



# Detection and Tracking of Targets in Forward-Looking InfraRed (FLIR) Imagery

Edited by

Andrea Sanna and Fabrizio Lamberti

Printed Edition of the Special Issue Published in *Sensors*



[www.mdpi.com/journal/sensors](http://www.mdpi.com/journal/sensors)

Andrea Sanna and Fabrizio Lamberti (Eds.)

# **Detection and Tracking of Targets in Forward-Looking InfraRed (FLIR) Imagery**



This book is a reprint of the Special Issue that appeared in the online, open access journal, *Sensors* (ISSN 1424-8220) in 2014 (available at: [http://www.mdpi.com/journal/sensors/special\\_issues/FLIR](http://www.mdpi.com/journal/sensors/special_issues/FLIR)).

*Guest Editors*

Andrea Sanna and Fabrizio Lamberti  
Politecnico di Torino, Dipartimento di Automatica e Informatica  
Italy

*Editorial Office*

MDPI AG  
Klybeckstrasse 64  
Basel, Switzerland

*Publisher*

Shu-Kun Lin

*Managing Editor*

Limei Huang

**1. Edition 2015**

MDPI • Basel • Beijing • Wuhan • Barcelona

ISBN 978-3-03842-052-1 (Hbk)

ISBN 978-3-03842-053-8 (PDF)

Articles in this volume are Open Access and distributed under the Creative Commons Attribution license (CC BY), which allows users to download, copy and build upon published articles even for commercial purposes, as long as the author and publisher are properly credited, which ensures maximum dissemination and a wider impact of our publications. The book taken as a whole is © 2015 MDPI, Basel, Switzerland, distributed under the terms and conditions of the Creative Commons by Attribution (CC BY-NC-ND) license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

## Biographical Sketch – Guest Editors

**Fabrizio Lamberti** received the M.Sc. and the Ph.D. degrees in computer engineering from Politecnico di Torino university in Italy in 2000 and 2005, respectively. He is an Associate Professor position at the Dipartimento di Automatica e Informatica at Politecnico di Torino university in Italy. He has published a number of technical papers in international books, journals and conferences in the areas of computer graphics, computer vision, human-machine interaction and intelligent information processing. He has served as a reviewer, program or organization committee member for several conferences. He is a member of the Editorial Advisory Board of international journals. He is a senior member of the IEEE and the IEEE Computer Society.

**Andrea Sanna** received the M.Sc. degree in electronic engineering and the Ph.D. degree in computer engineering from Politecnico di Torino university in Italy in 1993 and 1997, respectively. He is an Associate Professor at the Dipartimento di Automatica e Informatica at Politecnico di Torino university in Italy. He has authored and co-authored several papers in the areas of computer graphics, virtual reality, distributed computing, and computational geometry. He is a senior member of the ACM.



# Table of Contents

List of Contributors .....	VII
Preface .....	XI
<b>Mohammad S. Alam and Sharif M. A. Bhuiyan</b>	
Trends in Correlation-Based Pattern Recognition and Tracking in Forward-Looking Infrared Imagery	
Reprinted from: <i>Sensors</i> <b>2014</b> , <i>14</i> (8), 13437–13475	
<a href="http://www.mdpi.com/1424-8220/14/8/13437">http://www.mdpi.com/1424-8220/14/8/13437</a> .....	1
<b>Peter Christiansen, Kim Arild Steen, Rasmus Nyholm Jørgensen and Henrik Karstoft</b>	
Automated Detection and Recognition of Wildlife Using Thermal Cameras	
Reprinted from: <i>Sensors</i> <b>2014</b> , <i>14</i> (8), 13778–13793	
<a href="http://www.mdpi.com/1424-8220/14/8/13778">http://www.mdpi.com/1424-8220/14/8/13778</a> .....	41
<b>Antonio Fernández-Caballero, María T. López and Juan Serrano-Cuerda</b>	
Thermal-Infrared Pedestrian ROI Extraction through Thermal and Motion Information Fusion	
Reprinted from: <i>Sensors</i> <b>2014</b> , <i>14</i> (4), 6666–6676	
<a href="http://www.mdpi.com/1424-8220/14/4/6666">http://www.mdpi.com/1424-8220/14/4/6666</a> .....	57
<b>Rikke Gade and Thomas B. Moeslund</b>	
Thermal Tracking of Sports Players	
Reprinted from: <i>Sensors</i> <b>2014</b> , <i>14</i> (8), 13679–13691	
<a href="http://www.mdpi.com/1424-8220/14/8/13679">http://www.mdpi.com/1424-8220/14/8/13679</a> .....	68
<b>Jiulu Gong, Guoliang Fan, Liangjiang Yu, Joseph P. Havlicek, Derong Chen and Ningjun Fan</b>	
Joint Target Tracking, Recognition and Segmentation for Infrared Imagery using a Shape Manifold-based Level Set	
Reprinted from: <i>Sensors</i> <b>2014</b> , <i>14</i> (6), 10124–10145	
<a href="http://www.mdpi.com/1424-8220/14/6/10124">http://www.mdpi.com/1424-8220/14/6/10124</a> .....	80

- Riad I. Hammoud, Cem S. Sahin, Erik P. Blasch, Bradley J. Rhodes and Tao Wang**  
Automatic Association of Chats and Video Tracks for Activity Learning and Recognition in Aerial Video Surveillance  
Reprinted from: *Sensors* **2014**, *14*(10), 19843–19860  
<http://www.mdpi.com/1424-8220/14/10/19843> ..... 102
- Sungho Kim and Joohyoung Lee**  
Small Infrared Target Detection by Region-Adaptive Clutter Rejection for Sea-based Infrared Search and Track  
Reprinted from: *Sensors* **2014**, *14*(7), 13210–13242  
<http://www.mdpi.com/1424-8220/14/7/13210> ..... 120
- Xin Li, Rui Guo and Chao Chen**  
Robust Pedestrian Tracking and Recognition from FLIR Video: A Unified Approach via Sparse Coding  
Reprinted from: *Sensors* **2014**, *14*(6), 11245–11259  
<http://www.mdpi.com/1424-8220/14/6/11245> ..... 154
- Zheng-Zhou Li, Jing Chen, Qian Hou, Hong-Xia Fu, Zhen Dai, Gang Jin, Ru-Zhang Li and Chang-Ju Liu**  
Sparse Representation for Infrared Dim Target Detection via a Discriminative Over-Complete Dictionary Learned Online  
Reprinted from: *Sensors* **2014**, *14*(6), 9451–9470  
<http://www.mdpi.com/1424-8220/14/6/9451> ..... 169
- Gianluca Paravati and Stefano Esposito**  
Relevance-based Template Matching for Tracking Targets in FLIR Imagery  
Reprinted from: *Sensors* **2014**, *14*(8), 14106–14130  
<http://www.mdpi.com/1424-8220/14/8/14106> ..... 188
- Pablo Ricaurte, Carmen Chilán, Cristhian A. Aguilera-Carrasco, Boris X. Vintimilla and Angel D. Sappa**  
Feature Point Descriptors: Infrared and Visible Spectra  
Reprinted from: *Sensors* **2014**, *14*(2), 3690–3701  
<http://www.mdpi.com/1424-8220/14/2/3690> ..... 214

# List of Contributors

**Cristhian A. Aguilera-Carrasco:** Computer Science Department, Universitat Autònoma de Barcelona, Campus UAB, 08193 Bellaterra, Barcelona, Spain; Computer Vision Center, Edifici O, Campus UAB, 08193 Bellaterra, Barcelona, Spain.

**Mohammad S. Alam:** Department of Electrical and Computer Engineering, University of South, Alabama Mobile, AL 36688-0002, USA.

**Sharif M. A. Bhuiyan:** Department of Electrical Engineering, Tuskegee University, Tuskegee, AL 36088, USA.

**Erik P. Blasch:** Air Force Research Lab, Rome, NY 13441, USA.

**Chao Chen:** Department of Electrical and Computer Engineering, University of Missouri, Columbia, MO 65211, USA.

**Derong Chen:** School of Mechatronical Engineering, Beijing Institute of Technology, No. 5, Zhongguancun South Street, Haidian District, Beijing 100081, China.

**Jing Chen:** School of Electrical and Computer Engineering, University of Oklahoma, 110 West Boyd, DEH 150Norman, OK 73019, USA.

**Carmen Chilán:** CIDIS-FIEC, Escuela Superior Politécnica del Litoral (ESPOL), Campus Gustavo Galindo, Km 30.5 vía Perimetral, P.O. Box 09-01-5863, Guayaquil, Ecuador.

**Peter Christiansen:** Department of Engineering, Aarhus University, Finlandsgade 22, Aarhus, Denmark.

**Zhen Dai:** School of Electrical and Computer Engineering, University of Oklahoma, 110 West Boyd, DEH 150Norman, OK 73019, USA.

**Stefano Esposito:** Politecnico di Torino, Dipartimento di Automatica e Informatica, Corso Duca degli Abruzzi 24, 10129 Torino, Italy.

**Guoliang Fan:** School of Electrical and Computer Engineering, Oklahoma State University, 202 Engineering South, Stillwater, OK 74078, USA.

**Ningjun Fan:** School of Mechatronical Engineering, Beijing Institute of Technology, No. 5, Zhongguancun South Street, Haidian District, Beijing 100081, China.

**Antonio Fernández-Caballero:** Departamento de Sistemas Informáticos, Universidad de Castilla-La Mancha, 02071-Albacete, Spain; Instituto de Investigación en Informática de Albacete, 02071-Albacete, Spain.

**Hong-Xia Fu:** School of Electrical and Computer Engineering, University of Oklahoma, 110 West Boyd, DEH 150Norman, OK 73019, USA.

**Rikke Gade:** Visual Analysis of People Lab, Aalborg University, Rendsburggade 14, 9000 Aalborg, Denmark.

**Jiulu Gong:** School of Mechatronical Engineering, Beijing Institute of Technology, No. 5, Zhongguancun South Street, Haidian District, Beijing 100081, China.



**Rui Guo:** Department of EECS, University of Tennessee, Knoxville, TN 37996, USA.

**Riad I. Hammoud:** BAE Systems, Burlington, MA 01803, USA.

**Joseph P. Havlicek:** School of Electrical and Computer Engineering, University of Oklahoma, 110 West Boyd, DEH 150Norman, OK 73019, USA.

**Qian Hou:** School of Electrical and Computer Engineering, University of Oklahoma, 110 West Boyd, DEH 150Norman, OK 73019, USA.

**Gang Jin:** China Aerodynamics Research and Development Center, Mianyang 621000, China.

**Rasmus Nyholm Jørgensen:** Department of Engineering, Aarhus University, Finlandsgade 22, Aarhus, Denmark.

**Henrik Karstoft:** Department of Engineering, Aarhus University, Finlandsgade 22, Aarhus, Denmark.

**Sungho Kim:** Yeungnam University 280 Daehak-Ro, Gyeongsan, Gyeongbuk 712-749, Korea.

**Joohyoung Lee:** Agency for Defense Development, 111 Sunam-dong, Daejeon 305-600, Korea.

**Ru-Zhang Li:** National Laboratory of Analogue Integrated Circuits, No. 24 Research Institute of China Electronics Technology Group Corporation, Chongqing 400060, China.

**Xin Li:** Lane Department of CSEE, Morgantown, WV 26506-6109, USA.

**Zheng-Zhou Li:** School of Electrical and Computer Engineering, University of Oklahoma, 110 West Boyd, DEH 150Norman, OK 73019, USA; National Laboratory of Analogue Integrated Circuits, No. 24 Research Institute of China Electronics Technology Group Corporation, Chongqing 400060, China.

**Chang-Ju Liu:** National Laboratory of Analogue Integrated Circuits, No. 24 Research Institute of China Electronics Technology Group Corporation, Chongqing 400060, China.

**María T. López:** Departamento de Sistemas Informáticos, Universidad de Castilla-La Mancha, 02071-Albacete, Spain; Instituto de Investigación en Informática de Albacete, 02071-Albacete, Spain.

**Thomas B. Moeslund:** Visual Analysis of People Lab, Aalborg University, Rendsburggade 14, 9000 Aalborg, Denmark.

**Gianluca Paravati:** Politecnico di Torino, Dipartimento di Automatica e Informatica, Corso Duca degli Abruzzi 24, 10129 Torino, Italy.

**Bradley J. Rhodes:** BAE Systems, Burlington, MA 01803, USA.

**Pablo Ricaurte:** CIDIS-FIEC, Escuela Superior Politécnica del Litoral (ESPOL), Campus Gustavo Galindo, Km 30.5 vía Perimetral, P.O. Box 09-01-5863, Guayaquil, Ecuador.

**Cem S. Sahin:** BAE Systems, Burlington, MA 01803, USA.

**Angel D. Sappa:** CIDIS-FIEC, Escuela Superior Politécnica del Litoral (ESPOL), Campus Gustavo Galindo, Km 30.5 vía Perimetral, P.O. Box 09-01-5863, Guayaquil, Ecuador; Computer Vision Center, Edifici O, Campus UAB, 08193 Bellaterra, Barcelona, Spain.

**Juan Serrano-Cuerda:** Instituto de Investigación en Informática de Albacete, 02071-Albacete, Spain.

**Kim Arild Steen:** Department of Engineering, Aarhus University, Finlandsgade 22, Aarhus, Denmark.

**Boris X. Vintimilla:** CIDIS-FIEC, Escuela Superior Politécnica del Litoral (ESPOL), Campus Gustavo Galindo, Km 30.5 vía Perimetral, P.O. Box 09-01-5863, Guayaquil, Ecuador .

**Tao Wang:** BAE Systems, Burlington, MA 01803, USA .

**Liangjiang Yu:** School of Electrical and Computer Engineering, Oklahoma State University, 202 Engineering South, Stillwater, OK 74078, USA.



# Preface

Many vision-based applications, like security monitoring, autonomous guidance and activity recognition, to name a few, require fast and robust techniques for object detection and tracking. In the literature, a wide range of approaches have been presented to deal with challenges posed by changes in object appearance, occlusions, sensor motion, *etc.* Many of these approaches were designed to operate on visible light images, captured by monocular or stereo cameras. A possible limit of strategies working (only) in the visible light spectrum is that they might be unable to get the expected results under poor visibility conditions (at night, or with fog, rain, snow, *etc.*).

Thanks to the great advancements made in the last few years, after having been used for a long time only in the military domain infrared sensors started to be exploited extensively also for industrial and consumer applications. Particular attention has been dedicated to long-wave infrared (LWIR) light in the 8-12  $\mu\text{m}$  range and to forward-looking infrared (FLIR) sensors. In fact, intensity levels in FLIR images are mainly determined by objects temperature and radiated heat, which make them far less dependent on illumination conditions and visible surface characteristics like colors, textures, *etc.*

Infrared light images are often considered also in combination with visible light ones, with the aim of designing algorithms capable to fuse information gathered simultaneously by two (or more) sensors and of benefiting from the advantages of both the domains.

Despite significant results achieved by sensor fusion approaches, there are still important issues to address in the field of infrared object detection and tracking. For instance, in visible light imagery, occlusion conditions that may occur, e.g., when tracking pedestrian targets in crowded environments, could be effectively managed by exploiting clothes color information. At night, the above situations could be addressed only by working in the infrared domain. Unfortunately, even in the infrared domain, thermal signatures may lack the required discriminative power and, when pedestrians overlap, it might get extremely difficult (if not impossible) to separate them.

Detection and tracking in FLIR images are made even more complex by other critical factors. In fact, real-life images captured by surveillance cameras, Unmanned Aerial Vehicles (UAVs) and other stationary and non-stationary sensors are generally characterized by low resolutions, strong clutter and poor signal-to-noise ratios.

Important inputs to research in this field come from experiences made with visible light images, as a number of challenges are actually shared between the two domains. To make an example by still referring to the above scenario related to pedestrian tracking, both visible and infrared light algorithms have to deal with frequently changing target signatures, whose aspect and motion are hard to predict because of the complexity of the human gait as well as of the nature of pedestrian-to-pedestrian and pedestrian-to-environment relations.

The aim of this book is to provide a glimpse of some of the main trends that can be identified in today's strategies for target detection and tracking in FLIR imagery, by considering several orthogonal perspectives. One of the perspectives is about the richness of application fields, which can be rather heterogeneous and encompass dim target detection from ground-based Earth Observation and sea vessel-mounted sensors, automatic tracking of humans, animals and military vehicles, *etc.* Another perspective pertains the specific issues of detection and tracking in FLIR imagery, which include image registration, target representation and recognition, occlusion handling, multi-target association, clutter removal and latency reduction, among others. A final perspective that is taken into account comes from considering whether algorithms are designed to specifically cope with the peculiarities of FLIR imagery, adapt or combine well-known techniques used with visible light images to suit the requirements of the infrared spectrum, or study how existing alternatives could perform in the considered domain.

By motivating the attention that infrared spectrum-based algorithms, and related sensors, are receiving today from all the above points of view, the ultimate goal of this book is to provide a solid ground for research activities of tomorrow and to further support next advancements in this field.

Andrea Sanna and Fabrizio Lamberti  
*Guest Editors*





# Trends in Correlation-Based Pattern Recognition and Tracking in Forward-Looking Infrared Imagery

Mohammad S. Alam and Sharif M. A. Bhuiyan

**Abstract:** In this paper, we review the recent trends and advancements on correlation-based pattern recognition and tracking in forward-looking infrared (FLIR) imagery. In particular, we discuss matched filter-based correlation techniques for target detection and tracking which are widely used for various real time applications. We analyze and present test results involving recently reported matched filters such as the maximum average correlation height (MACH) filter and its variants, and distance classifier correlation filter (DCCF) and its variants. Test results are presented for both single/multiple target detection and tracking using various real-life FLIR image sequences.

Reprinted from *Sensors*. Cite as: Mohammad, S.A.; Sharif, M.A.B. Trends in Correlation-Based Pattern Recognition and Tracking in Forward-Looking Infrared Imagery. *Sensors* **2014**, *14*, 13437–13475.

## 1. Introduction

Pattern recognition deals with the detection and identification of a desired pattern or target in an unknown input scene, which may or may not contain the target, and the determination of the spatial location of any target present. In pattern recognition or classification, the input is an image while the output is a decision signal based on some characteristic features of the input image. The number of features is usually fewer than the total necessary to describe the complete target of interest, and this leads to a loss of information. Because of the crucial role of decision making required in pattern recognition, it is fundamentally an information reduction process, whereby, it is not possible to reconstruct the pattern but it is possible to give a precise decision [1–3].

Although a great deal of effort has been expended on detecting objects in visual images, only limited amount of work has been reported on the detection and tracking of targets in infrared images. In general, existing methods on infrared images work for a limited number of situations due to various practical constraints. An infrared sensor detects infrared radiation and converts it to an image by converting the temperature difference between an object and the surrounding background. The temperature scale is converted into a color scale or a gray scale on a display and in this way an image is obtained. This type of sensor or camera can image an object through smoke in a burning house, heat leaking from a house or objects in the absence of any reflected light (at night) [4]. The images captured by infrared sensors are becoming an integral part of the ongoing research on automatic target recognition (ATR).

Forward-looking infrared (FLIR) images are frequently used in ATR applications. The detection and discrimination of targets in infrared imagery has been a challenging problem due to low signal-to-noise ratio (SNR) and the variability of target and clutter signatures. The FLIR sequences, tested in this work, are recorded from a moving platform and include independently moving objects under various distortions and background variations. Thus, sensor ego-motion and object motions



induce coupled motions into the FLIR images, which make the detection and tracking of the objects extremely complicated. To detect independently moving objects in FLIR image sequences, the sensor properties must also be taken into account.

Real life FLIR imagery demonstrates a number of well-known challenges such as significantly high level of variability of target thermal signatures, size/aspect, locations within the scene; large number of target classes; lack of prior information; obscured targets; competing cluttered background scenery; different geographic, meteorological and weather conditions; time of the day; high ego-motion; sensor noise; and variations caused by translation, rotation, and scaling of the targets. Furthermore, inconsistencies in the signature of targets, similarities between the signatures of different targets, limited training and testing data, camouflaged targets, non-repeatability of target signatures, and difficulty in exploiting contextual information make the recognition problem even more challenging in target detection in FLIR imagery. In the case of FLIR images, additional challenges are caused due to the following important differences [5–7] with visual sequences:

- The thermal images are obtained by sensing the radiation in the infrared spectrum, which is either emitted or reflected by the object in the scene. Due to this property the images obtained from an infrared sensor have extremely low SNR, which results in limited information for performing detection or tracking task.
- FLIR imagery smoothes out object edges and corners leading to a reduction of distinct features.
- The generation and maintenance of kinetic energy usually heats up a moving object (e.g., friction, engine combustion). Consequently, moving objects often appear brighter than the background.
- FLIR images are noisy and have less contrast. Moreover, they often contain dirt on the lens, or local sensor failure at certain pixel locations.
- FLIR sequences are not easily available (especially not from controlled experiments) and have a lower resolution. The sequences available to us are  $128 \times 128$  pixels as compared to the  $512 \times 512$  pixels, and more, of standard visual cameras.
- FLIR sequences are often under difficult circumstances and may have abrupt discontinuities in motion.

Due to the limitations and difficulties of FLIR imagery, they demand more robust techniques than visual sequences. This paper presents some widely used pattern recognition and target tracking techniques adopted for FLIR imagery. In this paper, we discuss several target detection and tracking algorithms which are based on the recently reported matched filter-based correlation techniques such as the MACH, EMACH, DCCF, and PDCCF filters. The performance of these algorithms was tested using real life FLIR image sequences supplied by the Army Missile Command.

## 2. Matched Filter-Based Correlation

The matched filter-based correlator was first introduced in 1964 [8]. In this technique, the input signal  $f(\sigma, \epsilon)$  is Fourier transformed to yield

$$F(o, \nu) = \mathfrak{F} [f(\sigma, \epsilon)] \quad (1)$$

where  $\mathfrak{F}$  represents the Fourier transform operation,  $\sigma$  and  $\varepsilon$  represents the spatial domain variables, and  $\sigma$  and  $\nu$  represents the frequency domain variables, respectively. The correlation output is obtained by inverse Fourier transform operation given by

$$g(\sigma, \varepsilon) = \mathfrak{F}^{-1}[F(o, \nu)H(o, \nu)] \quad (2)$$

where,  $\mathfrak{F}^{-1}$  represents 2D inverse Fourier transform operation.

If the input object is moved laterally in the input plane, the Fourier transform remains fixed in space but is multiplied by a phase factor that depends on the lateral movement. Therefore, the coordinates of the bright correlation output is proportional to the coordinates of the signal  $f(\sigma, \varepsilon)$  located at the input plane. The intensity of the bright correlation spot is proportional to the degree to which the input and the filter functions are matched. This is also valid for multiple objects present at different locations of the input plane. This correlation system provides a great deal of sensitivity since it is both phase matched and amplitude matched [9–11].

For the complex matched filter (CMF), the frequency plane filter function is expressed by

$$H_{\text{cmf}} = R^*(o, \nu) = |R(o, \nu)|\exp[-j\phi(o, \nu)] \quad (3)$$

where,  $|R(o, \nu)|$  is the amplitude and  $\phi(o, \nu)$  is the phase factor of the Fourier spectrum of the reference function  $r(\sigma, \varepsilon)$ . When the input is similar to  $r(\sigma, \varepsilon)$ , the phase variation is canceled at the Fourier plane, thus producing a plane wave of light. The correlation peak corresponding to CMF is not very sharp due to the squaring of the magnitude of  $r(\sigma, \varepsilon)$ . Consequently, the resulting diffraction efficiency of a CMF is very poor. This filter is also unacceptably sensitive to even small changes in the reference signal or image. Currently available spatial light modulators (SLMs) cannot accommodate the full complex frequency response needed by CMFs.

### 3. Phase Only Filter (POF)

The optimum case with respect to light efficiency for a matched filter is realized by a phase only filter (POF) structure [10]. This filter is obtained by omitting the amplitude information and is defined as

$$H_{\text{pof}}(o, \nu) = \frac{R^*(o, \nu)}{|R(o, \nu)|} = \exp[-j\phi(o, \nu)] \quad (4)$$

Besides the improvement in light efficiency, the correlation peak intensity is also enhanced with a POF. However, although the autocorrelation peak intensity is higher than that of a CMF, it is not as sharp as might be produced by an IF since the product  $H_{\text{pof}}(o, \nu)R(o, \nu)$  is not necessarily a constant.

### 4. Amplitude-Modulated Phase Only Filter (AMPOF)

Amplitude modulated phase only filter (AMPOF) [12] is given by

$$H_{\text{ampof}}(o, \nu) = \frac{A}{B + |R(o, \nu)|} \exp[-j\phi(o, \nu)] \quad (5)$$

where  $A$  and  $B$  are either constants or functions of  $o$  and  $v$ . The gain control factor  $A$  guarantees that the transmittance of the filter is less than unity. With the inclusion of  $B$ ; the pole problem is solved and at the same time it is possible to yield very high autocorrelation peak.

## 5. Synthetic Discriminant Functions (SDF)

The SDF-based correlation filters had shown robust performance for distortion tolerant pattern recognition [13–15]. Assume  $x_1(\sigma, \varepsilon)$ ,  $x_2(\sigma, \varepsilon)$ , ...,  $x_N(\sigma, \varepsilon)$  denote  $N$  training images representing possible distortions to a reference image  $x(\sigma, \varepsilon)$ . The 2D Fourier transform of  $x(\sigma, \varepsilon)$  may be expressed as

$$X(o, v) = \iint x(\sigma, \varepsilon) \exp[-j2\pi(o\sigma + v\varepsilon)] \times d\sigma d\varepsilon \quad (6)$$

A composite image  $h(\sigma, \varepsilon)$  is designed from the training images such that when the complex conjugate of its Fourier transform, denoted as  $H^*(o, v)$ , is correlated with input Fourier transform, a similar output is obtained for all  $N$  inputs,  $x_1(\sigma, \varepsilon)$ ,  $x_2(\sigma, \varepsilon)$ , ...,  $x_N(\sigma, \varepsilon)$ . This type of correlator is known as frequency plane correlator. For this filter, the resulting spatial domain correlation output  $c(\tau_\sigma, \tau_\varepsilon)$  may be expressed as [13]

$$\begin{aligned} c(\tau_\sigma, \tau_\varepsilon) &= \iint F(o, v) H^*(o, v) \exp[j2\pi(o\tau_\sigma + v\tau_\varepsilon)] do dv \\ &= \iint h^*(\sigma, \varepsilon) f(\sigma + \tau_\sigma, \varepsilon + \tau_\varepsilon) d\sigma d\varepsilon \\ &= h(\sigma, \varepsilon) e f(\sigma, \varepsilon) \end{aligned} \quad (7)$$

where  $e$  denotes a two-dimensional cross-correlation operation.

In the Equal Correlation Peak SDF (ECP-SDF) design, the objective is to select a filter impulse response  $h(\sigma, \varepsilon)$  such that the resulting crosscorrelations with all the  $N$  training images are the same, which is impossible in practice. Hester and Casasent [13] introduced a technique that requires that only the values at the origin of these crosscorrelations should be the same as shown in the following equation,

$$\begin{aligned} h(\sigma, \varepsilon) e x_i(\sigma, \varepsilon) \Big|_{\tau_\sigma=0, \tau_\varepsilon=0} &= \iint h^*(\sigma, \varepsilon) x_i(\sigma, \varepsilon) d\sigma d\varepsilon \\ &= c, \quad i = 1, 2, \dots, N \end{aligned} \quad (8)$$

where  $c$  is a prespecified constant. Equation (8) shows that a  $h(\sigma, \varepsilon)$  would yield the same constant value  $c$  at the origin (location of the autocorrelation peak) for all  $N$  training images (*i.e.*,  $x_1(\sigma, \varepsilon)$  to  $x_N(\sigma, \varepsilon)$ ). When the input is a non-training image from the same class, the crosscorrelation output at the origin will be similar to this constant  $c$  and it can be recognized. The success of this approach depends on selecting a proper training set.

Assume that  $h(\sigma, \varepsilon)$  is a linear combination of the  $N$  training images, given by

$$h(\sigma, \varepsilon) = a_1 x_1(\sigma, \varepsilon) + \dots + a_N x_N(\sigma, \varepsilon) \quad (9)$$

where the coefficients  $a_1, a_2, \dots, a_N$  are determined in a way to satisfy the constraints of Equation (8). Substituting Equation (9) into Equation (8), we get

$$\sum_{i=1}^N a_i^* R_{ij} = c, \quad j = 1, 2, \dots, N, \quad (10)$$

where

$$R_{ij} = \iint x_i^*(\sigma, \varepsilon) x_j(\sigma, \varepsilon) d\sigma d\varepsilon \quad (11)$$

is the inner product, *i.e.*, the crosscorrelation at the origin of the training images  $x_i(\sigma, \varepsilon)$  and  $x_j(\sigma, \varepsilon)$ . If the training images are real, then there is no need for the conjugate operation shown in Equation (11). Equation (10) represents  $N$  complex linear equations with  $N$  complex unknowns,  $a_1, a_2, \dots, a_N$ , respectively. These equations can be solved by using any standard techniques, such as Gauss-Seidel Elimination method.

## 6. Modified Synthetic Discriminant Functions

The ECP-SDF is designed to produce a value  $c_i$  at the origin of the output plane when the  $i$ -th training image is used as the input. However, there are some practical problems in using this filter for pattern recognition applications [16–18]. Consequently, some improvements on ECP\_SDF are suggested which are introduced in the following subsections.

### 6.1. Generalized SDF

Assume that the training images  $x_1(\sigma, \varepsilon), x_2(\sigma, \varepsilon), \dots, x_M(\sigma, \varepsilon)$  are sampled to yield arrays  $x_1(m, n), x_2(m, n), \dots, x_M(m, n)$  each with  $\rho = \rho_1 \rho_2$  pixels, where  $\rho_1$  is the number of pixels in the vertical direction while  $\rho_2$  is the number of pixels in the horizontal direction of each image. It is also assumed that  $\rho$ -dimensional column vectors  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_M$  are obtained by placing the elements in these training images in vectors where the scanning direction is from left to right and from top to bottom. Similarly, the  $\rho$ -dimensional column vector  $\mathbf{h}$  is used to denote the composite image  $h(m, n)$ . Then the constraints in Equation (8) can be rewritten as

$$\mathbf{h}^+ \mathbf{x}_i = c_i, \quad i = 1, 2, 3, \dots, M \quad (12)$$

where the superscript  $+$  denotes the conjugate transpose operation. The data matrix  $\mathbf{X}$  is assumed to have the vector  $\mathbf{x}_i$  as its  $i$ -th column and is thus a  $\rho \times M$  matrix. It is also assumed that  $\rho \gg M$ , *i.e.*, the number of pixels in the training images is much larger than the number of training images, and that the columns of this matrix are linearly independent. Using this notation, Equation (12) can be rewritten as

$$\mathbf{X}^+ \mathbf{h} = \mathbf{c}^* \quad (13)$$

The ECP-SDF assumes that the composite image  $\mathbf{h}$  is of the following form

$$\mathbf{h} = \mathbf{X} \mathbf{a} \quad (14)$$

where  $\mathbf{a}$  is the vector of coefficients. Substituting Equation (14) into Equation (13) and solving for  $\mathbf{a}$  yields

$$\mathbf{a} = (\mathbf{X}^+ \mathbf{X})^{-1} \mathbf{c}^* \quad (15)$$

The filter vector  $\mathbf{h}$  can be obtained by substituting Equation (15) into Equation (14) to get

$$\mathbf{h}_{\text{ECP}} = \mathbf{X} (\mathbf{X}^+ \mathbf{X})^{-1} \mathbf{c}^* \quad (16)$$

A general expression for  $\mathbf{h}$  satisfying Equation (13) is given by

$$\mathbf{h}_{\text{GSDF}} = \mathbf{X}(\mathbf{X}^+\mathbf{X})^{-1}\mathbf{c}^* + [\mathbf{I}_d - \mathbf{X}(\mathbf{X}^+\mathbf{X})^{-1}\mathbf{X}^+]\mathbf{z} \quad (17)$$

where  $\mathbf{I}_d$  is the  $\rho \times \rho$  diagonal identity matrix and  $\mathbf{z}$  is any column vector with  $\rho$  complex entries. The ECP-SDF is obtained from Equation (17) when  $\mathbf{z} = 0$ . The filter vector in Equation (17) is known as the generalized SDF [17].

### 6.2. Minimum Variance SDF

Consider a situation where the input image is one of the training images  $\mathbf{x}_i$  corrupted by additive noise  $\mathbf{n}$ . Then the resulting output value  $y$ , *i.e.*, the value of the crosscorrelation at the origin is given by

$$\begin{aligned} y &= \mathbf{h}^+(\mathbf{x}_i + \mathbf{n}) \\ &= c_i + \mathbf{h}^+\mathbf{n} \end{aligned} \quad (18)$$

where  $\mathbf{h}$  is designed to satisfy Equation (12). From Equation (18) it is evident that the output  $y$  is the desired output  $c_i$  corrupted by the random variable  $(\mathbf{h}^+\mathbf{n})$ . The minimum variance synthetic discriminant function (MVSDF) [18] attempts to design  $\mathbf{h}$  such that the variance in the output caused by input noise is minimized while satisfying the constraints in Equation (13).

Assume that the real noise vector  $\mathbf{n}$  is a zero-mean vector with a  $\rho \times \rho$  covariance matrix  $\Sigma$ . The variance of  $y$  corresponding to  $\mathbf{h}^+\mathbf{n}$  can be expressed as

$$\sigma_y^2 = E\{|\mathbf{h}^+\mathbf{n}|^2\} = E\{\mathbf{h}^+\mathbf{n}\mathbf{n}^+\mathbf{h}\} = \mathbf{h}^+\Sigma\mathbf{h} \quad (19)$$

It is desired that  $\sigma_y^2$  in Equation (19) is as small as possible, which will ensure that the output values are close to the constrained values even in the presence of noise. Minimizing  $\sigma_y^2$  in Equation (19) subject to the constraints in Equation (13) leads to the following MVSDF [18]

$$\mathbf{h}_{\text{MVSDF}} = \Sigma^{-1}\mathbf{X}(\mathbf{X}^+\Sigma^{-1}\mathbf{X})^{-1}\mathbf{c}^* \quad (20)$$

This MVSDF is indeed optimal from noise tolerance considerations. One difficulty in using this MVSDF is that often  $\Sigma$  is not known. Even when it is known, it is impossible to calculate its inversion. Another problem is that the MVSDF controls only one point (the origin) in the output-correlation plane. Thus, large sidelobes may be observed in the correlation output.

