree of higher frequencies (hence sharpness), and the possible tonal components at lower and higher frequencies, mainly above 1 kHz based on the hearing model tonality. The same conclusion cannot be drawn from the use of DIN 45681. The effect of higher frequencies seems to be stronger than that of the low-frequency 100 Hz tone, which is also partly consistent with the results in [8], where the participants responded to the change in level up to 600 Hz with an increase in performance and loudness perception, while this change had no effect on annoyance perception. However, in this listening experiment, low-frequency tonality was moderately correlated with annoyance, although it was only valid for one tonality model.

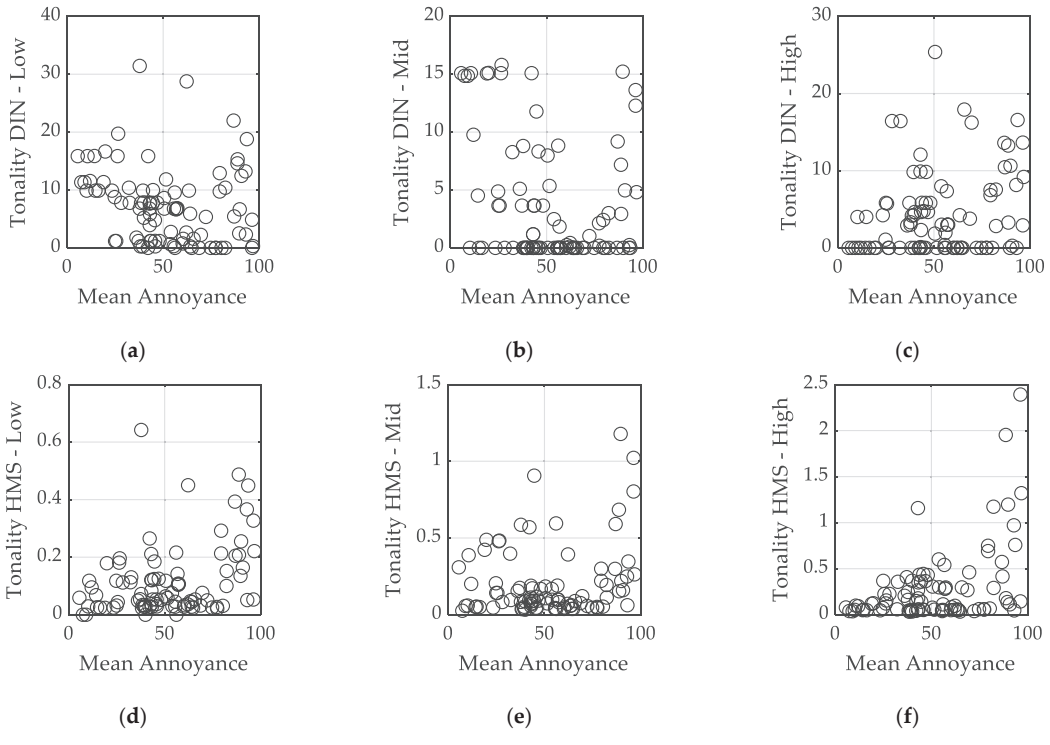


Figure 14. Mean annoyance evaluations vs. single value tonality calculations, based on (**a**) DIN 45681—Low; (**b**) DIN 45681—Mid; (**c**) DIN 45681—High; (**d**) Hearing Model—Low; (**e**) Hearing Model—Mid; (**f**) Hearing Model—High.

4. Listening Test 2 (Comparison of Different Test Methods and Loudness vs. Sharpness Factorial Design)

The second listening test was performed to investigate the possible interaction between loudness and sharpness. Stimuli were generated in the form of a factorial design, with selected loudness and sharpness values. One sample of vacuum cleaner noise without tonal components was selected, and its loudness and sharpness values were systematically changed by filtering. Four different sub-tests were conducted to investigate this possible interaction. In addition, three different experimental methods were used in these subtests to investigate the possible bias due to the experimental method. Nine participants were asked to rate annoyance of the vacuum cleaner noise signals in these three experiment methods. Finally, the results from the different test methodologies were compared and a repeated measures ANOVA was conducted to investigate the possible loudness sharpness interaction.

4.1. Stimuli, Subjects and Test Methods

Listening test 1 showed that loudness and sharpness have a significant effect on annoyance perception. However, it was not clear from this experiment if there was an interaction effect between these two parameters, i.e., higher-frequency content might influence the annoyance estimations as a function of the overall loudness of the stimulus. Multicollinearity is an important problem in statistical modeling that could lead to redundant input parameters in developed quality models. Moreover, the mathematical definitions of loudness and sharpness have a strong correlation from a purely acoustical point of view [40]. These two facts are particularly critical in sound quality evaluations of vacuum cleaners, where the loudness and sharpness play important roles. First, it can be interpreted from verbal

descriptions of the participants that, when the stimuli are louder, the stimuli with stronger high-frequency content are perceived as more annoying. However, we know from the definition that an increase in higher frequencies increases loudness as well as sharpness but at a different rate of change. For this reason, it is necessary to investigate whether there is an interaction effect between these two parameters.

To analyze this possible interaction effect, a series of listening tests was designed, in each of which the loudness and sharpness values were varied in the context of a factorial design. Listening test 2 was divided into four parts. Part 1 included a slider scale experiment with a 3 × 3 factorial experiment design for loudness and sharpness. Part 2 included a magnitude estimation test with a similar 3 × 3 factorial design. Part 3 of the listening test had the same 3 × 3 factorial design, but this time, a random access method was used. In Part 4, the factorial design was changed to 5 × 3 for loudness and sharpness, and the random access method was used.

Moreover, in addition to the possible loudness–sharpness interaction effect, this section also compares the different test methods to discuss the advantages and shortcomings in factorial design experiments. Mainly, for Parts 1, 2 and 3, the slider scale, magnitude estimation and random access methods were applied for the same stimuli under the same reproduction conditions. Finally, in Part 4, the variability in the loudness was extended in both directions so that the possible interaction effects can also be observable in the quieter and louder stimuli.

For Parts 1, 2 and 3, a 3 × 3 factorial design was used for the loudness and sharpness values. For Part 4, a 5 × 3 factorial experimental design was used for loudness and sharpness values. The only difference in Part 4 was that the maximum and minimum values of the loudness values were extended. The values used for each factorial design can be found in Table 7. Here, stimuli 4–12 were used for the 3 × 3 design (numbers with an asterisk), and stimuli 1, 2, 3, 13, 14 and 15 were added for the 5 × 3 design. The loudness values, calculated according to the standard ISO 532-1 [22], were selected to be approximately 16 sone, 20 sone and 25 sone for the 3 × 3 design. These values were selected such that they are in the limits measured for each vacuum cleaner given in Table 3. For the 5 × 3 factorial design, the loudness values were extended to 13, 16, 20, 25 and 30 sone. Meanwhile, the sharpness values, calculated according to the calculation method of Aures ([25] with [22]), were selected as 2.4, 2.9 and 3.3 acum.

To obtain vacuum cleaner noise with different sharpness values, a parametric IIR low-pass filter was applied to a selected vacuum cleaner recording. The cutoff frequency of the low-pass filter was set to 4000 Hz. Around this particular frequency, vacuum cleaner noise decreases, and this decrease is different for different vacuum cleaners. Three different parametric low-pass filters with three different Q values were used, so the slope of each line in the FFT was different. Therefore, it was possible to obtain vacuum cleaner noise with different high-frequency components and thus different sharpness values. Since changing the high-frequency content affects the overall loudness of the sound, the overall level is slightly shifted for each filter case. As a result, the same loudness values are obtained. One example is shown in Figure 15. Here, three stimuli have the same loudness but different sharpness values.

To generate the stimuli in this listening experiment, one original stimulus was taken as the basis. This original sound was selected such that the signal had no tonal components, a loudness of 20 sone (ISO 532-1) and a sharpness of 3.13 acum (Aures). Both loudness and sharpness values lie in the middle of the observed loudness and sharpness ranges. Intentionally, a stimulus without a tonal component was selected to eliminate any possible bias originating from the tonal component in this listening experiment.

Three different test methodologies were compared for factorial design experiments 1, 2 and 3 (Figure 16). For these three experiments, the slider scale, magnitude estimation and random access methods were used. The slider scale experiment (Figure 16, part a) used a quasi-continuous rating slider with verbal anchors (from 0 to 100 with a step size of 1) with equidistant neighboring categories (not at all, slightly, moderately, very or extremely), as in

listening test 1. Participants used this slider to rate the annoyance of the given stimuli. The appearance of the stimuli was randomized for each participant, and participants were not allowed to navigate back and to change their evaluations for previous stimuli.

Table 7. The 5 × 3 and 3 × 3 factorial designs (loudness vs. sharpness).

The 5 × 3 and 3 × 3 Factorial Designs on Loudness and Sharpness			
Stimulus #	Experiment Design	Loudness (sone)	Sharpness (acum)
1	5 × 3	13.1	2.4
2	5 × 3	13.3	2.97
3	5 × 3	13.4	3.47
4	3 × 3 & 5 × 3	16.4	2.42
5	3 × 3 & 5 × 3	16.2	2.9
6	3 × 3 & 5 × 3	16	3.34
7	3 × 3 & 5 × 3	20.8	2.48
8	3 × 3 & 5 × 3	20.2	2.87
9	3 × 3 & 5 × 3	20	3.34
10	3 × 3 & 5 × 3	25	2.57
11	3 × 3 & 5 × 3	25	2.93
12	3 × 3 & 5 × 3	25	3.34
13	5 × 3	30	2.65
14	5 × 3	30	2.91
15	5 × 3	30	3.29

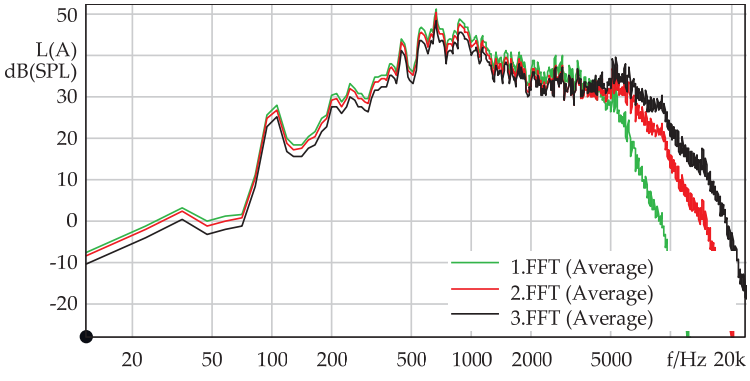


Figure 15. Frequency content of example stimuli 1, 2 and 3, which have the same loudness but different sharpness values (spec size: 4096).

In the magnitude estimation experiments (Figure 16, part b), an anchor stimulus and a defined annoyance value for that particular anchor stimulus were used. Participants were then asked to rate the annoyance of a particular stimulus relative to the anchor stimulus. The reference value for annoyance was set to 100 for the anchor stimulus. Participants could listen to the two given sounds as many times as necessary and they gave their ratings by entering a number in the free space below the play button. The order of the stimuli was also random, as in the slider scale experiment. This random order was different for each participant, and participants could not go back and change their ratings.

Lastly, the random access method (Figure 16, part c) used a user interface where all of the stimuli were presented to the participant simultaneously. At any time, the participant could click the play buttons in any order to listen the stimuli, could compare them in pairs and could change their previously established response. They could drag and drop the playback icons to the field, which contained the same verbal anchors as in slider scale experiment. The position of the playback icon (i.e., stimulus) on the y-axis was taken as the rating of a participant.

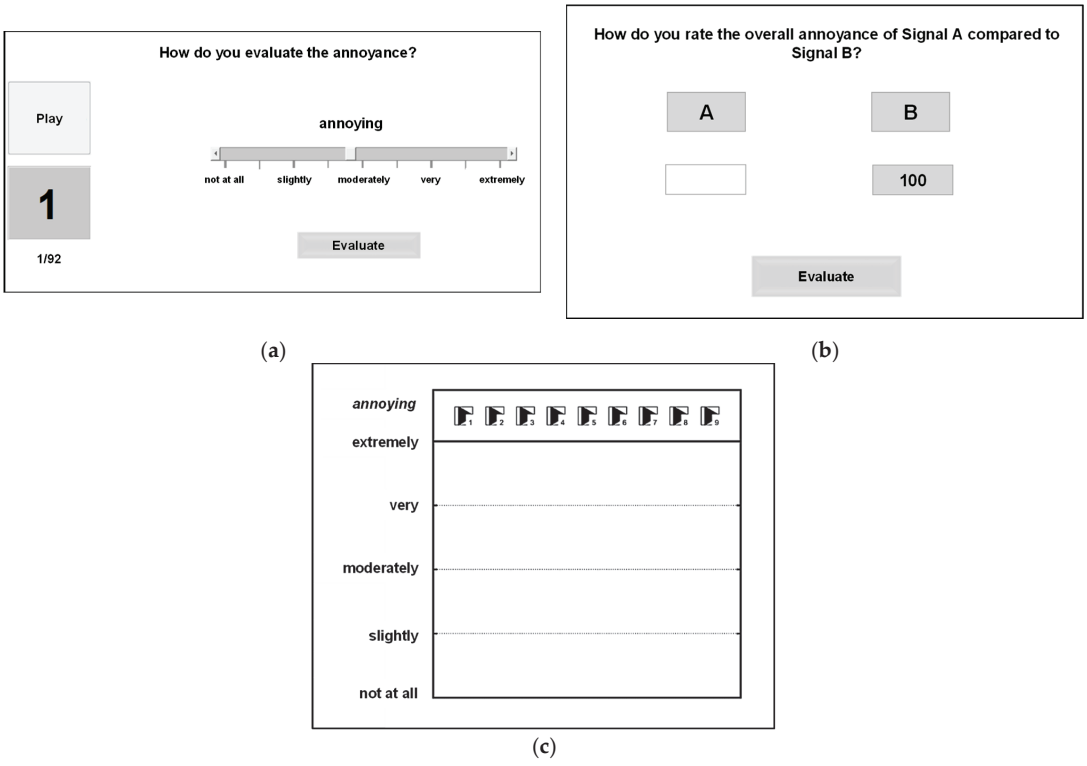


Figure 16. GUIs of the three different test methodologies used in Parts 1, 2 and 3: (**a**) slider scale; (**b**) magnitude estimation; (**c**) random access method.

The main difference between the random access method and the slider scale method is that, in the random access method, participants can always replay all stimuli, can change their decisions and have a better sense of control over their evaluations. However, the number of stimuli in such experiments is rather limited. Firstly, the user interface does not have enough space for an unlimited number of playback icons, and secondly, participants reported that, as the number of stimuli increased, it became more difficult to make a decision. When the "evaluation field" was filled (when a participant moved all the playback icons to their correct locations), participants clicked "evaluate" to submit the results.

Similar to listening test 1, the question was "How do you evaluate the annoyance?", and participants were given the categories "not at all", "slightly", "moderately", "very" or "extremely". For the magnitude estimation test, the question was changed to "How do you evaluate the annoyance of signal A, compared to the signal B?" For the magnitude estimation procedure, stimulus 1 (lowest loudness and sharpness values) was used as the anchor stimulus.

Part 1 used a slider scale evaluation, Part 2 used a magnitude estimation, and Parts 3 and 4 used the random access method. Nine subjects participated in all parts of the experiment. The experiments were conducted in a soundproof audiometric booth. Three participants were female, and six participants were male. The age of the participants ranged from 25 to 38 years, with a mean of 31.6 and a standard deviation of 3.9. None of the participants reported having a known hearing problem. In each of these experiments, participants were given instructions similar to those in listening test 1.

4.2. Results

The results of Parts 1, 2 and 3 can be seen together in Figure 17. For Part 1 and Part 3, the average annoyance evaluations and standard deviations were calculated from the individual ratings of participants. The results of Part 2 are shown on a second axis in the same graph. The main reason for this visualization was because the evaluations from a magnitude estimation are ratio-scaled quantities. The evaluations of each participant were linearized by taking the log10 of each value. Since the first stimulus was an anchor stimulus with a reference value of 100, participants evaluated this signal as 100, which was shown in a linearized way as the number "2" in this figure. Similarly, if a participant rated a stimulus as "200" (two times more annoying than the anchor stimulus), this was represented approximately as 2.30 on this graph.

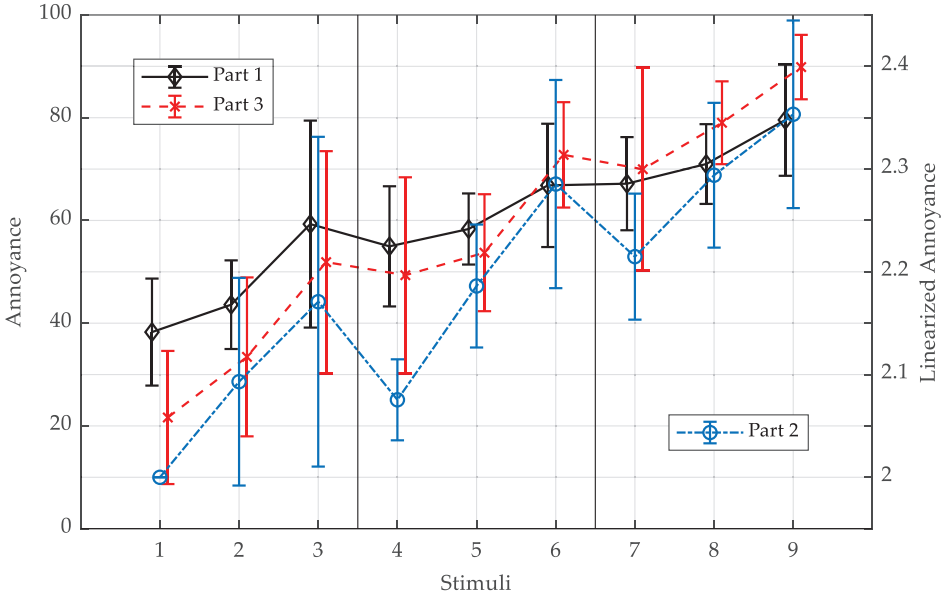


Figure 17. Results of Parts 1, 2 and 3 with the same stimuli and three different testing methods (slider scale, magnitude estimation and random access method, respectively). Results of Part 1 and Part 3 were gathered by calculating the arithmetic mean values of the results, as well as standard deviations. For Part 2, on the other hand, since the magnitude estimation test provides ratio-scaled data, results are linearized by taking the log10 of the values and given in the second y-axis. Averages and standard deviations were calculated after the linearization.

A similar trend was observed between the three different methodologies, whereas the 'drops' between the different loudness levels (stimuli 3 to 4 and stimuli 6 to 7) were more obvious in the magnitude estimation. The slider scale and the random access methods had similar trends. These evaluation methods used the same scale and eventually showed similar standard deviations. In both cases, participants had a limited response scale, where they had to provide answers between the predefined numbers (i.e., 0–100), which reflect the categorizations with verbal anchors. Depending on the number of stimuli used for a listening experiment in annoyance evaluations, both methods can be used interchangeably. However, for an experiment with a large number of stimuli, the random access method can be disadvantageous for a participant, since it might be overwhelming to place many sound samples on the evaluation surface at the same time. From a similar perspective, access to all stimuli encourages a participant to play back every possible pair, which might lead the multiple-stimulus evaluation method to become a pairwise evaluation method. In contrast, in the slide scale method, where participants evaluated a single stimulus in each round,

they usually reported that they were not sure at the beginning of the test, so they wanted to change their previous evaluations depending on the newly available stimulus. The slider scale method does not provide participants an opportunity to go "back" and "correct" their response. However, with a proper training session and randomization of the order of the stimuli for each participant, we can eliminate this possible bias, which we call "beginning bias". Eventually, for the case with nine stimuli, both methods showed similar tendencies.

However, a magnitude estimation has its own advantages and disadvantages. The main disadvantage of using a magnitude estimation in annoyance evaluations is the question itself. The main feedback from the participants was that they could not estimate "what was two/three times more/less annoying". These estimations are more suitable for evaluating better scalable quantities, for example, "two times longer" or "three times larger surface area". For a line with a given length, participants can better "estimate" the length of a second line; however, the same approach is not always clear for participants of annoyance evaluations. The second disadvantage can be seen in the selection of the anchor stimulus. Here, stimulus 1, which had the lowest loudness and sharpness values, was selected as the anchor stimulus. Each comparison that depends on this particular stimulus can generate different biases [41]. However, a more detailed investigation of every possible pairwise comparison of the data can be provided using a magnitude estimation. For example, a comparison of pairs 1–3 and 1–4 showed that participants could tolerate a louder tone (stimulus 4) better than a stimulus with the same loudness but relatively high sharpness (stimulus 3). However, the standard deviation of stimulus 3 makes this inference relatively difficult. In contrast, it was not possible to state a similar trend between stimuli 3 and 4 for Part 1 and Part 3.

The individual results for each part are shown in Figures 18–21. In these four figures, the annoyance estimations of each loudness level and sharpness level are averaged over the number of subjects, and the error bars represent the 95% confidence intervals. In all four results, curves representing the different sharpness levels were almost parallel to each other, indicating no interaction between these two quantities. The rate of change of annoyance with changing sharpness was not different at each loudness level. In addition, a repeated measures ANOVA was performed for all tests. There was a separate statistically significant effect of loudness for all parts. The same significant effect was observed for sharpness. However, the interaction effect between loudness and sharpness was not significant in all cases. The results of the repeated measures ANOVA can be found in Table 8.

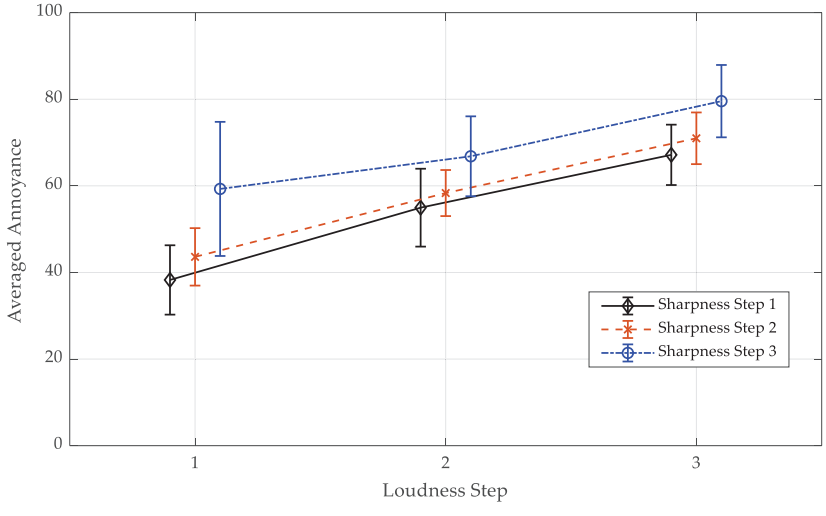


Figure 18. Average annoyance evaluations for each loudness and sharpness level for Part 1 (bars 95% CI).

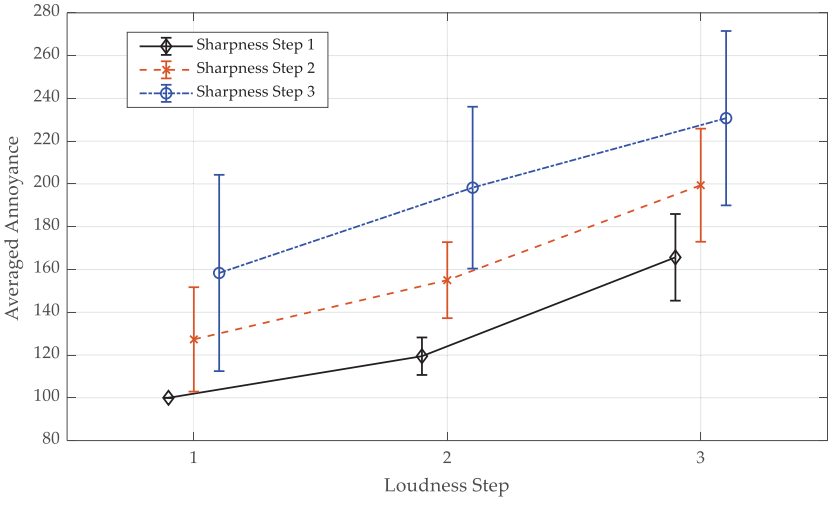


Figure 19. Geometric average annoyance evaluations for each loudness and sharpness level for Part 2 (bars 95% CI).

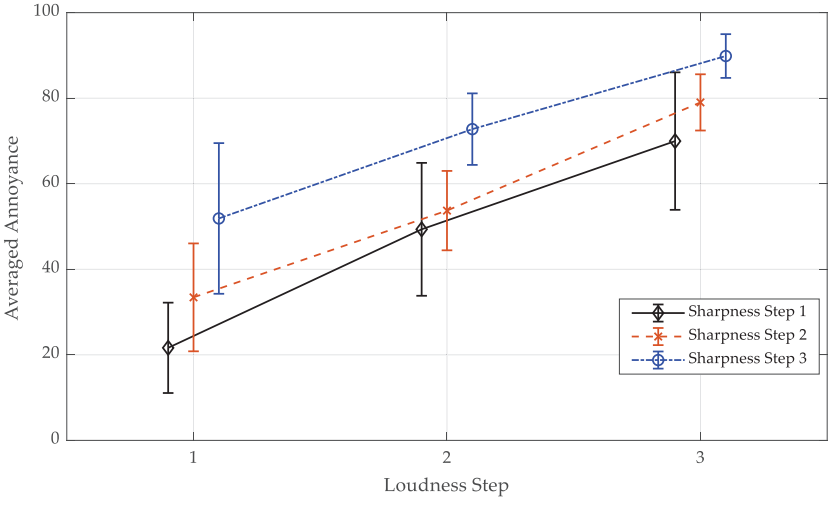


Figure 20. Average annoyance evaluations for each loudness and sharpness level for Part 3 (bars 95% CI).

Finally, a comparison between Part 3 and Part 4 is shown in Figure 22. The three panels of this figure show the three sharpness levels used in both experiments. Part 4 has five loudness levels, whereas Part 3 has three loudness levels. In each panel of this figure, it can be seen that the slopes are almost the same in both experiments. This means that changing the loudness at each sharpness level results in an equal change in annoyance for both experiments. In the right panel, it is possible to see that the absolute annoyance evaluations for Part 3 are higher than those for Part 4. It appears that the participants scaled their evaluations for the maximum loudness and sharpness levels to fit within the given evaluation space (from 0 to 100).

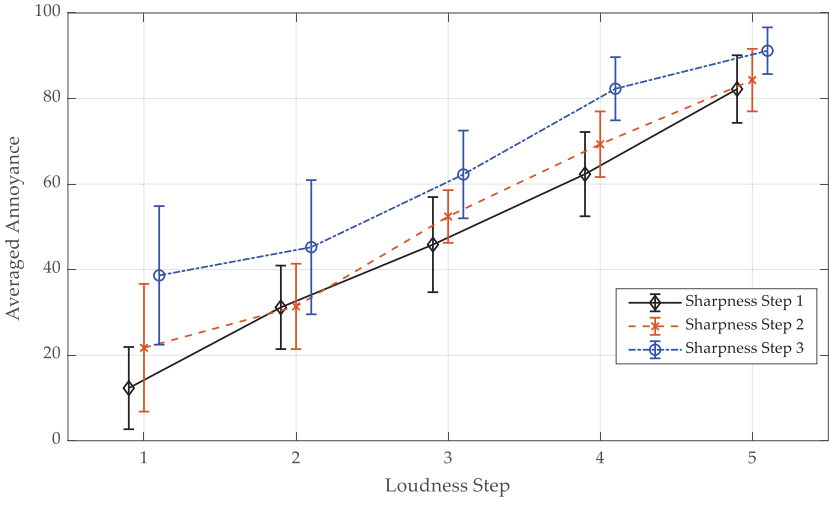


Figure 21. Average annoyance evaluations for each loudness and sharpness level for Part 4 (5N × 3S) (bars 95% CI).

Table 8. Results of the repeated measures ANOVA for loudness, sharpness and loudness–sharpness interaction.

Part	Loudness	Sharpness	Loudness × Sharpness
1	$F(1.15, 9.37) = 44.73$ $p < 0.001$, partial $\eta^2 = 0.85$	$F(2, 16) = 11.46$ $p < 0.001$, partial $\eta^2 = 0.59$	$F(4, 32) = 2.13$ $p = 0.100$, partial $\eta^2 = 0.21$
2	$F(1.15, 9.18) = 46.26$ $p < 0.001$, partial $\eta^2 = 0.86$	$F(1.14, 9.13) = 20.15$ $p < 0.001$, partial $\eta^2 = 0.72$	$F(1.88, 15) = 1.27$ $p = 0.308$, partial $\eta^2 = 0.14$
3	$F(2, 16) = 36.05$ $p < 0.001$, partial $\eta^2 = 0.82$	$F(1.2, 9.57) = 16.32$ $p = 0.002$, partial $\eta^2 = 0.67$	$F(4, 32) = 1.36$ $p = 0.270$, partial $\eta^2 = 0.15$
4	$F(1.31, 10.47) = 53.77$ $p < 0.001$, partial $\eta^2 = 0.87$	$F(1.13, 9) = 24.56$ $p < 0.001$, partial $\eta^2 = 0.75$	$F(8, 64) = 1.92$ $p = 0.072$, partial $\eta^2 = 0.19$

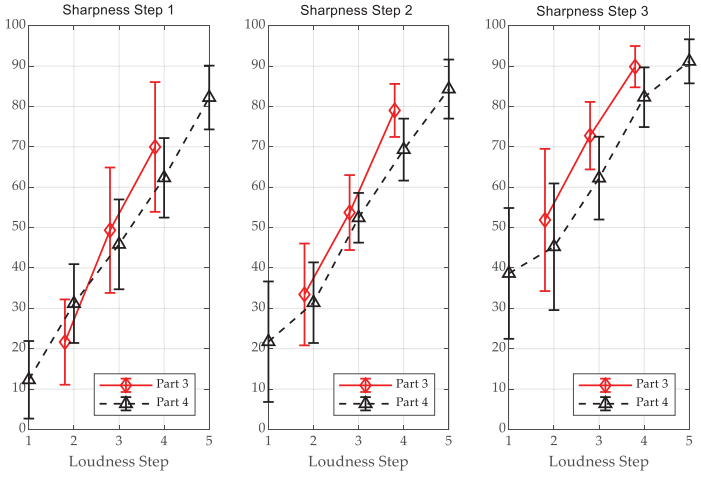


Figure 22. Comparison of Part 3 and Part 4: random access method with different loudness ranges (bars 95% CI).

5. Summary and Discussion

This study included a wide range of vacuum cleaner recordings selected from the market in a controlled manner. In particular, the sound power levels of the devices according to the manufacturers show a fine distribution among the observed ranges in the market, and the distribution is not stacked or concentrated on specific dB(A) values.

The recordings showed the variability in acoustic and psychoacoustic parameters and their ranges among the selected devices. Based on this observed variability, it was possible to derive the common characteristics of canister-type vacuum cleaner noise. These common characteristics were then compared with those in the literature on vacuum cleaner noise generation mechanisms. The observed variability was comparable with those in the literature. The measured ranges can be considered the limits of acoustic and psychoacoustic values available in the market. Ultimately, it was possible to define prototypical vacuum cleaner noise. This prototypical vacuum cleaner noise provided insight into the possible level ranges: frequency content and tonal content (i.e., frequency and intensity, respectively). Any reader working on vacuum cleaner noise can compare a measurement with the defined ranges in this study to verify that the limits defined in this study are adequate at representing the entire vacuum cleaner population. If new values emerge, either due to a new sampling method (e.g., selection of different vacuum cleaners) or a new technological advancement (e.g., decreasing levels), then it is possible to extend and improve this study to a more inclusionary approach between the vacuum cleaner noise annoyance studies available in the literature. In that manner, it should be possible to obtain reproducible results between different research groups working on the sound quality of vacuum cleaners. Furthermore, the definition of prototypical vacuum cleaner noise can help future studies make parametric modifications of the defined noise and investigate the influence of salient noise characteristics on annoyance ratings.

Recording condition is a static condition of a vacuum cleaner that must be taken into account because normal working conditions can change its emitted sound. However, this effect is rather random, and due to this complexity, it is not possible to generate a comparison baseline for different vacuum cleaners.

Prototypical vacuum cleaner noise can be explained as follows: vacuum cleaner noise is quasi-stationary and has an increasing A-weighted level of about 500 Hz, where the highest level is mostly reached. In this range, most vacuum cleaners have a tone of approximately at 100 Hz, which varies in amplitude depending on the device. At frequencies higher than 500 Hz, A-weighted vacuum cleaner noise tends to decrease, with the range changing depending on whether hard flooring or carpet is used. After 5 kHz, the rate of decrease in the A-weighted levels usually increases. The noise levels reach a value below a threshold of about 10 kHz. In this defined range, different vacuum cleaners show different levels, although the main structure remains essentially the same. Among the defined frequency ranges and their intensities, vacuum cleaners have many tonal components lying in different frequencies. However, it can be roughly categorized that the tones are concentrated in three regions: the first region is around 100 Hz, the second region is approximately 200–800 Hz, and the last region is approximately 1000–10,000 Hz. These values are calculated based on the tonality standard DIN 45681 [39] and the hearing model of Sottek [29]. Additionally, the ranges of the psychoacoustical metrics calculated in this study are given in Table 3, so any further study of vacuum cleaner sound quality can verify the reliability of these values, based on whether a new recording's values are inside or outside of these defined ranges, keeping the recording conditions in mind.

In the second part of this study, the main correlates of the annoyance evaluations of vacuum cleaner noise were obtained in two listening tests. The first listening test included original and modified vacuum cleaner noise samples. The main correlates of the annoyance evaluations were found in this listening experiment. The second listening experiment was divided into four parts, and each part was designed in a full factorial experiment (between loudness and sharpness) with different experimental methods and ranges. The possible interaction between loudness and sharpness was investigated in these experiments.

The first listening test showed that the overall loudness, sharpness and especially tonal components at lower and higher frequencies play crucial roles in annoyance perception. The correlations between these three parameters and annoyance were found to be significant. The coefficients for the three correlations were found to be 0.963 for loudness, 0.763 for sharpness, 0.354 for tonality at low frequencies and 0.454 for tonality at high frequencies.

However, there is a relatively strong correlation between loudness and sharpness (0.778) and a moderate correlation between loudness and hearing model tonality (0.493). Although there is a strong correlation between loudness and sharpness, which might hint at a degree of multi-collinearity, sharpness was taken into account due to two reasons: Firstly, based on the range of differences with high frequencies, observed in Figure 8, it makes sense to include sharpness as a parameter due to the variation. It is possible to have the same loudness values and different sharpness values. Secondly, the broadband noise-like nature of vacuum cleaner sounds changes its color significantly by changing the high-frequency content. An expert listening to the recordings can directly relate the mentioned characteristics: different vacuum cleaners have different band-noise characteristics with different amounts of high-frequency content. Moreover, changing the high-frequency content of vacuum cleaner noise is achievable by applying sound-absorbing materials at the air exit and other slits, as observable in some of the "low noise" vacuum cleaners on the market. Eventually, variation in the sharpness can be achieved by means of noise reduction techniques, as mentioned in different pieces of literature referred to in this study.

In the first three parts of the second experiment (Part 1, Part 2 and Part 3), a 3×3 factorial design was used in different experimental methods, and the significance of the loudness–sharpness interaction was tested using a repeated measures ANOVA. In the last part, the loudness range was extended in the 5×3 factorial design experiments so that the investigated range of loudness was close to the range observed from market research and the interaction could be investigated in louder and quieter stimuli. For all four parts, no significant interaction between loudness and sharpness was found.

In Parts 1, 2 and 3, three different experimental methods were compared with each other using the same stimuli, same subjects and same playback conditions. The investigated methods were slider scale (Part 1), magnitude estimation (Part 2) and random access (Part 3). As expected, the slider scale and random access methods showed quite high similarity, whereas the magnitude estimation method showed clear distinctions for loudness level changes, although statistical significance was not observed when the entire database was considered.

This different behavior from the magnitude estimation test can be the reason for the logarithmic bias [41] since the stimulus with the lowest loudness and sharpness values was used as the anchor stimulus. In future studies, this effect could be further investigated using another anchor stimulus, such as the other extreme of the stimulus pool (the loudest and sharpest stimulus) or a stimulus right in the middle. In addition, after the magnitude estimation tests, participants usually commented that evaluations such as "two times more annoying" were rather complicated for them, compared with using the available scale with verbal anchors.

The results found in this study are similar to those of the cited studies on this topic [2,4,8]. It was found that loudness and sharpness were strongly correlated with annoyance. In addition to these two terms, high correlations were found between roughness and fluctuation strength, and ratings of annoyance in the cited studies. Furthermore, tone-to-noise ratio was strongly correlated with the annoyance ratings. However, the cited studies do not include the correlations among the input parameters, so it is difficult to say whether the reported high correlations have direct psychoacoustic significance or whether the dominant effect of loudness is reflected in other input parameters due to multi-collinearity. Apart from that, a direct comparison with the other cited papers is not possible because they differ in content and methodology. Yoshida's [1] work is a special case for a listening attitude (active and passive listening), and since they have used only upright cordless vacuum cleaners, a direct comparison was not feasible. Additionally, Lyon [9] used

a different approach, where the variation in signal characteristics was obtained via real mechanical modification of the device. Hence, the results were given based on these mechanical modifications but were not dependent on the psychoacoustic parameters. Hence, a direct comparison was not possible.

Finally, it is important to point out the potential limitations of this study: Although the selection of vacuum cleaners using maximum power and sound power levels was justified, each sampling is inherently subject to error. As with any other study of sound quality, a different selection of stimuli could lead to different results in the correlations. However, this limitation was minimized by including additional synthetic stimuli. Sound recordings were made under anechoic conditions. It is reasonable to assume that the actual auditory effect might be different if the same devices were operated under normal room conditions. However, room acoustic conditions are completely arbitrary and cannot be a reasonable basis for comparing different devices. The correlations obtained in the first listening test are only valid for the applied test method. As can be seen from the results of listening experiment 2, different test methods can also lead to different results. Finally, the significance and effect size obtained in listening experiment 2 could be different in an experiment with more participants.

As future work, the effect of tonality should also be investigated in a factorial design, allowing for a full-factorial design between all the major correlates of annoyance found in listening test 1. However, this could involve many input parameters with many levels, resulting in too many stimuli, which is not feasible for use in a single experiment. There, an experiment method should be defined that allows for separate experiments and their combined evaluations with as little biases or errors as possible.

Author Contributions: Conceptualization, S.A.; data curation, S.A.; formal analysis, S.A.; funding acquisition, S.A. and M.E.A.; investigation, S.A.; methodology, S.A.; project administration, S.A. and M.E.A.; resources, S.A.; supervision, M.E.A.; validation, S.A.; visualization, S.A.; writing—original draft, S.A.; writing—review and editing, M.E.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Federal Ministry of Economic Affairs and Climate Action (BMWK) based on a decision by the German Bundestag, grant number KK5049502LB0.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki and approved by the Ethics Committee of TU Dresden (protocol code SR-EK-474102021, by 06.12.2021).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Yoshida, J.; Hatta, I. Influence of hearing attitude difference on sound quality evaluation of vacuum cleaner sound. *Acoust. Sci. Technol.* **2021**, *42*, 46–49. [CrossRef]
2. Kumar, S.; Wing, W.S.; Lee, H.P. Psychoacoustic Analysis of Vacuum Cleaner Noise. *Acoustics* **2021**, *3*, 545–559. [CrossRef]
3. Altinsoy, E.; Kanca, G.; Belek, H.T. A Comperative Study on the Sound Quality of Wet and Dry-Type Vacuum Cleaners. In Proceedings of the Sixth International Congress on Sound and Vibration, Copenhagen, Denmark, 5–8 July 1999; p. 8.
4. Altinsoy, M.E. Towards an European sound label for household appliances: Psychoacoustical aspects and challenges. In Proceedings of the 4th International Workshop on Perceptual Quality of Systems, Vienna, Austria, 2–4 September 2013.
5. Martin, N.; Lehmann, J.; Wadle, L.-M. Relationship between acoustic perception and overall user experience in vacuum cleaners. In Proceedings of the EuroNoise, Madeira, Portugal, 25–27 October 2021.
6. Takada, M.; Arase, S.; Tanaka, K.; Iwamiya, S. Economic valuation of the sound quality of noise emitted from vacuum cleaners and hairdryers by conjoint analysis. *Noise Control Eng. J.* **2009**, *57*, 263. [CrossRef]
7. Takada, M. Design and value of product sound. In Proceedings of the InterNoise, Madrid, Spain, 16–19 June 2019.

8. Ih, J.-G.; Lim, D.-H.; Shin, S.-H.; Park, Y. Experimental design and assessment of product sound quality: Application to a vacuum cleaner. *Noise Control Eng. J.* **2003**, *51*, 244. [CrossRef]
9. Lyon, R. *Designing for Product Sound Quality*, 1st ed.; CRC Press: New York, NY, USA, 2000.
10. Rukat, W. The Noise Emitted by a Vacuum Cleaner Treated as a Device with Extensive Sound Sources. *Vib. Phys. Syst.* **2018**, *29*, 2018015.
11. Buratti, C.; Moretti, E.; Urbani, M. Experimental Acoustics Characterization of a Wet Vacuum Cleaner. In Proceedings of the Sixteenth International Congress on Sound and Vibration, Krakow, Poland, 5–9 July 2009; p. 8.
12. Čudina, M.; Prezelj, J. Noise generation by vacuum cleaner suction units Part I: Noise generating mechanisms—An overview. *Appl. Acoust.* **2007**, *68*, 491–502. [CrossRef]
13. Čudina, M.; Prezelj, J. Noise generation by vacuum cleaner suction units Part II. Effect of vaned diffuser on noise characteristics. *Appl. Acoust.* **2007**, *68*, 503–520. [CrossRef]
14. Čudina, M.; Prezelj, J. Noise generation by vacuum cleaner suction units. Part III. Contribution of structure-borne noise to total sound pressure level. *Appl. Acoust.* **2007**, *68*, 521–537. [CrossRef]
15. Novaković, T.; Ogris, M.; Prezelj, J. Validating impeller geometry optimization for sound quality based on psychoacoustics metrics. *Appl. Acoust.* **2020**, *157*, 107013. [CrossRef]
16. Brungart, T.A.; Lauchle, G.C. Modifications of a handheld vacuum cleaner for noise control. *Noise Control Eng. J.* **2001**, *49*, 73. [CrossRef]
17. Brungart, T.A.; Lauchle, G.C.; Ramanujam, R.K. Installation effects on fan acoustic and aerodynamic performance. *Noise Control Eng. J.* **1999**, *47*, 3. [CrossRef]
18. Teoh, C.-Y.; Lim, J.; Hamid, M.N.A.; Ooi, L.E.; Tan, W.H. Suppression of Flow-Induced Noise of a Canister Vacuum Cleaner. *Int. J. Integr. Eng.* **2023**, *15*, 180–190.
19. Staubsaugern-Mit und Ohne Beutel, Saturn. Available online: https://www.saturn.de/de/category/staubsauger-1581.html (accessed on 20 February 2023).
20. Bodenstaubsaugern-mit und ohne Beutel, Mediamarkt. Available online: https://www.mediamarkt.de/de/category/bodenstaubsauger-830.html (accessed on 20 February 2023).
21. *IEC 60704-2-1:2020*; Household and Similar Electrical Appliances—Test Code for the Determination of Airborne Acoustical Noise—Part 2-1: Particular Requirements for Dry Vacuum Cleaners. International Electrotechnical Commission: Geneva, Switzerland, 2020.
22. *ISO 532-1*; Acoustics—Methods For Calculating Loudness—Part 1: Zwicker Method. International Organization for Standardization: Geneva, Switzerland, 2017.
23. *DIN 45631/A1*; Berechnung des Lautstärke-pegels und der Lautheit aus dem Geräuschspektrum–Verfahren nach E. Zwicker. German Institute for Standardization: Berlin, Germany, 2010.
24. *ISO 532-2*; Acoustics—Methods for Calculating Loudness—Part 2: Moore-Glasberg Method. International Organization for Standardization: Geneva, Switzerland, 2017.
25. Aures, W. Berechnungsverfahren für den sensorischen Wohlklang beliebiger Schallsignale. *Acta Acust. United Acust.* **1985**, *59*, 130–141.
26. von Bismarck, G. Sharpness as an Attribute of the Timbre of Steady Sounds. *Acta Acust. United Acust.* **1974**, *30*, 159–172.
27. *DIN 45692*; Messtechnische Simulation der Hörempfindung Schärfe. German Institute for Standardisation: Berlin, Germany, 2009.
28. Terhardt, E.; Stoll, G.; Seewann, M. Algorithm for extraction of pitch and pitch salience from complex tonal signals. *J. Acoust. Soc. Am.* **1982**, *71*, 679–688. [CrossRef]
29. ECMA International. *ECMA-74 15th Edition/July 2018, D.7, Tone-to- Noise Ratio Method*; ECMA International: Geneva, Switzerland, 2018.
30. Bray, W.R. A New Psychoacoustic Method for Reliable Measurement of Tonalities According to Perception. In *INTER-NOISE and NOISE-CON Congress and Conference Proceedings*; Institute of Noise Control Engineering: Reston, VA, USA, 2018.
31. Hots, J.; Verhey, J. Experimentelle Erfassung von Tonzuschlagen. In Proceedings of the DAGA 2015, Nürnberg, Germany, 16–19 March 2015.
32. Oliva, D.; Hongisto, V.; Haapakangas, A. Annoyance of low-level tonal sounds—Factors affecting the penalty. *Build. Environ.* **2017**, *123*, 404–414. [CrossRef]
33. Becker, J.; Sottek, R.; Lobato, T. *Progress in Tonality Calculation*; Universitätsbibliothek der RWTH Aachen: Aachen, Germany, 2019; p. 8.
34. Hongisto, V.; Saarinen, P.; Oliva, D. Annoyance of low-level tonal sounds—A penalty model. *Appl. Acoust.* **2019**, *145*, 358–361. [CrossRef]
35. Melian, A.G.; Nykänen, D.A. Perception of Tones below 1 kHz in Electric Vehicles. In *INTER-NOISE and NOISE-CON Congress and Conference Proceedings*; Institute of Noise Control Engineering: Reston, VA, USA, 2019; p. 11.
36. Pietila, G.; Seldon, W.; Roggenkamp, T.; Bohn, T. Tonal Annoyance Metric Development for Automotive Electric Vehicles. In Proceedings of the Noise and Vibration Conference & Exhibition, Grand Rapids, MI, USA, 10–13 June 2019. [CrossRef]
37. Sottek, R.; Becker, J. Tonal Annoyance vs Tonal Loudness and Tonality. In Proceedings of the InterNoise, Madrid, Spain, 16–19 June 2019.
38. Becker, J.; Sottek, R. Application of Psychoacoustic Tonality for Product Sound Development. In Proceedings of the DAGA 2020, Hannover, Germany, 16–19 March 2020.

39. *DIN 45681*; Bestimmung der Tonhaltigkeit von Geräuschen und Ermittlung Eines Tonzuschlages für die Beurteilung von Geräuschimmissionen. German Institute for Standardisation: Berlin, Germany, 2005.
40. Zwicker, E.; Fastl, H. *Psychoacoustics: Facts and Models*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2013; Volume 22.
41. Poulton, E.C. Models for Biases in Judging Sensory Magnitude. *Psychol. Bull.* **1979**, *86*, 777–803. [CrossRef] [PubMed]

applied sciences

MDPI

Article

Weakly Supervised U-Net with Limited Upsampling for Sound Event Detection

Sangwon Lee [1], Hyemi Kim [2] and Gil-Jin Jang [1,*]

[1] School of Electronic and Electrical Engineering, Kyungpook National University, Daegu 41566, Republic of Korea
[2] Electronics and Telecommunications Research Institute, Daejeon 34129, Republic of Korea
* Correspondence: gjang@knu.ac.kr; Tel.: +82-53-950-5517

Featured Application: Audio classification; music information retrieval; audio scene characterization; temporal localization of sound sources; audio indexing; audio surveillance systems; anomaly detection from audio sounds.

Abstract: Sound event detection (SED) is the task of finding the identities of sound events, as well as their onset and offset timings from audio recordings. When complete timing information is not available in the training data, but only the event identities are known, SED should be solved by weakly supervised learning. The conventional U-Net with global weighted rank pooling (GWRP) has shown a decent performance, but extensive computation is demanded. We propose a novel U-Net with limited upsampling (LUU-Net) and global threshold average pooling (GTAP) to reduce the model size, as well as the computational overhead. The expansion along the frequency axis in the U-Net decoder was minimized, so that the output map sizes were reduced by 40% at the convolutional layers and 12.5% at the fully connected layers without SED performance degradation. The experimental results on a mixed dataset of DCASE 2018 Tasks 1 and 2 showed that our limited upsampling U-Net (LUU-Net) with GTAP was about 23% faster in training and achieved 0.644 in audio tagging and 0.531 in weakly supervised SED tasks in terms of F1 scores, while U-Net with GWRP showed 0.629 and 0.492, respectively. The major contribution of the proposed LUU-Net is the reduction in the computation time with the SED performance being maintained or improved. The other proposed method, GTAP, further improved the training time reduction and provides versatility for various audio mixing conditions by adjusting a single hyperparameter.

Keywords: sound event detection; U-Net; weakly supervised learning; pooling

Citation: Lee, S.; Kim, H.; Jang, G.-J. Weakly Supervised U-Net with Limited Upsampling for Sound Event Detection. *Appl. Sci.* **2023**, *13*, 6822. https://doi.org/10.3390/app13116822

Academic Editor: Giovanni Costantini and Daniele Casali

Received: 18 February 2023
Revised: 29 May 2023
Accepted: 2 June 2023
Published: 4 June 2023

1. Introduction

The purpose of sound event detection (SED) is figuring out the types of sound sources contained in the input audio recordings. SED can be used in a surveillance system that detects the occurrence of a specific sound [1,2], voice activity detection (VAD) [3], or keyword spotting with the human voice [4]. Currently, SED systems are usually implemented by convolutional neural networks (CNNs) [5,6], recurrent neural networks (RNNs) [2], or combinations of them [7–9]. SED solutions early on usually relied on a supervised learning framework that required a completely labeled dataset [10] in which the onset and offset timings of the sound events were transcribed. However, complete labeling of all the timings would require too much effort from human listeners, and often, they are not available in practical situations. In more realistic cases, only the types of sound events of the audio recordings are known, as shown in Figure 1. Strong labels include onset and offset times in the given audio recordings, but weak labels give only the types of audio events. When incomplete information is given, the task is called weakly supervised classification [11–17]. Using weakly supervised learning with an incompletely labeled dataset might reduce the

costs for dataset construction. One of the weakly supervised classifications is generating a strongly labeled dataset from the given weakly supervised training data [16]. This framework generates strongly labeled sound event timings by clipping sound event audio, normalizing it, and then synthesizing it with normalized background audio or white noise. This is similar to data augmentation, but with this framework, it is possible to increase the quantity of the dataset. In addition, natural environments are polyphonic, meaning that multiple sounds may be active at the same time [1,2,17]. Therefore, the sound event occurrences overlap. There are no predefined rules on how sounds can co-occur, and the way to model co-occurrence is through random sampling with the degree of overlap of different classes given by or obtained from the data statistics. To detect polyphonic sound events, multi-label classifiers are required [1].

The SED model is usually based on weakly supervised CNNs [16–18]. When training CNNs, only the event labels are used as targets. In the detection stage, video object description models (VODMs) such as cascade R-CNN (regions with CNN features) [19], faster R-CNN [20], and the class activation map (CAM) approach [21] are used. The 2D object regions or class activation maps obtained by a trained CNN are used to predict the onset and offset times of the classified audio events. Another type of SED approach is using temporal models such as convolutional recurrent neural networks (CRNNs) [1,22] and Transformers [23]. These temporal models are able to find the onset and offset timings more accurately because they are inferred simultaneously with the class labels in a single learning stage. The aforementioned weakly supervised learning methods require generating onset and offset timings that are obtained in an unsupervised manner, so the accuracy of the detection varies greatly according to the types of audio sources and the characteristics of the recording environments. Self-supervised learning [24] and pseudo-labeling [25] methods for weakly supervised SED have been proposed, but their performances rely on specific audio events only. Recently, U-Net showed successful performance in image segmentation and also was applied to SED [16,26]. U-Net was originally designed for segmentation tasks, so in a weakly supervised manner, the segmentation targets are converted to sound event category labels by global pooling techniques such as global average pooling (GAP) [27,28], global max pooling (GMP) [16], and global weighted rank pooling (GWRP) [18]. However, these methods require averaging or sorting operations on the total feature map, so their is an extensive computational overhead.

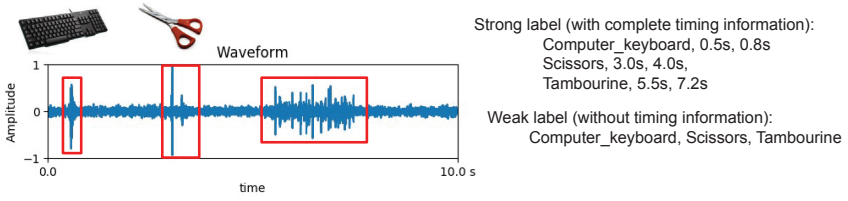


Figure 1. Examples of strong and weak labels. A strong label consists of a sequence of sound event types and onset and offset times in the given recording. A weak label consists of sound event types only.

In this paper, we propose two kinds of methods that can improve the performance of the weakly supervised SED, as well as reduce the time and space complexity. The proposed model is a modification of the U-Net structure [16,26] suited for sound event detection. The first method is a limited upsampling from the lower layers to the higher ones in the decoder part of U-Net. Because SED performs temporal segmentation only, we did not apply upsampling along the frequency axis in the U-Net decoder, and thus, the output map sizes were reduced greatly without performance loss. According to the experimental results, a total of 40% of the convolution output map size and 12.5% of the fully connected layer output were required to provide slightly better SED performance. The second proposed method is a global pooling technique called global threshold average

pooling (GTAP). The conventional GWRP requires sorting the total output map, and its time complexity grows with the growth of the output map size. The proposed method uses a fixed threshold to determine the set of the output map units for the average pooling. The threshold computation requires the mean and the standard deviation of the map values, which is much faster than the sorting operation. Moreover, higher SED performance can be obtained by adjusting the threshold with a single hyperparameter. The major contribution of the proposed methods is the reduction of the computation time without any performance loss. The size of the bottleneck layer of U-Net is the same for the proposed LUU-Net, so the amount of information for the audio events transferred to the final output is the same. The proposed LUU-Net limits the expansion along the frequency axis only, and the time resolution in the time domain is kept unchanged. While doing so, the network size of the encoder is reduced by up to $\frac{1}{8}$, resulting in a 40% reduction in the number of parameters. The proposed GTAP further improves the computation time by pruning unnecessary output map components, with the SED and AT performances being improved over the conventional CNN and U-Net.

The rest of this paper is organized as follows. Section 2 describes the conventional SED method based on U-Net and GWRP. Section 3 explains the proposed U-Net with limited upsampling and the proposed global threshold average pooling (GTAP). Section 4 gives the experiments' setup and the results with a detailed analysis. Finally, Section 6 summarizes our contribution.

2. Conventional Sound Event Detection

In this section, we explain the sound event detection problem in detail and the conventional methods for the problem.

2.1. Weakly Supervised Sound Event Detection Framework

Figure 2 shows the basic architecture of the weakly supervised learning framework for sound event detection. The input audio features are passed through a number of convolution layers to generate segmentation maps in 1D or 2D space. There are two outputs for audio tagging and sound event detection, respectively. In audio tagging, true labels are given for audio class indices, so a pooling is applied to convert the segmentation maps to probabilistic predictions, and the loss is computed with respect to the given true labels. In sound event detection, the output is interpreted as a segmentation map in 1D or 2D space. Because the onset and lengths of the sound events are not given in the training dataset, sound event timings cannot be the direct training targets. The audio tagging information—the class labels of the audio events—are used to infer the sound events. A convolutional neural network (CNN) model has been proposed for the weakly supervised learning [16]. It resembles the conventional VGG network [29] with modifications. In order to ensure for the output of the model is the same size as the input of the model, there are no downsampling layers such as max or average pooling, but only nine convolutional layers. However, this makes the receptive field smaller and computational cost very expensive.

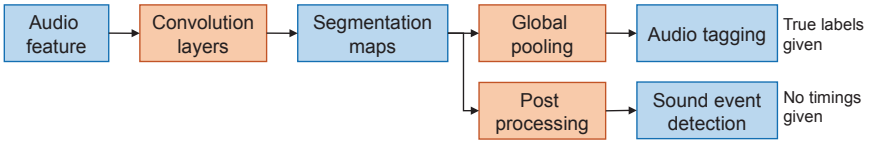


Figure 2. Weakly supervised sound event detection framework. The outputs of convolutional layers are segmentation maps and interpreted as audio tagging and sound event detection results. Because no ground truth for detection is provided, detection loss is not used in training the network. Only classification loss is computed.

2.2. U-Net for Sound Event Detection

We first describe the weakly supervised U-Net [26] for the sound event detection (SED) task. U-Net was proposed for medical image segmentation and has shown good performance in image segmentation tasks in various fields. This model has a convolutional encoder–decoder structure and shows high reconstruction performance thanks to the skip connection between the encoder and decoder. We decided to use this model because this kind of structure is good at solving the problem of the baseline model, and then, we used this model to implement a system that trains the SED task in a weakly supervised manner.

Figure 3 illustrates the U-Net architecture for the sound event detection model. In the left, the encoder part, a number of convolutional and pooling layers are repeatedly applied to reduce the input size from 311×64 to 39×8, with appropriate selections of the number of convolutional kernels. In the right, the decoder part, deconvolution is applied to restore the input size, 311×64, so that the final segmentation masks are of the same size as the input spectrogram. Note that the number of masks equals the number of classes. The category predictions are obtained by global pooling and used in computing the classification loss, and the detection results are obtained by averaging the frequency components. We used supervised cross-entropy loss to train the whole network [30], which is suitable for classification tasks with their targets expressed by a one-hot representation [31].