### 6.3. Frequency-Domain SDFs

It is often more convenient to design the filters in the frequency domain [16–18]. Assume  $f(\sigma, \varepsilon)$  is the image and  $H^*(o, v)$  is the complex filter function. Then the resulting correlation output  $c(\tau_\sigma, \tau_\varepsilon)$  at the origin is given by

$$\begin{aligned} c(0, 0) &= \iint H^*(o, v)F(o, v)dodv \\ &= \hat{\mathbf{h}}^*\hat{\mathbf{f}} \end{aligned} \quad (21)$$

where  $\hat{\cdot}$  indicates that the corresponding vector or matrix is obtained by sampling frequency domain functions and the superscript  $+$  indicates a conjugate transpose operation. Because  $c_i(0,0)$  is constrained to be  $c_i$ ,  $i = 1, 2, \dots, N$ , the constraints can be rewritten as

$$\hat{\mathbf{F}}^+ \hat{\mathbf{h}} = \mathbf{c}^* \quad (22)$$

where  $\hat{\mathbf{F}}$  is a matrix with  $N$  columns with the  $i$ -th column containing  $\hat{\mathbf{f}}_i$ . It is obvious that the Equations (13) and (22) are similar.

#### 6.4. Minimum Average Correlation Energy (MACE) Filter

The correlation filters discussed so far control only one point in the correlation plane. For good location accuracy and discrimination, it is necessary to design filters capable of producing sharp correlation peaks. One such filter is the minimum average correlation energy (MACE) filter [19]. Assume  $x_1(\sigma, \varepsilon)$ ,  $x_2(\sigma, \varepsilon)$ ,  $\dots$ ,  $x_N(\sigma, \varepsilon)$  denote the  $N$  training images and  $X_1(o, v)$ ,  $\dots$ ,  $X_N(o, v)$  denote their 2D Fourier transforms, respectively. If  $H^*(o, v)$  denotes the transmittance of the filter function, then the filter may be constructed to satisfy the following condition.

$$\iint X_i(o, v) H^*(o, v) d o d v = c_i, \quad i = 1, 2, \dots, N \quad (23)$$

In addition, the MACE filter minimizes the average correlation plane energy as shown below.

$$\begin{aligned} E_{\text{ave}} &= \frac{1}{N} \sum_{i=1}^N \iint |c_i(\tau_\sigma, \tau_\varepsilon)|^2 d \tau_\sigma d \tau_\varepsilon \\ &= \frac{1}{N} \sum_{i=1}^N \iint |X_i(o, v)|^2 |H(o, v)|^2 d o d v \end{aligned} \quad (24)$$

By minimizing  $E_{\text{ave}}$ , it is possible to keep the sidelobes in the correlation plane as low as possible. This is essentially an indirect attempt at reducing the problem of sidelobes. To carry out the minimization of  $E_{\text{ave}}$ , the usual vector notation is used. If  $\hat{\mathbf{x}}_i$  denote the  $\rho$ -dimensional complex column vector obtained by sampling  $X_i(o, v)$ , then the constraints in Equation (23) can be rewritten as

$$\hat{\mathbf{X}}^+ \hat{\mathbf{h}} = \mathbf{c}^* \quad (25)$$

where  $\hat{\mathbf{X}}$  is a  $\rho \times N$  matrix with  $\hat{\mathbf{x}}_i$  as its  $i$ -th column. The  $E_{\text{ave}}$  in Equation (24) can be expressed as

$$E_{\text{ave}} = \hat{\mathbf{h}}^+ \hat{\mathbf{D}} \hat{\mathbf{h}} \quad (26)$$

where  $\hat{\mathbf{D}}$  is a  $\rho \times \rho$  diagonal matrix. The entries along the diagonal are obtained by averaging  $|X_i(o, v)|^2$ ,  $i = 1, 2, \dots, N$ , and then scanning the average from left to right and from top to bottom.

Minimizing  $E_{\text{ave}}$  in Equation (26) subject to the constraints in Equation (25) leads to the following filter

$$\hat{\mathbf{h}}_{\text{MACE}} = \hat{\mathbf{D}}^{-1} \hat{\mathbf{X}} (\hat{\mathbf{X}}^+ \hat{\mathbf{D}}^{-1} \hat{\mathbf{X}})^{-1} \mathbf{c}^* \quad (27)$$

In many simulation studies, filters designed using this approach produced sharp correlation peaks. However, MACE filters appear to have two drawbacks. The first is that there is no noise tolerance built into these filters. The second is that these filters seem to be more sensitive to intra-class variations. Casasent *et al.* [20] proposed Gaussian MACE filters to reduce the sensitivity

of the MACE filters to intra-class variations. The idea behind Gaussian MACE filters is to reduce the sharpness of the resulting correlation peak and thus improve its noise tolerance. MACE filters appear to be the first set of composite filters that attempt to control the entire correlation plane.

### 6.5. Minimum Squared Error SDF (MSE-SDF) Filter

This SDF design approach yields better approximation of arbitrary output correlation shapes in the minimum squared error (MSE) sense over the MACE filter and this filter is termed as MSE-SDF [21]. Like MACE, this filter must satisfy the usual SDF constraint of Equation (25). Besides, in MSE-SDF, the filter function  $H(o, v)$  must make the correlation function  $c_i$  approximate a prespecified desired shape  $t_i$ ,  $i = 1, 2, \dots, N$ . One measure of how well  $c_i$  approximates  $t_i$  is the average squared error  $E$  defined as

$$\begin{aligned} E &= \frac{1}{N} \sum_{i=1}^N \iint |t_i(\tau_\sigma, \tau_\varepsilon) - c_i(\tau_\sigma, \tau_\varepsilon)|^2 d\tau_\sigma d\tau_\varepsilon \\ &= \frac{1}{N} \sum_{i=1}^N \iint |T_i(o, v) - X_i^*(o, v)H(o, v)|^2 do dv \end{aligned} \quad (28)$$

where  $T_i$  is the Fourier transform of  $t_i$ . If  $\hat{\mathbf{X}}_i^{\mathbf{D}}$  is obtained by converting the vectors  $\hat{\mathbf{x}}_i$  into diagonal matrices, then the average squared error of Equation (28) becomes

$$E = \left[ E_d - \hat{\mathbf{h}}^+ \hat{\mathbf{p}} - \hat{\mathbf{p}}^+ \hat{\mathbf{h}} + \mathbf{h}^+ \hat{\mathbf{M}} \hat{\mathbf{h}} \right] \quad (29)$$

where

$$E_d = \frac{1}{N} \sum_{i=1}^N (\mathbf{t}_i^+ \mathbf{t}_i) \quad (30)$$

$$\hat{\mathbf{p}} = \frac{1}{N} \sum_{i=1}^N (\mathbf{X}_i^{\mathbf{D}} \mathbf{t}_i) \quad (31)$$

$$\hat{\mathbf{M}} = \frac{1}{N} \sum_{i=1}^N (\mathbf{X}_i^{\mathbf{D}*} \mathbf{X}_i^{\mathbf{D}}) \quad (32)$$

Minimizing  $E$  in Equation (29) subject to the constraints in Equation (25) leads to the following MSE-SDF filter

$$\hat{\mathbf{h}}_{\text{MSE-SDF}} = \hat{\mathbf{M}}^{-1} \hat{\mathbf{p}} + \hat{\mathbf{M}}^{-1} \hat{\mathbf{X}} \left[ (\hat{\mathbf{F}}^+ \hat{\mathbf{M}}^{-1} \hat{\mathbf{F}})^{-1} \right] \left[ \hat{\mathbf{c}} - \hat{\mathbf{F}}^+ \hat{\mathbf{M}}^{-1} \hat{\mathbf{p}} \right] \quad (33)$$

The MSE-SDF filter allows approximating arbitrary correlation shapes rather than zero shape implied in MACE filter design. This explicit control has two benefits [21]. First, correlation shapes can be selected with the linear/nonlinear postprocessing in mind and second, those correlation shapes that lead to better filter design can be used instead of simply minimizing the average correlation energy.

## 7. Maximum Average Correlation Height (MACH) Filter

The primary objective of the correlation filters is to achieve distortion-tolerant recognition of objects in the presence of clutter. This problem is easier to solve for in-plane rotations and scale changes. However, the prevalent method for handling out-of-plane distortions is to use a training set of representative views of the object. Traditionally, in the design of SDF-type correlation filters, linear constraints are imposed on the training images to yield a known value at specific locations in the correlation plane. However, placing such constraints in the correlation plane satisfies conditions only at isolated points in the image space but does not explicitly control the filter's ability to generalize over the entire domain of the training images. Various filters exhibit different levels of distortion tolerance even with the same training set and constraints.

The MACH filter adopts a statistical approach for filter design [22,23]. In addition to yielding sharp peaks and being computationally simple, this filter offers improved distortion tolerance. The reason lies in the fact that training images are not treated as deterministic representations of the object but as samples of a class whose characteristic parameters should be used in encoding the filter.

It is assumed that the training set consists of  $N$  images, and that each image of size  $\rho_1 \times \rho_2$  contains  $\rho = \rho_1\rho_2$  pixels. The  $i$ -th training image for the target class is denoted by  $x_i(m,n)$  in the spatial domain, which is represented in the frequency domain by a  $\rho \times 1$  vector  $\mathbf{x}_i$ , obtained by lexicographically reordering its two-dimensional discrete Fourier transform,  $X_i(k,l)$ . The Fourier domain filter is denoted by the  $\rho \times 1$  vector  $\mathbf{h}$ . The two-dimensional filter  $H(k,l)$  is obtained by rearranging  $\mathbf{h}$  into a two-dimensional image. In this paper, matrices are denoted by uppercase bold-face and vectors by lowercase bold-face characters. The correlation of the  $i$ -th training image and the filter can be expressed in the frequency domain as

$$\mathbf{g}_i = \mathbf{X}_i \mathbf{h} \quad (34)$$

where  $\mathbf{X}_i$  is a  $\rho \times \rho$  diagonal matrix containing the elements of  $\mathbf{x}_i$ . Here,  $\mathbf{g}_i$  denotes the discrete Fourier transform of the  $i$ -th correlation output. The deviation in the shape of the correlation plane with respect to some ideal shape vector  $\mathbf{f}$  is quantified by the average squared error (ASE), defined as

$$ASE = \frac{1}{N} \sum_{i=1}^N (\mathbf{g}_i - \mathbf{f})^+ (\mathbf{g}_i - \mathbf{f}) \quad (35)$$

Thus, ASE is a measure of distortion with respect to reference shape  $\mathbf{f}$ , which can be chosen as desired.

In fact, the shape vector  $\mathbf{f}$  can be treated as a free parameter in the distortion minimization problem. In the design of MSE-SDF [21],  $\mathbf{f}$  is specified as Gaussian or ring-like shapes in order to sculpt the correlation surface into these forms. In MACH, the choice of  $\mathbf{f}$  is such that it causes least variation among the correlation planes and offers minimum ASE. To find the optimum shape  $\mathbf{f}_{\text{opt}}$ , the gradient of ASE with respect to  $\mathbf{f}$  is set to zero, given by

$$\nabla_{\mathbf{f}}(ASE) = \frac{2}{N} \sum_{i=1}^N (\mathbf{g}_i - \mathbf{f}) = 0 \quad (36)$$

or,



$$\mathbf{f}_{\text{opt}} = \frac{1}{N} \sum_{i=1}^N \mathbf{g}_i = \bar{\mathbf{g}} \quad (37)$$

where

$$\bar{\mathbf{g}} = \frac{1}{N} \sum_{i=1}^N \mathbf{X}_i \mathbf{h} = \mathbf{M} \mathbf{h} \quad (38)$$

is the average correlation plane and  $\mathbf{M} = (1/N) \sum_{i=1}^N \mathbf{X}_i$  is the average training image expressed as a diagonal matrix. Thus, among all possible reference shapes, the average correlation plane  $\bar{\mathbf{g}}$  offers the smallest possible ASE and the least distortion (in the squared error sense) among the correlation planes.

Substituting  $\mathbf{f} = \bar{\mathbf{g}}$  in the ASE expression, the average similarity measure (ASM) is obtained as

$$\begin{aligned} \text{ASM} &= \frac{1}{N} \sum_{i=1}^N (\mathbf{g}_i - \bar{\mathbf{g}})^+ (\mathbf{g}_i - \bar{\mathbf{g}}) \\ &= \frac{1}{N} \sum_{i=1}^N (\mathbf{X}_i \mathbf{h} - \mathbf{M} \mathbf{h})^+ (\mathbf{X}_i \mathbf{h} - \mathbf{M} \mathbf{h}) \\ &= \mathbf{h}^+ \left[ \sum_{i=1}^N (\mathbf{X}_i - \mathbf{M})^* (\mathbf{X}_i - \mathbf{M}) \right] \mathbf{h} \\ &= \mathbf{h}^+ \mathbf{S}_x \mathbf{h} \end{aligned} \quad (39)$$

where

$$\mathbf{S}_x = \sum_{i=1}^N (\mathbf{X}_i - \mathbf{M})^* (\mathbf{X}_i - \mathbf{M}) \quad (40)$$

is a diagonal matrix measuring the similarity of the training images to the class mean in the frequency domain. For example, if all training images are identical, then  $\mathbf{S}_x$  would be an all-zero matrix. From Parseval's theorem, it is easy to show that the average squared distance from the correlation planes to their mean is the same as that defined by Equation (39) in the frequency domain [22].

The ASM is one possible metric for distortion since it represents the average deviation of the correlation planes from the mean correlation shape,  $\bar{\mathbf{g}}$ . It is also a measure of the compactness of the class. If filter  $\mathbf{h}$  is viewed as a linear transform, then ASM measures the distances of the training images from the class center under this transform. Minimizing ASM, therefore, leads to a compact set of correlation planes that resemble each other and exhibit the least possible variations. The distortions of the object in the input plane are represented by the training images,  $\mathbf{x}_i$ . These distortions are reflected in the output as variations in the structure and shape of the corresponding correlation planes,  $\mathbf{g}_i$ , and are quantified by ASM. If the filter successfully reduces the distortions, then distorted input images should yield similar output planes, leading to a small value of ASM. Conversely, if ASM is minimum and it is well shaped by design, then all true-class correlation planes are expected to resemble  $\bar{\mathbf{g}}$  and to exhibit well-shaped structures.

The MACH filter relaxes the correlation peak constraints and maximizes the peak intensity of the average training image. The peak intensity of the average training image is  $|\bar{\mathbf{g}}(0,0)|^2$  expressed as

$$|\bar{g}(0,0)|^2 = \left| \frac{1}{N} \sum_{i=1}^N \mathbf{h}^+ \mathbf{x}_i \right|^2 = |\mathbf{h}^+ \mathbf{m}|^2 = \mathbf{h}^+ \mathbf{m} \mathbf{m}^+ \mathbf{h} \quad (41)$$

where  $\mathbf{m}$  is the Fourier transform of the average training image expressed as a vector.

Here, it is assumed without the loss of generality that the peak occurs at the origin of the correlation plane.

The smaller the value of ASM, the more invariant the response of the filter is. In other words, if ASM is small, then all true-class correlation planes are expected to resemble  $\bar{g}$ . Therefore, it is required by  $\mathbf{h}$  to produce high correlation peak with the mean image while making ASM small. In addition, it is also required to obtain some degree of noise tolerance to reduce the output noise variance (ONV). For additive input noise,  $\text{ONV} = \mathbf{h}^+ \mathbf{D} \mathbf{h}$ , where  $\mathbf{D}$  is the diagonal power spectral density matrix [23]. While practical noise may be multiplicative and more complicated than implied by simple additive noise, the simple additive noise model at least provides some robustness. The performance criterion used to optimize the MACH filter may be expressed as

$$\begin{aligned} J(\mathbf{h}) &= \frac{(\text{Average peak height})^2}{\text{ASM} + \text{ONV}} \\ &= \frac{|\bar{g}(0,0)|^2}{\text{ASM} + \text{ONV}} \\ &= \frac{|\mathbf{h}^+ \mathbf{m}|^2}{\mathbf{h}^+ \mathbf{S} \mathbf{h} + \mathbf{h}^+ \mathbf{D} \mathbf{h}} \\ &= \frac{\mathbf{h}^+ \mathbf{m} \mathbf{m}^+ \mathbf{h}}{\mathbf{h}^+ (\mathbf{S} + \mathbf{D}) \mathbf{h}} \end{aligned} \quad (42)$$

The optimum solution is found by setting the derivative of  $J(\mathbf{h})$  in Equation (42) with respect to  $\mathbf{h}$  to zero and is given by [22,23]

$$\mathbf{h} = (\mathbf{S} + \mathbf{D})^{-1} \mathbf{m} \quad (43)$$

The filter in Equation (43) is referred to as the MACH filter because it maximizes the height of the mean correlation peak relative to the expected distortions. For cases where an estimate of  $\mathbf{D}$  is not available, the white noise covariance matrix is substituted for  $\mathbf{D}$ , *i.e.*,  $\mathbf{D} = \sigma^2 \mathbf{I}$ , where  $\mathbf{I}$  is a diagonal identity matrix. Hence the simplified MACH filter becomes

$$\mathbf{h} = (\mathbf{S} + \sigma^2 \mathbf{I})^{-1} \mathbf{m} \quad (44)$$

Replacing  $\sigma^2$  by another constant  $\gamma$ , the filter equation becomes

$$\mathbf{h} = (\mathbf{S} + \gamma \mathbf{I})^{-1} \mathbf{m} \quad (45)$$

The robustness of the MACH filter is attributed to the inclusion of the ASM criterion, which reduces the filter sensitivity to distortions, and to the removal of hard constraints on the peak. The later fact enables the correlation planes to adjust to suitable values for optimizing the performance criterion. However, MACH filter can handle those distortions that are well represented in the training set.

The Fourier domain MACH filter obtained in Equation (45) may be converted to the 2D shape or size of the input training images expressed as  $H(k,l)$ . A 2D test image  $z(m,n)$  is Fourier transformed to obtain  $Z(k,l)$  which is then correlated with the 2D filter in the Fourier domain using the expression

$$G(k,l) = Z(k,l)H^*(k,l) \quad (46)$$

The output spatial domain correlation is obtained by applying the inverse Fourier transform operation to Equation (46) and recording the intensity, given by

$$g(m,n) = \left| \mathcal{F}^{-1} [G(k,l)] \right|^2 \quad (47)$$

## 8. Extended MACH (EMACH) Filter

The average training image used in the MACH filter design is good in representing the average behavior of the desired class, but it fails to capture the finer details of the desired class [24,25]. In fact, the average of training images sometimes looks like a clutter image. Thus, the MACH filter may be inadequate in discriminating the desired class from the clutter, leading to increased false alarm rate. The extended MACH (EMACH) filter is aimed at improving this clutter rejection capability.

The MACH filter is designed to maximize the intensity of the average correlation output at the origin due to training images. The average of correlation peaks is the correlation output due to the average training image. It also maximizes the similarity between the average training image correlation output and those outputs due to all training images from the desired class. Thus, the MACH filter forces all images from the desired class to follow the behavior of the average training image from that class. The MACH filter relies heavily on the mean training image. It amplifies the high-energy (usually low-frequency) components, and at the same time, attenuates the low-energy (usually high-frequency) components of the training set. Thus, by using the mean image  $\mathbf{m}$  as the only example that represents all training images, a filter may be obtained that does not capture the finer details of the training images. Therefore, this filter may fail to discriminate the desired class from the clutter. The MACH filters possess attributes that may lead to detect clutter images as targets. One such attribute is that all training images follow the same behavior as the average training image. However, the average training image is not necessarily a good representative of the desired class.

To control the relative contribution of the desired class training images as well as their average, a new metric, called all image correlation height (AICH) [25], is introduced and defined as

$$\begin{aligned} \text{AICH} &= \frac{1}{N} \sum_{i=1}^N \left[ \mathbf{h}^+ (\mathbf{x}_i - \beta \mathbf{m}) \right]^2 \\ &= \frac{1}{N} \sum_{i=1}^N \left( \mathbf{h}^+ \mathbf{x}_i - \beta \mathbf{h}^+ \mathbf{m} \right)^2 \end{aligned} \quad (48)$$

where  $\beta$  is a parameter that takes a value between 0 and 1 and governs the relative significance of the average training image in the filter design. By controlling  $\beta$ , the designed filter is prevented from being overwhelmed by the biased treatment of the low-frequency components represented by the average image. Here, the AICH must be optimized and to be able to do that, Equation (48) may be rewritten as

$$\begin{aligned}
\text{AICH} &= \frac{1}{N} \sum_{i=1}^N (\mathbf{h}^+ \mathbf{x}_i - \beta \mathbf{h}^+ \mathbf{m}) (\mathbf{h}^+ \mathbf{x}_i - \beta \mathbf{h}^+ \mathbf{m})^+ \\
&= \mathbf{h}^+ \left[ \frac{1}{N} \sum_{i=1}^N (\mathbf{x}_i - \beta \mathbf{m}) (\mathbf{x}_i - \beta \mathbf{m})^+ \right] \mathbf{h} \\
&= \mathbf{h}^+ \mathbf{C}_x^\beta \mathbf{h}
\end{aligned} \tag{49}$$

where

$$\mathbf{C}_x^\beta = \frac{1}{N} \sum_{i=1}^N (\mathbf{x}_i - \beta \mathbf{m}) (\mathbf{x}_i - \beta \mathbf{m})^+ \tag{50}$$

Thus, AICH can be described as the average of the correlation peak intensities of  $N$  exemplars where the  $i$ -th exemplar  $(\mathbf{x}_i - \beta \mathbf{m})$  is the  $i$ -th training image with part of the mean subtracted. Hence, it is desirable for all images in the training set to follow these exemplars' behavior. This can be done by forcing every image in the training set  $\mathbf{x}_i$  to have a similar correlation output plane to an ideal correlation output shape  $\mathbf{f}$ . To find the  $\mathbf{f}$  that best matches all these exemplars' correlation output planes, its deviation from their correlation planes is minimized. This deviation can be quantified by ASE, defined as

$$\text{ASE} = \frac{1}{N} \sum_{i=1}^N (\mathbf{g}_i - \mathbf{f})^+ (\mathbf{g}_i - \mathbf{f}) \tag{51}$$

where,

$$\mathbf{g}_i = (\mathbf{X}_i - \beta \mathbf{M}) \mathbf{h}^* \tag{52}$$

In Equation (52), the superscript  $*$  represents the complex conjugate operation. To find the optimum shape vector  $\mathbf{f}_{\text{opt}}$ , the gradient of ASE with respect to  $\mathbf{f}$  is set to zero, yielding

$$\begin{aligned}
\mathbf{f}_{\text{opt}} &= \frac{1}{N} \mathbf{g}_i \\
&= \frac{1}{N} \sum_{i=1}^N (\mathbf{X}_i - \beta \mathbf{M}) \mathbf{h}^* \\
&= (1 - \beta) \mathbf{M} \mathbf{h}^*
\end{aligned} \tag{53}$$

The ASM is modified such that it measures the dissimilarity of the training images to  $(1 - \beta) \mathbf{M} \mathbf{h}^*$ . This new measure is called as the modified ASM (MASM), given by

$$\begin{aligned}
\text{MASM} &= \frac{1}{N} \sum_{i=1}^N [\mathbf{X}_i \mathbf{h}^* - (1 - \beta) \mathbf{M} \mathbf{h}^*]^+ [\mathbf{X}_i \mathbf{h}^* - (1 - \beta) \mathbf{M} \mathbf{h}^*] \\
&= \mathbf{h}' \left\{ \frac{1}{N} \sum_{i=1}^N [\mathbf{X}_i - (1 - \beta) \mathbf{M}]^* [\mathbf{X}_i - (1 - \beta) \mathbf{M}] \right\} \mathbf{h}^* \\
&= \mathbf{h}' \mathbf{S}_x^\beta \mathbf{h}^* \\
&= \mathbf{h}^+ \mathbf{S}_x^\beta \mathbf{h}
\end{aligned} \tag{54}$$

where the superscript  $'$  represents the transpose operation and where it is considered that MASM is real in deriving the last equality in Equation (54). The diagonal matrix  $\mathbf{S}_x^\beta$  is given by

$$\mathbf{S}_x^\beta = \frac{1}{N} \sum_{i=1}^N [\mathbf{X}_i - (1-\beta)\mathbf{M}]^* [\mathbf{X}_i - (1-\beta)\mathbf{M}] \quad (55)$$

The ASM is a good measure for distortion tolerance; however, it lacks some discrimination capability that explains part of the MACH filter's inability to reject some clutter images. On the other hand, the MASM measure captures finer details of the training set that makes the EMACH filter more sensitive against clutter.

By maximizing the AICH and minimizing the MASM while controlling the parameter  $\beta$ , it is expected to explicitly keep a balance between the distortion tolerance and clutter rejection performance. Therefore, it is necessary to optimize the following new criterion

$$J^\beta(\mathbf{h}) = \frac{\text{AICH}}{\mathbf{h}^+ \boldsymbol{\gamma} \mathbf{I} \mathbf{h} + \mathbf{h}^+ \mathbf{S}_x^\beta \mathbf{h}} = \frac{\mathbf{h}^+ \mathbf{C}_x^\beta \mathbf{h}}{\mathbf{h}^+ (\boldsymbol{\gamma} \mathbf{I} + \mathbf{S}_x^\beta) \mathbf{h}} \quad (56)$$

where  $\mathbf{h}^+ \boldsymbol{\gamma} \mathbf{I} \mathbf{h}$  is the ONV term assuming an additive white noise with variance  $\gamma$ . The ONV helps to maintain noise tolerance when  $\beta$  increases, especially, at those low energy components. By maximizing the preceding criterion, the following condition is obtained for the EMACH filter

$$(\boldsymbol{\gamma} \mathbf{I} + \mathbf{S}_x^\beta)^{-1} \mathbf{C}_x^\beta \mathbf{h} = \lambda \mathbf{h} \quad (57)$$

where  $\lambda$  is a scalar identical to  $J^\beta(\mathbf{h})$ . Thus,  $\mathbf{h}$  must be an eigenvector of  $(\boldsymbol{\gamma} \mathbf{I} + \mathbf{S}_x^\beta)^{-1} \mathbf{C}_x^\beta$  with the corresponding eigenvalue  $\lambda$ . Since  $\lambda$  is identical to  $J^\beta(\mathbf{h})$ ,  $\mathbf{h}$  should be the eigenvector that corresponds to the maximum eigenvalue. The other eigenvectors corresponding to the other nonzero eigenvalues provide smaller  $J^\beta(\mathbf{h})$  values. However, they may provide better discriminatory performance, as  $\beta$  is not known *a priori*. So the EMACH filter may be expressed as

$$\mathbf{h} = \text{Dominant eigenvector} \{ (\boldsymbol{\gamma} \mathbf{I} + \mathbf{S}_x^\beta)^{-1} \mathbf{C}_x^\beta \} \quad (58)$$

## 9. Distance Classifier Correlation Filter (DCCF)

This is a correlation-based distance classifier scheme for recognition and classification of multiple similar or dissimilar objects. The underlying theory uses shift-invariant filters to compute distances between the input image and ideal references under an optimum transformation. The two ideas of relaxing the constraints on the correlation values at the origin and looking at the entire correlation plane rather than just the peak value led to the development of distance classifier correlation filters DCCFs [26–28]. The DCCF formulation can be used with any number of classes.

In the DCCF design, a global transformation is determined such that the transformed images from the same class are close to each other, whereas transformed images from different classes are separated from each other. This global transform leads to one correlation filter for each class. The use of these correlation filters is similar to the use of other correlation filters except in the final step. The test image is correlated with the correlation filter, and the resulting correlation peak is determined. This correlation-peak value is used to determine the distance of the test image to this class. Distances of the test image to all classes are determined and the class yielding the smallest distance

is chosen. This paradigm allows for the relaxation of the correlation output constraints and the use of the entire correlation output.

It is important to realize that the use of DCCFs is similar to the use of other correlation filters. This means that the correlation peaks move by the same amount corresponding to the shift in the input, *i.e.*, this is a shift-invariant operation. The DCCF concept has demonstrated promising performance on both infrared as well as synthetic aperture radar imagery [28]. Test results show that DCCFs outperform other correlation filters in recognizing targets while rejecting noise, clutter, and other confusing objects.

It is assumed that the training images are segmented and registered at a desired point. Fourier transform of an image  $x(m,n)$  of size  $\rho_1 \times \rho_2$  containing  $\rho = \rho_1\rho_2$  pixels can be expressed as a  $\rho \times 1$  dimensional column vector  $\mathbf{x}$  or as a  $\rho \times \rho$  diagonal matrix  $\mathbf{X}$  with the elements of  $\mathbf{x}$  as its diagonal elements, *i.e.*, diagonal  $\{\mathbf{x}\} = \mathbf{X}$ . Sometimes, the same quantity may be expressed both as a vector, say  $\mathbf{m}_x$ , and as a diagonal matrix  $\mathbf{M}_x$ . This implies that  $\mathbf{H}\mathbf{m}_x$  and  $\mathbf{M}_x\mathbf{h}$  are equivalent.

The distance classifier uses a global transform denoted by  $\mathbf{H}$  to separate the classes maximally while making them as compact as possible. For shift invariance, this transform matrix must be diagonal in the frequency domain. Multiplication of a vector  $\mathbf{x}$  by a diagonal matrix  $\mathbf{H}$  is equivalent to multiplying  $X(k,l)$  by  $H(k,l)$ .

Here, a general  $C$ -class distance-classifier problem is analyzed by assuming that the peak correlation values are as different as possible for each of the classes although hard constraints are not used to enforce this. In addition, for each class, the correlation planes or their inverse Fourier transforms should be almost similar to the transformed ideal reference shape for this class. The correlation peaks are most likely at the origin for the registered training images but can occur elsewhere in the test cases depending on the location of the target.

Assume  $\mathbf{x}_{ik}$  is the  $\rho$ -dimensional column vector containing the Fourier transform of the  $i$ -th image of the  $k$ -th class,  $1 \leq i \leq N$  and  $1 \leq k \leq C$ , where each class contains  $N$  training images. If  $\mathbf{m}_k$  is the mean Fourier transform of class  $k$ , then

$$\mathbf{m}_k = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_{ik}, \quad 1 \leq k \leq C \quad (59)$$

The correlation peak at the origin between the mean training image from the  $k$ -th class and the filter  $\mathbf{h}$  is given by  $\mathbf{m}_k^+\mathbf{h}$ . Thus, the overall mean Fourier transform of the entire training set becomes

$$\mathbf{m} = \frac{1}{C} \sum_{k=1}^C \mathbf{m}_k \quad (60)$$

The correlation peak of the overall mean image  $\mathbf{m}$  with the filter  $\mathbf{h}$  is given by

$$\mathbf{m}^+\mathbf{h} = \frac{1}{C} \sum_{k=1}^C \mathbf{m}_k^+\mathbf{h} \quad (61)$$

which represents the overall average of the origin values of all correlation planes.

If the transformation  $\mathbf{h}$  makes the in-class correlation planes similar, then the in-class peak values should be similar to each other and to their mean. Thus, to make the interclass separation between the correlation peaks large, the mean peak values of the classes are made as different as possible.

Although several possible criteria might achieve this objective, the approach here is to increase the distance of all classes from the central mean. Toward this end, the following distance measure, called class separation, has been formulated

$$\begin{aligned} A(\mathbf{h}) &= \frac{1}{C} \sum_{k=1}^C |\mathbf{m}_k^+ \mathbf{h} - \mathbf{m}^+ \mathbf{h}|^2 \\ &= \frac{1}{C} \sum_{k=1}^C \mathbf{h}^+ (\mathbf{m} - \mathbf{m}_k) (\mathbf{m} - \mathbf{m}_k)^+ \mathbf{h} \\ &= \mathbf{h}^+ \mathbf{W} \mathbf{h} \end{aligned} \quad (62)$$

where

$$\mathbf{W} = \frac{1}{C} \sum_{k=1}^C (\mathbf{m} - \mathbf{m}_k) (\mathbf{m} - \mathbf{m}_k)^+ \quad (63)$$

is a  $\rho \times \rho$ , full (*i.e.*, non-diagonal) matrix of rank less than or equal to  $(C-1)$ . The rank of  $\mathbf{W}$  is less than or equal to  $(C-1)$ , because it is obtained by the addition of  $C$  outer products of vectors  $(\mathbf{m} - \mathbf{m}_k)$ , but these  $C$  vectors add up to a zero vector. If  $A(\mathbf{h})$  of Equation (62) is maximized, the class mean correlation peaks  $(\mathbf{m}_k^+ \mathbf{h})$  will differ significantly. It is also desired that the distance of transformed inputs to their average be small. This distance  $B(\mathbf{h})$ , which measures the compactness of each class, is the same as ASM defined for each class as

$$\text{ASM}_k = \frac{1}{N} \sum_{i=1}^N |\mathbf{g}_{ik} - \bar{\mathbf{g}}_k|^2 \quad 1 \leq k \leq C \quad (64)$$

where

$$\begin{aligned} \mathbf{g}_{ik} &= \mathbf{X}_{ik} \mathbf{h}^* \\ \bar{\mathbf{g}}_k &= \mathbf{M}_k \mathbf{h}^* \end{aligned} \quad (65)$$

are the Fourier transforms of the correlation outputs due to the  $i$ -th training image  $\mathbf{x}_{ik}$  and the average training image  $\mathbf{m}_k$ , respectively from class  $k$ . Note that  $\mathbf{X}_{ik}$  and  $\mathbf{M}_k$  are diagonal matrices with  $\mathbf{x}_{ik}$  and  $\mathbf{m}_k$  along the diagonal. The ASM is a measure of the similarity of the training images of a class to their mean and hence a measure of the compactness of the class after transformation by  $\mathbf{H}$ . Using Equations (64) and (65),  $\text{ASM}_k$  can be rewritten as

$$\begin{aligned} \text{ASM}_k &= \frac{1}{N} \sum_{i=1}^N (\mathbf{X}_{ik} \mathbf{h}^* - \mathbf{M}_k \mathbf{h}^*)^+ (\mathbf{X}_{ik} \mathbf{h}^* - \mathbf{M}_k \mathbf{h}^*) \\ &= \mathbf{h}' \left[ \frac{1}{N} \sum_{i=1}^N (\mathbf{X}_{ik} - \mathbf{M}_k)^* (\mathbf{X}_{ik} - \mathbf{M}_k) \right] \mathbf{h}^* \\ &= \mathbf{h}^+ \mathbf{S}_k \mathbf{h} \end{aligned} \quad (66)$$

where

$$\mathbf{S}_k = \frac{1}{N} \sum_{i=1}^N (\mathbf{X}_{ik} - \mathbf{M}_k)^* (\mathbf{X}_{ik} - \mathbf{M}_k) \quad (67)$$

In Equation (67),  $\mathbf{S}_k$  is a  $\rho \times \rho$  diagonal matrix where each training image contains  $\rho$  pixels. The overall ASM for  $C$  classes is defined as

$$\text{ASM} = B(\mathbf{h}) = \frac{1}{N} \sum_{k=1}^C \mathbf{h}^+ \mathbf{S}_k \mathbf{h} = \mathbf{h}^+ \mathbf{S} \mathbf{h} \quad (68)$$

where

$$\mathbf{S} = \frac{1}{C} \sum_{k=1}^C \mathbf{S}_k \quad (69)$$

To make the in-class metric  $B(\mathbf{h})$  small and to make the inter-class distance metric  $A(\mathbf{h})$  large, the filter  $\mathbf{h}$  is designed to maximize the ratio

$$J(\mathbf{h}) = \frac{A(\mathbf{h})}{B(\mathbf{h})} = \frac{\mathbf{h}^+ \mathbf{W} \mathbf{h}}{\mathbf{h}^+ \mathbf{S} \mathbf{h}} \quad (70)$$

with respect to  $\mathbf{h}$ . The filter  $\mathbf{h}$  that maximizes  $J(\mathbf{h})$  in Equation (70) is the eigenvector of  $\mathbf{S}^{-1} \mathbf{W}$  with the largest eigenvalue [28]. Because  $\mathbf{W}$  is a non-diagonal matrix of rank less than or equal to  $(C-1)$ , finding the dominant eigenvector of  $\mathbf{S}^{-1} \mathbf{W}$  requires a special algorithm when the training images, and thus the desired filter is of larger size [28]. When  $J(\mathbf{h})$  is maximum, the correlation shape produced by an input image is expected to be similar to the mean shape for its true class with a peak value different from the average peak value of any other class. The DCCF filter may be expressed as

$$\mathbf{h} = \text{Dominant eigenvector}\{\mathbf{S}^{-1} \mathbf{W}\} \quad (71)$$

The DCCF is the first technique proposed for shift-invariant transform-domain distance calculations with a correlator and that are specifically designed to accommodate multiple targets at different locations in the same image. This filter deals with the entire correlation plane and not just one point at the origin. It transforms the input image into a new space in which the distance of a test input from the classes is computed. Given a test input  $\mathbf{z}$ , the distances  $d_k$  between the transformed input and the ideal shape for class  $k$  is computed as

$$\begin{aligned} d_k &= \left| \mathbf{H}^* \mathbf{z} - \mathbf{H}^* \mathbf{m}_k \right|^2 \\ &= \left| \mathbf{H}^* \mathbf{z} \right|^2 + \left| \mathbf{H}^* \mathbf{m}_k \right|^2 - 2\Re\{\mathbf{z}^+ \mathbf{H} \mathbf{H}^* \mathbf{m}_k\} \\ &= p + b_k - 2\Re\{\mathbf{z}^+ \mathbf{h}_k\} \end{aligned} \quad (72)$$

In Equation (72),  $p = \left| \mathbf{H}^* \mathbf{z} \right|^2$  is the energy (independent of the class) of the transformed input test image  $\mathbf{z}$ ;  $b_k = \left| \mathbf{H}^* \mathbf{m}_k \right|^2$  is the energy (independent of the  $\mathbf{z}$ ) of the transformed mean of class  $k$  and  $\mathbf{h}_k = \mathbf{H} \mathbf{H}^* \mathbf{m}_k$  is viewed as the effective filter for class  $k$ . Because there are only  $C$  classes for which distances must be computed, only  $C$  such filters are required.

In general, the targets may be anywhere in the input image. For shift-invariant distance calculation the interest is in the smallest value of  $d_k$  over all possible shifts of the target with respect to the class references. In Equation (72), because  $p$  and  $b_k$  are both positive and independent of the



position of the target, the smallest value of  $d_k$  over all shifts is obtained when the third term (*i.e.*,  $\mathbf{z}^+\mathbf{h}_k$ ) is as large as possible. Therefore, this term is chosen as the peak value for full cross correlation of  $\mathbf{z}$  and  $\mathbf{h}_k$ .

## 10. Polynomial DCCF (PDCCF)

Linear transformations such as the DCCF are attractive because of their optimality when the underlying statistics are Gaussian with equal covariances [29]. However, the DCCF uses transformations based on the second order statistics and does not capture higher-order statistics in images. Hence, it does not necessarily capture all of the discrimination information in some cases. Examples of such cases are encountered when signal dependent or multiplicative noises are present, and when inputs have non-Gaussian statistics. In non-Gaussian statistics cases, DCCF capabilities may be improved by applying different nonlinearities to the input image. By using nonlinear transformations, it may be possible to extract more useful information for discrimination. Thus, the classes that are not well separated in the original image space may become more separated in the nonlinearly mapped space. Also, point nonlinearities are preferred because they reduce the computational complexity because a simple nonlinearity is being applied to each point without considering all points in the neighborhood.

The polynomial DCCF (DCCF) extends the DCCF to include point nonlinear mappings of the input patterns [29]. Examples of such nonlinear mappings correspond to powers of pixels of the input images. Even though the resulting PDCCF system is not linear with respect to the input patterns, it is still linear in the kernel. This property allows frequency domain techniques for the design, analysis, and implementation of this filter. Another important property is that this system works on different powers of input image pixels, which corresponds to a multi-dimensional correlation operation and thus extends the linear DCCF classification optimization criterion to a nonlinear one, and no nonlinear optimization is involved. Moreover, the PDCCF system provides a new framework for combining different correlation filters, where each filter in the system is optimized jointly with other filters.

The PDCCF first maps the input image  $x(m,n)$  into  $x^j(m,n)$  via point nonlinearities  $\eta_j$ , where  $j = 1, 2, \dots, n$ . Thus  $x^j(m,n)$  is related to  $x(m,n)$  through the following relationship

$$\eta_j : x(m,n) \rightarrow x^j(m,n) \quad (73)$$

In Equation (73), all  $x^j(m,n)$  are assumed to have the same size as the input  $x(m,n)$ . Examples of such mappings include various powers, logarithms, cosines, *etc.* The nonlinearly mapped input image  $x(m,n)$  is transformed by the filter  $h_j(m,n)$  built using  $x^j(m,n)$ . The overall distance is obtained by adding the distances resulting from the shift-invariant minimum mean squared error computations between every transformed image of the input and its respective ideal transformed reference. Those ideal transformed references are computed *a priori* by using the nonlinear functions  $\eta_1, \eta_2, \dots$ , and  $\eta_n$  followed by the application of the filters  $h_1(m,n), h_2(m,n), \dots$ , and  $h_n(m,n)$ , respectively.

In the linear DCCF, the filtered image  $g(m,n)$  resulting from transformation of  $x^1(m,n)$  by  $h_1(m,n)$  can be written as

$$g(m,n) = h_1(m,n) e^{x^1(m,n)} \quad (74)$$

where,  $e$  denotes the crosscorrelation and  $x^1(m,n) = x(m,n)$ . By augmenting the input image with  $x^2(m,n)$ , another term can be added involving the crosscorrelation of the  $x^2(m,n)$  with  $h_2(m,n)$  to obtain an output transformed by the two-term PDCCF, as shown below.

$$g(m,n) = h_1(m,n) e^{x^1(m,n)} + h_2(m,n) e^{x^2(m,n)} \quad (75)$$

Thus, the PDCCF has more terms at its disposal with which it can achieve better discrimination. Clearly, these new nonlinear versions of inputs are completely dependent on the original inputs, and in that sense no new information is being created. However, the new representations enable the correlation filters to provide better recognition. By continuing to add more terms such as  $h_j(m,n) e^{x^j(m,n)}$ , the  $n$ -term PDCCF can be obtained as

$$g(m,n) = \sum_{j=1}^n h_j(m,n) e^{x^j(m,n)} \quad (76)$$

If the focus is on the point-wise power nonlinearities for all  $\eta_j$ 's the nonlinear mapping of the input,  $x^j(m,n)$  can be defined as

$$x^j(m,n) = [x(m,n)]^j \quad (77)$$

where,  $j \in 1 \dots \infty$ . The power nonlinearity plays an important role in SAR images as it can enhance the bright scatters or the overall contrast [29]. The filters  $h_j(m,n)$  are computed jointly, and thus the advantages of the closed form solutions and nonlinear systems are combined and exploited. In the following analysis,  $\Psi$  represents the set of all powers used to construct a particular PDCCF. Although the nonlinearity is applied in the spatial domain, in the analyses to follow (e.g., filter formation, distance calculation, *etc.*) all the quantities are actually in Fourier domain. For example,  $\mathbf{x}$  is a vector obtained by lexicographical rearrangement of the Fourier transform of  $x(m,n)$ . All vectors and matrices with superscript  $j$  (e.g.,  $\mathbf{x}^j$  and  $\mathbf{X}^j$ ) represent these vectors and matrices ( $\mathbf{x}$  and  $\mathbf{X}$ ) with each of their elements in the image (spatial) domain raised to the  $j$ -th power.

Assume  $\mathbf{m}_k^j$  is the mean of class  $k$  of the Fourier transforms of training images resulting from raising all their pixels to the  $j$ -th power and  $\mathbf{h}_j$  is the filter built for images raised to the  $j$ -th power. Each of the Fourier transforms of the original image along with the Fourier transforms of its variations can be represented by use of a single block vector. Thus, the Fourier transform of the mean images of class  $k$  after augmentation becomes

$$\mathbf{m}_k = \begin{pmatrix} \mathbf{m}_k^1 \\ \mathbf{m}_k^2 \\ \cdot \\ \cdot \\ \mathbf{m}_k^n \end{pmatrix} \quad (78)$$

Further, the filters,  $\mathbf{h}_1, \mathbf{h}_2, \dots$  and  $\mathbf{h}_n$  can be combined into one filter,  $\mathbf{h}$  as follows

$$\mathbf{h} = \begin{pmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \cdot \\ \cdot \\ \mathbf{h}_n \end{pmatrix} \quad (79)$$

With Equations (75), (78), and (79), the correlation peak at the origin produced in response to the mean image of class  $k$  is given by

$$\bar{g}_k(0,0) = \sum_{j=1}^n \mathbf{h}_j^+ \mathbf{m}_k^j = \mathbf{h}^+ \mathbf{m}_k \quad (80)$$

The distance between the classes, after being augmented and then transformed by the filters  $\mathbf{h}_1$ ,  $\mathbf{h}_2$ , ..., and  $\mathbf{h}_n$  can be expressed as

$$\begin{aligned} A(\mathbf{h}) &= \frac{1}{C} \sum_{k=1}^C \left| \sum_{j=1}^n \mathbf{h}_j^+ \mathbf{m}_k^j - \sum_{j=1}^n \mathbf{h}_j^+ \mathbf{m}^j \right|^2 \\ &= \frac{1}{C} \sum_{k=1}^C \left| \mathbf{h}^+ \mathbf{m}_k - \mathbf{h}^+ \mathbf{m} \right|^2 \\ &= \mathbf{h}^+ \mathbf{W} \mathbf{h} \end{aligned} \quad (81)$$

where

$$\mathbf{W} = \frac{1}{C} \sum_{k=1}^C (\mathbf{m}_k - \mathbf{m})(\mathbf{m}_k - \mathbf{m})^+ \quad (82)$$

To separate the classes as much as possible, the filter is required to produce a large  $A(\mathbf{h})$ . Simultaneously, the compactness of the classes needs to be increased after transformation by  $\mathbf{h}_1$ ,  $\mathbf{h}_2$ , ..., and  $\mathbf{h}_n$ . The compactness is measured by the similarity of the training images of a class to their mean. It can be represented by the ASM of that class. In general, the ASM for class  $k$  is defined as

$$\text{ASM}_k = \frac{1}{N} \sum_{i=1}^N |\mathbf{g}_{ik} - \bar{\mathbf{g}}_k|^2 \quad 1 \leq k \leq C \quad (83)$$

where

$$\begin{aligned} \mathbf{g}_{ik} &= \mathbf{X}_{ik}^j \mathbf{h}_j^* \\ \bar{\mathbf{g}}_k &= \mathbf{M}_k^j \mathbf{h}_j^* \end{aligned} \quad (84)$$

are the Fourier transforms of the filtered images produced by the transform filters in response to the input image  $\mathbf{x}^j$  and the mean image  $\mathbf{m}_k$ , respectively. Thus, from Equations (83) and (84),  $\text{ASM}_k$  can be written as

$$\begin{aligned}
\text{ASM}_k &= \frac{1}{N} \sum_{i=1}^N \left| \mathbf{X}_{ik}^1 \mathbf{h}_1^* + \mathbf{X}_{ik}^2 \mathbf{h}_2^* + \dots + \mathbf{X}_{ik}^n \mathbf{h}_n^* - \mathbf{M}_k^1 \mathbf{h}_1^* - \mathbf{M}_k^2 \mathbf{h}_2^* - \dots - \mathbf{M}_k^n \mathbf{h}_n^* \right|^2 \\
&= \frac{1}{N} \sum_{i=1}^N \left[ (\mathbf{X}_{ik}^1 - \mathbf{M}_k^1) \mathbf{h}_1^* + (\mathbf{X}_{ik}^2 - \mathbf{M}_k^2) \mathbf{h}_2^* + \dots + (\mathbf{X}_{ik}^n - \mathbf{M}_k^n) \mathbf{h}_n^* \right]^+ \\
&\quad \left[ (\mathbf{X}_{ik}^1 - \mathbf{M}_k^1) \mathbf{h}_1^* + (\mathbf{X}_{ik}^2 - \mathbf{M}_k^2) \mathbf{h}_2^* + \dots + (\mathbf{X}_{ik}^n - \mathbf{M}_k^n) \mathbf{h}_n^* \right] \\
&= \sum_{u=1}^n \sum_{v=1}^n \mathbf{h}_u^+ \mathbf{S}_{kuv} \mathbf{h}_v \\
&= (\mathbf{h}_1^+ \mathbf{h}_2^+ \dots \mathbf{h}_n^+) \begin{bmatrix} \mathbf{S}_{k11} & \mathbf{S}_{k12} & \dots & \mathbf{S}_{k1n} \\ \mathbf{S}_{k21} & \mathbf{S}_{k22} & \dots & \mathbf{S}_{k2n} \\ \cdot & & & \cdot \\ \cdot & & & \cdot \\ \mathbf{S}_{kn1} & \mathbf{S}_{kn2} & \dots & \mathbf{S}_{knn} \end{bmatrix} \begin{pmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \cdot \\ \cdot \\ \mathbf{h}_n \end{pmatrix} \\
&= \mathbf{h}^+ \mathbf{S}_k \mathbf{h}
\end{aligned} \tag{85}$$

where

$$\mathbf{S}_{kuv} = \left[ \frac{1}{N} \sum_{i=1}^N \mathbf{X}_{ik}^u (\mathbf{X}_{ik}^v)^* \right] - \mathbf{M}_k^u (\mathbf{M}_k^v)^* \tag{86}$$

are all diagonal matrices and

$$\mathbf{S}_k = \begin{bmatrix} \mathbf{S}_{k11} & \mathbf{S}_{k12} & \dots & \mathbf{S}_{k1n} \\ \mathbf{S}_{k21} & \mathbf{S}_{k22} & \dots & \mathbf{S}_{k2n} \\ \cdot & & & \cdot \\ \cdot & & & \cdot \\ \mathbf{S}_{kn1} & \mathbf{S}_{kn2} & \dots & \mathbf{S}_{knn} \end{bmatrix} \tag{87}$$

The overall ASM for  $C$  classes is then defined as

$$\text{ASM} = B(\mathbf{h}) = \frac{1}{N} \sum_{k=1}^C \mathbf{h}^+ \mathbf{S}_k \mathbf{h} = \mathbf{h}^+ \mathbf{S} \mathbf{h} \tag{88}$$

where

$$\mathbf{S} = \frac{1}{C} \sum_{k=1}^C \mathbf{S}_k \tag{89}$$

The filter  $\mathbf{h}$  that maximizes the ratio  $A(\mathbf{h})/B(\mathbf{h})$ , is the dominant eigenvector of  $\mathbf{S}^{-1} \mathbf{W}$  as in case of DCCF and given by

$$\mathbf{h} = \text{Dominant eigenvector} \{ \mathbf{S}^{-1} \mathbf{W} \} \tag{90}$$

Given a test input  $\mathbf{z}$ , the distances  $d_k$  between the transformed input and the ideal shape for class  $k$  is computed by using MSE-then-total approach [30]. In this approach,  $n$  distances are computed for class  $k$ . The  $j$ -th distance,  $d_{jk}$ , is defined as follows

$$d_{jk} = \left| \mathbf{H}_j^* \mathbf{z}^j - \mathbf{H}_j^* \mathbf{m}_k^j \right|^2 \quad 1 \leq j \leq n \quad (91)$$

The distance  $d_{jk}$  can be rewritten as

$$d_{jk} = p_j + b_{jk} - 2\Re\{(\mathbf{z}^j)^+ \mathbf{h}_{jk}\} \quad (92)$$

where

$$p_j = \left| \mathbf{H}_j^* \mathbf{z}^j \right|^2 \quad (93)$$

$$b_{jk} = \left| \mathbf{H}_j^* \mathbf{m}_k^j \right|^2 \quad (94)$$

$$\mathbf{h}_{jk} = \mathbf{H}_j \mathbf{H}_j^* \mathbf{m}_k^j \quad (95)$$

The inner products, shown as the third terms in Equation (92), are between the  $j$ -th variation of the input,  $\mathbf{z}^j$  and the corresponding  $j$ -th filter,  $\mathbf{h}_{kj}$ . The total distance  $d_k$  to a class is then found by

$$d_k = \sum_{j=1}^n d_{jk} \quad (96)$$

The input image is assigned to the class with the least total distance.

## 11. Target Tracking in FLIR Imagery

In general, tracking of a moving pattern/target requires recognizing and then locating the target in a scene, finding the target motion, understanding the direction of motion of the target, and then following that target as it moves through the sequence of image frames. The detection and tracking of desired targets in a real life image corrupted by noise, clutter, illumination and other three-dimensional (3D) artifacts, poses a very complex problem and demands sophisticated solutions using pattern recognition and motion estimation methods [31,32]. Things become more complicated if there are more than one target in the scene and simultaneous multiple targets tracking is required.

Forward-looking infrared (FLIR) images are frequently used in automatic target recognition (ATR) applications. It is challenging to detect and track targets in FLIR imagery. To detect independent moving objects in FLIR image sequences, the sensor properties have to be taken into account. Additional challenges are caused due to many important differences of FLIR images with visual sequences [33,34]. Many researchers have investigated various approaches for detection, recognition, classification and pose estimation of targets from FLIR images including both matched spatial filter (MSF) based correlators and joint transform correlators [33–40]. However, the application of MSFs or their variants (e.g., MACH, DCCF) for the FLIR imagery is very limited; although those have been used for the simulated and real synthetic aperture radar (SAR) and laser radar (LADAR) imagery [24,41–47].

In this section, three different algorithms are demonstrated for pattern recognition and tracking based on the combination of the detection and classification filters [48–51]:

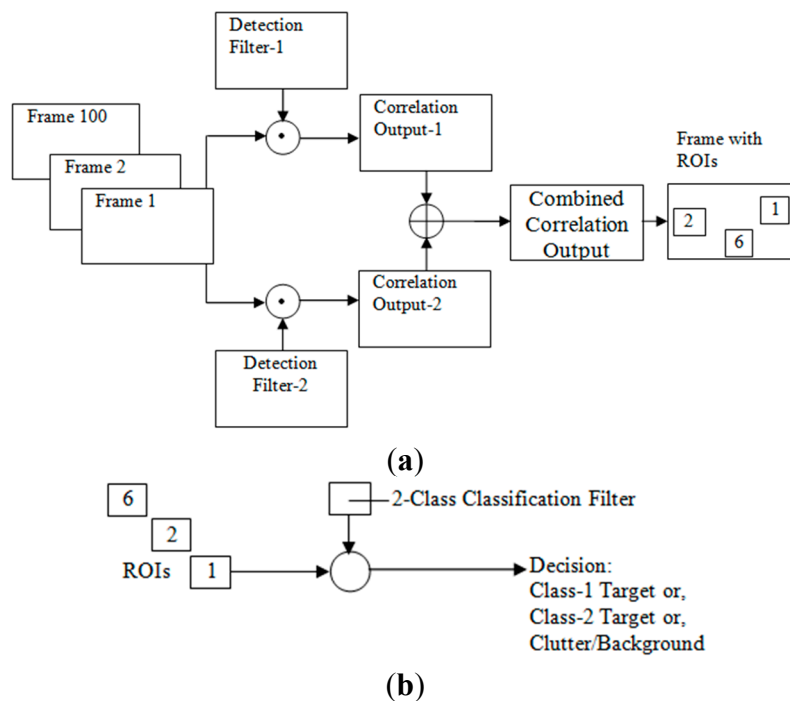
- MACH filter-based detection and DCCF-based classification (MACH-DCCF)
- MACH filter-based detection and PDCCF-based classification (MACH-PDCCF)

- EMACH filter-based detection and PDCCF-based classification (EMACH-PDCCF)

The detection filters are trained by the target images of expected size and orientation variations with expected size of input scene. The classification filters are formulated with the expected size of target images and trained by the target images of expected size and orientation variations. The first step of the real time system is detection, which involves correlating the input scene with all detection filters (one for each desired or expected target class) and combining the correlation outputs. In the second step, a predefined number of ROIs having the expected size of target images are selected based on the regions having higher correlation peak values in the combined correlation output. To ensure that all desired or expected targets are included in the ROIs, the number of ROIs should be at least three times higher than the number of expected targets. Classification filters are then applied to these ROIs and target types along with clutters are identified based on a distance measure and a threshold. Moving target detection and tracking are accomplished by following this technique for all incoming image frames by applying the same filters.

Multiple detection filters and classification filters are formulated for each target based on different size ranges or aspect angles. All of the filters for different ranges can be applied simultaneously through the whole range of the image sequence and decision can be made based on the output of the filter corresponding to the highest correlation peak or minimum distance. However, in this illustration, one detection filter and one classification filter is used for each class of targets for a particular range. A block diagram of the method for real time pattern recognition and tracking is shown in Figure 1 for a two-class detection system.

**Figure 1.** Schematic diagram of the proposed technique for (a) detection stage; and (b) classification stage.



### 11.1. Image Dataset

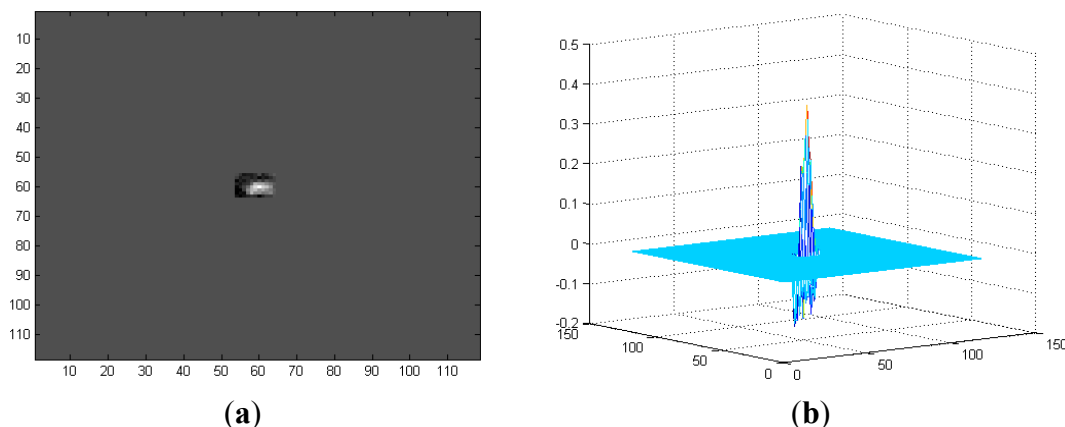
The FLIR image database used in this research is supplied by Army Missile Command (AMCOM). This image database has a total of 50 real life infrared video sequences, some of which contain a single target in the scene and some contain multiple targets. In general, the image sequences are closing sequences, *i.e.*, the targets become closer to the observer as the later frames appear in the scene. Thus, the size and signature of the targets change from the first frame to the last frame. Moreover, as the targets move, there are changes in the targets' orientations from one frame to the next. The database is also associated with ground truth data files containing the list of targets at each frame of each image sequence and their size and location in the frame. Among the 50 sequences, the proposed techniques have been applied for several single and multiple targets image sequences and the techniques are still being tested on other remaining sequences. For this paper, the analysis on 4 single target sequences (L1415, L2018, L2312, M1406), 2 two-target sequences (L1701, L1911) and 1 three-target sequence (L1618) have been reported.

### 11.2. Single-Target Image Sequences

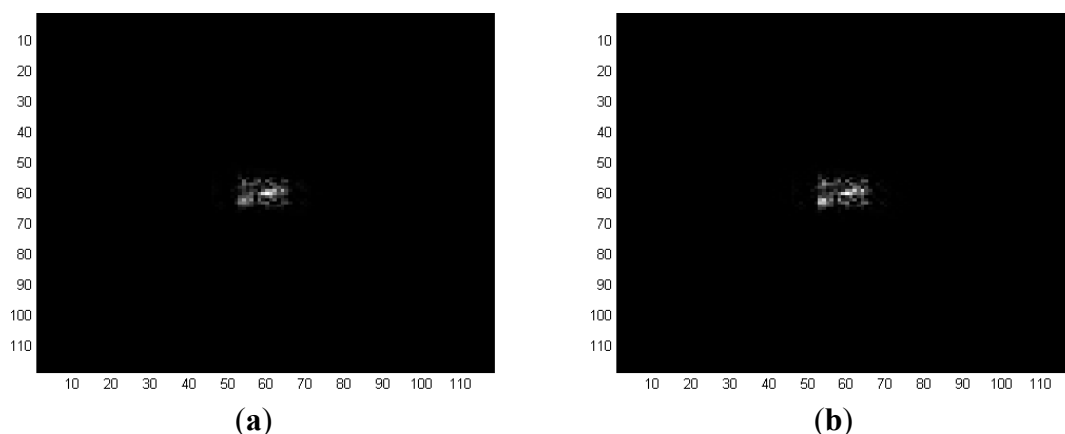
At first, consider Sequence L2018, which has the highest difficulty level among the selected single target image sequences. In this sequence, there are a total of 448 frames, each of which contains the same target (tank1). The size of this target in the first frame is  $3 \times 4$  pixels whereas that in the final frame is  $12 \times 23$  pixels. For effective detection and classification, proper selection of the training images is important. It is obvious that a single detection or classification filter obtained by exploiting the training images from this large variation of sizes may lose its selectivity. So, for this particular sequence, two different detection filters (MACH or EMACH) are formed for the target to use in two ranges of the image frames depending on the size of the target.

The first range detection filters are trained using the target patches taken from the first 200 frames at an interval of 5 frames (the 1st, 5th, 10th, 15th, ..., 200th frames). The ground truth data files are used to read the coordinates and sizes of the targets and then to segment out the target patches from the original frames. Each of these patches is then mean-subtracted and normalized dividing by the RMS value of the mean-subtracted patch. Thereafter, each patch is placed at the center of a  $118 \times 118$ -pixel zero padded matrix to form a full size training image for the detection filters. A sample training image of this type for Sequence L2018 is shown in Figure 2. The spatial domain representations of the MACH filter and EMACH filter for Range-1, trained by the above mentioned training images, are shown in Figure 3a,b, respectively.

**Figure 2.** (a) A full size ( $118 \times 118$ -pixel) training image for detection filters created from the target patch of Frame 100 of L2018; and (b) 3D mesh plot of (a).

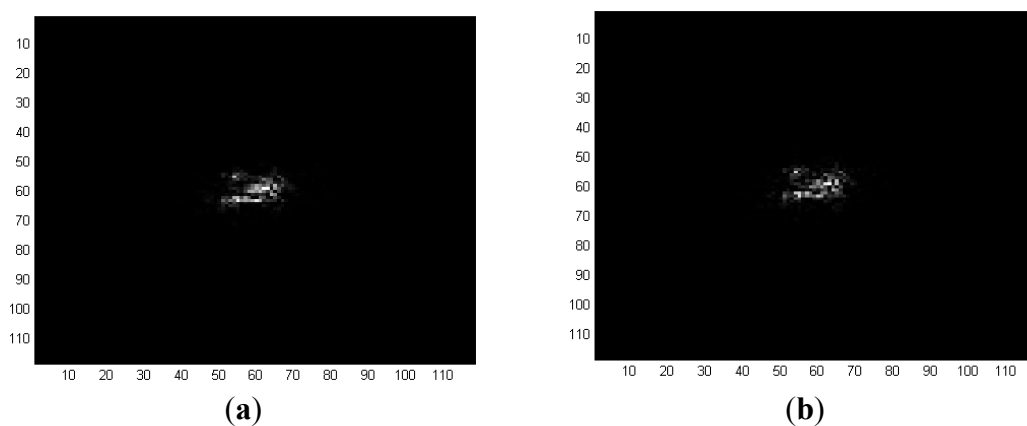


**Figure 3.** Spatial domain representation of Range-1 detection filters for the target (tank1) of Sequence L2018 (a) MACH filter; and (b) EMACH filter.



The detection filters for Range-2, to use from Frame 201 to Frame 448, are trained by the target patches taken from the Frames 201 to 300 at an interval of 5 frames. The corresponding Range-2 MACH and EMACH filters are shown in Figure 4a,b, respectively.

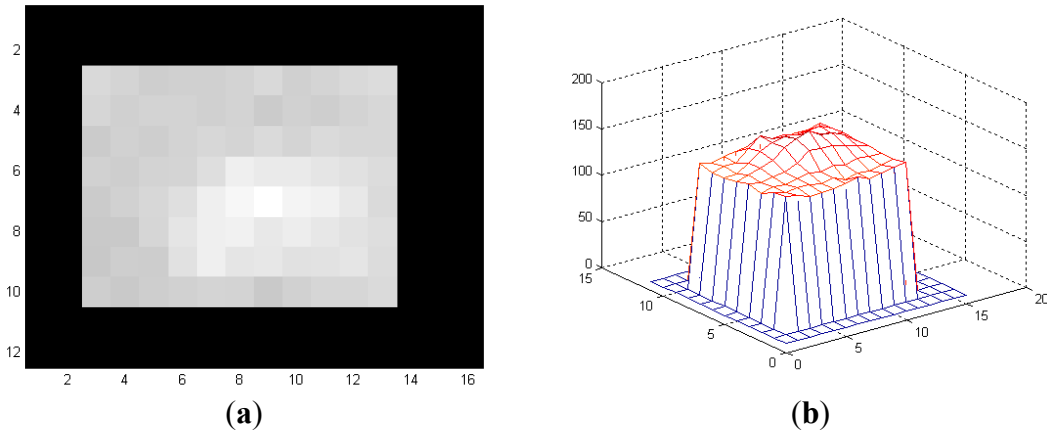
**Figure 4.** Spatial domain representation of Range-2 detection filters for the target (tank1) of Sequence L2018 (a) MACH filter; and (b) EMACH filter.





In general, the classification filters (DCCF or PDCCF) are also required to be formulated for different ranges of the targets for better selectivity. The sizes of these filters are usually chosen as the expected size of the targets in the corresponding range of image frames. Target patches are taken as before but they are not normalized for classification filters. Each patch is placed at the center of a zero padded background having the size of the classification filter. If the patch is larger than the filter size, it is truncated at the sides. For Sequence L2018, it is found that the classification filter (either DCCF or PDCCF) of  $12 \times 16$  pixels trained by the target patches of Frame 1 to Frame 200 at an interval of 4 frames works well for almost all frames. A  $12 \times 16$ -pixel sample training image of tank1 of Sequence L2018 for classification filter is shown in Figure 5.

**Figure 5.** (a) A  $12 \times 16$ -pixel training image for classification filters created from the target patch of Frame 100 of L2018; and (b) 3D mesh plot of (a).



It is assumed that the single target sequences are known to have only one target of interest. Hence, in this work, single class (1-class) classification filters (DCCF, PDCCF) are formulated for each particular sequence using the corresponding target patches. Although the classification filters actually need to be formulated using at least two classes of training images, in this work, 1-class filters are formulated by using slight modification in the basic formula. However, these 1-class approximated filters have been found to work well in most cases.

The training data or parameters for all the four filters (MACH, EMACH, DCCF, PDCCF) used for the three algorithms for detecting and tracking the target (tank1) in Sequence L2018 are summarized in Table 1. The first column of the table represents the name of the target along with the number of frames for which the ground truth data is available for it. The second column shows the filter names with their range indices; and the third column shows the range of the actual image frames where a particular filter is applied. The fourth and fifth columns provide the considered range of the frames and the order or interval of the frames used to take the target patch for training. In Table 1,  $\gamma$  is the filter parameter for MACH;  $\beta$  and  $\gamma$  are the filter parameters for EMACH and  $\Psi$  is the set of nonlinear powers used for PDCCF formulation.

**Table 1.** Training data for Sequence L2018.

Target (No. of Frames)	Filter	Working Frame Range	Training Frames		Filter Size	$\beta$	$\gamma$	$\Psi$
			Range Taken	Interval Taken				
tank1 (448)	MACH Range-1	1–200	1–200	5	$118 \times 118$	-	0.1	-
	MACH Range-2	201–448	201–300	5	$118 \times 118$	-	1.0	-
	EMACH Range-1	1–200	1–200	5	$118 \times 118$	0.2	0.1	-
	EMACH Range-2	201–448	201–300	5	$118 \times 118$	0.1	1.0	-
-	DCCF	1–448	1–200	4	$12 \times 16$	-	-	-
-	PDCCF	1–448	1–200	4	$12 \times 16$	-	-	1.0, 1.5, 2.0

Filter formulations of the three other single target sequences (L1415, L2312 and M1406) are also similar to Sequence L2018. To overcome the size variation from the initial frames to the final frames, multiple filters of multiple range bins are required in some cases to apply at different ranges. The training data for these three single target sequences are depicted in Tables 2–4, respectively.