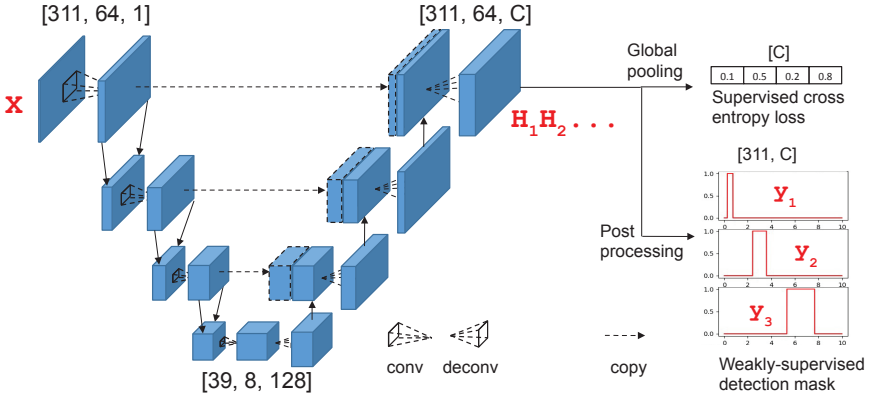


Figure 3. U-Net architecture for weakly supervised sound event detection with full upsampling in the deconvolutional layers. The left encoder part reduces the input size to a smaller size with a number of convolutional and pooling layers. The right half, the decoder part, restores the bottom output map of a reduced size to the original input size. The number of channels at the final output layer is the same as the number of classes (a positive integer "C" in this example). For each output channel, a global pooling layer is applied to derive the audio tagging outputs, so that the number of target nodes is the same as the number of tagging classes.

The final output of the conventional U-Net [26] is a event segmentation map of the detected classes with the same dimension as the input. Training U-Nets requires a true segmentation map, to minimize the amount mismatch between the predicted segmentation result and the ground truth for each sample. When the complete ground truth map is not available, but only the class labels that exist in the input sound are given, the connection from the segmentation layer of U-Net to the class label output layer is added. The size of the segmentation map is much larger than the number of classes, so we applied GWRP [18] to reduce the map size. The resultant dimension of the GWRP output is [batch_size, number_of_classes], so that audio tagging in time domain can be achieved in a supervised manner.

2.3. Postprocessing

Figure 4 shows the detailed postprocessing procedures. The output size should become $time \times length \times C$, where C is the number of event classes, as shown by the two-dimensional image in the bottom right part of Figure 3. Global pooling is performed to compute the prediction vector, whose dimension is the same as the number of classes. The prediction vector is then compared with the ground truth vector, and the prediction loss is used to train the model. In the three graphs at the bottom right of Figure 3, the y-axis is the class labels of the various sounds, and the x-axis represents the onset and the length of the sound events. This task is called audio tagging. The other task is sound event detection. The 2D classwise segmentation maps are then converted to class activation probabilities along the time axis (detection map), and simple thresholding yields the detection results, so the onset and offset times of the sound events are obtained. In sound event detection, the conversion of the 2D segmentation map to the 1D temporal detection map is the key issue. There are a number of global pooling methods for effective and efficient conversion.

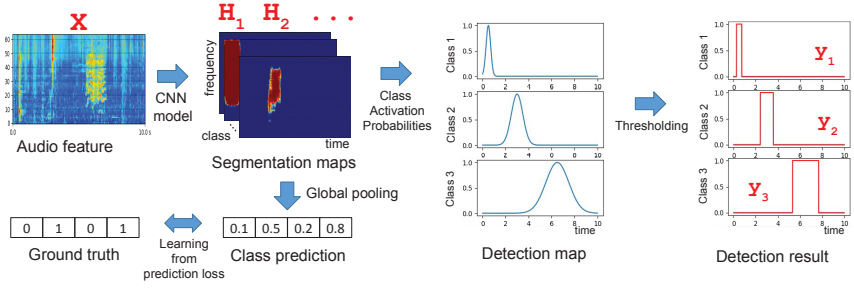


Figure 4. Postprocessing procedures for audio tagging and sound event detection. The spectro-temporal 2D audio feature map, denoted by $\mathbf{X}$, is converted to the 2D segmentation map of the same size ($\mathbf{H}_i$) and 1D class prediction vector of length C (the number of event classes) to compute the prediction loss for audio tagging and model training. The segmentation maps are converted to C detection maps of length T (time), and thresholding is performed to find the onset and offset of the sound events ($\mathbf{y}_i$).

2.4. Global Pooling

To make predictions for audio tagging, the 2D segmentation map is compressed to a scalar value, which is generally interpreted as a probability value of the occurrence of a specific event in the input recording. The occurrences do not have to be mutually exclusive, so each segmentation map is handled independently. Global max pooling (GMP) [27] is defined as follows:

$$\mathrm{GMP}(\mathbf{H}_c) = \max_{t,f} \mathbf{H}_c(t,f)\,, \tag{1}$$

where $\mathbf{H}_c$ is the segmentation map for class c and t and f are the indices of the time and frequency axes, respectively. The drawback of GMP is that it is sensitive to outliers and more affected as the segmentation map size increases. To overcome this problem, global average pooling (GAP) [27,28] provides much more stable prediction results, which is computed as

$$\mathrm{GAP}(\mathbf{H}_c) = \frac{1}{TF} \sum_{t=1}^{T} \sum_{f=1}^{F} \mathbf{H}_c(t,f)\,. \tag{2}$$

The GAP is less sensitive to outliers, but when an event occurs sparsely, the activation of the event becomes too small to be detected as having occurred. One of the efficient pooling methods that balances outlier robustness and the sparsity problem is global weighted rank pooling (GWRP) [18]. This GWRP operation is defined as follows:

$$\begin{aligned} \mathbf{h}_c &= \underset{t,f}{\text{downsort}}(\mathbf{H}_c) \\ \text{GWRP}(\mathbf{H}_c) &= \frac{1}{\sum_{i=1}^{TF} d^{i-1}} \sum_{i=1}^{TF} d^{i-1} \mathbf{h}_c(i)\,, \end{aligned} \tag{3}$$

where the function "downsort" sorts the 2D input in descending order to generate a 1D sorted list of the whole elements, so $\mathbf{h}_c^i$ is the ith largest value in $\mathbf{H}_c$ with 2D values flattened to a 1D vector $\mathbf{h}_c$. A hyperparameter $0 \leq d \leq 1$ is a decaying weight enabling small activations to contributed less to the computation of the pooling output. This is a generalized weighted pooling, which is equal to GAP for $d = 1$ and GMP for $d = 0$. GWRP is good for a weakly supervised SED task because GAP overestimates and GMP underestimates the tagging result [16]. However, GWRP also has a drawback that the computation speed is very slow because the sorting must be preceded by the calculation of the weight, and it is hard to find the proper hyperparameter d.

3. Proposed Method

In this section, we describe a combination of two proposed methods for sound event detection. The first one is a novel U-Net architecture and a pooling method to improve the computational efficiency compared to the conventional U-Net. The second method uses a subset of feature maps from U-Net without sorting, denoted as global threshold average pooling (GTAP).

3.1. U-Net with Limited Upsampling

Figure 5 describes the proposed U-Net architecture with limited upsampling. Even though the segmentation map is not learned by direct loss minimization, its prediction can be obtained while minimizing the classification loss of each time frame. The U-Net-based architecture learns to generate an activation map of each event in the time–frequency domain. However, some events that activate only a few parts of the frequency bins could be ignored after postprocessing because of the average pooling. To handle this problem, reducing the frequency axis to 1 inside the model is not a good way since it removes too much representation in the middle of the model. Therefore, we further applied limited upscaling along the frequency axis in the decoder part of U-Net to keep high activation in the encoded feature. More specifically, the baseline U-Net in Figure 3 uses deconvolution with stride $(2, 2)$ for upscaling in the decoder. However, the proposed U-Net uses deconvolution with stride $(2, 1)$. Therefore, compared with the existing U-Net, the size of the decoder's feature map is smaller, which reduces the computational cost. In addition, the size of data transmitted in the skip connection between the encoder and decoder is different from the feature of the decoder. To adjust this, we downscaled the data size by using average pooling with $kernel = stride = (1, 2^n)$.

The encoder part is unchanged, but the decoder upsamples only the time axis of the encoded features. In the example shown in Figure 5, the frequency range shrinks from 64 to 8, and the shrink size 8 is maintained in the final segmentation map. The time range changes from 311 to 39, and it goes back to 311 to restore the original time range. Upsampling in the deconvolutional layers is often unreliable because the information of the input is lost by the U-Net encoder and should be regenerated, resulting in a great amount of generation errors at the output. In a weakly supervised manner, the target information is incomplete, so more uncertainty is likely to propagate through the network, especially in the training steps. As a result, this structure reduces the size of the feature handled by the decoder, further increasing the computational efficiency. If the upscaling of the frequency axis actually interferes with learning in a weakly supervised manner, removing unnecessary upscaling may yield higher detection performance.

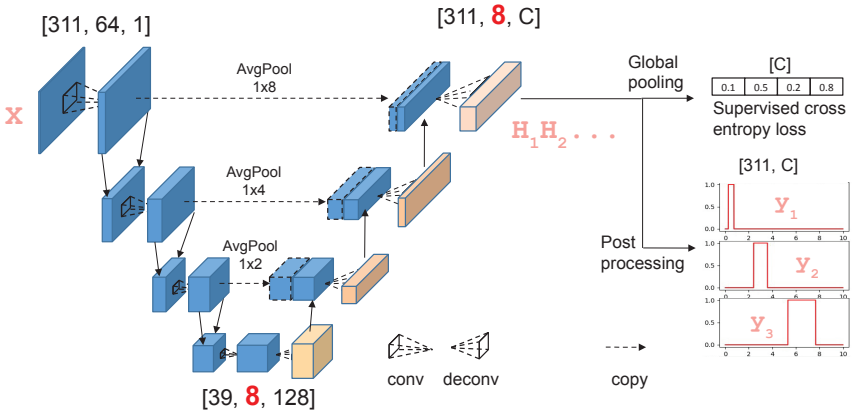


Figure 5. U-Net architecture with limited upsampling in the deconvolutional layers. The left encoder part is the same as U-Net, but in the right, the decoder part, it only upsamples along the time axis to match the input time range.

By comparing the U-Net in Figure 3 and the proposed LUU-Net in Figure 5, the size of the bottleneck layer between the encoder and decoder networks is $39 \times 8 \times 128$, and it is the same for both models. The information transferred from the input to the output through the network is the same. In U-Net, the deconvolutional layers in the decoder part restore the original frequency and time axes to obtain the segmentation masks in the frequency and time domain. However, SED does not require the segmentation mask in the frequency domain. The proposed LUU-Net expands the time axis only, and the segmentation mask can be obtained more reliably by removing unnecessary expansion from the bottleneck layers with the same amount of information. Therefore, LUU-Net is more efficient than U-Net in SED tasks without loss of information.

3.2. Global Threshold Average Pooling

The GWRP in Section 2.4 requires a sorted list of whole segmentation map elements. The sorting is required both in training and testing, so there is a large computational overhead. Moreover, the hyperparameter d is adjusted by the amount of events occurring in the map. In Equation (3), d^{i-1} decreases as i increases because $0 \leq d \leq 1$, and the value of d should be relatively large if the sound event occurs densely or the length of the event is long and small if it occurs sparsely or its length is short. When long and short events are mixed in the input recording, which is usual in real situations, it is very hard to determine a single value of d to guarantee the detection performance. Global max pooling is not influenced by the event lengths; however, it is sensitive to the outliers, and most of the segmentation map values, except the maximum, are discarded, so learning through error backpropagation may not work well.

To overcome the problems of GWRP, computational overhead, and hyperparameter adjustment, we propose a novel global pooling using a learnable threshold value. To begin with, we define the following threshold function, which is similar to the standard step function:

$$g(m, \theta) = \begin{cases} 1, & \text{if } m \geq \theta \\ 0, & \text{if } m < \theta \end{cases}, \tag{4}$$

where m is the input and θ is a given threshold value. The key idea of the proposed method is averaging only the values larger than the threshold. We define global threshold average pooling (GTAP) as follows:

$$
\begin{aligned}
h_{tf} &\triangleq \mathbf{H}_c(t,f) \\
\mathrm{GTAP}(\mathbf{H}_c) &= E_{\geq\theta}[\mathbf{H}_c] = \frac{\sum_t \sum_f h_{tf} g(h_{tf},\theta)}{\sum_t \sum_f g(h_{tf},\theta)} \quad , \qquad (5)
\end{aligned}
$$

where h_{tf} is the (t,f)-element of the 2D segmentation map for class c, $\mathbf{H}_c$, defined in Equation (1). $E_{\geq\theta}[\cdot]$ is a conditional expectation, so the GTAP function is the average of the values larger than threshold θ. In the actual implementation, we used the following equivalent equation for more efficient calculation.

$$
\begin{aligned}
\mathrm{ReLU}(x) &= \max(x,0) \\
\mathrm{GTAP}(\mathbf{H}_c) &= \frac{\sum_t \sum_f \mathrm{ReLU}(h_{tf}-\theta)}{\sum_t \sum_f g(\mathrm{ReLU}(h_{tf}-\theta),0)} + \theta \quad , \qquad (6)
\end{aligned}
$$

where "ReLU" is the rectified linear unit commonly used in deep neural networks.

The threshold θ is also relevant to the frequency and the length of the events. With the assumption that the segmentation map is distributed by a Gaussian, the appropriate threshold value is found by

$$\theta = mean(\mathbf{H}) + \alpha std(\mathbf{H}) \,, \qquad (7)$$

where "*mean*" and "*std*" are the mean and the standard deviation of the segmentation map and α is a hyperparameter to define how tightly to cut off the segmentation map. Because it is hard to use a class-specific threshold, we used all the training data to compute the mean and the standard variation. If $\alpha < 0$, the threshold value becomes smaller, which results in overestimation over $\alpha = 0$. If $\alpha > 0$, the threshold increases and the trained model may underestimate. A grid search is used to determine the appropriate value of α using a validation set. The detailed procedure is explained in the Experiments Section.

4. Experiments

We used DCASE 2018 Task 1 and Task 2 data [32,33] to create a mixed dataset of 8000 audio samples. The original sampling rates of the DCASE dataset are 48 kHz and 44.1 kHz, and we downsampled all the data to 32 kHz. The mixed dataset was created by adding the audio recordings of Task 1 and background sounds at 0 dB and adding other audio clips of various lengths for the simulated sound events.

4.1. Dataset Generation

According to the data augmentation in previous work [16], we built a large SED training dataset from audio tagging samples by synthesizing several audio clips with white noise or other types of audio sounds as background noise sounds. This framework is regarded as one of the data augmentation methods, allowing the generation of the training dataset with the various choices of different sounds. When adding a number of different sounds, their onset times and clipping lengths are varied to simulate various overlapping cases in real situations. Figure 6 shows the procedures of training data generation. The audio sounds are represented by 2D time–frequency spectrograms. There are many combinations of onset times and clipping lengths and how much audio is in a single recording. These combinations help the trained model work well with various SED tasks.

Audio classification experiments were carried out on 2 different datasets to evaluate the proposed method. DCASE 2018 Task 1 is a dataset consisting of a total of 8640 audio samples [32]. Each of them is 10 s long and sampled at 48 kHz. We used it as background sounds in evaluation data synthesis. The DCASE 2018 Task 2 is an audio tagging task [33]. The tagging dataset consists of about 9500 training samples of 41 categories, which are distributed unequally and sampled at 44.1 kHz. The smallest category has 94 samples and the largest 300 samples.

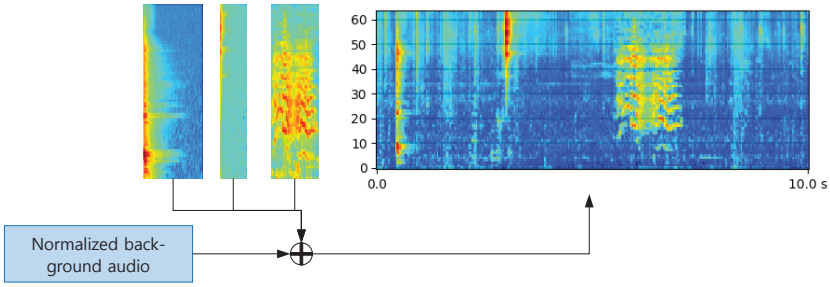


Figure 6. Training data generation procedure. Two audio samples are mixed with a background sound. Audio samples are clipped to a given length, normalized, and then mixed with normalized background noise. The x-axis is time in seconds, and for the y-axis frequency bin index, the larger the higher it is.

We adopted the policy suggested by the DCASE Challenge Guidelines [16] to generate the training dataset. In the original policy suggested by the DCASE Challenge, 3 distinct audio files were chosen from DCASE Task 2, clipped up to 2 s, and combined with additional background sounds to generate samples for training. The onset times were 0.5 s, 3 s, and 5.5 s, so there was no overlap among the 3 sound events. Besides the original policy, we performed several different synthetic polices: random onset, longer clipping, and mixed policies. Figure 7 shows the generated sample according to each policy. The longer clipping policy uses the maximum clip length of 5 s, which is longer than the 2 s suggested by the original guideline. This policy has a high probability of generating a sample containing overlapping events. In the random onset policy, the onsets of the events are randomly chosen from $[0.5, 6.5)$. This policy has a smaller probability of generating a sample containing overlapping events than the longer clipping policy, but in this sample, events can start at any time. The mixed policy uses both the random onset and longer clipping, making the generated sample very unpredictable. These policies are summarized in Table 1.

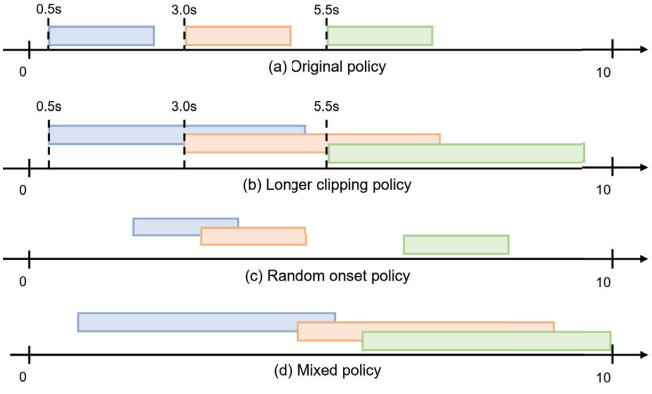


Figure 7. Comparison of temporal overlaps of different mixing polices. (**a**) Original policy with no overlap; (**b**) longer clipping policy allowing some overlaps between events by using longer clips; (**c**) random onset policy allowing events to start at any time; some samples overlap, but some other samples do not due to the randomness; (**d**) mixed policies in (**b**,**c**). Most overlaps are observed.

Table 1. Audio synthetic polices. All numbers are in seconds. The column names mean: onset times are the beginning of the events; max clip is the maximum clipping length; mean and std are the average and standard deviation of the clipping lengths. The row names: original is the policy suggested by the DCASE Challenge; longer clipping uses a longer maximum clipping length; random onset is varying the onset times randomly; mixed is the mixed policy of random onset and longer clipping.

	Onset Iimes	**Max Clip**	**Mean**	**Std**
original	0.5, 3.0, 5.5	2.0	1.7	0.51
longer clipping	0.5, 3.0, 5.5	5.0	3.14	1.67
random onset	uniformly random in $[0.5, 6.5)$	2.0	1.7	0.51
mixed	uniformly random in $[0.5, 6.5)$	5.0	3.13	1.67

4.2. Model Configurations

We compared the proposed U-Net model with the basic CNN and conventional U-Net in terms of SED performances. The basic building blocks of the models are listed in Tables 2 and 3. We generally stacked 3×3 convolutional layers ($Conv(3, K)$) and used 1×1 convolutions ($Conv(1, K)$) to resize the number of output maps if necessary. One deconvolutional layer, denoted by $DeConv22(3, K)$, has stride $(2, 2)$, and is used to uncompress the x- and y-axes by a factor of 2. Another deconvolutional layer $DeConv21(3, K)$ has stride $(2, 1)$ and uncompresses the x-axis only, so it doubles the output map size along the time axis, but not along the frequency axis. $DeConv21(3, K)$ was adopted in the proposed method only. In all the convolutional layer types, the number of output channels is given by a parameter K, and the number of inputs is determined automatically according to the previous output layers. At all the outputs of the convolutional and deconvolutional layers, we performed batch normalization and applied the rectified linear unit activation function (BN-ReLU) [34,35], as shown in Table 2. Two types of average pooling layers are used. In Table 3, $AvgPool(2, 2)$ uses a 2×2 pooling window with the same stride size in both the time and frequency axes, so the size of the output map is reduced to half of the original in both axes. $AvgPool(1, s)$ uses a $1 \times s$ pooling window with moving s samples along the time axis and 1 bin along the frequency axis, so the size of the output map decreases along the time axis only.

Table 2. Basic convolutional blocks used in SED model construction. There are two types of convolutional layers, $Conv(3, K)$ and $Conv(1, K)$, with 3×3 and 1×1 kernel sizes, respectively. There are also two types of deconvolutional layers. $DeConv22(3, K)$ and $DeConv21(3, K)$ have strides $(2, 2)$ and $(2, 1)$, respectively. BN-ReLU is batch normalization and rectified linear unit activation at the output.

Name	**Kernel Size**	**Strides**	**Output Channels**	**Post Processing**
$Conv(3, K)$	3×3	$(1, 1)$	K	BN-ReLU
$Conv(1, K)$	1×1	$(1, 1)$		
$DeConv22(3, K)$	3×3	$(2, 2)$		
$DeConv21(3, K)$	3×3	$(2, 1)$		

Table 3. Pooling blocks and dropout layer used in SED model construction. The layer $AvgPool(2, 2)$ reduces the sizes by half in both the x- and y-axes, but $AvgPool(1, f)$ reduces the y-axis by a factor of f, resizing the frequency axis, but not the time axis. $AvgPool(2, 2)$ is usually added after a convolutional layer, and $AvgPool(1, f)$ is used in concatenating the output maps of different sizes in U-Net.

Name	**Description**	**Input Size**	**Output Size**
$AvgPool(2, 2)$	2×2 average pooling, stride $(2, 2)$	(w, h, K)	$(\frac{w}{2}, \frac{h}{2}, K)$
$AvgPool(1, s)$	$1 \times s$ average pooling, stride $(1, s)$		$(w, \frac{h}{s}, K)$
$Dropout(p)$	dropout with probability p		

The baseline CNN for the SED task is configured as shown in Table 4. It is constructed by stacking 3×3 convolutional layers, gradually enlarging the number of output maps from 1 to 128. There is no pooling layer between the convolutional layers, so the output map sizes are all the same as the input sizes. The last layer is a 1×1 convolutional layer and converts 128 output maps to the number of classes (C). The advantage of the baseline CNN is that it is very simple and the sound events are detected either in the time or frequency domain. However, if there is not enough training data, the model may underestimate. The classification targets are obtained by global weighted rank pooling (GWRP) on the individual feature maps, as explained in Section 2.4.

Table 4. Baseline CNN design. It is composed of 4 convolutional layers with kernel size 3×3, followed by a 1×1 convolutional layer. The output of the last layer is for sound event detection.

Name	Input Shape	Output Shape	Output Size
$Conv(3, 32)$	$(311, 64, 1)$	$(311, 64, 32)$	636,928
$Conv(3, 64)$	$(311, 64, 32)$	$(311, 64, 64)$	1,273,856
$Conv(3, 128)$	$(311, 64, 64)$	$(311, 64, 128)$	2,547,712
$Conv(3, 128)$	$(311, 64, 128)$	$(311, 64, 128)$	2,547,712
$Conv(1, C)$	$(311, 64, 128)$	$(311, 64, C)$	$19{,}904 \times C$
total output size			$7{,}006{,}208 + 19{,}904 \times C$

The detailed configuration of the conventional U-Net in Figure 3 is shown in Table 5 [26]. In encoding, there are 3 convolutional blocks with average pooling of size 2×2, so the feature map sizes are divided by 2 in both the x- and y-axes. Therefore, the original input size 312×64 is divided by $2^3 = 8$, resulting in feature maps of size 39×8. Another convolutional block without average pooling, but with dropout is added to the end of the encoder. The number of convolutional kernels is 16, 32, 64, and 128, so the final 3-dimensional output is of size $39 \times 8 \times 128$. The decoder basically reverses the encoding process. The *DeConv*22 layer applies convolution with doubling both the x- and y-axes, followed by the *Concat* layer with a skip connection to the corresponding encoder output, as shown in Figure 3. The final feature maps for C classes are obtained by the 1×1 convolutional layer, $Conv(1, C)$, and GWRP is applied.

The proposed U-Net with limited upsampling, denoted as LUU-Net, is similarly configured as the conventional U-Net. It also consists of four convolutional blocks, three deconvolutional blocks without residual connection [36], and one of 1×1 convolutional layer. Because the decoder of LUU-Net performs 2×1 upsampling instead of 2×2, an additional average pooling is employed to match the input size at the skip connection. The detailed configuration with the input and output shape is shown in Table 6. The *Concat* layer concatenates the output of the last *DeConv*22 block and *Conv* block, which has the same number of channels. The proposed method uses about 20% fewer parameters than the conventional U-Net. We trained the three models by applying GWRP to the baseline CNN, U-Net, and LUU-Net, respectively. The performances were evaluated by the prediction accuracies of the audio tagging (AT) and sound event detection (SED) tasks. We also trained the three models by applying MEX [37], AlphaMEX [6], and the proposed global threshold average pooling (GTAP) in Section 3.2.

Table 5. U-Net design for sound event detection. It is divided into the encoder and decoder. The encoder consists of 3 convolutional blocks with 2×2 average pooling, followed by a convolutional layer with dropout. The decoder is composed of 3 deconvolutional blocks with skip connections to the encoder feature maps, and the final 1×1 convolutional layer is for event classification.

	Name	Input Shape	Output Shape	Output Size
encoder	$Conv(3,16)$	$(312,64,1)$	$(312,64,16)$	
	$AvgPool(2,2)$	$(312,64,16)$	$(156,32,16)$	79,872
	$Conv(3,16)$	$(156,32,16)$	$(156,32,32)$	
	$AvgPool(2,2)$	$(156,32,32)$	$(78,16,32)$	39,936
	$Conv(3,64)$	$(78,16,32)$	$(78,16,64)$	
	$AvgPool(2,2)$	$(78,16,64)$	$(39,8,64)$	19,968
	$Conv(3,128)$	$(39,8,64)$	$(39,8,128)$	
	$Dropout(0.2)$	$(39,8,128)$	$(39,8,128)$	39,936
decoder	$DeConv22(3,64)$	$(39,8,128)$	$(78,16,64)$	
	$Concat$	$(78,16,64 \times 2)$	$(78,16,128)$	79,872
	$DeConv22(3,32)$	$(78,16,128)$	$(156,32,32)$	
	$Concat$	$(156,32,32 \times 2)$	$(156,32,64)$	159,744
	$DeConv22(3,16)$	$(156,32,64)$	$(312,64,16)$	
	$Concat$	$(312,64,16 \times 2)$	$(312,64,32)$	319,488
	$Conv(1,C)$	$(312,64,32)$	$(312,64,C)$	$19{,}968 \times C$
total output size				$738{,}816 + 19{,}968 \times C$

Table 6. The proposed LUU-Net (U-Net with limited upsampling) design for sound event detection. The encoder blocks are identical to U-Net, but the decoder used $DeConv21$, which upsamples along the time axis, but not along the frequency axis. Therefore, the vertical size does not change in the decoder, all 8. In the *Concat* layers, $AvgPool(1,s)$ is applied, where $s \in \{2,4,8\}$ to match the vertical lengths of the encoder and decoder outputs.

	Name	Input Shape	Output Shape	Output Size
encoder	$Conv(16)$	$(312,64,1)$	$(312,64,16)$	
	$AvgPool(2,2)$	$(312,64,16)$	$(156,32,16)$	79,872
	$Conv(16)$	$(156,32,16)$	$(156,32,32)$	
	$AvgPool(2,2)$	$(156,32,32)$	$(78,16,32)$	39,936
	$Conv(64)$	$(78,16,32)$	$(78,16,64)$	
	$AvgPool(2,2)$	$(78,16,64)$	$(39,8,64)$	19,968
	$Conv(128)$	$(39,8,64)$	$(39,8,128)$	
	$Dropout2D(0.2)$	$(39,8,128)$	$(39,8,128)$	39,936
decoder	$DeConv21(64)$	$(39,8,128)$	$(78,8,64)$	
	$Concat$ with $AvgPool(1,2)$	$(78,8,64 \times 2)$	$(78,8,128)$	39,936
	$DeConv21(32)$	$(78,8,128)$	$(156,8,32)$	
	$Concat$ with $AvgPool(1,4)$	$(156,8,32 \times 2)$	$(156,8,64)$	39,936
	$DeConv21(16)$	$(156,8,64)$	$(312,8,16)$	
	$Concat$ with $AvgPool(1,8)$	$(312,8,16 \times 2)$	$(312,8,32)$	39,936
	$Conv(1,C)$	$(312,8,32)$	$(312,8,C)$	$2{,}496 \times C$
total output size				$299{,}520 + 2{,}496 \times C$

4.3. Performance Evaluation Metrics

The output of binary classifiers is *true* or *false*, where *true* means that the corresponding event is active and *false* means being inactive. The predicted output is compared to the ground truth, and it is indicated as true positive (TP), true negative (TN), false positive

(FP), and false negative (FN) [38], as shown in Table 7. From the sample counts in the TP, TN, FP, and FN bins, we can compute 3 different performance indexes as follows:

$$precision = \frac{TP}{TP + FP} \tag{8}$$

$$recall = \frac{TP}{TP + FN} \tag{9}$$

where *precision* is the ratio of correctly indicated samples to all *true* predicted outputs and *recall* is the ratio of correct samples to all *true* ground truth labels. If *precision* is higher, the output indicates that *true* is more reliable. If *recall* is higher, the samples with *true* ground truth labels are less likely misclassified. Both *precision* and *recall* can be related to the accuracy of the predicted outputs, but in somewhat different manners. To obtain a balanced metric, the F1 score is computed by the harmonic mean of *precision* and *recall* [38]:

$$F1 = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} = \frac{2precision \cdot recall}{precision + recall} \tag{10}$$

The performance metrics, precision, recall, and F1 scores are computed differently for audio tagging and sound event detection. In audio tagging tasks, the event is active, meaning that the corresponding event is present in the input recording. There is no consideration where the event starts or ends. Therefore, the label is "*true*" if the generated sample of 10 s has the event. There are 41 audio event categories and 41 binary classifiers for those categories, and the individual performance metrics are computed by the following equation:

$$precision_{AT}(c) = \frac{TP_{AT}(c)}{TP_{AT}(c) + FP_{AT}(c)} \tag{11}$$

$$recall_{AT}(c) = \frac{TP_{AT}(c)}{TP_{AT}(c) + FN_{AT}(c)} \tag{12}$$

$$F1_{AT}(c) = \frac{2precision_{AT} \cdot recall_{AT}}{precision_{AT} + recall_{AT}} \tag{13}$$

where c is the class index and $TP_{AT}(c), FP_{AT}(c), FN_{AT}(c)$ are computed using the ground truth audio tagging labels and the predicted labels by classifier c. The computation is sample-based, for example

$$TP_{AT}(c) = \sum_{k=1}^{K(c)} I(GT(c,k) = prediction(c,k) = true) \tag{14}$$

$$FP_{AT}(c) = \sum_{k=1}^{K(c)} I(GT(c,k) = false \text{ and } prediction(c,k) = true) \tag{15}$$

$$FN_{AT}(c) = \sum_{k=1}^{K(c)} I(GT(c,k) = true \text{ and } prediction(c,k) = false) \tag{16}$$

where k is the generated sample index, $GT(c,k)$ and $prediction(c,k)$ are the true and predicted label for class c and sample k, and $I(\cdot)$ is an indicator function returning 1 if the given logical expression is true and 0 if false. $K(c)$ is the number of audio samples for class k, which is different for different classes ranging from 94 to 300, as shown in Section 4.1. The precision, recall, and F1 scores of 41 event classes were averaged to obtain a single, mean performance metric:

$$mPrc_{AT} = \frac{1}{C}\sum_{c=1}^{C} precision_{AT}(c) \quad (17)$$

$$mRcl_{AT} = \frac{1}{C}\sum_{c=1}^{C} recall_{AT}(c) \quad (18)$$

$$mF1_{AT} = \frac{1}{C}\sum_{c=1}^{C} F1_{AT}(c) \quad (19)$$

where $mPrc_{AT}/mRcl_{AT}/mF1_{AT}$ stand for "mean precision/recall/F1 of multiple event tagging", respectively. To compute the SED performance metrics, we adopted segment-based evaluation metrics [1]. The TP, FP, and FN of the SED outputs using segment-based evaluation are computed by:

$$TP_{SED}(c) = \sum_{k=1}^{K(c)}\sum_{n=1}^{N(k)} I(GT(c,k,n) = prediction(c,k,n) = true) \quad (20)$$

$$FP_{SED}(c) = \sum_{k=1}^{K(c)}\sum_{n=1}^{N(k)} I(GT(c,k,n) = false \text{ and } prediction(c,k,n) = true) \quad (21)$$

$$FN_{SED}(c) = \sum_{k=1}^{K(c)}\sum_{n=1}^{N(k)} I(GT(c,k,n) = true \text{ and } prediction(c,k,n) = false) \quad (22)$$

where n is the analysis frame index, $N(k)$ is the number of frames for generated sample k, and $GT(c,k,n)$ and $prediction(c,k,n)$ are the true and predicted label for class c, sample k, and frame n. The segmented-based performance metrics for SED, $mPrc_{SED}$, $mRcl_{SED}$, and $mF1_{SED}$, are obtained by substituting $TP_{AT}(c)$, $FP_{AT}(c)$, and $FN_{AT}(c)$ by $TP_{SED}(c)$, $FP_{SED}(c)$, and $FN_{SED}(c)$ in Equations (11)–(13) and (17)–(19).

Table 7. Classification of prediction results by being compared with ground truth labels. Ground truth labels are given, and predicted labels are the output of the binary classifiers. Symbols T, F, TP, FP, FN, and TN are true, false, true positive, false negative, and true negative, respectively.

		Ground Truth	
		T	**F**
predicted	T	TP	FP
	F	FN	TN

4.4. Original Synthetic Policy

The sound event detection experiments were carried out on the generated dataset according to the original synthetic policy [16]. As shown in Table 1, the maximum event length was set to be 2.0 s, and the mean and standard deviation of the event length were 1.7 and 0.51 s, respectively. To determine the value of the hyperparameter α in Equation (7), we performed a grid search. α was varied from -1.0 to 1.0 with 0.2 displacement, a total of 10 cases for the grid search. The result is shown in Table 8. There was 90% of the generated training dataset used to train the proposed LUU-Net with the pooling method GTAP, and the remaining 10% was used as a validation set. The classwise mean F1 scores of the audio tagging (AT) and sound event detection (SED) tasks were computed, and their average values were used to rank different α values. We selected $\alpha \in \{0.2, 0.0, -0.4\}$ according to the average of $mF1_{AT}$ and $mF1_{SED}$ and used them in the subsequent experiments.

Table 9 shows the AT and SED performances with various model configurations. The baseline line CNN does not have any downsampling layers. Therefore, there was no reduction in the output map sizes along the forward path, requiring huge convolutional operations. The number of iterations per unit second, at the last column, was 3.69 and

relatively small when compared to the other models. When GTAP was applied, the number of iterations per unit second was 4.54, meaning that a 23% faster training speed was obtained. However, both the AT and SED performances degraded greatly in terms of the mF1. Because there was no reduction in the output map sizes across the convolutional layers, as shown in Table 4, the sound event information was distributed in the final segmentation mask. According to Equations (4) and (5), the proposed GTAP does not use inactive or little active outputs, so it is not suited to CNNs. The second part shows the performance with the conventional U-Net. Significant improvements were gained in terms of computational overhead, as well as AT and SED performances by replacing the CNN with U-Net, with the help of the recursive shortcut paths from the previous layers. For both of the AT and SED tasks, the precision, recall, and F1 scores all increased by about 10%. F1 scores of 53.1 and 39.5 were obtained by the CNN and 62.9 and 49.2 by U-Net. The number of training steps per second was 8.97, which was 2.43-times faster than the CNN. Further improvements were obtained by GTAP. The number of steps per unit second was 13.11, 68% faster than GWRP. In the AT tasks, the mean F1 scores were 58.9 and 58.0 with $\text{GTAP}_{\alpha} = 0.2$ and $\text{GTAP}_{\alpha=0}$, which were lower than 62.9 with GWRP. However, $\text{GTAP}_{\alpha=-0.4}$ showed 68.0, which was the largest among all the U-Net results. In the SED tasks, $\text{GTAP}_{\alpha} = 0.2$ and $\text{GTAP}_{\alpha=0}$ were better than GWRP, but $\text{GTAP}_{\alpha=-0.4}$ was worse. With lower α, a smaller threshold is obtained by Equation (7), and more components in the segmentation map are used, so higher precision was obtained in the AT task. However, segment-based metrics were used in the SED task, and more false positive segments were included by the lower threshold in GTAP, resulting in very low precision (38.8).

Table 8. Grid search results on the dataset generated by the original policy. The model is the proposed LUU-Net. Classwise mean F1 scores of audio tagging ($mF1_{AT}$) and sound event detection ($mF1_{SED}$) tasks were computed, and their average was used to rank the hyperparameter α values. The top 3 were $\{0.2, 0.0, -0.4\}$.

α	$mF1_{AT}$	$mF1_{SED}$	Average	Rank
1.0	65.26	50.67	57.96	7
0.8	66.44	52.81	59.62	5
0.6	65.77	53.51	59.64	4
0.4	66.34	52.50	59.42	6
0.2	65.64	53.93	59.78	3
0.0	65.33	54.36	59.85	2
−0.2	64.27	50.35	57.31	8
−0.4	69.66	51.02	60.34	1
−0.6	65.72	47.03	56.37	9
−0.8	63.80	45.26	54.53	11
−1.0	64.07	45.38	54.72	10

The next 3 rows show the precision, recall, and F1 scores obtained by the proposed LUU-Net with 3 different types of global pooling methods. The LUU-Net with GWRP showed improved F1 scores of 64.1 and 50.7, when compared to 62.0 and 49.2 with the conventional U-Net. The number of steps per second was 28.99, 3.23-times faster than U-Net. We also compared the pooling method GWRP with AlphaMEX [6] and MEX [37]. The F1 score of the AT task was 64.0 with AlphaMEX, which was similar compared to the 64.1 with GWRP. However, the F1 score of the SED was 40.7 with AlphaMEX, which was much lower than the 50.7 with GWRP. The F1 scores with MEX were 64.1 and 51.5, which were the best among the conventional 3 pooling methods, GWRP, AlphaMEX, and MEX.

The final 3 rows combine the proposed LUU-Net with the proposed GTAP pooling. For hyperparameters $\alpha = \{0.2, 0, -0.4\}$, the F1 scores for the AT task were $(64.5, 64.4, 68.8)$, respectively. By setting $\alpha = -0.4$, the highest F1 score for the AT task was obtained. The F1 scores for the SED task were $(52.5, 53.1, 50.0)$, respectively. For $\alpha = -0.4$, the F1 score was much lower than the others. The highest SED score was obtained by setting $\alpha = 0$. However, $\alpha = 0.2$ gave a slightly better AT F1 score, so it is also a well-balanced

hyperparameter value. According to Equations (5) and (7), the smaller value of α produced a lower threshold, so more units from the segmentation map were chosen in the average pooling. Hence, it was more advantageous in finding a single audio event label, i.e., the audio tagging task. However, more units in the boundary region having weak activations were included in the average pooling, and the SED performance, therefore, degraded. When $\alpha = 0$, the cut-threshold became the simple average of the whole segmentation map and provided well-balanced performance in both the AT and SED task. There were no meaningful differences in the computation time by varying α, so the average number of steps per second is given in the rightmost column of Table 9. The number of steps per second of GTAP was higher than all the other methods due to the reduced number of segmentation units in the average pooling.

Table 9. Audio tagging (AT) and sound event detection (SED) results on the dataset generated by the original synthetic policy. The neural network models were CNN, U-Net, and the proposed LUU-Net, whose configurations are shown in Tables 4–6, respectively. Various pooling methods were applied to the output of the LUU-Net: AlphaMEX, MEX, GWRP, and the proposed global threshold average pooling (GTAP) explained in Section 3.2 with varying $\alpha \in \{0.2, 0.0, -0.4\}$. GTAP with the same α values was also applied to the CNN and U-Net to compare the performance variations with LUU-Net. For all the experiments, the mean precision (mPrc), mean recall (mRcl), mean F1 scores (mF1), and the number of steps per unit second were measured.

Model	**Pooling Method**	**AT Task**			**SED Task**			**#Step/s**
		mPrc	**mRcl**	**mF1**	**mPrc**	**mRcl**	**mF1**	
CNN	GWRP	47.1	70.7	53.1	40.2	45.1	39.5	3.69
	$GTAP_{\alpha=0.2}$	35.2	75.9	46.9	74.1	11.7	19.1	
	$GTAP_{\alpha=0}$	34.3	75.1	45.8	72.6	13.3	21.2	4.54
	$GTAP_{\alpha=-0.4}$	52.8	40.1	39.5	28.4	37.2	27.0	
U-Net	GWRP	53.0	80.9	62.9	40.2	66.7	49.2	8.97
	$GTAP_{\alpha=0.2}$	48.2	79.3	58.9	56.7	49.0	50.6	
	$GTAP_{\alpha=0}$	46.5	81.2	58.0	53.0	53.3	51.7	13.11
	$GTAP_{\alpha=-0.4}$	68.6	70.3	68.0	38.8	66.1	47.4	
LUU-Net	AlphaMEX	58.5	73.4	64.0	62.1	32.7	40.7	21.11
	MEX	56.7	76.6	64.1	50.1	55.6	51.5	32.25
	GWRP	56.8	77.4	64.1	45.9	60.0	50.7	28.99
	$GTAP_{\alpha=0.2}$	56.7	77.9	64.5	56.0	52.0	52.5	
	$GTAP_{\alpha=0}$	55.7	79.0	64.4	53.2	55.6	53.1	35.68
	$GTAP_{\alpha=-0.4}$	67.0	72.6	68.8	42.3	64.2	50.0	

Figure 8 shows the sound event detection results with various models and various global pooling methods. The models in (c–e) were trained by the CNN, U-Net, and LUU-Net, respectively, and GWRP was adopted. In (c), event labels predicted by the CNN, the first event was almost missing, and many falsely detected units were seen and scattered in the upper part of the figure. In (d), U-Net prediction, the first event was detected, and much fewer false prediction units were observed. In (e), by the proposed LUU-Net, the event labels were more clearly detected, and a very small amount of false predictions were observed. To show the differences of the global pooling methods, event prediction examples are shown in (f–h), whose models were configured with AlphaMEX, MEX, and the proposed GTAP, respectively. There was no noticeable difference between (e) GWRP and (g) MEX, but the first event was almost missing in (f) AlphaMEX. This explains the low F1 score of AlphaMEX in Table 9 for the SED task. In (h), the proposed GTAP with $\alpha = 0$, the detection results were much more distinctive than the others, and the false predictions almost disappeared.

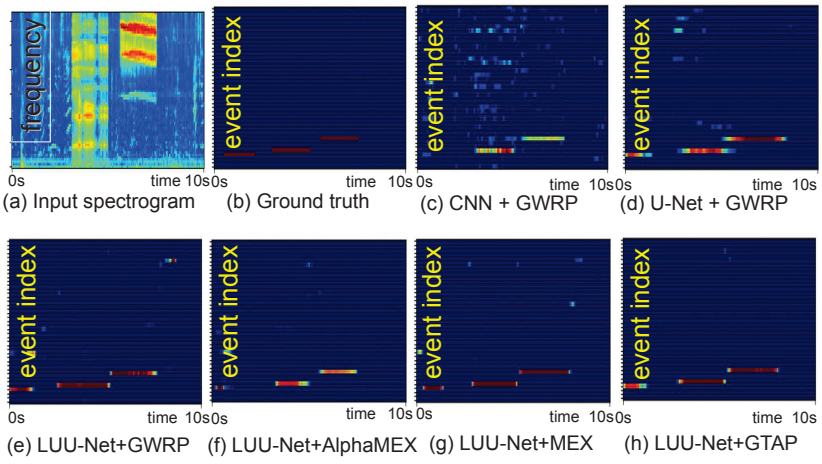


Figure 8. Sound event detection examples with various models and global pooling methods. The vertical axis in (**b**–**h**) represents the class number. (**a**) Spectrogram of the input mixture sample generated by the original synthetic policy. The x-axis is the time in seconds (10 s long), and the y-axis is the frequency (only $0 \sim 4$kHz are shown). (**b**) Ground truth labels. There are 3 distinctive sound events, represented by bright red lines. For (**b**–**h**), the x-axis is the time in seconds aligned with the x-axis of the spectrogram in (**a**), and the y-axis represents the event labels. (**c**–**h**) display sound event labels predicted by (**c**) CNN with GWRP, (**d**) U-Net with GWRP, (**e**) LUU-Net with GWRP, (**f**) LUU-Net with AlphaMEX, (**g**) LUU-Net with MEX, and (**h**) LUU-Net with GTAP$_{\alpha=0}$.

4.5. Longer Clipping Synthetic Policy

This section describes the experimental results on the dataset synthesized by the longer clipping policy given in Table 1. The mean and standard deviation of the event length included in this dataset were 3.14 and 1.67 s, respectively. The event length was limited to 5 s, so this dataset also included overlaps between events around 3 and 5.5 s. The audio tagging (AT) and sound event detection (SED) results are shown in Table 10. Similar to the original synthetic policy, U-Net showed much higher tagging and detection performances (63.7 and 47.4 F1 scores), as well as reduced computational overhead. Comparing U-Net and LUU-Net, with the same GWRP, LUU-Net showed 0.7 higher tagging (64.4) and 0.6 lower detection (46.8) F1 scores. With longer event lengths, the proposed LUU-Net was less advantageous, except in the computational efficiency. This is because the model parameters of U-Net were better trained with longer event lengths, so more stable performance than that of the original synthetic policy was obtained. The tagging F1 scores of GWRP, AlphaMEX, and MEX with LUU-Net were all similar, but AlphaMEX showed lower detection performance than the other pooling methods.

For the CNN, U-Net, and LUU-Net, we applied the proposed GTAP with $\alpha \in \{0.2, 0, -0.4\}$ in terms of the mF1. There were huge performance degradations in the CNN with GTAP, similar to the original synthetic policy. Especially in SED, much higher precision was obtained, but recall dropped drastically. Because the proposed GTAP cuts off the output map, many activations were lost, and therefore, the recall metrics dropped greatly. With U-Net, little performance drops were observed with GTAP. With the proposed LUU-Net, GTAP improved both the AT and SED performances. The tagging F1 score with GWRP was 64.4 and with GTAP with $\alpha \in \{0.2, 0, -0.4\}$, 64.4, 64.1, and 64.7, respectively, so the best F1 score was obtained with $\alpha = -0.4$, which is the same as the experimental results with the original synthetic policy. The detection F1 scores were 44.8, 46.6, and 48.0, respectively, and the best was also with $\alpha = -0.4$. Interestingly, it was not as good as the other α values in the original policy. However, $\alpha = -0.4$ was best in both the tagging and detection tasks. This implies that longer event lengths should provide more obvious unit labels and improve both tagging and detection performances.

Table 10. AT and SED results on the dataset generated by the longer clipping policy. The CNN, U-Net, and proposed LUU-Net are shown in Tables 4–6, respectively. Pooling methods AlphaMEX, MEX, GWRP, and the proposed GTAP with varying α values.

Model	Pooling Method	AT Task			SED Task			#Step/s
		mPrc	mRcl	mF1	mPrc	mRcl	mF1	
CNN	GWRP	48.0	71.1	54.2	46.1	34.4	36.7	3.57
	$GTAP_{\alpha=0.2}$	35.7	74.0	46.8	77.2	7.2	12.5	
	$GTAP_{\alpha=0}$	34.4	77.7	46.6	76.3	9.2	15.7	4.37
	$GTAP_{\alpha=-0.4}$	52.4	43.1	42.5	36.7	34.4	31.3	
U-Net	GWRP	54.2	80.8	63.7	48.0	50.1	47.4	8.64
	$GTAP_{\alpha=0.2}$	50.2	78.2	59.7	62.5	34.6	42.8	
	$GTAP_{\alpha=0}$	48.2	80.4	59.2	60.4	37.5	44.4	12.54
	$GTAP_{\alpha=-0.4}$	67.6	69.0	66.9	47.2	51.4	47.3	
LUU-Net	AlphaMEX	60.0	73.7	64.9	67.6	25.9	35.8	21.13
	MEX	57.0	77.0	64.3	56.6	40.0	45.4	32.33
	GWRP	56.8	78.0	64.4	53.0	44.6	46.8	28.38
	$GTAP_{\alpha=0.2}$	57.2	76.8	64.4	62.3	36.9	44.8	
	$GTAP_{\alpha=0}$	55.8	77.8	64.1	60.1	40.1	46.6	35.12
	$GTAP_{\alpha=-0.4}$	66.0	71.2	67.4	50.5	48.6	48.0	

Figure 9 shows the sound event detection examples for the mixtures generated by the longer clipping policy. Various models and various global pooling methods were applied. The U-Net (d) and LUU-Net (e) results were better than those of CNN (c), but there were still prediction errors, as well as false detections. The proposed GTAP with $\alpha = -0.4$ (h) showed the best prediction performance. Almost no false predictions were observed, and the overlap between the second and the third events was also detected.

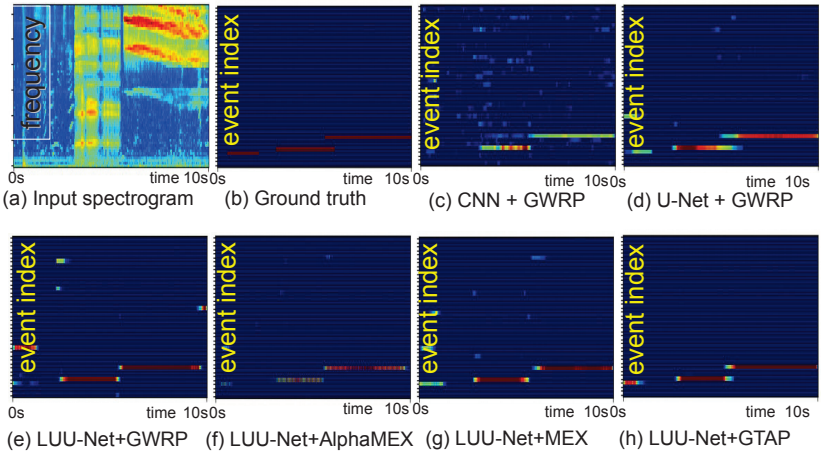


Figure 9. Sound event detection examples with various models and global pooling methods. The vertical axis in (**b**–**h**) represents the class number. (**a**) Spectrogram of the input mixture sample generated by the longer clipping policy. (**b**) Ground truth labels. (**c**–**h**) display sound event labels predicted by (**c**) CNN with GWRP, (**d**) U-Net with GWRP, LUU-Net with (**e**) GWRP, (**f**) AlphaMEX, (**g**) MEX, and (**h**) $GTAP_{\alpha=-0.4}$. The x-axis is the time in seconds, and the y-axis represents the frequency (**a**) or event labels (**b**–**h**). Sound event detection examples with CNN, U-Net, and LUU-Net with GWRP.

4.6. Random Onset Synthetic Policy

Table 11 shows the audio tagging and sound event detection results of the dataset synthesized by the random onset policy in Table 1. The maximum event length was set to be 2.0 s, and the mean and standard deviation of the event length were 1.7 and 0.51 s, which were the same as the original synthetic policy. In the original policy, the event onset and offset times were configured so that there should not be any overlap. However, the random onset policy does not have such requirements, so there were significantly many overlaps among the events. In Table 11, U-Net with GWRP showed 48.1 tagging and 35.6 detection F1 scores, which were much lower than the 62.9 and 49.2 for the original policy. By using the proposed LUU-Net, the F1 scores were 49.8 and 37.1, which were 1.7 and 1.5 higher than those of U-Net. U-Net showed good recall scores in both the tagging and detection tasks, but other measures were lower than LUU-Net. The number of steps increased from 8.67 to 27.26, so LUU-Net was 3.14-times faster than U-Net. GWRP showed a good F1 score, which was similar to MEX, but the iteration speed was about 30% slower because of sorting.

The last 3 rows use the proposed GTAP with $\alpha \in \{0.2, 0, -0.4\}$. The tagging F1 scores were 50.5, 50.0, and 52.1, respectively. The best and the second-best F1 scores were obtained with $\alpha = -0.4$ and 0.2, respectively, which were the same as the original policy. However, the F1 score difference was much less: original policy 4.3 and random onset 1.6. The detection F1 scores were 40.8, 40.0, and 34.0, respectively. The sum of the F1 scores of the tagging and detection tasks was 91.3 with $\alpha = 0.2$ and 86.1 with $\alpha = -0.4$. Therefore, the value that showed a higher sum of F1 scores and well-balanced performances in the tagging and detection tasks was $\alpha = 0.2$.

Table 11. AT and SED results on the dataset generated by the random onset policy. CNN, U-Net, and the proposed LUU-Net are shown in Tables 4–6, respectively. Pooling methods AlphaMEX, MEX, GWRP, and the proposed GTAP with varying α values.

Model	Pooling Method	AT Task			SED Task			#Step/s
		mPrc	mRcl	mF1	mPrc	mRcl	mF1	
CNN	GWRP	36.9	60.9	42.6	31.0	37.0	30.8	3.66
	$GTAP_{\alpha=0.2}$	30.6	66.1	40.4	61.7	13.5	20.7	4.54
	$GTAP_{\alpha=0}$	28.4	68.4	38.7	54.9	17.0	24.2	
	$GTAP_{\alpha=-0.4}$	39.1	33.9	32.0	19.8	30.1	20.2	
U-Net	GWRP	39.4	66.7	48.1	29.2	49.2	35.6	8.67
	$GTAP_{\alpha=0.2}$	38.0	65.7	46.8	48.1	36.1	39.6	12.97
	$GTAP_{\alpha=0}$	35.5	66.6	45.1	43.4	38.5	39.6	
	$GTAP_{\alpha=-0.4}$	53.7	50.8	50.7	26.7	45.8	32.5	
LUU-Net	AlphaMEX	43.9	60.7	49.4	51.2	19.5	26.7	21.13
	MEX	40.8	64.2	48.6	35.6	41.9	37.2	32.33
	GWRP	42.9	64.5	49.8	34.3	43.7	37.1	27.26
	$GTAP_{\alpha=0.2}$	43.4	64.2	50.5	46.9	37.7	40.8	33.64
	$GTAP_{\alpha=0}$	42.3	65.0	50.0	42.3	39.7	40.0	
	$GTAP_{\alpha=-0.4}$	51.3	55.7	52.1	28.4	45.6	34.0	

Figure 10 shows the sound event detection examples for the mixtures generated by the random onset synthetic policy. Various models and various global pooling methods were applied. In the CNN detection example in (c), there were many false detections scattered in the segmentation map. As shown in (d,e), U-Net and LUU-Net provided relatively clean detection results, but there were still many false detections. In (f,g), using AlphaMEX and MEX, it can be seen that the false detections mostly disappeared. In (h), using the proposed GTAP with $\alpha = 0.2$, it provided very clean detection results.

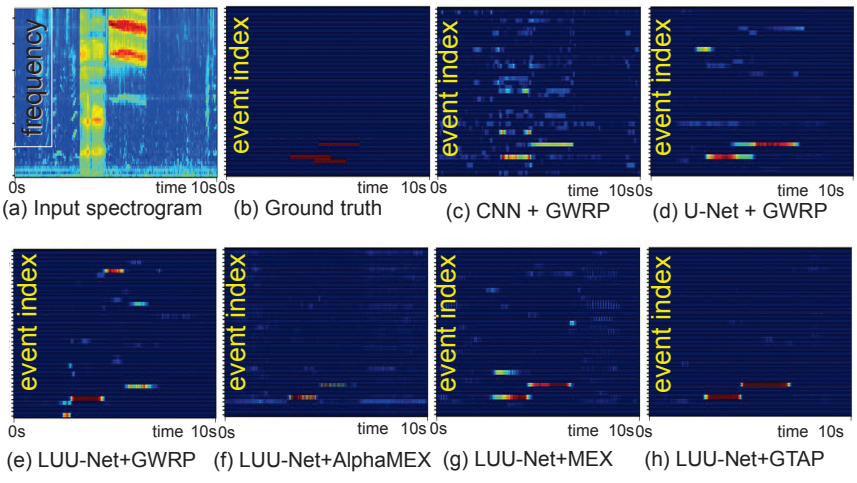


Figure 10. Sound event detection examples with various models and global pooling methods. The vertical axis in (**b**–**h**) represents the class number. (**a**) Spectrogram of the input mixture sample generated by the random onset policy. (**b**) Ground truth labels. (**c**–**h**) display sound event labels predicted by (**c**) CNN with GWRP, (**d**) U-Net with GWRP, LUU-Net with (**e**) GWRP, (**f**) AlphaMEX, (**g**) MEX, and (**h**) $\text{GTAP}_{\alpha=0.2}$. The x-axis is the time in seconds, and the y-axis represents the frequency (**a**) or event labels (**b**–**h**).

4.7. Mixed Synthetic Policy

Table 12 shows the audio tagging and sound event detection results of the dataset synthesized by the mixed policy in Table 1. The maximum event length was set to be 5.0 s, and the mean and standard deviation of the event length were 3.13 and 1.67 s, respectively. The mean length was 0.01 seconds shorter than the longer policy because there were more chances of sound events being cut off. This was the most-difficult dataset, so the F1 scores were overall much lower than the other synthetic policies. First, in the first 3 rows, we compare the CNN, U-Net, and the proposed LUU-Net with the GWRP method. Both U-Net and LUU-Net were much better than the CNN in both the tagging and detection F1 scores. LUU-Net was slightly better than U-Net, with a 3.27-times faster training speed. Comparing GWRP, AlphaMEX, and MEX, they showed similar performances in tagging, but AlphaMEX was the worst at detection. Lastly, we compared different values of the hyperparameter α with the proposed GTAP method. In tagging, $\alpha = -0.4$ showed a 2.8 higher F1 than $\alpha = 0$, and $\alpha = 0$ showed a 0.8 higher F1 than $\alpha = -0.4$. The sum of F1 scores was $(83.3, 81.3)$ with $\alpha = (-0.4, 0)$, so -0.4 was the best value for the mixed synthetic policy. The number of steps increased from the 8.64 of U-Net to the 34.92 of LUU-Net with GTAP, so the proposed method was about 4-times faster.