**Table 2.** Training data for Sequence L1415.

Target (No. of Frames)	Filter	Working Frame Range	Training frames		Filter Size	$\beta$	$\gamma$	$\Psi$
			Range Taken	Interval Taken				
Mantruck (281)	MACH	1–281	1–100	5	$118 \times 118$	-	1	-
	EMACH	1–281	1–100	5	$118 \times 118$	0.1	1	-
-	DCCF	1–281	1–100	5	$12 \times 16$	-	-	-
-	PDCCF	1–281	1–100	5	$6 \times 8$	-	-	1.0, 1.5, 2.0

**Table 3.** Training data for Sequence L2312.

Target (No. of Frames)	Filter	Working Frame Range	Training Frames		Filter Size	$\beta$	$\gamma$	$\Psi$
			Range Taken	Interval Taken				
APC1 (368)	MACH	1–368	1–100	5	$118 \times 118$	-	1	-
	EMACH	1–368	1–300	10	$118 \times 118$	0.1	1	-
-	DCCF	1–368	1–300	10	$12 \times 16$	-	-	-
-	PDCCF	1–368	1–300	10	$8 \times 12$	-	-	1.0, 1.5, 2.0

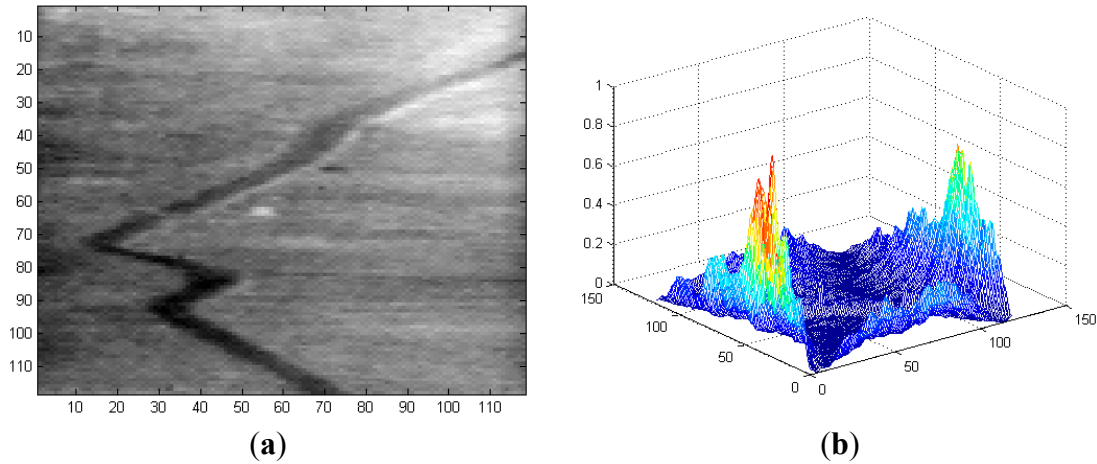
**Table 4.** Training data for Sequence M1406.

Target (No. of Frames)	Filter	Working Frame Range	Training Frames		Filter Size	$\beta$	$\gamma$	$\Psi$
			Range Taken	Interval Taken				
Bradley (380)	MACH	1–380	1–100	5	$118 \times 118$	-	1	-
	EMACH	1–380	1–100	5	$118 \times 118$	0.1	1	-
-	DCCF	1–380	1–300	10	$14 \times 16$	-	-	-
-	PDCCF Range-1	1–200	1–100	5	$8 \times 10$	-	-	1.0, 1.5, 2.0
-	PDCCF Range-2	201–380	201–300	5	$8 \times 10$	-	-	1.0, 1.5, 2.0

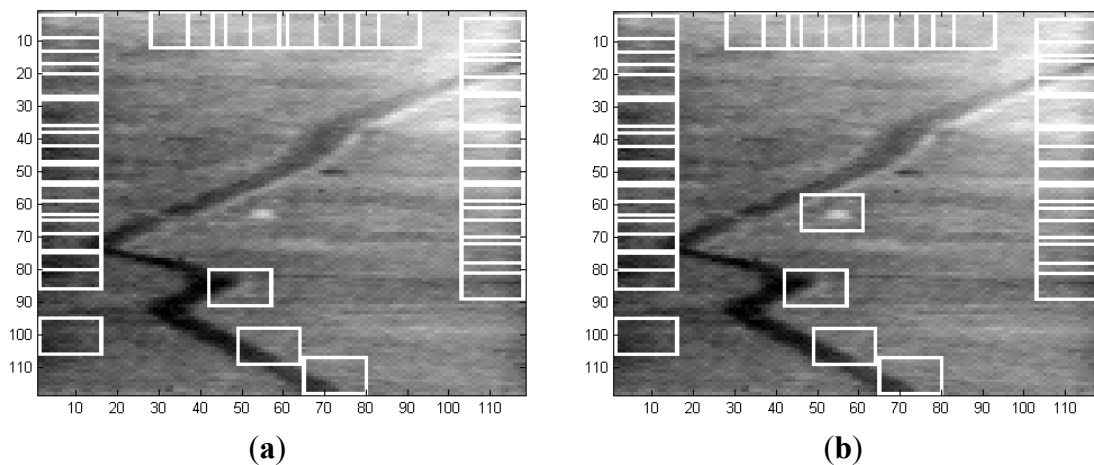
For analysis, consider the first frame (Frame 1) of Sequence L2018 shown in Figure 6a, apply Fourier transform to it and then correlate with the Fourier domain filter MACH Range-1 to obtain

the correlation output of Figure 6b. It is obvious that the correlation output contains false peaks and high-energy diffractions at low frequency which make the actual correlation signal negligible. For this reason, it requires 33 ROIs to include the target of interest as shown in Figure 7b.

**Figure 6.** (a) Frame 1 of Sequence L2018; and (b) correlation output with the filter maximum average correlation height (MACH) Range-1.

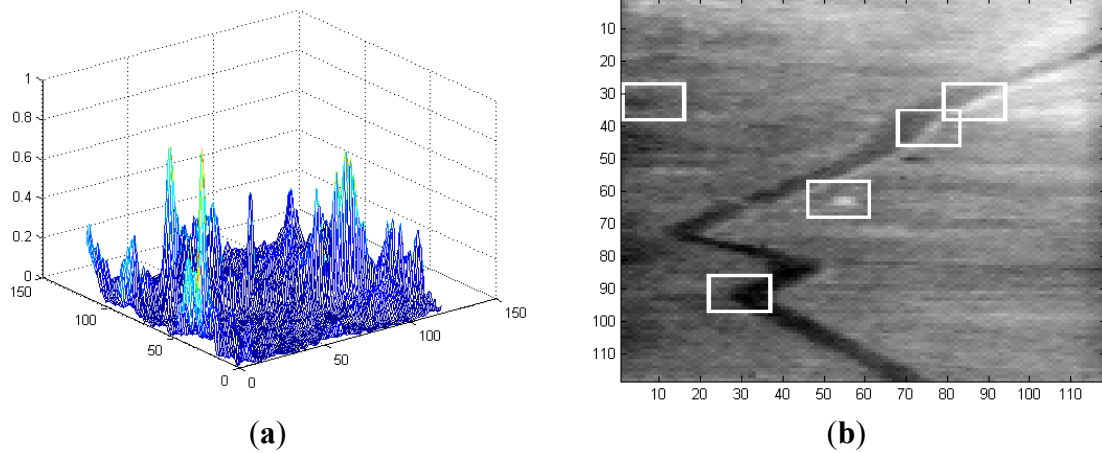


**Figure 7.** (a) 32 ROIs in Frame 1 of Sequence L2018; and (b) 33 ROIs in Frame 1 of Sequence L2018.



To eliminate the strong low frequency components, a notch filter is used before applying the inverse Fourier transform operation, which actually suppresses the Fourier domain components along the axes to zero. The correlation output after applying this notch filter is shown in Figure 8a. Five ROIs in Frame 1, selected based on this correlation output, are shown in Figure 8b where the target (tank1) is included within these ROIs. This notch filter is used in the detection stage of each frame using either MACH or EMACH filter for all image sequences analyzed in this work. The detection results using EMACH filter are similar to MACH filter with or without the use of notch filter.

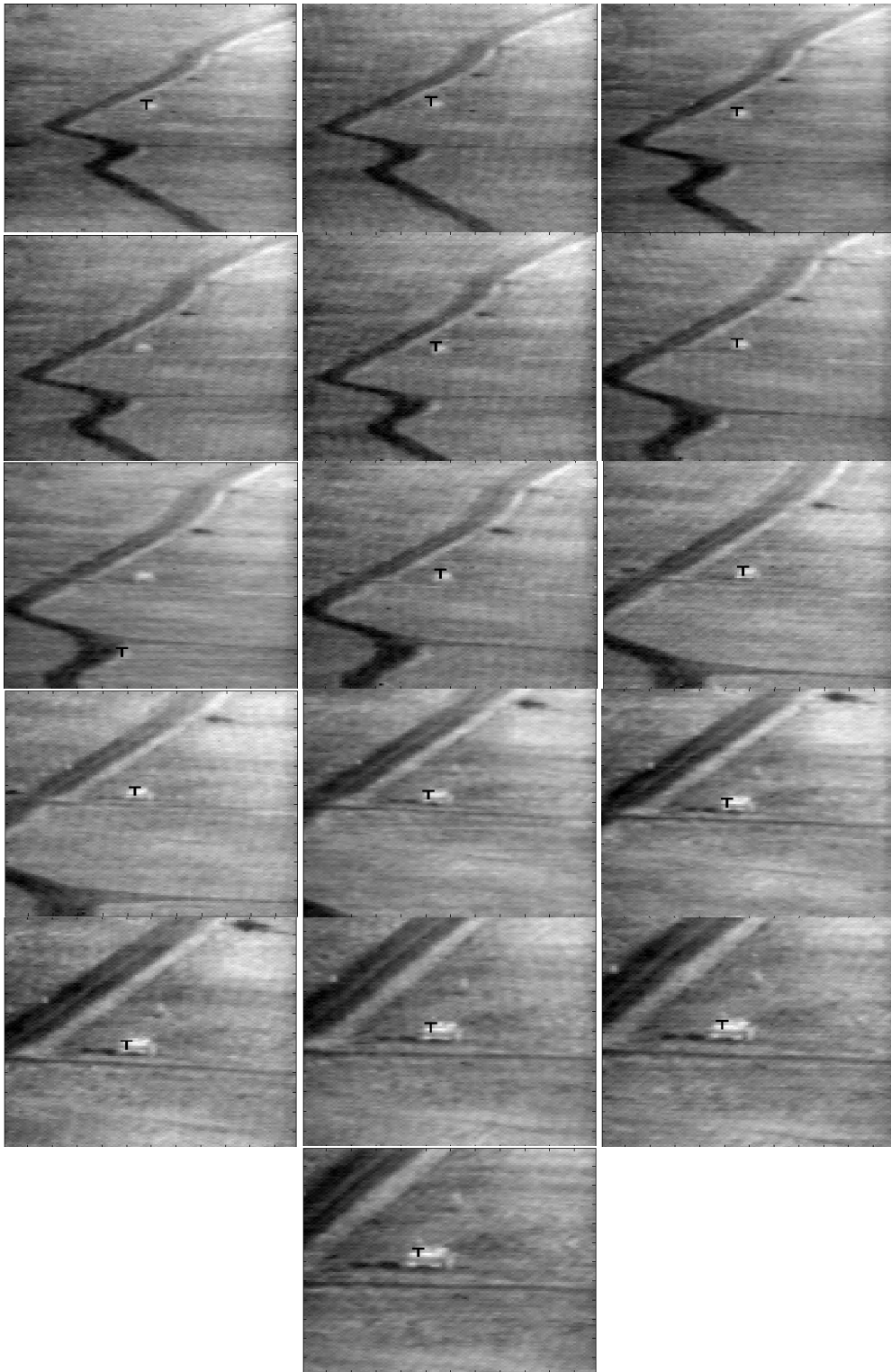
**Figure 8.** (a) Correlation output for Frame 1 of Sequence L2018 using MACH Range-1 filter along with notch filter; and (b) 5 ROIs in Frame 1.



It is obvious that the detection and classification cannot be done correctly with detection filters alone. Because of the presence of multiple identical and different targets and clutters, the highest correlation peak or PSR value is not always guaranteed for the desired target. Therefore, classification filters are used for improved discrimination for the single target image sequences too. It is found that the 1-class classification filters work well in rejecting the clutters and backgrounds. However, to make a decision for identifying the target and rejecting the clutter the ROI having the least distance is considered as the potential target. If the distance to the ROI having the second minimum distance is not higher than the prescribed percentage of the least distance, then the ROI having the least distance is rejected as clutter or background. Since EMACH filter has improved clutter rejection capability, the application of EMACH filter in the detection stage facilitates in lowering the number of ROIs introduced into the classification stage. For this particular sequence (L2018), it is found in the simulation that 8 ROIs are required to include the target in almost all frames in the case of MACH filter, whereas 6 ROIs are sufficient with EMACH filter. However, for generality, 8 ROIs are also considered in case of EMACH filter for this sequence.

All single target sequences are tested by using the three developed algorithms: MACH-DCCF, MACH-PDCCF and EMACH-PDCCF. Some of the frames of Sequence L2018 showing the results after applying the EMACH-PDCCF algorithm are given in Figure 9. The tracking algorithm inserts a “T” mark at the locations of the detected targets (tank1) at each frame as shown in Figure 9. The number at the lower left corner of each frame indicates the frame number. In Figure 9, some sample frames are also included where the classification is incorrect or no decision is inferred. The detection and tracking results for all single target sequences are summarized for all the three algorithms in Table 5. In the threshold column of the table, a single value indicates that a single threshold is used for all ranges. From Table 5, it is observed that the poorest performance is achieved for Sequence L2018 among the four single target sequences presented. This may be attributed to the scene complexities and the drawback of using same classification filters through all the ranges of the sequence.

Figure 9. Target detection and classification results of Sequence L2018.



**Table 5.** Tracking results of single-target sequences.

Seq. Name	Total Frames	No. of ROIs Taken	Threshold	Target Name	No. of Frames Target Present	No. of Frames Detected Correctly	Total No. of False Alarms	Percentage of Successful Detection
MACH-DCCF Algorithm								
L1415	281	4	1.00	mantruck	281	281	0	100
L2018	448	8	0.99	tank1	448	357	36	80
L2312	368	4	1.00	APC1	368	366	2	99
M1406	380	4	1.00	Bradley	380	380	0	100
MACH-PDCCF Algorithm								
L1415	281	4	1.00	mantruck	281	263	9	94
L2018	448	8	0.99	tank1	448	371	37	83
L2312	368	4	1.00	APC1	368	368	0	100
M1406	380	4	1.00	Bradley	380	353	27	93
EMACH-PDCCF Algorithm								
L1415	281	4	1.00	mantruck	281	274	1	98
L2018	448	8	1.00	tank1	448	386	60	86
L2312	368	4	1.00	APC1	368	368	0	100
M1406	380	4	1.00	Bradley	380	354	26	93

### 11.3. Two-Target Image Sequences

To assess the performance of the algorithms with two targets present in the scene, consider Sequences L1701 and L1911. The two targets in Sequence L1701 are Bradley and pickup. Out of 388 frames of this sequence, ground truth data for Bradley is available for 371 frames and that for pickup is available for 43 frames. The pickup disappears from the scene at Frame 31 and reappears at Frame 81 and again disappears at a later frame. The two targets, APC1 and tank1, in Sequence L1911 are present in all the 165 frames of the sequence. In these image sequences, the target sizes increase significantly from the first frame to the final frame. Therefore, different filters must be used for different ranges. For a particular range, two detection filters (MACH or EMACH) and a 2-class classification filter (DCCF or PDCCF) are required for a two-target sequence. Table 6 shows the training data/parameters selected for different filters for Sequence L1701 while Table 7 displays the same parameters for Sequence L1911. It may be mentioned that even if one target disappears after a few frames, both the detection filters and the 2-class classification filter are continued to apply to the remaining image frames; because in reality, it is not known whether any target is going out or coming back. This ensures the detection of a target that may reappear after a few frames.

**Table 6.** Training data for Sequence L1701.

Target (No. of Frames)	Filter	Working Frame Range	Training Frames		Filter Size	$\beta$	$\gamma$	$\Psi$
			Range Taken	Interval Taken				
Bradley (371)	MACH Range-1	1–100	1–100	5	$118 \times 118$	-	1.0	-
	MACH Range-2	101–200	101–200	5	$118 \times 118$	-	1.0	-
	MACH Range-3	201–300	201–300	5	$118 \times 118$	-	0.1	-
	MACH Range-4	301–388	301–370	5	$118 \times 118$		0.1	-
	EMACH Range-1	1–100	1–100	5	$118 \times 118$	0.1	1.0	-
	EMACH Range-2	101–200	101–200	5	$118 \times 118$	0.1	1.0	-
	EMACH Range-3	201–300	201–300	5	$118 \times 118$	0.1	0.1	-
	EMACH Range-4	301–388	301–370	5	$118 \times 118$	0.1	0.1	-
Pickup (43)	MACH Range-1	1–388	1–43	3	$118 \times 118$		1.0	-
	EMACH Range-1	1–388	1–43	3	$118 \times 118$	0.1	1.0	-
-	DCCF Range-1	1–100	1–100	5	$6 \times 8$	-	-	-
-	DCCF Range-2	101–200	201–300	5	$8 \times 12$	-	-	-
-	DCCF Range-3	201–300	201–300	5	$10 \times 18$	-	-	-
-	DCCF Range-4	301–388	301–370	5	$10 \times 18$	-	-	-
-	PDCCF Range-1	1–100	1–100	5	$8 \times 8$	-	-	1.0, 1.5, 2.0
-	PDCCF Range-2	101–200	101–200	5	$8 \times 12$	-	-	1.0, 1.5, 2.0
-	PDCCF Range-3	201–300	201–300	5	$10 \times 18$	-	-	1.0, 1.5, 2.0
-	PDCCF Range-4	301–388	301–370	5	$10 \times 18$	-	-	1.0, 1.5, 2.0

**Table 7.** Training data for Sequence L1911.

Target (No. of Frames)	Filter	Working Frame Range	Training Frames		Filter Size	$\beta$	$\gamma$	$\Psi$
			Range Taken	Interval Taken				
APC1 (165)	MACH Range-1	1:100	1:100	5	$118 \times 118$	-	0.1	-
	MACH Range-2	101:165	101:150	5	$118 \times 118$	-	0.1	-
	EMACH Range-1	1:100	1:100	5	$118 \times 118$	0.1	0.1	-
	EMACH Range-2	101:165	101:150	5	$118 \times 118$	0.1	0.1	-
tank1 (165)	MACH Range-1	1:100	1:100	5	$118 \times 118$	-	0.1	-
	MACH Range-2	1:165	101:150	5	$118 \times 118$	-	0.1	-
	EMACH Range-1	1:100	1:100	5	$118 \times 118$	0.1	0.1	-
	EMACH Range-2	1:165	101:150	5	$118 \times 118$	0.1	0.1	-
-	DCCF Range-1	1:100	1:100	5	$8 \times 16$	-	-	-
-	DCCF Range-2	101:130	101:130	5	$8 \times 16$	-	-	-
-	DCCF Range-3	131:165	131:160	5	$16 \times 36$	-	-	-
-	PDCCF Range-1	1:130	1:100	5	$8 \times 16$	-	-	1.0, 1.5, 2.0
-	PDCCF Range-2	1:130	1:100	5	$8 \times 16$	-	-	1.0, 1.5, 2.0
-	PDCCF Range-3	131:165	131:160	5	$16 \times 36$	-	-	1.0, 1.5, 2.0

Using the detection and classification filters having the design parameters in Tables 6 and 7, all algorithms are tested for detection, classification and tracking of the objects in Sequences L1701 and L1911. The results obtained after applying the EMACH-PDCCF algorithm on Sequence L1701 are shown for some sample frames in Figure 10. The tracking algorithm inserts “T1” for the detected Class-1 type targets (Bradley) and “TII” for the detected Class-2 type targets (pickup) at

their corresponding location in the image frame. The frame number is shown at the lower left corner of each frame displayed. From Figure 10, it is obvious that the tracking algorithm can successfully detect and classify the targets when they are present in the input scene. The complete tracking results for both the sequences are summarized in Table 8 for all algorithms. In the threshold column of the table, a single value indicates that a single threshold is used for all ranges. Otherwise, different threshold values in that column displays the thresholds used for various ranges of classification filters employed. It is observed that MACH-DCCF algorithm fails for Sequence L1911 in detecting and classifying the targets, and rejecting the clutters. On the other hand, EMACH-PDCCF algorithm provides the best results considering all the factors simultaneously, such as, required number of ROIs, percentage of successful detection and the total number of false alarms.

**Figure 10.** Target detection and classification results of Sequence L1701.

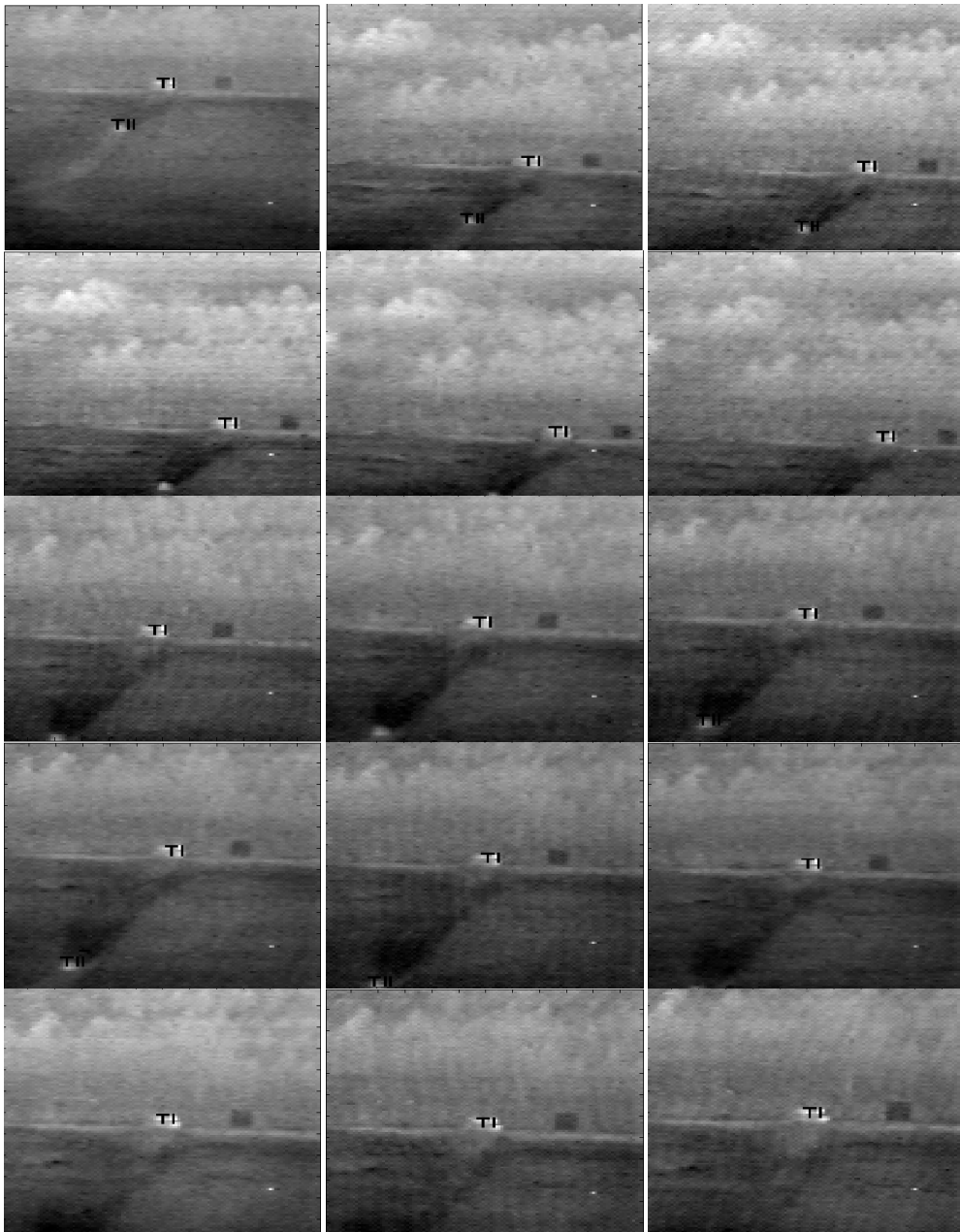




Figure 10. Cont.

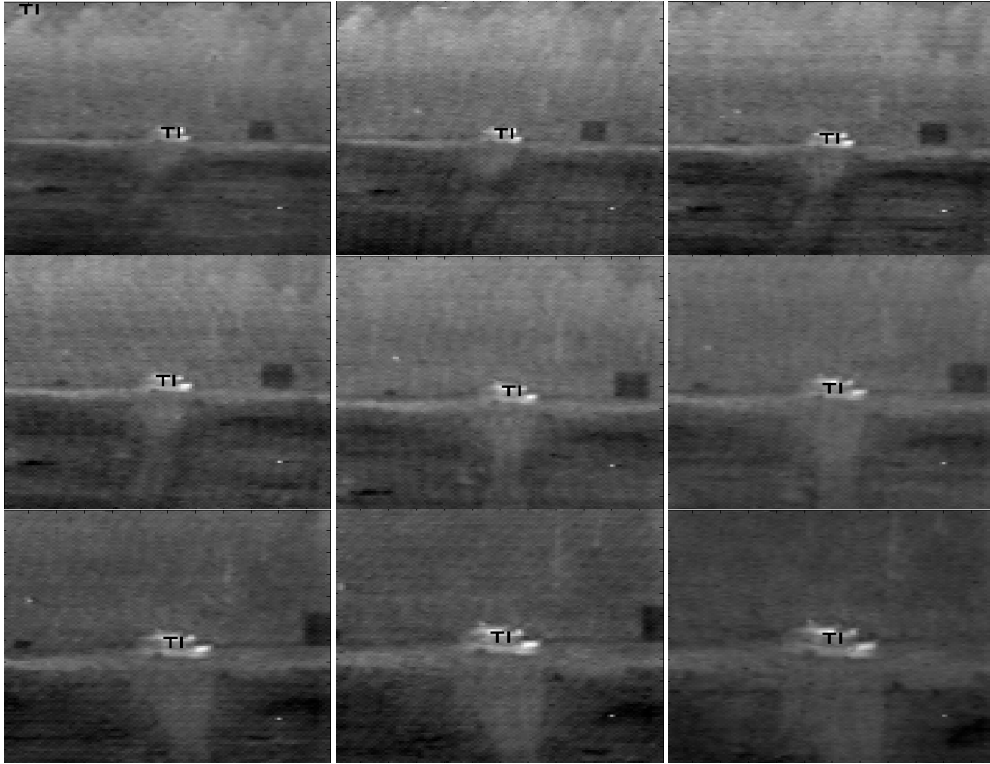


Table 8. Tracking results of two-target sequences.

Seq. Name	Total Frames	No. of ROIs Taken	Threshold	Target Name	No. of Frames Target Present	No. of Frames Detected Correctly	Total No. of False Alarms	Percentage of Successful Detection
MACH-DCCF Algorithm								
L1701	388	6	0.70, 0.30, 0.05, 0.04	Bradley	388	230	248	59
				pickup	45	26	2	58
L1911	165	6	0.40	APC1	-	-	-	Fails
				Tank1	-	-	-	Fails
MACH-PDCCF Algorithm								
L1701	388	6	0.40, 0.05, 0.06, 0.06	Bradley	388	369	12	95
				pickup	45	28	1	62
L1911	165	6	0.40	APC1	165	152	0	92
				tank1	165	165	3	100
EMACH-PDCCF Algorithm								
L1701	388	4	0.40, 0.05, 0.06, 0.06	Bradley	388	369	9	95
				pickup	45	31	8	69
L1911	165	4	0.40	APC1	165	160	0	97
				tank1	165	165	5	100

#### 11.4. Three-Target Image Sequences

The three target images in Sequence L1618 are APC1, M60 and truck. Out of 300 frames in this sequence, ground truth data for APC1 is available for 291 frames, ground truth data for M60 is

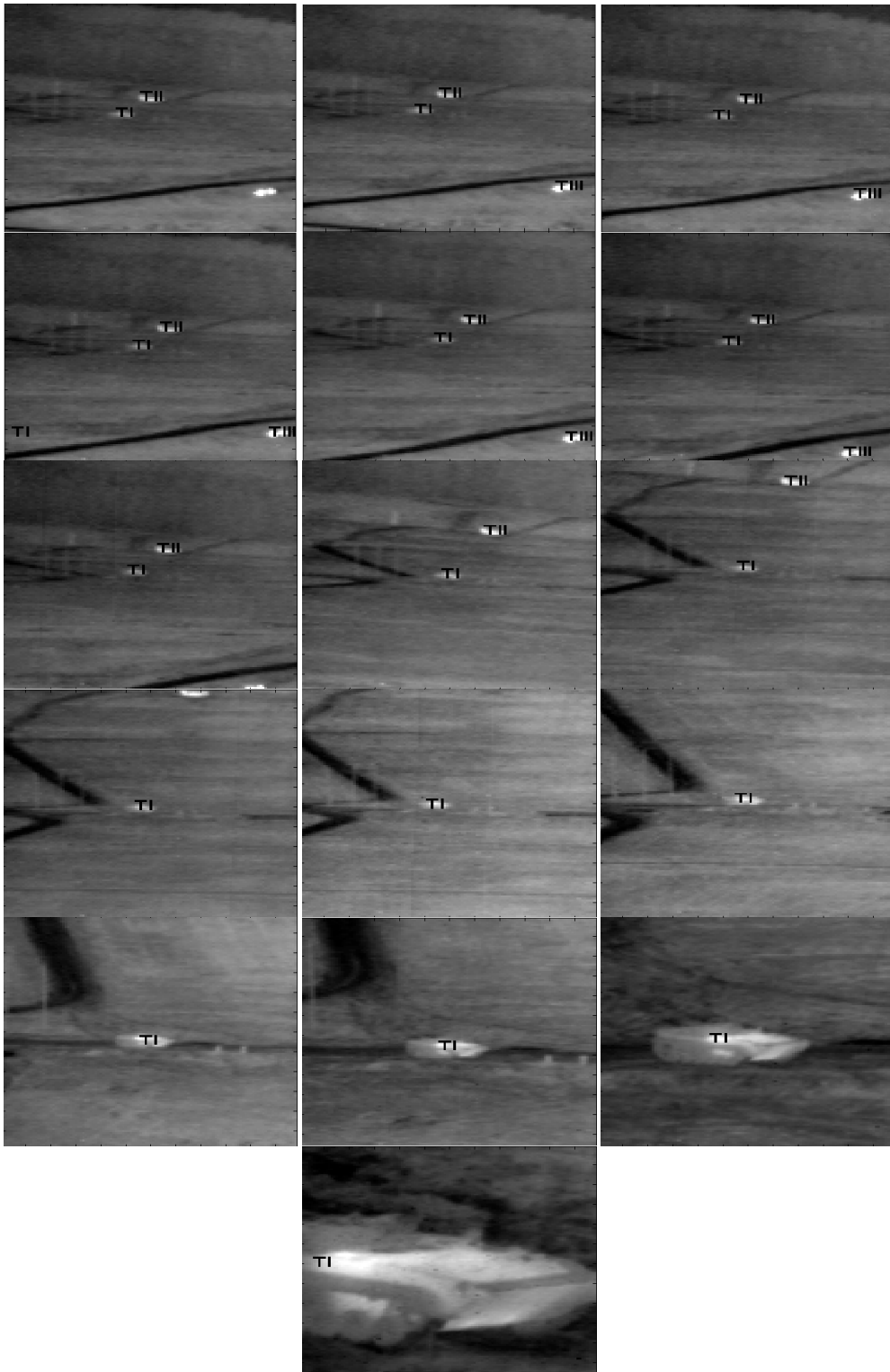
found for 101 frames and that for truck is found for 6 frames. The truck disappears from the sequence at Frame 18 and M60 disappears from the sequence at Frame 104. Like the other sequences, the size of the targets increases significantly from the first frame to the final frame. Thus, different filters are required for different ranges of this sequence. Assuming three expected targets in the scene for any particular range, three detection filters (MACH or EMACH) and a three-class classification filter (DCCF or PDCCF) are required for every frame in each range of this sequence. The training data/parameters for different filters, chosen for Sequence L1618 are displayed in Table 9.

**Table 9.** Training data for Sequence L1618.

Target (No. of Frames)	Filter	Working Frame Range	Training Frames		Filter Size	$\beta$	$\gamma$	$\Psi$
			Range Taken	Interval Taken				
APC1 (291)	MACH Range-1	1–100	1–100	5	$118 \times 118$	-	1.0	-
	MACH Range-2	101–200	101–200	5	$118 \times 118$	-	1.0	-
	MACH Range-3	201–300	201–290	5	$118 \times 118$	-	1.0	-
	EMACH Range-1	1–100	1–100	5	$118 \times 118$	0.1	1.0	-
	EMACH Range-2	101–200	101–200	5	$118 \times 118$	0.1	1.0	-
	EMACH Range-3	201–300	201–290	5	$118 \times 118$	0.1	1.0	-
M60 (101)	MACH Range-1	1–300	1–100	5	$118 \times 118$	-	1.0	-
	EMACH Range-1	1–300	1–100	5	$118 \times 118$	0.1	1.0	-
Truck (6)	MACH Range-1	1–300	1–6	1	$118 \times 118$	-	1.0	-
	EMACH Range-1	1–388	1–6	1	$118 \times 118$	0.1	1.0	-
-	DCCF Range-1	1–100	1–100	5	$6 \times 12$	-	-	-
-	DCCF Range-2	101–200	201–300	5	$8 \times 16$	-	-	-
-	DCCF Range-3	201–300	201–290	5	$8 \times 16$	-	-	-
-	PDCCF Range-1	1–100	1–100	5	$6 \times 12$	-	-	1.0, 1.5, 2.0
-	PDCCF Range-2	101–200	101–200	5	$8 \times 16$	-	-	1.0, 1.5, 2.0
-	PDCCF Range-3	201–300	201–290	5	$8 \times 16$	-	-	1.0, 1.5, 2.0

To evaluate the performance of all the algorithms for detection, clutter rejection, and classification as well as tracking of the three objects in three-target sequence, the designed detection and classification filters of Table 9 for different ranges are applied to Sequence L1618. The results obtained for the EMACH-PDCCF algorithm with Sequence L1618 are shown for some sample frames in Figure 11. The tracking algorithm places “T1” for the detected Class-1 type targets (APC1), “TII” for the detected Class-2 type targets (M60) and “TIII” for the detected Class-3 type targets (truck) at their corresponding locations in the frames. In Figure 11, the frame numbers are shown at the lower left corner of each frame. The complete tracking results for the sequence are summarized in Table 10 for all algorithms. In the threshold column of the table, a single value indicates that a single threshold is used for all ranges. Otherwise, different threshold values in that column displays the thresholds used for various ranges of classification filters employed.

Figure 11. Target detection and classification results of Sequence L1618.



**Table 10.** Tracking results of three-target sequences.

Seq. Name	Total Frames	No. of ROIs Taken	Threshold	Target Name	No. of Frames Target Present	No. of Frames Detected Correctly	Total No. of False Alarms	Percentage of Successful Detection
MACH-DCCF Algorithm								
L1618	300	6	0.30, 0.52, 0.52	APC1	300	273	54	91
				M60	103	71	0	69
				truck	17	0	0	0
MACH-PDCCF Algorithm								
L1618	300	6	0.52	APC1	300	297	20	99
				M60	103	99	0	96
				truck	17	12	0	71
EMACH-PDCCF Algorithm								
L1618	300	4	0.52	APC1	300	297	23	99
				M60	103	99	0	96
				truck	17	12	0	71

## 12. Conclusions

Pattern recognition and tracking in FLIR imagery is a challenging problem due to various factors such as low resolution, low signal-to-noise ratio, different 3D orientations of the targets, effects of global motion, and close proximity with similar objects. In this paper, we reviewed the recent trends and advancements in distortion-invariant pattern recognition algorithms for single/multiple target detection and tracking in FLIR imagery using correlation filters. Each detection/tracking algorithm utilizes various properties of targets and image frames of a given FLIR sequence. Test results using real life FLIR image sequences are presented to verify the effectiveness of the filter-based pattern recognition and tracking techniques. Future work in this area would include a review of techniques beyond correlation that are particularly useful for high resolution targets. Also, development and inclusion of techniques with a dynamic update of the target model may be considered.