Figure 11 shows the sound event detection examples for the mixtures generated by the mixed synthetic policy. Various models and various global pooling methods were applied. In the CNN detection example in (c), there were many false detections scattered in the segmentation map. As shown in (d,e), U-Net and LUU-Net provided relatively clean detection results, but there were still many false detections. In (f,g), using AlphaMEX and MEX, it can be seen that the false detections mostly disappeared. In (h), using the proposed GTAP with $\alpha = -0.4$, it provided very clean detection results.

Table 12. AT and SED results on the dataset generated by the mixed policy. CNN, U-Net, and the proposed LUU-Net are shown in Tables 4–6, respectively. Pooling methods AlphaMEX, MEX, GWRP, and the proposed GTAP with varying α values.

Model	Pooling Method	AT Task			SED Task			#Step/s
		mPrc	mRcl	mF1	mPrc	mRcl	mF1	
CNN	GWRP	32.2	57.4	38.7	32.4	27.8	27.9	3.59
	$GTAP_{\alpha=0.2}$	27.4	62.3	36.7	59.8	9.0	14.8	
	$GTAP_{\alpha=0}$	26.6	62.9	36.1	62.6	10.4	16.7	4.42
	$GTAP_{\alpha=-0.4}$	41.6	33.8	32.3	30.0	28.3	24.9	
U-Net	GWRP	36.0	62.3	44.2	33.8	37.3	34.1	8.64
	$GTAP_{\alpha=0.2}$	34.4	61.1	42.9	54.2	25.8	33.5	
	$GTAP_{\alpha=0}$	33.5	62.0	42.5	50.9	28.2	34.8	12.51
	$GTAP_{\alpha=-0.4}$	50.6	50.0	48.8	33.8	40.1	34.8	
LUU-Net	AlphaMEX	39.0	57.9	45.4	51.4	15.8	22.7	21.05
	MEX	36.8	60.7	44.3	37.8	32.4	33.1	32.35
	GWRP	38.5	60.1	45.5	37.8	33.8	34.3	28.21
	$GTAP_{\alpha=0.2}$	37.7	60.4	45.4	50.4	27.3	34.1	
	$GTAP_{\alpha=0}$	38.7	60.9	46.0	47.4	29.9	35.3	34.92
	$GTAP_{\alpha=-0.4}$	47.3	53.4	48.8	34.4	38.4	34.5	

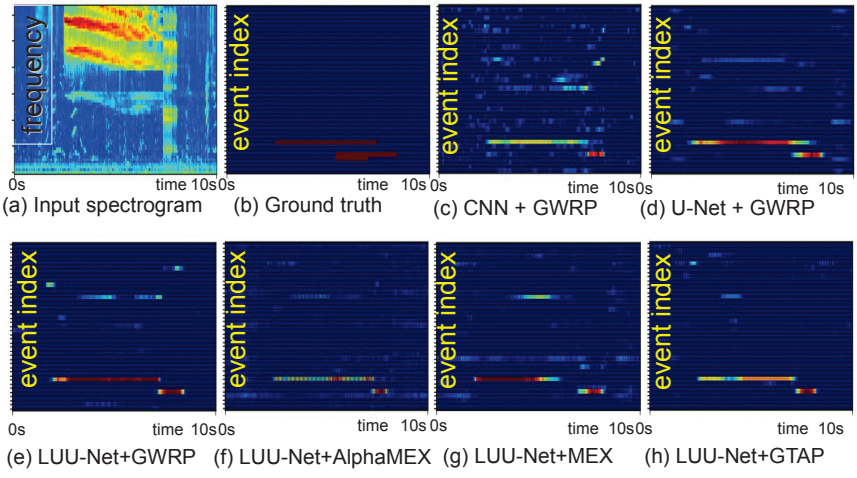


Figure 11. Sound event detection examples with various models and global pooling methods. The vertical axis in (**b**–**h**) represents the class number. (**a**) Spectrogram of the input mixture sample generated by the mixed synthetic policy. (**b**) Ground truth labels. (**c**–**h**) display sound event labels predicted by (**c**) CNN with GWRP, (**d**) U-Net with GWRP, LUU-Net with (**e**) GWRP, (**f**) AlphaMEX, (**g**) MEX, and (**h**) $GTAP_{\alpha=-0.4}$. The x-axis is the time in seconds, and the y-axis represents the frequency (**a**) or event labels (**b**–**h**).

4.8. Summary of Experimental Results

As shown in Tables 9–12, the proposed methods improved the audio tagging and sound event detection performances in most of the cases. Comparing U-Net and the proposed LUU-Net with the same GWRP, LUU-Net improved the tagging F1 score by up to 1.7% and the detection score by up to 1.5%. The training became more than three-times faster. In summary, the proposed LUU-Net slightly improved the tagging and detection performances with much faster model learning. Looking into the detailed precision and

recall scores, an interesting property was observed. In all four synthetic cases (original, longer clipping, random onset, and mixed), U-Net with GWRP usually showed relatively low precision scores and high recall scores when compared to those of LUU-Net with the same global pooling. Because the sound event onset timings were the same for all the frequency units, the segmentation targets should be varied in time only. The limited upsampling in the LUU-Net decoder provided blocked averaging in the frequency axis, resulting in low frequency resolution. A lower resolution is less likely to be affected by outliers, and LUU-Net did not change the time resolution, so there was no difference in the detection targets. This explains the high precision scores of LUU-Net. On the contrary, lower resolution is less effective in the precise exclusion of false units, so the recall scores of LUU-Net were lower than U-Net. This property appeared in all of the global pooling methods with LUU-Net. If the given application requires higher precision than recall, i.e., higher true detection rate, LUU-Net is preferred.

The proposed GTAP also improved the tagging F1 score by up to 4.6 and the detection F1 score by up to 5.2 with appropriate selections of the hyperparameter α. In audio tagging tasks where the outputs were directly used in computing the training loss, its value was more effective than the sound event detection tasks. Generally, with small α values, for example if we compare the results of $\alpha = -0.4$ with those of $\alpha = 0$ in our experiments, relatively high precision and low recall scores of the audio tagging tasks were observed. According to Equation (7), the threshold θ becomes smaller as α becomes smaller, more units are chosen to be averaged in Equation (5), and the average is used for computing the classification loss. Therefore, about 10% higher precision scores were obtained for all four synthetic datasets, as shown in Tables 9–12. However, the recall scores were about 7–10% lower, resulting in 2.1–4.6% higher F1 scores because of the higher false positive rates. For $\alpha = 0.2$, it also showed higher precision and lower recall scores than those of $\alpha = 0$, but the differences were not large enough to make a general statement. In the sound event detection tasks, opposite results were observed. Comparing the results of $\alpha = -0.4$ with those of $\alpha = 0$, relatively low precision and high recall scores were observed. Precisely speaking, when compared to $\alpha = 0$, there were 10.4–23.9% lower precision and 5.9–8.6% higher recall scores, resulting in up to 6.0% lower F1 scores. When $\alpha = 0.2$, there were 2.2–4.6% higher precision and 2.0–3.6% lower recall scores, and the F1 scores were up to 1.8% higher. This can be explained by the fact that the prediction of segmentation masks was not directly used for the computation of the target loss function. The learning process did not directly improve the prediction accuracy of the segmentation, which was not tightly related to the combination of the precision and recall scores, but the individual scores, so it can be biased to either the recall or prediction. The combined F1 scores were best with $\alpha = 0$, except the random onset dataset, because it provided a well-balanced prediction of the segmentation masks in a weakly supervised manner.

5. Discussion

We analyzed the experimental results to show the detailed contributions of the proposed LUU-Net and GTAP. The experimental results in Tables 9–12 are drawn as graphical charts for better visualization and analysis.

5.1. Execution Time Comparison

Figure 12 visualizes the differences of the number of steps per unit second. Comparing the CNN and LUU-Net, the training time of LUU-Net was about eight-times faster in most cases. Comparing U-Net and LUU-Net, it was about three-times faster. Because LUU-Net reduces the output map sizes by $\frac{1}{2}$, $\frac{1}{4}$, and $\frac{1}{8}$ in the decoder part, the computation time was reduced drastically. There were no notable differences among the synthetic policies. Comparing GWRP and GTAP, GTAP was about 1.2-times faster with the CNN and LUU-Net and 1.4-times faster with U-Net. Because about half of the output map components were discarded at the final output layer with $\text{GTAP}_{\alpha=0}$, this improvement in time with GTAP is reasonable. Combining GTAP with LUU-Net, a 4-times improvement in the

execution time over GWRP with U-Net and up to 10-times over GWRP with the CNN were expected.

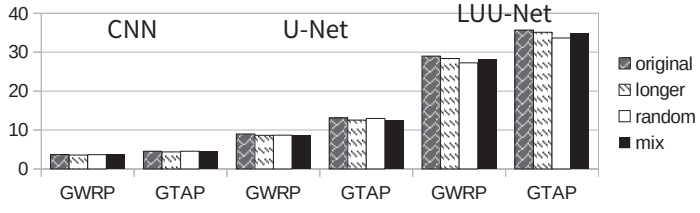


Figure 12. Comparison of the number of steps per unit second. The first two sets of bars represent the CNN with GWRP and GTAP, with 4 bars measured from the original, longer clips, random onsets, and mixed synthetic policies. The second two sets of bars represent U-Net, and the third two sets of bars are for LUU-Net.

5.2. Audio Tagging Performance Comparison

To show the performance variations with the change of the models and pooling methods under different conditions, the mean F1 scores from Tables 9–12 are drawn as two-dimensional charts. Figure 13 shows the average F1 scores for audio tagging tasks with the original, longer clipping, random onsets, and mixed synthetic policies. In all of the policies, the performances of GWRP and $\text{GTAP}_{\alpha=-0.4}$ on the U-Net and LUU-Net were almost the same, but there was a 3–5% drop with $\text{GTAP}_{\alpha=0.2}$ and $\text{GTAP}_{\alpha=0}$ on U-Net. Those performance drops were not observed for LUU-Net, so relative improvements were obtained with $\text{GTAP}_{\alpha=0.2}$ and $\text{GTAP}_{\alpha=0}$. For the CNN, there was too much degradation with GTAP compared to GWRP. The baseline CNN design in Table 4 does not have any bottleneck layer, which compresses the information from the input, so the output map pruning in GTAP resulted in information loss. Among the GTAP methods, $\alpha = -0.4$ was the best with both U-Net and LUU-Net. In most of the cases, the proposed LUU-Net was superior to the conventional U-Net, and the proposed GTAP was slightly better than the conventional GWRP.

5.3. Sound Event Detection Performance Comparison

A similar chart is drawn with the SED tasks in Figure 14. In all cases, except GWRP and the longer clipping policy, LUU-Net outperformed U-Net and the CNN. For the original and random policies, where the audio clip lengths were less than 2 s, GTAP with $\alpha \in \{0.2, 0.0\}$ was better than $\alpha = -0.4$ and GWRP. For the longer and mixed policies with longer audio clips lengths, less than 5 s, GTAP with $\alpha \in \{0.2, 0\}$ was not as good as GWRP and GTAP with $\alpha = -0.4$. Longer clip lengths require larger amounts of activation, so a smaller value of α is more suited to the longer and mixed policies.

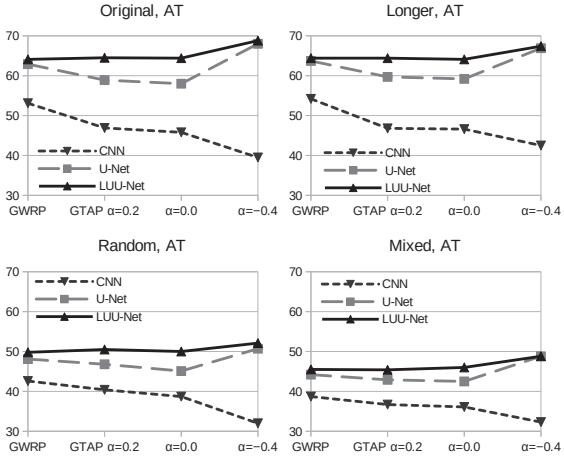


Figure 13. Audio tagging (AT) performance comparison with various deep learning models, pooling methods, and training data generation policies. Average F1 scores drawn; the x-axis is pooling methods; lines are CNN, U-Net, and LUU-Net. The individual charts are the results with the original, longer clipping, random onsets, and mixed synthetic policies.

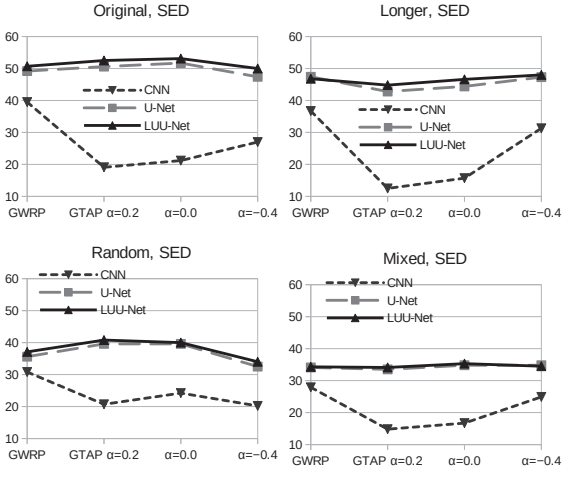


Figure 14. Sound event detection (SED) performance comparison with various deep learning models, pooling methods, and training data generation policies by average F1 scores.

5.4. Further Analysis of LUU-Net Results

Figures 13 and 14 show that the proposed LUU-Net and GTAP improved both the audio tagging and sound event detection performances over the conventional U-Net and GWRP. However, it is difficult to choose an optimal value for the hyperparameter α, so that it works best with all the experimental conditions. Therefore, we made a detailed analysis of the experimental results of LUU-Net with GTAP to find a relationship of the hyperparameter α and audio mixing conditions.

Figure 15 shows the AT and SED results of LUU-Net only. All the performance metrics, precision, recall, and F1 scores are drawn. For the mean precision, AT, $\alpha = -0.4$ was the best among the three α values for all mixing conditions. For the mean recall, AT, $\alpha = 0$ was the best and $\alpha = -0.4$ the worst. Combining the precision and recall, the F1 scores of $\alpha = -0.4$ were overall the best. In the SED tasks, the opposite observation was made. $\alpha = -0.4$ is the worst in the precision scores and was the best in recall scores. The combined F1 scores

were almost the same for all three α values. If $\alpha = -0.4$ in Equation (7), more output map values were chosen to compute the class label predictions for the whole clip in the AT tasks. Having more output map components, the class activation information was kept more in the pooling, so higher precision was obtained. For the SED tasks, the performance metrics were computed as segment-based, so more output map components would lead to higher false positive rates. Therefore, lower precision is obtained by Equation (8). The opposite explanation can be applied to the lower recall scores on the AT tasks with $\alpha = -0.4$ and the higher recall scores on the SED tasks. In Equation (9), reducing false negatives is directly related to higher recall scores. A larger amount of output maps in GTAP with $\alpha = -0.4$ would lead to less inclusion of inactive outputs, so higher recall scores were observed in the SED tasks computed in a segment-based manner.

In summary, the proposed LUU-Net with GTAP was better than the conventional U-Net with GWRP in most cases. The proposed GTAP can be configured to provide versatile choices of target applications by varying a single hyperparameter α. A smaller threshold by a negative α value is suggested when higher audio tagging performance is required, i.e., the identities of the audio sources are more important. When a higher recall rate in tagging or better sound event detection performance is required, choosing $\alpha = 0$ is suggested.

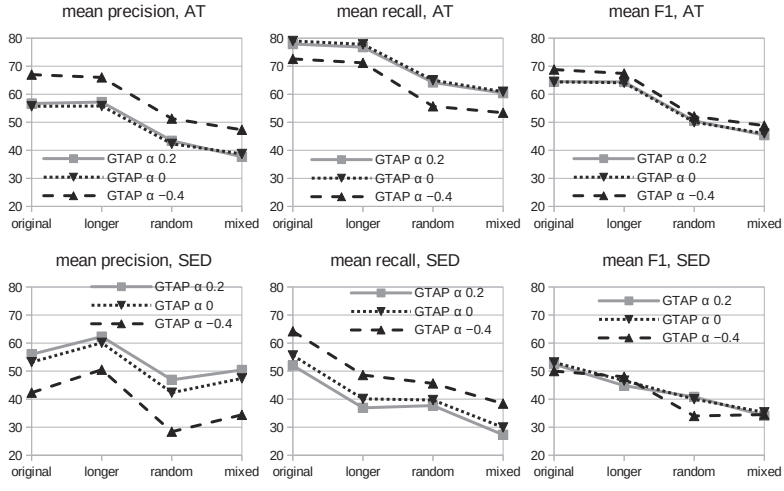


Figure 15. Illustrations of audio tagging and sound event detection performances by varying audio mixing conditions. The subfigures on the left are graphical charts of average precision scores of GTAP with $\alpha \in \{0.2, 0.0, -0.4\}$. The x-axis is the audio synthetic policies. The upper chart is audio tagging performances, and the lower one is sound event detection performances. The subfigures at the center are graphical charts of the average recall scores, and those on the right are the average F1 scores.

5.5. Code Availability

All the source codes for training the models and performing the experiments are publicly available at https://github.com/lsw0767/SED (accessed on 29 May 2023). The source dataset was taken from the *DCASE 2018 Challenge*, which is accessible at https://dcase.community/challenge2018 (accessed on 18 February 2023).

6. Conclusions

In this paper, we proposed two methods to improve the performance of weakly supervised sound event detection. The first method was a modification of the conventional U-Net to perform limited upscaling (LUU-Net). Assuming that the upscaling function along the frequency axis of the U-Net is not necessary in sound event detection, upsampling gradually to the original size was applied to the time domain only. The second method was global threshold average pooling (GTAP), which replaced the conventional global

weighted rank pooling (GWRP). GWRP showed higher detection performance compared to global average pooling (GAP) and global max pooling (GMP) [16]. However, GWRP requires output map sorting in every pooling step. The proposed GTAP eliminates the sorting operation and replaces it with simple thresholding. To find the threshold, the mean and the standard deviation of the output feature map are necessary, which are much more computationally efficient than sorting. GTAP performs the global pooling by calculating the average of only the value above a certain threshold by using the characteristic that the rank-based weight of GWRP almost ignores smaller values. There was a significant improvement in training speed, with small or huge performance improvements depending on the mixing conditions. According to the experimental results, the proposed LUU-Net with GTAP greatly outperformed the CNN and U-Net in various mixing datasets. The advantage of applying the proposed LUU-Net is the improved computation time. The amount of model parameters was about 40% of the conventional U-Net, and the measured training time was 3-times faster than U-Net and 8-times faster than the baseline CNN. With the proposed GTAP, an additional 1.2-times faster speed was obtained. In terms of audio tagging and sound event detection performance, the proposed LUU-Net outperformed the U-Net and the CNN in almost all cases. Another advantage of the proposed GTAP is that, by varying a single hyperparameter, it can be adapted to various target applications with different requirements. As a conclusion, the major contribution of the proposed LUU-Net and GTAP is the reduction of the computation time without any performance loss. Future work includes the automatic adaptation of the hyperparameter α for real applications.

Author Contributions: Conceptualization, S.L., H.K. and G.-J.J.; methodology, S.L.; software, S.L.; validation, S.L. and G.-J.J.; formal analysis, S.L. and H.K.; investigation, H.K.; resources, G.-J.J.; data curation, S.L.; writing–original draft preparation, S.L.; writing–review and editing, H.K. and G.-J.J.; visualization, S.L.; supervision, G.-J.J.; project administration, G.-J.J.; funding acquisition, G.-J.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency grant funded by the Ministry of Culture, Sports and Tourism in 2022 (project name: Development of high-speed music search technology using deep learning; project number: CR202104004).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original audio data are available at https://dcase.community/challenge2018 (accessed on 18 February 2023). Newly created data such as audio labels and Source codes are available at https://github.com/lsw0767/SED (accessed on 29 May 2023).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

SED	Sound event detection
CNN	Convolutional neural network
RNN	Recurrent neural network
VGG	Visual geometry group
GAP	Global average pooling
GMP	Global max pooling
GWRP	Global weighted rank pooling
GTAP	Global threshold average pooling

References

1. Mesaros, A.; Heittola, T.; Virtanen, T.; Plumbley, M.D. Sound event detection: A tutorial. *IEEE Signal Process. Mag.* **2021**, *38*, 67–83. [CrossRef]
2. Parascandolo, G.; Huttunen, H.; Virtanen, T. Recurrent neural networks for polyphonic sound event detection in real life recordings. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; pp. 6440–6444.
3. Sehgal, A.; Kehtarnavaz, N. A convolutional neural network smartphone app for real-time voice activity detection. *IEEE Access* **2018**, *6*, 9017–9026. [CrossRef] [PubMed]
4. Sainath, T.; Parada, C. *Convolutional Neural Networks for Small-Footprint Keyword Spotting*; Google, Inc.: New York, NY, USA , 2015; pp. 1478–1482.
5. Takahashi, N.; Gygli, M.; Pfister, B.; Van Gool, L. Deep convolutional neural networks and data augmentation for acoustic event detection. *arXiv* **2016**, arXiv:1604.07160 .
6. Zhang, B.; Zhao, Q.; Feng, W.; Lyu, S. AlphaMEX: A smarter global pooling method for convolutional neural networks. *Neurocomputing* **2018**, *321*, 36–48. [CrossRef]
7. Tarvainen, A.; Valpola, H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Proc. Adv. Neural Inf. Process. Syst.* **2017**, *30* , 1195–1204.
8. Lu, R.; Duan, Z. Bidirectional GRU for sound event detection. In Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2017), Munich, Germany, 16 November 2017.
9. JiaKai, L. Mean Teacher Convolution System for DCASE 2018 Task 4. In Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018), Surrey, UK, 19–20 November 2018.
10. Mesaros, A.; Heittola, T.; Virtanen, T. TUT database for acoustic scene classification and sound event detection. In Proceedings of the European Signal Processing Conference (EUSIPCO), Budapest, Hungary, 29 August–2 September 2016; pp. 1128–1132.
11. Kumar, A.; Raj, B. Audio Event Detection using Weakly Labeled Data. *arXiv* **2016**, arXiv:1605.02401 .
12. Turpault, N.; Serizel, R.; Parag Shah, A.; Salamon, J. Sound event detection in domestic environments with weakly labeled data and soundscape synthesis. In *Workshop on Detection and Classification of Acoustic Scenes and Events*; HAL: Bengaluru, India , 2019
13. Salamon, J.; MacConnell, D.; Cartwright, M.; Li, P.; Bello, J.P. Scaper: A Library for Soundscape Synthesis and Augmentation. In Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), New Paltz, NY, USA, 15–18 October 2017.
14. McFee, B.; Salamon, J.; Bello, J.P. Adaptive pooling operators for weakly labeled sound event detection. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2018**, *26*, 2180–2193. [CrossRef]
15. Pankajakshan, A.; Bear, H.L.; Benetos, E. Polyphonic Sound Event and Sound Activity Detection: A Multi-task approach. In Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), New Paltz, NY, USA, 20–23 October 2019 .
16. Kong, Q.; Xu, Y.; Sobieraj, I.; Wang, W.; Plumbley, M.D. Sound Event Detection and Time–Frequency Segmentation from Weakly Labelled Data. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2019**, *27*, 777–787. [CrossRef]
17. Pandeya, Y.R.; Bhattarai, B.; Lee, J. Visual Object Detector for Cow Sound Event Detection. *IEEE Access* **2020**, *8*, 162625–162633. [CrossRef]
18. Kolesnikov, A.; Lampert, C.H. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 695–711.
19. Cai, Z.; Vasconcelos, N. Cascade R-CNN: Delving into high quality object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 6154–6162.
20. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
21. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; Torralba, A. Learning deep features for discriminative localization. *arXiv* **2016**, arXiv:1512.04150 .
22. Dinkel, H.; Wu, M.; Yu, K. Towards duration robust weakly supervised sound event detection. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2021**, *29*, 887–900. [CrossRef]
23. Miyazaki, K.; Komatsu, T.; Hayashi, T.; Watanabe, S.; Toda, T.; Takeda, K. Weakly-supervised sound event detection with self-attention. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 66–70.
24. Deshmukh, S.; Raj, B.; Singh, R. *Improving Weakly Supervised Sound Event Detection with Self-Supervised Auxiliary Tasks*; Google, Inc.: New York, NY, USA, 2021; pp. 596–600.
25. Park, C.; Kim, D.; Ko, H. Sound Event Detection by Pseudo-Labeling in Weakly Labeled Dataset. *Sensors* **2021**, *21*, 8375. [CrossRef] [PubMed]
26. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; Springer: Berlin/Heidelberg, Germany, 2015; pp. 234–241.

27. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 834–848. [CrossRef] [PubMed]
28. Lin, M.; Chen, Q.; Yan, S. Network In Network. *arXiv* **2013**, arXiv:1312.4400.
29. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
30. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016. Available online: http://www.deeplearningbook.org (accessed on 16 February 2023).
31. Harris, S.L.; Harris, D.M. *Digital Design and Computer Architecture*; Elsevier: Amsterdam, The Netherlands, 2016. [CrossRef]
32. Mesaros, A.; Heittola, T.; Virtanen, T. A multi-device dataset for urban acoustic scene classification. In Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018), Surrey, UK, 19–20 November 2018; pp. 9–13.
33. Fonseca, E.; Plakal, M.; Font, F.; Ellis, D.P.W.; Favory, X.; Pons, J.; Serra, X. General-purpose Tagging of Freesound Audio with AudioSet Labels: Task Description, Dataset, and Baseline. In Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018), Surrey, UK, 19–20 November 2018; pp. 69–73.
34. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv* **2015**, arXiv:1502.03167.
35. Nair, V.; Hinton, G.E. Rectified linear units improve restricted boltzmann machines. In Proceedings of the International Conference on Machine Learning (ICML-10), Madison, WI, USA, 21–24 June 2010; pp. 807–814.
36. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. *arXiv* **2015**, arXiv:1512.03385.
37. Cohen, N.; Sharir, O.; Shashua, A. Deep SimNets. *arXiv* **2015**, arXiv:1506.03059.
38. Powers, D.M.W. Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation. *J. Mach. Learn. Technol.* **2011**, *2*, 37–63.

applied sciences

MDPI

Article

Building Ensemble of Resnet for Dolphin Whistle Detection

Loris Nanni [1,*], Daniela Cuza [1] and Sheryl Brahnam [2]

[1] Department of Information Engineering, University of Padua, Via Gradenigo 6, 35131 Padova, Italy; daniela.cuza@studenti.unipd.it
[2] Department of Information Technology and Cybersecurity, Missouri State University, 901 S. National Street, Springfield, MO 65804, USA; sbrahnam@missouristate.edu
* Correspondence: loris.nanni@unipd.it

Abstract: Ecoacoustics is arguably the best method for monitoring marine environments, but analyzing and interpreting acoustic data has traditionally demanded substantial human supervision and resources. These bottlenecks can be addressed by harnessing contemporary methods for automated audio signal analysis. This paper focuses on the problem of assessing dolphin whistles using state-of-the-art deep learning methods. Our system utilizes a fusion of various resnet50 networks integrated with data augmentation (DA) techniques applied not to the training data but to the test set. We also present training speeds and classification results using DA to the training set. Through extensive experiments conducted on a publicly available benchmark, our findings demonstrate that our ensemble yields significant performance enhancements across several commonly used metrics. For example, our approach obtained an accuracy of 0.949 compared to 0.923, the best reported in the literature. We also provide training and testing sets that other researchers can use for comparison purposes, as well as all the MATLAB/PyTorch source code used in this study.

Keywords: convolutional neural network; dolphin whistle; ensemble; spectrogram classification

Citation: Nanni, L.; Cuza, D.; Brahnam, S. Building Ensemble of Resnet for Dolphin Whistle Detection. *Appl. Sci.* **2023**, *13*, 8029. https://doi.org/10.3390/app13148029

Academic Editors: Giovanni Costantini and Daniele Casali

Received: 6 June 2023
Revised: 6 July 2023
Accepted: 8 July 2023
Published: 10 July 2023

1. Introduction

Marine ecosystems play a critical role in maintaining the balance of our planet's ecosystem by supporting food security and contributing to climate regulation [1], making their preservation essential for the long-term sustainability of the earth's environment. Thus, there is a growing need to develop and test innovative monitoring systems to ensure the natural preservation of marine habitats. Modern technologies have already shown great potential in monitoring habitats and advancing our understanding of marine communities [2]. Acoustic methods are commonly used for underwater investigations because they can detect and classify sensitive targets, even in low visibility conditions. Passive acoustic technologies (PAM), such as underwater microphones, or hydrophones, are particularly attractive, as they allow for non-invasive continuous monitoring of marine ecosystems without interfering with biological processes [3]. PAM has been shown to achieve various research and management goals by effectively detecting animal calls [4]. These objectives may include tracking and localizing animals [5,6], species identification, identifying individuals [3,7], analyzing distributions and behavior [8], and estimating animal density [9].

The bottlenose dolphin (Tursiops truncatus) is a highly intelligent marine mammal and a critical species for researchers studying marine ecosystems [10]. Like many other marine mammals, dolphins are acoustic specialists that rely on sounds for communication, reproduction, foraging, and navigational purposes. The acoustic communication of dolphins employs a wide range of vocalizations, including clicks, burst-pulses, buzzes, and whistles [11]. Whistles, in particular, serve various social functions such as individual identification, group cohesion, and coordination of activities, such as feeding, resting,

socializing, and navigation [12]. Understanding and accurately detecting dolphin vocalizations is essential for monitoring their populations and assessing their role within marine ecosystems.

Traditional bioacoustics tools and algorithms for detecting dolphins have relied on spectrogram analysis, manual signal processing, and statistical methods [13]. For example, the reference approach pursued in [14] applies three noise removal algorithms to the spectrogram of a sound sample. Then, a connected region search is conducted to link together sections of the spectrogram that are above a predetermined threshold and close in time and frequency. A similar technique exploits a probabilistic Hough transform algorithm to detect ridges similar to thick line segments, which are then adjusted to the geometry of the potential whistles in the image via an active contour algorithm [15]. Other algorithmic methods aim to quantify the variation in complexity (randomness) occurring in the acoustic time series containing the vocalization; for example, by measuring signal entropy [16]. While these techniques have helped the study of dolphin vocalizations, they can be time-consuming and may not always provide accurate results due to the complexity and variability of the signals. Researchers have thus turned to machine learning methods to improve detection accuracy and efficiency.

Early machine learning studies in the field of dolphin detection applied traditional classifiers, such as Hidden Markov Models (HMM) [17] and Support Vector Machines (SVMs) [18]. For instance, in [19], a hidden Markov model was utilized for whistle classification, and in [20], classification and regression tree analysis was employed along with discriminant function analysis for categorizing parameters extracted from whistles. In [21], a multilayer perceptron classifier was implemented for classifying short-time Fourier transforms (STFTs) and wavelet transform coefficient energies of whistles. Lastly, in [15] a random forest algorithm and a support vector machine were combined to classify features derived from the duration, frequency, and cepstrum domain of whistles (see [22] for a review of the early literature).

More recently, researchers have employed deep learning methods to detect whistle vocalizations. Deep neural networks have demonstrated great potential in general sound detection [23] and specific underwater acoustic monitoring [24]. The Convolutional Neural Network (CNN) is one of the best-known deep learners. Though commonly considered an image classifier, CNNs have been applied to whale vocalizations, significantly reducing the false-positive rates compared to traditional algorithms, while at the same time enhancing call detection [25,26]. In [27], the authors compared four traditional methods for detecting dolphin echolocation clicks with six CNN architectures, demonstrating the superiority of the CNNs. In [28], CNNs were shown to outperform human experts in dolphin call detection accuracy. CNNs have also been applied to automatically categorize dolphin whistles into distinct groups, as in [29], and to extract whistle contours either by leveraging peak tracking algorithms [30] or by training CNN-based models for semantic segmentation [31].

Several studies of dolphin whistle classification have used data augmentation on the training set to enhance the performance of CNNs by reducing overfitting and increasing the size and variability of the available datasets [29,30,32]. Dolphin vocalizations are complex and highly variable, as analyzed in [33]. Unsurprisingly, some traditional music data augmentation methods, such as pitch shifting, time stretching, and adding background noise, have proven effective at this classification task. When synthesizing dolphin calls, care should be taken to apply augmentations to the audio signal rather than to the spectrograms, since altering the spectrogram could distort the time–frequency patterns of dolphin whistles, which would result in the semantic integrity of the labels being compromised [29,34]. In [29], primitive shapes were interjected into the audio signal to generate realistic ambient sounds in negative samples, and classical computer vision methods were used to create synthetic time–frequency whistles, which replaced the training data. Generative Adversarial Networks (GANs) have also been employed to generate synthetic dolphin vocalizations [32]. This research underscores the efficacy of data augmentation and synthesis

methods in enhancing both the precision and stability of dolphin whistle categorization models, especially in situations where the datasets are restricted or imbalanced.

The goal of this work is to continue exploring data augmentation techniques for the task of dolphin vocalization detection. To this end, we use the benchmark dolphin whistles dataset developed by Korkmaz et al. [28], but apply data augmentation to the original test set of spectrograms to enlarge it rather than the training set. The training set contains all the spectrograms obtained from audio files recorded between 24 June and 30 June, while the test set is composed of the spectrograms of audio files recorded between 13 July and 15 July, a three-day window. Aside from augmenting the test set, we extract a three-day window (24–26 June) from the training set as the validation set.

The proposed system outperforms previous state-of-the-art methods on the same dataset using the same testing protocol. We find our results interesting, especially since many misclassified audio samples are unclassifiable, even by humans. Therefore, the classification result of our method is likely very close to maximum performance (AUC = 1 is not obtainable).

The main contributions of this study are the following:

- The creation of a new baseline on this benchmark (note: using data augmentation on the testing set increased performance);
- Clear and repeatable criteria for testing various new developments in machine learning on this dataset by providing fixed training and test sets (both augmented and not augmented) rather than a protocol involving randomization;
- Access to all the MATLAB/PyTorch source code used in this study https://github.com/LorisNanni/ (accessed on 7 July 2023).

The remainder of this paper is organized into three sections. In Section 2, we present the material and methods, and Sections 2.1 and 2.2 provide a complete description of the dataset and baseline method presented in [28]. In Section 2.3, we offer a detailed account of our proposed approach. In Section 3, we present the results of tests comparing a standard ResNet with a set of ensembles, a comparison of our best ensemble with the state-of-the-art, and the results of using data augmentation on both the training and the test set. The conclusion in Section 4 discusses the shortcomings with the benchmark dataset and suggestions for further research.

2. Materials and Methods

2.1. Dataset

In this section, we describe the dataset developed by Korkmaz et al. [28] and detailed in that paper. The dataset contains 108,317 spectrograms, of which 49,807 are tagged as noise and 58,510 as dolphin whistles. The test set contains 6869 spectrograms. The data were collected with hydrophones during the summer of 2021 for 27 days from the dolphin's reef in Eilat, Israel. Following retrieval, a quality assurance (QA) process was conducted on the data to eliminate occasional disruptions and prolonged periods of noise. This QA procedure included the elimination of noise transients through wavelet denoising and the identification and removal of cut-off events via thresholding and bias reduction.

2.1.1. Data Preprocessing and Tagging

As described in [28], the collected data were subjected to a bandpass filter in the range of 5–20 kHz to align with the majority of dolphins' whistle vocalizations. The data were then passed through a whitening filter designed to rectify the hydrophone's open circuit voltage response ripples and the sensitivity of the sound card. The recorded audio files, which consisted of two channels, were averaged before the creation of spectrograms to decrease noise. In addition, the preprocessing pipeline eliminated signal outliers based on their length using the quartiles-based Tukey method [35], which led to the exclusion of signals that were longer than 0.78 s and shorter than 0.14 s.

The short-time fast Fourier transform of the signal was computed using MATLAB's spectrogram function from the digital signal processing toolbox to create the dolphin

whistle spectrograms. SFFT was performed with a Blackman function window with 2048 points, periodic sampling, and a hop size achieved by multiplying the window length by 0.8. The subsequent spectrograms were computed by shifting the signal window by 0.4 s. These spectrogram images were finalized by applying a gray-scale colormap, converting the frequency to kHz and the power spectrum density to dB, and restricting the y-axis between 3 and 20 kHz to emphasize the most significant (dominant) frequency range [36].

The spectrograms were then manually labeled by a human expert in two steps: initial tagging and validation tagging. The first step involved precise annotation of 5 s spectrograms over ten days of data collection, which were used to train an initial version of a deep learning classifier. This classifier was then used to select new portions of recordings containing potential dolphin sounds, which made tagging the remaining data in the validation phase more efficient. The validation phase only required the verification of positive samples detected by the preliminary deep learning classifier.

A human expert was tasked with identifying dolphin whistles as curving lines in the time–frequency domain and disregarding the contour lines generated by shipping radiated noise. When the discrimination process was complex, the expert directly listened to the recorded audio track to identify whistle-like sounds. The tagging resulted in a binary classification (whistle vs. noise) and a contour line marking the time–frequency characteristic of the identified whistle. This contour was used to assess the quality of the manual tagging by ensuring that the bandwidth of the identified whistle fell within the expected thresholds for a dolphin's whistle, specifically between 3 and 20 kHz. A second quality assessment was conducted by measuring the variance of the acoustic intensity of the identified whistle along the time–frequency contour, where the acoustic intensity of a valid whistle was expected to be stable.

2.1.2. Original Training and Test Sets

As mentioned in the introduction, the training set [28] contained all the spectrograms obtained from audio files recorded between 24 June and 30 June, while the test set was composed of the spectrograms of audio files recorded between 13 July and 15 July, a three-day window. The rationale given by the authors for dividing the training and test sets in this manner was primarily to test the generalizability of models using completely disparate sets of recordings, as this would better assess the detection accuracy amidst varying sea conditions.

As detailed in Section 2.3, we extracted a validation set from the training set obtained from audio files recorded between 24 June and 26 June. We used the validation set for learning the weights of the weighted sum rule, and then the whole training set was fed into the networks for classifying the test set.

2.2. Baseline Detection

PamGuard [14] is a widely used software designed to automatically recognize marine mammal vocalizations. It provides an interesting baseline method since it is widely used. The operational parameters of PamGuard were used as follows:

- The "Sound Acquisition" module from the "Sound Processing" section was included to manage the data acquisition device and convey its data to other modules;
- The "FFT (spectrogram) Engine" module from the "Sound Processing" section was incorporated to calculate spectrograms;
- The "Whistle and Moan Detector" module from the "Detectors" section was added for detecting dolphin whistles;
- The "Binary Storage" module from the "Utilities" section was incorporated to preserve information from various modules;
- A new spectrogram display was created by adding the "User Display" module from the "Displays" section.

Input spectrograms were devised utilizing the FFT analysis mentioned above with identical parameters: FFT window length was assigned 2048 points, and the hop size was

set to the length multiplied by 0.8 using the Blackman window in the "FFT (spectrogram) Engine" module under the software settings. The frequency range was determined between 3 and 20 kHz, and the "FFT (spectrogram) Engine Noise free FFT data" was chosen as the source of FFT data in the "Whistle and Moan Detector" module settings. During the creation of a new spectrogram display, the number of panels was assigned as 2 to visualize both channels. A detection by PamGuard was classified as a true positive if the signal window identified by the software overlapped with at least 5% of the ground truth signal interval. While this criterion may appear lenient, it allowed for the inclusion of many PamGuard detections that might have otherwise been disregarded.

2.3. Proposed Approach

The approach proposed in this study is illustrated in Figure 1. Our method is based on the combination of ten ResNet50 networks. The data augmentation phase was applied only to the test set and not to the training set, since it is already a large set of spectrograms. The data augmentation methods were selected using the validation set. Moreover, by using the validation set, the weights of the weighted sum rule are fixed (see Section 2.3.2). As illustrated in Figure 1, for each image of the test set, we classified three images: the original and two created by the data augmentation methods. The scores of these three images were combined using the weighted sum rule (see Section 3 for details), where the weights were found using the validation set. The weighted sum rule is a machine learning approach that combines the predictions of multiple models, in which a factor weights the contribution of each model, here learned on the validation set. Altogether, we had ten ResNet50 networks (each obtained by simply reiterating training), which produced ten weighted sums. These ten scores (i.e., the output of the ten weighted sum rules, one for each network) were combined with the classic sum rule, obtaining the final score of the method.

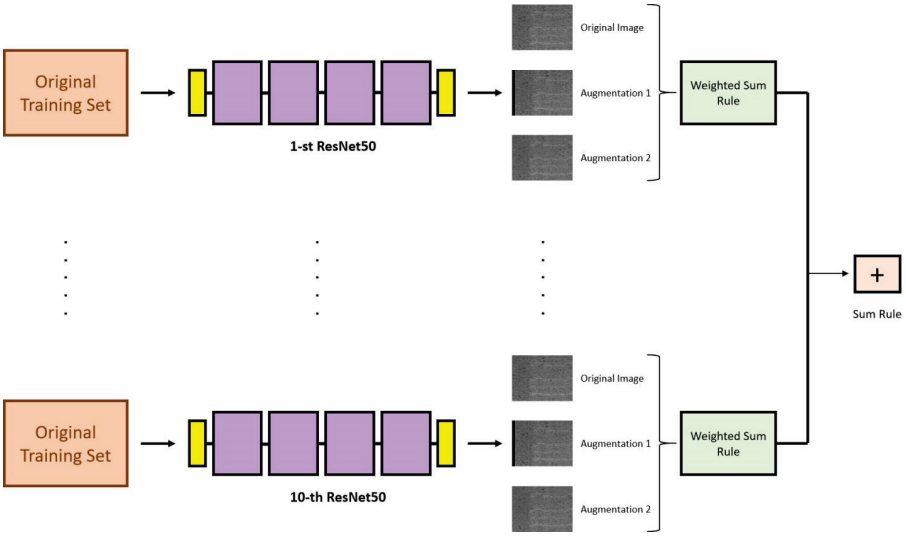


Figure 1. Proposed ensemble: for each image in the test set, we classified three images (the original and two augmented images) combined using the weighted sum rule.

In summary, we trained 10 resnet50 by simply tuning 10 times the ResNet50 network on the training dataset, then we used each of these 10 networks to classify the three images related to each pattern in the test set (original pattern and the two created by unsupervised data augmentation). For each network, we calculated the final score of each test pattern using the weighted sum rule, then these 10 scores (related to the 10 networks) were combined using the sum rule. These steps are described in more detail below.

2.3.1. ResNet50

ResNet50 is a convolutional neural network (CNN) architecture introduced by Microsoft Research in 2015 that belongs to a family of models called Residual Networks, or ResNets [37], which are widely used for various computer vision tasks, including image classification, object detection, and image segmentation. The key innovation of ResNet is the introduction of residual, or skip, connections for optimal gradient flow. ResNet enables the training of much deeper networks with improved performance by using skip connections. The name "ResNet50" signifies that this particular model has 50 layers.

The architecture of ResNet50 can be divided into several blocks. The input to the network is a 224 $\times$ 224 RGB image. The initial layer is a standard convolutional layer followed by a batch normalization layer and a ReLU activation function. This layer is followed by a max-pooling layer that reduces the spatial dimensions of the input. The main building blocks of ResNet50 are the residual blocks. Each residual block consists of a series of convolutional layers with batch normalization and ReLU activation. The output of these convolutional layers is added to the original input of the block through a skip connection. This addition operation allows the network to learn the residual information, i.e., the difference between the desired output and the input, which can be thought of as the "error" to be corrected.

ResNet50 contains several stacked residual blocks, with the number of blocks varying depending on the specific architecture. The model also includes bottleneck layers, which are 1 $\times$ 1 convolutional layers used to reduce the dimensionality of the feature maps, making the network more computationally efficient.

Towards the end of the network, a global average pooling layer spatially averages the feature maps, resulting in a fixed-length vector representation. This vector is fed into a fully connected layer with a softmax activation function, producing the final class probabilities.

Overall, ResNet50 is a powerful and influential CNN architecture that has significantly advanced the field of computer vision. Its use of residual connections has paved the way for the development of even deeper and more accurate neural networks, and it continues to serve as a benchmark for many state-of-the-art models in the field.

2.3.2. Validation Set Construction

The original training and test sets in [28], as described in Section 2.1.2, were used in this study. However, unlike the original authors, we extracted a validation set from the training set using all the spectrograms related to the three-day recording period of 24 June to 26 June. The validation set was used to fix the parameters of the weights for combining the scores using the sum rule of the different augmented spectrograms created for each test pattern. Our testing set was composed of the original image and two augmented images. The data augmentation approaches are detailed in Section 2.3.3.

Using the validation set, we combined the following three spectrograms for each test pattern using the weighted sum rule:

1. Original pattern;
2. Random shift with black or wrap;
3. Symmetric alternating diagonal shift.

2.3.3. Test Set Construction

The following two unsupervised data augmentation functions (see Figures 2 and 3) were used to generate two images for each test image:

1 The Random shift with black or wrap (RS) augmentation function undertakes the task of randomly shifting the content of each image. The shift can be either to the left or right, determined by an equal probability of 50% for each direction. The shift's magnitude falls within a specified shift width. Upon performing the shift, an empty space is created within the image. To handle this void, the function uses one of two strategies, each of which is selected with an equal chance of 50%. The first strategy is

to fill the space with a black strip, and the second is to wrap the cut piece from the original image around to the other side, effectively reusing the displaced part of the image. In our tests, we utilized a shift_width randomly selected between 1 and 90.

2 The symmetric alternating diagonal shift (SA) augmentation function applies diagonal shifts to distinct square regions within each image. Specifically, the content of a selected square region is moved diagonally in the direction of the top-left corner. The subsequent square region undergoes an opposite shift, with its content displaced diagonally towards the bottom-right corner. The size of the square regions is chosen randomly within the specified minimum and maximum size range.

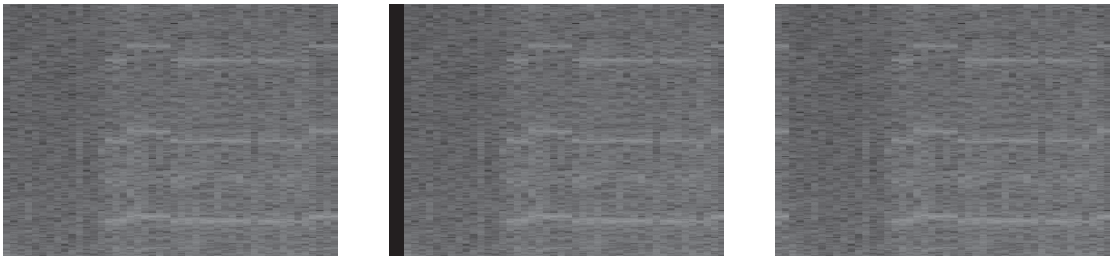


Figure 2. Spectrograms illustrating the RS method described in Section 2.3.3, with time on the *x*-axis and frequency in hertz on the *y*-axis. The **left** image showcases the original spectrogram. The **center** image presents the spectrogram after applying the random shift. The **right** image demonstrates the filled version of the spectrogram.

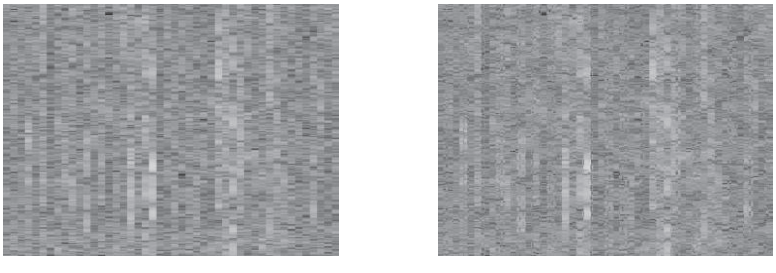


Figure 3. Illustration of the SA method described in Section 2.3.3, with time on the *x*-axis and frequency in hertz on the *y*-axis. The **left** image showcases the original spectrogram. The **right** image presents the spectrogram after SA.

We tested many data augmentation methods. Due to space constraints, we only present the the methods that were selected based on the validation set.

3. Experimental Results

The protocol used in our experiments mirrored that proposed in [28]. However, we used the validation set described in Section 2.3.2 to learn which data augmentation methods to apply and the weights of the weighted sum rule. After choosing the weights based on the validations set, we used the subdivision of the training and testing set described in [28]. We wish to stress that the validation set was extracted from the training set, so there was no overfitting on the test set. We gauged the performance of the model on the distinct test set by calculating the same performance indicators used in [28]. The true positive rate and the false positive rate was used to ascertain precision/recall. These metrics were used to generate the receiver operating characteristic (ROC) curves and evaluate the corresponding area under the ROC curve (AUC):

$$Precision = TP/(TP + FP) \quad (1)$$

$$Recall = TP/(TP + FN) \tag{2}$$

$$True\ Positive\ Rate = TP/(TP + FN); \tag{3}$$

where *TP* indicates true positives, *TN* indiates true negatives, *FP* indicates false positives, and *FN* indicates false negatives.

In Table 1, we present a comparison between the baseline ResNet50 and the proposed data augmented ResNet50 (named ResNet50_DA). ResNet50(x)_DA indicates the combination of x ResNet50_DA networks using the sum rule. Figure 4 reports the ROC curve for ResNet50(1) vs. ResNet50(10)_DA.

Table 1. Comparison (Area under the ROC curve) of baseline ResNet50(1) with the proposed augmented ensembles of ResNet50s (ResNet50(x)_DA (bold indicates best performance).

ResNets	AUC
ResNet50(1)	0.960
ResNet50(1)_DA	0.964
ResNet50(5)_DA	0.972
ResNet50(10)_DA	**0.973**

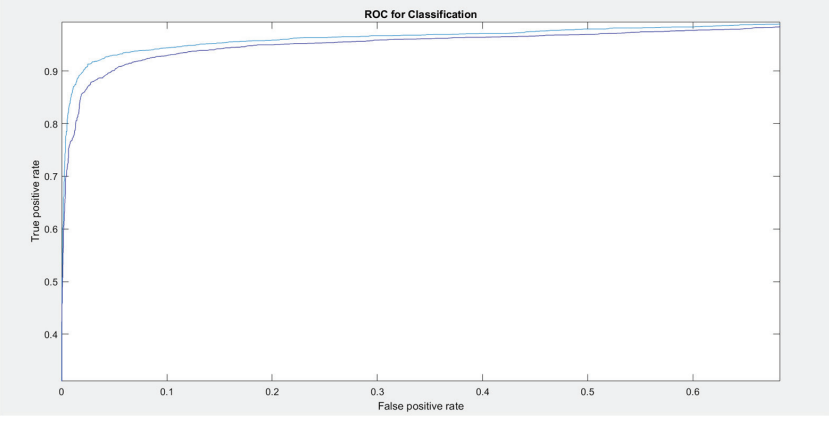


Figure 4. ROC curve (ratios of the true positives on the y-axis and false positives on the x-axis). The light blue represents our proposed ensemble, and the dark blue represents ResNet50(1), a single network.

We acknowledge that the performance increase recorded in Table 1 may not appear high compared to the baseline. However, our results are interesting because many of the misclassified samples are unclassifiable by humans. Thus, we are likely already very close to the maximum performance ($AUC = 1$ not obtainable). Furthermore, our results create a new baseline on an available dataset that can be repeated for testing other methods. The plot of the ROC curve in Figure 4 clearly shows that our proposed approach outperforms ResNet50(1). It is important to note that we obtained a true positive rate of 0.9 and a false positive rate of 0.02. Moreover, it is clear that the ResNet50(10)_DA improves ResNet50(1). The number of false positives of the standalone networks was more then two times the number of false positives of the ensemble.

In Table 2, we present a comparison between our proposed method and two other approaches using the same dataset with the same testing protocol, reporting a full set of performance indicators (accuracy, AUC, precision, and recall). Clearly, the proposed ensemble performed better than the methods reported in the literature, although with higher computational costs. We do not believe this is a problem, considering that the current computing power of GPUs and the developments expected in the coming years will

reduce the considerations of such costs. For example, using a NVIDIA 1080, we were able to classify a batch of 100 spectrograms in ~0.3 s (considering a standalone ResNet50). Using a TitanRTX, we were able to classify a batch of 100 spectrograms in ~0.195 s (considering a standalone ResNet(50).

Table 2. Comparison with the literature using four measures.

Method	Accuracy	AUC	Precision	Recall
Pamguard [14]	0.664	---	0.755	0.195
[28]	0.923	0.960	0.905	0.896
ResNet50(10)_DA	**0.949**	**0.973**	**0.965**	**0.902**

In Table 3, we present a report of the confusion matrix obtained by our proposed ensemble and the previous baseline on the same dataset. This test shows that the reliability of the proposed method reduces the number of false noise and false whistle classifications with respect to the previous baselines. In addition, Cohen's kappa coefficient is also shown in the same table; this performance indicator also shows that the proposed ensemble outperformed the previous baseline.

Table 3. Confusion matrices and Cohen's kappa coefficient.

	Here		**[28]**		**Pamguard [14]**		**Cohen's Kappa**		
							Here	**[28]**	**Pamguard [14]**
	Noise	Whistle	Noise	Whistle	Noise	Whistle			
Noise	4124	88	3963	249	4044	168	0.8919	0.8383	0.1797
Whistle	260	2397	277	2380	2139	518			

In addition to the tests reported above, we conducted experiments in which the two data augmentation approaches selected on the validation set were applied to the whole training set. Due to the large size of the augmented training set, the training time increased to ~2100 min using a machine with a NVIDIA Titan X with 12 GB of ram. Increasing the size of the training set only slightly increased the performance. Once again, applying data augmentation to the test data using the weighted sum rule adopted in this paper resulted in better performance than using only the original test set. We obtained the following performance metrics:

1 Data augmentation applied to the training set, with the test set consisting of only the original images: AUC: 0.968; Accuracy: 0.940; Recall: 0.911 Precision: 0.931;
2 Data augmentation applied to both the training set and test set, with the proposed weighted sum rule used for the test set: AUC: 0.970; Accuracy: 0.941; Recall: 0.911; Precision: 0.934.

4. Conclusions

The surge in human activities in marine environments has led to an influx of boats and ships that emit powerful acoustic signals, often impacting areas larger than 20 square kilometers. The underwater noise from larger vessels can surpass 100 PSI, disturbing marine mammals' hearing, navigation, and foraging abilities, particularly for coastal dolphins [38,39]. Therefore, the monitoring and preservation of marine ecosystems and wildlife is paramount. However, conventional monitoring technologies depend on detection methods that are less than ideal, thereby hindering our capacity to carry out extensive, long-term surveys. While automatic detection methods could significantly enhance our survey capabilities, their performance is typically subpar amidst high background noise levels.

In this paper, we illustrated how deep learning techniques involving data augmentation can identify dolphin whistles with remarkable accuracy, positioning them as a promising candidate for standardizing the automatic processing of underwater acoustic signals.

We obtained state-of-the-art results and provided a training and test set for fair comparison. In terms of accuracy, we obtained a nearly 0.03 accuracy gain. The MATLAB/PyTorch source code used in this study is freely provided (https://github.com/LorisNanni/ accessed on 7 July 2023).

Despite the need for additional research to confirm the efficacy of such techniques across various marine environments and animal species, we are confident that deep learning will pave the way for developing and deploying economically feasible monitoring platforms. We hope that our new baseline will further the comparison of future deep learning techniques in this area.

Finally, we should stress the main cons of using this dataset as a benchmark: the training and test set were from the same region (Dolphin's Reef in Eilat, Israel), and the samples were collected using the same acoustic recorder.

Author Contributions: Conceptualization, L.N.; methodology, L.N. and D.C.; software, L.N. and D.C.; writing—original draft preparation, L.N. and S.B.; writing—review and editing, L.N., D.C. and S.B.; supervision, L.N. and S.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: https://csms-acoustic.haifa.ac.il/index.php/s/2UmUoK80Izt0Roe accessed on 7 July 2023.

Acknowledgments: The authors would like to acknowledge the support that NVIDIA provided through the GPU Grant Program. The authors also used a donated TitanX GPU to train the deep networks used in this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Halpern, B.S.; Frazier, M.; Afflerbach, J.; Lowndes, J.S.; Micheli, F.; O'hara, C.; Scarborough, C.; Selkoe, K.A. Recent pace of change in human impact on the world's ocean. *Sci. Rep.* **2019**, *9*, 11609. [CrossRef] [PubMed]
2. Danovaro, R.; Carugati, L.; Berzano, M.; Cahill, A.E.; Carvalho, S.; Chenuil, A.; Corinaldesi, C.; Cristina, S.; David, R.; Dell'Anno, A.; et al. Implementing and Innovating Marine Monitoring Approaches for Assessing Marine Environmental Status. *Front. Mar. Sci.* **2016**, *3*, 213. [CrossRef]
3. Gibb, R.; Browning, E.; Glover-Kapfer, P.; Jones, K.E. Emerging opportunities and challenges for passive acoustics in ecological assessment and monitoring. *Methods Ecol. Evol.* **2019**, *10*, 169–185. [CrossRef]
4. Desjonquères, C.; Gifford, T.; Linke, S. Passive acoustic monitoring as a potential tool to survey animal and ecosystem processes in freshwater environments. *Freshw. Biol.* **2020**, *65*, 7–19. [CrossRef]
5. Macaulay, J.; Kingston, A.; Coram, A.; Oswald, M.; Swift, R.; Gillespie, D.; Northridge, S. Passive acoustic tracking of the three-dimensional movements and acoustic behaviour of toothed whales in close proximity to static nets. *Methods Ecol. Evol.* **2022**, *13*, 1250–1264. [CrossRef]
6. Wijers, M.; Loveridge, A.; Macdonald, D.W.; Markham, A. CARACAL: A versatile passive acoustic monitoring tool for wildlife research and conservation. *Bioacoustics* **2021**, *30*, 41–57. [CrossRef]
7. Ross, S.R.P.; O'Connell, D.P.; Deichmann, J.L.; Desjonquères, C.; Gasc, A.; Phillips, J.N.; Sethi, S.S.; Wood, C.M.; Burivalova, Z. Passive acoustic monitoring provides a fresh perspective on fundamental ecological questions. *Funct. Ecol.* **2023**, *37*, 959–975. [CrossRef]
8. Kowarski, K. Humpback Whale Singing Behaviour in the Western North Atlantic: From Methods for Analysing Passive Acoustic Monitoring Data to Understanding Humpback Whale Song Ontogeny. Ph.D. Thesis, Dalhousie University, Halifax, NS, Canada, 2020.
9. Arranz, P.; Miranda, D.; Gkikopoulou, K.C.; Cardona, A.; Alcazar, J.; de Soto, N.A.; Thomas, L.; Marques, T.A. Comparison of visual and passive acoustic estimates of beaked whale density off El Hierro, Canary Islands. *J. Acoust. Soc. Am.* **2023**, *153*, 2469. [CrossRef]
10. Lusseau, D. The emergent properties of a dolphin social network. *Proc. R. Soc. B Boil. Sci.* **2003**, *270* (Suppl. 2), S186–S188. [CrossRef]

11. Lehnhoff, L.; Glotin, H.; Bernard, S.; Dabin, W.; Le Gall, Y.; Menut, E.; Meheust, E.; Peltier, H.; Pochat, A.; Pochat, K.; et al. Behavioural Responses of Common Dolphins Delphinus delphis to a Bio-Inspired Acoustic Device for Limiting Fishery By-Catch. *Sustainability* **2022**, *14*, 13186. [CrossRef]
12. Papale, E.; Fanizza, C.; Buscaino, G.; Ceraulo, M.; Cipriano, G.; Crugliano, R.; Grammauta, R.; Gregorietti, M.; Renò, V.; Ricci, P.; et al. The Social Role of Vocal Complexity in Striped Dolphins. *Front. Mar. Sci.* **2020**, *7*, 584301. [CrossRef]
13. Oswald, J.N.; Barlow, J.; Norris, T.F. Acoustic identification of nine delphinid species in the eastern tropical Pacific Ocean. *Mar. Mammal Sci.* **2003**, *19*, 20–37. [CrossRef]
14. Gillespie, D.; Caillat, M.; Gordon, J.; White, P. Automatic detection and classification of odontocete whistles. *J. Acoust. Soc. Am.* **2013**, *134*, 2427–2437. [CrossRef] [PubMed]
15. Serra, O.; Martins, F.; Padovese, L. Active contour-based detection of estuarine dolphin whistles in spectrogram images. *Ecol. Informatics* **2020**, *55*, 101036. [CrossRef]
16. Siddagangaiah, S.; Chen, C.-F.; Hu, W.-C.; Akamatsu, T.; McElligott, M.; Lammers, M.O.; Pieretti, N. Automatic detection of dolphin whistles and clicks based on entropy approach. *Ecol. Indic.* **2020**, *117*, 106559. [CrossRef]
17. Parada, P.P.; Cardenal-López, A. Using Gaussian mixture models to detect and classify dolphin whistles and pulses. *J. Acoust. Soc. Am.* **2014**, *135*, 3371–3380. [CrossRef] [PubMed]
18. Jarvis, S.; DiMarzio, N.; Morrissey, R.; Morretti, D. Automated classification of beaked whales and other small odontocetes in the tongue of the ocean, bahamas. In Proceedings of the OCEANS 2006, Boston, MA, USA, 18–21 September 2006.
19. Ferrer-I-Cancho, R.; McCowan, B. A Law of Word Meaning in Dolphin Whistle Types. *Entropy* **2009**, *11*, 688–701. [CrossRef]
20. Oswald, J.N.; Rankin, S.; Barlow, J.; Lammers, M.O. A tool for real-time acoustic species identification of delphinid whistles. *J. Acoust. Soc. Am.* **2007**, *122*, 587–595. [CrossRef]
21. Mouy, X.; Bahoura, M.; Simard, Y. Automatic recognition of fin and blue whale calls for real-time monitoring in the St. Lawrence. *J. Acoust. Soc. Am.* **2009**, *126*, 2918–2928. [CrossRef] [PubMed]
22. Usman, A.M.; Ogundile, O.O.; Versfeld, D.J.J. Review of Automatic Detection and Classification Techniques for Cetacean Vocalization. *IEEE Access* **2020**, *8*, 105181–105206. [CrossRef]
23. Abayomi-Alli, O.O.; Damaševičius, R.; Qazi, A.; Adedoyin-Olowe, M.; Misra, S. Data Augmentation and Deep Learning Methods in Sound Classification: A Systematic Review. *Electronics* **2022**, *11*, 3795. [CrossRef]
24. Testolin, A.; Diamant, R. Combining denoising autoencoders and dynamic programming for acoustic detection and tracking of underwater moving targets. *Sensors* **2020**, *20*, 2945. [CrossRef] [PubMed]
25. Jiang, J.-J.; Bu, L.-R.; Duan, F.-J.; Wang, X.-Q.; Liu, W.; Sun, Z.-B.; Li, C.-Y. Whistle detection and classification for whales based on convolutional neural networks. *Appl. Acoust.* **2019**, *150*, 169–178. [CrossRef]
26. Zhong, M.; Castellote, M.; Dodhia, R.; Ferres, J.L.; Keogh, M.; Brewer, A. Beluga whale acoustic signal classification using deep learning neural network models. *J. Acoust. Soc. Am.* **2020**, *147*, 1834–1841. [CrossRef] [PubMed]
27. Buchanan, C.; Bi, Y.; Xue, B.; Vennell, R.; Childerhouse, S.; Pine, M.K.; Briscoe, D.; Zhang, M. Deep convolutional neural networks for detecting dolphin echolocation clicks. In Proceedings of the 2021 36th International Conference on Image and Vision Computing New Zealand (IVCNZ), Tauranga, New Zealand, 9–10 December 2021.
28. Korkmaz, B.N.; Diamant, R.; Danino, G.; Testolin, A. Automated detection of dolphin whistles with convolutional networks and transfer learning. *Front. Artif. Intell.* **2023**, *6*, 1099022. [CrossRef] [PubMed]
29. Li, L.; Qiao, G.; Liu, S.; Qing, X.; Zhang, H.; Mazhar, S.; Niu, F. Automated classification of *Tursiops aduncus* whistles based on a depth-wise separable convolutional neural network and data augmentation. *J. Acoust. Soc. Am.* **2021**, *150*, 3861–3873. [CrossRef] [PubMed]
30. Li, P.; Liu, X.; Palmer, K.J.; Fleishman, E.; Gillespie, D.; Nosal, E.M.; Shiu, Y.; Klinck, H.; Cholewiak, D.; Helble, T.; et al. Learning deep models from synthetic data for extracting dolphin whistle contours. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020.
31. Jin, C.; Kim, M.; Jang, S.; Paeng, D.-G. Semantic segmentation-based whistle extraction of Indo-Pacific Bottlenose Dolphin residing at the coast of Jeju island. *Ecol. Indic.* **2022**, *137*, 108792. [CrossRef]
32. Zhang, L.; Huang, H.-N.; Yin, L.; Li, B.-Q.; Wu, D.; Liu, H.-R.; Li, X.-F.; Xie, Y.-L. Dolphin vocal sound generation via deep WaveGAN. *J. Electron. Sci. Technol.* **2022**, *20*, 100171. [CrossRef]
33. Kershenbaum, A.; Sayigh, L.S.; Janik, V.M. The encoding of individual identity in dolphin signature whistles: How much information is needed? *PLoS ONE* **2013**, *8*, e77671. [CrossRef] [PubMed]
34. Padovese, B.; Frazao, F.; Kirsebom, O.S.; Matwin, S. Data augmentation for the classification of North Atlantic right whales upcalls. *J. Acoust. Soc. Am.* **2021**, *149*, 2520–2530. [CrossRef] [PubMed]
35. Tukey, J.W. Comparing Individual Means in the Analysis of Variance. *Biometrics* **1949**, *5*, 99–114. [CrossRef] [PubMed]
36. Jones, B.; Zapetis, M.; Samuelson, M.M.; Ridgway, S. Sounds produced by bottlenose dolphins (*Tursiops*): A review of the defining characteristics and acoustic criteria of the dolphin vocal repertoire. *Bioacoustics* **2020**, *29*, 399–440. [CrossRef]
37. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
38. Ketten, D.R. Underwater ears and the physiology of impacts: Comparative liability for hearing loss in sea turtles, birds, and mammals. *Bioacoustics* **2008**, *17*, 312–315. [CrossRef]

39. Erbe, C.; Marley, S.A.; Schoeman, R.P.; Smith, J.N.; Trigg, L.E.; Embling, C.B. The Effects of Ship Noise on Marine Mammals—A Review. *Front. Mar. Sci.* **2019**, *6*, 606. [CrossRef]

applied sciences

MDPI

Article

OneBitPitch (OBP): Ultra-High-Speed Pitch Detection Algorithm Based on One-Bit Quantization and Modified Autocorrelation

Davide Coccoluto, Valerio Cesarini and Giovanni Costantini *

Department of Electronic Engineering, University of Rome Tor Vergata, 00133 Rome, Italy; davide.coccoluto@gmail.com (D.C.); valerio.cesarini@uniroma2.it (V.C.)
* Correspondence: costantini@uniroma2.it

Featured Application: Fast pitch detection algorithm for the real-time estimation of the fundamental frequency, optimized for hardware implementation.

Abstract: This paper presents a novel, high-speed, and low-complexity algorithm for pitch (F0) detection, along with a new dataset for testing and a comparison of some of the most effective existing techniques. The algorithm, called OneBitPitch (OBP), is based on a modified autocorrelation function applied to a single-bit signal for fast computation. The focus is explicitly on speed for real-time pitch detection applications in pitch detection. A testing procedure is proposed using a proprietary synthetic dataset (SYNTHPITCH) against three of the most widely used algorithms: YIN, SWIPE (Sawtooth Inspired Pitch Estimator) and NLS (Nonlinear-Least Squares-based). The results show how OBP is 9 times faster than the fastest of its alternatives, and 50 times faster than a gold standard like SWIPE, with a mean elapsed time of 4.6 ms, or 0.046 × realtime. OBP is slightly less accurate for high-precision landmarks and noisy signals, but its performance in terms of acceptable error (<2%) is comparable to YIN and SWIPE. NLS emerges as the most accurate, but it is not flexible, being dependent on the input and requiring prior setup. OBP shows to be robust to octave errors while providing acceptable accuracies at ultra-high speeds, with a building nature suited for FPGA (Field-Programmable Gate Array) implementations.

Keywords: pitch detection; F0; algorithm; auto-tune; audio signal processing

Citation: Coccoluto, D.; Cesarini, V.; Costantini, G. OneBitPitch (OBP): Ultra-High-Speed Pitch Detection Algorithm Based on One-Bit Quantization and Modified Autocorrelation. *Appl. Sci.* **2023**, *13*, 8191. https://doi.org/10.3390/app13148191

Academic Editor: Douglas O'Shaughnessy

Received: 4 June 2023
Revised: 10 July 2023
Accepted: 11 July 2023
Published: 14 July 2023

1. Introduction

A signal is defined as periodic when the same sequence of values re-occurs after a fixed amount of time, defined as a period, whose inverse is the frequency. By Fourier's principles, a real-world signal can be described as a sum of sinusoids (or "pure tones") that only carry one frequency [1]. The fundamental frequency, or F0, is defined as the lowest frequency describing a periodic component of a signal. In the audio domain, F0 is defined as the pitch, which in music is translated to a specific note.

Detecting the F0 of a signal is a crucial application in many fields, with audio, music, and speech processing being heavily reliant on pitch-detection-based technologies, as well as other fields such as fault detection of moving parts, which are related to their resonance frequency [2], or sonar systems for target detection, classification and localization [3]. Moreover, pitch detection can also be employed in the characterization of accurate sinusoidal voltages, as described by Krajewski et al. [4].

The problem of pitch detection is crucial in all the applications that rely on knowing the fundamental frequency in order to perform periodicity-related computations, such as acoustic feature extraction relying on prosodic metrics such as HNR, jitter or shimmer that evaluate "cycle-to-cycle" variations [5]. Moreover, professional audio relies on pitch detection to build tuners for real instruments, or for real-time pitch re-adjusting applications

especially directed toward vocal tuning. Real-time detectors are thus necessary to enable performers to monitor pitch accuracy and trigger events in real time, especially related to MIDI applications [6]. Additionally, fast, real-time pitch detection is valuable in interactive audio applications, such as games and virtual reality, where it enables dynamic audio synthesis and effects as well as responsive processing.

The two main characteristics of pitch detection algorithms are speed and accuracy, which often imply a trade-off depending on the specific application—for example, posterior analyses do not need real-time F0 estimation, as opposed to live music [6].

With the diffusion of technologies such as "Autotune" for real-time pitch correction in singers [7], and with the widespread use of MIDI instruments and/or live MIDI transformers to digitalize acoustic instruments, real-time pitch detection sees a crucial application in the music industry. Especially for MIDI purposes where a sound needs to be translated into a discrete note, speed is favored over accuracy due to the need for low-latency live solutions, and thanks to the fact that the detected frequency is discretized into a note of the tempered system allowing for a certain range for errors. The problem of detecting the fundamental frequency in real time is crucial whenever live performances are involved, as even minuscule latencies of a few milliseconds can be perceived by the musician or operator.

In speech analysis, F0 carries a crucial role as a biometric feature for characterizing voice impairment, up to singlehandedly being used for pre-diagnostic purposes, where F0-related features and their variations are used for the detection of respiratory [8], phonatory [9,10] or neurodegenerative diseases [11–13].

Given the multifaceted applications and different industry needs, the state of the art of pitch detection depends on the application.

From the mathematical point of view, although an ever-growing plethora of algorithms are being developed, the vast majority can be generally categorized into three approaches: time-based, frequency or Cepstrum-based, full heuristic. Time domain approaches are generally based on the mathematical principles behind autocorrelation, which inherently has peaks every time a signal repeats (maximum correlation with itself): the problem of pitch detection is thus translated into the problem of finding the maximum of the autocorrelation function. Frequency domain algorithms are based on Fourier domain or cepstral analysis, with the Harmonic Product Spectrum (HPS) [14] as a notable example: it computes the product of the power spectra of a signal and its downsampled versions to emphasize harmonic components. The fundamental frequency is then estimated by identifying peaks in the resulting spectrum. In recent years, the research trend was predominantly based on Deep Learning methods based on Convolutional Neural Networks (CNN) [15], which offer the advantages of being able to control the size of the computational net and also to specifically train on suitable data, since the problem of pitch detection is data-dependent [16]. However, CNN-based methods are not easily generalizable, and require data and especially time for training time before they are usable in their "inference" form—which usually brings high accuracies but slightly less optimal speed [17].

The prior definition of the latency/complexity of a pitch detection algorithm is hard to determine, since each algorithm and performance inherently depends on the acoustic and digital nature of the data—such as the number of bits for quantization.

Theoretically, HPS-based methodologies yield a complexity of O(NlogN), and straightforward autocorrelation is at $O(N^2)$, whereas FFT-based autocorrelation is O(NlogN) yet again, being comparable to HPS. The Fast Fourier Transform (FFT) algorithm is used to speed up the computation of autocorrelation. By leveraging the symmetry properties in the autocorrelation function, the complexity of FFT-based autocorrelation is reduced. The process involves padding the input signal to the nearest power of 2, computing the FFT of the padded signal, squaring the magnitude of each frequency bin to obtain the power spectrum, computing the inverse FFT of the power spectrum and normalizing the result by dividing it by the length of the input signal.

After a rough estimate of F0, many algorithms have to rely on some corrective heuristics to fine-tune the result and/or avoid octave errors. Common signal processing solutions

may be employed for this purpose as well, with notable results such as the work by Khadem-hosseini et al. [18] employing HPS and Euclidean summation. However, although this results in improved accuracy, it is an inherently computationally expensive mean, and real-time pitch detectors might choose to avoid relying on correctors—this is also the approach that we chose in the present paper.

Due to the inherent harmonic nature of most real-world sound signals, especially when dealing with speech or music analysis, octave errors are a common criticality among pitch detection algorithms, being triggered by the eventual presence of strong first- or second-order harmonics and, partially, by aliasing.

Two of the most widely used algorithms, SWIPE [19] and YIN [20], which will be detailed later in Section 2, are based on autocorrelation. More algorithms are listed in the works by Camacho and Harris [19] and Ruslan et al. [1].

Attempts at fast pitch detectors are based on the simplification of the transformation procedures, such as the work by Grinewitschus et al. [21], which leverages the constant-Q Gabor transform for a threshold-based approach within a four-dimensional logarithmic harmonic spectrum shift. A work by Mnasri et al. [22] aims to avoid short-time analysis and thus the underlying approximations about local stationarity by employing the Hilbert transform to derive "instantaneous" frequency components to contour F0: the performances might be comparable to YIN or SWIPE, but no indication on speed is given.

In general, the problem of real-time, high-speed pitch detection has to be faced with the development of a computationally light algorithm that still retains a relative error suitable for the required application (mainly professional audio and live performances).

The scope of this paper is to propose a novel, high-speed implementation of a pitch detection algorithm based on a modified version of the autocorrelation, and to assess the performances of the most highly regarded algorithms in terms of speed and accuracy, on a suitable dataset purposefully built as a test bench.

For the purposes of testing pitch detection algorithms on sheer speed or recognition capabilities, a custom dataset named SYNTHPITCH was built by producing synthetic signals so that the original pitch/F0 is objective and priorly known.

Other algorithms such as YAAPT [23], SHRP [24] or the CREPE [25] CNN approach have been experimented with, but their preliminary results were not notable with respect to the others considered, especially for the speed vs. accuracy tradeoff. With speed being the main characteristic to search for, non-notable algorithms that provide high accuracies but poor speed have not been included in the present assessment although experimentations were made on them in order to rule them out, and synthetic signals are employed to evaluate the sheer computational complexity, while not forgoing the ability to infer pitch.

The main contributions of this paper lie in the presentation of a novel pitch detection algorithm, based on a partially unexplored approach focused on high-speed and low bit depth, very suitable for hardware implementations. All of these characteristics make it a good candidate for live performance applications or MIDI instruments, which rely on real-time note detection. The mathematical and signal processing theories behind our novel algorithm explore the characteristics of the autocorrelation function, its maximization and its approximations, as well as the effect of quantization on the fundamental frequency of a signal.

Along with the new algorithm, a testing paradigm for evaluating the speed and computational complexity of pitch detection algorithms is proposed, and a custom, synthetic dataset is produced and made available to the public. State-of-the-art, pre-existing pitch detection algorithms, especially those focusing on speed, are thoroughly explored and tested. The article is organized as follows. Section 2 presents the OneBitPitch algorithm along with its mathematical discussion, theoretical derivation and implementation, as well as three other algorithms (YIN, NLS, SWIPE) used for comparison and the custom SYNTHPITCH dataset produced and used in this paper. Section 3 presents the numerical results obtained from the simulations, as well as a statistical analysis. Then, Section 4 provides an in-depth discussion of the obtained results, especially focusing on real-time implementation and

runtime speed, which are the main focuses of the analyses. The strengths and weaknesses of every algorithm are analyzed along with practical situations where each algorithm is best suited. Limitations and future works, especially regarding hardware solutions, are presented at the end of the Discussion and before the Conclusions.

2. Materials and Methods

This paper proposes a novel pitch detection algorithm called OneBitPitch, in short, OBP, based on a modified autocorrelation function applied to a one-bit version of the original signal, for maximum speed and hardware implementation capabilities. In order to evaluate the performance of the OneBitPitch algorithm, a custom synthetic dataset (SYNTHPITCH) was built and 4 different algorithms were compared on it.

The choice of the algorithms was based on well-known, state-of-the-art pitch estimators especially directed toward high speed or high accuracy. The main focus is the sheer algorithmic performance, although it is well known that the effectiveness of any pitch detection model is inherently dependent on the dataset and purpose (i.e., pure signals, voice data, etc.).

This section first presents the public dataset built for the purpose of this study, then details the OBP algorithm along with its mathematical basis and briefly presents the three algorithms used for comparison: YIN, SWIPE and NLS.

2.1. SYNTHPITCH Dataset

The SYNTHPITCH dataset was purposefully built for the scope of the analyses presented in this work, i.e., for testing pitch detection algorithms on arbitrarily complex signals in terms of fundamental frequency intelligibility, and to evaluate their computational complexity/speed.

All signals were sampled at 96 ksps and represented in floating points. A typical audio setup was reproduced, so a 20 kHz low-pass filter was applied. Twelve different categories are present, with each one encompassing 99 signals with increasing fundamental frequency, starting from 100 Hz up to 5000 Hz with a 50 Hz step size.

The categories were built with increasing pitch complexity, the simplest one being pure sine waves, to which multiple artifacts have been applied to generate sounds of increasing complexity. There are also two categories encompassing square waves; the amplitude of each starting wave is normalized to have a peak of 1. Table 1 details the characteristics of each category and its name, as well as the number of artifacts applied, with the following macroscopic characteristics:

- Harmonics: Addition of a number of harmonic frequencies, i.e., integer multiples of F0. The amplitude of each harmonic is a random number between 0 and 1, sampled from the Gaussian distribution, and eventually re-scaled if more/less amplitude is needed;
- Partials: Addition of non-integer, random multiples of F0, with random starting amplitude linearly scaled according to the order, and random phase. The following formula explains the construction of an i-th order partial, with A_i, f_i and φ_i being random amplitude (scaled according to the order), random frequency obtained by multiplying F0 by a random number between 0 and 1 (scaled according to the order) and random phase. All random quantities are obtained by sampling a Gaussian distribution with max = 1, and the formula is as follows:

$$Partial_i(t) = \frac{A_i}{\mathrm{i}} \cdot \sin(2\pi i f_i t + \varphi_i); \quad (1)$$

- White Gaussian Noise: Addition of white Gaussian noise of a given SNR, after measuring the power of the signal with added harmonics/partials;
- Reverb: Addition of a reverberated copy of the signal with amplitude equal to 0.1 of the measured amplitude of the starting signal.

Table 1. Name and description of the signal categories making up the SYNTHPITCH dataset. The order is alphabetical.

Name	Description
2harm	Pure sine waves plus the first two harmonics with random amplitude between 0 and 1
2harm_wgn15	Pure sine waves plus 2 harmonics (random amplitude between 0 and 1) plus white Gaussian noise with SNR = 15
4harm	Pure sine waves plus 4 harmonics with random amplitude between 0 and 1
4harm_4part_wgn15	Pure sine waves plus 4 harmonics (random amplitude between 0 and 1) and 4 partials (linearly decreasing amplitude)
4harm_high	Pure sine waves plus 4 harmonics with random amplitude between 0 and 3
4harm_wgn15	Pure sine waves plus 4 harmonics with random amplitude between 0 and 1 plus white Gaussian noise with SNR = 15
full1	Pure sine waves plus 10 harmonics (random amplitude between 0 and 2) and 10 partials (maximum amplitude = 2), plus white Gaussian noise with SNR = 1 and reverb (0.1 RMS)
full2	Pure sine waves plus 10 harmonics (random amplitude between 0 and 2) and 10 partials (maximum amplitude = 2), plus white Gaussian noise with SNR = 10 and reverb (0.1 RMS)
pure	Pure sine waves
pure_wgn0P3	Pure sine waves plus white Gaussian noise with SNR = 0.3
square_pure	Square waves
square_wgn10	Square waves plus white Gaussian noise with SNR = 10

Figure 1 details some examples of signals found on the SYNTHPITCH dataset; notice how, with complex/dirty signals such as those present in the "full1" category, pitch and sinusoidal behavior become very hard to infer. The dataset is free to use for the public.

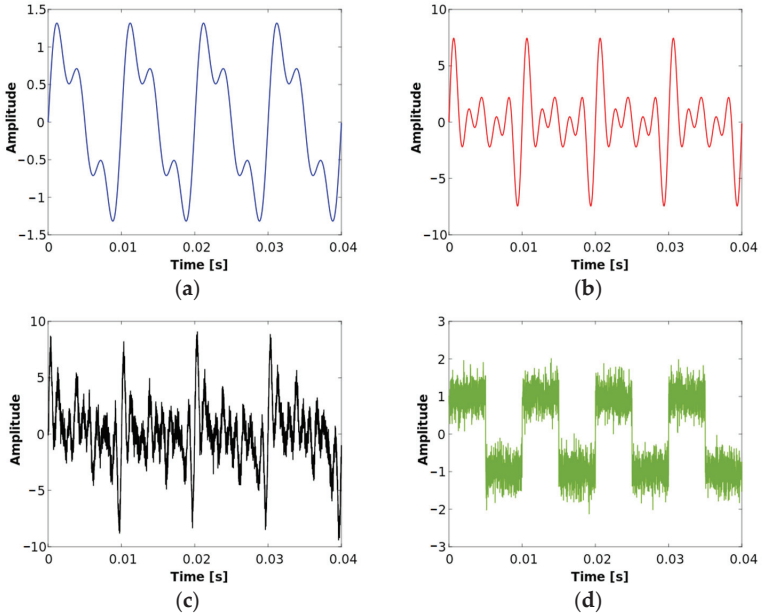


Figure 1. Sample signals from the SYNTHPITCH dataset (F0 = 100 Hz): (**a**) "2harm" sample (blue); (**b**) "4harm_high" sample (red); (**c**) "full1" sample (black); (**d**) "square_wgn10" sample (green).

2.2. OneBitPitch Algorithm

The algorithm proposed in this paper is aimed at ultra-fast pitch detection, for real-time usage, and was developed as a starting point for future hardware implementation and for heavy-duty, latency-free live use.

With these premises, our proposed algorithm aims to reduce the computational complexity to its bare minimum, sacrificing accuracy while still staying in acceptable territories, to provide the highest possible speed performances.

The OneBitPitch (OBP for short) algorithm exploits a modified version of the autocorrelation in time approach highly optimized for execution time and computational complexity.

The basic idea is that reducing the resolution of a signal, i.e., the number of quantization bits, worsens the signal but retains its periodicity. Taking this idea to its limit, we can state the following:

Proposition 1. *Let x be a digital signal modeled as a zero-average periodic sequence quantized with N bits and with F0 being its fundamental frequency. Re-quantizing x with $M < N$ bits, the original F0 is preserved in the re-quantized signal.*

This can be easily proven by considering that, for a periodic signal in which F0 is the reciprocal of a period T0, the time duration of such a period does not change with truncation (re-quantization), although the exact timeframe can be anticipated/delayed according to rounding conventions [26–28]. With the assumption of no aliasing (anti-aliasing has been performed priorly), this holds true for any amount of bits, up to the minimum limit of 1-bit quantization, which basically results in the sign function. Although discretizing the amplitudes might indeed insert artifacts that generate new periodicities, the original frequency that acted as a fundamental is ultimately preserved. No matter how coarsely quantized, a sequence will still be repeating itself regardless of the discretization steps. Figure 2 explains this by showing an infinite precision signal, a down-quantized version and a "sign" signal quantized with 1 bit. With the period starting at 0, it can be shown that all signals re-start at the exact same moment, despite the quantization error being increasingly high.

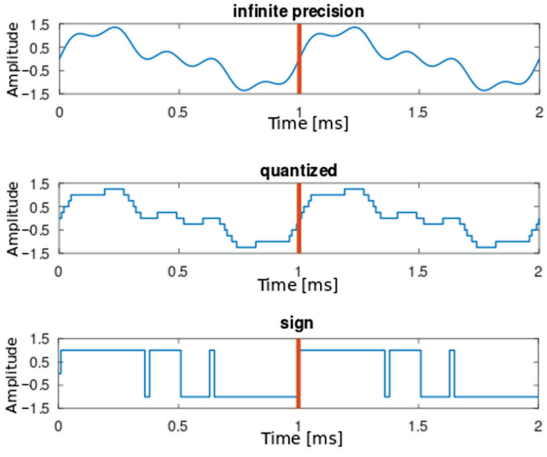


Figure 2. Period of a periodic signal and its quantized versions. The time at which the signal is measured to be repeating (period) is in orange. The "quantized" version is at N = 4 bits (16 values) and the "sign" version is quantized with 1 bit.

The mathematical principles for pitch detection are based on the intuition that the maximum of the autocorrelation [29] of a signal is when such a signal repeats itself, i.e., at its fundamental frequency.

The common definition for the autocorrelation of a signal x is [30]:

$$r_{xx}(l) = \frac{1}{\mathrm{N}} \cdot \sum_{k=1}^{N} x(k)x(k+l) \tag{2}$$

With l being the "lag" (progressive shift of a signal to have it slide on the other), N is the length of the signal in terms of the number of samples, and k is the discrete time (number of samples).

Implementing a full autocorrelation function has some drawbacks: it is relatively computationally expensive due to the need for reiterated multiplications, leading to a complexity of $O(N^2)$, and further normalization is required because the output is heavily dependent on the magnitude of the input signal and its variations, which skew the autocorrelation [20].

However, for a periodic signal, which can be defined as $x(k)$, which repeats after a period of T samples, so that $x(k) = x(k + T)$, a "difference function" can be defined so that its minimum corresponds to the period. The formula, expressed in terms of the lag l as the independent variable (in a digital sequence simply refers to the sample number), is the following:

$$d_{xx}(l) = \sum_{k=1}^{N} |x(k) - x(k+l)| \tag{3}$$

with l being the lag (sample), and N is the length of the signal x in samples. This formula will be referred to, for simplicity, as "modified autocorrelation", with the basic idea that instead of searching for the maximum of the product like pitch detection algorithms employing "usual" autocorrelation, we can search for the minimum of the difference. Moreover, this function is inherently independent of the input amplitude and its variations [31].

Figure 3 displays an example comparing the autocorrelation and the difference function ("modified autocorrelation"), showing how the maxima of the first roughly correspond to the minima of the latter.

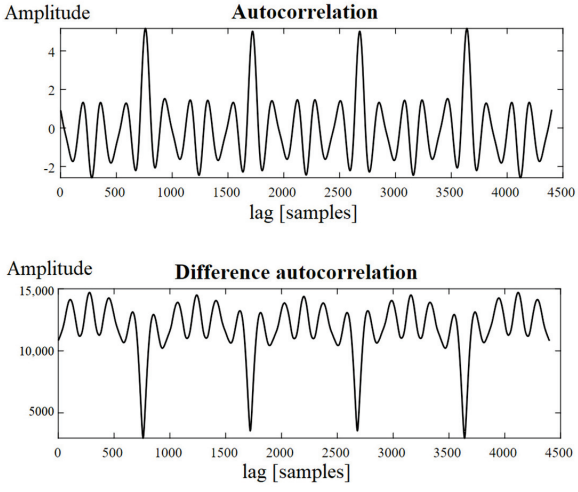


Figure 3. Comparison of autocorrelation versus the modified difference function version, computed on a signal from the "full2" category (F0 = 100 Hz). Notice how a maximum of the autocorrelation corresponds to a minimum in the difference function.

The operation to make the sum independent from negative values can be the square, as implemented by de Cheveignè and Kawahara [20], or the absolute, as we chose. In our specific case, this actually becomes irrelevant due to the 1-bit quantization, which essentially renders this formula into an XOR operation, which is inherently optimized for hardware implementations and parallelization.

However, for more than 1-bit quantization, the absolute is still more efficient, especially for FPGA implementations, because it can essentially be realized with a multiplexer for the MSB and a conditioned re-assign of the sign.

From the $O(N^2)$ computational complexity of the normal autocorrelation function, a single-bit XOR implementing the difference autocorrelation yields a theoretical constant complexity of O(1), being essentially a binary addiction without carry [32,33].

The idea behind the OBP algorithm is thus to apply an optimized autocorrelation-based pitch detection to a 1-bit version of the original signal, which drastically decreases computation complexity, especially considering that common bit depths employ 16 to 64 bits. Within this picture, Kawecka and Podahjecki explored the probabilistic properties of quantizers [34]. The operation of using 1 bit is logically equal to the "sign" operation. In the most common digital representations (two's complement, floating-point IEEE-754 [35], sign/magnitude [36]), the MSB (most significant bit) is used as the sign and thus a simple truncation of the original sequence is required—this could also be performed in the hardware domain for maximum speed.

The building blocks of the OBP algorithm are represented in Figure 4 and described as follows:

- Input: The input signal is assumed to be limited in band, with no DC components, which can be obtained before ever entering the digital realm by analog filtering; this also guarantees the zero average.
- Sign: From the original signal, only the sign is extracted; in any common digital environment, this is obtained by just retaining the MSB. For different kinds of representation, a comparator would be required.
- Buffer: It is needed to store a fixed amount of the previous samples of the signal and its length is related to the minimum frequency that needs to be detected (*Fmin*). With *Fs* being the sampling rate, a buffer length of *Fs/Fmin* samples is required.
- Modified autocorrelation: The difference autocorrelation previously defined is applied to the sign signal within the buffer. Instead of performing a circular correlation, like other algorithms (namely, YIN), a linear correlation is applied on a total frame length of one buffer and a half to avoid phase jumps. For periodic signals, autocorrelation tends to rise and fall from/to the minimum symmetrically: for this reason, the center of the minimum is to be considered to evaluate the period.
- Threshold: The search for the minimum of the difference autocorrelation function is simply performed by thresholding the signal with a fixed threshold. This can be visualized with a "Thresholding" logical signal that is 1 only when the autocorrelation is under the threshold. Although more sophisticated methods can be implemented, such a naïve implementation is computationally efficient and has been empirically observed to be a good trade-off between accuracy and speed.
- Output: As previously stated, the very minimum point is the center of the sections below the thresholds, which is equal to a logical 1 in the "Thresholding" signal. The frequency result is simply obtained by counting the number of samples between two minima.

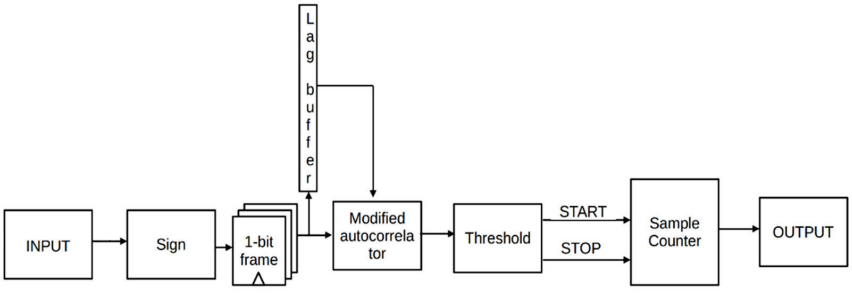


Figure 4. Block diagram of the OneBitPitch (OBP) pitch detection algorithm.

Figure 5 displays the progression of the algorithm by showing each internal signal.

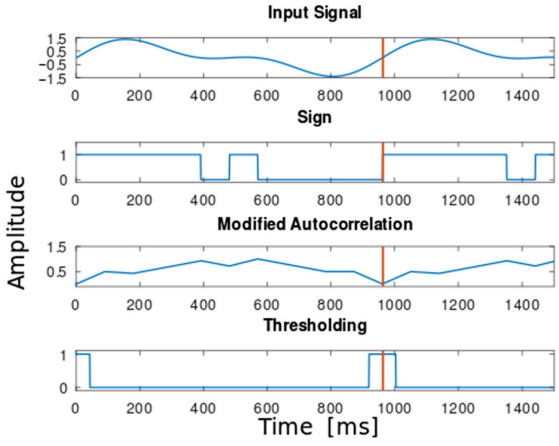


Figure 5. Processing steps within the OBP pipeline. The orange line shows the period, which is correctly detected as the center of the section below the threshold. The "Thresholding" signal is 1 when the "Modified Autocorrelation" signal goes below a fixed threshold (minimum search).

2.3. YIN Algorithm

The YIN algorithm [20] is an autocorrelation-based pitch estimator allegedly bringing high speed and good accuracy with few parameters to tune. It utilizes a modified autocorrelation difference function called Cumulative Mean Normalized Difference Function, used iteratively to avoid zero lag and to normalize the difference function with respect to large lags. Parabolic interpolation is then employed to obtain sub-sampling resolutions. In order to reduce octave errors and optimize the result, the search for the pitch candidate is aided by a heuristic based on range reduction. As for many algorithms, additional tuning possibilities are forecasted for unclean signals (i.e., presence of additive noise or additive frequencies/harmonics), such as comb filtering.

2.4. SWIPE (Sawtooth-Inspired Pitch Estimator) Algorithm

SWIPE was developed by Camacho and Harris [19]. It consists of measuring the average distance between valleys and peaks on the spectrum, at harmonics of the pitch. After the first estimation, SWIPE tries to refine the output by exploiting a variable window size and evaluating the best pitch candidate.

The pitch is estimated by comparing the spectrum of the signal to the sawtooth waveform whose spectrum is most similar. This is achieved by calculating a normalized inner product between the signal spectrum and a modified cosine. The analysis window size is adjusted to align the main lobes of the spectrum with the positive lobes of the cosine and parabolic interpolation is employed for added accuracy. SWIPE' (or SWIPE prime or SWIPEP) is a variant of SWIPE built to minimize subharmonic errors, which the original algorithm was prone to, by only employing the first harmonic and the prime ones. It is the most widespread version, present in advanced libraries such as Tsanas' Voice Analysis Toolbox [37], and will thus be the one used in this paper.

2.5. NLS (Nonlinear Least Squares) Algorithm

The Nonlinear Least Squares (NLS) principle is based on a statistical Maximum Likelihood (ML) candidate research [38,39]. This class of algorithms can theoretically achieve the highest degree of accuracy, especially on discretized pitches, at the cost of greater computational complexity.

The algorithm proceeds by iteratively minimizing the cost function for the estimation error.

The algorithm starts by selecting a range of pitch candidates that cover the expected range of the fundamental frequency: this operation requires a prior setup of the algorithm,

which adds latency before being usable. For each candidate, a synthesized signal is generated with harmonics or sinusoids at that pitch frequency. The objective is to find the pitch candidate that best matches the observed signal by minimizing the sum of squared differences between the observed and synthesized signals. Optimization techniques like gradient descent are used to update the pitch candidates iteratively until convergence. Finally, the pitch candidate with the minimum objective function value is selected as the estimated fundamental frequency or pitch of the signal.

The NLS algorithm is known for its ability to handle complex harmonic structures, variable pitch signals, and noisy environments. It is particularly effective when the signal contains multiple overlapping or interfering harmonic components.

For the purpose of this study, a fast implementation of an NLS-based algorithm by Wang et al. [40] is employed, shown to reduce the complexity by solving two Toeplitz-plus-Hankel systems of equations and using the recursive-in-order matrix structures.

2.6. Test Conditions

All the algorithms were tested on the same set of signals from the SYNTHPITCH dataset. The sampling rate was 96 KHz, the considered duration of the signals was 100 ms (9600 samples) and the frequency limits for algorithms that require it (e.g., YIN needs to set the minimum detectable frequency) were between 50 and 5000 Hz, in order to also observe eventual subharmonic errors on even the lowest of the frequencies in the dataset (which is 100 Hz). This resulted in buffer lengths of $Fs/Fmin$ = 1920 samples. The following hyperparameters and setup were employed for each algorithm under test:

- OBP: Our algorithm is used with a fixed threshold at 400. Tuning the threshold, predictably, allows us to adapt to different input classes or characteristics.
- YIN: The resolution for the parabolic interpolation is 1 cent, the fixed threshold is at 0.1.
- SWIPE: The resolution for the parabolic interpolation is 1 cent, the harmonics considered are only the first and the prime ones (making it SWIPEP), the timeframe is at 10 ms and the final result is the mean of the tracked frequencies. Using a bigger or smaller timeframe does not sensibly change the elapsed time due to the inherent nature of SWIPE and it being reliant on generating sawtooth waves.
- NLS: An NLS-based model is generated from a synth sample having a number of harmonics equal to 4. Increasing the number of harmonics, in the case of the present study, leads to sensibly higher elapsed times while not improving accuracy. The time needed to generate the model, i.e., the unavoidable latency at the beginning, is on average around 40 ms. For simplicity and uniformity, this latency will not be considered when discussing elapsed times, with the assumption that in a real-world scenario, the model is pre-made. However, this is a small disadvantage.

The test involves running all of the algorithms on each signal (duration = 100 ms) in each category of the SYNTHPITCH dataset. The main metrics are time elapsed in seconds (TE) and relative absolute error (RAE) computed by comparing the estimated frequency with the known F0 of each signal according to the following formula:

$$RAE = \left| \frac{y_E - y}{y} \right| \tag{4}$$

with y_E being the estimated value and y being the real/target one.

Both the RAE and the TE are averaged for each category. Due to the real-world applications of pitch detection algorithms, and also taking into account the fact that real-world signals might not present a discrete F0, the accuracy within a certain range is also presented. Specifically, the ranges 1%, 2% and 10% of the true F0 are considered, producing the metrics ACC-1, ACC-2 and ACC-10. These accuracies are presented as averages over categories as well.

The ranges were chosen empirically, with 2% being a truly "acceptable" error in most music/MIDI-related applications, and with 10% ruling out octave errors, which inherently produce 100% or above errors. Acceptable RAE values can be approximately below 0.025, because most musical applications use discretized pitches that do not result in note errors if within a range < 2.5% around the starting pitch. We chose 2% as a safety measure: this range is well represented by the "ACC-2" metric, which is the percentage of instances in which the algorithm brings an error equal to or lower than 2%. Section 4 will further detail this.

In order to assess the statistical validity of the results, a Mann–Whitney U-test was performed on each table reporting RAE, TE and ACC metrics within the results [41]. With OBP as the main algorithm under test, U, Pearson's p (or "rho") and the z-score were derived for each one-vs.-one comparison, with $p < 0.05$ used as the significant threshold [42].

3. Results

Tables 2 and 3 show the performances of each algorithm in terms of time elapsed (TE) and relative absolute error (RAE). Tables 4–6 detail the accuracy with acceptance rates of 1% (ACC-1), 2% (ACC-2) and 10% (ACC-10). Table 7 presents the results of the statistical analysis.

Table 2. RAE (relative absolute error) of each algorithm averaged over each category of the SYNT-PITCH dataset, along with the mean for each algorithm over the whole dataset.

RAE (Average)				
Dataset Category	**Algorithm**			
	YIN	**SWIPE**	**NLS**	**OBP**
2harm	0.003157	0.002998	0.000000	0.013210
2harm_wgn15	0.003071	0.002738	0.000000	0.014091
4harm	0.001324	0.002568	0.000000	0.052182
4harm_4part_wgn15	0.420658	0.304243	0.000000	0.154955
4harm_high	0.005729	0.004558	0.069865	0.985475
4harm_wgn15	0.001185	0.002136	0.000000	0.014095
full1	0.396471	0.252746	0.473064	0.201402
full2	0.254010	0.210246	0.483956	0.369483
pure	0.007338	0.004879	0.000000	0.013462
pure_wgn0P3	0.008713	0.019694	0.000000	0.012983
square_pure	0.334296	0.011628	0.000000	0.013462
square_wgn10	0.262356	0.013746	0.000000	0.013683
MEAN	0.141526	0.069348	0.085574	0.154873

Table 3. TE (time elapsed) in seconds for each algorithm averaged over each category of the SYNT-PITCH dataset, along with the mean for each algorithm over the whole dataset.

Time Elapsed (TE, Average)				
Dataset Category	**Algorithm**			
	YIN	**SWIPE**	**NLS**	**OBP**
2harm	0.040090	0.266015	0.025799	0.004731
2harm_wgn15	0.033833	0.248275	0.025342	0.005095
4harm	0.033381	0.251229	0.026266	0.004853
4harm_4part_wgn15	0.033516	0.253206	0.025506	0.004677
4harm_high	0.035921	0.245899	0.025181	0.004831
4harm_wgn15	0.035036	0.271767	0.025841	0.005500
full1	0.036627	0.267752	0.031347	0.004833
full2	0.035664	0.254639	0.024700	0.004661
pure	0.035881	0.247622	0.025245	0.005238
pure_wgn0P3	0.035488	0.248970	0.025257	0.004766
square_pure	0.035509	0.246896	0.024810	0.004741
square_wgn10	0.035650	0.246456	0.028136	0.004685
MEAN	0.035550	0.254061	0.026119	0.004884

Table 4. ACC-1: percentage of the times each algorithm has provided an estimated frequency that brings RAE < 0.01. Averaged over each category of the SYNTHPITCH dataset, along with the mean for each algorithm over the whole dataset.

ACC-1				
Dataset Category	**Algorithm**			
	YIN	**SWIPE**	**NLS**	**OBP**
2harm	0.949495	0.989899	1.000000	0.474747
2harm_wgn15	0.969697	0.979798	1.000000	0.393939
4harm	0.989899	0.989899	1.000000	0.434343
4harm_4part_wgn15	0.414141	0.242424	1.000000	0.303030
4harm_high	0.989899	0.989899	0.939394	0.303030
4harm_wgn15	1.000000	0.989899	1.000000	0.424242
full1	0.434343	0.191919	0.595960	0.414141
full2	0.575758	0.262626	0.646465	0.474747
pure	0.777778	0.868687	1.000000	0.444444
pure_wgn0P3	0.979798	0.606061	1.000000	0.484848
square_pure	0.626263	0.939394	1.000000	0.444444
square_wgn10	0.606061	0.858586	1.000000	0.424242
MEAN	0.776094	0.742424	0.931818	0.418350

Table 5. ACC-2: percentage of the times each algorithm has provided an estimated frequency that brings RAE < 0.02. Averaged over each category of the SYNTHPITCH dataset, along with the mean for each algorithm over the whole dataset.

ACC-2				
Dataset Category	**Algorithm**			
	YIN	**SWIPE**	**NLS**	**OBP**
2harm	1.000000	1.000000	1.000000	0.818182
2harm_wgn15	1.000000	1.000000	1.000000	0.777778
4harm	1.000000	1.000000	1.000000	0.737374
4harm_4part_wgn15	0.414141	0.333333	1.000000	0.545455
4harm_high	0.989899	1.000000	0.939394	0.464646
4harm_wgn15	1.000000	1.000000	1.000000	0.757576
full1	0.434343	0.191919	0.595960	0.575758
full2	0.575758	0.272727	0.646465	0.717172
pure	1.000000	1.000000	1.000000	0.818182
pure_wgn0P3	0.989899	0.797980	1.000000	0.818182
square_pure	0.676768	0.949495	1.000000	0.818182
square_wgn10	0.606061	0.858586	1.000000	0.797980
MEAN	0.807239	0.783670	0.931818	0.720539

Table 6. ACC-10: percentage of the times each algorithm has provided an estimated frequency that brings RAE < 0.10. Averaged over each category of the SYNTHPITCH dataset, along with the mean for each algorithm over the whole dataset.

ACC-10				
Dataset Category	**Algorithm**			
	YIN	**SWIPE**	**NLS**	**OBP**
2harm	1.000000	1.000000	1.000000	1.000000
2harm_wgn15	1.000000	1.000000	1.000000	1.000000
4harm	1.000000	1.000000	1.000000	0.979798
4harm_4part_wgn15	0.414141	0.484848	1.000000	0.757576
4harm_high	0.989899	1.000000	0.939394	0.585859
4harm_wgn15	1.000000	1.000000	1.000000	1.000000
full1	0.434343	0.383838	0.595960	0.696970
full2	0.575758	0.444444	0.646465	0.828283
pure	1.000000	1.000000	1.000000	1.000000
pure_wgn0P3	0.989899	0.989899	1.000000	1.000000
square_pure	0.727273	0.979798	1.000000	1.000000
square_wgn10	0.606061	0.979798	1.000000	1.000000
MEAN	0.811448	0.855219	0.931818	0.904040

Table 7. Results of the Mann–Whitney statistical test for the relevance of each metric for each couple of algorithms. Comparisons involving OBP are in bold. A z-score < −4 associated with $p < 0.000001$ is related to a confidence higher than 99.997%.

Comparison	Statistical Measures		
	U	**Pearson *p***	**z-Score**
RAE (OBP vs. SWIPE)	42	0.08914	1.70318
RAE (OBP vs. NLS)	30	0.0164	2.396
RAE (OBP vs. YIN)	52	0.25848	1.12583
RAE (YIN vs. SWIPE)	60	0.50926	0.66395
RAE (YIN vs. NLS)	31	0.01928	2.33827
RAE (SWIPE vs. NLS)	33	0.02642	2.2228
TE (OBP vs. SWIPE)	0	<0.000001	<−4
TE (OBP vs. NLS)	0	<0.000001	<−4
TE (OBP vs. YIN)	0	<0.000001	<−4
TE (YIN vs. SWIPE)	0	<0.000001	<−4
TE (YIN vs. NLS)	0	<0.000001	<−4
TE (SWIPE vs. NLS)	0	<0.000001	<−4
ACC-1 (OBP vs. SWIPE)	36	0.04036	−2.04959
ACC-1 (OBP vs. NLS)	0	<0.000001	<−4
ACC-1 (OBP vs. YIN)	14	0.0009	−3.31976
ACC-1 (YIN vs. SWIPE)	70	0.92828	0.0866
ACC-1 (YIN vs. NLS)	26.5	0.00932	−2.59808
ACC-1 (SWIPE vs. NLS)	22.5	0. 00466	−2.82902
ACC-2 (OBP vs. SWIPE)	40.5	0.07346	−1.78979
ACC-2 (OBP vs. NLS)	18	0.002	−3.08882
ACC-2 (OBP vs. YIN)	51.5	0.25014	−1.1547
ACC-2 (YIN vs. SWIPE)	72	0.97606	0.02887
ACC-2 (YIN vs. NLS)	46.5	0.14986	−1.44338
ACC-2 (SWIPE vs. NLS)	52	0.25848	−1.12583
ACC-10 (OBP vs. SWIPE)	65	0.70394	0.37528
ACC-10 (OBP vs. NLS)	61.5	0.56192	−0.57735
ACC-10 (OBP vs. YIN)	55.5	0.35758	0.92376
ACC-10 (YIN vs. SWIPE)	67	0.79486	−0.25981
ACC-10 (YIN vs. NLS)	46.5	0.14986	−1.44338
ACC-10 (SWIPE vs. NLS)	54	0.3125	−1.01036

Figure 6 shows a sample of the RAE plotted for each signal within two categories of the dataset, as an RAE vs. frequency plot, where it can be appreciated how OBP performs in a solid way throughout all the categories, to the point where, for complex signals like those included in "full2", it actually brings lower errors than most of the other algorithms; it can also be observed how most of the errors in NLS are octave errors (integer RAE). The exemplified categories are "2harm" and "full2" in order to show the behavior of the four algorithms on clean vs. unclean signals.

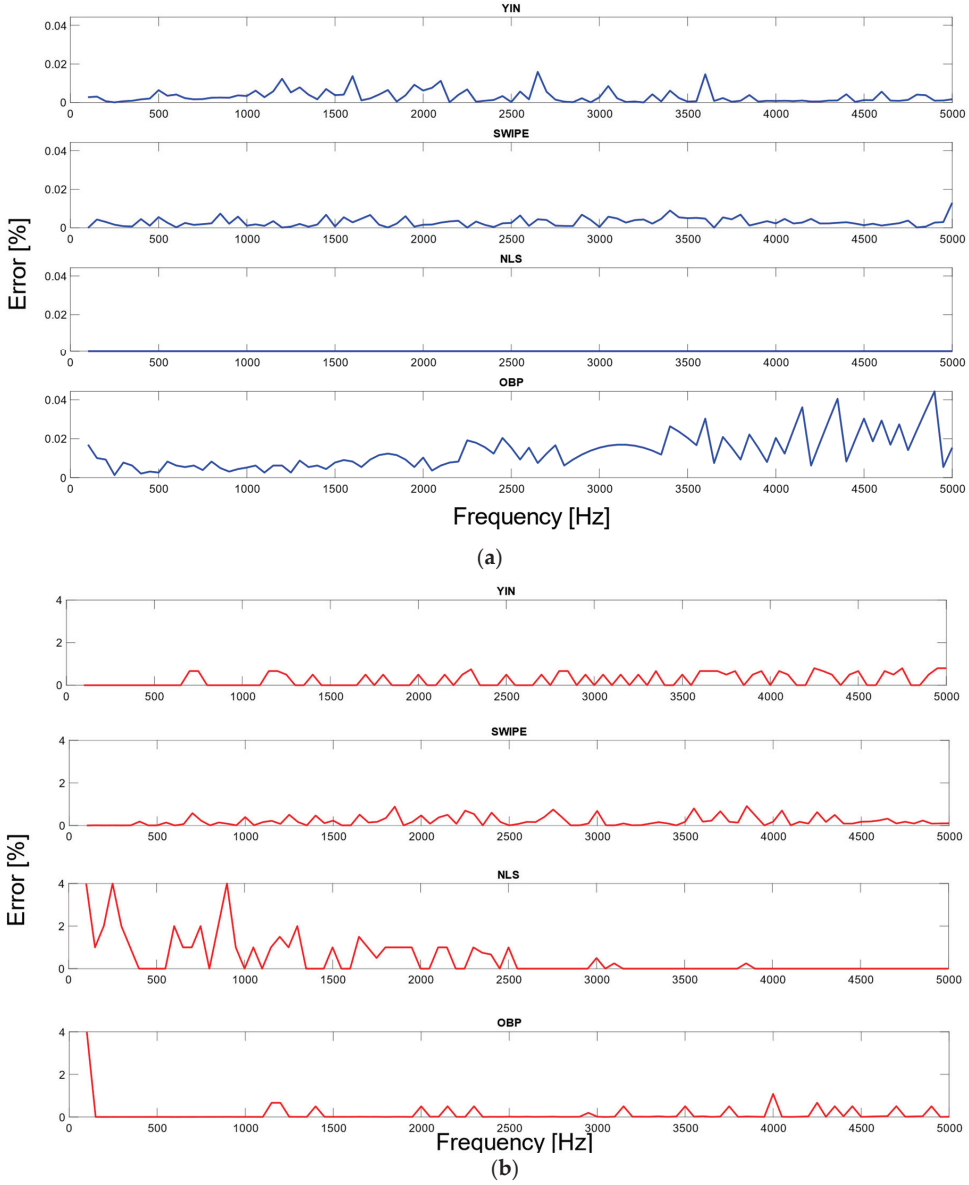


Figure 6. Example of RAE (percentage error) vs. frequency plots for two categories (a very clean one vs. a very complex one) of the SYNTHPITCH dataset. Frequencies span from 100 Hz to 5000 Hz with a step size of 50 Hz. (**a**) "2harm" category (blue); (**b**) "full2" category (red).

All the analyses, data creation, tests, simulations and algorithm implementations were performed using MATLAB® R2023a (by Mathworks Inc., Natick, MA, USA [43]) on a Dell Latitude E5550 computer, with an Intel Core i5 5200U processor and a 16 GB Dual-Channel DDR3 RAM.

Figure 6 details an example of the percentage error (RAE) with respect to the frequency, i.e., the different signals within a category of the SYNTPITCH dataset.

The time elapsed (TE) was empirically shown to be independent of the frequency of the input; plots are thus not shown as they are erratic and only depend on the randomness within the signals, given for example by the white Gaussian noise. TE was also shown to be independent of the dataset category, i.e., from the pitch complexity of the input signal.

4. Discussion

The premises of this study, besides the presentation of the custom-made SYNTHPITCH dataset, were to implement an ultra-fast pitch detection algorithm for real-time applications and ease of hardware implementation. Our proposed algorithm, OneBitPitch (OBP), despite running on a software environment, confirms the premises by providing by far the fastest results in terms of time elapsed for pitch detection. On the other hand, as is expected due to its nature, SWIPE is the heaviest algorithm and takes an average of 256 ms to be executed, despite the "large" window frame selected and the SWIPEP variant.

YIN, considered one of the fastest pitch detection algorithms, being based on a modified autocorrelation, was about nine times faster than SWIPE, with only a 38 ms average. NLS needs a special mention, as its accuracy values are outstanding, and, on synthetic signals, its time performances are too, with only a 27 ms average.

However, OBP shows its real strength with an average elaboration time of only 4.6 ms. It is 50 times faster than SWIPE and even 9 times faster than YIN, which is considered a fast algorithm. Figure 7 displays an alternative visualization of the performance of the four algorithms in terms of speed, plotting the elapsed time, which immediately shows the differences in speed.

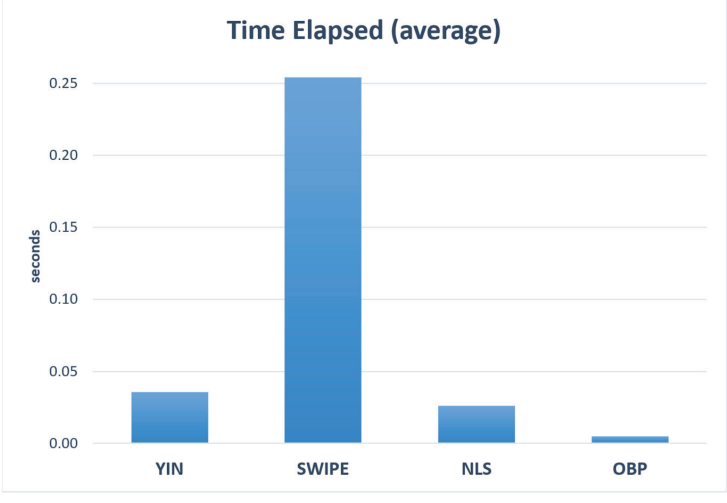


Figure 7. Average elapsed time (in seconds) for each algorithm. See Table 3 for numeric data.

Other works such as that by Grinewitschus et al. [21] report speed as a multiplier of the "realtime", which is the duration of the audio segment to be evaluated. Their approach, based on leveraging the constant-Q Gabor Transform followed by harmonic shift algorithms and corrective heuristics, reaches a declared 0.29 × realtime speed on a more powerful machine than what has been used in the present work; nonetheless, OBP reaches 0.046 × realtime.

The choice of the right algorithm for a specific application is mainly based on the latency and accuracy tradeoff required, and the computational power available, so it is safe to say that each of the proposed algorithms has a specific field and reason to be applied.

Looking at the RAE and accuracies, the NLS appears as the best-performing algorithm, always providing less than 1% RAE on average, followed by SWIPE. It is, however, worth

noting that the NLS was one of the algorithms suffering the most from octave errors, especially when processing noisy signals or square waves. Moreover, this test confirms the noise-robust nature of SWIPE, performing well even with harsh signals like square waves and/or noise.

Looking at the accuracy tables for ACC-1, ACC-2 and ACC-10, NLS confirms its performances by always providing the highest accuracy, at 93%. However, this value stays the same for all three ranges, which indicates that all the errors reported by NLS within these tests were greater than 10% in RAE, which is a good indicator of a certain proneness to octave errors.