## Conflicts of Interest

The authors declare no conflict of interest.

## References

- Schalkoff, R. *Pattern Recognition, Statistical, Structural and Neural Approaches*; John Wiley & Sons: New York, NY, USA, 1992.
- Duda, R.O.; Hart, P.E.; Stork, D.G. *Pattern Classification*, 2nd ed.; Wiley-Interscience: New York, NY, USA, 2000.
- Gonzalez, R.C.; Woods, R.E.; *Digital Image Processing*, 2nd ed.; Prentice-Hall, Inc.: Taiwan, 2002.

4. Brundstrom, K.; Schenkman, B.N.; Jacobson, B. Object detection in cluttered infrared images. *Opt. Eng.* **2003**, *42*, 388–399.
5. Strehl, A.; Aggarwal, J.K. Detecting moving objects in airborne forward looking infra-red sequences. *J. Mach. Vis. Appl.* **2000**, *11*, 267–276.
6. Yilmaz, A.; Shafique, K.; Shah, M. Target tracking in airborne forward looking infrared imagery. *Image Vis. Comput. J.* **2003**, *21*, 623–635.
7. Alam, M.S.; Haque, M.; Khan, J.F.; Kettani, H. Fringe-adjusted JTC based target detection and tracking in FLIR image sequence. *Opt. Eng.* **2004**, *43*, 1407–1413.
8. VanderLugt, A. Signal detection by complex spatial filtering. *IEEE Trans. Inf. Theory* **1964**, *10*, 139–146.
9. Casasent, D.; Furman, A. Sources of correlation degradation. *Appl. Opt.* **1977**, *16*, 1652–1661.
10. Gianino, P.D.; Horner, J.L. Phase-only matched filtering. *Appl. Opt.* **1984**, *23*, 812–816.
11. Mu, G.G.; Wang, X.M.; Wang, Z.Q. Amplitude-compensated matched filtering. *Appl. Opt.* **1988**, *27*, 3461–3463.
12. Awwal, A.A.S.; Karim, M.A.; Jahan, S.R. Improved correlation discrimination using an amplitude modulated phase-only filter. *Appl. Opt.* **1990**, *29*, 233–236.
13. Hester, C.F.; Casasent, D. Multivariant technique for multiclass pattern recognition. *Appl. Opt.* **1980**, *19*, 1758–1761.
14. Alam, M.S.; Awwal, A.A.S. Scale invariant amplitude modulated phase-only filtering. *J. Opt. Laser Technol.* **2000**, *32*, 231–234.
15. Alam, M.S.; Chen, X.; Karim, M.A. Distortion invariant fringe-adjusted joint transform correlator. *J. Appl. Opt.* **1997**, *36*, 7422–7427.
16. Casasent, D. Unified synthetic discriminant function computational formulation. *Appl. Opt.* **1984**, *23*, 1620–1627.
17. Bahri, Z.; Kumar, B.V.K.V. Generalized synthetic discriminant functions. *J. Opt. Soc. Am. A* **1988**, *5*, 562–571.
18. Kumar, B.V.K.V. Minimum variance synthetic discriminant function. *J. Opt. Soc. Am. A* **1986**, *3*, 1579–1584.
19. Mahalanobis, A.; Kumar, B.V.K.V.; Casasent, D. Minimum average correlation energy filters. *Appl. Opt.* **1987**, *26*, 3633–3640.
20. Casasent, D.; Ravichandran, G.; Bollapraggada, S. Gaussian MACE correlation filters. *Appl. Opt.* **1991**, *30*, 5176–5181.
21. Kumar, B.V.K.V.; Mahalanobis, A.; Song, S.; Sims, S.R.F.; Epperson, J.F. Minimum squared error synthetic discriminant functions. *Opt. Eng.* **1992**, *31*, 915–922.
22. Mahalanobis, A.; Kumar, B.V.K.V.; Sims, S.R.F.; Epperson, J. Unconstrained correlation filters. *Appl. Opt.* **1994**, *33*, 3751–3759.
23. Mahalanobis, A.; Kumar, B.V.K.V. Optimality of the maximum average correlation height filter for detection of targets in noise. *Opt. Eng.* **1997**, *36*, 26423–2648.
24. Alkanhal, M.; Kumar, B.V.K.V.; Mahalanobis, A. Improved clutter rejection in automatic target recognition (ATR) synthetic aperture radar (SAR) imagery using the extended maximum average correlation height (EMACH) filter. *Proc. SPIE.* **2000**, *4053*, 332–339.

25. Alkanhal, M.; Kumar, B.V.K.V.; Mahalanobis, A. Improving the false alarm capabilities of the maximum average correlation height correlation filter. *Opt. Eng.* **2000**, *39*, 1133–1141.
26. Mahalanobis, A.; Carlson, D.W.; Kumar, B.V.K.V.; Sims, S.R.F. Distance classifier correlation filters. *Proc. SPIE* **1994**, *2238*, 2–13.
27. Mahalanobis, A.; Kumar, B.V.K.V.; Sims, S.R.F. Distance classifier correlation filters for distortion tolerance, discrimination and clutter rejection. *Proc. SPIE* **1993**, *2026*, 325–335.
28. Mahalanobis, A.; Kumar, B.V.K.V.; Sims, S.R.F. Distance classifier correlation filters for multiclass target recognition. *Appl. Opt.* **1996**, *35*, 3127–3133.
29. Alkanhal, M.; Kumar, B.V.K.V. Polynomial distance classifier correlation filter for pattern recognition. *Appl. Opt.* **2003**, *42*, 4688–4708.
30. Muise, R.; Mahalanobis, A.; Mohapatra, R.; Li, X.; Han, D.; Mikhael, W. Constrained quadratic correlation filters for target detection. *Appl. Opt.* **2004**, *43*, 304–314.
31. Hwang, J.; Ooi, Y.; Ozawa, S. Visual feedback control system for tracking and zooming a target. *Proc. Int. Conf. Power Electron. Motion Control IEEE* **1992**, *2*, 740–745.
32. Lipton, A.J.; Fujiyoshi, H.; Patil, R.S. Moving target classification and tracking from real-time video. In Proceedings of the Workshop of the Application of Computer Vision, Princeton, NJ, USA, 19–21 October 1998; pp. 8–14.
33. Bal, A.; Alam, M.S. Automatic Target tracking in FLIR image sequences using intensity variation function and template modeling. *IEEE Trans. Instrum. Meas.* **2005**, *54*, 1846–1852.
34. Dawoud, A.; Alam, M.S.; Bal, A.; Loo, C. Target tracking in infrared imagery using weighted composite reference function based decision fusion. *IEEE Trans. Image Process.* **2006**, *15*, 404–410.
35. Alam, M.S.; Bal, A.; Horache, E.; Goh, S.F.; Loo, C.; Regula, S.; Sharma, A. Metrics for evaluating the performance of joint transform correlation based target recognition and tracking algorithms. *Opt. Eng.* **2005**, *44*, 067005.
36. Bal, A.; Alam, M. Dynamic target tracking using fringe-adjusted joint transform correlation and template matching. *Appl. Opt.* **2004**, *43*, 4874–4881.
37. Dawoud, A.; Alam, M.S.; Bal, A.; Loo, C. Decision fusion algorithm for target tracking in infrared imagery. *Opt. Eng.* **2005**, *44*, doi:10.1117/1.1844534.
38. Alam, M.S.; Khan, J.; Bal, A. Heteroassociative multiple-target tracking by fringe-adjusted joint transform correlation. *Appl. Opt.* **2004**, *43*, 358–365.
39. Mahalanobis, A. Correlation filters for object tracking target re-acquisition and smart aimpoint selection. *Proc. SPIE* **1997**, *3073*, 25–32.
40. Mahalanobis, A.; Muise, R. Advanced detection and correlation based automatic target detection. *Proc. SPIE* **2001**, *4379*, 466–471.
41. Mahalanobis, A.; Carlson, D.W.; Kumar, B.V.K.V. Evaluation of MACH and DCCF correlation filters for SAR ATR using MSTAR public data base. *Proc. SPIE* **1998**, *3370*, 460–468.
42. Mahalanobis, A.; Ortiz, L.A.; Kumar, B.V.K.V. Performance of the MACH filter and DCCF algorithms on the 10-class public release MSTAR data set. *Proc. SPIE* **1999**, *3721*, 285–289.
43. Carlson, D.W.; Riddle, J.G. Clutter background spectral density estimation for SAR target recognition with composite correlation filters. *Proc. SPIE* **2003**, *5106*, 64–71.

44. Perona, M.T.; Mahalanobis, A.; Zachery, K.N. LADAR automatic target recognition using correlation filters. *Proc. SPIE* **1999**, *3718*, 388–396.
45. Perona, M.T.; Mahalanobis, A. System-level evaluation of LADAR ATR using correlation filters. *Proc. SPIE* **2000**, *4050*, 69–75.
46. Nevel, A.V.; Mahalanobis, A. Comparative study of maximum average correlation height filter variants using ladar imagery. *Opt. Eng.* **2003**, *42*, 541–550.
47. Sims, S.R.F.; Mahalanobis, A. Performance evaluation of quadratic correlation filters for target detection and discrimination in infrared imagery. *Opt. Eng.* **2004**, *43*, 1705–1711.
48. Bhuiyan, S.; Alam, M.S.; Alkanhal, M. A new two-stage correlation based approach for target detection and tracking in FLIR imagery using EMACH and PDCCF filters. *Opt. Eng.* **2007**, *46*, 086401.
49. Islam, M.F.; Alam, M.S. Improved clutter rejection in automatic target recognition and tracking using eigen-extended maximum average correlation height (EEMACH) filter and polynomial distance classifier correlation filter (PDCCF). *Proc. SPIE* **2006**, *6245*, 62450B.
50. Bhuiyan, S.M.A.; Alam, M.S.; Sims, S.R.F. Target detection, classification and tracking using MACH and PDCCF filter combination. *Opt. Eng.* **2006**, *45*, 116401.
51. Bhuiyan, S.; Khan, J.F.; Alam, M.S. Power enhanced extended maximum average correlation height filter for target detection, to appear. In Proceedings of the IEEE SoutheastCon, Lexington, Kentucky, KY, USA, 13–16 March 2014.

# Automated Detection and Recognition of Wildlife Using Thermal Cameras

Peter Christiansen, Kim Arild Steen, Rasmus Nyholm Jørgensen and Henrik Karstoft

**Abstract:** In agricultural mowing operations, thousands of animals are injured or killed each year, due to the increased working widths and speeds of agricultural machinery. Detection and recognition of wildlife within the agricultural fields is important to reduce wildlife mortality and, thereby, promote wildlife-friendly farming. The work presented in this paper contributes to the automated detection and classification of animals in thermal imaging. The methods and results are based on top-view images taken manually from a lift to motivate work towards unmanned aerial vehicle-based detection and recognition. Hot objects are detected based on a threshold dynamically adjusted to each frame. For the classification of animals, we propose a novel thermal feature extraction algorithm. For each detected object, a thermal signature is calculated using morphological operations. The thermal signature describes heat characteristics of objects and is partly invariant to translation, rotation, scale and posture. The discrete cosine transform (DCT) is used to parameterize the thermal signature and, thereby, calculate a feature vector, which is used for subsequent classification. Using a k-nearest-neighbor (kNN) classifier, animals are discriminated from non-animals with a balanced classification accuracy of 84.7% in an altitude range of 3–10 m and an accuracy of 75.2% for an altitude range of 10–20 m. To incorporate temporal information in the classification, a tracking algorithm is proposed. Using temporal information improves the balanced classification accuracy to 93.3% in an altitude range 3–10 of meters and 77.7% in an altitude range of 10–20 m

Reprinted from *Sensors*. Cite as: Christiansen, P.; Steen, K.A.; Jørgensen, R.N.; Karstoft, H. Automated Detection and Recognition of Wildlife Using Thermal Cameras. *Sensors* **2014**, *14*, 13778–13793.

## 1. Introduction

In agricultural mowing operations, thousands of animals are injured or killed each year, due to the increased working widths and speeds of agricultural machinery. Several methods and approaches have been used to reduce this wildlife mortality. Delayed mowing date, altered mowing patterns (e.g., mowing from the center outwards [1]) or strategy (e.g., leaving edge strips), longer mowing intervals, the reduction of speed or higher cutting height [1] have been suggested to reduce wildlife mortality rates. Likewise, searches with trained dogs prior to mowing may enable the farmer to remove, e.g., leverets and fawns to safety, whereas areas with bird nests can be marked and avoided. Alternatively, various scaring devices, such as flushing bars [1] or plastic sacks set out on poles before mowing [2], have been reported to reduce wildlife mortality. However, wildlife-friendly farming often results in lower efficiency. Therefore, attempts have been made to develop automatic systems capable of detecting wild animals in the crop without unnecessary cessation of the farming operation. For example, a detection system based on infrared sensors has been reported to reduce wildlife



mortality in Germany [3]. The disadvantage of the system proposed in [3] is its low efficiency, as the maximum search power is around 3 ha/h, when the weather conditions are fit.

In the [4], principles from [3] were further developed and tested. They conclude that vision systems are not a viable solution when the cameras are mounted on the agricultural machinery, as image quality is highly affected by the speed and vibrations of the machine. Instead a UAV-based system is utilized [5]. Using this solution, the movement of the tractor does not affect the image quality, and it is possible to manually scan large areas. The authors show that thermal imaging can be used to detect roe deer fawns based on aerial footage. However, the detection is performed manually and should be automated to increase efficiency. They conclude that the thermal imaging strategy is sensitive to the detection of false positives, meaning that objects that are heated by the Sun are falsely labeled (manually) as roe deer fawns.

UAVs are an emerging technology, and in modern agriculture, it can be utilized for many purposes. The UAV technology is capable of performing advanced and high precision tasks, due to the flight capabilities and the possibility to equip the aerial vehicle with computers and sensors, including thermal cameras. During the last two decades, thermal imaging has gained more and more attention in computer vision and digital image processing research and applications. Thermal imaging has become an interesting technology in outdoor surveillance, pedestrian detection and agriculture, due to the invariance to illumination and the lowered price of thermal cameras [6].

In [7,8], thermal imaging is used for person detection. The authors present thermal images of people at different times of the day and during summer and winter. Here, it is clear that the object of interest (people) does not always appear brighter (higher temperature) than the background. They propose background subtraction techniques, followed by a contour-based approach to detect people in the thermal images. Background subtraction is also utilized in [9–11]. However, this approach is not suitable for our UAV-based application with non-stationary cameras, as the background changes rapidly over time, and it is not possible to construct a background image. Another approach is the detection of hot spots based on a fixed temperature threshold [12–15]. In [16], a probabilistic approach for defining the threshold value is presented; however, it is still a fixed value.

There is little research within the automatic detection and recognition of animals in thermal images. Most research with thermal cameras involve static cameras, where background subtraction has been used for robust people detection in thermal images. In [5], a UAV, equipped with a thermal camera, is used for the detection of roe deer fawns in agricultural fields. Detection is based on manual visual inspection, and the author utilizes automatic gain control to enhance the appearance of living objects. An algorithm for the classification of roe deer fawns in thermal images is presented in [17]. They utilize normalized compression distance as the features followed by a clustering algorithm for classification. The dataset consists of 103 images, with 26 containing fawns hidden in grass. The same dataset is used in [18], where fast compression distance is applied in the feature extraction step and a nearest neighbor classifier is used for classification. In both papers, the features are derived from a dictionary, generated by a compression algorithm. These features are scale invariant; however, they are not rotation invariant, and they rely on absolute temperature measurements, which could be invalidated if animals are heated by the Sun. An algorithm for automatic detection of wildlife

in agricultural fields is presented in [19]. However, the distinction between animals and other hot objects is not a part of the results presented. An algorithm for the identification of deer, to avoid deer-vehicle crashes, is presented in [20]. The histogram of oriented gradient (HOG) is used for feature extraction, and support vector machines are utilized in the classification step. Their method relies on occlusion-free side-view images and performs poorly if these criteria are not met.

This paper presents a method for detecting and recognizing animals in thermal images. The method is based on a threshold, dynamically fitted for each frame, and a novel feature extraction algorithm, which is invariant to rotation, scaling and, partly, posture. Detected objects are tracked in subsequent images to include temporal information within the recognition part of the algorithm. The algorithm has been tested in a controlled experiment, using real animals, in the context of wildlife-friendly farming.

## 2. Materials and Methods

A telescopic boom is used to capture top-view images above a stationary scene, as shown in Figure 1. By using a telescopic boom lift, images can be captured at different altitudes, thus simulating the UAV. Unlike, using a UAV, the captured images are not affected by wind or vibrations within the UAV, which could affect image quality. Furthermore, the setup also avoids the compression of data, which might degrade data quality with respect to classification.

**Figure 1.** The setup used for capturing visual RGB and thermal images.



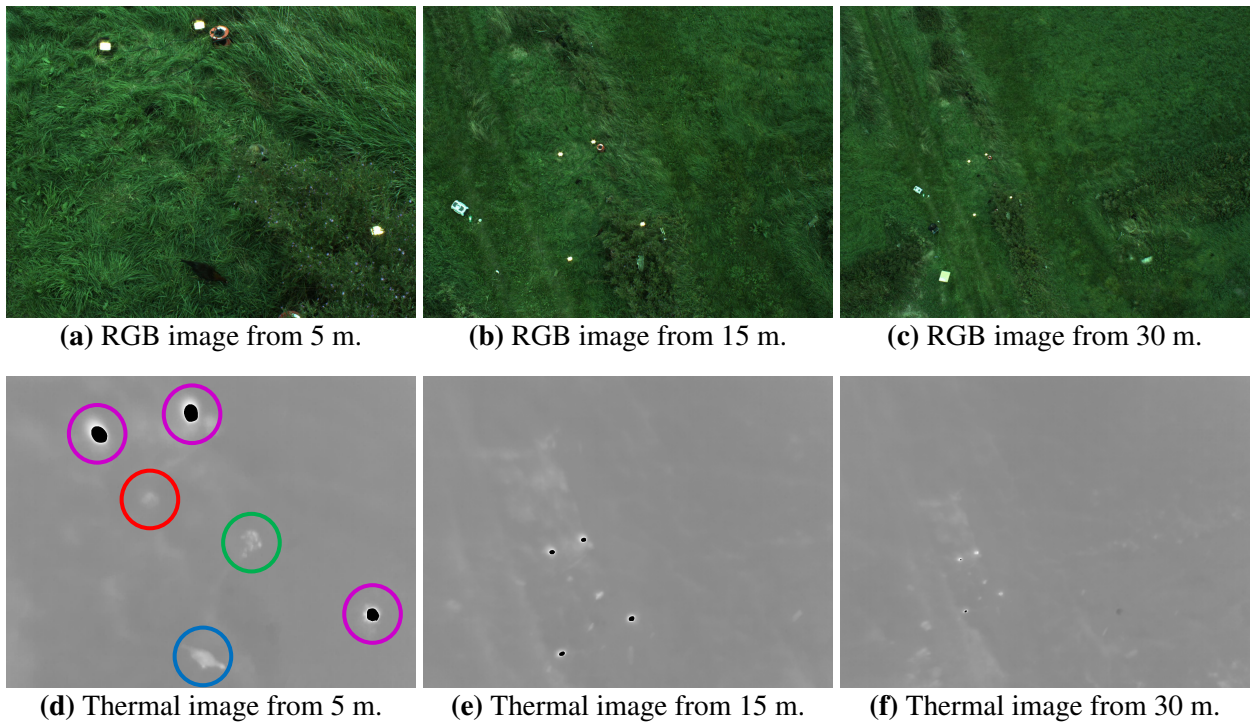
### 2.1. Data

A rig with a thermal and a regular RGB camera is mounted on the lift, recording 9 frames per second with a resolution of  $320 \times 240$  and  $1624 \times 1234$ , respectively. Animals and four halogen spotlights (used as reference points) were manually placed below the lift within the field-of-view of the two imaging sensors. The altitude of the cameras was measured with a GPS. A total of six recordings were made through two days with temperatures of  $15\text{--}19\text{ }^{\circ}\text{C}$  and  $16\text{--}23\text{ }^{\circ}\text{C}$  respectively. The recordings were captured using different areas around the scene shown in Figure 1.

Each recording starts at three meters followed by an increase in height of up to 25–35 m and then back again. The telescopic boom alternates the height position of the camera, while keeping the scene within the image frame. The use of a lift instead of an actual UAV results in less motion blur.

The data used in this paper consist of a total of 3987 frames with the presence of animals (rabbit and chicken), together with other hot objects (halogen spotlights, molehills, wooden poles, *etc.*). Animals were able to move within in a certain area due to fixation by a 30-cm leash. In Figure 2 the same scene is captured from 5 m, 15 m and 30 m. All thermal images are rescaled to the same size as the RGB images.

**Figure 2.** Visual RGB and thermal images capture the same scene from 5 m (a), 15 m (b) and 30 m (c). The scene consists of four halogen spotlights, a molehill, a rabbit and a chicken. The halogen spotlights are easily visible in all images. In (d) a molehill, a rabbit, a chicken and three halogen spotlights are marked.



## 2.2. Detection

The measured temperature is not the actual body temperature of the animal, as the measurement is also dependent on heating from the Sun, the insulative properties of the fur, or feather coat, and the distance between the animal and the camera [21]. These factors may vary in outdoor environments; hence, the segmentation and subsequent blob detection needs to adapt to this environment.

We use a threshold dynamically adjusted to each frame by using the median temperature  $\tilde{t}$  in the image, to exclude outliers. The threshold value is set by:

$$th = \tilde{t} + c \quad (1)$$

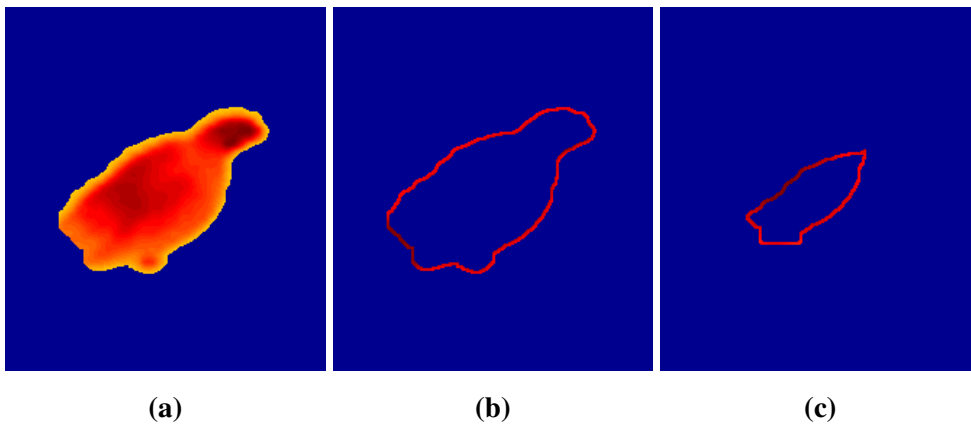
where the constant  $c$  ensures that only objects that are significantly warmer than the background are detected.

### 2.3. Feature Extraction: Thermal Signatures

We propose a novel feature, extracted from the thermal images, that is invariant to translation, rotation, scale and, partly, posture.

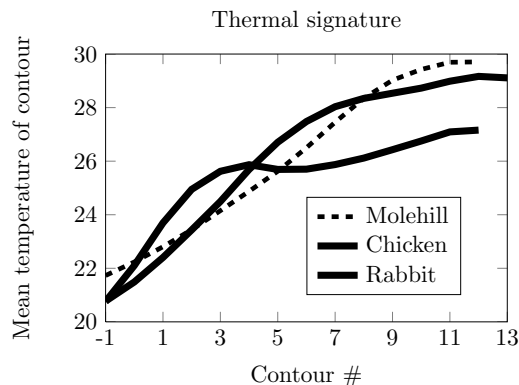
Based on the detected object as in Figure 3a, the perimeter contour is extracted using a four-connected neighborhood structuring element. An example of an extracted contour is shown in Figure 3b. For each iteration, the mean value of the contour is determined, and the object is shrunk by the contour. The procedure continues to iterate, until no more contours can be extracted from the object (e.g., in Figure 3b,c, the first and seventh contour are shown).

**Figure 3.** The process of extracting the thermal signature. (a) Thermal image of the detected object; (b) the first contour of the detected object; (c) the seventh contour of the detected object.



The thermal signature of an object is defined as the mean thermal value of the contour in each iteration  $i$  and denoted as  $cm(i)$  for  $i = -1, \dots, M$ , where  $M$  is the maximum number of iterations possible for the given object. The first iteration is defined as  $i = -1$ , as the object is initially dilated once to get edge information just outside the object. In Figure 4,  $cm(i)$  is shown for different objects. In our dataset, a typical animal signature has a greater temperature increase close to the object boundary than a non-animal object.

**Figure 4.** Thermal signatures extracted from shrinking thermal contours at a height of 4.9 m. Contour number  $-1$  is not part of the object, but used for edge feature extraction.



### 2.3.1. Parameterization of Thermal Signatures

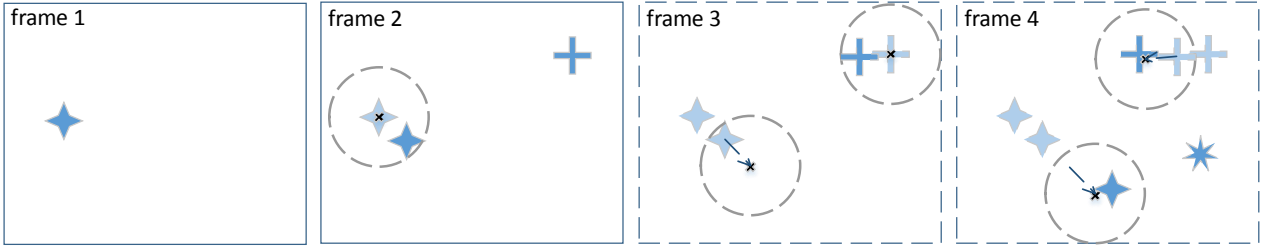
The thermal signature describes certain characteristics of the objects. The signature is normalized by subtracting it with the mean temperature of the first contour. To make it invariant to the maximum number of contours, the signature can be approximated by resampling or by matching it to a high order polynomial. However, as the signature has sinusoidal characteristics, a Fourier-related transform is applicable. The discrete cosine transform (DCT) is chosen for its sinusoidal basis functions and its decorrelation properties. A fixed number of DCT coefficients will provide an approximation of the thermal signature and a set of features to be used in the classification. These feature vectors are then classified as either animal or non-animal, based on the k-nearest-neighbor (kNN) algorithm, which is briefly described in the next section.

### 2.4. Classification

The kNN algorithm is a supervised learning algorithm, which can be used for both classification and clustering [22]. When used for classification, the algorithm is based on labeled training data. We extract 140 animal-feature vectors and 359 non-animal feature vectors as training data for the kNN classifier. More non-animal data are used, as the non-animal class contains more objects with different thermal characteristics. Thus, more training data is required to model this. Based on empirical experiments, the  $k$  parameter was set to 11, thereby including the nearest 11 training points during classification, which is based on majority voting.

### 2.5. Classification Using Temporal Information

A classification based on only a single frame using, e.g., a kNN classifier, discards the important temporal information provided in a recording. A lightweight tracking algorithm is used to link similarly positioned objects through consecutive images in the recordings. As the experiment has been done with a lift in a controlled setting, the tracking algorithm is not designed to compensate for movements of a potential UAV. To end or start new tracks, each track predicts a region defined as a guess region, where a new object needs to be positioned. An object is added to a track if it is within the guess region. A new track is created if a newly detected hot object is outside the guess region of any current tracks. A track is terminated when it fails to include any new objects for a defined number of frames. The guess region is described by a center point and a radius, where the center point is extended by the movement between the two previous objects included in the specific track. An example of the algorithm is provided in Figure 5, where tracks one, two and three are marked with  $\blacklozenge$ ,  $\blackplus$  and  $\blackstar$ , respectively.

**Figure 5.** Tracking procedure.

- In Frame 1, a single object has been detected inside the frame. As no tracks have been registered, the newly detected point creates the first track,  $\star$ .
- In Frame 2 two objects are detected. One point is within the guess region of the first track and is added to the first track. The second point is outside the guess region, and a new track is created,  $+$ .
- In Frame 3, new points are added to the second track. Notice that a new guess region is predicted by the previous movement, but as no animal has been detected within the guess region, no point is added to the track.
- In Frame 4, three objects are detected. Two points are added to the current two tracks, and the third point creates a new track,  $\ast$ .

Every time an object is being assigned to a certain track, the belief is updated to identify the tracked object as either animal or non-animal. The belief of track  $m$  is defined as the posterior probability and formulated as the probability of a detected element being an animal  $A$  given the newly observed data  $D_n$  in frame  $n$ .

$$Bel_{A,m}(n) = P(A | D_n) = \frac{P(A) \cdot P(D_n | A)}{P(D_n)} \quad (2)$$

The term  $P(D_n)$  describes the evidence of the observed data. The evidence is a scale factor that ensures that the posterior probability sums to one and can be rewritten by using the law of total probability:

$$P(D_n) = P(A) \cdot P(D_n | A) + P(A^c) \cdot P(D_n | A^c) \quad (3)$$

where  $A^c$  defines the non-animal objects. The term  $P(A)$  is the prior probability and describes the belief of an object being an animal before the data  $D_n$  have been observed, also defined as the belief at  $n - 1$ .

$$P(A) = Bel_{A,m}(n - 1) \quad (4)$$

The probability  $P(D_n | A)$  is described as the likelihood and defined as the discriminant function  $g_A(D_n)$  of  $kNN$  given by the ratio of  $k_A$  and  $k$ .

$$P(D_n | A) = g_A(D_n) = \frac{k_A}{k} \quad (5)$$

where  $k_A$  is the number of  $kNN$  samples that are animals, e.g., if  $k_A = 6$  (majority vote), the probability is  $P(D|A) = \frac{6}{11} \approx 0.55$ .

Substituting Equations (3)–(5) into Equation (2) yields an updating scheme for every newly detected object.

$$Bel_{A,m}(n) = \frac{Bel_{A,m}(n-1) \cdot g_A(D_n)}{Bel_{A,m}(n-1) \cdot g_A(D_n) + Bel_{A^c,m}(n-1) \cdot g_{A^c}(D_n)} \quad (6)$$

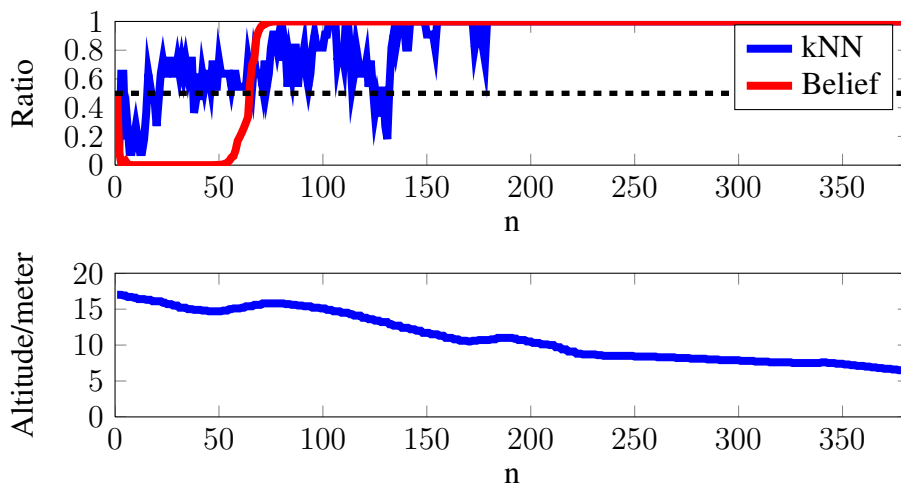
The belief updates every time an object is added to the track, but the track is finally identified as an animal if the belief exceeds 0.5. The prior probability of the first object in a track ( $n = 1$ ) is set to  $Bel_{A,m}(0) = 0.5$ . The belief has a high chance of getting stuck in zero or one if the classifier returns, respectively, zero and one. To avoid this, the classifier will, as a minimum, return 0.05 and maximum 0.95.

The algorithm for tracking objects and building belief is fit for detecting animals in large fields using a UAV. The scenario is as follows: The UAV detects hot objects at high altitudes, thus allowing the UAV to cover large areas in a short time. Due to limited resolution, the detected objects are both small and almost uniform in thermal signature at high altitudes. As presented in the results section, this affects detection and recognition performance.

Therefore, the UAV should approach the objects to increase thermal image quality with respect to classification. By using the tracking algorithm, the belief is constantly calculated. Based on this temporal update of the belief, the algorithm can classify a detected object as an animal or a non-animal.

In Figure 6, the uppermost plot presents the kNN ratio from Equation (5) and the belief from Equation (6), which should be read as  $1 = animal$  and  $0 = non-animal$ . The bottom plot shows the altitude of the recording rig. The example shows how the belief of an object evolves as the altitude decreases. In the example, it is seen that the algorithm believes that the detected object is non-animal. However, as the belief updates, the algorithm discards this when the altitude decreases.

**Figure 6.** Building a belief for decreasing altitudes.



### 3. Results

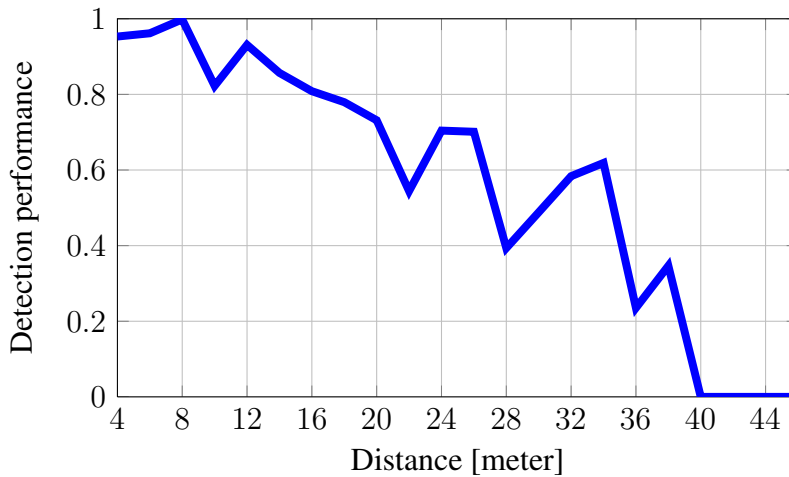
#### 3.1. Detection

Objects are detected using the threshold set by Equation (1). The parameter  $c$  is set to  $c = 5^\circ C$  based on empirical experiments. The detection performance is defined as the ratio between the number of objects detected by the algorithm  $l_{detected}$  and the actual number of animals  $l$  found by manual labeling.

$$D_{performance} = \frac{l_{detected}}{l}$$

Figure 7 shows how the detection performance rapidly degrades until it reaches zero for increasing altitude.