Although OBP underperforms when it comes to 1% accuracy, that is to be expected: only 42% of the time was the error smaller than 1%. On the other hand, its performances become comparable to YIN and SWIP for ACC-2, which represents the acceptability range. The discrete nature of musical notes within the Western tempered system is so that a 2.5% error still leads to a discretized pitch being correct.

For larger-scale errors, OBP actually brings better ACC-10 values than YIN and SWIPE, with more than 90% accuracy.

TE can be considered as the most crucial metric for the present paper, due to the improvement that OBP aims to bring, which is specifically in terms of speed.

Within this context, the differences between all of the algorithms considered, especially OBP, have been proven to be statistically significant with a Mann–Whitney test yielding a U close to 0, which brings $p < 0.00001$, which is much lower than the desired threshold of 0.05. A z-score < -4 points to a confidence higher than 99.997%.

RAE and ACC-1 also generally present statistical significance in the comparisons, especially when OBP is involved. On the other hand, results regarding ACC-10 are not statistically significant (all of the algorithms perform similarly).

In general, differences between YIN and SWIPE within this picture appear to bear less statistical significance, being only significant for the TE—which is the most crucial metric for the scope of this study.

These results in terms of time vs. accuracy are completely in line with the premises, since OBP was designed to be minimalistic, forgoing sheer, pinpointed accuracy (hence the lower ACC-1 values) but still staying within an acceptable range for most applications (hence the higher ACC-2 values), with an inherent observed robustness with respect to signal variations and to octave errors.

It is worth noting that despite the average accuracy, the average RAE of OBP is comparable to the YIN algorithm.

Moreover, all the presented algorithms have been used in one fixed setup with a fixed threshold. However, in real-world applications, hyperparameters can be tuned to better adapt to the nature and variation of the input: in fact, manually changing the threshold of the OBP algorithm provides better performances on certain datasets, especially the noisiest.

The noisiest, most unclean dataset categories are full1, full2, 4harm_4part_wgn15 and 4harm_high: on these sets, most algorithms provide poor performances. However, the average RAEs that would be obtained on all of the other sets are as follows:

- YIN: 0.077680, i.e., 7.7% average error;
- SWIPE: 0.007548, i.e., 0.7% average error;
- NLS: 0, i.e., no errors. Due to its heuristic-powered nature, NLS is able to pinpoint discrete frequencies on synthetic datasets;
- OBP: 0.018396, i.e., 1.8% average error.

On cleaner signals, such as those produced by many musical instruments that do not have added noise, OBP actually performs better in terms of error than YIN, while providing more than acceptable results overall.

The minimalistic, stripped-down nature of OBP does not allow it to reach almost-perfect accuracy levels; however, it is critical to assess its "acceptability" ranges, given that it is by far the fastest [44].

YIN, on the other hand, is a fast and simple algorithm. It suffered from octave errors 12% of the time, but when the estimation was right, the precision was satisfying, with an 87% chance of obtaining a result with less than 1% error.

The OBP algorithm presents a peculiar behavior because it has only a 4% chance of suffering from octave errors.

An advantage of the mathematical model behind OBP over other similar autocorrelation-based methods is that, being based on the product of signals, the output heavily depends on the magnitude of the input and its variations. Because of that, an additional normalization process is usually needed, increasing the computational complexity, requiring knowledge of the energy of the signals and adding more multipliers.

On the other hand, OBP only processes sign signals, inherently independent from the original magnitude, removing the need for explicit normalization procedures.

Despite the performances reported in these tests, different environments and data see the algorithms behave differently, at least in terms of accuracy. In fact, SWIPE is one of the gold standards for feature extraction for medical or Machine Learning purposes, due to it being well suited for noisy environments and due to the applications not needing real-time elaboration. On the other hand, NLS is not robust with respect to the nature of the input and is more prone to suffer from octave errors, making it sometimes an inconsistent choice despite the high performances on the proposed synthetic dataset.

This study employed the SYNTHPITCH dataset because the large-scale speed was the main indicator to be evaluated; however, future steps will evolve around the experimentation on real-world, validated data for pitch detection, which also leads to the application of pitch-tracking, to follow in real-time the varying pitch of a real sound/speech signal [45].

Future Works

We are currently working on expanding the present experimentation with other test datasets, focusing especially on real-world scenarios such as sounds from real instruments (as those employed for MIDI conversion) or vocal signals.

Technically speaking, the most likely future implementations of OBP will definitely focus on hardware implementation, due to its bitwise nature and maximum speed. Its simple structure and the low usage of memory and computations make it perfectly suitable for a hardware-only implementation—for example, with a DSP—to exploit its speed capabilities and inherent characteristics suitable for digital electronics. A future FPGA implementation is foreseen, aimed to produce a stand-alone IpCore that, using just a small amount of logic, can provide ultra-low-latency pitch tracking [46,47]. With the increasing trend and the low cost/low performance of System on a Chip (SoC) technologies, the OBP algorithm can be a perfect candidate for wearable electronic, embedded music processors for Autotune or real-time MIDI conversion [48], as well as IoT applications, surveillance or vocal recognition. We can thus summarize some of the future directions of OBP as follows:

- Test on real-world datasets, especially within the professional audio/musical department;
- Full-hardware FPGA implementation;
- Addition of optional features such as posterior (heuristic) correction for added accuracy at the cost of speed, selectable N-bit expansion, etc.

5. Conclusions

In this paper, a novel algorithm was proposed for ultra-fast pitch detection for real-time applications, based on a modified autocorrelation implemented on a single-bit signal. The OneBitPitch (OBP) algorithm was compared with the most widely used models for high-speed F0 detection, namely, YIN, SWIPE and an NLS-based implementation. Additionally, a custom dataset made of synthetic waves has been proposed and made available to the public: the SYNTHPITCH dataset encompasses sinusoidal and square waves with added artifacts for an increasing pitch complexity, realized through the addition of harmonics, partials, white noise and reverb.

OBP is shown to be an ultra-fast, reliable pitch detection algorithm for real-time applications: it has been built with a minimalist approach that aims to reduce all computationally expensive steps in pitch detection, which are mainly represented by a multi-bit elaboration, the transformation/metric production and the eventual presence of corrective heuristics.

The focus of the OBP algorithm is solely on speed, provided that acceptable accuracy levels are reached, and its characteristics make it exceptionally suitable for an easy and lightweight hardware implementation on FPGA or SoC, for ever-higher customization possibilities as well as lower latencies.

The comparison of different state-of-the-art algorithms shows that OBP is 9 times faster than the peak speed of the other algorithms (namely, YIN) and 50 times faster than SWIPE. OBP is shown to be the fastest pitch detection algorithm within the presented test, with only 4.6 ms of mean elapsed time on each data instance, or $0.046 \times$ realtime runtime, which is the lowest reported to date.

On the other hand, although relative error might be increased for certain datasets, OBP is less prone to octave errors, and in general demonstrates the ability to stay within the acceptability range on most of the tested signals. The main compromise was between speed and accuracy, and OBP stays within acceptable ranges of 2% accuracy, with a 72% average that peaks at almost 82% for less noisy signals, which is enough for most discrete-note musical applications. The NLS-based algorithm is the most accurate one, although it is less robust to noise or input variations and requires a prior building of a model; in fact, SWIPE is one of the most employed algorithms in real-world applications not centered on speed. Additional tests are needed on real-world signals, such as validated voices of known F0, and a hardware implementation, with more parameters and selectable options is foreseen for the OBP algorithm.

Author Contributions: Conceptualization, D.C.; methodology, D.C. and V.C.; software, D.C. and V.C.; validation, V.C. and G.C.; formal analysis, V.C.; investigation, D.C. and V.C.; resources, V.C.; data curation, V.C.; writing—original draft preparation, D.C.; writing—review and editing, V.C.; visualization, D.C. and V.C.; supervision, G.C.; project administration, G.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The proposed dataset SYNTHPITCH of synthetic signals for testing pitch detection algorithms is available at the following link: https://drive.google.com/drive/folders/1YP15ULjyyeI27k_2wPzWFxuig7fbSveb?usp=sharing (accessed on 10 July 2023).

Acknowledgments: The authors would like to thank Voicewise S.r.l. for the support.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ruslan, N.; Mamat, M.; Porle, R.; Parimon, N. A Comparative Study of Pitch Detection Algorithms for Microcontroller Based Voice Pitch Detector. *Adv. Sci. Lett.* **2017**, *23*, 11521–11524. [CrossRef]
2. Qurthobi, A.; Maskeliūnas, R.; Damaševičius, R. Detection of Mechanical Failures in Industrial Machines Using Overlapping Acoustic Anomalies: A Systematic Literature Review. *Sensors* **2022**, *22*, 10. [CrossRef] [PubMed]
3. Kim, J.; Kim, J.; Nguyen, L.T.; Shim, B.; Hong, W. Tonal signal detection in passive sonar systems using atomic norm minimization. *EURASIP J. Adv. Signal Process.* **2019**, *2019*, 43. [CrossRef]
4. Krajewski, M.; Sienkowski, S.; Kawecka, E. Properties of selected frequency estimation algorithms in accurate sinusoidal voltage measurements. *Prz. Elektrotechniczny* **2018**, *94*, 52–55.
5. Teixeira, J.P.; Oliveira, C.; Lopes, C. Vocal Acoustic Analysis—Jitter, Shimmer and HNR Parameters. *Procedia Technol.* **2013**, *9*, 1112–1122. [CrossRef]
6. Bharathi, V.; Abraham, A.; Ramya, R. Vocal pitch detection for musical transcription. In Proceedings of the 2011 International Conference on Signal Processing, Communication, Computing and Networking Technologies, Thuckalay, India, 21–22 July 2011. [CrossRef]
7. Hildebrand, H.A. Pitch Detection and Intonation Correction Apparatus and Method. U.S. Patent No. 5,973,252, 26 October 1999.

8. Costantini, G.; Cesarini, V.; Robotti, C.; Benazzo, M.; Pietrantonio, F.; Di Girolamo, S.; Pisani, A.; Canzi, P.; Mauramati, S.; Bertino, G.; et al. Deep learning and machine learning-based voice analysis for the detection of COVID-19: A proposal and comparison of architectures. *Knowl. Based Syst.* **2022**, *253*, 109539. [CrossRef] [PubMed]
9. Cesarini, V.; Robotti, C.; Piromalli, Y.; Mozzanica, F.; Schindler, A.; Saggio, G.; Costantini, G. Machine Learning-based Study of Dysphonic Voices for the Identification and Differentiation of Vocal Cord Paralysis and Vocal Nodules. In Proceedings of the 15th International Conference on Bio-inspired Systems and Signal Processing, Online, 9–11 February 2022; pp. 265–272. [CrossRef]
10. Costantini, G.; Parada-Cabaleiro, E.; Casali, D.; Cesarini, V. The Emotion Probe: On the Universality of Cross-Linguistic and Cross-Gender Speech Emotion Recognition via Machine Learning. *Sensors* **2022**, *22*, 7. [CrossRef]
11. Costantini, G.; Cesarini, V.; Di Leo, P.; Amato, F.; Suppa, A.; Asci, F.; Pisani, A.; Calculli, A.; Saggio, G. Artificial Intelligence-Based Voice Assessment of Patients with Parkinson's Disease Off and On Treatment: Machine vs. Deep-Learning Comparison. *Sensors* **2023**, *23*, 4. [CrossRef]
12. Amato, F.; Saggio, G.; Cesarini, V.; Olmo, G.; Costantini, G. Machine learning- and statistical-based voice analysis of Parkinson's disease patients: A survey. *Expert Syst. Appl.* **2023**, *219*, 119651. [CrossRef]
13. Fant, G. *Acoustic Theory of Speech Production*; Walter de Gruyter: Berlin, Germany, 1970.
14. Hermes, D. Measurement of pitch by subharmonic summation. *J. Acoust. Soc. Am.* **1988**, *83*, 257–264. [CrossRef]
15. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2012; Available online: https://proceedings.neurips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html (accessed on 28 February 2023).
16. Illner, V.; Sovka, P.; Rusz, J. Validation of freely-available pitch detection algorithms across various noise levels in assessing speech captured by smartphone in Parkinson's disease. *Biomed. Signal Process. Control.* **2020**, *58*, 101831. [CrossRef]
17. Su, H.; Zhang, H.; Zhang, X.; Gao, G. Convolutional neural network for robust pitch determination. In Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016. [CrossRef]
18. Khadem-hosseini, M.; Ghaemmaghami, S.; Abtahi, A.; Gazor, S.; Marvasti, F. Error Correction in Pitch Detection Using a Deep Learning Based Classification. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2020**, *28*, 990–999. [CrossRef]
19. Camacho, A.; Harris, J.G. A sawtooth waveform inspired pitch estimator for speech and music. *J. Acoust. Soc. Am.* **2008**, *124*, 1638–1652. [CrossRef]
20. De Cheveigne, A.; Kawahara, H. YIN, a fundamental frequency estimator for speech and music. *J. Acoust. Soc. Am.* **2002**, *111*, 4. [CrossRef]
21. Grinewitschus, L.; Jung, P. The Harmonic Shift Algorithm for Efficient Multi-Pitch Detection. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2022**, *30*, 548–561. [CrossRef]
22. Mnasri, Z.; Rovetta, S.; Masulli, F. A Novel Pitch Detection Algorithm Based on Instantaneous Frequency. In Proceedings of the 2021 29th European Signal Processing Conference (EUSIPCO), Dublin, Ireland, 23–27 August 2021; pp. 16–20. [CrossRef]
23. Zahorian, S.; Dikshit, P.; Hu, H. A spectral-temporal method for pitch tracking. In Proceedings of the Ninth International Conference on Spoken Language Processing, Pittsburgh, PN, USA, 17–21 September 2006. [CrossRef]
24. Staudacher, M.; Steixner, V.; Griessner, A.; Zierhofer, C. Fast fundamental frequency determination via adaptive autocorrelation. *EURASIP J. Audio Speech Music. Process.* **2016**, *2016*, 17. [CrossRef]
25. Kim, J.; Salamon, J.; Li, P.; Bello, J. CREPE: A Convolutional Representation for Pitch Estimation. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018.
26. Kay, S.M. *Fundamentals of Statistical Signal Processing*; Prentice Hall Signal Processing Series; Prentice-Hall PTR: Englewood Cliffs, NJ, USA, 1993.
27. Kehtarnavaz, N. Analog-to-Digital Signal Conversion. In *Digital Signal Processing System Design*; Elsevier: Amsterdam, The Netherlands, 2008; pp. 57–91. [CrossRef]
28. Host-Madsen, A.; Handel, P. Effects of sampling and quantization on single-tone frequency estimation. *IEEE Trans. Signal Process.* **2000**, *48*, 650–662. [CrossRef]
29. Apicella, B.; Bruno, A.; Wang, X.; Spinelli, N. Fast Fourier Transform and autocorrelation function for the analysis of complex mass spectra. *Int. J. Mass Spectrom.* **2013**, *338*, 30–38. [CrossRef]
30. Ortigueira, M. On the estimation of the autocrrelation function. *Discuss. Mathematicae. Probab. Stat.* **2010**, *30*, 103–115. [CrossRef]
31. Hess, W. *Pitch Determination of Speech Signals*; Springer Series in Information Sciences; Springer: Berlin/Heidelberg, Germany, 1983; Volume 3. [CrossRef]
32. Granlund, T. Instruction Latencies and Throughput for AMD and Intel x86 Processors 2019. Online x86-Timing.pdf. Available online: https://gmplib.org/ (accessed on 20 June 2023).
33. Dodmane, R.; Aithal, G.; Shetty, S. Construction of vector space and its application to facilitate bitwise XOR—Free operation to minimize the time complexity. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 9836–9843. [CrossRef]
34. Kawecka, E.; Podhajecki, J. Probabilistic Properties of Deterministic and Randomized Quantizers. *Procedia Comput. Sci.* **2022**, *207*, 754–768. [CrossRef]
35. *IEEE Std 754-2019 (Revision of IEEE 754-2008)*; IEEE Standard for Floating-Point Arithmetic. IEEE: New York, NY, USA, 2019; pp. 1–84. [CrossRef]
36. Samavi, S. *Representing Signed Numbers*; McMaster University: Hamilton, ON, Canada, 2014.

37. Tsanas, A.; Little, M.; Mcsharry, P.; Ramig, L. New nonlinear markers and insights into speech signal degradation for effective tracking of Parkinson's disease symptom severity. In Proceedings of the International Symposium on Nonlinear Theory and Its Applications (NOLTA), Krakow, Poland, 5–8 September 2010; pp. 457–460.
38. Teunissen, P. Nonlinear least-squares. *Manuscripta Geod.* **1990**, *15*, 137–150.
39. Marquardt, D.W. An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *J. Soc. Ind. Appl. Math.* **1963**, *11*, 431–441. [CrossRef]
40. Wang, D.; Wei, Y.; Wang, Y.; Wang, J. A Robust and Low Computational Cost Pitch Estimation Method. *Sensors* **2022**, *22*, 16. [CrossRef] [PubMed]
41. Nachar, N. The Mann-Whitney U: A Test for Assessing Whether Two Independent Samples Come from the Same Distribution. *Tutor. Quant. Methods Psychol.* **2008**, *4*, 13–20. [CrossRef]
42. Kirch, W. (Ed.) Pearson's Correlation Coefficient. In *Encyclopedia of Public Health*; Springer: Dordrecht, The Netherlands, 2008; pp. 1090–1091. [CrossRef]
43. The MathWorks Inc. *MATLAB Version: 9.13.0 (R2022b)*; The MathWorks Inc.: Natick, MA, USA, 2022; Available online: https://www.mathworks.com (accessed on 10 June 2023).
44. Hess, W. Pitch and Voicing Determination of Speech with an Extension Toward Music Signals. In *Springer Handbook of Speech Processing*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 181–212. [CrossRef]
45. Host-Madsen, A.; Händel, P. The effect of sampling and quantization on frequency estimation. In Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181), Seattle, WA, USA, 15–15 May 1998; Volume 4, p. 2220. [CrossRef]
46. Temple, A.R. Real-Time FPGA Implementation of a Neuromorphic Pitch Detection System. Ph.D. Thesis, Loughborough University, Loughborough, UK, 1999. Available online: https://hdl.handle.net/2134/13610 (accessed on 20 May 2023).
47. A Simplified Speaker Recognition System Based on FPGA Platform | IEEE Journals & Magazine | IEEE Xplore. Available online: https://ieeexplore.ieee.org/document/8897096 (accessed on 28 February 2023).
48. Monti, G.; Sandler, M. Monophonic transcription with autocorrelation. In Proceedings of the COST G-6 Conference on Digital Audio Effects (DAFX-00), Verona, Italy, 7–9 December 2023; pp. 257–260.

applied sciences

MDPI

Article

A Feasibility Study for a Hand-Held Acoustic Imaging Camera

Danilo Greco [1,2]

[1] DiSEGIM—Department of Economics, Law, Cybersecurity, and Sports Sciences, Università Degli Studi di Napoli Parthenope, Via Guglielmo Pepe, 80035 Nola, Italy; danilo.greco@uniparthenope.it; Tel.: +39-0815476619

[2] DIBRIS—Department of Informatics, Bioengineering, Robotics and Systems Engineering, Università Degli Studi di Genova, Via Dodecaneso 35, 16146 Genova, Italy

Abstract: Acoustic imaging systems construct spatial maps of sound sources and have potential in various applications, but large, cumbersome form factors limit their adoption. This paper investigates methodologies to miniaturize acoustic camera systems for improved mobility. Our approach optimizes planar microphone array design to achieve directional sensing capabilities on significantly reduced footprints compared to benchmarks. The current prototype utilizes a 128−microphone, 50×50 cm^2 array with beamforming algorithms to visualize acoustic fields in real time but its stationary bulk hampers portability. We propose minimizing the physical aperture by carefully selecting microphone positions and quantities with tailored spatial filter synthesis. This irregular array geometry concentrates sensitivity toward target directions while avoiding aliasing artefacts. Simulations demonstrate a 32−element, $\approx 20 \times 20$ cm^2 array optimized this way can outperform the previous array in directivity and noise suppression in a sub-range of frequencies below 4 kHz, supporting a $4\times$ surface factor reduction with acceptable trade-offs. Ongoing work involves building and testing miniature arrays to validate performance predictions and address hardware challenges. The improved mobility of compact acoustic cameras could expand applications in car monitoring, urban noise mapping and other industrial fields limited by current large systems.

Keywords: acoustic imaging; microphone arrays; robust super directive beamforming; array processing; miniaturization; aperiodic sparse planar arrays; filter-and-sum beamforming; data-independent 3-D digital beamforming; low-cost acoustic camera; sensor mismatches

Citation: Greco, D. A Feasibility Study for a Hand-Held Acoustic Imaging Camera. *Appl. Sci.* **2023**, *13*, 11110. https://doi.org/10.3390/app131911110

Academic Editors: Dimitrios G. Aggelis, Wen Wang and Emanuel Guariglia

Received: 15 June 2023
Revised: 7 September 2023
Accepted: 4 October 2023
Published: 9 October 2023

1. Introduction

Acoustic imaging is an emerging methodology that aims to create spatial maps of sound sources analogous to conventional optical cameras. It digitally reconstructs acoustic fields based on the analysis of sound waves captured by microphone arrays and advanced signal processing algorithms [1,2]. Well-known and widespread acoustic imaging applications include sonar and ultrasound [3]. Potential applications include pinpointing mechanical faults in machines [4], monitoring transport noise pollution [5], locating sniper fire in combat zones [6], validating room acoustics models [7], and many others spanning industrial inspection, public health, security, and virtual reality domains. While optical cameras form images along physical sight lines, acoustic cameras sample sound arriving from diverse directions and computationally focus on particular points in space to create visualizations of sound intensity and origin. This allows passive localization and separation of multiple simultaneous sources based on spatial diversity. The core signal processing operation is known as beamforming, which applies carefully engineered delays and filters to the microphone signals to isolate particular propagation directions [3]. However, performance is subject to physical constraints and trade-offs inherent to the microphone array design [2,8]. In particular, existing real-time acoustic imaging systems utilize large multi-microphone apertures to achieve sufficient angular resolution and sensitivity [9].

This leads to bulky configurations unsuitable for portable applications with limited size, weight, and power budgets (see for instance https://www.flir.com/browse/industrial/acoustic-imaging-cameras/, accessed on 14 June 2023). There is strong motivation to miniaturize such cameras for more accessible and more extensive deployment. This paper investigates methodologies to reduce the form factor of real-time acoustic imaging systems by an order of magnitude while minimizing losses in spatial filtering fidelity. We specifically consider the case study of the *Dual Cam* (Figure 1), an acoustic camera prototype developed at the Italian Institute of Technology [10]. It combines a 0.5×0.5 m^2, 128−element microphone array with an embedded system for real-time beamforming and visualization over wideband [500, 6400] Hz. While high-performing, the large stationary apparatus restricts usage scenarios. Our approach is to co-optimize the array configuration and beamforming filters through simulations to retain directional acoustic sensing capability on dramatically smaller footprints. We quantitatively demonstrate that a 32−microphone array over a 0.21×0.21 m^2 aperture optimized for the acoustic frequencies of interest can provide better directivity than the 128−microphone, 0.50 m aperture *Dual Cam* array from 2 kHz to 6.4 kHz. This supports reducing system size by up to $4\times$ with tolerable imaging trade-offs. Ongoing efforts are focused on constructing miniature microphone arrays guided by these simulations to develop portable acoustic cameras that interface with tablets/laptops and smartphones for easy deployment. Enabling compact, real-time acoustic imaging could expand applications in machine health monitoring where vibration analysis indicates developing faults before catastrophic failure [11], urban noise pollution mapping to improve public health interventions [12], and augmented/virtual reality scene analysis for realistic audio rendering [13,14]. The methodologies and insights presented provide an array of signal processing starting points for researchers and engineers aiming to transform acoustic imaging capabilities from the lab to the field.

Figure 1. *Dual Cam* prototype integrates co-located acoustic and visual imaging modalities using a planar microphone array paired with a video camera [10].

2. Acoustic Imaging Concepts

Acoustic imaging seeks to form a spatial map of sound sources in a scene analogous to standard cameras that produce visual images using projected light patterns. Conventional optics passively focus rays along physical lines of sight to reconstruct perspectives. In contrast, acoustic imaging relies on digital sampling, processing, and interpreting acoustic fields using microphone array receivers and beamforming algorithms [15,16]. We provide an overview of fundamental principles including angular resolution, aliasing, array geometry considerations, and beamforming basics.

2.1. Angular Resolution

A key parameter in acoustic imaging is the angular resolution, which determines the camera's ability to spatially discriminate sources [1]. This is influenced by the acoustic

wavelength λ, propagation medium sound speed c, and array physical aperture dimensions. The angle θ between two visible sources must satisfy:

$$\theta \geq \frac{\lambda}{L} \tag{1}$$

where L is the array size normal to the direction of arrival; the constraint arises because waves emitted from within θ will produce signals separated by less than a wavelength, making them indistinguishable. The approximate relationship shows that larger apertures provide finer angular resolution. However, simply using more microphones is insufficient—their positioning is critical, as discussed next.

2.2. Aliasing

The spatial sampling pattern of the microphones can result in aliasing artefacts that distort the acoustic image (Figure 2).

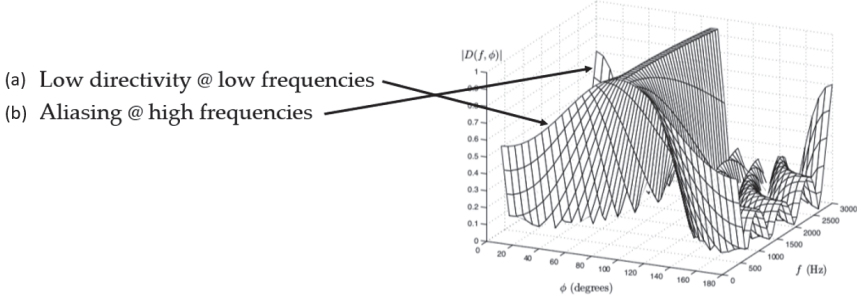


Figure 2. Broadband beamforming issues (1-D): (a) low directivity at low frequencies and (b) aliasing at high frequencies. $B(f, \phi)$, beampattern; f, function of frequency; ϕ, DOA (direction of arrival) [17].

Aliasing occurs when sources at different angular positions generate identical array signals, preventing unique localization. Uniform linear or grid arrays are especially prone due to their periodic sampling structure. Sources separated by multiples of the angular period:

$$p = \sin^{-1}\left(\frac{\lambda}{d}\right) \tag{2}$$

where d is the grid spacing, will be aliased since the path length difference between microphones is identical. The resulting grating lobes complicate acoustic imaging by introducing ghost sources and ambiguity. For instance, if we simulate a periodic displacement in a planar array of 25×25 cm^2 by putting 32 microphones in a regular grid (Figure 3), analysing the beampattern in the window of frequencies [2, 6.4] kHz, we found grating lobes, more evident at higher frequency (Figure 4). A common solution is breaking periodicity by using randomized or aperiodic array layouts [18,19]. However, this must be balanced with microphone density and area coverage to retain sensitivity. Careful array optimization is required to design alias-free configurations suited for imaging.

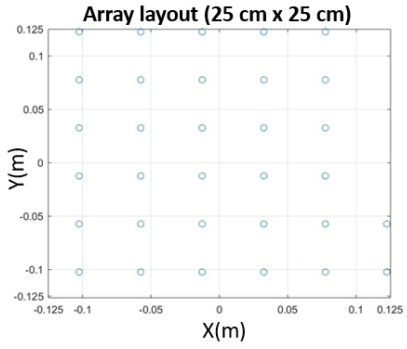


Figure 3. Simulation of a periodic 32−microphone positioning on a planar array 25 × 25 cm^2.

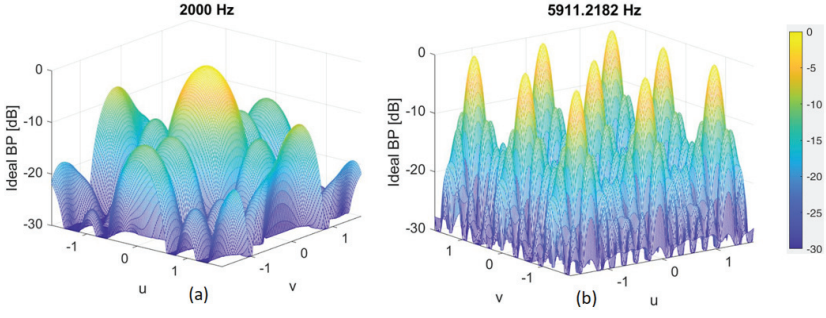


Figure 4. Two−dimensional beam pattern using a periodic positioning of 32−microphones on a planar array 25 × 25 cm^2 (Figure 3) at different frequencies: (**a**) beam pattern at 2 kHz, (**b**) beam pattern at ≈6 kHz, both functions of θ and ϕ (u and v; see later on). Increasing the frequency, the grating lobes equal the main central lobe. The colorbar maps from 0 dB (yellow) to -30 dB (blue).

2.3. Array Geometry

Microphone array geometry plays a critical role in acoustic imaging performance. Key factors are:

- Aperture —The overall physical size determines angular resolution. Larger apertures improve discrimination.
- Number of microphones—More microphones provide enhanced spatial sampling at the cost of complexity.
- Layout—Positions within the aperture area. Uniform grids simplify analysis but suffer aliasing. Randomized arrangements help reduce lobes.
- Symmetry—Circular/spherical arrays enable uniform coverage but planar designs are easier to manufacture.

The greater the physical dimensions that an array of sensors has compared to what a single transducer allows, with the same wavelength λ considered, the greater the capacity for spatial discrimination of the directions of origin of the signals, and therefore the greater the resolution, referred to in this context as angular resolution. Usually, the dimensions of an array are quantified by evaluating its spatial opening D defined as the maximum distance that separates two elements belonging to it. Therefore, the spatial discrimination capacity of an array coincides with a value proportional to D/λ. An array has properties of flexibility unattainable by the single sensor with the same implementation simplicity. In fact, in many applications, it may be necessary to modify the spatial filtering function in real time to maintain an effective attenuation of the interfering signals to the advantage of the desired ones. This becomes essential in imaging applications in which the pointing

direction changes constantly in order to scan all possible directions of arrival of the signal. This change, in a system that adopts an array of transducers, is achieved simply by varying the way in which the beamforming combines the data coming from each sensor in a linear fashion; in the case of a single transducer, the change is impractical as it would be necessary to act directly on the physical characteristics of the sensor.

2.4. Beamforming

Beamforming is the digital signal processing technique that allows microphone arrays to focus on particular directions [20]. It computationally mimics the capability of parabolic dish antennas to isolate radio sources. Delay-and-sum is the simplest beamforming approach. Signals originating from the look direction arrive simultaneously and in phase at the central reference point when appropriately time-shifted. Coherent summation of the aligned microphone signals passes the source undistorted. Off-axis sources remain misaligned, causing attenuation after summing. More advanced optimal and adaptive methods synthesize filters to achieve configurable directional selectivity. The ability to digitally steer the focus point enables scanning to form full acoustic images. Beamforming transforms the microphone array into a highly directional virtual sensor with sensitivity patterns tailored through data-dependent signal processing. However, fundamental limits arise from the array geometry and ambient noise. Robust acoustic imaging requires jointly optimizing the array configuration with advanced beamforming techniques [21–24] designed to maximize directional resolution.

Ideally, the array would be infinitely large with continuous spatial sampling. In practice, size constraints necessitate designing optimized configurations to maximize imaging capabilities given physical limitations. There are inherent trade-offs between aperture dimensions, microphone density, aliasing artefacts, and processing load that acoustic camera architectures must balance.

The filter-and-sum beamforming algorithm [17,19,25–34] provides improved performance over the delay-and-sum algorithm by applying filters to the microphone signals. This allows the array to focus on a specific direction more effectively and reduce the sidelobes, resulting in a clearer and more detailed acoustic image.

3. *Dual Cam* Acoustic Camera

We provide an overview of *Dual Cam*, an acoustic camera prototype developed at the Italian Institute of Technology [10,18]. It combines a co-located planar microphone array and video camera for aligned audiovisual imaging, as illustrated in Figure 1. The current implementation utilizes a 0.5×0.5 m^2 128−element microphone array fabricated on a custom-printed circuit board working over wideband [500–6400] Hz. Each microphone output is digitized and processed in real time by an embedded system that performs beamforming over an azimuth–elevation scan region, where (θ, ϕ) equals (90×360) degrees. This generates acoustic images registered to the synchronized video feed, enabling visualization of spatial sound sources. However, the large form factor makes the device cumbersome for portable applications. Our goal is to significantly miniaturize the system while retaining imaging fidelity. Reducing the form factor exacerbates grating lobes and limits low-frequency coverage. Advanced optimization of the layout and beamforming filters is necessary to recover imaging performance on smaller scales through irregular configurations with microphone positioning tailored to the sensors and frequencies of interest.

Acoustic imaging systems utilizing microphone arrays enable novel techniques for localizing and separating multiple simultaneous sound sources. However, real-world deployment remains limited given the unwieldy equipment required. The array's 128 microphones are strategically positioned using an optimized irregular layout [18,19,35–39] to synthesize directional acoustic images of the sound field when paired with beamforming algorithms [40] (Figure 5). These acoustic images represent spatial auditory information by mapping frequencies to pixels corresponding to locations. While originally high-dimensional, the key acoustic data can be compressed into perceptually relevant

mel-frequency cepstral coefficients to reduce computational costs [41] (Figure 5). Studies demonstrate acoustic images can boost model performance by transferring spatial representations to improve audio classification accuracy. The addition of spatial audio details also helps disambiguate sources and generalize to new datasets [42]. However, real-world systems may lack these imaging capabilities. This work examines methodologies to distil the benefits of acoustic images even without access to specialized hardware. Optimized planar arrays provide more accurate spatial audio details compared to individual microphones by sampling sound fields from varied directions. Advanced beamforming techniques enable directionally focused listening to isolate specific sources in noisy scenes. Processing multi-microphone signals remains computationally intensive, though emerging algorithms and parallel computing facilitate real-time performance. The filter-and-sum [3,43] beamformer synthesizes an array of impulse responses to steer directional sensitivity. While specially designed microphone arrays can provide valuable spatial auditory images, this research investigates generalized approaches using array signal processing to improve audio sensing tasks without access to imaging hardware.

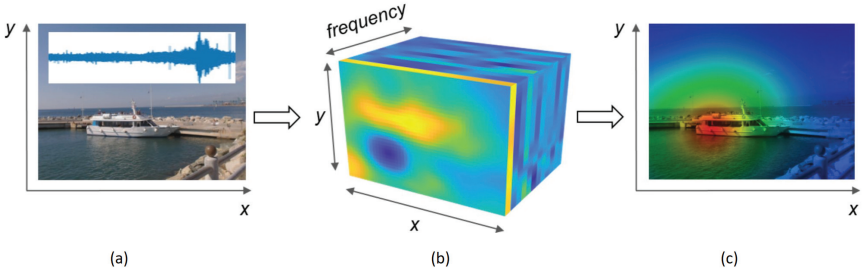


Figure 5. From raw audio (**a**) to 3-D acoustic image (**b**) to 2-D energy heatmap (**c**) (from red maximum sound to blue minimum sound) [44].

The filter-and-sum beamforming algorithm is a method for synthesizing the finite impulse response (FIR) coefficients [17,25,43] for small-sized two-dimensional microphone arrays [19]. This method can be used to generate acoustic images by focusing the array on a specific direction in space and enhancing the signal coming from that direction.

The live acoustic imaging pipeline consists of:

1. Digitizing microphone outputs through multichannel audio sampling.
2. Partitioning the multichannel record into short time frames.
3. Synthesizing beamformer filters according to designed array geometry.
4. Applying filters and aligning signals for each scanning direction.
5. Coherently summing aligned microphone channels to obtain beam pattern power.
6. Repeating overall look directions to generate acoustic image frames registered to video.

This digital signal chain transforms the raw multichannel audio into visualizations of spatial sound intensity (Figures 5 and 6). However, the fidelity is contingent on array configuration, density, and beamforming approach. We investigate techniques to co-optimize these parameters for compact, real-time acoustic cameras without prohibitive degradation compared to larger form factors.

(**a**) (**b**) (**c**)

Figure 6. Three examples from a collected dataset. We visualize the acoustic image by summing the energy of all frequencies for each acoustic pixel. The resulting map (from red maximum sound to blue minimum sound) is overlaid on the corresponding RGB frame. From left to right: (**a**) drone, (**b**) train, (**c**) vacuum cleaner [42].

4. Materials and Methods

Recent advancements in acoustic imaging have enabled novel techniques for localizing and separating multiple sound sources within complex auditory scenes. However, current implementations are often constrained to laboratory settings due to large, unwieldy equipment. This research aims to transform an existing prototype (Figure 1) into an engineered portable device for real-world sound source separation (Figure 7). Through compact microphone array design and machine/deep learning algorithms run on a coupled tablet/laptop (for instance Microsoft Surface Pro or Dell Latitude 7230EX) (Figure 8), we can achieve a handheld multimodal camera that captures and processes synchronized audio and video to map multiple simultaneous sounds.

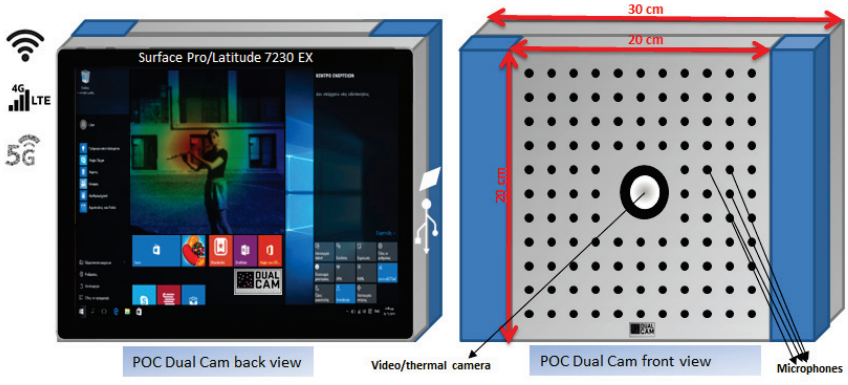


Figure 7. New *Dual Cam* 2.0 POC (proof of concept) idea. The periodic positioning of the microphones is generic and for illustrative purposes only.

CNNs (convolutional neural networks) frequently employ image classification and segmentation tasks through acoustics. Acoustic images provide automatic learning opportunities for CNNs' relevant features. Leading CNN structures like ResNet and U-Net have been adopted for acoustic image evaluation. Using RNNs (recurrent neural networks) like LSTMs enables the examination of audio visualization sequences evolving with time. Useful applications exist for monitoring items or procedures within video acoustic microscopy information. Acoustic image denoising and reconstruction are tasks that autoencoders excel at performing. The encoded data enables the decoder to recreate the revitalized picture flawlessly. Using GANs (generative adversarial networks) realistic sound imagery is generated, benefiting data amplification and modelling initiatives. Coordinated development through shared training brings the generator and discriminator closer to perfection. From a historical perspective, these classic ML algorithms—including random forests and support vector machines—continue to serve us well. Manual feature creation makes their

training process speedier and more straightforward. Two prominent unsupervised learning approaches—k-means clustering and principal component analysis—assist in identifying hidden patterns within acoustic information. Interactive queries enable users to efficiently annotate crucial data points through active learning techniques. Model selection hinges on parameters like dataset size, work objective, and processing capacity. By investing time and resources into rigorous evaluation, we can create models capable of producing consistent outputs. To retain directional sensitivity in a smaller form factor, we optimize array layouts using analytical filter synthesis and stochastic optimization, assessing robustness via statistical error analysis. Novel steps in our approach are:

- We optimize the analytic form of the cost function in order to cut the simulation computational load. This optimization of the cost function is a novel contribution beyond the existing state-of-the-art methods, improving computational efficiency.
- We include the statistical evaluation of the mismatches of the microphones that are more important in shrinking the array size. The statistical characterization of microphone mismatches enables novel array size reduction.
- We optimize the FOV (field of view) and the frequency bandwidth according to the array size reduction to explore upper harmonic reconstruction to determine whether intelligibility is retained without fundamental frequencies. The joint optimization of FOV, frequency band, and array size reduction using the upper harmonics for intelligibility preservation is an unexplored area representing a novel research direction.

A key goal is extending as much as possible the minimum detectable frequency to improve directivity with fewer elements. While reducing array aperture, we must balance performance trade-offs from decreased low-frequency directivity and potential undersampling artefacts. This work details simulations on irregular aperiodic subsampling to concentrate high-frequency information while avoiding grating lobes (Figure 4) and exploring upper harmonic reconstruction to determine whether intelligibility is retained without fundamental frequencies reducing the device's bandwidth to optimize the simulation metrics. Following prototype optimization and evaluation using audio test signals, we compare metrics like signal-to-noise ratio to the original large-scale system. This study aims to progress acoustic imaging capabilities from constrained laboratory settings towards real-world applications through engineered mobile platforms. This work aims to re-engineer a compact, portable prototype (Figure 7) that transmits synchronized audio and video data streams to a commercial tablet or laptop. The audio is captured by an array of microphones on the primary module, while the video is acquired by a thermographic or conventional camera. These peripheral modules interface with the central unit via multiple USB connections. Embedding an FPGA onboard the central module alongside an ARM processor enables straightforward interfacing leveraging their integrated architecture (Figure 8). The system operates on battery power with LED indicators and debugging ports and can dock to the tablet mechanically. A remote internet link via WiFi, LTE, or 5G facilitates control and data sharing. The tablet/laptop display provides a visualization interface to process the multimodal data streams using algorithms, machine learning, and deep neural networks. This integrated design retains the core functionality of the original laboratory prototype while minimizing size and maximizing portability for real-world deployment. Ongoing work focuses on implementation challenges including power optimization, heat dissipation, enclosure design, calibration, and field testing. By progressing acoustic imaging capabilities from constrained lab settings to handheld adaptable platforms, this research aims to unlock new applications in machine condition monitoring, spatial sound mapping, and other domains limited by current large-scale wired systems.

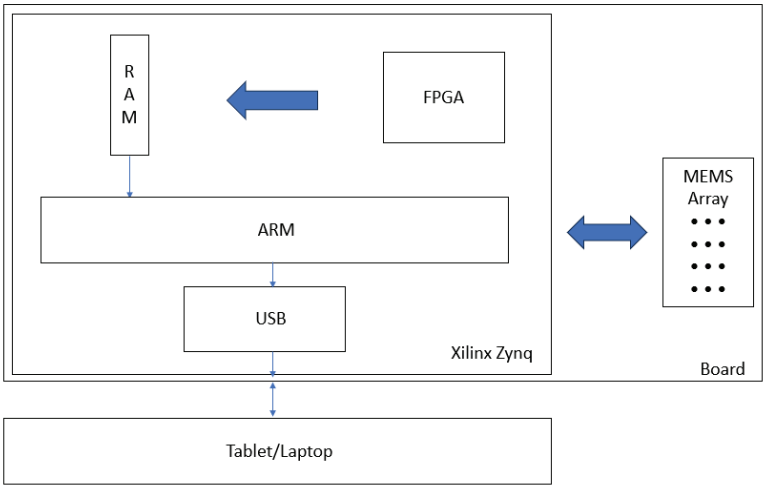


Figure 8. The *Dual Cam* board is based on an FPGA that reads continuously from the I2S MEMS microphones (TDK/InvenSense); using a programmed DMA the read microphones values are stored in a RAM buffer. In our new proof of concept (POC) the processor can easily read data from RAM and redirect them to the USB port.

5. Array Optimization Methodology

Reducing the physical array aperture while maintaining usable imaging resolution requires balancing size, microphone number, spatial sampling, and angular coverage. Simply downscaling a regular grid array would significantly increase grating lobes (Figures 4 and 9). We instead utilize array signal processing optimization procedures that allow unconventional configurations with microphone numbers and positions tailored to imaging requirements. Irregular layouts are synthesized based on maximizing acoustic power focused toward directions of interest and minimizing ghost images. Key concepts are briefly introduced below (Figure 9), with formulations adapted from [8,18,35].

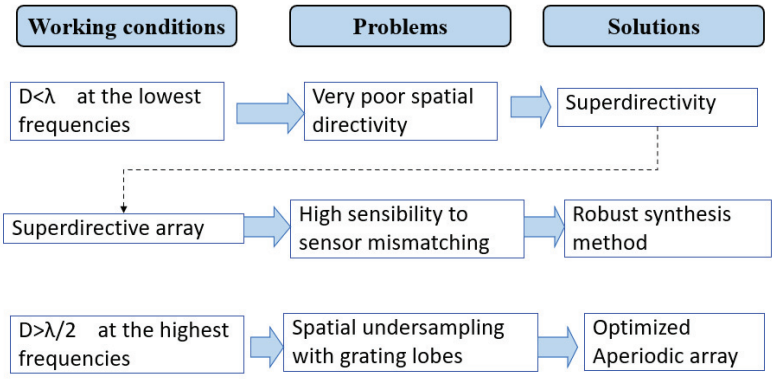


Figure 9. Conceptual framework in the microphone array simulation.

5.1. Problem Parameterization

We consider a planar array of N microphones located at positions $\mathbf{r}_n = (x_n, y_n)$ in the xy plane (Figure 10). Acoustic sources at frequency f impinge on the array from angles θ and ϕ. The goal is to generate high-resolution, low-artefact acoustic images over the

signal band $[f_{\min}, f_{\max}]$. The array geometry and frequency-dependent beamformer filter coefficients $\mathbf{w}(f) = [w_1(f), \ldots, w_N(f)]^T$ are jointly optimized to maximize directional sensitivity. The complex beam pattern $B(\theta, \phi, f)$ encodes the array response at each look angle and is parameterized as:

$$B(\theta, \phi, f) = \sum_{n=1}^{N} w_n(f) e^{-j2\pi f \hat{\mathbf{r}}(\theta,\phi)\cdot \mathbf{r}_n} \tag{3}$$

where $\hat{\mathbf{r}}(\theta, \phi)$ is the source position unit vector. The expression depends on both the layout $\mathbf{r}_n$ and filter coefficients $w_n(f)$ which are microphone-specific filters to be optimized. The expression has directionality dependence on both the layout $\mathbf{r}_n$ and filter responses $w_n(f)$. To allow joint optimization, a cost function $J(\mathbf{w}, \mathbf{r})$ is formulated that balances simultaneously directional focus, artefact suppression, frequency coverage, and robustness. It incorporates an idealized unity gain beampattern $B_0(\theta, \phi, f)$ at the look direction and minimizes the deviation from this response over angle-frequency space. Regularization terms manage overall beamformer gain and robustness. The optimization determines array configurations and filters customized for the imaging application. With the beam pattern expressed as $B(\theta, \phi, f) = \mathbf{w}^T(f)\mathbf{V}(\theta, \phi)$, the filter coefficients $\mathbf{w}(f)$ can be analytically extracted from the cost function into a closed-form solution $\mathbf{w}_{\text{opt}}(f) = \mathbf{R}^{-1}(f)\mathbf{q}(f)$. For the array layout optimization with $\mathbf{w}_{\text{opt}}(f)$ fixed, simulated annealing avoids poor local minima. Iterative stochastic perturbations to microphone locations $\mathbf{r}_n$ are accepted probabilistically based on the cost function to enable escaping local minima. After sufficient iterations, the array geometry converges to enhance directionality. To improve robustness, the cost function is averaged over possible microphone gain and phase errors modelled as random variables. This penalizes configurations with low white noise gain, minimizing sensitivity to imperfections. The expected beam pattern $E[B(\theta, \phi, f)]$ is incorporated to account for errors; this optimization framework (Figure 9) allows the designing of array geometries and filters customized for compact, robust acoustic imaging over desired frequency bands. The resulting unconventional configurations maximize power focused on look directions while minimizing off-axis contributions and artefacts using small apertures.

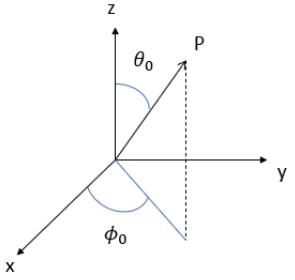


Figure 10. Cartesian coordinates system and steering angles (θ_0, ϕ_0).

The cost function $J(\mathbf{w}, \mathbf{r})$ balances several competing objectives:

1. Directional focus. Minimizing deviation of the achieved beam pattern $B(\theta, \phi, f)$ from the ideal unity gain pattern $B_0(\theta, \phi, f)$ at the look direction over angle-frequency space. This is quantified by the integral term:

$$\iiint_{\Theta,\Phi,F} |B(\theta, \phi, f) - B_0(\theta, \phi, f)|^2 d\theta d\phi df$$

2. Artefact suppression. Minimizing the beam pattern gain away from the look direction, incorporated through:
$$\iiint_{\Theta,\Phi,F} |B(\theta,\phi,f)|^2 d\theta d\phi df$$
3. Frequency coverage. Optimizing over the full band [f_{min}, f_{max}] through integration over f.
4. Robustness. Averaging over microphone imperfections by modelling gain and phase as random variables A_n.

The overall form is a weighted combination of these terms:

$$J(\mathbf{w},\mathbf{r}) = \alpha \iiint |B(\theta,\phi,f) - B_0(\theta,\phi,f)|^2 d\theta d\phi df + (1-\alpha) \iiint |B(\theta,\phi,f)|^2 d\theta d\phi df$$

where $\alpha \in [0,1]$ controls the trade-off between directional focus and artefact suppression. To minimize J, the filter coefficients $\mathbf{w}(f)$ are first optimized analytically for a fixed array layout by extracting them into a quadratic form with closed-form solution $\mathbf{w}_{\text{opt}}(f)$.

The microphone locations $\mathbf{r}_n$ are then optimized stochastically using simulated annealing to avoid poor local minima:

- Iterative random perturbations $\Delta\mathbf{r}_n$ are applied to the microphone locations.
- New locations are accepted probabilistically based on the cost J.
- Acceptance probability is higher at higher initial "temperatures" and cooled over iterations.
- After sufficient iterations, $\mathbf{r}_n$ converges to a geometry minimizing J.

This joint optimization determines array layouts and filters tailored for directional imaging over the specified band with artefact suppression and robustness. In the simulation of the beampattern of a planar array ($z = 0$) of microphones we have two angles of arrival θ and ϕ, two steering angles θ_0 and ϕ_0 (Figure 10) and two coordinates for the microphones x_n and y_n. The mathematical expression of the ideal superdirective beampattern B in far-field is:

$$B(\theta,\phi,\theta_0,\phi_0,f) = \sum_{n=1}^{N} w_n(f) e^{-j2\pi f \cdot \left[x_n \cdot \frac{\sin(\theta)-\sin(\theta_0)}{c} + y_n \cdot \frac{\sin(\varphi)-\sin(\varphi_0)}{c}\right]} \quad (4)$$

where N is the number of microphones, c = 340 m/s is the speed of the acoustic waves into the medium ($\lambda = c/f$), and $w_n(f)$ is the frequency response of the n-th filter:

$$w_n(f) = \sum_{k=1}^{K} w_{n,k} \cdot e^{-j2\pi f \cdot kT_c} \quad (5)$$

5.2. Cost Function Definition

We recall the cost function formulated to allow optimizing the array layout and beamformer filters for directional acoustic imaging:

$$J(\mathbf{w},\mathbf{r}) = \alpha \iiint_{\Theta,\Phi,F} |B(\theta,\phi,f) - B_0(\theta,\phi,f)|^2 \, d\theta \, d\phi \, df + (1-\alpha) \iiint_{\Theta,\Phi,F} |B(\theta,\phi,f)|^2 \, d\theta \, d\phi \, df \quad (6)$$

$B_0(\theta,\phi,f)$ is the idealized beam pattern with unity gain at the main look direction and zero elsewhere. The first term drives the achieved response toward the desired spatial selectivity. The second term balances overall beamformer gain and robustness. $\alpha \in [0,1]$ controls the trade-off. The integrals are approximated over discrete grids of angles and frequencies. This cost function steers the optimization toward arrays with high directionality for acoustic imaging. It encapsulates the desired balance of sharp focus, minimal artefacts, wide frequency coverage, and robustness within a single numerical

measure of performance. In order to find the position of the microphones, we have to minimize the J cost function [18] that we rewrite as:

$$J(\mathbf{w},\mathbf{r}) = \int_{\theta_{0_{\min}}}^{\theta_{0_{\max}}} \int_{\phi_{0_{\min}}}^{\phi_{0_{\max}}} \int_{\theta_{\min}}^{\theta_{\max}} \int_{\phi_{\min}}^{\phi_{\max}} \int_{f_{\min}}^{f_{\max}} |B(\mathbf{w},\mathbf{r},\theta,\phi,\theta_0,\varphi_0,f) - 1|^2 + C|B(\mathbf{w},\mathbf{r},\theta,\phi,\theta_0,\phi_0,f)|^2 d\theta d\phi d\theta_0 d\phi_0 df \tag{7}$$

where $\mathbf{r}$ is the vector with the positions of the microphones, $\mathbf{w}$ is the vector of the filter coefficients, and C is a real constant. This tunes the minimization of the first term of the cost function which is the adherenceterm and the second one which is the energy weighted term. We want to joint optimization of weights and microphones' positions and account for superdirectivity and aperiodicity. Then, the expression (4) of the beampattern B of a planar array ($z = 0$) of microphones in $2-$D becomes:

$$B(\mathbf{w},\mathbf{r},\theta_0,\phi_0,\theta,\phi,f) = \sum_{n=1}^{N}\sum_{k=1}^{K} w_{n,k} e^{-j2\pi f \cdot \left[x_n \frac{\sin(\theta)-\sin(\theta_0)}{c} + y_n \cdot \frac{\sin(\phi)-\sin(\phi_0)}{c} + kT_c\right]} \tag{8}$$

where K is the length of the FIR filter and T_c is the sampling period.

5.3. Directivity Optimization

The beam pattern expression can be reduced to:

$$B(\theta,\phi,f) = \mathbf{w}^T(f)\mathbf{V}(\theta,\phi) \tag{9}$$

where $\mathbf{w}(f) = [w_1(f),\ldots,w_N(f)]^T$ and $\mathbf{V}(\theta,\phi)$ is an array manifold vector with phase terms dependent on look direction. This allows extracting the filter coefficients from the cost function, converting optimization over $\mathbf{w}(f)$ into a quadratic form with a closed-form solution:

$$\mathbf{w}_{opt}(f) = \mathbf{R}^{-1}(f)\mathbf{q}(f) \tag{10}$$

where $\mathbf{R}(f)$ and $\mathbf{q}(f)$ accumulate integration terms. The optimal $\mathbf{w}_{opt}(f)$ maximizes directionality for a given layout.

5.4. Layout Optimization

In order to achieve and improve robustness against microphone imperfections, we perform an optimization of the mean performance i.e., the multiple integrals of the cost function over the sensors' phase $e^{-\gamma_n}$ and gain a_n $A_n = a_n \cdot e^{-\gamma_n}$ considered as random variables, getting a *robust* cost function with the PDF (probability density function) of the random variable A_n [38]. The cost function $J(\mathbf{w},\mathbf{r})$ is averaged over possible gain and phase errors by modelling the microphone responses A_n as random variables:

$$J^{\text{tot}}(\mathbf{w},\mathbf{r}) = \int_{A_0} \cdots \int_{A_{N-1}} J(\mathbf{w},\mathbf{r},A_0,\ldots,A_{N-1}) f_A(A_0)\cdots f_A(A_{N-1}) dA_0 \cdots dA_{N-1} \tag{11}$$

where $f_A(A_n)$ is the PDF of the random variable A_n. This incorporates robustness into the optimization. However, evaluating the multiple integrals results in a large number of variables (microphone positions and FIR filter coefficients) making direct optimization of J^{tot} computationally infeasible. To address this, a change of variables is made:

$$\begin{cases} u = \sin(\theta) - \sin(\theta_0) \\ v = \sin(\phi) - \sin(\phi_0) \end{cases} \tag{12}$$

Substituting into the beam pattern expression gives:

$$B(\mathbf{w}, \mathbf{r}, u, v, f) = \sum_{n=1}^{N} \sum_{k=1}^{K} w_{n,k} e^{-j2\pi f(x_n \frac{u}{c} + y_n \frac{v}{c} + kT_c)} \tag{13}$$

This allows for defining a simplified cost function:

$$J^{\text{tot}}(\mathbf{w}, \mathbf{r}) = \int_{u_{\min}}^{u_{\max}} \int_{v_{\min}}^{v_{\max}} \int_{f_{\min}}^{f_{\max}} |B(\mathbf{w}, \mathbf{r}, u, v, f) - 1|^2 + C|B(\mathbf{w}, \mathbf{r}, u, v, f)|^2 \, df \, du \, dv \tag{14}$$

The filter coefficients $\mathbf{w}$ can then be analytically extracted into a quadratic form with a closed-form solution:

$$J^{\text{tot}}(\mathbf{w}, \mathbf{r}) = \mathbf{w}^T \mathbf{M} \mathbf{w} - 2\mathbf{w}^T \mathbf{r} + s \tag{15}$$

Further , the robustness integrals over A_n can be approximated in closed form. The microphone positions $\mathbf{r}_n$ are then numerically optimized using simulated annealing to avoid local minima. Iterative stochastic perturbations escape suboptimal configurations based on the cost. This joint optimization determines robust array geometries and filters for directional imaging. Performance is evaluated using metrics such as directivity $D(f)$ and white noise gain $WNG(f)$. The expected beam pattern power $E\{|B(\theta, \phi, f)|^2\}$ is also incorporated to account for microphone imperfections. Minimizing tolerance to errors improves reliability. This framework enables the designing of robust, compact arrays tailored for spatial acoustic imaging over desired bands. The new cost function is a good approximation of the original one, allowing the number of integrals to be reduced. The vector $\mathbf{w}$ can be extracted from the multiple integrals in the robust cost function obtaining a quadratic form in $\mathbf{w}$ [18]. With ideal filters derived analytically, the microphone locations $\mathbf{r}_n$ are optimized stochastically. A simulated annealing approach is used to avoid poor local minima. Iterative perturbations to $\mathbf{r}_n$ are accepted probabilistically based on the cost, allowing escape from local minima at high process "temperatures" that are gradually cooled. After sufficient iterations, the microphone layout converges toward a configuration with scattering tailored to enhance directional sensitivity and suppress off-target responses. The joint optimization determines array geometries and beamformers customized for compact acoustic imaging over specified bands. For a fixed microphone displacement, the global minimum of the robust cost function can be calculated in a closed form. Conversely, the presence of local minima with respect to the microphone position prevents the use of gradient-like iterative methods. The final solution is given by a hybrid strategy analytic and stochastic based on the Simulated Annealing algorithm [36,45] (Figure 11). The steps are:

- Iterative procedure aimed at minimizing an energy function $f(y)$.
- At each iteration, a random perturbation is induced in the current state y_i.
- If the new configuration, y^*, causes the value of the energy function to decrease, then it is accepted.
- If y^* causes the value of the energy function to increase, it is accepted with a probability dependent on the system temperature, in accordance with the Boltzmann distribution.
- The temperature is a parameter that is gradually lowered, following the reciprocal of the logarithm of the number of iterations.
- The higher the temperature, the higher the probability of accepting a perturbation causing a cost increase and of escaping, in this way, from unsatisfactory local minima.