**Figure 7.** Detection performance for animals relative to altitude.



#### 3.2. Feature Extraction and Classification

The thermal signature is approximated using seven DCT coefficients, as this describes 95% of the signature information for more than 95% of the provided data. Figure 8 presents an approximation of the thermal signature for two objects using seven DCT coefficients.

The classification accuracy is a common measure for classifier performance, but as presented in Figure 9a, fewer animals are detected by the segmentation algorithm for increasing altitudes. The loss of detected animals will make the data unbalanced, as it becomes dominated by non-animal samples in high altitudes.

To adjust the unbalanced data, a balanced classification accuracy is used to evaluate the classifier performance:

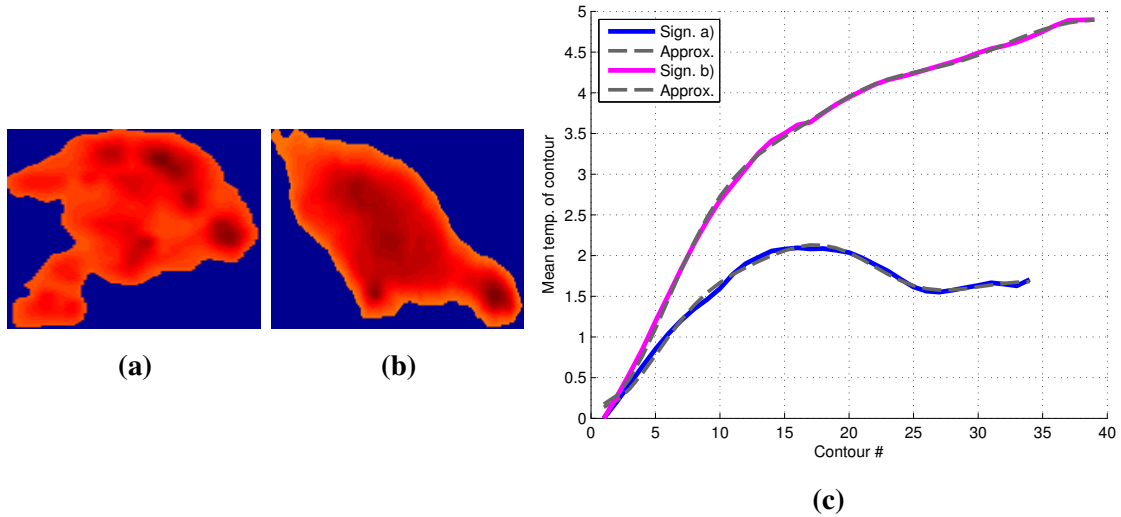
$$C_{accuracy,balanced} = \frac{sensitivity + specificity}{2} = \frac{TP / (TP + FN) + TN / (FP + TN)}{2}$$

where TP, FN, TN and FP are, respectively, true positive, false negative, true negative and false positive. Figure 9b shows the balanced accuracy and how performance degrades for increasing

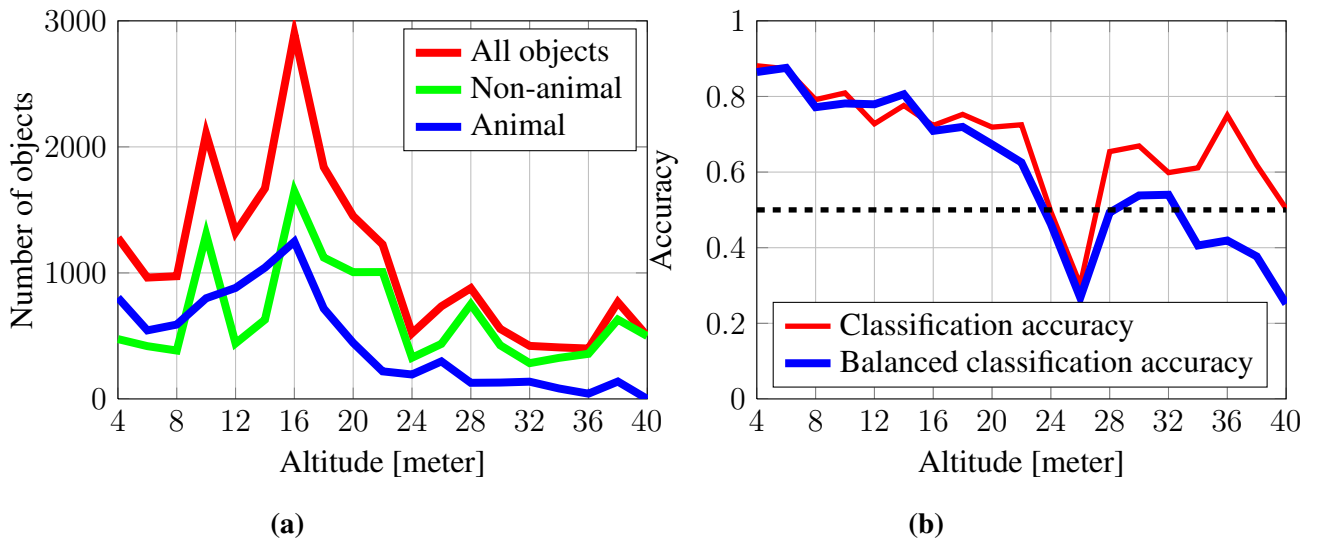


altitudes. The figure also shows that the algorithm is not able to provide satisfactory results for altitudes above 22 m, as the balanced accuracy drops below or around 0.5.

**Figure 8.** Thermal images and approximations of the thermal signature of a rabbit and chicken. (a) Thermal image of a rabbit; (b) thermal image of a chicken; (c) thermal signature and its seven discrete cosine transform coefficient approximation.



**Figure 9.** Evaluation of the classifier relative to altitude. (a) The number of detected objects, animals and non-animals relative to altitude; (b) the classifier performance using classification accuracy and balanced accuracy relative to altitude.



As the segmentation and classification are highly dependent on altitude, the classification is evaluated in the two different altitude ranges of 3–10 m and 10–20 m, defined as, respectively, the short- and far-range altitudes.

### 3.3. Tracking

The tracking algorithm has been setup to allow tracking of an object with three missing points and a maximum uncertainty of 190 pixels. The tracks are identified and labeled as animal or non-animal based on the updating scheme from Equation (6). After a track has been identified, all other objects in the track are changed to the similar label.

#### 3.3.1. Short Range Altitudes (3–10 m)

In the altitude range of 3–10 m, the tracker distributes 4173 out of 4381 objects (95.3%) into tracks containing more than five points, where 4104 out of 4173 objects (98.3%) are placed in a track with the majority of the same label, meaning that 1.7% objects are placed in the wrong track. The balanced classification accuracy is 84.8% before the tracks have been identified, e.g., only kNN classification is performed. Combining the classification results from each frame with the temporal information in terms of tracks, the balanced accuracy is improved by 8.7 percentage points to 93.5%. The confusion matrix before and after tracking is provided in Tables 1 and 2. Table 3 shows different performance measures with and without tracking. Sensitivity or the true positive rate (TPR) describes the classifiers ability to identify an animal object correctly. Specificity or the true negative rate (TNR) describes the classifiers ability to identify a non-animal object correctly. After tracking, the TPR and TNR are 90.8% and 96.2%, respectively, indicating that the classifier has an advantage when classifying non-animal objects.

**Table 1.** Confusion matrix before track identification in the close-range altitudes (3–10 m).

		O b s e r v a t i o n	
		Animal	Non-animal
P r e d i c t i o n	Animal	2056	332
	Non-animal	330	1663

**Table 2.** Confusion matrix after track identification in close-range altitudes (3–10 m).

		O b s e r v a t i o n	
		Animal	Non-animal
P r e d i c t i o n	Animal	2167	76
	Non-animal	219	1919

**Table 3.** Performance measure in close-range altitudes (3–10 m).

Performance measure		No tracking	Tracking
Range 3–10m	Classification accuracy	0.849	0.933
	Balanced classification accuracy	0.848	0.935
	Sensitivity, True positive rate	0.862	0.908
	Specificity, True negative rate	0.834	0.962

### 3.3.2. Far-Range Altitudes (10–20 m)

At an altitude of 10 to 20 m, the tracker distributes 8024 out of 8456 objects (94.9%) into tracks containing more than five points, where 7673 out of 8024 objects (95.6%) are placed in a track with the majority of the same label, meaning that 4.4% are placed in the wrong track.

The balanced classification accuracy is 75.2% before the tracks have been identified, while the balanced accuracy improves by 2.5 percentage points to 77.7%, when tracks are being identified. The confusion matrix before and after tracking is provided in Tables 4 and 5. Table 6 shows different performance measures with and without tracking. After tracking, the TPR and TNR are 63.4% and 90.2%, respectively, indicating that the classifier especially has difficulties classifying animal objects correctly in far-range altitudes.

**Table 4.** Confusion matrix before track identification in far-range altitudes 10–20 m.

		Observation	
		Animal	Non-animal
Prediction	Animal	2735	515
	Non-animal	1606	3600

**Table 5.** Confusion matrix after track identification in far-range altitudes 10–20 m.

		Observation	
		Animal	Non-animal
Prediction	Animal	2753	331
	Non-animal	1588	3784

**Table 6.** Performance measure in far-range altitudes 10–20 m.

Performance measure		No tracking	Tracking
Range 10–20m	Classification accuracy	0.749	0.773
	Balanced classification accuracy	0.752	0.777
	Sensitivity, True positive rate	0.630	0.634
	Specificity, True negative rate	0.875	0.920

The results show that information from consecutive frames in terms of determining and identifying tracks will improve performance by 8.7 and 2.5 percentage points for the short- and far-range altitudes, respectively. The system performs best in close-range altitudes with an accuracy of 93.5%, providing a lead of 15.8 percentage points compared to the far altitude range. The system maintains, though, a low number of FP or a high TNR of, respectively, 96.2% and 92.0% for short and far altitudes, meaning that the system preserves the ability to classify non-animals correctly in both ranges. Conversely, the TPR drops from 90.8% to 63.5%, meaning that the classifier especially has difficulties in recognizing animal objects correctly in far-range altitudes.

#### 4. Discussion

The presented feature extraction and classification scheme shows good detection and classification performance for recording heights under 10 m with a balanced classification accuracy of 84.8%. In the altitude range of 10–20 m, the performance drops, having a balanced classification accuracy of 75.2%. The procedure becomes unfit for altitudes above 20–22 m, as detection performance decreases, but the altitude limit is ultimately set by a bad recognition, as the balanced classification accuracy drops below or around 0.5. Multiple arguments demonstrate that the application degrades for increasing altitudes, ultimately making it unfit for detecting and classifying small animals in altitudes above 20 m. The decreased detection relative to altitude is explained by the following reasons:

- (1) The thermal radiation received by the sensor decreases as the distance to animal increases.
- (2) The size of an animal is decreased for increasing altitudes, allowing the animal to be dominated by its colder surroundings.
- (3) For a given image resolution and FOV, the spatial resolution or ground sample distance will, above a certain altitude, exceed the size of the animal, making it undetectable for the thermal imaging sensor.

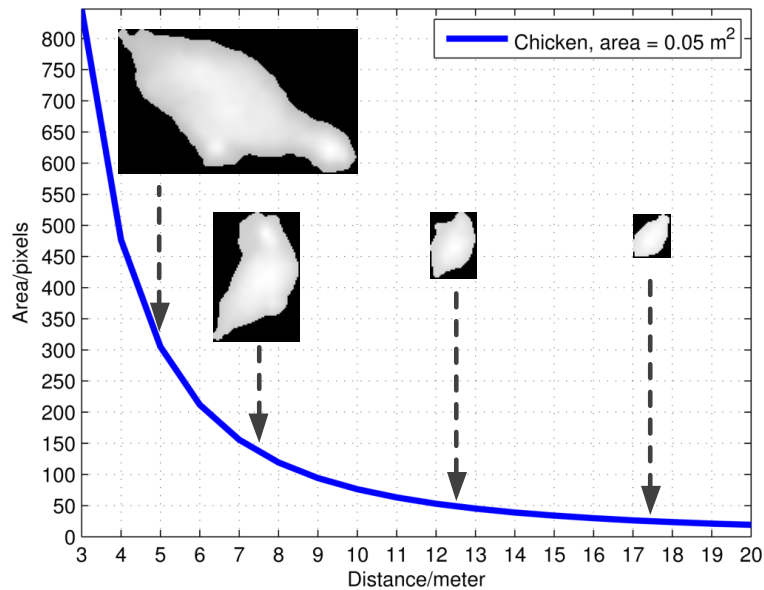
The drop in classifier performance for increasing altitudes is explained by the increasing ground sample distance, causing the object to be presented in lower resolution or by less information, e.g., the area of a chicken (around 0.05 m<sup>2</sup>) will, from an altitude of 5 m, theoretically be presented by 305 pixels, while the same chicken is presented by only 19 pixels from an altitude of 20 m. Figure 10 shows how the pixel area of a chicken theoretically decreases relative to altitude and how a chicken, in practice, ends up losing characteristics.

Performance can, though, be improved in high altitudes for both detection and classification by using a higher resolution camera, a more narrow FOV or optical zoom. The decrease in performance for increasing altitudes fits well with observations from [5], where the authors were able to manually detect roe deer fawns at 30 m, but had problems at 50 m with a thermal camera with a resolution of 640 × 512 pixels. The animals used in this paper are smaller than roe deer fawns, which results in fewer thermal pixels, compared to the roe deer fawns.

Tracking objects in subsequent images enables us to exploit the temporal information in the recording and improve performance. The proposed tracking algorithm improves the balanced

accuracy by 8.7 percentage points to 93.5% in short-range altitudes and by 2.5 percentage points to 77.7% in far-range altitudes. A lightweight tracking algorithm has been applied to simply prove how performance can be improved by exploiting the temporal data. Tracking should, in a real application, handle larger movements in the horizontal plane and could be combined with a gimbal to stabilize the camera, independent of yaw, roll and pitch.

**Figure 10.** The pixel area relative to the distance for a ground area of  $0.05 \text{ m}^2$ . Thermal images of a rescaled chicken from different altitudes.



The manually-extracted training data is based on two types of animals: rabbits and chickens. However, other animals are of interest within the scope of wildlife-friendly agriculture. More experiments, including different weather conditions, vegetation, animals and more non-animal candidates to extend the variation of our somewhat limited dataset, should be conducted. These experiments could help improve the existing algorithm or increase our knowledge of using thermal cameras for automatic detection and recognition of wildlife. Furthermore, the applicability of the used methods should be evaluated using footage taken from an actual UAV in motion to include the effects of wind, UAV movements, moving animals and to more easily extend the variety of the dataset.

The set used for the testing and training of the classifier has no overlapping data. However, as the training data have been selected from, e.g., every 50th frame in a recording, the data used for testing and training are correlated to some extent.

This paper focuses on thermal imaging and the proposed feature extraction method. However, sensor fusion, using the RGB camera, could potentially increase classification performance. Therefore, sensor fusion methods should be investigated to accomplish this.

## 5. Conclusion

We have introduced a method for the automatic detection and recognition of wildlife using thermal cameras for UAV technology. Based on a dynamic threshold, hot objects are detected and

subsequent feature extraction is performed. The novel feature extraction method, presented in this paper, consist of an extraction of thermal signatures for each detected object and a parameterization of this based on DCT.

Methods for classification using measurements from both single and multiple frames is presented. Combining measurements from multiple frames achieves the best performance, with a balanced classification accuracy of 93.5% in the altitude range of 3–10 m and 77.7% in the altitude range of 10–20 m, thus demonstrating a clear relationship between the performance of detection and classification relative to altitude. The simulated and limited dataset is favorable in terms of performance for the given algorithms. The actual applicability of the system should therefore be determined using footage from an actual UAV. The proposed detection and classification scheme is based on top-view images of wildlife, as seen by a UAV. The use of UAV-technology for automatic detection and recognition of wildlife is currently part of ongoing research towards wildlife-friendly agriculture.

### Author Contributions

Peter Christiansen and Kim Arild Steen have made substantial contributions in the development of the algorithms presented in this paper. Henrik Karstoft has made a substantial contribution in manuscript preparation. Rasmus Nyholm Jørgensen has made a contribution to the definition of the research and data acquisition.

### Conflicts of Interest

The authors declare no conflict of interest.

### References

1. Green, C. *Reducing Mortality of Grassland Wildlife during Haying and Wheat-Harvesting Operations*; Oklahoma State University Forestry Publications: Stillwater, OK, USA, 1998; pp. 1–4.
2. Jarnemo, A. Roe deer *Capreolus capreolus* fawns and mowing-mortality rates and countermeasures. *Wildl. Biol.* **2002**, *8*, 211–218.
3. Haschberger, P.; Bundschuh, M.; Tank, V. Infrared sensor for the detection and protection of wildlife. *Opt. Eng.* **1996**, *35*, 882–889.
4. Israel, M.; Schlagenhauf, G.; Fackelmeier, A.; Haschberger, P.; Oberpfaffenhofen, D.; GmbH, C.S.; MÄijnchen, T. Study on Wildlife Detection During Pasture Mowing. 2011. Available online: <http://elib.dlr.de/65977/1/WildretterVDIv4.pdf> (accessed on 26 February 2014); p. 6.
5. Israel, M. A UAV-based roe deer fawn detection system. In Proceedings of the International Conference on Unmanned Aerial Vehicle in Geomatics, Zurich, Switzerland, 14 September 2011; pp. 1–5.
6. Gade, R.; Moeslund, T.B. Thermal cameras and applications: A survey. *Mach. Vision Appl.* **2013**, *25*, 1–18.

7. Davis, J.W.; Sharma, V. Robust background-subtraction for person detection in thermal imagery. In Proceedings of the IEEE International Workshop on Object Tracking and Classification Beyond the Visible Spectrum, New York, NY, USA, 22 June 2004.
8. Davis, J.W.; Keck, M.A. A Two-Stage Template Approach to Person Detection in Thermal Imagery. In Proceedings of the Workshop Applications of Computer Vision, Breckenridge, CO, USA, 5–7 January 2005; pp. 364–369.
9. Goubet, E.; Katz, J.; Porikli, F. Pedestrian tracking using thermal infrared imaging. In Proceedings of the SPIE Conference Infrared Technology and Applications, Cambridge, UK, 15 May 2006; pp. 797–808.
10. Leykin, A.; Ran, Y.; Hammoud, R. Thermal-visible video fusion for moving target tracking and pedestrian classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR'07, Minneapolis, MA, USA, 17–22 June 2007; pp. 1–8.
11. Torresan, H.; Turgeon, B.; Ibarra-Castanedo, C.; Hebert, P.; Maldague, X.P. Advanced surveillance systems: Combining video and thermal imagery for pedestrian detection. Defense and Security. International Society for Optics and Photonics, 2004; pp. 506–515.
12. Cielniak, G.; Duckett, T. People recognition by mobile robots. *J. Intell. Fuzzy Syst.* **2004**, *15*, 21–27.
13. Li, J.; Gong, W.; Li, W.; Liu, X. Robust pedestrian detection in thermal infrared imagery using the wavelet transform. *Infrared Phys. Technol.* **2010**, *53*, 267–273.
14. Fernández-Caballero, A.; López, M.T.; Serrano-Cuerda, J. Thermal-Infrared Pedestrian ROI Extraction through Thermal and Motion Information Fusion. *Sensors* **2014**, *14*, 6666–6676.
15. Rudol, P.; Doherty, P. Human body detection and geolocalization for UAV search and rescue missions using color and thermal imagery. In Proceedings of the 2008 IEEE Aerospace Conference, Big Sky, MT, USA, 1–8 March 2008; pp. 1–8.
16. Nanda, H.; Davis, L. Probabilistic template based pedestrian detection in infrared videos. In Proceedings of the IEEE Intelligent Vehicle Symposium, Dearborn, MI, USA, 17–21 June 2002; Volume 1, pp. 15–20.
17. Cerra, D.; Israel, M.; Datcu, M. Parameter-free clustering: Application to fawns detection. In Proceedings of the 2009 IEEE International, IGARSS, Geoscience and Remote Sensing Symposium, Cape Town, South Africa, 12–17 July 2009; Volume 3, p. III-467.
18. Israel, M.; Evers, S.; für Luft, D.Z.; Raumfahrt, O. Mustererkennung zur Detektion von Rehkitzen in Thermal-Bildern. 2011; pp. 20–28. (In German)
19. Steen, K.A.; Villa-Henriksen, A.; Therkildsen, O.R.; Green, O. Automatic Detection of Animals in Mowing Operations Using Thermal Cameras. *Sensors* **2012**, *12*, 7587–7597.
20. Zhou, D.; Wang, J.; Wang, S. Countour Based HOG Deer Detection in Thermal Images for Traffic Safety. In Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition, Las Vegas, NV, USA, 16–19 July 2012; pp. 1–6.
21. Systems, F.C.V. *Thermal Imaging: How Far Can You See with It?*; Technical Note; FLIR Systems: Wilsonville, OR, USA; Available online: [http://www.flir.com/uploadedfiles/eng\\_01\\_howfar.pdf](http://www.flir.com/uploadedfiles/eng_01_howfar.pdf) (accessed on 17 January 2014); pp. 1–4.
22. Bishop, C.M. *Pattern Recognition and Machine Learning*; Springer: New York, NY, USA, 2006.

# Thermal-Infrared Pedestrian ROI Extraction through Thermal and Motion Information Fusion

Antonio Fernández-Caballero, María T. López and Juan Serrano-Cuerda

**Abstract:** This paper investigates the robustness of a new thermal-infrared pedestrian detection system under different outdoor environmental conditions. In first place the algorithm for pedestrian ROI extraction in thermal-infrared video based on both thermal and motion information is introduced. Then, the evaluation of the proposal is detailed after describing the complete thermal and motion information fusion. In this sense, the environment chosen for evaluation is described, and the twelve test sequences are specified. For each of the sequences captured from a forward-looking infrared FLIR A-320 camera, the paper explains the weather and light conditions under which it was captured. The results allow us to draw firm conclusions about the conditions under which it can be affirmed that it is efficient to use our thermal-infrared proposal to robustly extract human ROIs.

Reprinted from *Sensors*. Cite as: Fernández-Caballero, A.; López, M.T.; Serrano-Cuerda, J. Thermal-Infrared Pedestrian ROI Extraction through Thermal and Motion Information Fusion. *Sensors* **2014**, *14*, 6666–6676.

## 1. Introduction

The detection of pedestrians is a key application in the video surveillance domain [1]. Indeed, a number of surveillance applications require the detection and tracking of people to ensure security and safety [2,3]. The most widespread sensor technology for detecting pedestrians is for sure the use of gray scale [4,5] and color cameras [6,7]. However, using the visible-light information is problematic when facing quick changes in lighting or illumination problems. Now, thermal-infrared images have a number of distinctive features compared to frames acquired by a visible-light spectrum camera [8–11].

In thermal-infrared video, the gray level value of the objects is set by their temperature and radiated heat, and is independent from lighting conditions. The most intuitive idea when performing a pedestrian detection algorithm in the thermal-infrared spectrum is to take advantage of the fact that humans usually appear warmer than other objects in the scene [12,13]. However, this is not always the case [14]. The main reason is that the properties of the objects in the scene (*i.e.*, emissivity, reflectivity and transmissivity) and their wavelength affect the infrared images' intensity, especially in summer afternoon. Obviously, the condition is usually well satisfied during winter and at night. These drawbacks make it impossible to detect humans exclusively using their intensity value. On the other hand, a great amount of infrared images have low spatial resolution and lower sensitivity than visible spectrum images due to the technological limitations of thermal-infrared cameras. These defects often result in low image quality and a great amount of image noise.

Many approaches in this spectrum combine appearance and shape properties since humans are initially detected according to the former (their appearance is usually brighter than other objects in the scene) and are filtered and classified based on the latter [15]. This paper introduces a new



algorithm for robust ROI extraction of pedestrians in thermal-infrared video based on the authors' previous works [16,17]. In addition to presenting the algorithm, the main objective of this article is to draw firm conclusions about the environmental conditions under which it can be affirmed that it is efficient to use thermal-infrared cameras to robustly detect pedestrians.

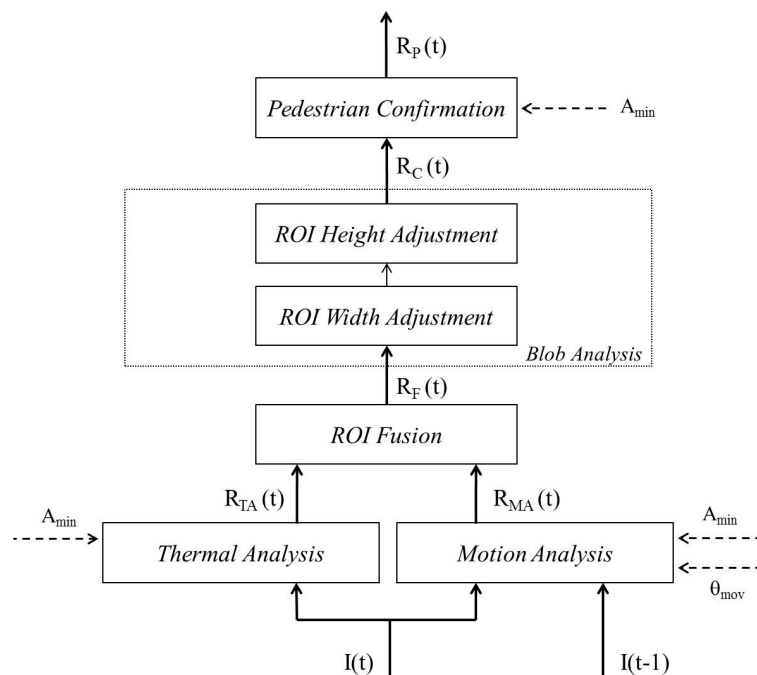
The rest of the article is organized as follows: Section 2 describes the new algorithm for pedestrian ROI extraction in the thermal-infrared spectrum. In Section 3 the algorithm is applied to twelve different video sequences recorded under very different environmental conditions. This way it is possible to determine which the suited ambient conditions are for using a thermal-infrared sensor in the proposed monitoring task. Finally, some conclusions are provided in Section 4.

## 2. Pedestrian ROI Extraction in Thermal-Infrared Video

As previously explained, the infrared spectrum has many interesting features which can be exploited for robust human detection. Two of these properties are clearly important: (1) the independence of lighting conditions of the scene, and specially, (2) the fact that humans tend to be clearly highlighted respect to the background of the picture. Usually, humans' heads also appear hotter than the rest of the body covered with clothes. This is why a *Thermal Analysis* is developed using these properties on each single frame of the video, that is, the current image frame,  $I(t)$ .

In parallel, motion information between the current frame  $I(t)$  and the previous frame  $I(t-1)$  is performed under *Motion Analysis*. A visual representation of the approach is provided in Figure 1. Notice that the results of *Thermal Analysis* and *Motion Analysis* are fused (*ROI Fusion*) to take advantage of both thermal and motion information provided in the video sequence. *Blob Analysis* validates if a given blob corresponding to a supposed pedestrian contains one or more than one human. Lastly, *Pedestrian Confirmation* validates that a refined blob actually contains a valid pedestrian.

**Figure 1.** Algorithm for pedestrian ROI extraction in thermal-infrared video.



## 2.1. Thermal Analysis

A pedestrian ROI extraction based on thermal information is developed in the thermal-infrared spectrum using the properties already mentioned [15]. Pedestrian candidates are extracted in each image frame, solely based on their thermal properties. A set of restrictions on size and shape are applied on the adjusted candidates to eliminate potential false positives. Each one of the stages is now explained in more detail.

The algorithm starts with the analysis of input image,  $I(t)$ , captured at time  $t$ . Image  $I$  is binarised in accordance with a threshold with the aim of isolating the spots related to the pedestrian candidates. This threshold obtains the image areas containing moderate heat blobs, thus probably belonging to pedestrians (pedestrian candidates). This way, warmer zones of the image are isolated where humans could be present. The threshold  $\theta_{TA}$  is calculated in function of the mean ( $\bar{I}$ ) and the standard deviation ( $\sigma_I$ ) of image  $I$ , as shown in Equation (1):

$$\theta_{TA} = \frac{5}{4}(\bar{I} + \sigma_I) \quad (1)$$

Next, the algorithm performs morphological opening and closing operations to eliminate isolated pixels and to unite areas split during the binarization into image blobs. A minimum area,  $A_{\min}$ —function through triangulation of the distance of the camera to the farthest objective—is established for a blob to be considered to contain one or more humans. The output of *Thermal Analysis* towards *ROI Fusion* is a list of regions of interest (ROIs) denominated  $R_{TA}(t)$ .

## 2.2. Motion Analysis

We have previously explained that certain environmental conditions affect negatively the visual contrast in the thermal-infrared spectrum. For example, humans are very hard to find in warm environments where the scene temperature is similar to people's temperature. Yet, if using the motion information in the scene, we can find humans in it since they do not tend to be static during long periods of time. Therefore, *Motion Analysis* is developed to take advantage of the motion information in the scene.

Here, the previous image,  $I(t-1)$ , and the current one,  $I(t)$ , are used. Notice that images are captured a frame rate of 5 images per second, which ensures enough movement and enables processing all the image frames in real-time. An image subtraction and thresholding is performed on these frames. The threshold is experimentally fixed to 16% of the maximum value of a 256 gray levels image; thus, threshold  $\theta_{mov}$  takes the value 16. It is calculated that a pixel  $(x,y)$  is “warm” if:

$$|I(x, y, t) - I(x, y, t - 1)| > \theta_{mov} \quad (2)$$

Now, ROIs with area superior to  $A_{\min}$  and with a percentage of “warm” pixels greater than a rate threshold (experimentally fixed to 5% of the area of the ROI) are extracted into list  $R_{MA}(t)$ .

### 2.3. ROI Fusion

The objective of *ROI Fusion* is to sum up or overlap the ROIs coming from *Thermal Analysis* and *Motion Analysis* to get a unique list of regions of interest  $R_F(t)$ . We are faced with three possibilities:

- (1) A ROI belonging to list  $R_{TA}(t)$  has no common pixel with any ROI belonging to  $R_{MA}(t)$ : the ROI from  $R_{TA}(t)$  is included as is in the new list of ROIs called  $R_F(t)$ .
- (2) A ROI belonging to list  $R_{MA}(t)$  has no common pixel with any ROI belonging to  $R_{TA}(t)$ : the ROI from  $R_{MA}(t)$  is included as is in the new list of ROIs called  $R_F(t)$ .
- (3) A ROI belonging to list  $R_{TA}(t)$  has some common pixels with a given ROI belonging to  $R_{MA}(t)$ : the ROIs from  $R_{TA}(t)$  and  $R_{MA}(t)$  compose a new ROI containing all pixels from the previous ones; this new ROI is included in the new list of ROIs called  $R_F(t)$ .

Rules (1) and (2) show the possibilities to sum up the ROIs coming from both Thermal Analysis and Motion Analysis. Rule (3) demonstrates the case when both Thermal Analysis and Motion Analysis have detected the same candidates as pedestrians (or at least part of them).

### 2.4. Blob Analysis

This part of the algorithm works with the list  $R_F(t)$ . This list was obtained at the end of the previous section. At this point, there is a need to validate the content of each ROI to find out if it contains one single human candidate or more than one. Therefore, each detected ROI is individually processed.

#### 2.4.1. ROI Width Adjustment

The first step of *Blob Analysis* consists in scanning  $R_F$  by columns, adding the gray level value corresponding to each pixel in that column. This way, a histogram  $H[i]$  is obtained (see Equation (3)), which shows the zones of the current ROI that contain greater heat concentrations:

$$H[i] = \sum_j R_F(i, j), \forall i \quad (3)$$

A double purpose is pursued when computing the histogram. In first place, we want to increase the certainty of the presence of human heads. Secondly, as a ROI may contain several persons that are close enough to each other, the histogram helps separating human groups (if any) into single humans. This method, when looking for maxima and minima within the histogram allows differentiating among the people actually present in a particular ROI.

So, the histogram  $H[i]$  is scanned to separate grouped humans, if they exist in that ROI. Local maxima and local minima are searched in the histogram to establish the different heat sources with this purpose. To assess whether a histogram column contains a local maximum or minimum, a new threshold is fixed. We are looking for columns where the 60% of their pixels are below the mean gray value of  $R_F$ , since those regions are supposed to belong to gaps between two humans.

This way the list  $R_F$  will form a new list of sub-ROIs  $sR_F(t)$ . Notice that if each  $R_F$  contains a single human,  $sR_F(t)$  will be equivalent to  $R_F(t)$ .

#### 2.4.2. ROI Height Adjustment

All humans contained in a given sub-ROI of list  $sR_F(t)$ , obtained in the previous section, still possess the same height, namely the height of the original ROI. Now, we want to fit the height of each sub-ROI to the real height of the humans contained in it. For this purpose row adjustment is performed. The calculation is done separately on each sub-ROI to avoid the influence of the rest of image pixels on the result. This threshold uses the value of the sub-ROI mean gray level. Each sub-ROI is binarised in order to delimit its upper and lower limits. After this, a closing operation is performed to unite spots isolated in the binarisation. The newly obtained ROIs are now enlisted into  $R_C(t)$ .

#### 2.5. Pedestrian Confirmation

Now a final stage is needed for each ROI of list  $R_C(t)$  to confirm if the human candidate is actually a human. Indeed, some incandescent spots in an image (such as light bulbs or big heat sources in general) can still be confused under certain circumstances with humans due to their heat properties. So an important step consists in verifying if one of these spots is being scanned instead of a human.

For this sake, firstly the human candidate's ROI dimensions are checked. The first check consists in testing the ROI's height/width ratio. If the human candidate's width is larger than its height, the standard deviation of the brightness of the ROI is checked. This is due to the fact that incandescent spots such as lamps or fuses have a low standard deviation since their heat distribution is uniform. On the contrary, humans have different heat concentrations in their body parts, such as the head being warmer than the rest of the body. We have determined experimentally that the standard deviation of the human ROI has to be greater than 12.

The human candidate's area is also required to be above a minimum area  $A_{\min}$  experimentally fixed according to features such as the camera height or the extension of the scenario. Finally, the final list of ROIs containing humans is the output of the people detection algorithm, that is,  $R_P(t)$ .