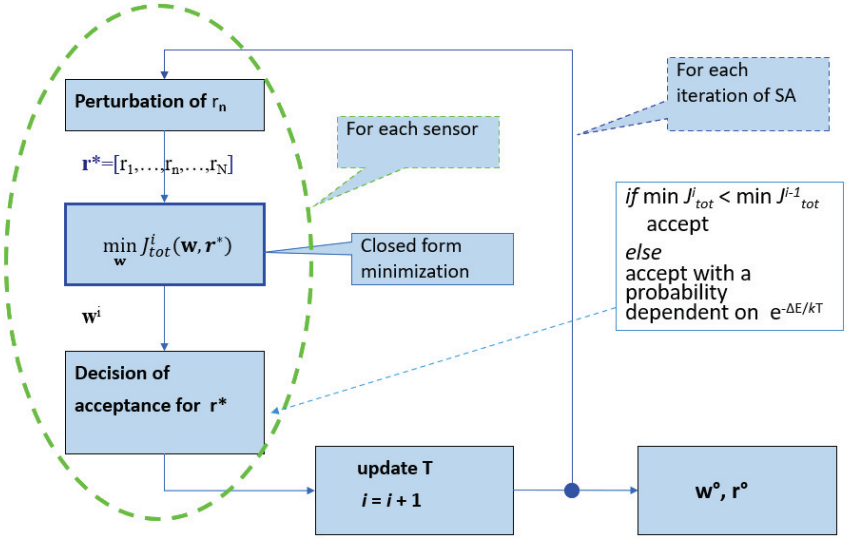


Figure 11. Flow chart of Simulated Annealing algorithm for function cost minimization.

5.5. Robustness Constraints

We require quantitative metrics to evaluate the algorithm's beamforming performance. The key metrics utilized are the frequency-dependent directivity $D(f)$ and white noise gain $WNG(f)$, computed for steering angles θ_0 and ϕ_0. For a planar array, the directivity (in dB) is defined as:

$$D(f) = \frac{|B(\theta_0,\phi_0,f)|^2}{\frac{1}{4\pi}\int_0^{2\pi}\int_0^{\pi}|B(\theta,\phi,f)|^2\sin(\theta)d\theta d\phi} \tag{16}$$

The white noise gain (in dB) quantifies robustness towards array imperfections:

$$WNG(f) = \frac{|B(\theta_0,\phi_0,f)|^2}{\sum_{n=1}^{N}|w_n(f)|^2} \tag{17}$$

We propose the expected beam pattern power (EBPP) metric to statistically evaluate the impact of variance in array gain and phase on the beam pattern $B(f)$:

$$\begin{aligned} &B_e^2(\theta,\phi,f) = E|B(\theta,\phi,f)|^2 = \\ &\int_{A_0}\cdots\int_{A_{N-1}}|B(\theta,\phi,f)|^2\cdot f_{A_0}(A_0)\ldots f_{A_{N-1}}(A_{N-1})dA_0\ldots dA_{N-1} \end{aligned} \tag{18}$$

Microphone imperfections can distort the array away from the ideal modelled response. As in [38], robustness is incorporated by averaging the cost function over possible gain and phase errors through the expectation operator $E\{.\}$.

$$E[B(\theta,\phi,f)] \approx |B(\theta,\phi,f)|^2 + \frac{1}{WNG(f)}(\sigma_g^2+\sigma_\psi^2) \tag{19}$$

where σ_g^2 and σ_ψ^2 are variances of normally distributed microphone magnitude and phase mismatches, and $WNG(f)$ is a white noise gain term. Minimizing cost tolerance to modelled errors helps ensure reliable performance. This full optimization framework allows the designing of microphone array geometries and beamformers customized for compact acoustic imaging over desired signal bands. The unconventional configurations maximize power focused toward look directions while suppressing artefacts and minimizing off-axis contributions to enable resolving spatial sound fields from small apertures. We next utilize this approach in a simulation case study of miniature array optimization. With the

proposed metrics in place, we evaluate the directivity $D(f)$, white noise gain $WNG(f)$, and expected beam pattern power (EBPP) for our initial simulated array design. We model the microphone mismatches as Gaussian distributions with $\sigma_g = 0.03 = 3\%$ for gain error and $\sigma_\psi = 0.035$ rad $\cong 2^\circ$ for phase error. This preliminary simulation provides favourable results across the three figures of merit, indicating promising performance in reshaping the sensor array for the *Dual Cam* 2.0 system. The directivity quantifies the main lobe sharpness, white noise gain captures robustness, and the expected beam pattern incorporates statistical variations—together assessing the shaped array's directional sensitivity, imperfections tolerance, and expected real-world behaviour. Further refinements to the array geometry and element tuning will build upon these initial positive findings, working toward an optimal miniature microphone configuration.

6. Simulation Configuration

We implement the array optimization procedures in MATLAB to enable rapid evaluation of miniaturized acoustic camera designs. The custom cost function represents the mismatch between achieved and ideal beampatterns over angles Θ, Φ and frequencies $F = [0.5, 6.4]$ kHz. At each iteration, filter coefficients are computed analytically then microphone positions are perturbed stochastically to minimize artefacts. The optimization concentrates power within a ± 20 degree main lobe while suppressing sidelobes. Array performance is assessed by analysing:

- Directivity—angular discrimination capability;
- White noise gain (WNG)—robustness to fabrication variations;
- Beam patterns and sidelobe levels—imaging artefacts.

The numerical approach allows for efficient simulation of miniaturized configurations to quantify expected imaging performance and determine plausible hardware parameters. We consider three scenarios:

1. A 32−microphone 0.25 m square array optimized from 2 to 6.4 kHz.
2. A 32−microphone 0.21 m square array optimized from 2 to 6.4 kHz.
3. A 32−microphone 0.21 m square array covering [0.5, 6.4] kHz for comparison with *Dual Cam* specifications (128−microphone on a 0.5 m square array).

The different number of microphones and expanded frequency range in Case 3 demonstrates trading off aperture size versus density given constraints. Comparisons with a modelled 128−element 0.5 m array representing *Dual Cam* provide context on expected miniaturization imaging trade-offs. The simulation results guide physical prototype development by predicting achievable performance bounds with compact arrays.

7. Miniaturized Array Optimization: Results and Discussion

We present synthesized array configurations from the three simulated case studies along with an analysis of beam patterns, directivity, and white noise gain for *Dual Cam* 2.0.

7.1. Thirty-Two-Microphones, [2, 6.4] kHz, Array 25 × 25 cm^2

The first scenario optimizes a 0.25×0.25 m^2 32−microphone array for a 4.4 kHz bandwidth. After 100 iterations, the cost function converges as shown in Figure 12. The corresponding irregular array geometry has an aperiodic structure with variable microphone spacing tailored for the acoustic parameters.

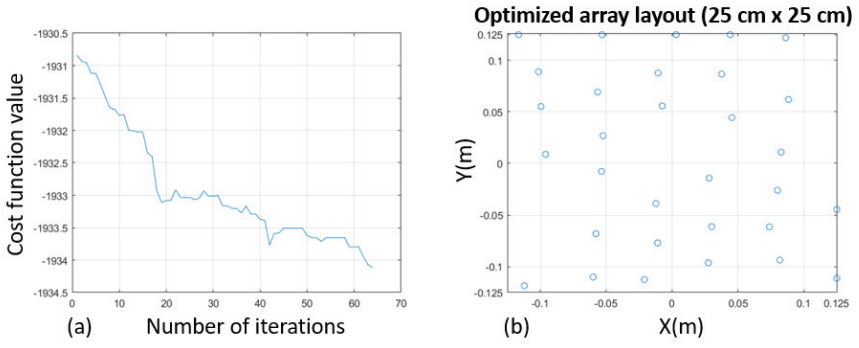


Figure 12. Simulation results for [2, 6.4] kHz optimization of a 32−elements 0.25 m acoustic array: (**a**) cost function convergence over 100 iterations, (**b**) optimized 32−microphone 0.25 m array layout.

We tried to reduce the planar array aperture (u and v range) to adjust the FOV (field of view) and to increase the number of iterations. We tested this full setting:

- L = 25 cm;
- $N°$ of microphones = 32 mic;
- K = 31 (FIR length);
- $u \in [-1.\ 5; 1.5]$;
- $v \in [-1.5; 1.5]$;
- $N°$ of iterations = 100;
- Bandwidth = [2000, 6400] Hz.

As expected, low-frequency performance is improved relative to the bandwidth [2, 6.4] kHz 32−elements array. The directivity comparison (Figure 13a) with a 0.21 × 0.21 m^2 prototype in the bandwidth [0.5, 6.4] kHz indicates better sensitivity below 4 kHz. This demonstrates the potential for substantial miniaturization through optimization over the frequency bandwidth. The 32−microphone design retains 15 dB (Figure 13b) robustness, with improved low-frequency gains offsetting minor high-frequency trade-offs. The beam patterns in Figure 14 verify directional selectivity and sidelobe suppression within the band.

The beam patterns in Figure 14 show low sidelobes within the band. However, some aliasing emerges at higher frequencies due to the reduced aperture size. The 10 dB directivity in Figure 13a confirms directional sensitivity is retained over most of the band. Figure 13b indicates above 15 dB robustness to standard fabrication imperfections. In this first simulation, the performance predictions verify that a ≈4× footprint reduction of the array surface from the 0.5 m *Dual Cam* design is plausible with tolerable trade-offs.

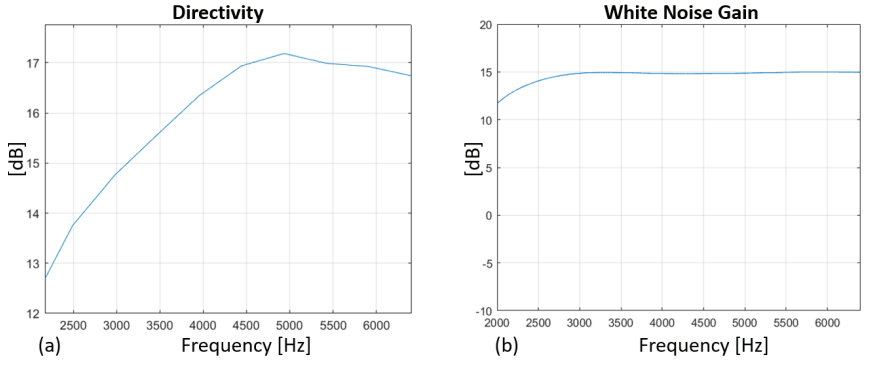


Figure 13. (**a**) Directivity and (**b**) white noise gain metrics confirm reasonable performance across the [2, 6.4] kHz band from the 0.25 m array.

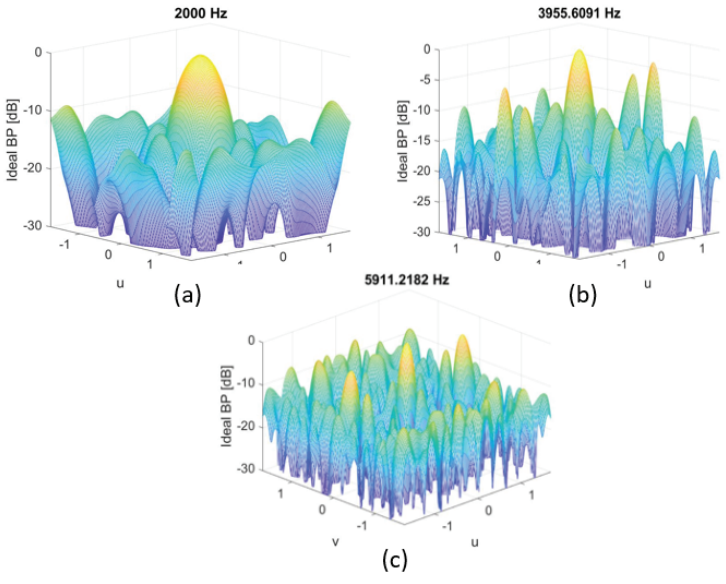


Figure 14. Simulation array 25 × 25 cm^2. Beam patterns within the optimization band and the aperiodic microphone localization exhibit low sidelobes and main lobe focusing (Figure 4). (**a**) 2 kHz, (**b**) ≈4 kHz, (**c**) ≈6 kHz.

7.2. Thirty-Two-Microphones, [2, 6.4] kHz, Array 21 × 21 cm^2

Increasing the number of iterations in the optimization algorithm alone does not fully suppress grating lobes at higher frequencies, even in an optimized field of view. We hypothesize that enlarging the dimensions of the main acoustic lobe provides additional degrees of freedom for the algorithm to minimize secondary grating lobes. A wider main lobe increases the target spatial region for overall sidelobe suppression. However, expanding the filter tap length can sometimes yield unstable and non-convergent solutions. We experimentally evaluate these trade-offs between main lobe width, number of taps (FIR length), and iterations. The following experiments systematically vary lobe parameters and tap length to assess their impact on grating lobe artefacts and algorithm stability. We optimize over an expanded parameter space to determine configurations that maximize grating lobe mitigation while maintaining convergence and solution integrity. This exploration provides practical insights into the interaction between beam pattern specifications, filter design constraints, and robust algorithm convergence for optimal array performance. We report the better results of the following simulation (Figures 15 and 16) in comparison to the previous case:

- L = 21 cm;
- $N°$ of microphones = 32 mic;
- K = 31 (FIR length);
- $u \in [-1.5; 1.5]$; $v \in [-1.41; 1.41]$;
- $N°$ of iterations $\approx 10^5$;
- $uMainLobe_{low} = -0.2$; $uMainLobe_{high} = 0.2$;
- $vMainLobe_{low} = -0.2$; $vMainLobe_{high} = 0.2$;
- Bandwidth = [2000, 6400] Hz.

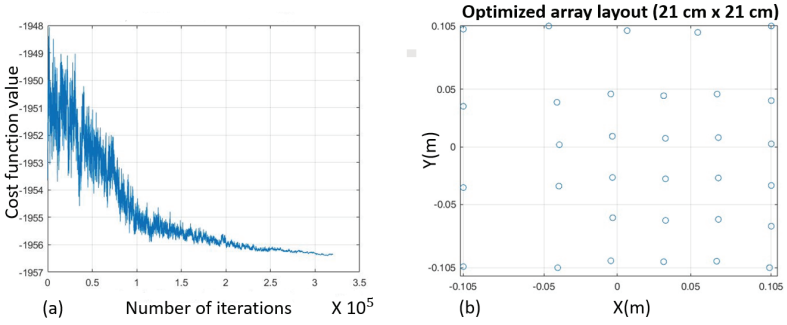


Figure 15. Simulation results for [2, 6.4] kHz optimization of a 32−element 0.21 m acoustic array: (**a**) cost function convergence over $\approx 10^5$ iterations, (**b**) optimized 32−microphone 0.21 m array layout.

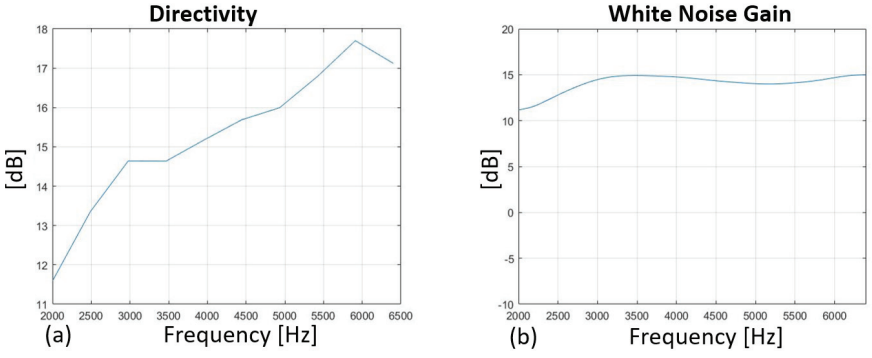


Figure 16. (**a**) Directivity and (**b**) white noise gain metrics confirm reasonable performance across the [2,6.4] kHz band even from the reduced 0.21 m array.

The 0.21 m array in the bandwidth [2000, 6400] Hz achieves better directivity than the 0.21 m array in the full range [500, 6400] Hz below 4 kHz, confirming substantial miniaturization optimized in a sub-range of frequencies is viable. A sub-range of u and v now optimizes the beampatterns avoiding grating lobes at higher frequencies (Figures 17 and 18), even if this action reduces of course the FOV of *Dual Cam* 2.0.

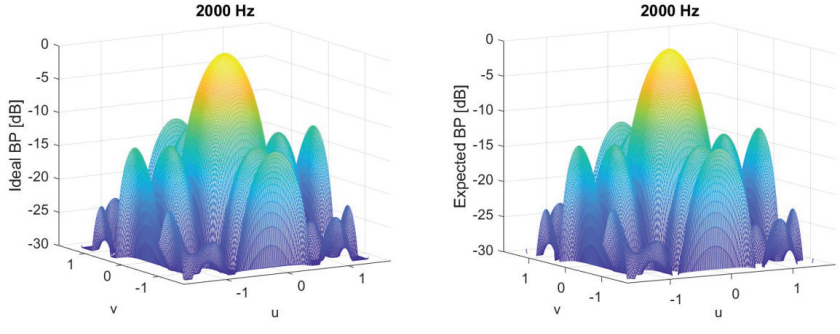


Figure 17. Simulation results for [2, 6.4] kHz optimization of a 32−element 0.21 m acoustic array. Beampatten comparison: BP (**left**) vs EBPP (**right**) at 2 kHz.

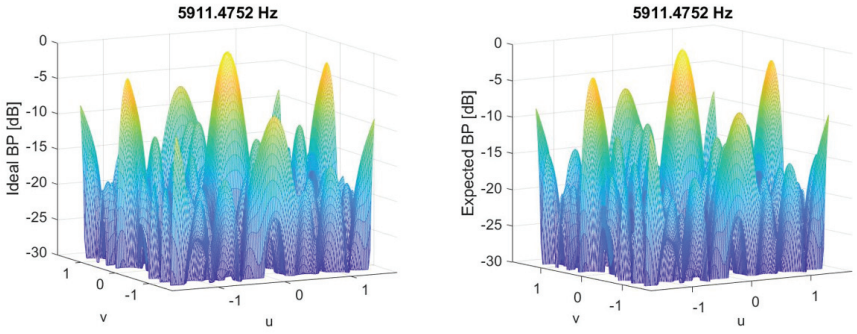


Figure 18. Simulation results for [2, 6.4] kHz optimization of a 32−element 0.21 m acoustic array. Beampatten comparison: BP (**left**) vs EBPP (**right**) at ≈6 kHz.

7.3. Thirty-Two-Microphones, [0.5, 6.4] kHz, 21 × 21 cm^2 Array

This section compares the performance of the current acoustic device *Dual Cam* against a new prototype with shortened dimensions but equivalent bandwidth. We simulate this current experimental condition:

- L = 50 cm;
- $N°$ of microphones = 128 mic;
- K = 7 (FIR length);
- $u \in [-1.5; 1.5]$; $v \in [-1.41; 1.41]$;
- $N°$ of iterations $\approx 10^5$;
- $uMainLobe_{low} = -0.06$; $uMainLobe_{high} = 0.06$;
- $vMainLobe_{low} = -0.06$; $vMainLobe_{high} = 0.06$;
- Bandwidth = [500, 6400] Hz.

Simulations are conducted to analyse the key metrics of white noise gain, directivity patterns, and grating lobes. The results demonstrate the feasibility of achieving a compact form factor while maintaining wideband performance through optimization of design parameters.

In our simulation, the third scenario expands the bandwidth to cover *Dual Cam* 2.0 specifications for comparison with the current device:

- L = 21 cm;
- $N°$ of microphones = 32 mic;
- K = 31 (FIR length);
- $u \in [-1{,}5; 1.5]$; $v \in [-1{,}41; 1.41]$;
- $N°$ *of iterations* $\approx 10^5$;
- $uMainLobe_{low} = -0.2$; $uMainLobe_{high} = 0.2$;
- $vMainLobe_{low} = -0.2$; $vMainLobe_{high} = 0.2$;
- Bandwidth = [500, 6400] Hz.

Compensating for the larger wavelength at 0.5 kHz required shrinking once again the array to 0.21 × 0.21 m^2 to maintain the density with 32 microphones to give the area reduction. The cost convergence in Figure 19a follows a similar trend but more iterations are needed to escape poor local minima. The layout in Figure 19b retains an irregular structure with permutations tailored to the acoustic parameters. The current *Dual Cam* working prototype (Figure 20) has a better WNG and directivity, especially at low frequencies (Figures 21 and 22); also, the main lobe of the BP and EBPP is sharper (Figures 23 and 24). Instead, the grating lobes at high frequencies are more or less the same. The directivity comparison (Figure 13a) with a 0.21 × 0.21 m^2 prototype in the bandwidth [0.5, 6.4] kHz in Figure 21a indicates better sensitivity below 4 kHz. The beam patterns in Figure 14 verify directional selectivity and sidelobe suppression within the

band (Figure 21b). Some tradeoffs are highlighted between miniaturization and resolution. Further work is suggested to refine the simulations and investigate the impacts of varying filter lengths (Figure 25). The simulated performance demonstrates that high-fidelity acoustic imaging over audio frequencies can plausibly be achieved with array apertures $4\times$ smaller than the *Dual Cam* benchmark.

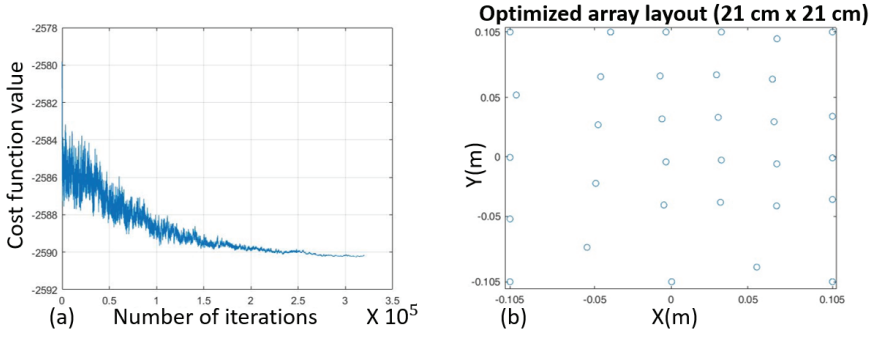


Figure 19. Simulation results for [0.5, 6.4] kHz optimization of a 32−element 0.21 × 0.21 m^2 planar acoustic array: (**a**) cost function convergence over ≈ 10^5 iterations, (**b**) optimized 32−microphone 0.21 m array layout.

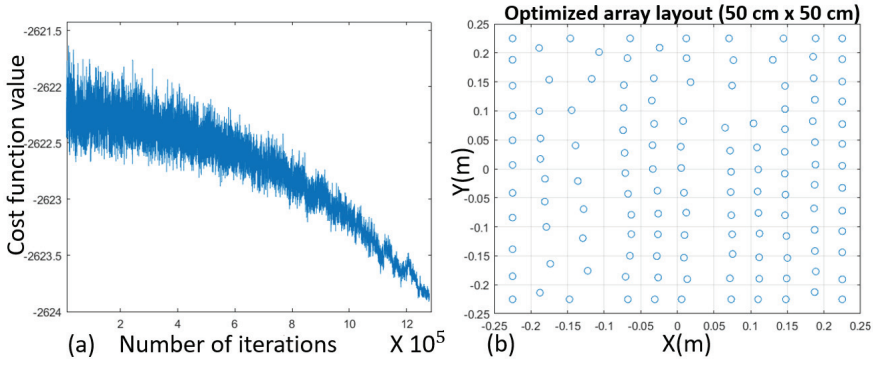


Figure 20. Current *Dual Cam* prototype. Simulation results for [0.5, 6.4] kHz optimization of a 128−element 0.50 × 0.50 m^2 planar acoustic array: (**a**) cost function convergence over ≈10^5 iterations, (**b**) optimized 128−microphone 0.50 m array layout.

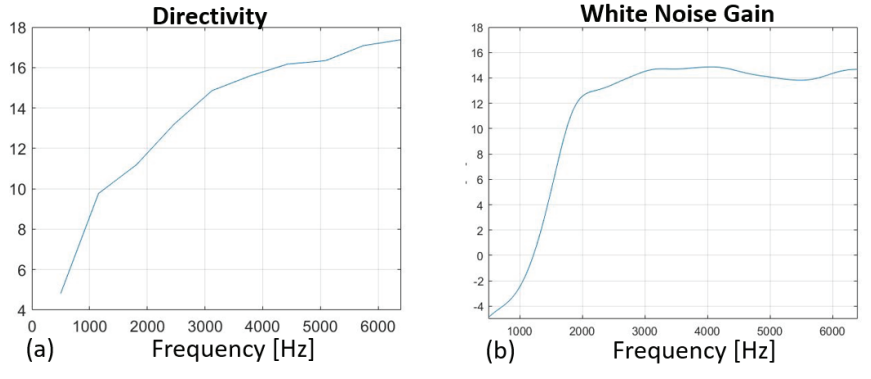


Figure 21. *Dual Cam* 2.0 with larger audio bandwidth [500, 6400] Hz on a 0.21 × 0.21 m^2 array. Directivity (**a**) and WNG (**b**).

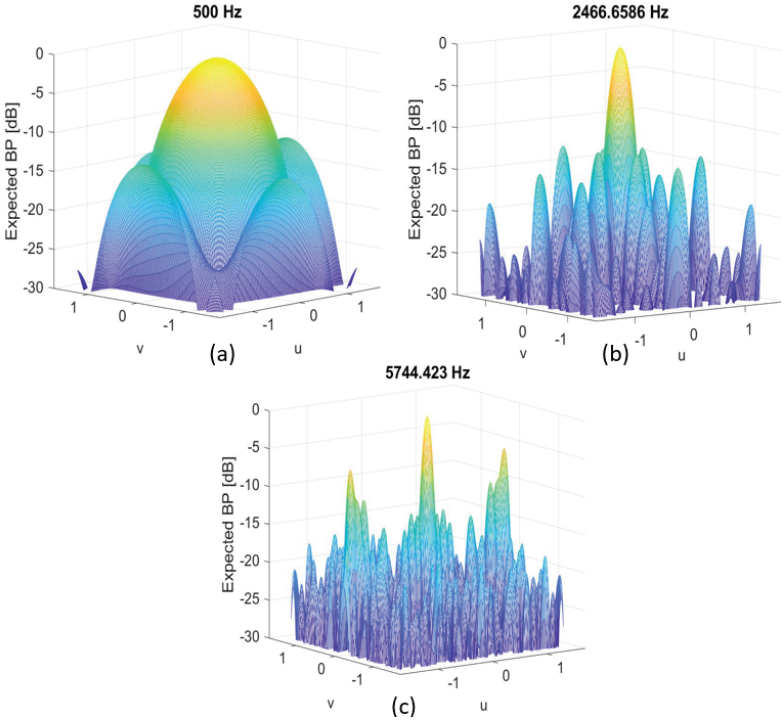


Figure 22. Current *Dual Cam* prototype. Expected beam pattern power at different frequencies: (**a**) 500 Hz, (**b**) ≈2.5 kHz, (**c**) ≈6 kHz.

This supports developing compact prototypes based on these optimized configurations. Ongoing research is focused on the physical implementation of miniaturized arrays guided by these modelling results.

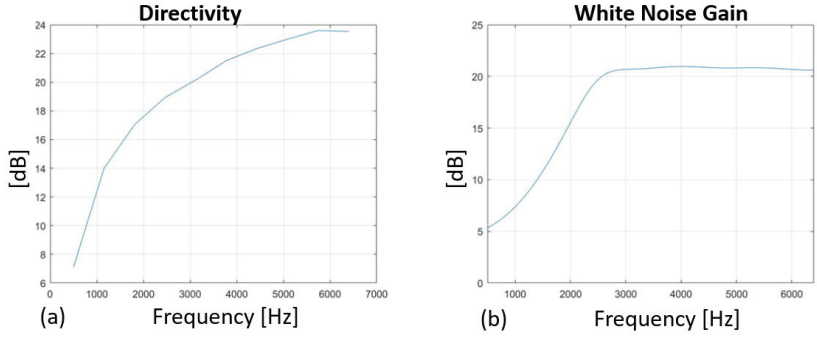


Figure 23. Current *Dual Cam* prototype: (**a**) Directivity and (**b**) WNG over the full bandwidth [500, 6400] Hz on a 50 × 50 cm^2 array.

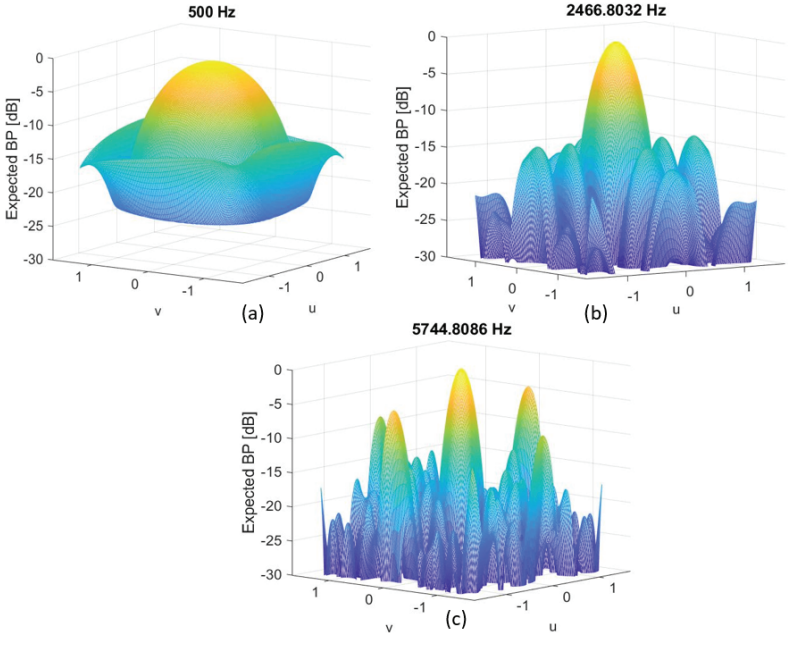


Figure 24. *Dual Cam* 2.0 cted beam pattern power at different frequencies: (**a**) 500 Hz, (**b**) ≈2.5 kHz, (**c**) ≈6 kHz.

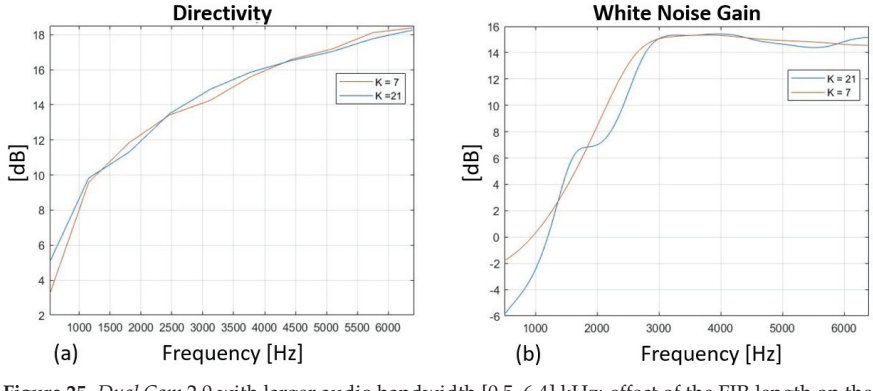


Figure 25. *Dual Cam* 2.0 with larger audio bandwidth [0.5, 6.4] kHz: effect of the FIR length on the metrics of the evaluation. (**a**) Directivity with K = 7 (red) vs K = 21 (blue); (**b**) WNG with K = 7 (red) vs. K = 21 (blue).

8. Hardware Development Considerations

Constructing optimized microphone arrays to realize portable acoustic cameras presents additional implementation challenges including:

- Fabricating irregular array geometries with a large number of elements;
- Microphone calibration and mismatch compensation;
- Embedded platform with multichannel digitization and processing;
- Robust beamforming algorithms executable in real time;
- Packaging, power, and interfacing for field deployment.

The printed circuit board population provides a potential fabrication approach for unconventional layouts. The layout can be rendered as copper traces linking microphone footprints. Micro-electromechanical system (MEMS) technology enables compact sen-

sors [46]. Calibration tools measure each microphone response to derive compensation filters. FPGAs offer parallelism for multichannel acquisition and beamforming [47,48]. Robust and adaptive algorithms help counteract model errors. Energy-efficient architectures would enable battery-powered operation. A USB- or WiFi-linked interface with a smartphone or tablet app could provide deployment flexibility. Ongoing research is focused on addressing these areas to translate the simulated performance gains into practical miniature acoustic cameras for expanded applications. In addition to hardware, robust calibration procedures and beamforming software refinement would help approximate idealized models. User studies assessing video augmented with acoustic imaging for tasks like machine diagnostics can quantify real-world benefits. Developing the signal processing, microphone technologies, and system integration techniques to realize compact acoustic cameras would represent a breakthrough for broader adoption in noise monitoring, condition-based maintenance, virtual reality, and other fields constrained by large form factors today.

9. Conclusions

This paper investigated methodologies to minimize the physical aperture of real-time acoustic cameras for improved mobility.

Acoustic systems rely on an array of microphones to perform directional processing. Reducing the physical aperture of the array typically decreases the angular resolution and introduces grating lobes. However, making the assumption of using higher audio harmonics while maintaining intelligibility without the fundamental frequencies and with careful selection of design parameters, compact arrays may still achieve wideband performance on par with larger arrays. This work investigates this premise through simulation of a current acoustic system and proposed compact prototype. A case study of the *Dual Cam* prototype that utilizes a 0.5×0.5 m^2, 128−microphone planar array to generate acoustic field visualizations in real time revealed limitations around size, weight, power, and computational complexity that restrict widespread adoption. To transform such cameras into portable devices, we proposed co-optimizing the array layout and beamforming filters through simulations to concentrate directional sensitivity and minimize artefacts. Analyses quantified that a 32−element 0.21×0.21 m^2 array optimized for the bandwidth [2, 6.4] kHz operation could theoretically achieve better directivity than the full-scale *Dual Cam* prototype up to 4 kHz, confirming substantial miniaturization is viable with tolerable performance trade-offs. Ongoing efforts are focused on constructing miniature microphone arrays guided by these numerical optimizations to develop hand-held acoustic cameras that interface with tablets and smartphones for easy deployment. Realizing compact, real-time acoustic imaging devices could expand applications in structural health monitoring, urban noise mapping, VR/AR audio rendering, and other fields currently constrained by large form factors. This paper provided an array of signal processing insights to guide physical prototype development towards transforming acoustic imaging capabilities from constrained lab settings into widely accessible mobile platforms. With further progress in microphone technologies, embedded computing, and calibration techniques, ubiquitous acoustic imaging could become viable—providing uniquely valuable spatial and semantic context across applications ranging from the industrial internet of things to smart city sound monitoring (Figure 26).

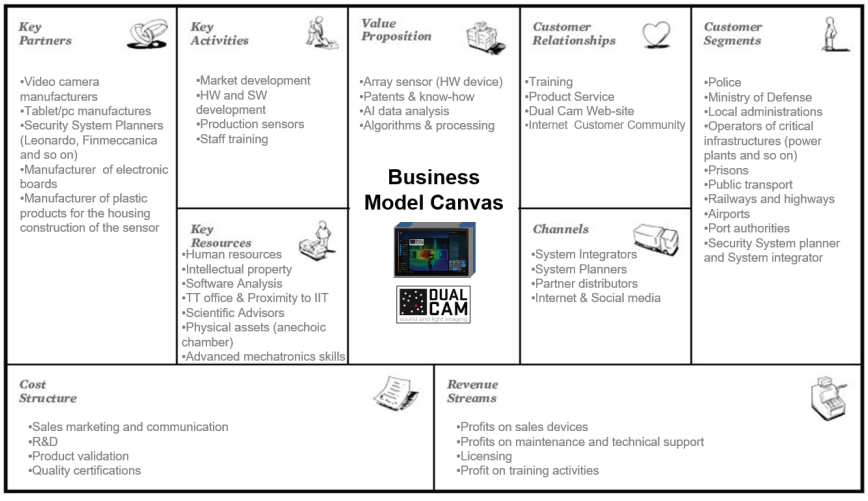


Figure 26. *Dual Cam* 2.0 Business Model Canvas.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Dmochowski, J.P.; Benesty, J. Steered Beamforming Approaches for Acoustic Source Localization. In *Speech Processing in Modern Communication*; Cohen, I., Benesty, J., Gannot, S., Eds.; Springer Topics in Signal Processing; Springer: Berlin/Heidelberg, Germany, 2010; Volume 3. [CrossRef]
2. Rafaely, B. *Fundamentals of Spherical Array Processing*; Springer: Berlin/Heidelberg, Germany, 2019.
3. Van Veen, B.D.; Buckley, K.M. Beamforming: A versatile approach to spatial filtering. *IEEE Assp Mag.* **1988**, *5*, 4–24. [CrossRef] [PubMed]
4. Jombo, G.; Zhang, Y. Acoustic-Based Machine Condition Monitoring—Methods and Challenges. *Eng* **2023**, *4*, 47–79. [CrossRef]
5. Na, Y. (2015). An Acoustic Traffic Monitoring System: Design and Implementation. In Proceedings of the 2015 IEEE 12th International Conference on Ubiquitous Intelligence and Computing and 2015 IEEE 12th International Conference on Autonomic and Trusted Computing and 2015 IEEE 15th International Conference on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom), Beijing, China, 10–14 August 2015. [CrossRef]
6. António, R. On acoustic gunshot localization systems. In Proceedings of the Society for Design and Process Science SDPS-2015, Fort Worth, TX, USA, 1–5 November 2015; pp. 558–565.
7. Lluís, F.; Martínez-Nuevo, P.; Møller, M.B.; Shepstone, S.E. Sound field reconstruction in rooms: Inpainting meets super-resolution. *J. Acoust. Soc. Am.* **2020**, *148*, 649–659. [CrossRef] [PubMed]
8. Yan, J.; Zhang, T.; Broughton-Venner, J.; Huang, P.; Tang, M.-X. Super-Resolution Ultrasound Through Sparsity-Based Deconvolution and Multi-Feature Tracking. *IEEE Trans. Med. Imaging* **2022**, *41*, 1938–1947. [CrossRef]
9. Crocco, M.; Cristani, M.; Trucco, A.; Murino, V. Audio Surveillance: A Systematic Review. *ACM Comput. Surv.* **2016**, *48*, 46. [CrossRef]
10. Crocco, M.; Martelli, S.; Trucco, A.; Zunino, A.; Murino, V. Audio Tracking in Noisy Environments by Acoustic Map and Spectral Signature. *IEEE Trans. Cybern.* **2018**, *48*, 1619–1632. [CrossRef]
11. Worley, R.; Dewoolkar, M.; Xia, T.; Farrell, R.; Orfeo, D.; Burns, D.; Huston, D. Acoustic Emission Sensing for Crack Monitoring in Prefabricated and Prestressed Reinforced Concrete Bridge Girders. *J. Bridge Eng.* **2019**, *24*, 04019018. [CrossRef]
12. Bello, J.P.; Silva, C.; Nov, O.; Dubois, R.L.; Arora, A.; Salamon, J.; Mydlarz, C.; Doraiswamy, H. SONYC: A system for monitoring, analyzing, and mitigating urban noise pollution. *Commun. ACM* **2019**, *62*, 68–77. [CrossRef]
13. Sun, X. Immersive audio, capture, transport, and rendering: A review. *Apsipa Trans. Signal Inf. Process.* **2021**, *10*, e13. [CrossRef]
14. Zhang, H.; Wang, J.; Li, Z.; Li, J. Design and Implementation of Two Immersive Audio and Video Communication Systems Based on Virtual Reality. *Electronics* **2023**, *12*, 1134. [CrossRef]
15. Padois, T.; St-Jacques, J.; Rouard, K.; Quaegebeur, N.; Grondin, F.; Berry, A.; Nélisse, H.; Sgard, F.; Doutres, O. Acoustic imaging with spherical microphone array and Kriging. *JASA Express Lett.* **2023**, *3*, 042801. [CrossRef]

16. Chu, Z.; Yin, S.; Yang, Y.; Li, P. Filter-and-sum based high-resolution CLEAN-SC with spherical microphone arrays. *Appl. Acoust.* **2021**, *182*, 108278. [CrossRef]
17. Ward, D.B.; Kennedy, R.A.; Williamson, R.C.; Brandstein, M. *Microphone Arrays Signal Processing Techniques and Applications (Digital Signal Processing)*; Springer: Berlin/Heidelberg, Germany, 2001.
18. Crocco, M.; Trucco, A. Design of Superdirective Planar Arrays With Sparse Aperiodic Layouts for Processing Broadband Signals via 3-D Beamforming. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2014**, *22*, 800–815. [CrossRef]
19. Crocco, M.; Trucco, A. Stochastic and Analytic Optimization of Sparse Aperiodic Arrays and Broadband Beamformers WITH Robust Superdirective Patterns. *IEEE Trans. Audio Speech Lang. Process.* **2012**, *20*, 2433–2447. [CrossRef]
20. Mars, R.; Reju, V.G.; Khong, A.W.H.; Hioka, Y.; Niwa, K. *Chapter 12—Beamforming Techniques Using Microphone Arrays*; Chellappa, R., Theodoridis, S., Eds.; Academic Press Library in Signal Processing; Academic Press: Cambridge, MA, USA, 2018; Volume 7, pp. 585–612. ISBN 9780128118870. [CrossRef]
21. Hansen, R.C. *Phased Array Antennas*, 2nd ed.; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2009; ISBN: 978-0-470-40102-6.
22. Cox, H.; Zeskind, R.; Kooij, T. Practical supergain. *IEEE Trans. Acoust. Speech, Signal Process.* **1986**, *34*, 393–398. [CrossRef]
23. Kates, J.M. Superdirective arrays for hearing aids. *J. Acoust. Soc. Am.* **1993**, *94*, 1930–1933. [CrossRef] [PubMed]
24. Bitzer, J.; Simmer, K.U. Superdirective microphone arrays. In *Microphone Arrays: Signal Processing Techniques and Applications*; Brandstein, M.S., Ward, D.B., Eds.; Springer: New York, NY, USA, 2001; pp. 19–38.
25. Van Trees, H.L. *Optimum Array Processing: Part IV of Detection, Estimation, and Modulation Theory*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2002.
26. Ward, D.B.; Kennedy, R.A.; Williamson, R.C. Theory and design of broadband sensor arrays with frequency invariant far-field beam patterns. *J. Acoust. Soc. Am.* **1995**, *97*, 1023–1034. [CrossRef]
27. Crocco, M.; Trucco, A. Design of Robust Superdirective Arrays With a Tunable Tradeoff Between Directivity and Frequency-Invariance. *IEEE Trans. Signal Process.* **2011**, *59*, 2169–2181. [CrossRef]
28. Doclo, S. Multi-microphone noise reduction and dereverberation techniques for speech applications. Ph.D. Thesis, Katholieke Universiteit Leuven, Leuven, Belgium, 2003. Available online: ftp://ftp.esat.kuleuven.ac.be/stadius/doclo/phd/ (accessed on 14 June 2023).
29. Crocco, M.; Trucco, A. The synthesis of robust broadband beamformers for equally-spaced linear arrays. *J. Acoust. Soc. Am.* **2010**, *128*, 691–701. [CrossRef]
30. Mabande, E. Robust Time-Invariant Broadband Beamforming as a Convex Optimization Problem. Ph.D. Thesis, Friedrich-Alexander-Universität, Erlangen, Germany, 2015. Available online: https://opus4.kobv.de/opus4-fau/frontdoor/index/index/year/2015/docId/6138/ (accessed on 14 June 2023)
31. Trucco, A.; Traverso, F.; Crocco, M. Robust superdirective end-fire arrays. In Proceedings of the 2013 MTS/IEEE OCEANS—Bergen, Bergen, Norway, 10–13 June 2013; pp. 1–6. [CrossRef]
32. Traverso, F.; Crocco, M.; Trucco, A. Design of frequency-invariant robust beam patterns by the oversteering of end-fire arrays. *Signal Process.* **2014**, *99*, 129–135. [CrossRef]
33. Trucco, A.; Crocco, M. Design of an Optimum Superdirective Beamformer Through Generalized Directivity Maximization. *IEEE Trans. Signal Process.* **2014**, *62*, 6118–6129. [CrossRef]
34. Trucco, A.; Traverso, F.; Crocco, M. Broadband performance of superdirective delay-and-sum beamformers steered to end-fire. *J. Acoust. Soc. Am.* **2014**, *135*, EL331–EL337. [CrossRef] [PubMed]
35. Doclo, S.; Moonen, M. Superdirective Beamforming Robust Against Microphone Mismatch. *IEEE Trans. Audio, Speech, Lang. Process.* **2007**, *15*, 617–631. [CrossRef]
36. Crocco, M.; Trucco, A. A Synthesis Method for Robust Frequency-Invariant Very Large Bandwidth Beamforming. In Proceedings of the 18th European Signal Processing Conference (EUSIPCO 2010), Aalborg, Denmark, 23–27 August 2010; pp. 2096–2100.
37. Trucco, A.; Crocco, M.; Traverso, F. Avoiding the imposition of a desired beam pattern in superdirective frequency-invariant beamformers. In Proceedings of the 26th Annual Review of Progress in Applied Computational Electromagnetics, Tampere, Finland, 26–29 April 2010; pp. 952–957.
38. Doclo, S.; Moonen, M. Design of broadband beamformers robust against gain and phase errors in the microphone array characteristics. *IEEE Trans. Signal Process.* **2003**, *51*, 2511–2526. [CrossRef]
39. Mabande, E.; Schad, A.; Kellermann, W. Design of robust superdirective beamformers as a convex optimization problem. In Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing, Taipei, Taiwan, 19–24 April 2009; pp. 77–80. [CrossRef]
40. Greco, D.; Trucco, A. Superdirective Robust Algorithms' Comparison for Linear Arrays. *Acoustics* **2020**, *2*, 707–718. [CrossRef]
41. Pérez, A.F.; Sanguineti, V.; Morerio, P.; Murino, V. Audio-Visual Model Distillation Using Acoustic Images. In Proceedings of the 2020 IEEE Winter Conference on Applications of Computer Vision (WACV), Snowmass, CO, USA, 1–5 March 2020; pp. 2843–2852. [CrossRef]
42. Sanguineti, V.; Morerio, P.; Pozzetti, N.; Greco, D.; Cristani, M.; Murino, V. Leveraging acoustic images for effective self-supervised audio representation learning. In Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Part XXII 16, pp. 119–135.
43. Ward, D.B.; Kennedy, R.A.; Williamson, R.C. FIR filter design for frequency invariant beamformers. *IEEE Signal Process. Lett.* **1996**, *3*, 69–71. [CrossRef]

44. Sanguineti, V.; Morerio, P.; Bue, A.D.; Murino, V. Unsupervised Synthetic Acoustic Image Generation for Audio-Visual Scene Understanding. *IEEE Trans. Image Process.* **2022**, *31*, 7102–7115. [CrossRef]
45. Kirkpatrick, S.; Gelatt, C.D., Jr.; Vecchi, M.P. Optimization by Simulated Annealing. *Science* **1983**, *220*, 671–680. [CrossRef]
46. Chae, M.S.; Yang, Z.; Yuce, M.R.; Hoang, L.; Liu, W. A 128-channel 6 mW wireless neural recording IC with spike feature extraction and UWB transmitter. *IEEE Trans. Neural Syst. Rehabil. Eng*. **2009**, *17*, 312–321. [CrossRef]
47. Paolo, M.; Alessandro, C.; Gianfranco, D.; Michele, B.; Francesco, C.; Davide, R.; Luigi, R.; Luca, B. Neuraghe: Exploiting CPU-FPGA synergies for efficient and flexible CNN inference acceleration on zynQ SoCs. *ACM Trans. Reconfigurable Technol. Syst*. **2017**, *11*, 1–24. [CrossRef]
48. Da Silva, B.; Braeken, A.; Touhafi, A. FPGA-Based Architectures for Acoustic Beamforming with Microphone Arrays: Trends, Challenges and Research Opportunities. *Computers* **2018**, *7*, 41. [CrossRef]

applied sciences

MDPI

Article

A Stable Sound Field Control Method for a Personal Audio System

Song Wang [1] and Cong Zhang [2,*]

[1] School of Mathematics and Computer Science, Wuhan Polytechnic University, Wuhan 430048, China; wnsg2001@163.com
[2] School of Electrical and Electronic Engineering, Wuhan Polytechnic University, Wuhan 430048, China
* Correspondence: hb_wh_zc@163.com

Abstract: A personal audio system has a wide application prospect in people's lives, which can be implemented by sound field control technology. However, the current sound field control technology is mainly based on sound pressure or its improvement, ignoring another physical property of sound: particle velocity, which is not conducive to the stability of the entire reconstruction system. To address the problem, a sound field method is constructed in this paper, which minimizes the reconstruction error in the bright zone, minimizes the loudspeaker array effort in the reconstruction system, and at the same time controls the particle velocity and sound pressure of the dark zone. Five unevenly placed loudspeakers were used as the initial setup for the computer simulation experiment. Simulation results suggest that the proposed method is better than the PM (pressure matching) and EDPM (eigen decomposition pseudoinverse method) methods in the bright zone in an acoustic contrast index, the ACC (acoustic contrast control) method in a reconstruction error index, and the ACC, PM, and EDPM methods in the bright zone in a loudspeaker array effort index. The average array effort of the proposed method is the smallest, which is about 9.4790, 8.0712, and 4.8176 dB less than that of the ACC method, the PM method in the bright zone, and the EDPM method in the bright zone, respectively, so the proposed method can produce the most stable reconstruction system when the loudspeaker system is not evenly placed. The results of computer experiments demonstrate the performance of the proposed method, and suggest that compared with traditional methods, the proposed method can achieve more balanced results in the three indexes of acoustic contrast, reconstruction error, and loudspeaker array effort on the whole.

Keywords: personal audio system; sound field control; acoustic contrast; reconstruction error; array effort

Citation: Wang, S.; Zhang, C. A Stable Sound Field Control Method for a Personal Audio System. *Appl. Sci.* **2023**, *13*, 12209. https://doi.org/10.3390/app132212209

Academic Editors: Giovanni Costantini and Daniele Casali

Received: 25 September 2023
Revised: 1 November 2023
Accepted: 6 November 2023
Published: 10 November 2023

1. Introduction

Based on multizone sound field reconstruction technologies, a personal audio system can be used to concentrate sound effects in a listening area without affecting listeners in other areas. A bright zone and a dark zone can be generated by multizone sound field reconstruction techniques. The desired sound pressure field could be generated in the bright zone, while the sound pressure level is attenuated in the dark zone. Personal audio systems can be used in game systems, audio system exhibits, home theaters, car audio systems, etc. [1]. With the development of sound field reconstruction technology, more and more people pay attention to personal audio systems. Many researchers have put forward many new research methods and techniques for a personal audio system.

The acoustic contrast control (ACC) method was introduced by Choi et al., which solves loudspeakers' allocation coefficients by maximizing the acoustic contrast between the bright zone and the dark zone [2]. However, the ACC method does not control the phase, and its energy is unevenly distributed in the bright zone, which will affect the subjective listening experience of listeners. The pressure matching (PM) method can solve

these problems by minimizing the error between the desired sound pressure and the reconstructed sound pressure at control points [3]. Additionally, the planarity control (PC) method was proposed [4], which filters the sound pressure at the control points spatially, and maximizes the energy of the bright zone under the constraint of the dark zone energy based on the super-directional beamforming. The advantage of the PM method and the PC method is that they can generate a directional sound field in the range of the bright zone, while the disadvantage of the PM method and the PC method is that the acoustic contrasts generated by them are smaller than that of the ACC method. Shin et al. introduced a method that maximizes the acoustic energy difference, which changes the focus of the ACC method by focusing on the ratio of the mean square sound pressure difference between the bright and dark zones. The comparison experiments with the ACC method show that this method can maintain the sound pressure difference between two listening zones and improve the radiation efficiency of sound space. In addition, this method is simple to calculate [5]. Chang et al. tried to take advantage of the ACC method and the PM method and proposed the ACC–PM method by combining these two methods. By selecting the appropriate adjustment factor, the ACC–PM method can obtain the tradeoff effect between the bright zone reconstruction error and acoustic contrast [6]. Based on the PM approach, Olivieri et al. proposed a beamforming approach to control the tradeoff between listener position sound field quality and directivity performance. This proposed method selects control points that contribute to the PM cost function in a frequency-dependent manner according to the angular distance between control points relative to a given wavelength [7]. Based on the PM method, Afghah et al. proposed the eigen decomposition pseudoinverse method (EDPM) [8]. This method constructs a regularization strategy to solve the pseudoinverse of ill-conditioned matrices, replacing the Tikhonov regularization method traditionally used in the PM method. This method is used to improve the performance at dark points without generating artifacts at bright points. In view of the robustness and regularization of the ACC method, Elliott et al. proposed that robustness could be enhanced by imposing limits on array effort. Robustness can generally be improved by a regularization method, but good regularization parameters are often difficult to choose [9]. Zhu et al. proposed a robust sound zone reconstruction design framework that solves the parameters by acoustic modeling. However, they did not examine the relationship between different loudspeaker directivities and robust optimization and explore the accuracy required for robust ACC acoustic modeling [10]. Han et al. proposed a method for acoustic contrast control in a wave domain for three dimension cases. The three-dimensional sound field is represented by spherical harmonic decomposition, and the sound energy of the interested region is calculated. The three-dimensional multizone sound field is reconstructed by using a planar circular loudspeaker array instead of a spherical one. Experiments show that this method is better than the ACC method in high-frequency acoustic contrast performance [11]. The time-domain ACC method (TACC) is favored because it can optimize the entire bandwidth by one step, but the disadvantage of TACC is that the frequency response is not uneven. In order to explain this problem theoretically, Hu et al. constructed a progressively equivalent form of TACC in frequency domain, and demonstrated that TACC has an inclination to fetch frequency constituents of the largest contrast [12]. By considering the signal feature and human auditory system, Lee et al. introduced a sound zone control frame called perception VAST. The experiments suggest that the perception VAST method performs better than the ACC and PM methods in terms of the perceptive index of STOI and PESQ [13]. Then, in order to make the leakage error under a fixed acoustic contrast as imperceptible as possible, they extended this work to an adaptive case and proposed a time-domain sound region control method using a variable span tradeoff filter [14]. To reduce computational complexity, they then proposed two VAST framework-based approaches (narrowband approach and broadband approach) in the discrete Fourier transform domain [15]. Ryu et al. proposed a personal audio effect controller whose effects change linearly with the customer's input. The weight trajectories of loudspeakers are modeled as continuous functions based on piecewise linear approximations of the

performance variation. The experimental results show that the controller can regulate the system performance linearly [16]. Hu et al. introduced a sound field control method in a time domain based on sound pressure and particle velocity for single-zone sound field reconstruction. In this method, the eigenvalue-decomposition-based way and the conjugate-gradient way are used to decrease the complexity of computing [17]. In order to reduce the requirements on microphone placement geometry and a control sound field in the whole region, Du et al. proposed a two-region 2D sound field reproduction approach based on equivalent source and the ACC method. The goal of this method is to maximize the acoustic energy difference between the two regions [18]. Additionally, Du et al. introduced another two-region sound field reconstruction method based on equivalent source and the PM and ACC methods. Compared with the traditional method, sound field control in a dark region by Du's method is not limited to several points at the boundary of the dark zone, but extends to the whole dark zone [19]. The number and location of loudspeakers have important influence on the reconstruction effect in multizone sound field reconstruction. When people perform multizone sound field reconstruction, people often arrange the loudspeakers in a circular, linear, and curved fashion, but these arrangements are obtained by experience. To solve the problem, Zhu et al. introduced a method to select the required number of loudspeaker locations from candidate loudspeaker locations iteratively [20]. Then, enlightened by the theory in [21], Zhao et al. proposed an evolutionary optimization approach based on a candidate location set to optimize loudspeaker array placement [22]. Different from the iterative method, this method eliminates one loudspeaker from the loudspeaker candidate set at each iteration. Zhong et al. investigated the feasibility of using multiparameter array loudspeakers to remotely generate a quiet zone in a free sound field. They obtained the relationship between the size of the quiet zone and the number of secondary sound sources and the wavelength of secondary sound sources in two and three dimension cases [23]. Abe et al. proposed an amplitude matching algorithm for multizone sound field control, which produces the expected amplitude distribution in the designated region by alternating the direction method of multipliers, without paying attention to phase distribution [24].

There are also some experts who have studied the application of personal audio systems. Elliott et al. studied the application of the ACC method to a headrest [25]. Cheer et al. conducted some research on the application of personal audio systems to mobile phones [26]. Cheer et al. also put personal audio systems into a car with two loudspeaker arrays [27]. Different from the method described in the literature [27], Liao et al. proposed to fix loudspeaker arrays on the ceiling of a car compartment in order to produce separate high-frequency listening regions in the front and rear seats of the car compartment [28]. They also investigated the geometric size design method for ceiling-mounted loudspeaker arrays and target sound pressures' selection method. Compared with the system in the literature [27], the system proposed by Liao et al. shows obvious performance improvement in the frequency range of 1–4 kHz. Based on an line loudspeaker array mounted on a flat-screen TV in the form of a bar loudspeaker, Choi proposed two kinds of sound field control systems for home applications that are real-time. One kind of system is a personal audio system that makes users at different positions experience independent sound effects by reducing the mutual interference of different sound zones; another kind of system can exert an influence on the spatial audio scene [29].