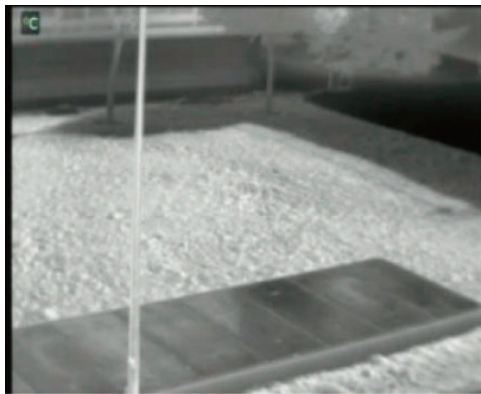
### 3. Results and Discussion

#### 3.1. Test Environment

The selected test environment is an outdoor scenario where a forward-looking infrared FLIR A-320 camera has been placed 6 meters above the ground level. The decision to use an outdoor environment is due to the fact that this kind of scenario offers a greater number of variations in temperature and lighting conditions, whereas an indoor environment is usually more controlled. The scenario does not have any predefined access, so that a pedestrian enters into the scene from the lower limits as well as at the left or right sides of the image. A platform constructed of concrete is located in the lower part of the scene. This material quickly absorbs the temperature of the

environment. The same property is also present in the building placed in the scene background. The building shows additional problems for thermal-infrared human detection. The reason is that the thermal-infrared camera automatically performs thermal attenuation, which results in the lack of accuracy in obtaining far objects' temperatures. The attenuation causes the thermal readings of pedestrians to be confused with the temperature of the building, this way hardening their isolation from the scene background. Figure 2 shows an image of the scenario as captured by the FLIR camera.

**Figure 2.** Environment for validating the robustness of the approach.



### 3.2. Test Sequences

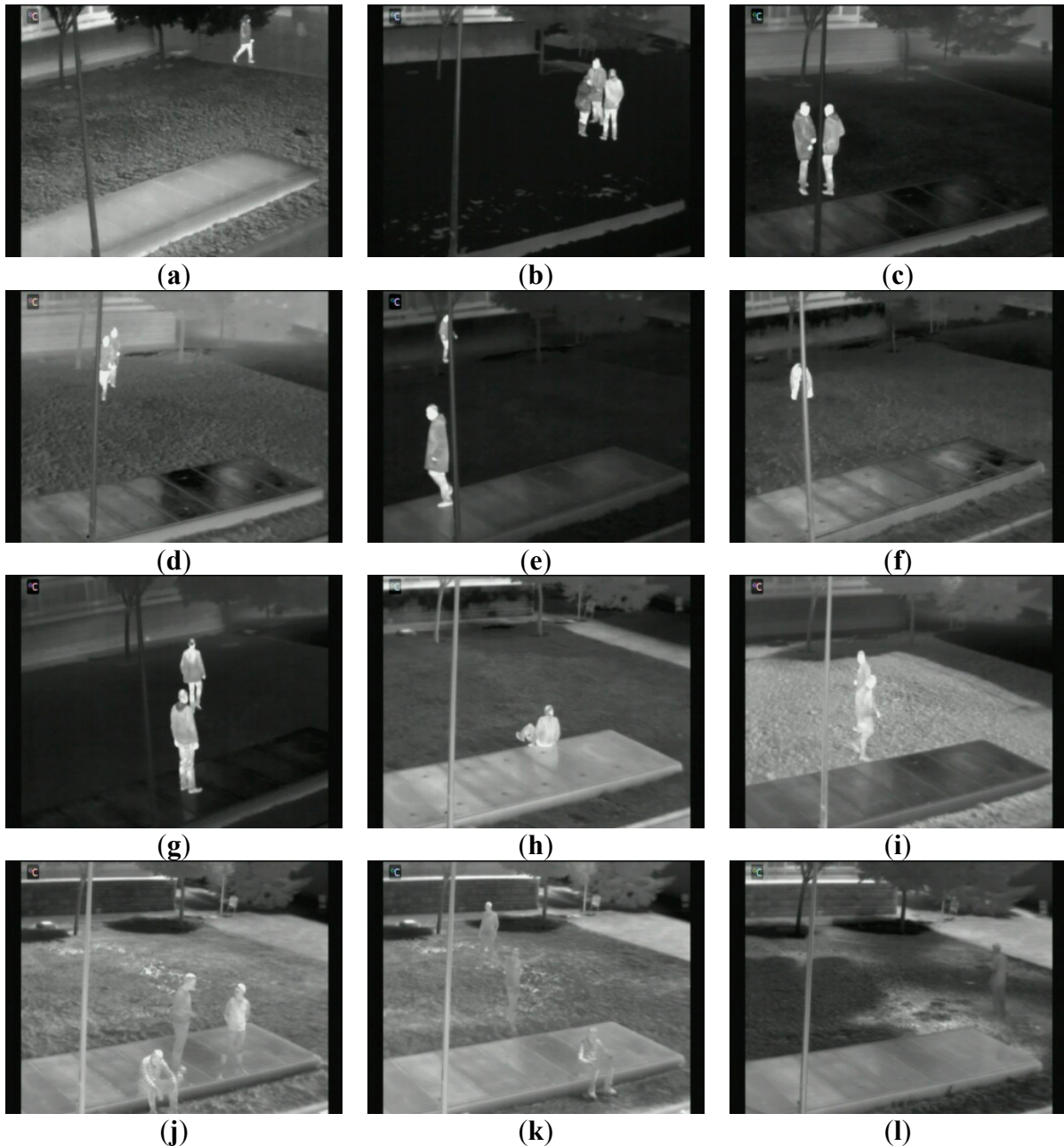
To evaluate our algorithms, we have tested a number of sequences at different temperatures and under different conditions. The main objective is to cover the maximum possible number of situations, both in complexity and variation of temperature. To do this, it was decided to include a range of winter and summer temperatures, ranging between  $-2^{\circ}$  and  $33^{\circ}$ . We have also sought to work under different weather conditions from snow to sunshine. In addition, we used situations of varying complexity, from a single human walking on the scene up to three people meeting, with various actions that pedestrians can perform on an exterior scene. These actions range from attitudes in which humans are easy to detect such as walking or running to other more difficult, because people change the proportions the space they occupy, such as bending, sitting or even lying on the floor. Next, the different recorded sequences are described. Each of these twelve sequences is referred to by the temperature at which it was captured, followed by the atmospheric conditions at the time of the recording.

- Sequence  $-2^{\circ}$ *Foggy* features a human in the scenario (see Figure 3a). The pedestrian is mostly walking, but also performs actions such as crouching, running or sitting in the central concrete platform. The sequence was recorded in a moment where fog was partially covering the scene. It is not difficult to distinguish humans in the thermal-infrared spectrum, except when they are approach the building.
- Sequence  $2^{\circ}$ *Snowy* was recorded after a snowfall, and therefore all the ground appears covered by snow (see Figure 3b). Behaviors within the sequence have a high complexity. During the course of the sequence three human repeatedly appear together (so that the

algorithm has difficulty to separate them, as they often occlude each other). Various activities such as running, walking, bending or dropping items on the floor are made.

- Sequence 3°*Sunny* (see Figure 3c) starts with a human walking in the environment. Sometimes, he/she carries out different actions such as crouching. Later, a second human is walking in different trajectories. Finally, both humans cross their paths, meeting on the concrete platform.
- Another sequence named 8°*Night* was recorded to evaluate the performance of the approach under night conditions (see Figure 3d). The thermal-infrared spectrum introduces a number of problems. Indeed, buildings in the environment are still warm due to the heat accumulated during the day hours. Thus, the buildings are sometimes confused with humans walking in front of them. The sequence features two people walking in the scenario, occasionally crossing their paths.
- Sequence 9°*Cloudy* was captured on a cloudy day (see Figure 3e), and in it, two people follow random paths across the stage. In the thermal-infrared spectrum humans remain easily distinguishable from the rest of the environment.
- Now, sequence 10°*Cloudy* presents a simpler version of the above sequence, with one person walking across the stage and performing various actions such as bending and strolling along the worst lit areas of the stage, as are the shadows of the trees (see Figure 3f).
- Sequence 15°*Dawning* was filmed at sunrise (see Figure 3g). During the scene, gradual changes in illumination and temperature are recorded, starting with the very dim lighting and increasing as the sequence advances. In the sequence two pedestrians continuously gather and meet, so that there are many occlusions.
- In the sequence 15°*Cloudy* some more complex actions are performed by a single human, such as sitting in the central platform (see Figure 3h). The temperature rise causes the apparition of human reflections on the concrete platform, this way augmenting the difficulty for human detection in the infrared spectrum.
- Sequence 18°*Sunny* contains groups of pedestrians (see Figure 3i). There is also the added difficulty that at this temperature the heat of the lawn and the environment in general increases, making it harder to distinguish humans, even to the naked eye in the captured frames in the thermal-infrared.
- Sequence 23°*Sunny* (see Figure 3j) is much more complex than before, because, this time increases to three the number of humans who walk through the scene and gather several times, sitting or simply crossing. Again, the high temperature makes it difficult to distinguish humans in the thermal-infrared spectrum, the area above the concrete platform being especially critical.

**Figure 3.** Example frames of the twelve sequences. (a)  $-2^{\circ}$ Foggy; (b)  $2^{\circ}$ Snowy; (c)  $3^{\circ}$ Sunny; (d)  $8^{\circ}$ Night; (e)  $9^{\circ}$ Cloudy; (f)  $10^{\circ}$ Cloudy; (g)  $15^{\circ}$ Dawning; (h)  $15^{\circ}$ Cloudy; (i)  $18^{\circ}$ Sunny; (j)  $23^{\circ}$ Sunny; (k)  $28^{\circ}$ Sunny; (l)  $33^{\circ}$ Sunny.



- Sequence  $28^{\circ}$ Sunny augments the difficulty of thermal-infrared pedestrian detection with the apparition of up to three pedestrians walking in the scene and performing actions such as sitting, crossing their paths, and meeting. The high temperature makes it quite difficult to distinguish humans in the infrared spectrum, especially on the concrete platform (see Figure 3k).
- Finally, sequence  $33^{\circ}$ Sunny was recorded with much heat. Humans are almost indistinguishable from the background in the thermal-infrared spectrum and appear always cooler than the rest of the environment (see Figure 3l).

### 3.3. Assessment Criteria

Some measures widely used by the computer vision community, such as recall, precision and F-score, were considered to evaluate the performance of the previously described segmentation algorithms. These measures are calculated as shown in Equations (4)–(6), respectively:

$$recall = TP / (TP + FN) \quad (4)$$

$$precision = TP / (TP + FP) \quad (5)$$

$$F - score = recall / (precision + recall) \quad (6)$$

where TP (true positives) is the amount of correct detections in the sequence, FP (false positives) are the mistaken detections gotten and FN (false negatives) is the amount of humans really present in the scene but not detected.

The precision shows the percentage of true positives with respect to the total number of detections, *i.e.*, the probability of detections which really correspond to a human. On the other hand, the recall shows the probability of a human on the scene to be really detected. Finally, F-score is a weighted average, which provides an overall vision of the system performance, considering precision and recall.

### 3.4. ROI Extraction Results

The results obtained are shown in Table 1. The first conclusion to be drawn is quite obvious. In general, the thermal-infrared spectrum is suitable for detecting human under low and medium recorded temperatures. Notice that the sequence captured at 8° shows worse results, as was recorded in the early hours of the night and the temperature had not yet fallen. Under all these thermal conditions the *F-score* is maintained over a good 0.83 value.

However, the performance declines drastically when the temperature of the scene rises above 20°. This is due to the fact that the thermal radiation of humans is very similar to the temperature of the buildings. Indeed, the sun warms the scene directly, affecting the elements of it. This has a significant impact on the final sequence, in which humans are totally “unified” with the environment and the distinction is almost impossible, even for a human observer who is supervising the frames captured in thermal-infrared. Notice that the *recall* value falls down dramatically by only incrementing a few degrees in the ambient temperature. The 33° *Sunny* sequence shows a very bad performance (0.03).

Some other conclusions can also be drawn. These are related with atmospheric environmental conditions. In accordance with the results obtained in Table 1, we can conclude that there is no difference between snowy, cloudy and sunny conditions beneath a given temperature (around 20°). Indeed, the *recall* and the *F-score* are always kept above excellent 0.91 and 0.94 values, respectively. However, notice that the foggy sequence drops the value of *recall* down to 0.71, which is still a good value, but nor comparable to other scores obtained for a similar temperature.



This way we can conclude that pedestrian ROI extraction in the thermal-infrared spectrum provides excellent results for low and medium ambient temperatures, but the results could be affected by some specific weather conditions.

**Table 1.** Results of pedestrian ROI extraction in thermal-infrared.

<b>Sequence</b>	<b>TP</b>	<b>FP</b>	<b>FN</b>	<b>Recall</b>	<b>Precision</b>	<b>F-Score</b>
<i>2°Foggy</i>	11,928	147	4,784	0.71	0.99	0.83
<i>2°Snowy</i>	3,224	163	156	0.95	0.95	0.95
<i>3°Sunny</i>	2,902	295	44	0.98	0.91	0.94
<i>8°Noche</i>	4,787	766	1,112	0.81	0.86	0.83
<i>9°Cloudy</i>	1,618	61	105	0.94	0.96	0.95
<i>10°Cloudy</i>	1,827	12	22	0.99	0.99	0.99
<i>15°Dawning</i>	3,957	12	293	0.93	1.00	0.96
<i>15°Cloudy</i>	1,684	51	160	0.91	0.97	0.94
<i>18°Sunny</i>	2,185	19	176	0.93	0.99	0.96
<i>23°Sunny</i>	2,174	363	1,448	0.60	0.86	0.71
<i>28°Sunny</i>	3,077	160	4,861	0.39	0.96	0.55
<i>33°Sunny</i>	123	23	3,393	0.03	0.84	0.04

#### 4. Conclusions

This article has provided comprehensive information about tests that have been conducted to evaluate the performance of a new algorithm developed for detecting human in thermal-infrared video. The paper has described our thermal-infrared pedestrian ROI extraction algorithm. Then, the evaluation of the proposal has been introduced in detail. The results allowed us to assess the validity of our thermal-infrared proposal to robustly detect pedestrians under varying dynamic outdoor conditions. We have also been able to study under which weather conditions and temperatures the approach is consistent and throws from good up to excellent detection results for videos captured by a forward-looking infrared FLIR A-320 camera.

#### Acknowledgments

This work was partially supported by Spanish Ministerio de Economía y Competitividad/FEDER under TIN2010-20845-C03-01 and TIN2013-47074-C2-1-R grants.

#### Author Contributions

Antonio Fernández-Caballero, María T. López and Juan Serrano-Cuerda have made substantial contributions in the definition of the research line, as well as in experimentation, data analysis, and manuscript preparation.

#### Conflicts of Interest

The authors declare no conflict of interest.

## References

1. Dollár, P.; Wojek, C.; Schiele, B.; Perona, P. Pedestrian detection: An evaluation of the state of the art. *IEEE Trans. Patt. Anal. Mach. Intell.* **2012**, *34*, 743–761.
2. Navarro, E.; Fernández-Caballero, A.; Martínez-Tomás, R. Intelligent multisensory systems in support of information society. *Int. J. Syst. Sci.* **2014**, *45*, 711–713.
3. Costa, D.G.; Guedes, L.A.; Vasques, F.; Portugal, P. Adaptive monitoring relevance in camera networks for critical surveillance applications. *Int. J. Distri. Sens. N.* **2013**, *2013*, 836721:1–836721:14.
4. Enzweiler, M.; Gavrilu, D. Monocular pedestrian detection: Survey and experiments. *IEEE Trans. Patt. Anal. Mach. Intell.* **2009**, *31*, 2179–2195.
5. Fernández-Caballero, A.; López, M.T.; Saiz-Valverde, S. Dynamic stereoscopic selective visual attention (DSSVA): Integrating motion and shape with depth in video segmentation. *Expert Syst. Appl.* **2008**, *34*, 1394–1402.
6. Fernández-Caballero, A.; López, M.T.; Castillo, J.M.; Maldonado-Bascón, S. Real-time accumulative computation motion detectors. *Sensors* **2009**, *9*, 10044–10065.
7. Schwartz, W.; Kembhavi, A.; Harwood, D.; Davis, L. Human detection using partial least squares analysis. In Proceedings of the IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 27 September–4 October 2009; pp. 24–31.
8. Olmeda, D.; de la Escalera, A.; Armingol, J. Far infrared pedestrian detection and tracking for night driving. *Robotica* **2011**, *29*, 495–505.
9. Li, J.; Gong, W.; Li, W.; Liu, X. Robust pedestrian detection in thermal infrared imagery using the wavelet transform. *Infrared Phys. Tech.* **2010**, *53*, 267–273.
10. Kumar, P.; Mittal, A.; Kumar, P. Fusion of thermal infrared and visible spectrum video for robust surveillance. In Proceedings of the Computer Vision, Graphics and Image Processing, Madurai, India, 13–16 December 2006; pp. 528–539.
11. Lamberti, F.; Sanna, A.; Paravati, G. Improving robustness of infrared target tracking algorithms based on template matching. *IEEE Aerosp. Electron. Syst. Mag.* **2011**, *47*, 1467–1480.
12. Wang, J.-T.; Chen, D.-B.; Chen, H.-Y.; Yang, J.-Y. On pedestrian detection and tracking in infrared videos. *Patt. Recog. Lett.* **2012**, *33*, 775–785.
13. Xu, F.; Liu, X.; Fujimura, K. Pedestrian detection and tracking with night vision. *IEEE Trans. Intell. Transp. Syst.* **2005**, *6*, 63–71.
14. Goubet, E.; Katz, J.; Porikli, F. Pedestrian tracking using thermal infrared imaging. In *Infrared Technology and Applications XXXII*; SPIE: Bellingham, WA, USA, 2006; pp. 797–808.
15. Fang, Y.; Yamada, K.; Ninomiya, Y.; Horn, B.; Masaki, I. A shape-independent method for pedestrian detection with far-infrared images. *IEEE Trans. Veh. Technol.* **2004**, *53*, 1679–1697.
16. Fernández-Caballero, A.; Castillo, J.C.; Serrano-Cuerda, J.; Maldonado-Bascón, S. Real-time human segmentation in infrared videos. *Expert Syst. Appl.* **2011**, *38*, 2577–2584.
17. Sokolova, M.V.; Serrano-Cuerda, J.; Castillo, J.C.; Fernández-Caballero, A. Fuzzy model for human fall detection in infrared video. *J. Intell. Fuzzy Syst.* **2013**, *24*, 215–228.

# Thermal Tracking of Sports Players

Rikke Gade and Thomas B. Moeslund

**Abstract:** We present here a real-time tracking algorithm for thermal video from a sports game. Robust detection of people includes routines for handling occlusions and noise before tracking each detected person with a Kalman filter. This online tracking algorithm is compared with a state-of-the-art offline multi-target tracking algorithm. Experiments are performed on a manually annotated 2-minutes video sequence of a real soccer game. The Kalman filter shows a very promising result on this rather challenging sequence with a tracking accuracy above 70% and is superior compared with the offline tracking approach. Furthermore, the combined detection and tracking algorithm runs in real time at 33 fps, even with large image sizes of  $1920 \times 480$  pixels.

Reprinted from *Sensors*. Cite as: Gade, R.; Moeslund, T.B. Thermal Tracking of Sports Players. *Sensors* **2014**, *14*, 13679–13691.

## 1. Introduction

Traditionally, visual cameras, capturing RGB or greyscale images, have been the obvious choice of sensor in surveillance applications. However, in dark environments, this sensor has serious limitations, if capturing anything at all. This is one of the reasons that other types of sensors are now taken into consideration. One of these sensors is the thermal camera, which has recently become available for commercial and academic purposes, although originally developed for military purposes [1]. The light-independent nature of this sensor makes it highly suitable for detection and tracking of people in challenging environments. Privacy has also become a big issue, as the number of surveillance cameras have increased rapidly. For video recording in sensitive locations, thermal imaging might be a good option to cover the identity of the people observed, in some applications it might even be the only legal video modality. However, like any other sensor type, the thermal sensor has both strengths and weaknesses, which are discussed in the survey on thermal cameras and applications [1]. One way of overcoming some of these limitations is to combine different sensors in a multi-modal system [2].

The visual and thermal sensors complement each other very well. Temperature and colour information are independent, and besides adding extra information on the scene each sensor might be able to detect targets in situations where the other sensor completely fails. However, registration and fusion of the two image modalities can be challenging, since there is not necessarily any relation between brightness level in the different spectra. Generally, three types of fusion algorithms exist; fusion on pixel level, feature level, or decision level. Several proposed fusion algorithms are summarised in the survey [1].

It is clear that multi-modal detection and tracking systems have several advantages for robust performance in changing environments, which is also shown in recent papers on tracking using thermal-visible sensors [3,4]. The drawbacks of these fused systems primarily relates to the fusion part, which requires an additional fusion algorithm that might be expensive in time and computations.

Furthermore, when applying a visual sensor, the possibility of identification and recognition of people exists, causing privacy issues that must be considered for each application.

A direct comparison of tracking performance in multi-modal images versus purely thermal images in different environments would be interesting, but this is out of scope for this paper. Here we choose to take another step towards privacy-preserving systems and work with thermal data only. While tracking people in RGB and greyscale images has been and is still being extensively researched [5,6], the research in tracking in thermal images is still rather limited. Therefore, in this paper we wish to explore the possibility of applying tracking algorithms in the thermal image modality.

### *1.1. Related Work*

Two distinct types of thermal tracking of humans exist. One is tracking of human faces, which requires high spatial resolution and good quality images to detect and track facial features [7–9]. The other direction, which we will focus on, is tracking of whole-body individual people in surveillance-like settings. In this type of applications the spatial resolution is normally low and the appearance of people is very similar. We cannot rely on having enough unique features for distinguishing people from each other, and we must look for tracking methods using only anonymous position data.

For tracking in traditional RGB or greyscale video, the tracking-by-detection approach has recently become very popular [10–12]. The classifier is either based on a pre-trained model, e.g., a pedestrian model, or it can be a model-free tracker initialised by a single frame, learning the model online. The advantage of online learning is the ability to update the classifier, as the target may change appearance over time. In order to apply this approach for multi-target tracking, the targets should be distinguishable from each other. This is a general problem in thermal images, the appearance information is very sparse, as no colour, texture, *etc.*, are sensed by the camera.

Other approaches focus on constructing trajectories from “anonymous” position detections. Both online (recursive) and offline (batch optimisation) approaches has proven to be successful. Online approaches cover the popular Kalman filter [13] and particle filters [14,15]. The methods are recursive, processing each frame as soon as it is obtained, and assigning the detection to a trajectory. Offline methods often focus on reconstructing the trajectories by optimising an objective function. Examples are presented in [16] by posing the problem as an integer linear program and solving it by LP-relaxation, or in [17] solving it with the k-shortest path algorithm.

Tracking in thermal video has often been applied in real-time applications for pedestrian tracking or people tracking for robot-based systems. Fast online approaches have therefore been preferred, such as the particle filter [18,19] and the Kalman filter [20,21].

While most works on tracking people in thermal images have focused on pedestrians with low velocity and highly predictable motion, we apply tracking to real sports video, captured in a public sports arena. It is highly desired to track sports players in order to analyse the activities and performance of both teams and individuals, as well as provide statistics for both internal and commercial use. However, sports video is particularly challenging due to a high degree of physical interaction, as well as abrupt and erratic motion.

Figure 1 shows an example frame from the video used for testing. The video is captured with three cameras in order to cover the entire field of  $20\text{ m} \times 40\text{ m}$ . The images are rectified and stitched per frame to images of  $1920 \times 480$  pixels.

**Figure 1.** Example of a frame from the thermal sports video.

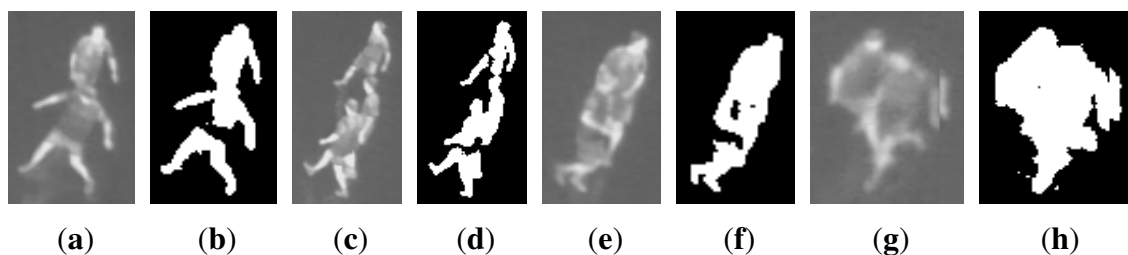


This paper will investigate the applicability and performance of two different tracking approaches on thermal data. First, we design an algorithm based on the Kalman filter. Then, we test a publicly available state-of-the-art multi-target tracking algorithm [22]. The algorithms are evaluated on a 2 min manually annotated dataset from an indoor soccer game.

## 2. Detection

Detecting people in thermal images may seem simple, due to an often higher temperature of people compared with the surroundings. In this work we focus on indoor environments, more specifically a sports arena. This scene is quite simple in terms of a plain background with relatively stable temperature. Hence, people can often be segmented from the background by only thresholding the image. The challenges occur in the process of converting the binary foreground objects into individual people. In the ideal cases each blob is simply considered as one person. However, when people interact with each other, they overlap in the image and cause occlusions, resulting in blobs containing more than one person. The appearance of people in thermal images is most often as simple as grey blobs, making it impossible to robustly find the outline of individual people in overlaps. Figure 2 shows four examples of occlusions and the corresponding binarised images.

**Figure 2.** Examples of occlusions between people. For each example the corresponding binarised image is shown, found by automatic thresholding.



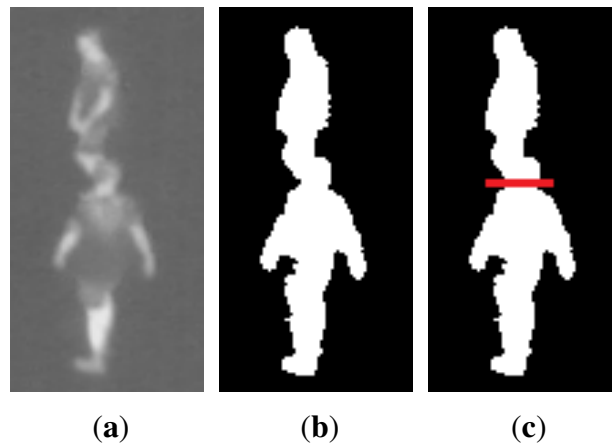
While full or severe occlusions (like Figure 2e) cannot be solved by detection on frame basis, we aim to solve situations where people are only partly occluded and can be split into single

person. Likewise, we want to detect only one person even when it has been split into several blobs during thresholding. We implement three rather simple but effective routines aiming at splitting or connecting the blobs into single person. These routines are described in the following sections.

### 2.1. Split Tall Blobs

People standing behind each other, seen from the camera, might be detected as one blob containing more than one person. In order to split these blobs into single detection we here adapt the method from [23]. First, it must be detected when the blob is too tall to contain only one person. If the blob has a pixel height that corresponds to more than a maximum height at the given position, found by an initialising calibration, the algorithm should try to split the blob horizontally. The point to split from is found by analysing the convex hull and finding the convexity defects of the blob. Of all the defect points, the point with the largest depth and a given maximum absolute gradient should be selected, meaning that only defects coming from the side will be considered, discarding, e.g., a point between the legs. Figure 3 shows an example of how a tall blob containing two people will be split.

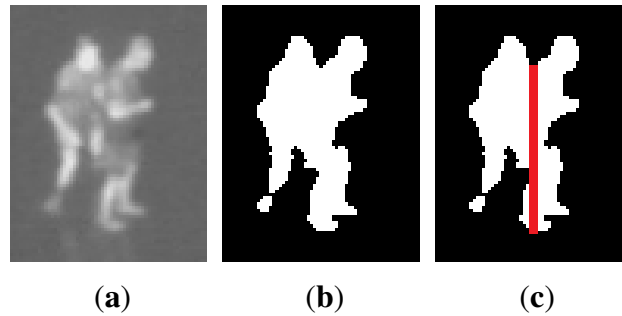
**Figure 3.** Example of how a tall blob containing two people will be split into two.



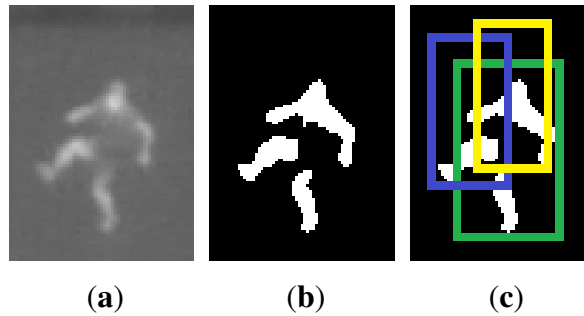
### 2.2. Split Wide Blobs

Groups of people standing next to each other might be found as one large blob. To identify which blobs contain more than one person, the height/width ratio and the perimeter are considered, as done in [23]. If the criteria are satisfied, the algorithm should try to split the blob. For this type of occlusion, it is often possible to see the head of each person, and split the blob based on the head positions. Since the head is narrower than the body, people can be separated by splitting vertically from the minimum points of the upper edge of a blob. These points can be found by analysing the convex hull and finding the convexity defects of the blob. Figure 4 shows an example of how a wide blob containing two people will be split.

**Figure 4.** Example of how a wide blob containing two people will be split into two.



**Figure 5.** Example of a person that is split into three blobs by thresholding. Three overlapping candidates are evaluated (green, blue and yellow rectangles). Only the green candidate will be kept, because it has the highest ratio of white pixels.



### 2.3. Connect Blobs

One person can often be split into several blobs during thresholding if some areas of the body appear colder, e.g., due to loose or several layers of clothing. In order to merge these parts into only one detected person, we consider each binary blob a candidate, and generate a rectangle of standard height at the given position (calculated during calibration) and the width being one third of the height. For each rectangle we evaluate the ratio of foreground (white) pixels. If the ratio of white pixels is below 15%, the blob is discarded, otherwise the candidate is added for further processing. The second step is to check if the candidate rectangles overlap significantly, hence probably belonging to the same person. If two rectangles overlap by more than 45%, only the candidate with highest ratio of white pixels is kept as a true detection. These threshold values are chosen experimentally by evaluating 340 positive samples and 250 negative samples. Figure 5 illustrates this situation, where one person has been split into three blobs.

The ultimate goal for the detection algorithm is to detect each person, and nothing else, in each frame. However, with a side-view camera angle and a number of people interacting, missing detections and noise must be considered when using the detections as input for the tracking algorithms described next.

### 3. Tracking

#### 3.1. Kalman Filter

The Kalman filter, introduced in the early 1960s, is a now well-known algorithm used in a wide range of signal processing applications. The recursive algorithm filters noisy measurements by predicting the next step from previous state and use the new measurement as feedback for updating the estimate. The Kalman filter estimates the state  $x$  of a discrete-time controlled process controlled by the linear stochastic difference equation [24]:

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1} \quad (1)$$

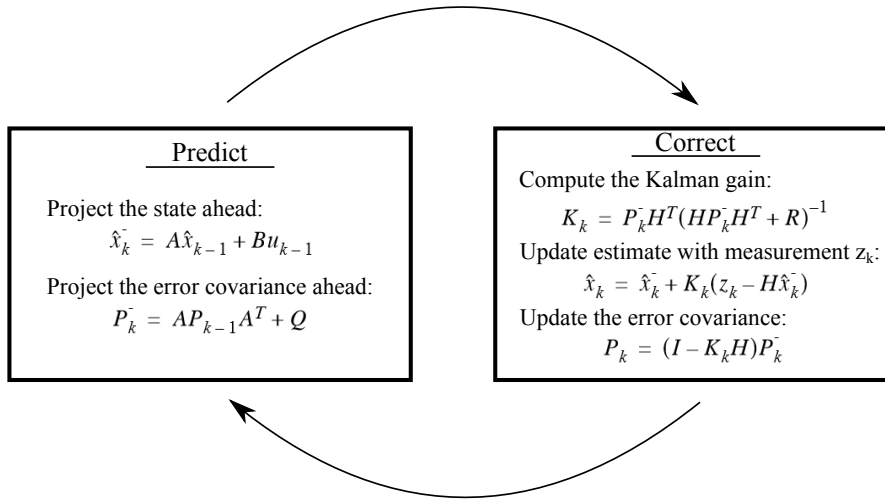
with a measurement  $z$ :

$$z_k = Hx_k + v_k \quad (2)$$

where  $w_k$  and  $v_k$  are random variables representing the process and measurement noise, respectively. The matrix  $A$  is the transition matrix that relates the state  $x$  at the previous time step  $k - 1$  to the state at the current step  $k$ . The matrix  $B$  relates the control input  $u_{k-1}$  to the state  $x_k$  (the control input is optional, and this term is often discarded). The matrix  $H$  relates the state  $x_k$  to the measurement  $z_k$ .

Figure 6 illustrates the procedure of the Kalman filter, shifting between predicting the next step from the previous state and correcting the state using a new observed measurement.

**Figure 6.** Procedure of the Kalman filter.



Using the Kalman filter for tracking an object in 2D, the state  $x$  consists of four dynamic variables; x-position, y-position, x-velocity and y-velocity. The measurement  $z$  represents the observed x- and y-positions for each frame.

When implementing a Kalman filter, the measurement noise covariance  $R$  and the process noise covariance  $Q$  must be tuned.  $R$  represents the measurement noise variance, meaning that a high value will tell the system to rely less on the measurements and vice versa.

For more details on the Kalman filter, we refer to the introduction in [24] or the original paper [13].



### 3.2. Multi-Target Data Association

Each Kalman filter maintains only the estimated state of one object. In order to keep track of several targets simultaneously, the association between detections and Kalman filters must be handled explicitly. For each frame, a list of detections are obtained as described in Section 2. Each existing Kalman filter is then assigned the nearest detection, within a given distance threshold  $th$ . For each detection that is not assigned to a Kalman filter, a new track is started, by creating a new Kalman filter. Kalman filters that have no assigned detections will be continued based on the predicted new positions. After a given time period without detections, experimentally set to 10 frames, the track will be terminated.