The sound property considered by most existing sound field control technologies is sound pressure. Additionally, many of the sound field control technologies proposed by researchers are improved techniques based on sound pressure. Sound pressure and particle velocity can be used to describe sound [30]. Additionally, it is pointed out in [31] that when loudspeakers are not evenly placed, the loudspeaker signal obtained by controlling the sound pressure of a single region is less stable than the loudspeaker signal obtained by controlling the particle velocity of a single region. Because in real life, many loudspeaker arrays are not evenly placed, particle velocity should be considered in addition to sound pressure in the design of sound field control methods. On the basis of traditional methods,

a sound field control method based on sound pressure and particle velocity is introduced in order to reconstruct a sound field in the bright zone better, reduce acoustic energy in the dark zone, and pay attention to the stability of the system.

The main contributions of this paper are as follows: based on the traditional sound field control technology, a new sound field control method is proposed by introducing particle velocity. The proposed method attempts to minimize the reconstruction error in the bright zone and minimize the loudspeaker array effort while controlling the sound pressure and particle velocity in the dark zone. The proposed method attempts to control or optimize acoustic contrast, reconstruction error, and array effort at the same time. The model of the proposed method contains three weight factors; this paper introduces their functions and gives their selection methods. The content of this paper is arranged as follows: Section 1 mainly describes the development status of sound field control technology. Section 2 describes three existing sound field control technologies and analyzes their advantages and disadvantages. Section 3 introduces the model of this new method and parameter selection method of the model. Section 4 introduces the comparison experiment between traditional methods and the proposed method, and analyzes and discusses the results. Section 5 gives the conclusions.

2. Three Traditional Sound Field Control Methods

This part introduces three traditional acoustic field reconstruction methods: ACC [2], PM [3], and EDPM [8]. The ACC and PM methods in the bright zone are relatively important methods and have important influence in the field of sound field control. Acoustic contrast and reconstruction error are important indicators to measure the effect of sound field control. The ACC method is used to maximize acoustic contrast, so it has the maximum acoustic contrast. The PM method in the bright zone works to minimize the reconstruction error in the bright zone, so it has minimal reconstruction error. The ACC method and the PM method in the bright zone are the best methods in a certain index, and compared with them, the advantages and disadvantages of the proposed methods can be better displayed. The EDPM method in the bright zone is an improvement of the PM method in the bright zone, and they are essentially the same class of methods that try to minimize the reconstruction error in the bright zone. The EDPM method in the bright zone is relatively new compared with the PM method in the bright zone, so it is necessary to compare the proposed method with the EDPM method in the bright zone.

2.1. ACC Method [2]

The ratio of the sound potential energy density of the bright region to the sound potential energy density of the dark region is the definition of acoustic contrast. The goal of the ACC method is to make the acoustic contrast to obtain the maximum value. The formula for calculating acoustic contrast is

$$\phi = \frac{\frac{1}{Z_1}\int_{Z_b} \boldsymbol{p}_b^H \boldsymbol{p}_b dz}{\frac{1}{Z_2}\int_{Z_d} \boldsymbol{p}_d^H \boldsymbol{p}_d dz} = \frac{\boldsymbol{q}^H\left(\frac{1}{Z_1}\int_{Z_b} \boldsymbol{G}_b^H \boldsymbol{G}_b dz\right)\boldsymbol{q}}{\boldsymbol{q}^H\left(\frac{1}{Z_2}\int_{Z_d} \boldsymbol{G}_d^H \boldsymbol{G}_d dz\right)\boldsymbol{q}} = \frac{\boldsymbol{q}^H \boldsymbol{W}_b \boldsymbol{q}}{\boldsymbol{q}^H \boldsymbol{W}_d \boldsymbol{q}} \tag{1}$$

where the bright zone is labeled Z_b, and the dark zone is labeled Z_d. The bright zone volume is labeled Z_1, and the dark zone volume is labeled Z_2. $\boldsymbol{p_b}$ is the sound pressure in the bright zone, and $\boldsymbol{p_d}$ is the sound pressure in the dark zone as shown in Figure 1.

$$\begin{aligned} \boldsymbol{p_b} &= \boldsymbol{G_b q} \\ \boldsymbol{p_d} &= \boldsymbol{G_d q} \end{aligned} \tag{2}$$

$\boldsymbol{q}$ is source strengths, which is the $M \times 1$ vector. $\boldsymbol{G_b}$ is the sound pressure transfer function vector in the bright zone,which is a $1 \times M$ dimension, and $\boldsymbol{G_d}$ is the sound pressure transfer function vector in the dark zone, which also is a $1 \times M$ dimension. M is the number of sources

or loudspeakers. Spatial correlation matrices in the bright zone and the dark zone are labeled $\boldsymbol{W_b}$ and $\boldsymbol{W_d}$, respectively.

The acoustic contrast ϕ reaches its maximum value when the eigenvector associated with the maximum eigenvalue of $\boldsymbol{W_d^{-1}W_b}$ is equal to the source strengths. The source strengths at this point are the optimal source strengths.

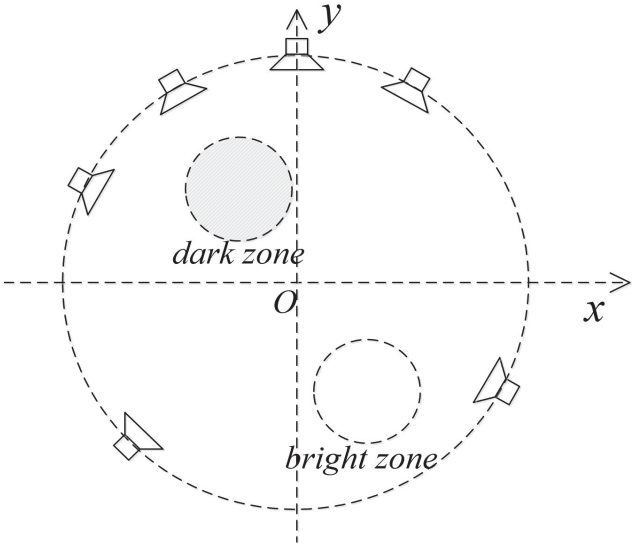


Figure 1. Loudspeaker array and sound zone diagram.

2.2. PM Method [3]

As shown in Figure 1, it is assumed that M loudspeakers $\overrightarrow{l_1}, \overrightarrow{l_2}, \ldots, \overrightarrow{l_M}$ (they also indicate the location of loudspeakers) are placed on the same ring, with s sampling points $\overrightarrow{b_1}, \overrightarrow{b_2}, \ldots, \overrightarrow{b_s}$ (they also indicate the location of sampling points) in the bright zone and t sampling points $\overrightarrow{d_1}, \overrightarrow{d_2}, \ldots, \overrightarrow{d_t}$ in the dark zone. We suppose that the sound pressure produced by the original source in the bright and the dark zone, respectively, are

$$\begin{aligned} \boldsymbol{p_{bo}} &= \left(p(\overrightarrow{b_1}), p(\overrightarrow{b_2}), \ldots, p(\overrightarrow{b_s})\right)^T \\ \boldsymbol{p_{do}} &= \delta\left(p(\overrightarrow{d_1}), p(\overrightarrow{d_2}), \ldots, p(\overrightarrow{d_t})\right)^T \end{aligned} \tag{3}$$

where δ is the amplitude modulation factor. Suppose that the sound pressure generated by M loudspeakers at point $\overrightarrow{b}$ ($\overrightarrow{b} \in \{\overrightarrow{b_1}, \overrightarrow{b_2}, \ldots, \overrightarrow{b_s}\}$) in the bright zone and at point $\overrightarrow{d}$ ($\overrightarrow{d} \in \{\overrightarrow{d_1}, \overrightarrow{d_2}, \ldots, \overrightarrow{d_t}\}$) in the dark zone are, respectively,

$$\begin{aligned} p_{br} &= \sum_{n=1}^{M} h(\overrightarrow{l_n}, \overrightarrow{b}) q_n \\ p_{dr} &= \sum_{n=1}^{M} h(\overrightarrow{l_n}, \overrightarrow{d}) q_n \end{aligned} \tag{4}$$

The sound pressure transfer function is labeled $h(\overrightarrow{l_n}, \overrightarrow{b})$ [32] or $h(\overrightarrow{l_n}, \overrightarrow{d})$, where

$$h(\overrightarrow{l_n}, \overrightarrow{b}) = \frac{e^{-ik|\overrightarrow{b} - \overrightarrow{l_n}|}}{4\pi|\overrightarrow{b} - \overrightarrow{l_n}|} \tag{5}$$

The strength of the n-th loudspeaker is labeled q_n. Formula (4) can be expressed as matrices:

$$\begin{aligned} \boldsymbol{p_{br}} &= \boldsymbol{H_b q} \\ \boldsymbol{p_{dr}} &= \boldsymbol{H_d q} \end{aligned} \tag{6}$$

where

$$\boldsymbol{H_b} = \begin{pmatrix} h(\overrightarrow{l_1}, \overrightarrow{b_1}) & h(\overrightarrow{l_2}, \overrightarrow{b_1}) & \dots & h(\overrightarrow{l_M}, \overrightarrow{b_1}) \\ h(\overrightarrow{l_1}, \overrightarrow{b_2}) & h(\overrightarrow{l_2}, \overrightarrow{b_2}) & \dots & h(\overrightarrow{l_M}, \overrightarrow{b_2}) \\ \dots & \dots & \dots & \dots \\ h(\overrightarrow{l_1}, \overrightarrow{b_s}) & h(\overrightarrow{l_2}, \overrightarrow{b_s}) & \dots & h(\overrightarrow{l_M}, \overrightarrow{b_s}) \end{pmatrix} \tag{7}$$

$$\boldsymbol{H_d} = \begin{pmatrix} h(\overrightarrow{l_1}, \overrightarrow{d_1}) & h(\overrightarrow{l_2}, \overrightarrow{d_1}) & \dots & h(\overrightarrow{l_M}, \overrightarrow{d_1}) \\ h(\overrightarrow{l_1}, \overrightarrow{d_2}) & h(\overrightarrow{l_2}, \overrightarrow{d_2}) & \dots & h(\overrightarrow{l_M}, \overrightarrow{d_2}) \\ \dots & \dots & \dots & \dots \\ h(\overrightarrow{l_1}, \overrightarrow{d_t}) & h(\overrightarrow{l_2}, \overrightarrow{d_t}) & \dots & h(\overrightarrow{l_M}, \overrightarrow{d_t}) \end{pmatrix} \tag{8}$$

$$\boldsymbol{q} = (q_1, q_2, \dots, q_M)^T \tag{9}$$

If we want to recover the sound pressure produced by the original source in the bright zone, the equation constructed by the PM method is

$$\boldsymbol{p_{bo}} = \boldsymbol{p_{br}} \tag{10}$$

Formula (10) can also be expressed as

$$\boldsymbol{p_{bo}} = \boldsymbol{H_b q} \tag{11}$$

The solution of Equation (11) is $\boldsymbol{q} = \boldsymbol{H_b^{-1} p_{bo}}$; the inverse of a matrix is labeled -1. We call this method the PM method in the bright zone.

If we want to recover the pressure generated by the original sound source in both the bright and dark zones, the equation constructed by the PM method is

$$\boldsymbol{p} = \boldsymbol{H q} \tag{12}$$

Formula (12) can also be written as

$$\begin{pmatrix} p_{bo} \\ p_{do} \end{pmatrix} = \begin{pmatrix} \boldsymbol{H_b} \\ \boldsymbol{H_d} \end{pmatrix} \boldsymbol{q} \tag{13}$$

The solution of Formula (12) can be obtained by $\boldsymbol{q} = \boldsymbol{H}^{-1}\boldsymbol{p}$.

2.3. EDPM Method [8]

The EDPM method is described in detail in [8]. Here, we mainly introduce the EDPM method in the bright zone, which is similar to the EDPM method. The eigen decomposition theory tells us that a square matrix $\boldsymbol{A}$ can be decomposed as follows:

$$\boldsymbol{A} = \boldsymbol{B\Lambda B^{-1}} \tag{14}$$

where $\boldsymbol{B}$ is a matrix of columnwise eigenvectors, $\boldsymbol{\Lambda}$ is a diagonal matrix of eigenvalues, and its main diagonal elements are eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_n, \dots$ When Tikhonov regularization is applied to the matrix $\boldsymbol{A}$, we can obtain

$$(\boldsymbol{A} + \epsilon I)^{-1} = \boldsymbol{B}(\boldsymbol{\Lambda} + \epsilon I)^{-1}\boldsymbol{B^{-1}} \tag{15}$$

Inspired by Tikhonov regularization, the pseudoinverse of $(\boldsymbol{A}+\epsilon I)^{-1}$ can be replaced as follows:

$$(\boldsymbol{A}+\epsilon I)^{-1} = \boldsymbol{B}\boldsymbol{\Lambda'^{-1}}\boldsymbol{B^{-1}} \tag{16}$$

$$\lambda'_n = \begin{cases} \lambda_n, & if\ \lambda_n > \alpha \\ \alpha, & else \end{cases} \tag{17}$$

where α is a scalar and moderator; it goes from 0 to infinity. The main diagonal elements of $\boldsymbol{\Lambda'}$ are $\lambda'_1, \lambda'_2, \ldots, \lambda'_n, \ldots$

The solution of Formula (11) can be obtained by Tikhonov regularization:

$$\boldsymbol{q} = \begin{cases} (\boldsymbol{H_b^H H_b}+\epsilon I)^{-1}\boldsymbol{H_b^H}\boldsymbol{p_{bo}}, & if\ s > M \\ (\boldsymbol{H_b}+\epsilon I)^{-1}\boldsymbol{p_{bo}}, & if\ s = M \\ \boldsymbol{H_b^H}(\boldsymbol{H_b}\boldsymbol{H_b^H}+\epsilon I)^{-1}\boldsymbol{p_{bo}}, & if\ s < M \end{cases} \tag{18}$$

where the Hermitian transpose is labeled H, the identity matrix is labeled I, and the regularization factor is labeled ϵ.

Similar to the square matrix $\boldsymbol{A}$, the terms $\boldsymbol{H_b^H H_b}(if\ s > M)$, $\boldsymbol{H_b}(if\ s = M)$, and $\boldsymbol{H_b}\boldsymbol{H_b^H}(if\ s < M)$ in Equation (18) are all square matrices. Additionally, the solution in Equation (18) becomes the following:

$$\boldsymbol{q} = \begin{cases} (\boldsymbol{B}\boldsymbol{\Lambda'^{-1}}\boldsymbol{B^{-1}})\boldsymbol{H_b^H}\boldsymbol{p_{bo}}, & if\ s > M \\ (\boldsymbol{B}\boldsymbol{\Lambda'^{-1}}\boldsymbol{B^{-1}})\boldsymbol{p_{bo}}, & if\ s = M \\ \boldsymbol{H_b^H}(\boldsymbol{B}\boldsymbol{\Lambda'^{-1}}\boldsymbol{B^{-1}})\boldsymbol{p_{bo}}, & if\ s < M \end{cases} \tag{19}$$

2.4. Comparison of Three Traditional Methods

The ACC method can obtain the maximum acoustic contrast, but it does not take into account the sound field reproduction error in the bright region. The PM method in the bright zone can reduce the sound field reproduction error in the bright zone, but it does not pay attention to the acoustic contrast about the bright zone and the dark zone. In the PM method, increasing array efforts by Tikhonov regularization does not minimize well the reconstruction error at dark points. However, the EDPM method can improve dark points' performance close to that of an elimination method. The EDPM method in the bright zone also does not focus on the acoustic contrast about the bright zone and the dark zone.

3. Proposed Method

This section describes a sound field control optimization model.

3.1. Sound Field Control Method Optimization Model

Reference [30] points out that sound can be described by sound pressure and particle velocity. Reference [31] points out that when loudspeakers are placed unevenly, the loudspeaker signal obtained by controlling the sound pressure of a single region is less stable than the loudspeaker signal obtained by controlling the particle velocity of a single region. Inspired by these conclusions and combined with traditional sound field control methods, this paper proposes a sound field control method that minimizes the error of sound field reproduction in the bright zone, minimizes array effort, and controls the sound pressure and particle velocity in the dark zone at the same time. It is expected to improve the sound field reproduction effect in the bright zone, improve the stability of a loudspeaker array system, and reduce the acoustic energy radiation in the dark zone. The proposed method can be formulated as a constrained optimization problem as follows:

$$\begin{aligned} \min_{\boldsymbol{q}} \quad & \kappa\|\boldsymbol{p_{br}} - \boldsymbol{p_{bo}}\|_2^2 + (1-\kappa)\|\boldsymbol{p_{dr}}\|_2^2 + \sigma\|\boldsymbol{q}\|_2^2 \\ s.t. \quad & \|\boldsymbol{u_{dr}}\|_2 \le \zeta \end{aligned} \tag{20}$$

where κ and σ are weight factors and $0 < \kappa < 1$, $0 < \sigma < 1$, ζ indicates the threshold and is greater than zero, $\|\boldsymbol{p_{br}} - \boldsymbol{p_{bo}}\|_2^2$ stands for the power of sound pressure error between the original sound field pressure and the reproduced sound field pressure in the bright zone, $\|\boldsymbol{p_{dr}}\|_2^2$ stands for dark zone sound energy, $\|\boldsymbol{q}\|_2^2$ stands for sound source power or control effort, and $\boldsymbol{u_{dr}} = \boldsymbol{U_{dr}}\boldsymbol{q}$ stands for the radial particle velocity of the dark zone:

$$\boldsymbol{U_{dr}} = \begin{pmatrix} u_r(\overrightarrow{l_1}, \overrightarrow{d_1}) & u_r(\overrightarrow{l_2}, \overrightarrow{d_1}) & \cdots & u_r(\overrightarrow{l_M}, \overrightarrow{d_1}) \\ u_r(\overrightarrow{l_1}, \overrightarrow{d_2}) & u_r(\overrightarrow{l_2}, \overrightarrow{d_2}) & \cdots & u_r(\overrightarrow{l_M}, \overrightarrow{d_2}) \\ \cdots & \cdots & \cdots & \cdots \\ u_r(\overrightarrow{l_1}, \overrightarrow{d_t}) & u_r(\overrightarrow{l_2}, \overrightarrow{d_t}) & \cdots & u_r(\overrightarrow{l_M}, \overrightarrow{d_t}) \end{pmatrix} \tag{21}$$

$$u_r(\overrightarrow{l_n}, \overrightarrow{d}) = \overrightarrow{u}(\overrightarrow{l_n}, \overrightarrow{d}) \cdot \overrightarrow{v}_r(\overrightarrow{d}) \tag{22}$$

$$\overrightarrow{u}(\overrightarrow{l_n}, \overrightarrow{d}) = \frac{ike^{-ikr}}{4\pi r}\left(1 + \frac{1}{ikr}\right)\frac{(\overrightarrow{d} - \overrightarrow{l_n})}{r} \tag{23}$$

where r is the distance between point $\overrightarrow{d}$ and $\overrightarrow{l_n}$ and $r = |\overrightarrow{d} - \overrightarrow{l_n}|$, k is the wave number, $i = \sqrt{-1}$, $\overrightarrow{u}(\overrightarrow{l_n}, \overrightarrow{d})$ is the particle velocity transmission function between a loudspeaker at $\overrightarrow{l_n}$ and point $\overrightarrow{d}$ [31], $u_r(\overrightarrow{l_n}, \overrightarrow{d})$ stands for the radial particle velocity transmission function, and $\overrightarrow{v}_r(\overrightarrow{d})$ is the unit radial vector perpendicular to the surface of the dark zone and is inward.

The cost function of the proposed method includes the error of reconstructed sound pressure in the bright zone, acoustic energy in the dark zone, and sound source power. The three weight factors κ, σ, and ζ have their own functions in the model. κ is used to modulate the sound pressure error in the bright zone and acoustic energy in the dark zone, σ is used to adjust sound source power, and ζ is used to control the value of particle velocity in the dark zone. The selection strategy of the parameters κ, σ, and ζ is described in the next part. The optimization problem in Equation (20) can be solved using the convex optimization software CVX [33–35].

3.2. The Selection of Parameters κ and ζ

Assume that the initial setup of a loudspeaker system is consistent with the experimental section below. Suppose that the value of κ ranges from 0.1 to 0.9; the change step is 0.1; the value of σ are 0.1, 0.4, 0.6, and 0.9, respectively; and the original sound source signals are 200, 500, 800, and 1000 Hz, respectively. With different values of κ, the relationship between the cost function and the parameter ζ is shown in Figures 2–5, respectively. These graphs show that the value of the relevant cost function increases with the gradual increase in the value of κ on the whole. For all values of κ, the cost function ranges from approximately 0.01 to 0.1. The smaller the value of κ, the stronger the control on the acoustic energy minimization in the dark zone, and the weaker the control on the sound pressure error minimization in the bright zone. The principle for selecting the value of the parameter κ is that the smaller the cost function is, the better the system performs. However, the cost function changes little from 0.01 to 0.1. In the meantime, the sound pressure error in the bright zone has great influence on the auditory sensation. Taking these factors into consideration, the value of κ is chosen to be 0.9. We can see from these figures that, with the constant increase in the parameter ζ, when ζ reaches a certain value, the value of the corresponding cost function basically remains unchanged for a selected value of κ. Therefore, the value of ζ is chosen to be 1.4384. We also performed other simulation experiments: the value of κ ranges from 0.1 to 0.9, and the change step is 0.1; σ equals 0.2, 0.3, 0.5, 0.7, and 0.8, respectively; the original sound source signals are 200, 500, 800, and 1000 Hz, respectively; the value of κ ranges from 0.1 to 0.9, and the change step is 0.1; σ equals 0.9; the original sound source signals are 2000, 4000, 8000, 10,000, and 20,000 Hz, respectively, and can obtain a similar trend.

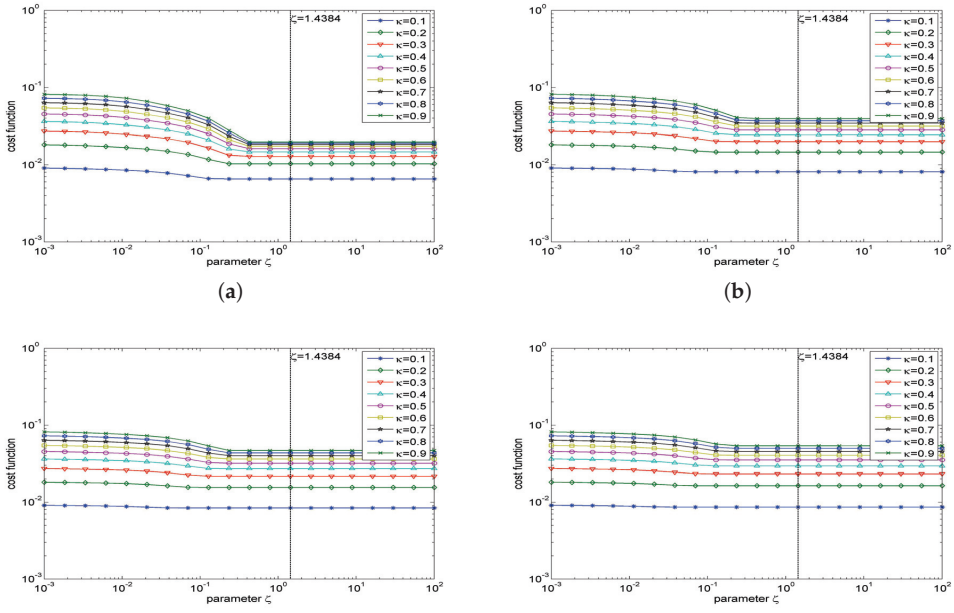


Figure 2. Graph of the relationship between the cost function and the parameter ζ; κ varies from 0.1 to 0.9, and the original sound source signals are 200 Hz. (**a**) $\sigma = 0.1$; (**b**) $\sigma = 0.4$; (**c**) $\sigma = 0.6$; (**d**) $\sigma = 0.9$.

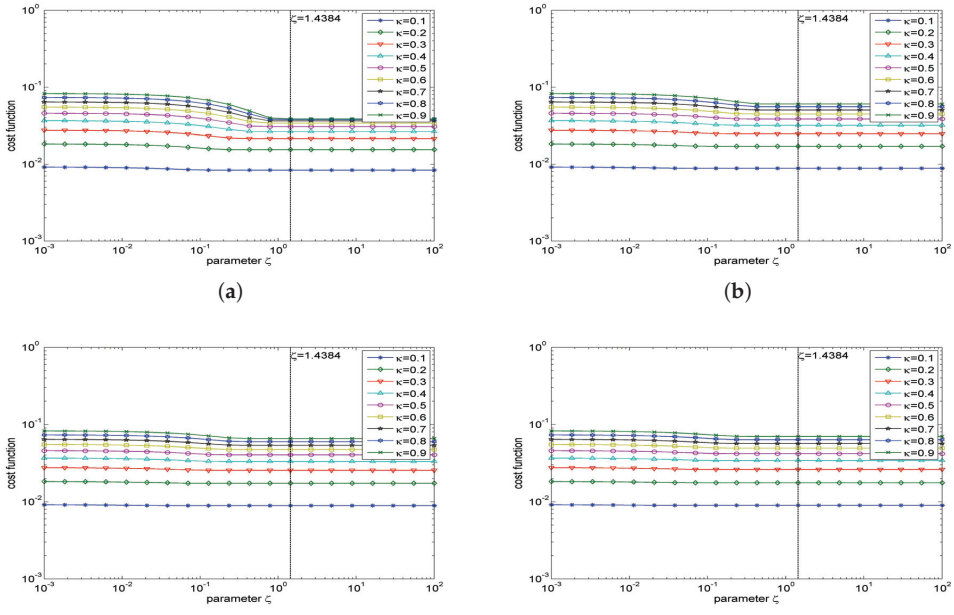


Figure 3. Graph of the relationship between the cost function and the parameter ζ; κ varies from 0.1 to 0.9, and the original sound source signals are 500 Hz. (**a**) $\sigma = 0.1$; (**b**) $\sigma = 0.4$; (**c**) $\sigma = 0.6$; (**d**) $\sigma = 0.9$.

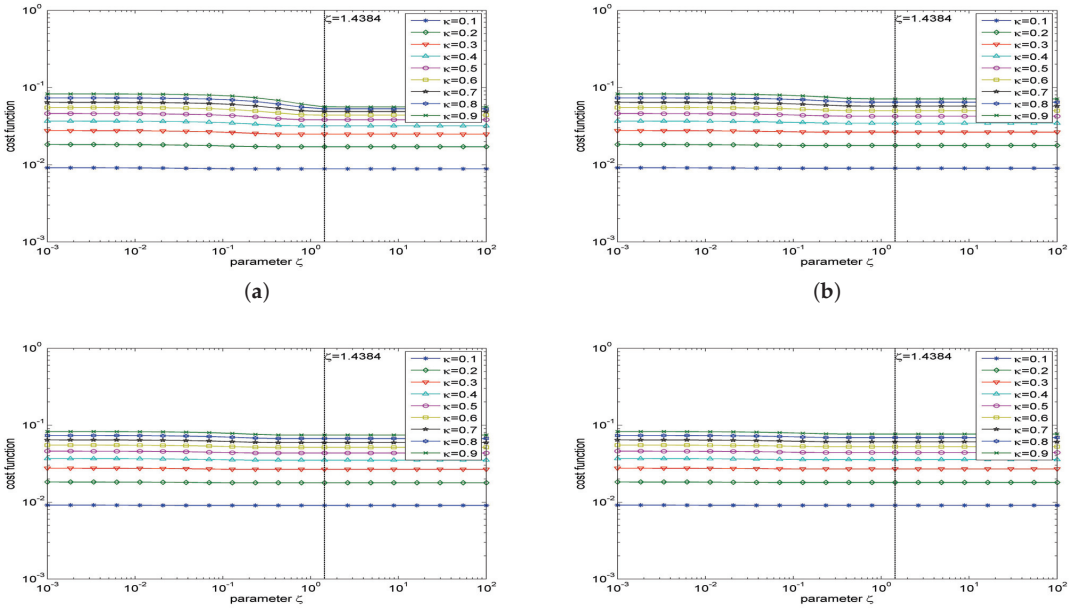


Figure 4. Graph of the relationship between the cost function and the parameter ζ; κ varies from 0.1 to 0.9, and the original sound source signals are 800 Hz. (**a**) $\sigma = 0.1$; (**b**) $\sigma = 0.4$; (**c**) $\sigma = 0.6$; (**d**) $\sigma = 0.9$.

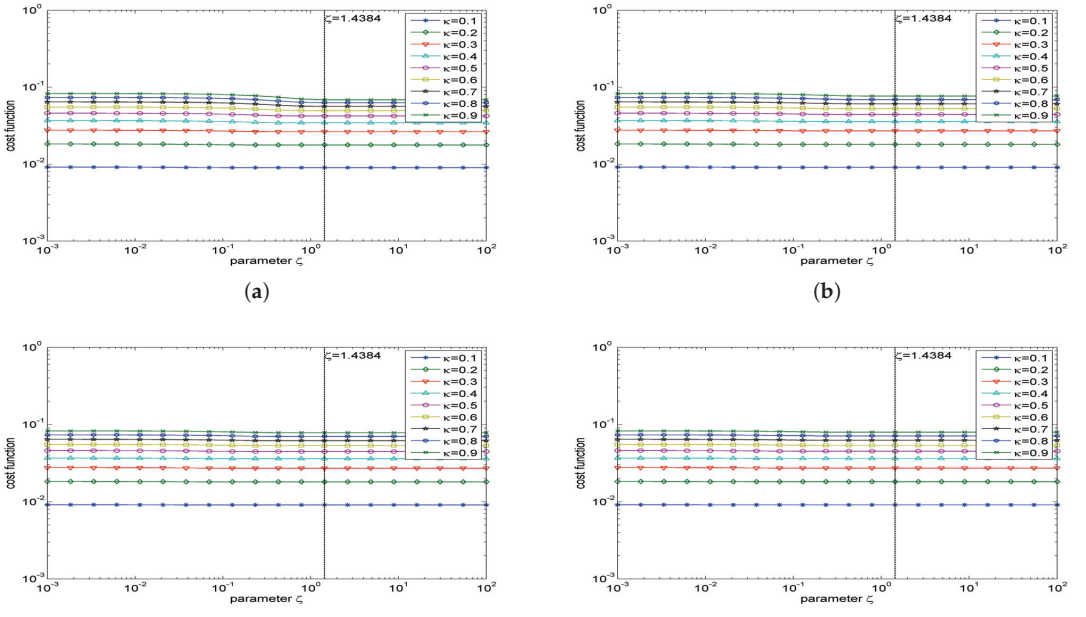


Figure 5. Graph of the relationship between the cost function and the parameter ζ; κ varies from 0.1 to 0.9, and the original sound source signals are 1000 Hz. (**a**) $\sigma = 0.1$; (**b**) $\sigma = 0.4$; (**c**) $\sigma = 0.6$; (**d**) $\sigma = 0.9$.

3.3. Quality of Sound Field Control

Reconstruction error (RE), acoustic contrast (AC), and array effort (AE) are indicators used to indicate the quality of sound field control. RE is the sound pressure error generated by the sound source and the reconstruction system in the bright zone, and its calculation formula is

$$\beta = 10\log_{10}\left(\frac{(\boldsymbol{p_{bo}} - \boldsymbol{p_{br}})^H(\boldsymbol{p_{bo}} - \boldsymbol{p_{br}})}{\boldsymbol{p}_{\boldsymbol{bo}}^H \boldsymbol{p_{bo}}}\right) \tag{24}$$

AC is the ratio of the sound potential energy density of the bright region to the sound potential energy density of the dark region, which is introduced in Section 2.1; here we discretize it and take its logarithm to obtain the calculation formula as

$$\tau = 10\log_{10}\left(\frac{\boldsymbol{p}_{\boldsymbol{br}}^H \boldsymbol{p_{br}}/s}{\boldsymbol{p}_{\boldsymbol{dr}}^H \boldsymbol{p_{dr}}/t}\right) \tag{25}$$

AE is the sum of the square of loudspeakers' distribution coefficient, calculated as follows:

$$\eta = 10\log_{10}\left(\sum_{n=1}^{M} |q_n|^2\right) \tag{26}$$

The measurement criteria of these three indicators are as follows: the smaller the RE, the better the corresponding method; the larger the AC, the better the corresponding method; the smaller the AE, the better the corresponding method, and the more robust and stable the system [16].

3.4. The Selection of Parameter σ

Then let us choose the value of σ. Assuming that the source signal's frequency ranges from 100 to 1000 Hz, the step size is 100 Hz, and σ changes from 0.1 to 0.9, the change step is 0.1; the relationship between the average RE, average AC, average AE, and σ is shown in Figure 6. From Figure 6, we can see that with the increase in σ, the average AC increases gradually, but the increase is relatively small; the average RE increases gradually, from -5.7584 to -1.6931 dB; the average AE decreases gradually, from -9.4790 to -20.4003 dB. Considering the measurement criteria of RE, AC, and AE comprehensively, the value of σ is selected as 0.1 for this example. The is because the average AC at this time is not much different from the average AC when σ takes other values. The average RE at this time is the smallest, and the average RE increases gradually when other σ values are taken, which is not good for listening experience. The average AE at this time is -9.4790 dB, which is small enough, although the average AE corresponding to other σ values is even smaller.

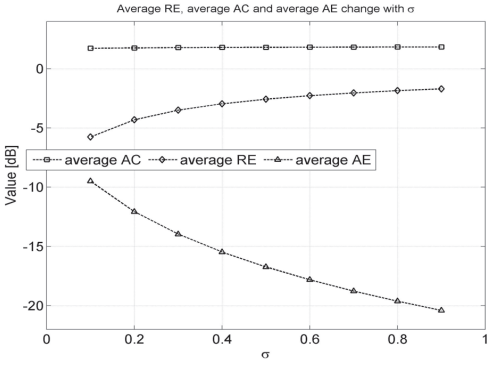


Figure 6. Graph of the relationship between average RE, average AC, average AE, and σ.

4. Simulation Experiment

In this part, the performance of the proposed method is compared with that of the ACC method, the PM method in the bright zone, and the EDPM method in the bright zone in Section 2 through computer simulation experiments. In this paper, an unevenly placed five-channel system is used to verify that the proposed method can ensure the stability of the reconstruction system under the condition of ensuring relatively larger acoustic contrast and smaller reconstruction error. The relationship between the number of loudspeakers and system's performance is not the subject of this paper.

4.1. Experiment Settings

Five loudspeakers, labeled ld 1, ld 2, . . ., ld 5, are placed unevenly on the same ring, and their coordinates are shown in Table 1. The loudspeaker array has a radius of 2 m. The loudspeaker array surrounds the bright zone and dark zone, which are 0.4 m in diameter. The bright zone can accommodate a listener to listen to the sound. The distance from the center point of the bright zone to the center point of the dark zone is 0.6 m. See Table 1 for the coordinates of the center points of the bright and dark zones. The origin is point O. The frequency of the original source signal changes from 100 to 1000 Hz. The original source's location is shown in Table 1. The sound speed expressed in c is 340 m per second, the wavenumber is $k = 2\pi f/c$, and f stands for signal frequency. The sampling interval in both the bright zone and the dark zone is 0.0364 m. The relative position of the unevenly placed five-channel system is shown by Figure 7.

Table 1. Relevant coordinates in the system.

Name	Polar Radius (m)	Azimuthal Angle (Radian)
ld 1	2	0
ld 2	2	$\pi/4$
ld 3	2	$3\pi/4$
ld 4	2	$5\pi/4$
ld 5	2	$7\pi/4$
Original sound source	2.4	$5\pi/9$
Original point O	0	0
Center of the dark zone	0.3	π
Center of the bright zone	0.3	0

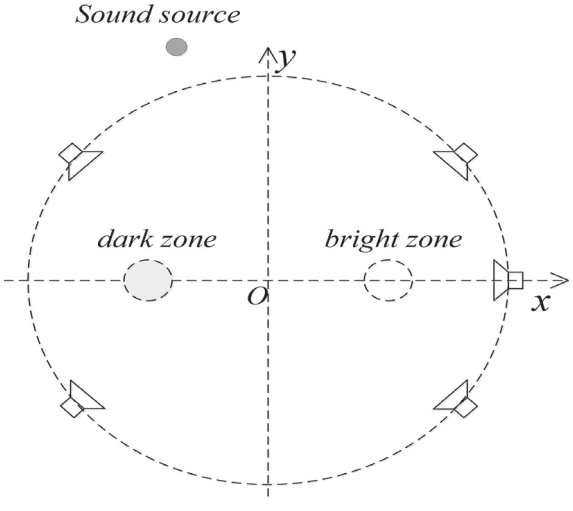


Figure 7. Five-channel sound system structure diagram.

4.2. Experiment Results

Figure 8 shows a comparison of the acoustic contrast about four methods relative to the change with frequency. From Figure 8, we can see that the acoustic contrast of the ACC method ranges from 8.1056 to 31.1125 dB, and on the whole, different methods produce sound contrast in the order of high and low: ACC method, our method, EDPM method in the bright zone, and PM method in the bright zone. Because the goal of the ACC method is to obtain the maximum acoustic contrast, neither the EDPM method in the bright zone nor the PM method in the bright zone focuses on acoustic contrast; the proposed method controls the sound energy in the dark zone, which will improve the acoustic contrast, so the proposed method produces higher acoustic contrast than the EDPM method in the bright zone and the PM method in the bright zone. In addition to improving the acoustic contrast, the proposed method also needs to consider other factors, which affects the effect of the proposed method on acoustic contrast improvement, so the proposed method produces lower acoustic contrast than the ACC method.

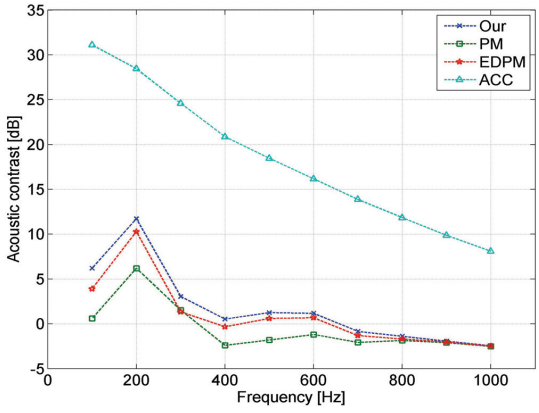


Figure 8. Acoustic contrast comparison diagram of four methods with respect to frequency variation.

The average acoustic contrasts generated by four methods with respect to frequency are shown in Table 2. The average acoustic contrast results generated by four methods are in keeping with Figure 8. The ACC method achieves an average acoustic contrast of 18.3381 dB, the largest of all methods. The average acoustic contrast produced by our method is 2.2898 dB larger than that produced by the PM method in the bright zone and 0.8362 dB larger than that produced by the EDPM method in the bright zone.

Table 2. Average acoustic contrast comparison of four methods with respect to frequency.

Method	Average Acoustic Contrast (dB)
ACC	18.3381
PM method in the bright zone	−0.5581
EDPM method in the bright zone	0.8955
Ours	1.7317

Figure 9 shows a comparison of the reconstruction error of four methods with respect to the change in frequency. The reconstruction errors of the ACC method are greater than 0 dB, which are the largest among all methods. The reconstruction errors of our method are smaller than those produced by the ACC method, and their values range from −12.2945 dB to −1.3299 dB, but are larger than those produced by the EDPM method in the bright zone and the PM method in the bright zone. The PM method in the bright zone has the smallest reconstruction error, and its values vary from −52.5606 dB to −2.0855 dB. The reason is

that the goal of the PM method in the bright zone is to reduce reconstruction error as much as possible and restore the original sound field in the bright zone. The EDPM method in the bright zone is an improvement of the PM method in the bright zone in solving algorithm, and they are similar methods. The ACC method does not pay attention to reconstruction error. Our method controls reconstruction error in the bright zone, so the reconstruction errors of our method are smaller than those of the ACC method. However, our method needs to control other factors in the sound field in addition to controlling the reconstruction error in the bright zone, so the reconstruction errors of our method are larger than those of the PM method in the bright zone and the EDPM method in the bright zone.

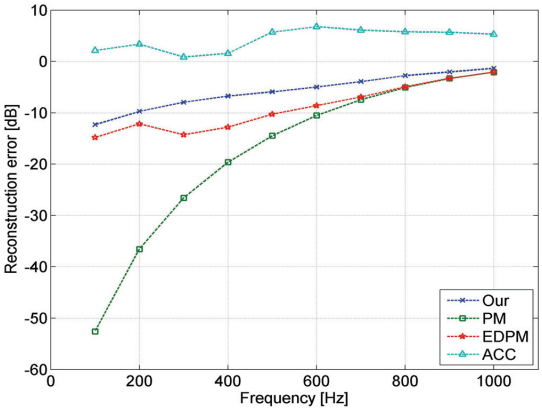


Figure 9. Reconstruction error comparison diagram of four methods with respect to the change in frequency.

The average reconstruction errors generated by four methods with respect to frequency are shown in Table 3. The average reconstruction error results generated by different methods are consistent with Figure 9. The average reconstruction error produced by the PM method in the bright zone is −17.8140 dB, which is the smallest of all methods. The average reconstruction error produced by the ACC method is 4.3408 dB, which is the largest among all methods. The average reconstruction error produced by our method is −5.7584 dB, which is 10.0992 dB smaller than that of the ACC method, but larger than that produced by the PM method in the bright zone and the EDPM method in the bright zone.

Table 3. Average reconstruction error comparison of different methods with respect to frequency.

Method	Average Reconstruction Error (dB)
ACC	4.3408
PM method in the bright zone	−17.8140
EDPM method in the bright zone	−9.0047
Ours	−5.7584

Figure 10 shows a comparison of the array effort of four methods with respect to the change in frequency. The array effort produced by the ACC method is about 0 dB at 10 frequencies, which is the largest of all methods. The array efforts produced by the PM method in the bright zone are smaller than the ACC method at most frequencies, but higher than the ACC method at 100 and 200 Hz. The array efforts at all frequencies produced by the EDPM method in the bright zone are smaller than those produced by the ACC method and the PM method in the bright zone. Our method produces the lowest array efforts at all frequencies of all methods, and its value ranges from −11.8789 dB to −8.0994 dB. Our method controls array effort in the process of sound field reconstruction, while the ACC and PM methods in the bright zone and the EDPM method in the bright zone do not

consider the optimization of array effort in the process of sound field reconstruction, so the array efforts generated by them are larger than those generated by our method.

The average array efforts generated by four methods with respect to frequency are shown in Table 4. The average array effort results generated by different methods are consistent with Figure 10. The average array effort produced by the ACC method is 0 dB, which is the largest of all methods. The average array effort produced by the other three methods is smaller than the average array effort produced by the ACC method. The average array effort produced by our method is −9.4790 dB and the smallest among all methods, which is 9.4790 dB smaller than that of the ACC method, about 8.0712 dB smaller than that of the PM method in the bright zone, and about 4.8176 dB smaller than that of the EDPM method in the bright zone.

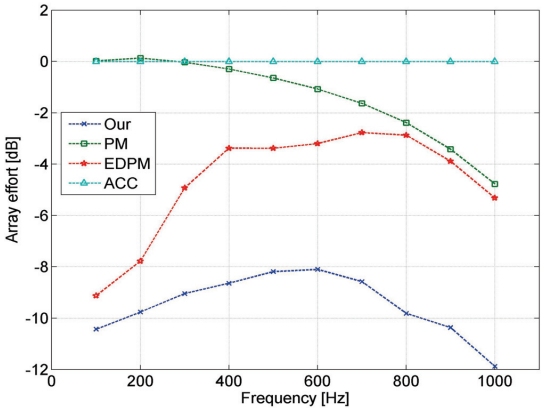


Figure 10. Array effort diagram of four methods with respect to frequency variation.

Table 4. Average array effort comparison of four methods with respect to frequency variation.

Method	Average Array Effort (dB)
ACC	0
PM method in the bright zone	−1.4078
EDPM method in the bright zone	−4.6614
Ours	−9.4790

4.3. Discussion

Acoustic contrast, reconstruction error, and array effort are important indexes to demonstrate the sound field control ability. These three indicators are difficult to achieve the best at the same time, and there is a tradeoff relationship between them. Generally, if one of the three indicators is better for a sound field control method, the other two indicators are worse. Overall, as shown in Section 4.2 of this article, the largest acoustic contrast is obtained by the ACC method due to its focus on maximizing acoustic contrast, but it does not perform well in terms of reconstruction error and array effort; the smallest reconstruction error is achieved by the PM method in the bright zone, as it strives to minimize the reconstruction error in the bright zone, but it presents poor results in acoustic contrast and loudspeaker array effort. The EDPM method in the bright zone is similar to the PM method in the bright zone, but it has some improvements in the solving method, which is superior to the PM method in the bright zone in acoustic contrast and array effort performance, but inferior to the PM method in the bright zone in reconstruction error performance, and fails to perform better than the PM method in the bright zone in all indexes. The ACC method, PM method in the bright zone, and EDPM in the bright zone all control or optimize only one of these indexes: acoustic contrast, reconstruction error, and

loudspeaker array effort, and they do not take into account the other two indexes, so they perform better in one index and poorly in the other two indexes. However, our approach is designed to reduce reconstruction error in the bright zone and loudspeaker array effort as much as possible, and to exert influence on sound pressure and particle velocity in the dark zone range. Our approach attempts to control or optimize acoustic contrast, reconstruction error, and array effort at the same time. Therefore, it is superior to the PM method in the bright zone and the EDPM method in the bright zone in terms of acoustic contrast, better than the ACC method in terms of reconstruction error, and significantly better than the ACC and PM methods in the bright zone and the EDPM method in the bright zone in terms of array effort, which can ensure the most stable reconstruction system.

5. Conclusions

Traditional sound field control technology is mainly based on sound pressure or sound pressure improvement technology, without considering another physical property of sound: particle velocity; if the loudspeaker array is placed unevenly, it will make the reconstruction system unstable. In order to solve this problem, we introduce a new sound field control method based on the traditional sound field control technology. This method pays attention to both sound pressure and particle velocity, exerts influence on the sound pressure and particle velocity in the dark zone, and tries its best to reduce the reconstruction error in the bright zone and reduce the loudspeaker array effort. The model of our method contains three weight factors: κ, σ, and ζ, and their function and selection method are introduced in detail in this paper. Computer simulation experiments are carried out on a system with five unevenly placed loudspeaker, and our method is compared with the ACC method, PM method in the bright zone and the EDPM method in the bright zone, and the comparison indexes are reconstruction error, acoustic contrast, and loudspeaker array effort. Although the ACC method performs best in acoustic contrast and the PM method in the bright zone performs best in reconstruction error, they both focus on only one indicator and perform poorly in the other two. The EDPM method in the bright zone is an improvement of the PM method in the bright zone. Like the PM method in the bright zone, the EDPM method in the bright zone focuses on only one indicator. Although it outperforms the PM method in the bright zone on acoustic contrast and array effort, it is inferior to the PM method in the bright zone on reconstruction error. Compared with the three traditional methods, our method has the best compromise on reconstruction error, acoustic contrast, and array effort. Our method significantly outperforms other comparison methods in array effort performance. The average array effort produced by our method is about 9.4790 dB smaller than that produced by the ACC method, about 8.0712 dB smaller than that produced by the PM method in the bright zone, and about 4.8176 dB smaller than that produced by the EDPM method in the bright zone. Therefore, our method can guarantee optimal system stability in the case of uneven placement of loudspeaker array.

Author Contributions: Conceptualization, S.W.; methodology, S.W.; software, S.W.; validation, S.W.; formal analysis, C.Z.; investigation, S.W.; resources, S.W. and C.Z.; data curation, S.W.; writing—original draft preparation, S.W.; writing—review and editing, S.W. and C.Z.; visualization, S.W.; supervision, C.Z.; project administration, S.W. and C.Z.; funding acquisition, S.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Science and Technology Research Project of the Education Department of Hubei Province (No. B2022245).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Yang, J.; Wu, M.; Lu, H. A review of sound field control. *Appl. Sci.* **2022**, *12*, 7319. [CrossRef]
2. Choi, J.-W.; Kim, Y.-H. Generation of an acoustically bright zone with an illuminated region using multiple sources. *J. Acoust. Soc. Am.* **2002**, *111*, 1695–1700. [CrossRef] [PubMed]
3. Poletti, M. An investigation of 2D multizone surround sound systems. In Proceedings of the 125th AES Convention, San Francisco, CA, USA, 2–5 October 2008.
4. Coleman, P.; Jackson, P.J.B.; Olik, M.; Pedersen, J.A. Personal audio with a planar bright zone. *J. Acoust. Soc. Am.* **2014**, *136*, 1725–1735. [CrossRef] [PubMed]
5. Shin, M.; Lee, S.Q.; Fazi, F.M.; Nelson, P.A.; Kim, D.; Wang, S.; Park, K.H.; Seo, J. Maximization of acoustic energy difference between two spaces. *J. Acoust. Soc. Am.* **2010**, *128*, 121–131. [CrossRef] [PubMed]
6. Chang, J.-H.; Jacobsen, F. Sound field control with a circular double-layer array of loudspeakers. *J. Acoust. Soc. Am.* **2012**, *131*, 4518–4525. [CrossRef]
7. Olivieri, F.; Fazi, F.M.; Shin, M.; Nelson, P. Pressure-matching beamforming method for loudspeaker arrays with frequency dependent selection of control points. In Proceedings of the 138th AES Convention, Warsaw, Poland, 7–10 May 2015.
8. Afghah, T.; Patros, E.; Puckette, M. A pseudoinverse technique for the pressure-matching beamforming method. In Proceedings of the 145th AES Convention, New York, NY, USA, 17–20 October 2018.
9. Elliott, S.J.; Cheer, J.; Choi, J.-W.; Kim, Y. Robustness and regularization of personal audio systems. *IEEE Trans. Audio Speech Lang. Process.* **2012**, *20*, 2123–2133. [CrossRef]
10. Zhu, Q.; Coleman, P.; Wu, M.; Yang, J. Robust acoustic contrast control with reduced in-situ measurement by acoustic modeling. *J. Audio Eng. Soc.* **2017**, *65*, 460–473. [CrossRef]
11. Han, Z.; Wu, M.; Zhu, Q.; Yang, J. Three-dimensional wave-domain acoustic contrast control using a circular loudspeaker array. *J. Acoust. Soc. Am.* **2019**, *145*, EL488–EL493. [CrossRef]
12. Hu, M.; Lu, J. Theoretical explanation of uneven frequency response of time-domain acoustic contrast control method. *J. Acoust. Soc. Am.* **2021**, *149*, 4292–4297. [CrossRef]
13. Lee, T.; Nielsen, J.K.; Christensen, M.G. Towards perceptually optimized sound zones: A proof-of-concept study. In Proceedings of the 2019 ICASSP, Brighton, UK, 12–17 May 2019; pp. 136–140.
14. Lee, T.; Nielsen, J.K.; Christensen, M.G. Signal-adaptive and perceptually optimized sound zones with variable span trade-off filters. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2020**, *28*, 2412–2426. [CrossRef]
15. Lee, T.; Shi, L.; Nielsen, J.K.; Christensen, M.G. Fast generation of sound zones using variable span trade-off filters in the dft-domain. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2021**, *29*, 363–378. [CrossRef]
16. Ryu, H.; Wang, S.; Kim, S.M. Development of a personal audio performance controller with efficient, fine, and linear tunable functions. *IEEE Access* **2020**, *8*, 123916–123928. [CrossRef]
17. Hu, X.; Wang, J.; Zhang, W.; Zhang, L. Time-domain sound field reproduction with pressure and particle velocity jointly controlled. *Appl. Sci.* **2021**, *11*, 10880. [CrossRef]
18. Du, B.; Zeng, X.; Wang, H. A two-zone sound field reproduction based on the region energy control. In Proceedings of the Inter Noise 2021, Washington, DC, USA, 1–5 August 2021; pp. 348–354.
19. Du, B.; Zeng, X.; Vorländer, M. Multizone sound field reproduction based on equivalent source method. *Acoust. Aust.* **2021**, *49*, 317–329. [CrossRef]
20. Zhu, M.; Zhao, S. An iterative approach to optimize loudspeaker placement for multi-zone sound field reproduction. *J. Acoust. Soc. Am.* **2021**, *149*, 3462–3468. [CrossRef] [PubMed]
21. Xie, Y.M.; Steven, G.P. A simple evolutionary procedure for structural optimization. *Comput. Struct.* **1993**, *49*, 885–896. [CrossRef]
22. Zhao, S.; Burnett, I.S. Evolutionary array optimization for multizone sound field reproduction. *J. Acoust. Soc. Am.* **2022**, *151*, 2791–2801. [CrossRef]
23. Zhong, J.; Zhuang, T.; Kirby, R.; Karimi, M.; Zou, H.; Qiu, X. Quiet zone generation in an acoustic free field using multiple parametric array loudspeakers. *J. Acoust. Soc. Am.* **2022**, *151*, 1235–1245. [CrossRef]
24. Abe, T.; Koyama, S.; Ueno, N.; Saruwatari, H. Amplitude matching for multizone sound field control. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2023**, *31*, 656–669. [CrossRef]
25. Elliott, S.J.; Jones, M. An active headrest for personal audio. *J. Acoust. Soc. Am.* **2006**, *119*, 2702–2709. [CrossRef]
26. Cheer, J.; Elliott, S.J.; Kim, Y. Practical implementation of personal audio in a mobile device. *J. Audio Eng. Soc.* **2013**, *61*, 290–300.
27. Cheer, J.; Elliott, S.J.; Gálvez, M.F.S. Design and implementation of a car cabin personal audio system. *J. Audio Eng. Soc.* **2013**, *61*, 412–424.
28. Liao, X.; Cheer, J.; Elliott, S.J.; Zheng, S. Design of a loudspeaker array for personal audio in a car cabin. *J. Audio Eng. Soc.* **2017**, *65*, 226–238. [CrossRef]
29. Choi, J.W. Real-Time demonstration of personal audio and 3D audio rendering using line array systems. In Proceedings of the MMM 2020, Daejeon, Republic of Korea, 5–8 January 2020.
30. Pierce, A.D. *Acoustics, an Introduction to Its Physical Principles and Applications*; Acoustical Society of America: New York, NY, USA, 1989.
31. Shin, M.; Nelson, P.A.; Fazi, F.M.; Seo, J. Velocity controlled sound field reproduction by non-uniformly spaced loudspeakers. *J. Sound Vib.* **2016**, *370*, 444–464. [CrossRef]
32. Olivieri, F.; Fazi, F.M.; Nelson, P.A.; Fontana, S. Comparison of strategies for accurate reproduction of a target signal with compact arrays of loudspeakers for the generation of zones of private sound and silence. *J. Audio Eng. Soc.* **2016**, *64*, 905–917. [CrossRef]

33. Grant, M.; Boyd, S. CVX, Version 1.21 MATLAB Toolbox for Disciplined Convex Programming. Available online: http://cvxr.com/cvx (accessed on 5 November 2023).
34. Bai, M.R.; Chen, C.C. Application of convex optimization to acoustical array signal processing. *J. Sound Vib* **2013**, *332*, 6596–6616. [CrossRef]
35. Grant, M.; Boyd, S.; Ye, Y. Disciplined convex programming. In *Global Optimization: From Theory to Implementation, Nonconvex Optimization and Applications*; Liberti, L., Maculan, N., Eds.; Springer: New York, NY, USA, 2006; pp. 155–210.

applied sciences

MDPI

Article

A Scalogram-Based CNN Approach for Audio Classification in Construction Sites

Michele Scarpiniti [1,*], Raffaele Parisi [1] and Yong-Cheol Lee [2]

[1] Department of Information Engineering, Electronics and Telecommunications (DIET), Sapienza University of Rome, via Eudossiana 18, 00184 Rome, Italy; raffaele.parisi@uniroma1.it
[2] Department of Construction Management, Louisiana State University, Baton Rouge, LA 70803, USA; yclee@lsu.edu
* Correspondence: michele.scarpiniti@uniroma1.it; Tel.: +39-06-44585869

Abstract: The automatic monitoring of activities in construction sites through the proper use of acoustic signals is a recent field of research that is currently in continuous evolution. In particular, the use of techniques based on Convolutional Neural Networks (CNNs) working on the spectrogram of the signal or its mel-scale variants was demonstrated to be quite successful. Nevertheless, the spectrogram has some limitations, which are due to the intrinsic trade-off between temporal and spectral resolutions. In order to overcome these limitations, in this paper, we propose employing the scalogramas a proper time–frequency representation of the audio signal. The scalogram is defined as the square modulus of the Continuous Wavelet Transform (CWT) and is known as a powerful tool for analyzing real-world signals. Experimental results, obtained on real-world sounds recorded in construction sites, have demonstrated the effectiveness of the proposed approach, which is able to clearly outperform most state-of-the-art solutions.

Keywords: automatic construction site monitoring (ACSM); environmental sound classification (ESC); deep learning; convolutional neural network (CNN); continuous wavelet transform (CWT); scalogram; audio processing

Citation: Scarpiniti, M.; Parisi, R.; Lee, Y.-C. A Scalogram-Based CNN Approach for Audio Classification in Construction Sites. *Appl. Sci.* **2024**, *14*, 90. https://doi.org/10.3390/app14010090

Academic Editors: Giovanni Costantini and Daniele Casali

Received: 20 November 2023
Revised: 15 December 2023
Accepted: 20 December 2023
Published: 21 December 2023

1. Introduction

In recent years significant research efforts have been made in the field of Environmental Sound Classification (ESC) [1], allowing significant results to be obtained in practical sound classification applications. This initiative has been enabled by the use of Convolutional Neural Networks (CNNs), which allowed a superior performance in image processing problems [2] to be obtained. In order to extend the use of CNNs to the field of audio processing, the audio input signal is usually transformed into suitable bi-dimensional image-like representations, such as spectrograms, mel-scale spectrograms, and other similar methods [3,4].

Recently, the approaches employed in ESC have been transferred to advancing the construction domain by converting vision-based work monitoring and management systems into audio-based ones [5–7]. In fact, audio-based systems not only are more cost-effective than video-based ones, but they also work more effectively in a construction field when sources are far from the light of sight of sensors, making these systems very flexible and appropriate for combining other sensor-based applications or Artificial Intelligence (AI)-based technologies [7]. Furthermore, the amount of memory and data flow needed to handle audio data is much smaller than the one needed for video data. In addition, audio-based systems outperform accelerometer-based ones since there is no need to place sensors onboard, thus promoting 360-degree-based activity detection and surveillance without having an illumination issue [8].

Such audio-based systems can be successfully used as Automatic Construction Site Monitoring (ACSM) tools [7,9–11], which can represent an invaluable instrument for project

managers to promptly identify severe and urgent problems in fieldwork and quickly react to unexpected safety and hazard issues [12–16].

ACSM systems are usually implemented by exploiting both machine learning (ML) and deep learning (DL) techniques [17]. Specifically, several ML approaches, including Support Vector Machines (SVMs), the k-Nearest Neighbors (k-NN) algorithm, the Multilayer Perceptron (MLP), random forests, Echo State Networks (ESN), and others, have already demonstrated their effectiveness in properly performing activity identification and detection in a construction site [5,16]. However, DL approaches generally outperform ML-based solutions providing much improved results [6]. We expect that DL techniques including CNNs, Deep Recurrent Neural Networks (DRNNs) implemented with the Long Short-Term Memory (LSTM) cell, Deep Belief Networks (DBNs), Deep ESNs, and others can produce more suitable and qualified performances than ML ones for robustly managing construction work and safety issues.

Approaches based on CNNs have demonstrated good flexibility and considerably convincing performance in these applications. In fact, CNNs exhibit advanced accuracy in image classification [18]. In order to meet the bi-dimensional format of images, the audio waveform can be transformed into a bi-dimensional representation by a proper time–frequency transformation. The main time–frequency representation used in audio applications is the spectrogram, i.e., the squared magnitude of the Short Time Fourier Transform (STFT) [19,20]. The spectrogram is very rich in peculiar information that can be successfully exploited by CNNs. Instead of using the STFT spectrogram, in audio processing, it is very common to use some well-known variants, such as the constant-Q spectrogram, which uses a log-frequency mapping, and the mel-scale spectrogram, which uses the mel-scale of frequency to better capture the intrinsic characteristic of the human ear. Similarly, the Bark and/or ERB scales can be used, producing other variants of the spectrogram [21].

Although the spectrogram representation and its variants provide an effective way to extract features from audio signals, they entail some limitations due to the unavoidable trade-off between the time and frequency resolutions. Unfortunately, it is hard to provide an adequate resolution in both domains: a shorter time window provides a better time resolution, but it reduces the frequency resolution, while using longer time windows improves the frequency resolution but obtains a worse time resolution. Even if some solutions have been proposed to mitigate such an unwanted effect (such as the time–frequency reassignment and synchrosqueezing approach [22]), the problem can still affect the performance of deep learning methods. Moreover, the issue is also complicated by the fact that sound information is usually available at different time scales that cannot be captured by the STFT.

Motivated by these considerations, in this paper, we propose a new approach for the automatic monitoring of construction sites based on CNNs and scalograms. The scalogram was defined as the squared magnitude of the Continuous Wavelet Transform (CWT) [23]. By overcoming the intrinsic time–frequency trade-off, the scalogram is expected to offer an advanced and robust tool to improve the overall accuracy and performance of ACSM systems. In addition, the wavelet transform allows to it work at different time scales, which is a useful characteristic for the processing of audio data. Hence, the main idea of the paper is to use the scalogram instead of the spectrogram as the input to a CNN-based deep learning model. Although the methodology is not new, the proposed idea has been extensively tested on real data acquired in construction sites and, compared to most popular state-of-the-art methodologies, shows clear and significant improvements.

The rest of this paper is organized as follows. Section 2 shows the related work. Section 3 introduces the CWT, while Section 4 describes the proposed approach. Then, Section 5 explains the adopted experimental setup. Section 6 describes some implementation aspects, while Section 7 shows the obtained numerical results and confirms the effectiveness of the proposed idea. Finally, Section 8 concludes the work and outlines some hints for future research.

2. Related Work

In the digital era, great and increasing attention has been devoted to research on automated methods for real-time monitoring of activities in construction sites [15,24,25]. These modern approaches are able to offer better performance with respect to the most traditional techniques, which are typically based on manual collection of on-site work data and human-based construction project monitoring. In fact, these activities are typically time-consuming, inaccurate, costly, and labor-intensive [13]. In the last years, the literature related to applications of deep learning techniques to the construction industry has been continuously increasing [26,27]. In particular, many works have been published describing proper exploitation of audio data [5,16].

The work of Cao et al. in [28] was one of the first attempts in this direction. They introduced an algorithm based on the processing of acoustic data for the classification of four representative excavators. This approach is based on some acoustic statistical features. Namely, for the first time the short frame energy ratio, concentration of spectrum amplitude ratio, truncated energy range, and interval of pulse (i.e., the time interval between two consecutive peaks) were developed in order to characterize acoustic signals. The obtained results were quite effective for this kind of source; however, no other types of equipment were considered.

Paper [29] proposed the construction of a dataset of four classes of equipment and tested several ML classifiers. The results obtained in this work were aligned to those shown in [5], which compared and assessed the accuracy of 17 classifiers on nine classes of equipment. These two papers work on both temporal and spectral features extracted from audio signals. Similarly, Ref. [30] compared some ML approaches on five input classes by using a single in-pocket smartphone, obtaining similar numerical results.

Akbal et al. [14] proposed an SVM classifier. After an iterative neighborhood component analysis selector chooses the most significant features extracted from audio signals, this classifier produces an effective accuracy on two experimental scenarios. Moreover, Kim et al. [7] proposed a sound localization framework for construction site monitoring able to work in both indoor and outdoor scenarios.

Maccagno et al. [31] proposed a deep CNN-based approach for the classification of five pieces of construction site machinery and equipment. This customized CNN is fed by the STFT spectrograms extracted from different-sized audio chunks. Similarly, Sherafat et al. [32] proposed an approach for multiple-equipment activity recognition using CNNs, tested on both synthetic and real-world equipment sound mixtures. Different from [31], this work implements a data augmentation method to enlarge the used dataset. Moreover, this model uses a moving mode function to find the most frequent labels in a period ranging from 0.5 to 2 s, which generates an acceptable output accuracy. The idea to join different output labels inside a short time period was also exploited in [33,34], which implement a Deep Belief Network (DBN) classifier and an Echo State Network (ESN), respectively.

Kim et al. in [35] applied CNNs and RNNs to spectrograms for monitoring concrete pouring work in construction sites, while Xiong et al. in [6] used a convolutional RNN (CRNN) for activity monitoring. Moreover, Peng et al. in [36] used a similar DL approach for a denoising application in construction sites. On the other hand, Akbal et al. [37] proposed an approach, called DesPatNet25, which extracts 25 feature vectors from audio signals by using the data encryption standard cipher and adopts a k-NN and an SVM classifier to identify seven classes.

Additionally, some other approaches also fused information from two different modalities. For example, the work in [38] used an SVM classifier by combining both auditory and kinematics features, showing an improvement of about 5% when compared to the use of only individual sources of data. Similarly, Ref. [39] exploited visual and kinematic features, while [40] utilized location data from a GPS and a vision-based model to detect construction equipment. Finally, a multimodal audio–video approach

was presented in [41], based on the use of different correlations of visual and auditory features, which has shown an overall improvement in detection performance.

In addition, Elelu et al. in [42] exploited CNN architectures to automatically detect collision hazards between construction equipment. Similarly, the work in [43] presented a critical review of recent DL approaches for fully embracing construction workers' awareness of hazardous situations in construction sites by the employment of auditory systems.

Most of the DL approaches described in this section work on the spectrogram extracted from audio signals or some variants, such as the mel-scaled spectrogram. However, the idea of exploiting different time scales (which is an intrinsic property of audio signals) can be used to improve the overall accuracy of such methodologies. For this purpose, the use of scalograms can be recommended. In fact, while spectrograms are suitable for the analysis of stationary signals providing a uniform resolution, the scalogram is able to localize transients in non-stationary signals. Recently, in fact, Ref. [44] introduced a wavelet filter bank for the audio scene modeling task. A deep CNN fed by the scalogram of data outperformed the results provided by the mel spectrogram. However, differently from our approach, the work in [44] considers a scalogram of smaller size and a simpler CNN architecture. The work in [45] adopted scalograms for removing background noise in the fault diagnosis of rotating machinery, obtaining excellent experimental results. However, differently from our approach, given the specific nature of the considered sounds, the authors used a low sampling frequency and frame size, resulting in a very small scalogram size (64×64 pixels). Interestingly enough, [45] considers three different methods to obtain the scalograms, including the CWT. No significant statistical differences have been observed between such methods. In addition, a couple of papers used scalograms also for audio scene classification purposes [46,47]. Both of these works showed very good results when compared to previous solutions. As a matter of fact, the use of the scalogram results in a general improvement in performance as highlighted in all these works. Specifically, the work in [46] exploits a pre-trained CNN to extract, at a specific architecture-dependent layer, useful features to be used by a subsequent linear SVM classifier for the identification of ten environmental categories. This work also uses AlexNet but, differently from our approach, it does not train the CNN layers and does not adopt fully connected layers as a classifier. The work in [47] again uses a pre-trained AlexNet or VGG16/19 nets to extract meaningful features, but, differently, it exploits a Bidirectional Gated Recurrent Neural Network followed by a highway layer to classify fifteen classes. Differently from our approach, the authors of [47] adopt an early data fusion technique by feeding the proposed model with a three-channel image composed of a spectrogram, a scalogram extracted with the Bump wavelet, and a scalogram obtained with the Morse wavelet. However, the high computational cost of this approach, compared with the proposed one, makes it not very suitable for working with the construction site sounds, where only a small number of classes are present.

3. The Continuous Wavelet Transform (CWT) and the Scalogram

In order to overcome the trade-off between the time and frequency resolution in STFT, the Continuous Wavelet Transform (CWT) was introduced [23]. The CWT acts as a "mathematical" microscope in the sense that different parts of the signal may be examined by adjusting the focus.

Given a stationary signal $x(t)$, the CWT is defined as the product of $x(t)$ with the following basis function family:

$$\Psi_{\tau,a}(t) = |a|^{-1/2}\,\Psi\left(\frac{t-\tau}{a}\right), \tag{1}$$

where $a \neq 0$ is a scaling factor (also known as dilation parameter) and τ is the time delay, i.e., $\Psi_{\tau,a}(t)$ is a scaled and translated version of the mother wavelet function $\Psi(t)$. Hence, the CWT of signal $x(t)$ is formulated as:

$$W_x(\tau, a) = |a|^{-1/2} \int_{-\infty}^{\infty} x(t) \Psi^* \left(\frac{t-\tau}{a} \right) dt, \tag{2}$$

where * represents the complex conjugation operator. The delay parameter τ provides the time position of the wavelet $\Psi_{\tau,a}(t)$, while the scaling factor a rules its frequency content. For $|a| << 1$, the wavelet $\Psi_{\tau,a}(t)$ is a very concentrated and narrow version of the mother wavelet $\Psi(t)$, with a frequency content mainly condensed at high frequencies. On the other hand, for $|a| >> 1$, the wavelet $\Psi_{\tau,a}(t)$ is much more broadened and concentrated towards low frequencies.

In the wavelet analysis, the similarity between the signal $x(t)$ and the wavelet $\Psi_{\tau,a}(t)$ is measured as τ and a vary. Dilation by a factor $1/a$ results in different enlargements of the signal with distinct resolutions. Specifically, the properties of the time–frequency resolution of the CWT are summarized as follows:

1. The temporal resolution $\Delta\tau$ varies inversely to the carrier frequency ω_0 of the wavelet $\Psi_{\tau,a}(t)$; therefore, it can be made arbitrarily small at high frequencies.
2. The frequency resolution $\Delta\omega$ varies linearly with the carrier frequency ω_0 of the wavelet $\Psi_{\tau,a}(t)$; therefore, it can be made arbitrarily small at low frequencies.

Hence, the CWT is well suited for the analysis of non-stationary signals containing high-frequency transients superimposed on long-lasting low-frequency components [23].

The CWT implements the signal analysis at various time scales. For this reason, the squared absolute value of the CWT is called a scalogram, and it is defined as:

$$\mathcal{S}(\tau, a) \triangleq |W_x(\tau, a)|^2 = \frac{1}{a} \left| \int_{-\infty}^{\infty} x(t) \Psi^* \left(\frac{t-\tau}{a} \right) dt \right|^2. \tag{3}$$

The scalogram $\mathcal{S}(\tau, a)$ provides a bi-dimensional graphical representation of the signal energy at the specific scale parameter a and time location τ.

In general, the mother wavelet $\Psi(t)$ can be any band-pass function [23]. The Haar wavelet is the simplest example of a wavelet, while the Daubechies one is a more sophisticated example. Both of these wavelets have a finite (and compact) support in time. The Daubechies wavelet has a longer length than the Haar wavelet and is therefore less localized than the latter. However, the Daubechies wavelet is continuous and has a better frequency resolution than the Haar one [23]. Other famous wavelet families are the Mexican Hat wavelet (which is proportional to the second derivative function of the Gaussian probability density function), the Bump wavelet, the generalized Morse wavelet, and the Morlet one, also known as the Gabor wavelet. This last wavelet is composed of a complex exponential multiplied by a Gaussian window, and it is very suitable for audio and vision applications since it is closely related to human perception. For this purpose, we remark that it is strongly related to the short-time analysis performed by the peripheral auditory system and to the mechanical spectral analysis performed by the basilar membrane in the human ear [48]. As a matter of fact, the Morlet wavelet is the most widely used wavelet for audio applications [49], and its effectiveness has been shown in analyzing machine sounds [50]. Motivated by these considerations, in the rest of the paper, we use the Morlet wavelet, which is defined as:

$$\Psi(t) = C_\psi \, e^{-\frac{t^2}{2\sigma^2}} e^{j\omega_0 t}, \tag{4}$$

where C_ψ is a normalization factor used to meet the admissibility condition, ω_0 is the central frequency of the mother wavelet (the carrier), and σ^2 is the variance of the Gaussian window equal to: $\sigma = n/\omega_0$. The parameter n, called the number of wavelet cycles and set in this paper to $n = 6$, defines the time–frequency precision trade-off.

4. Proposed Approach

Scalograms obtained from CWT are very rich in information and can improve the results obtained by other approaches, such as the spectrogram or its mel-scale version. The proposed idea consists in extracting the scalograms from the recorded signals after splitting

them into chunks of a suitable length (usually 30–50 ms). The extracted scalograms, saved as image files, are fed as input to a CNN architecture. In fact, it is well known that CNNs are very effective for image classification. The literature is rich in state-of-the-art CNNs that perform very well in image classification. Since the number of classes considered in a construction site is limited, and given the richness of the input representation (the scalogram), in this work, we propose the use of a simple CNN, i.e., the AlexNet one (see Section 5.3). A picture of the proposed idea is shown in Figure 1. A step-by-step flowchart of the proposed methodology is shown in Figure 2.

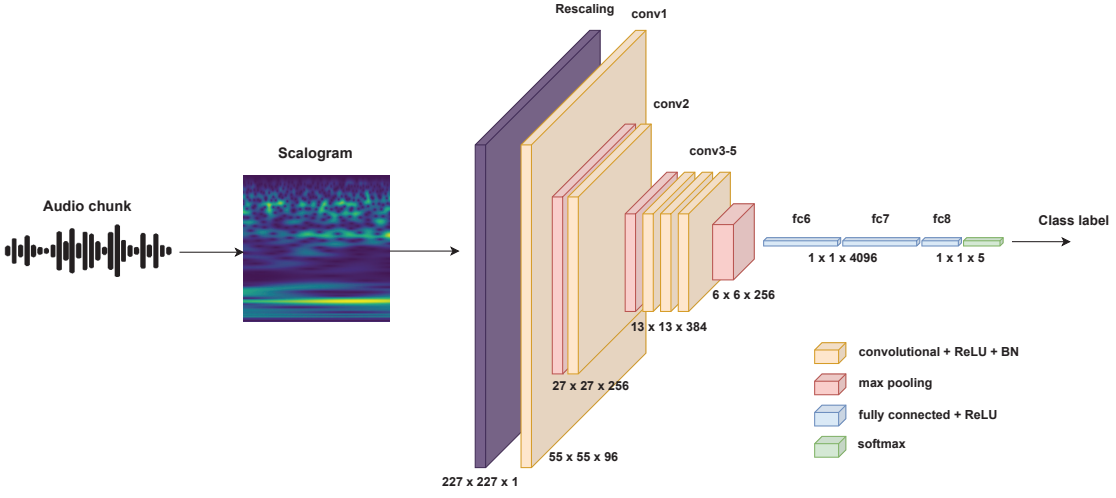


Figure 1. A picture of the proposed idea.

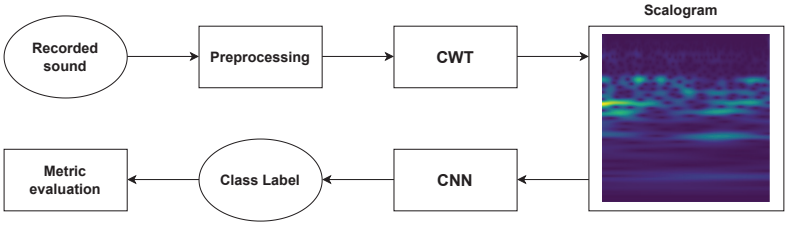


Figure 2. A step-by-step flowchart of the proposed methodology.

5. Experimental Setup

5.1. Dataset

The used dataset consists of a set of recordings related to five machines working in a real-world construction site. Sounds have been recorded with a Zoom H1 digital recorder with a sampling frequency of 44,100 Hz and saved as wave files. The five classes considered in this work are related to three excavators (two compact excavators and a large one), a compactor, and a concrete mixer. For each class, 15 min of recordings are available. The recording of each piece of machinery has been made by placing the recorder about 5–6 m in front of the activities of interest, without any obstacle in the middle. The recorded sounds are related to normal construction site activities (i.e., excavation and concrete mixing work) performed in outdoor scenarios. The subset of sounds considered in this work is related to a single source at a time; segments where more than one piece of equipment is active at the same time have been preventively removed from the dataset. Additional details on the operating scenario can be found in [5].

Each file has been split into chunks of 30 ms each. The entire dataset has been split into a training and a test set, with proportions of 75% and 25%, respectively. In addition, 10% of the training set has been devoted as the validation set to check the convergence performance during the training phase. Details of the used dataset, along with the number of chunks and related training/test splits, are reported in Table 1.

Table 1. Details of the used dataset.

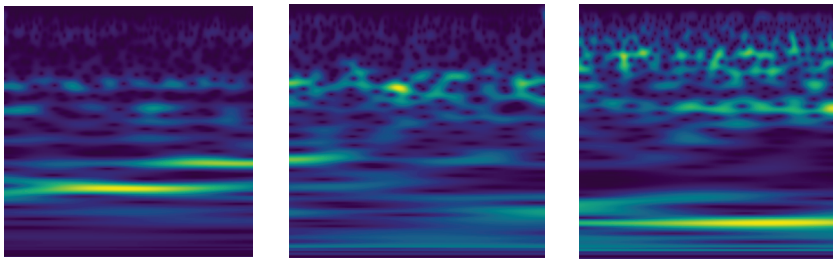
N.	Class	Equipment	Data	Chunks	Split
1	JD50D	Compact excavator John Deere 50D	15:00	30,000	22,500/7500
2	IRCOM	Ingersoll Rand Compactor	15:00	30,003	22,502/7501
3	Mixer	Concrete mixer Mercedes-Benz Actros	14:59	29,999	22,499/7500
4	CAT320E	Hydraulic excavator Caterpillar 320E	15:00	30,001	22,500/7501
5	Hitachi50U	Compact excavator Hitachi ZX50U	14:59	29,999	22,499/7500
		Total	01:14:58	150,002	112,500/37,502

5.2. Preprocessing

After the audio signals have been split into chunks of 30 ms, they have been resampled to 22,050 Hz for memory-saving purposes. This resampling procedure does not affect the quality of the classification, since the energy of audio signals related to construction sites is vanishing at frequencies higher than 10 kHz.

For each resampled chunk, the CWT has been extracted (we used the Python ssqueezepy package, available at: https://github.com/OverLordGoldDragon/ssqueezepy, accessed on 10 November 2023). The Morlet wavelet [23] has been used in this work. The obtained matrix has been then resized to 227×227 in order to be compliant with the input layer of the used AlexNet (see Section 5.3). For simplicity and memory-saving purposes, the obtained resized matrix has been rescaled to the interval $[0, 255]$, converted to integer numbers, and saved as images.

Some random images related to the extracted CWT from Classes 1, 3, and 4, respectively, are shown in Figure 3. These scalograms clearly capture salient localized events in sound frames, as shown by the horizontal lines or cloud-like points in Figure 3. Spectrograms of the same signals are generally unable to capture salient time/scale characteristics.



(**a**) (**b**) (**c**)

Figure 3. Examples of some scalogram images: (**a**) Class 1, (**b**) Class 3, and (**c**) Class 4.

5.3. Model

The literature is rich in well-performing and famous CNN architectures, as well as customized models for specific applications. Since the problem has been converted into a standard image classification task and the number of classes is limited, in this paper, we consider the well-known AlexNet [51] architecture. Specifically, AlexNet is composed of a cascade of five convolutional layers and three (dense) fully connected ones.

With respect to the original version, we introduce three modifications:

1. The number of channels of the input layer is reduced to only one since the network is fed by the scalogram, which is a single-channel image;

2. We add, after the input layer, a Rescaling layer in order to transform the integer input into floating-point numbers inside the interval $[0,\ 1]$;
3. The number of output classes has been reduced to five (the original AlexNet works with 1000 classes).

The details of the network organization, layers' shape, and number of parameters of the customized version of AlexNet are summarized in Table 2. Refer also to Figure 1 for a graphical representation.

AlexNet has been trained by minimizing the categorical cross-entropy defined as:

$$\mathcal{L}(y, \widehat{y}, \boldsymbol{\theta}) \triangleq -\frac{1}{B} \sum_{n=1}^{B} \sum_{i=1}^{N_C} y_n^{(i)} \log \widehat{y}_n^{(i)}, \tag{5}$$

where $\boldsymbol{\theta}$ is the vector collecting all of the network parameters, $N_C = 5$ is the number of classes, B is the mini-batch size, $y_n^{(i)}$ is the actual label of the n-th sample and i-th class, and $\widehat{y}_n^{(i)}$ is the corresponding predicted label. The minimization is performed by the gradient descent algorithm:

$$\boldsymbol{\theta}_k = \boldsymbol{\theta}_{k-1} - \eta \nabla_{\boldsymbol{\theta}_k} \mathcal{L}(y, \widehat{y}, \boldsymbol{\theta}_{k-1}), \tag{6}$$

where η is the learning rate and k is the iteration index; the gradient $\nabla_{\boldsymbol{\theta}} \mathcal{L}(\cdot)$ is computed over a mini-batch. In this work, the Adam optimizer, a variant of the gradient descent, has been used [52]. Specifically, the Adam algorithm incorporates an estimate of the first- and second-order moments of the gradient with a bias correction to speed up the convergence process. Details of the Adam algorithm can be found in [52]. The learning rate is set to $\eta = 10^{-4}$ (parameters β_1, β_2, and ε are left at their default values), and a batch size of $B = 32$ is used. The training is run for 10 epochs.

Table 2. Layers and number of parameters of the customized AlexNet.

Layer (Type)	Output Shape	Number of Parameters
Rescaling	(None, 227, 227, 1)	0
Conv2D	(None, 55, 55, 96)	11,712
BatchNormalization	(None, 55, 55, 96)	384
MaxPooling2D	(None, 27, 27, 96)	0
Conv2D	(None, 27, 27, 256)	614,656
BatchNormalization	(None, 27, 27, 256)	1024
MaxPooling2D	(None, 13, 13, 256)	0
Conv2D	(None, 13, 13, 384)	885,120
BatchNormalization	(None, 13, 13, 384)	1536
Conv2D	(None, 13, 13, 384)	1,327,488
BatchNormalization	(None, 13, 13, 384)	1536
Conv2D	(None, 13, 13, 256)	884,992
BatchNormalization	(None, 13, 13, 256)	1024
MaxPooling2D	(None, 6, 6, 256)	0
Flatten	(None, 9216)	0
Dense	(None, 4096)	37,752,832
Dropout	(None, 4096)	0
Dense	(None, 4096)	16,781,312
Dropout	(None, 4096)	0
Dense	(None, 5)	20,485
	Total parameters:	58,284,101
	Trainable parameters:	58,281,349
	Non-trainable parameters:	2752

6. Implementation Aspects

In this section, we provide some important remarks about the implementation aspects of the proposed idea.

The computation of the CWT can be memory and computationally demanding. For this reason, we recommend not exceeding the chunk size; 30 ms or 50 ms represents a good compromise between the efficiency and tracking performance of the classifier due to the intrinsic non-stationarity of audio signals.

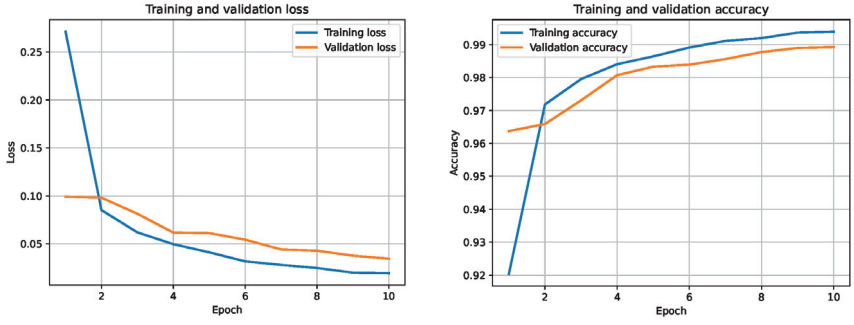
The CWT applied to a 30 ms chunk returns a matrix of the size 230×662. In view of using such data as the input to a state-of-the-art CNN, it is convenient to resize the matrix to a commonly used size. Generally, 227×227 (for AlexNet) or 224×224 (for GoogLeNet, ResNet, and similar architectures) are adequate choices.

However, saving more than 150,000 (see Table 1) floating-point matrices of 227×227 entries requires a large amount of disk space and a consistent quantity of RAM memory to load and process the dataset. For this purpose, after the resize, these matrices have been scaled to the interval $[0, 255]$, converted to integer numbers, and saved as images. In this way, it is possible to work with this dataset on a normal office PC while avoiding memory explosion.

Finally, to deal with such data, an additional Rescaling layer has been used in the customized version of AlexNet. This layer converts the integer input data back into the float interval $[0, 1]$.

7. Experimental Results

The proposed model has been trained on the considered dataset for 10 epochs by using 10% of the training set as the validation set (Python 3.10 source code can be downloaded from: https://github.com/mscarpiniti/CS-scalogram, accessed on 20 November 2023). The training and validation losses obtained during the training phase are shown in Figure 4a, while Figure 4b shows the corresponding training and validation accuracy. These figures demonstrate the effectiveness of the training, showing that the training procedure is quite stable after about seven epochs. Figure 4b also shows that, at convergence, the training accuracy is about 99.5%, while the validation one is about 99%.



(**a**) Loss. (**b**) Accuracy.

Figure 4. Training and validation loss (**a**) and accuracy (**b**) of the proposed approach.

To evaluate the proposed approach, we have also used the overall accuracy, the per-class precision, the per-class recall, and the per-class F1-score, as well as their weighted averages [53], computed on the test set. Moreover, the confusion matrix is shown in Figure 5. The confusion matrix clearly shows that the proposed approach is able to provide very good results for the classification of real-world signals recorded in construction sites. In fact, most of the instances are in the main diagonal of the matrix. There is a little confusion between the compactor (IRCOM), which has been confused with the JD50D excavator and the concrete mixer, and the CAT320E excavator, which is, again, mainly confused with the JD50D excavator and the concrete mixer. This behavior is due to the fact that all of these pieces of equipment have similar engines.

The results in terms of the precision, recall, and F1-score of the proposed approach are summarized in Table 3. In addition, this table confirms the conclusion drawn from the

confusion matrix in Figure 5: the JD50D and Concrete Mixer classes have lower precision, while the compactor (IRCOM) and CAT320E excavator show lower recall. However, the F1-score is quite stable among all classes. The Hitachi 50U excavator performs the best in between the five considered classes. Despite this slight variability in performance, the weighted averages of the considered metrics are very good and settled at 0.989.

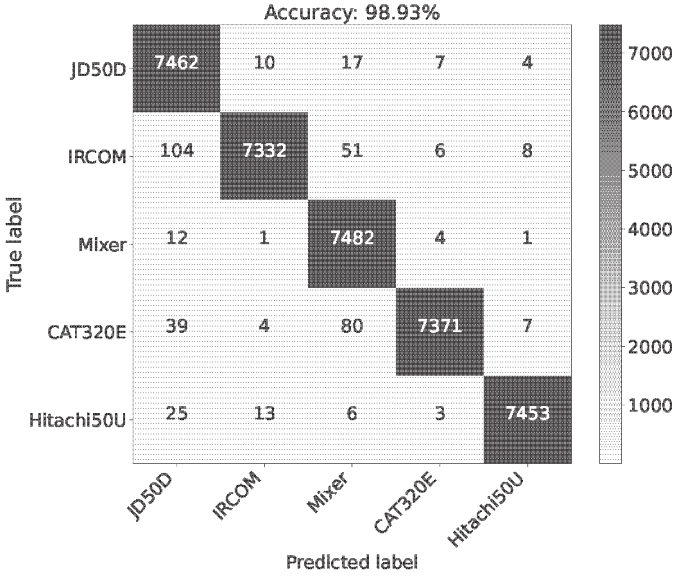


Figure 5. Confusion matrix obtained by the proposed approach.

Table 3. Per-class performance of the proposed approach.

Class	Precision	Recall	F1-Score
JD50D	0.976	0.995	0.986
IRCOM	0.996	0.977	0.987
Mixer	0.979	0.998	0.988
CAT320E	0.997	0.983	0.989
Hitachi50U	0.997	0.994	0.996
All classes	0.989	0.989	0.989

The proposed approach was compared with similar state-of-the-art solutions. Specifically, we compared our approach to the one proposed by Piczak in [4], based on a CNN fed by the spectrograms with corresponding deltas (i.e., the difference of the feature among two consecutive time instants); the approach proposed by Maccagno et al. in [31], based on a custom deep CNN (DCNN) fed by the spectrograms; and the approach proposed by Scarpiniti et al. in [34], based on an ESN working on several spectral features and a majority voting between adjacent chunks. The results obtained by these state-of-the-art approaches in terms of precision, recall, F1-score, and their weighted averages are shown in Tables 4, 5, and 6, respectively. The results presented in these tables confirm that the approach proposed in this paper (see Table 3) performs better than the state of the art for all of the considered metrics. Figure 6 summarizes all of the considered metrics for these compared approaches.

Table 4. Per-class performance of the Piczak approach in [4].

Class	Precision	Recall	F1-Score
JD50D	0.981	0.965	0.973
IRCOM	0.959	0.982	0.970
Mixer	0.942	0.945	0.943
CAT320E	0.894	0.973	0.932
Hitachi50U	0.944	0.795	0.863
All classes	0.944	0.932	0.936

Table 5. Per-class performance of the DCNN-based approach in [31].

Class	Precision	Recall	F1-Score
JD50D	0.955	0.972	0.963
IRCOM	0.957	0.979	0.968
Mixer	0.975	0.985	0.980
CAT320E	0.986	0.973	0.979
Hitachi50U	0.972	0.978	0.975
All classes	0.973	0.973	0.973

Table 6. Per-class performance of the ESN-based approach in [34].

Class	Precision	Recall	F1-Score
JD50D	0.901	0.937	0.919
IRCOM	0.899	0.974	0.935
Mixer	0.837	0.819	0.828
CAT320E	0.769	0.629	0.692
Hitachi50U	0.763	0.823	0.792
All classes	0.834	0.837	0.833

In addition, Tables 7 and 8 show the results of the works proposed by [44,46], which use scalogram-based approaches for acoustic scene classification. We adapt these approaches to work with the scalograms extracted from the construction site sounds. These tables show that, although the works proposed in [44,46] provide good results, the performance is slightly lower than the proposed approach reported in Table 3.

Table 7. Per-class performance of the approach proposed by Chen et al., 2018, in [44].

Class	Precision	Recall	F1-Score
JD50D	0.974	0.975	0.974
IRCOM	0.982	0.979	0.980
Mixer	0.981	0.984	0.982
CAT320E	0.988	0.979	0.984
Hitachi50U	0.975	0.983	0.979
All classes	0.980	0.980	0.980

Table 8. Per-class performance of the approach proposed by Copiaco et al., 2019, in [46].

Class	Precision	Recall	F1-Score
JD50D	0.972	0.973	0.972
IRCOM	0.981	0.977	0.979
Mixer	0.979	0.982	0.981
CAT320E	0.986	0.977	0.982
Hitachi50U	0.972	0.981	0.977
All classes	0.978	0.978	0.978

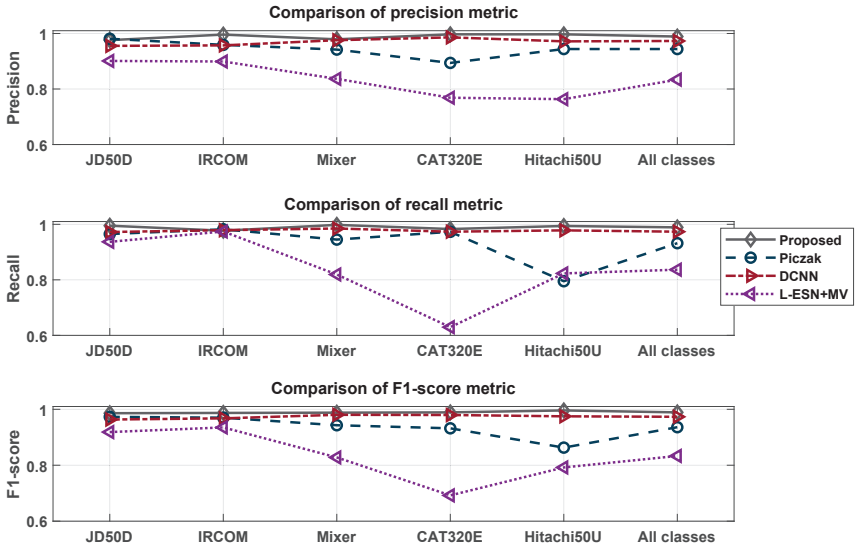


Figure 6. A visual comparison of all the compared approaches for the precision (**top**), recall (**middle**), and F1-score (**bottom**).

Although the Morlet wavelet in (4) is the most used and effective wavelet family in nonstationary audio analysis, we have also tested two other well-known and well-used wavelet families: the generalized Morse and the Bump wavelets [47], respectively. The results in terms of per-class precision, recall, and F1-score, and their related weighted averages, are shown in Table 9. From this table, we can argue that results of the generalized Morse wavelet are quite similar to those obtained by using the Morlet one (see Table 3). On the other hand, the results related to the Bump wavelet are slightly worse, even if they are quite good. The overall accuracies of these two approaches were 98.50% and 97.45%, respectively. These considerations confirm the effectiveness of the Morlet wavelet for analyzing audio signals in general and engine sounds in particular.

Finally, in Table 10, we summarize the previous results of the proposed approach (last row) and the compared ones by considering some additional machine learning and deep learning approaches. Specifically, Figure 7 shows the accuracy of the compared approaches as a bar plot. Among the machine learning techniques, we considered the results obtained by using a Support Vector Machine (SVM), the k-Nearest Neighbors (k-NN), the Multilayer Perceptron (MLP), and a random forest. All of these approaches provided reasonable results [5], though the results were worse than those provided by deep learning techniques. Among these last methods, we also considered an approach based on a Deep Recurrent Neural Network (DRNN) that exploits different spectral features [54] and one based on a Deep Belief Network (DBN) that works on a statistical ensemble of different spectral features [33]. For the implementation details, we refer to the related references. The results reported in Table 10 and Figure 7 clearly show once again the effectiveness of the proposed idea, which can be considered an effective and reliable approach for classifying real-world signals recorded in construction sites.

As a discussion, we can observe that scalograms generally capture salient localized events in sound frames, as shown by the horizontal lines or cloud-like points in Figure 3. In addition, the convolutional layers of the CNN are able to learn discriminative features, as shown in Figure 8, which, for example, shows the 256 feature maps of the fifth and last convolutional layer of the used architecture. Although single plots in the figure are quite small, it is clear that feature maps in the final layer are more specialized in detecting specific time scales. In fact, the scalogram in Figure 8 shows clear horizontal lines localized at a

specific scale. This kind of scale localization is typical of engines, a fundamental part of the machines considered in our dataset. This behavior justifies the better performance of the proposed approach for the classification of equipment sounds in construction sites.

Table 9. Per-class performance of the proposed approach by using the generalized Morse wavelet and the Bump wavelet. Overall accuracy is 98.50% and 97.45%, respectively.

Class	Generalized Morse Wavelet			Bump Wavelet		
	Precision	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
JD50D	0.985	0.977	0.981	0.964	0.983	0.974
IRCOM	0.986	0.983	0.985	0.990	0.967	0.978
Mixer	0.975	0.994	0.984	0.989	0.960	0.974
CAT320E	0.985	0.986	0.986	0.995	0.966	0.980
Hitachi50U	0.995	0.985	0.990	0.938	0.996	0.966
All classes	0.985	0.985	0.985	0.975	0.974	0.975

Table 10. Results of the compared approaches.

Approach	Accuracy	Precision	Recall	F1-Score
SVM [5]	83.66	0.846	0.838	0.842
k-NN [5]	85.28	0.860	0.853	0.857
MLP [5]	91.06	0.913	0.932	0.923
Random Forest [5]	93.16	0.934	0.932	0.933
Piczak [4]	90.03	0.944	0.932	0.936
DRNN [54]	95.32	0.955	0.953	0.954
DCNN [31]	97.08	0.973	0.973	0.973
DBN [33]	97.79	0.978	0.978	0.978
L-ESN+MV [34]	95.26	0.957	0.953	0.952
CNN [44]	97.99	0.980	0.980	0.980
CNN+SVM [46]	97.79	0.978	0.978	0.978
Proposed	98.93	0.989	0.989	0.989

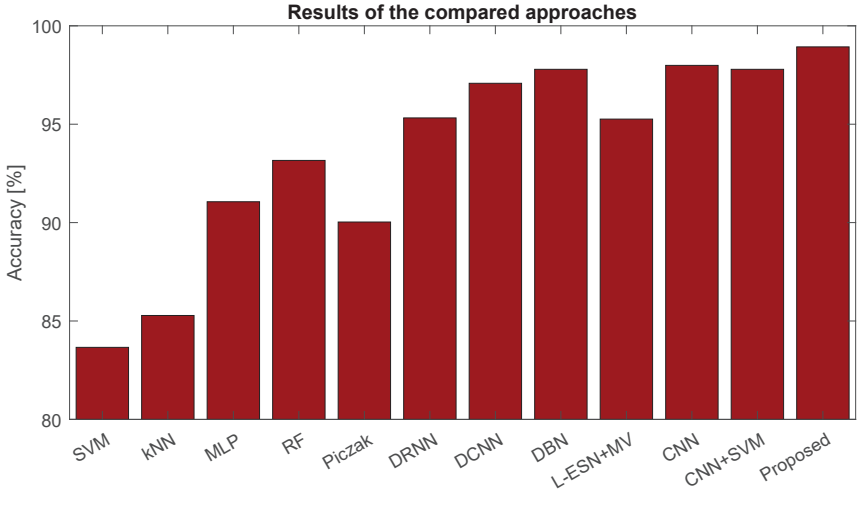


Figure 7. Accuracy of the compared approaches.

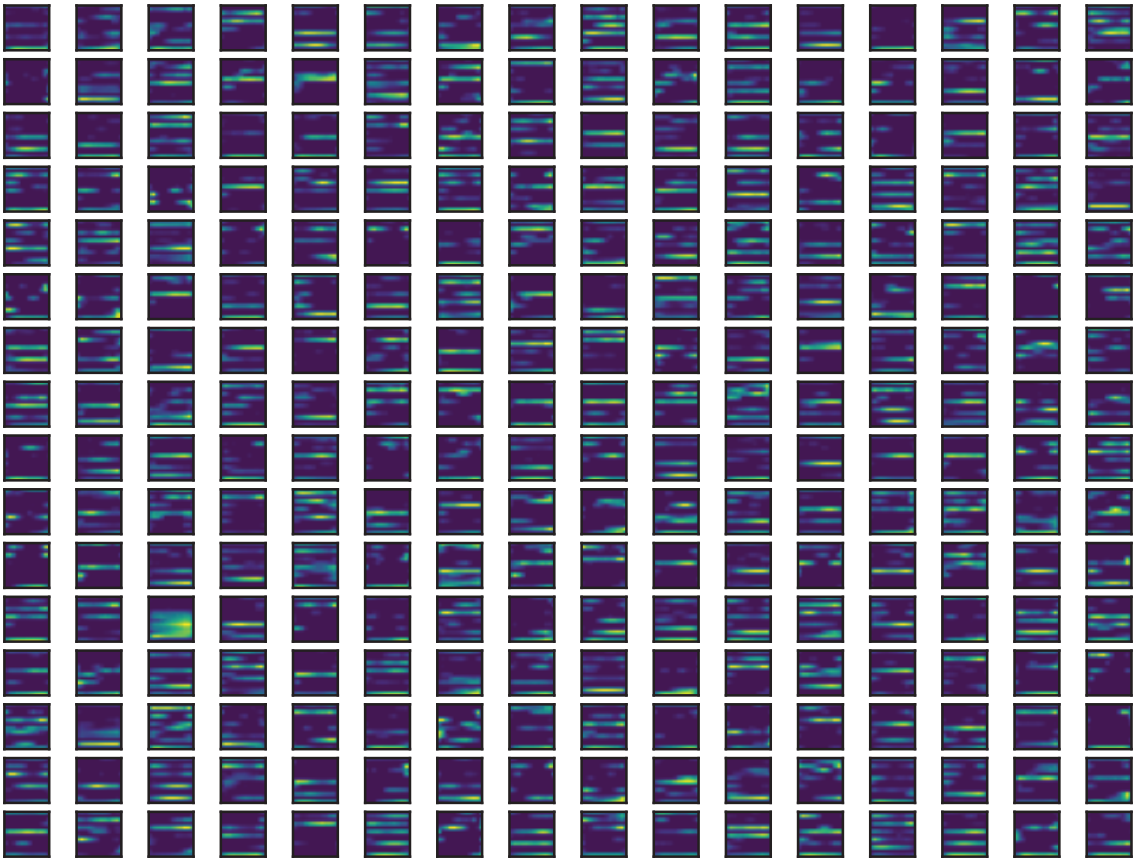


Figure 8. The 256 feature maps of the fifth and last convolutional layer of the trained architecture. Maps have been normalized in $[0, 1]$ for visualization: lighter colors are close to 1, while darker colors are close to 0.

8. Conclusions

In this paper, we have investigated the effectiveness of a Convolutional Neural Network (CNN) fed by scalograms in the classification of audio signals acquired in real-world construction sites. Specifically, after splitting the recorded signals into smaller chunks, the scalogram (i.e., the squared magnitude of the Continuous Wavelet Transform) has been computed and used as input to a customized version of the well-known AlexNet. The customization takes into account the single channel of the scalogram input and the reduced number of output classes. Some experimental results and comparisons with other state-of-the-art approaches confirm the effectiveness of the proposed idea, showing an overall accuracy of 98.9%.

In future work, we will investigate the effect of choosing different types of wavelet functions and the idea of early data fusion, i.e., by joining the scalograms with other bi-dimensional representations, such as the spectrogram or similar ones, and providing this augmented representation as the input to a CNN.

Author Contributions: Conceptualization, M.S.; methodology, R.P. and Y.-C.L.; software, M.S. and R.P.; validation, M.S. and Y.-C.L.; formal analysis, Y.-C.L. and M.S.; investigation, M.S., Y.-C.L. and R.P.; data curation, Y.-C.L.; writing—original draft preparation, M.S.; writing—review and editing, Y.-C.L., R.P. and M.S.; visualization, M.S.; funding acquisition, M.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially funded by Sapienza University of Rome grant numbers RM12117A39E2E9A7 and RM122180FB3CA3F2.

Institutional Review Board Statement: Not applicable

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ACSM	Automatic Construction Site Monitoring
AI	Artificial Intelligence
CNN	Convolutional Neural Network
CWT	Continuous Wavelet Transform
DBN	Deep Belief Network
DCNN	Deep Convolutional Neural Network
DL	Deep Learning
DRNN	Deep Recurrent Neural Network
ESC	Environmental Sound Classification
ESN	Echo State Network
GPS	Global Positioning System
LSTM	Long Short-Term Memory
ML	Machine Learning
MLP	Multilayer Perceptron
RNN	Recurrent Neural Network
STFT	Short Time Fourier Transform
SVM	Support Vector Machine

References

1. Bansal, A.; Garg, N.K. Environmental Sound Classification: A descriptive review of the literature. *Intell. Syst. Appl.* **2022**, *16*, 200115. [CrossRef]
2. Zaman, K.; Sah, M.; Direkoglu, C.; Unoki, M. A Survey of Audio Classification Using Deep Learning. *IEEE Access* **2023**, *11*, 106620–106649. [CrossRef]
3. Demir, F.; Abdullah, D.A.; Sengur, A. A New Deep CNN Model for Environmental Sound Classification. *IEEE Access* **2020**, *8*, 66529–66537. [CrossRef]
4. Piczak, K.J. Environmental sound classification with convolutional neural networks. In Proceedings of the 2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP 2015), Boston, MA, USA, 17–20 September 2015; pp. 1–6. [CrossRef]
5. Lee, Y.C.; Scarpiniti, M.; Uncini, A. Advanced Sound Classifiers and Performance Analyses for Accurate Audio-Based Construction Project Monitoring. *ASCE J. Comput. Civ. Eng.* **2020**, *34*, 1–11. [CrossRef]
6. Xiong, W.; Xu, X.; Chen, L.; Yang, J. Sound-Based Construction Activity Monitoring with Deep Learning. *Buildings* **2022**, *12*, 1947. [CrossRef]
7. Kim, I.C.; Kim, Y.J.; Chin, S.Y. Sound Localization Framework for Construction Site Monitoring. *Appl. Sci.* **2022**, *12*, 783. [CrossRef]
8. Sanhudo, L.; Calvetti, D.; Martins, J.; Ramos, N.; Mêda, P.; Gonçalves, M.; Sousa, H. Activity classification using accelerometers and machine learning for complex construction worker activities. *J. Build. Eng.* **2021**, *35*, 102001. [CrossRef]
9. Jungmann, M.; Ungureanu, L.; Hartmann, T.; Posada, H.; Chacon, R. Real-Time Activity Duration Extraction of Crane Works for Data-Driven Discrete Event Simulation. In Proceedings of the 2022 Winter Simulation Conference (WSC 2022), Singapore, 11–14 December 2022; pp. 2365–2376. [CrossRef]

10. Sherafat, B.; Ahn, C.R.; Akhavian, R.; Behzadan, A.H.; Golparvar-Fard, M.; Kim, H.; Lee, Y.C.; Rashidi, A.; Azar, E.R. Automated Methods for Activity Recognition of Construction Workers and Equipment: State-of-the-Art Review. *J. Constr. Eng. Manag.* **2020**, *146*, 03120002. [CrossRef]
11. Rao, A.; Radanovic, M.; Liu, Y.; Hu, S.; Fang, Y.; Khoshelham, K.; Palaniswami, M.; Ngo, T. Real-time monitoring of construction sites: Sensors, methods, and applications. *Autom. Constr.* **2022**, *136*, 104099. [CrossRef]
12. Zhou, Z.; Wei, L.; Yuan, J.; Cui, J.; Zhang, Z.; Zhuo, W.; Lin, D. Construction safety management in the data-rich era: A hybrid review based upon three perspectives of nature of dataset, machine learning approach, and research topic. *Adv. Eng. Inform.* **2023**, *58*, 102144. [CrossRef]
13. Navon, R.; Sacks, R. Assessing research issues in Automated Project Performance Control (APPC). *Autom. Constr.* **2007**, *16*, 474–484. [CrossRef]
14. Akbal, E.; Tuncer, T. A learning model for automated construction site monitoring using ambient sounds. *Autom. Constr.* **2022**, *134*, 104094. [CrossRef]
15. Meng, Q.; Peng, Q.; Li, Z.; Hu, X. Big Data Technology in Construction Safety Management: Application Status, Trend and Challenge. *Buildings* **2022**, *12*, 533. [CrossRef]
16. Rashid, K.M.; Louis, J. Activity identification in modular construction using audio signals and machine learning. *Autom. Constr.* **2020**, *119*, 103361. [CrossRef]
17. Jacobsen, E.; Teizer, J. Deep Learning in Construction: Review of Applications and Potential Avenues. *J. Comput. Civ. Eng.* **2022**, *36*, 1010. [CrossRef]
18. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Los Alamitos, CA, USA, 18–22 June 2015; pp. 1–9. [CrossRef]
19. Wyse, L. Audio Spectrogram Representations for Processing with Convolutional Neural Networks. In Proceedings of the First International Workshop on Deep Learning and Music joint with IJCNN, Anchorage, AK, USA, 17–18 May 2017; pp. 37–41.
20. Dörfler, M.; Bammer, R.; Grill, T. Inside the spectrogram: Convolutional Neural Networks in audio processing. In Proceedings of the 2017 International Conference on Sampling Theory and Applications (SampTA), Bordeaux, France, 8–12 July 2017; pp. 152–155. [CrossRef]
21. Traunmüller, H. Analytical expressions for the tonotopic sensory scale. *J. Acoust. Soc. Am.* **1990**, *88*, 97–100. [CrossRef]
22. Auger, F.; Flandrin, P.; Lin, Y.T.; McLaughlin, S.; Meignen, S.; Oberlin, T.; Wu, H.T. Time-Frequency Reassignment and Synchrosqueezing: An Overview. *IEEE Signal Process. Mag.* **2013**, *30*, 32–41. [CrossRef]
23. Mallat, S. *A Wavelet Tour of Signal Processing: The Sparse Way*, 3rd ed.; Academic Press: Cambridge, MA, USA, 2009.
24. Sacks, R.; Brilakis, I.; Pikas, E.; Xie, H.; Girolami, M. Construction with digital twin information systems. *Data-Centric Eng.* **2020**, *1*, e14. [CrossRef]
25. Deng, R.; Li, C. Digital Intelligent Management Platform for High-Rise Building Construction Based on BIM Technology. *Int. J. Adv. Comput. Sci. Appl.* **2022**, *13*, 1057–1067. [CrossRef]
26. Mansoor, A.; Liu, S.; Ali, G.; Bouferguene, A.; Al-Hussein, M. Scientometric analysis and critical review on the application of deep learning in the construction industry. *Can. J. Civ. Eng.* **2023**, *50*, 253–269. [CrossRef]
27. Garcia, J.; Villavicencio, G.; Altimiras, F.; Crawford, B.; Soto, R.; Minatogawa, V.; Franco, M.; Martínez-Muñoz, D.; Yepes, V. Machine learning techniques applied to construction: A hybrid bibliometric analysis of advances and future directions. *Autom. Constr.* **2022**, *142*, 104532. [CrossRef]
28. Cao, J.; Wang, W.; Wang, J.; Wang, R. Excavation Equipment Recognition Based on Novel Acoustic Statistical Features. *IEEE Trans. Cybern.* **2017**, *47*, 4392–4404. [CrossRef] [PubMed]
29. Jeong, G.; Ahn, C.R.; Park, M. Constructing an Audio Dataset of Construction Equipment from Online Sources for Audio-Based Recognition. In Proceedings of the 2022 Winter Simulation Conference (WSC), Singapore, 11–14 December 2022; pp. 2354–2364. [CrossRef]
30. Wang, G.; Yu, Y.; Li, H. Automated activity recognition of construction workers using single in-pocket smartphone and machine learning methods. In *Proceedings of the IOP Conference Series: Earth and Environmental Science*; IOP Publishing: Bristol, UK, 2022; Volume 1101, p. 072008. [CrossRef]
31. Maccagno, A.; Mastropietro, A.; Mazziotta, U.; Scarpiniti, M.; Lee, Y.C.; Uncini, A. A CNN Approach for Audio Classification in Construction Sites. In *Progresses in Artificial Intelligence and Neural Systems*; Esposito , A.; Faudez-Zanuy, M., Morabito, F.C., Pasero, E., Eds.; Springer: Singapore, 2021; Volume 184, pp. 371–381. [CrossRef]
32. Sherafat, B.; Rashidi, A.; Asgari, S. Sound-based multiple-equipment activity recognition using convolutional neural networks. *Autom. Constr.* **2022**, *135*, 104104. [CrossRef]
33. Scarpiniti, M.; Colasante, F.; Di Tanna, S.; Ciancia, M.; Lee, Y.C.; Uncini, A. Deep Belief Network based audio classification for construction sites monitoring. *Expert Syst. Appl.* **2021**, *177*, 1–14. [CrossRef]
34. Scarpiniti, M.; Bini, E.; Ferraro, M.; Giannetti, A.; Comminiello, D.; Lee, Y.C.; Uncini, A. Leaky Echo State Network for Audio Classification in Construction Sites. In *Applications of Artificial Intelligence and Neural Systems to Data Science*; Esposito, A.; Faudez-Zanuy, M.; Morabito, F.C.; Pasero, E., Eds.; Springer: Singapore, 2023; Volume 360. [CrossRef]
35. Kim, I.; Kim, Y.; Chin, S. Deep-Learning-Based Sound Classification Model for Concrete Pouring Work Monitoring at a Construction Site. *Appl. Sci.* **2023**, *13*, 4789. [CrossRef]

36. Peng, Z.; Kong, Q.; Yuan, C.; Li, R.; Chi, H.L. Development of acoustic denoising learning network for communication enhancement in construction sites. *Adv. Eng. Inform.* **2023**, *56*, 101981. [CrossRef]
37. Akbal, E.; Barua, P.D.; Dogan, S.; Tuncer, T.; Acharya, U.R. DesPatNet25: Data encryption standard cipher model for accurate automated construction site monitoring with sound signals. *Expert Syst. Appl.* **2022**, *193*, 116447. [CrossRef]
38. Sherafat, B.; Rashidi, A.; Lee, Y.C.; Ahn, C.R. A Hybrid Kinematic-Acoustic System for Automated Activity Detection of Construction Equipment. *Sensors* **2019**, *19*, 4286. [CrossRef]
39. Kim, J.; Chi, S. Action recognition of earthmoving excavators based on sequential pattern analysis of visual features and operation cycles. *Autom. Constr.* **2019**, *104*, 255–264. [CrossRef]
40. Soltani, M.M.; Zhu, Z.; Hammad, A. Framework for Location Data Fusion and Pose Estimation of Excavators Using Stereo Vision. *J. Comput. Civ. Eng.* **2018**, *32*, 04018045. [CrossRef]
41. Jung, S.; Jeoung, J.; Lee, D.E.; Jang, H.; Hong, T. Visual–auditory learning network for construction equipment action detection. *Comput. Aided Civ. Infrastruct. Eng.* **2023**, *38*, 1916–1934. [CrossRef]
42. Elelu, K.; Le, T.; Le, C. Collision Hazard Detection for Construction Worker Safety Using Audio Surveillance. *J. Constr. Eng. Manag.* **2023**, *149*. [CrossRef]
43. Dang, K.; Elelu, K.; Le, T.; Le, C. Augmented Hearing of Auditory Safety Cues for Construction Workers: A Systematic Literature Review. *Sensors* **2022**, *22*, 9135. [CrossRef] [PubMed]
44. Chen, H.; Zhang, P.; Bai, H.; Yuan, Q.; Bao, X.; Yan, Y. Deep Convolutional Neural Network with Scalogram for Audio Scene Modeling. In Proceedings of the Interspeech, Hyderabad, India, 2–6 September 2018; pp. 3304–3308. [CrossRef]
45. Faysal, A.; Ngui, W.K.; Lim, M.H.; Leong, M.S. Noise Eliminated Ensemble Empirical Mode Decomposition Scalogram Analysis for Rotating Machinery Fault Diagnosis. *Sensors* **2021**, *21*, 8114. [CrossRef] [PubMed]
46. Copiaco, A.; Ritz, C.; Fasciani, S.; Abdulaziz, N. Scalogram Neural Network Activations with Machine Learning for Domestic Multi-channel Audio Classification. In Proceedings of the 2019 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), Ajman, United Arab Emirates, 10–12 December 2019; pp. 1–6. [CrossRef]
47. Ren, Z.; Qian, K.; Zhang, Z.; Pandit, V.; Baird, A.; Schuller, B. Deep Scalogram Representations for Acoustic Scene Classification. *IEEE/CAA J. Autom. Sin.* **2018**, *5*, 662–669. [CrossRef]
48. Flanagan, J.L. *Speech Analysis, Synthesis and Perception*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 1972. [CrossRef]
49. Gupta, P.; Chodingala, P.K.; Patil, H.A. Morlet Wavelet-Based Voice Liveness Detection using Convolutional Neural Network. In Proceedings of the 2022 30th European Signal Processing Conference (EUSIPCO), Belgrade, Serbia, 29 August–2 September 2022, pp. 100–104. [CrossRef]
50. Lin, J. Feature extraction of machine sound using wavelet and its application in fault diagnosis. *NDT E Int.* **2001**, *34*, 25–30. [CrossRef]
51. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of the 25th International Conference on Neural Information Processing Systems, Stateline, NV, USA, 3–8 December 2012, pp. 1097–1105. [CrossRef]
52. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the 3rd International Conference for Learning Representations (ICLR 2015), San Diego, USA, 7–9 May 2015; pp. 1–15.
53. Powers, D.M.W. Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness and Correlation. *J. Mach. Learn. Technol.* **2011**, *2*, 37–63. [CrossRef]
54. Scarpiniti, M.; Comminiello, D.; Uncini, A.; Lee, Y.C. Deep recurrent neural networks for audio classification in construction sites. In Proceedings of the 28th European Signal Processing Conference (EUSIPCO 2020), Amsterdam, The Netherlands, 24–28 August 2020; pp. 810–814. [CrossRef]

MDPI
St. Alban-Anlage 66
4052 Basel
Switzerland
www.mdpi.com

Applied Sciences Editorial Office
E-mail: applsci@mdpi.com
www.mdpi.com/journal/applsci

Academic Open
Access Publishing

mdpi.com

ISBN 978-3-7258-1060-4