### 3.3. Tracking by Continuous Energy Minimization (CEM)

The choice of tracking based on the Kalman filter leaves no possibility for connecting broken tracks, as it is a purely recursive approach. This possibility of optimising both forward and backward in time is instead exploited in offline algorithms based on batch optimisation. We will here test one of these algorithms, using code available online. This algorithm minimises an energy function of five terms [22]:

$$E(X) = E_{obs} + \alpha E_{dyn} + \beta E_{exc} + \gamma E_{per} + \delta E_{reg} \quad (3)$$

$E_{obs}$  represents the likelihood of object presence, determined by the object detector.  $E_{dyn}$  is the dynamic model, using a constant velocity model.  $E_{exc}$  is a mutual exclusion term, introducing the physical constraint that two objects cannot be present at the same space simultaneously. The target persistence term  $E_{per}$  penalises trajectories with start or end points far from the image border. The last term,  $E_{reg}$ , is a regularisation term that favours fewer targets and longer trajectories.

Given the set of detections for all frames, this algorithm will try to minimise the energy function (3) by growing, shrinking, splitting, merging, adding or removing until either convergence or reaching the maximum number of iterations. For further details, see [22].

The set of detections are found as described in Section 2. Being the same detection algorithm used for both Kalman filter and CEM tracker, the tracking algorithms can be compared directly.

## 4. Experiments

In this section we test the tracking algorithm on a 2-minutes (3019 frames) video from an indoor soccer game. The frames are manually annotated using bbLabeler from Piotr's Image & Video MATLAB Toolbox [25]. All frames are annotated by the same person in order to ensure consistency.

### 4.1. Kalman Tracker

The Kalman filter tracker is implemented in C# using EMGU CV wrapper for the OpenCV library [26]. Through experiments the measurement noise covariance  $R$  has been tuned to 0.1 and the process noise covariance  $Q$  is tuned to 0.002 for position and 0.003 for velocity.

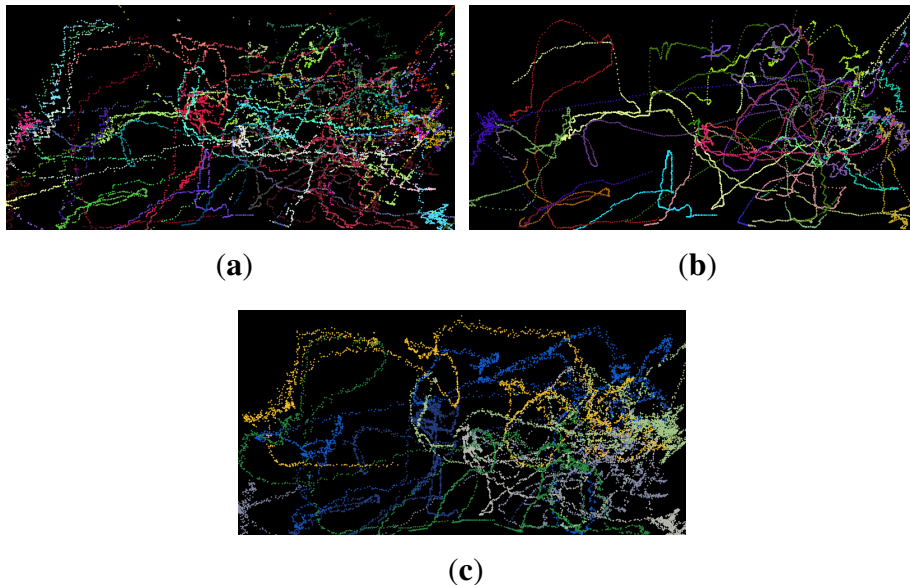
#### 4.2. CEM Tracker

The CEM tracker is downloaded from the author’s website (<http://www.milanton.de/contracking/index.html>). We use the 2D tracking option, tracking in image coordinates. Default parameter values are used, except for three parameters: Target size is reduced to ImageWidth/200 (approx. 10 pixels) due to the relatively small object size in the test video. The maximum number of global iterations is varied between 15, 30 and 60 iterations, along with the maximum number of iterations for each gradient descent, which is varied between 30, 60 and 120 iterations.

#### 4.3. Results

The trajectories found by the Kalman tracker, CEM tracker and manually annotated trajectories, respectively, are plotted in Figure 7. The trajectories are plotted in world coordinates, thus each image represents the sports field seen from above. Each new identity found by the tracker is plotted in a new colour assigned randomly. The figure shows that while the trajectories found by the CEM tracker is longer and smoother, the Kalman tracker produces more tracks, which are also very close to the ground truth.

**Figure 7.** Trajectories plot in world coordinates with each identity assigned a random colour. (a) Trajectories found by Kalman tracker; (b) trajectories found by CEM tracker (60 epochs) and (c) manually annotated trajectories.



We evaluate the tracking results using CLEAR MOT metrics [27], calculated by publicly available MATLAB code [28]. The results are measured by true positives (TP), false positives (FP), false negatives (FN), ID switches and the two combined quality measures: multiple object tracking precision (MOTP) and multiple object tracking accuracy (MOTA):

$$MOTP = \frac{\sum_{i,t} d_t^i}{\sum_t c_t} \quad (4)$$

where  $d_t^i$  is the distance between the object  $o_i$  and its corresponding hypothesis.  $c_t$  is the number of matches found for time  $t$ . Hence, MOTP is the total error in estimated position for matched object–hypothesis pairs over all frames, averaged by the total number of matches made.

$$MOTA = 1 - \frac{\sum_t (FN_t + FP_t + IDS_t)}{\sum_t g_t} \quad (5)$$

where  $FN_t$ ,  $FP_t$  and  $IDS_t$  are the number of false negatives, false positives and ID switches, respectively, for time  $t$ , while  $g_t$  is the true number of objects at time  $t$ .

The results of the Kalman tracker and the CEM tracker with three different numbers of maximum iterations are presented in Table 1. The result for Kalman filtering is very good, considering the complexity of the data. The true positive rate exceeds 80% and the accuracy (MOTA) is 70.36%. For the CEM tracker these numbers are significantly lower, the best true positive rate is 18.14% obtained after 60 epochs. This implies a high false negative rate of 81.6%, but also a high false positive rate of 38.06%. The resulting MOTA ends up being negative. The results are clearly related to the total track length; the Kalman filter constructs more than twice the total length of tracks, which is closer to the total length of ground truth tracks of 3241.52 m.

**Table 1.** Tracking results for Kalman filter and continuous energy minimization (CEM) algorithms.

	TP	FP	FN	ID Switch	MOTP	MOTA	Total Track Length	#ID's
<b>KF</b>	80.22%	9.86%	18.86%	219	0.75	70.36%	2506.78 m	218
<b>CEM - 15 epochs</b>	11.61%	27.38%	88.14%	60	0.58	−15.77%	933.66 m	37
<b>CEM - 30 epochs</b>	17.07%	33.19%	82.72%	51	0.59	−16.11%	1100.69 m	31
<b>CEM - 60 epochs</b>	18.14%	38.06%	81.60%	60	0.60	−19.91%	1228.14 m	33

#### 4.4. Processing Time

The processing time, calculated for 3019 frames of video containing 8 people, is for the MATLAB implementation of the CEM tracker (excluding detection) with 15 epochs: 6.03 min (0.12 s per frame), 30 epochs: 8.75 min (0.17 s per frame), 60 epochs: 16.34 min (0.32 s per frame). For the C# implementation of Kalman filter tracking (with integrated detection) the processing time is only 1.55 min (0.03 s per frame).

Both methods are tested on an Intel Core i7-3770K CPU 3.5 GHz with 8 GB RAM.

## 5. Discussion

We have tested the CEM tracker with three different numbers of maximum iterations, in order to investigate whether more iterations would allow the algorithm to reach a better estimate. From 15 to 30 epochs we observe clear improvements, from a true positive rate of 11.61% to 17.07% and the false negative rate decreasing accordingly. The false positive rate increases from 27.38% to 33.19%, though. Increasing the maximum number of iterations from 30 to 60 gives only a small improvement

in true positive rate from 17.07% to 18.14%, while the false positive rate increases from 33.19% to 38.06%. This indicates that further iterations will not improve the accuracy.

Given that the CEM tracker is an offline algorithm, processing a batch of frames, it is able to run the optimisation both forward and backward in time. That makes it more likely to connect broken trajectories compared with the Kalman tracker, which is recursive and needs to start a new trajectory if it loses one. As expected, this is observed as more identity switches by the Kalman tracker (219 switches) compared with the CEM tracker (51–60 switches). It is also reflected in the mean length of each trajectory; for the Kalman tracker the mean length is 11.5 m, compared with 25.2–37.2 m for the CEM tracker.

The processing time of the two algorithms indicates another big difference between online and offline approaches. The Kalman filter is well-suited for real-time applications with a processing time of only 0.03 s per frame including both detection and tracking. For the CEM tracker the detections must be saved for the full batch of frames before starting to construct trajectories. The processing time is then 0.12–0.32 s per frame, depending on the number of iterations. Furthermore, the processing time might increase significantly with the number of targets.

Both tracking algorithms are independent of the type of detection algorithm, making it possible to apply tracking to a wide range of applications. In this work we demonstrated the approach on a video from an indoor sports arena, but it could be applied directly in any scene where the human temperature is different from the background, including outdoor scenes. The performance depends on the quality of detections. In order to significantly reduce the occlusions between people, the camera could be mounted above the scene, capturing a top-view instead of the side-view shown in Figure 1.

## 6. Conclusions

We have presented an online multi-target tracking algorithm based on the Kalman filter and compared with a state-of-the-art offline multi-target tracking algorithm. In terms of accuracy the Kalman tracker is far superior in this application and constructs more than twice the total length of tracks. The drawback of this online approach is the number of split tracks and identity switches. Depending on the application and importance of identity, a post-processing method could be applied in order to optimise and connect trajectories.

## Acknowledgements

This project is funded by *Nordea-fonden* and *Lokale-og Anlægsfonden*, Denmark. We would also like to thank Aalborg Municipality for support and for providing access to their sports arenas.

## Author Contributions

Rikke Gade has designed the Kalman filter, performed the experiments and prepared this manuscript. Thomas Moeslund has been supervising the work and revising the paper.

## Conflicts of Interest

The authors declare no conflict of interest.

## References

1. Gade, R.; Moeslund, T. Thermal cameras and applications: A survey. *Mach. Vis. Appl.* **2014**, *25*, 245–262.
2. Zhu, Z.; Huang, T.S. *Multimodal Surveillance: Sensors, Algorithms and Systems*; Artech House Publisher: Norwood, MA, USA, 2007.
3. Airouche, M.; Bentabet, L.; Zelmat, M.; Gao, G. Pedestrian tracking using color, thermal and location cue measurements: A DSMT-based framework. *Mach. Vis. Appl.* **2012**, *23*, 999–1010.
4. Torabi, A.; Masse, G.; Bilodeau, G.A. An iterative integrated framework for thermal-visible image registration, sensor fusion, and people tracking for video surveillance applications. *Comput. Vis. Image Underst.* **2012**, *116*, 210–221.
5. Watada, J.; Musa, Z.; Jain, L.; Fulcher, J. Human Tracking: A State-of-Art Survey. In *Knowledge-Based and Intelligent Information and Engineering Systems*; Setchi, R., Jordanov, I., Howlett, R., Jain, L., Eds.; Springer Berlin Heidelberg: Berlin, Germany, 2010; Volume 6277, pp. 454–463.
6. Wu, Y.; Lim, J.; Yang, M.H. Online Object Tracking: A Benchmark. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013; pp. 2411–2418.
7. Alkali, A.; Saatchi, R.; Elphick, H.; Burke, D. Facial Tracking in Thermal Images for Real-Time Noncontact Respiration Rate Monitoring. In Proceedings of the European Modelling Symposium (EMS), Manchester, UK, 20–22 November 2013; pp. 265–270.
8. AL-Khalidi, F.; Saatchi, R.; Burke, D.; Elphick, H. Tracking human face features in thermal images for respiration monitoring. In proceedings of the IEEE/ACS International Conference on Computer Systems and Applications (AICCSA), Hammamet, Tunisia, 16–19 May 2010; pp. 1–6.
9. Lee, W.; Jung, K.; Kim, Y.; Lee, G.; Park, C. Implementation of Face Tracking System for Non-Contact Respiration Monitoring. In Proceedings of the 2012 International Conference on Future Information Technology and Management Science & Engineering (FITMSE 2012), Hong Kong, China, 12–13 April 2012; pp. 160–163.
10. Kalal, Z.; Matas, J.; Mikolajczyk, K. P-N Learning: Bootstrapping Binary Classifiers from Unlabeled Data by Structural Constraint. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, 13–18 June 2010; pp. 49–56.
11. Hare, S.; Saffari, A.; Torr, P.H.S. Struck: Structured output tracking with kernels. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Barcelona, Spain, 6–13 November 2011; pp. 263–270.
12. Henriques, J.; Caseiro, R.; Martins, P.; Batista, J. Exploiting the Circulant Structure of Tracking-by-Detection with Kernels. In *Lecture Notes in Computer Science*; Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C., Eds.; Springer Berlin Heidelberg: Berlin, Germany, 2012; Volume 7575, pp. 702–715.

13. Kalman, R.E. A New Approach to Linear Filtering and Prediction Problems. *Trans. ASME J. Basic Eng.* **1960**, 82, 35–45.
14. Vermaak, J.; Doucet, A.; Pérez, P. Maintaining Multi-Modality through Mixture Tracking. In Proceedings of the Ninth IEEE International Conference on Computer Vision, Nice, France, 13–16 October 2003; pp. 1110–1116.
15. Breitenstein, M.D.; Reichlin, F.; Leibe, B.; Koller-Meier, E.; Van Gool, L. Robust Tracking-by-Detection using a Detector Confidence Particle Filter, In Proceedings of the IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; pp. 1515–1522.
16. Berclaz, J.; Fleuret, F.; Fua, P. Multiple Object Tracking Using Flow Linear Programming. In Proceedings of the 12th IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (Winter-PETS), Snowbird, UT, USA, 7–9 December 2009; pp. 1–8.
17. Berclaz, J.; Fleuret, F.; Türetken, E.; Fua, P. Multiple Object Tracking using K-Shortest Paths Optimization. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, 33, 1806–1819.
18. Skoglar, P.; Orguner, U.; Törnqvist, D.; Gustafsson, F. Pedestrian tracking with an infrared sensor using road network information. *EURASIP J. Adv. Signal Process.* **2012**, 2012, 1–18.
19. Treptow, A.; Cielniak, G.; Duckett, T. Real-time people tracking for mobile robots using thermal vision. *Robot. Auton. Syst.* **2006**, 54, 729–739.
20. Jüngling, K.; Arens, M. Local Feature Based Person Detection and Tracking Beyond the Visible Spectrum. In *Machine Vision Beyond Visible Spectrum*; Springer-Verlag: Berlin, Germany, 2011; pp. 3–32.
21. Lee, S.; Shah, G.; Bhattacharya, A.; Motai, Y. Human tracking with an infrared camera using a curve matching framework. *EURASIP J. Adv. Signal Process.* **2012**, 2012, 1–15.
22. Andriyenko, A.; Schindler, K. Multi-target tracking by continuous energy minimization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 20–25 June 2011; pp. 1265–1272.
23. Gade, R.; Jørgensen, A.; Moeslund, T.B. Occupancy Analysis of Sports Arenas Using Thermal Imaging. In Proceedings of the International Conference on Computer Vision and Applications, Rome, Italy, 24–26 February 2012; pp. 277–283.
24. Welch, G.; Bishop, G. An Introduction to the Kalman Filter. Technical report, Chapel Hill, NC, USA, 1995.
25. Dollár, P. Piotr's Image and Video Matlab Toolbox (PMT). Available online: <http://vision.ucsd.edu/~pdollar/toolbox/doc/index.html> (accessed on 28 July 2014).
26. EMGU CV. Documentation. Available online: <http://www.emgu.com/wiki/index.php/Documentation> (accessed on 28 July 2014).
27. Bernardin, K.; Stiefelhagen, R. Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics. *EURASIP J. Adv. Signal Process.* **2008**, 2008, 246309.
28. Bagdanov, A.D.; Del Bimbo, A.; Dini, F.; Lisanti, G.; Masi, I. Compact and efficient posterity logging of face imagery for video surveillance. *IEEE MultiMed.* **2012**, 19, 48–59.

# Joint Target Tracking, Recognition and Segmentation for Infrared Imagery Using a Shape Manifold-Based Level Set

Jiulu Gong, Guoliang Fan, Liangjiang Yu, Joseph P. Havlicek, Derong Chen and Ningjun Fan

**Abstract:** We propose a new integrated target tracking, recognition and segmentation algorithm, called ATR-Seg, for infrared imagery. ATR-Seg is formulated in a probabilistic shape-aware level set framework that incorporates a joint view-identity manifold (JVIM) for target shape modeling. As a shape generative model, JVIM features a unified manifold structure in the latent space that is embedded with one view-independent identity manifold and infinite identity-dependent view manifolds. In the ATR-Seg algorithm, the ATR problem is formulated as a sequential level-set optimization process over the latent space of JVIM, so that tracking and recognition can be jointly optimized via implicit shape matching where target segmentation is achieved as a by-product without any pre-processing or feature extraction. Experimental results on the recently released SENSIAC ATR database demonstrate the advantages and effectiveness of ATR-Seg over two recent ATR algorithms that involve explicit shape matching.

Reprinted from *Sensors*. Cite as: Gong, J.; Fan, G.; Yu, L.; Havlicek, J.P.; Chen, D.; Fan, N. Joint Target Tracking, Recognition and Segmentation for Infrared Imagery Using a Shape Manifold-Based Level Set. *Sensors* **2014**, *14*, 10124–10145.

## 1. Introduction

As a challenging problem in pattern recognition and machine learning for decades, automatic target tracking and recognition (ATR) has been an important topic for many military and civilian applications. Infrared (IR) ATR is a more challenging problem due to two main reasons. First, an IR target's appearance may change dramatically under different working conditions and ambient environment. Second, the IR imagery usually has poor quality compared with the visible one. There are two important and related research issues in ATR research, appearance representation and motion modeling [1]. The former one focuses on capturing distinct and salient features (e.g., edge, shape, texture) of a target, and the latter one tries to predict the target's state (e.g., position, pose, velocity) during sequential estimation. They could play a complementary role in an ATR process [2].

Shape is a simple yet robust, feature for target representation in many ATR applications. There are three commonly used ways of shape representation: a 3D mesh model [3], 2D shape templates [4,5] and a manifold-based shape generative model learned from 2D snapshots [6–8]. When a 3D model was used, a 3D-to-2D projection is needed to get the 2D shapes according to the camera model and the target's position. Using a 3D model for shape modeling usually needs more memory and expensive computational resources. In [5], a 2D shape template was used to represent the target's appearance, and an online learning was used to update this shape model under different views. Manifold learning methods have proven to be powerful for shape modeling by providing a variety of meaningful shape prior to assist or constrain the shape matching process. In [8], a couplet of view and identity manifolds (CVIM) was proposed for multi-view and multi-target shape modeling, where

target pre-segmentation was implemented via background subtraction and the ATR inference involves explicit shape matching between segmented targets and shapes hypothesis generated by CVIM.

In this work, we propose a new particle filter-based ATR-Seg (segmentation) algorithm that integrates JVIM (joint view-identity manifold) with a shape-aware level set energy function which leads to a joint tracking, recognition and segmentation framework. JVIM encapsulates two shape variables, identity and view, in a unified latent space, which is embedded with one view-independent identity manifold and infinite identity-dependent view manifolds. Unlike CVIM obtained via nonlinear tensor decomposition, JVIM is learned via a modified Gaussian process latent variable model [9] which leads to a probabilistic shape model. Also, a stochastic gradient descent method [10] is developed to speed up JVIM learning, and a local approximate method is used for fast shape interpolation and efficient shape inference. Furthermore, we integrate JVIM with a level set energy function that is able to evaluate how likely a shape synthesized by JVIM can segment out a valid target from an image. This energy function is adopted as the likelihood function in the particle filter where a general motion model is used for handling highly maneuverable targets. The performance of ATR-Seg was evaluated using the SENSIAC (Military Sensing Information Analysis Center) IR dataset [11], which demonstrated the advantage of the proposed method over several methods that involve target pre-segmentation and explicit shape matching.

The remainder of this paper is organized as follow. In Section 2, we review some related works on shape manifold learning and shape matching. In Section 3, we use a graphical model to develop a probabilistic framework of our ATR-Seg algorithm. In Section 4, we introduce JVIM for general shape modeling. In Section 5, we present a shape-aware level set energy function for implicit shape matching. In Section 6, we present a particle filter-based sequential inference method for ATR-Seg. In Section 7, we evaluate the proposed ATR-Seg algorithm in two aspects, *i.e.*, JVIM-based shape modeling and implicit shape matching which are involved in the likelihood function of the particle filter. We conclude our paper in Section 8.

## 2. Related Works

ATR itself is a broad field involving diverse topics. Due to the fact that shape modeling is the key issue in our ATR research, our review below will be focused on two shape-related topics, manifold-based shape modeling and shape matching.

### 2.1. Manifold-Based Shape Modeling

A manifold-based shape model can be learned from a set of exemplar shapes and is able to interpolate new shapes from the low-dimensional latent space. Roughly speaking, there are three manifold learning approaches for shape modeling, geometrically-inspired methods, latent variable models, and hybrid models. The first approach seeks to preserve the geometric relationships among the high-dimensional data in the low-dimensional space, *e.g.*, IsoMap [12], Local Linear Embedding (LLE) [13], Diffusion Maps [14] and Laplacian Eigenmaps [15]. These methods focus on how to explore the geometric structure among the high-dimensional data and how to maintain this structure in the low dimensional embedding space. However, the mapping relationship from the latent space



and the data space is not available and has to be learned separately. The second approach represents the shape data by a few latent variables along with a mapping from the latent space to the data space, such as PCA [16], PPCA [17], KPCA [18], Gaussian Process Latent Variable Models (GPLVM) [19] and tensor decomposition [20], *et al.* GPLVM is a probabilistic manifold learning method which employs the Gaussian process as the nonlinear mapping function. Above approaches are data driven shape modeling methods without involving prior knowledge in the latent space, and as a result, the shape-based inference process may be less intuitive due to the lack of a physically meaningful manifold structure.

To support a more meaningful and manageable manifold structure while preserving the mapping function, there is a trend to combine the first two approaches along with some topology prior for manifold learning [21]. In [9], the local linear GPLVM (LL-GPLVM) was proposed for complex motion modeling, which incorporates a LLE-based topology prior in the latent space. Specifically, a circular-shaped manifold prior is used to jointly model both “walking” and “running” motion data in a unified cylinder-shaped manifold. In [8], CVIM was proposed for shape modeling via nonlinear tensor decomposition where two independent manifolds, an identity manifold and a view manifold, were involved. Specifically, the view manifold was assumed to be a hemisphere that represents all possible viewing angles for a ground target, and the identity manifold was learned from the tensor coefficient space that was used to interpolate “intermediate” or “unknown” target types from known ones. A key issue about the identity manifold is the determination of manifold topology, *i.e.*, the ordering relationship across all different target types. Sharing a similar spirit of IsoMap, the shortest-closed-path is used to find the optimal manifold topology that allows targets with similar shapes to stay closer and those with dissimilar shapes far away. This arrangement ensures the best local smoothness and global continuity that are important for valid shape interpolation along the identity manifold.

## 2.2. Shape Matching

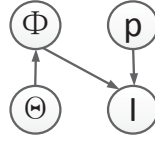
In shape-based tracking algorithms, there are two ways to measure shape similarity: explicit shape matching and implicit shape matching. The former one involves a direct spatial comparison between two shapes, an observed one and a hypothesized one, by using a certain distance metric. In such a case, pre-processing or feature extraction, *e.g.*, background subtraction in [8], is needed prior to tracking and recognition, which is relatively manageable for a stationary sensor platform and may need additional computational load in a general case. Moreover, the overall ATR performance could be sensitive to the pre-processing results. The latter one represents a shape implicitly by a level set embedding function which can be used to evaluate the segmentation quality of a given shape in an image. For example, a shape-constrained energy function was used in [6,7] to evaluate how likely the given shape can segment out a valid object, where a gradient descent method was used to optimize this energy function to achieve tracking and segmentation jointly. Therefore, implicit shape matching does not involve any pre-processing or feature extraction beforehand, however, due to the lack of dynamic modeling in level set optimization, it is still hard to track highly maneuverable targets by the traditional data-driven gradient descent optimization method. As pointed in [22], motion/dynamic

modeling is an important step for most ATR applications. This motivates our research to augment a motion model in implicit shape matching for maneuverable target tracking.

### 3. ATR-Seg Problem Formulation

We list all symbols used in this paper in Table 1. Given the observed video sequence  $\mathbf{I}_t$ , with  $t = 1, \dots, T$ , where  $T$  is the total number of image frames, the objective of ATR-Seg is to (1) find the 3D position of a target in the camera coordinate  $\mathbf{p}$  (tracking) or 2D image coordinate, (2) to identify the target type  $\alpha$  (recognition), along with the view angle  $\varphi$  (pose estimation), and (3) to segment the target-of-interest that best explains the observation data  $\Phi$  (segmentation). The 2D shape of a target can be determined by the target type  $\alpha$ , and view angle  $\varphi$ , so we define  $\Theta = [\alpha, \varphi]$  to represent two shape related variables. The conditional dependency among all variables is shown in Figure 1.

**Figure 1.** Graphical modeling for the proposed ATR-Seg algorithm, where  $\mathbf{I}_t$  represents an image frame,  $\mathbf{p}$  3D target position,  $\Phi$  target segmentation, and  $\Theta$  the set of shape variables.



According to Figure 1, we define the objective function of ATR-Seg from the joint distribution  $p(\mathbf{p}_t, \Theta_t, \Phi, \mathbf{I}_t)$  which can be written ( $t$  is omitted for simplicity) as:

$$p(\mathbf{p}, \Theta, \Phi, \mathbf{I}) = p(\mathbf{I}|\mathbf{p}, \Phi)p(\Phi|\Theta)p(\Theta)p(\mathbf{p}) \quad (1)$$

By using the Bayesian theorem, we can get the posterior as:

$$p(\mathbf{p}, \Theta, \Phi|\mathbf{I}) \propto p(\mathbf{I}|\mathbf{p}, \Phi)p(\Phi|\Theta)p(\Theta)p(\mathbf{p}) \quad (2)$$

which encapsulates three major components in the proposed ATR-Seg algorithm, as shown below:

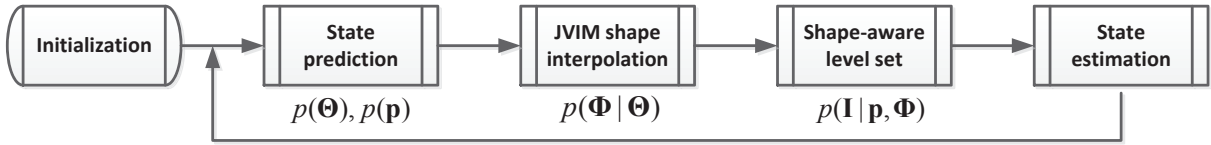
- Shape manifold learning provides a mapping from  $\Theta$  to  $\Phi$ , *i.e.*,  $p(\Phi|\Theta)$ . In Section 4, JVIM is proposed for multi-view and multi-target shape modeling, which features a novel manifold structure with one view-independent identity manifold and infinite identity-dependent view manifolds to impose a conditional dependency between the two shape-related factors, view and identity, in a unified latent space.
- Shape-aware level set  $p(\mathbf{I}|\mathbf{p}, \Phi)$  measures how likely  $\Phi$  can segment a valid target at position  $\mathbf{p}$  in image  $I$ . In Section 5, a shape-aware level set energy function is proposed for implicit shape matching, which evaluates the segmentation quality.
- Pose/position priors  $\Theta$  and  $\mathbf{p}$ , *i.e.*,  $p(\Theta)$  and  $p(\mathbf{p})$ , *i.e.*, are important to track highly manoeuvrable targets in a sequential manner. In Section 6, sequential shape inference method is presented that involve dynamic priors for  $\Theta$  and  $\mathbf{p}$  using a 3D motion model.

**Table 1.** Descriptions of all mathematical symbols.

Symbols used in Problem Formulation (Section 3)	
$\mathbf{p}$	the target's 3D position $\mathbf{p} = [p_x, p_y, p_z]^T$
$\alpha$	the target type
$\varphi$	the view angle
$\Theta$	shape related variables ( $[\alpha, \varphi]$ )
$\Phi$	a target shape segmentation
$\mathbf{I}_t$	an observed image frame at time $t$
Symbols used in JVIM-based shape modeling (Section 4)	
$\mathbf{Y}$	JVIM training data
$\mathbf{X}$	JVIM latent space
$\theta$	the aspect angle of a target
$\phi$	the elevation angle of a target
$\beta$	the kernel hyper-parameters of JVIM
$d$	the dimension of the shape space
$\mathbf{w}$	the LLE coefficients for local topology encoding
$L_{JVIM}$	the JVIM objective function
$L_D$	the data term in $L_{JVIM}$
$L_T$	the topology term in $L_{JVIM}$
$\mathbf{K}_Y$	the covariance matrix of JVIM learning
$\mathbf{x}_r$	a reference latent point in JVIM learning
$\mathbf{X}_R$	the neighborhood of $\mathbf{x}_r$ for local learning
$M_1$	the size of $\mathbf{X}_R$ (the range of local learning)
$\mathbf{Y}_R$	the corresponding shape for $\mathbf{X}_R$
$N$	the size of training data
$\mathbf{x}'$	a new latent point for JVIM-based shape interpolation
$\mathbf{X}'$	the neighborhood of $\mathbf{x}'$ for local inferencing
$M_2$	the size of $\mathbf{X}'$ (the range of local inferencing)
$\mathbf{Y}'$	the corresponding shape data for $\mathbf{X}'$
$k(\mathbf{x}_1, \mathbf{x}_2)$	a RBF kernel function in JVIM
$\hat{\boldsymbol{\mu}}_{\mathbf{x}'}$	an interpolated shape at $\mathbf{x}'$ via JVIM
$\hat{\sigma}_{\mathbf{x}'}$	uncertainty of shape interpolation at $\mathbf{x}'$
Symbols used in shape-aware level set (Section 5)	
$x$	a 2D pixel location in an image frame
$y$	a pixel intensity value
$M$	foreground/background models $M = \{M_f, M_b\}$
$H_\epsilon[\cdot]$	the smoothed Heaviside step function
Symbols used in sequential inference (Section 6)	
$\psi_t$	the heading direction of a ground vehicle in frame $t$
$v_t$	the target velocity along $\psi_t$ in frame $t$
$\Delta t$	the time interval of two adjacent frames
$\mathbf{Z}_t$	the state vector in frame $t$ ( $\mathbf{Z}_t = [\mathbf{p}_t^T, v_t, \psi_t, \alpha_t]^T$ )

The flowchart for ATR-Seg is shown in Figure 2 where four steps are involved sequentially and recursively. First, state prediction will draw a set of samples to predict all state variables (position/angle/identity). Second, a series of shape hypotheses are created via JVIM in some hypothesized locations according to predicted state information. Third, a level-set energy function is used as the likelihood function to weight each hypothesized shape/location that quantifies how well that shape can segment a valid target in that particular location. Fourth, state estimation at the current frame is obtained by the conditional mean of all weighted samples and will be used for state prediction in the next frame.

**Figure 2.** Flowchart for ATR-Seg.



#### 4. Joint View-Identity Manifold (JVIM)

JVIM is learned from a set of 2D shape exemplars  $\mathbf{Y}$  generated from a set of 3D CAD models. The latent space  $\mathbf{X}$  can be represented by two variables, identity  $\alpha$  and view  $\varphi$  (including aspect angle  $\theta$  and elevation angle  $\phi$ ), which are defined along their respective manifolds. Considering the fact that all targets have different 3D structures, leading to different view manifolds, and they keep the same identity under different views, we impose a conditional dependency between  $\alpha$  and  $\varphi$  in JVIM that encapsulates one view-independent identity manifold and infinite identity-dependent view manifolds. Specifically, the identity manifold represents the view-independent shape variability across different target types, and an identity-specific view manifold captures the shape variability of a target under different views. Motivated by [8,23], the identity manifold is simplified to have a circular-shaped topology prior, which facilitates manifold learning and shape inference. Intuitively, a hemispherical-shaped topology prior is assumed for identity-specific view manifold, which represents all possible aspect and elevation angles for ground vehicle. All topology priors are encoded by LLE and incorporated into the GPLVM-based learning framework, as shown in Figure 3.

The objective of JVIM learning is to find  $\mathbf{X}$  and  $\beta$  by maximizing  $p(\mathbf{Y}|\mathbf{X}, \beta, \mathbf{w})$ , where  $\beta$  is the mapping parameter and  $\mathbf{w}$  represents the LLE-based topology prior in the latent space. The Gaussian process (GP) is used as the nonlinear mapping function from the latent space to the shape space ( $\mathbf{X} \rightarrow \mathbf{Y}$ ), and the objective function of JVIM learning is written as:

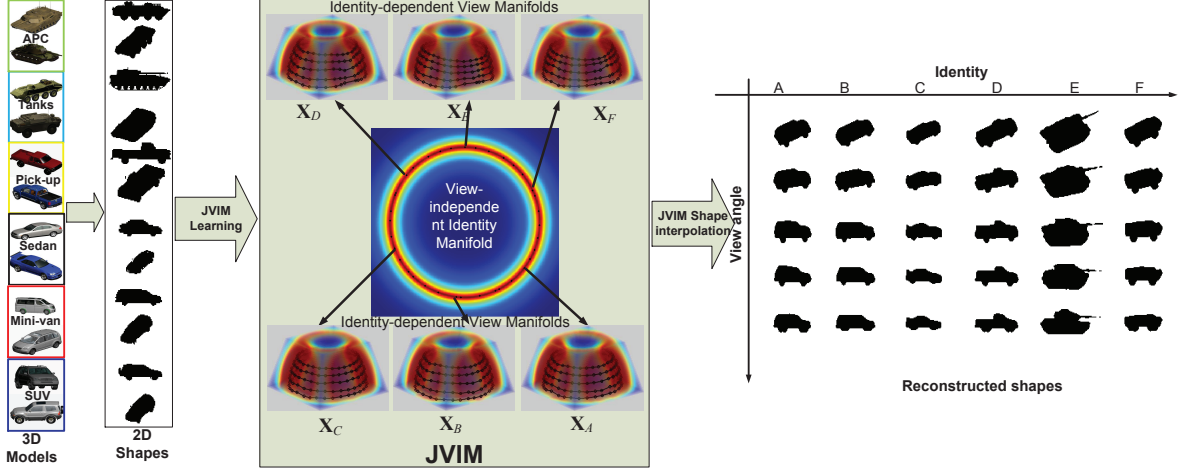
$$p(\mathbf{Y}|\mathbf{X}, \beta, \mathbf{w}) = p(\mathbf{Y}|\mathbf{X}, \beta)p(\mathbf{X}|\mathbf{w}) \quad (3)$$

where:

$$p(\mathbf{Y}|\mathbf{X}, \beta) = \frac{1}{\sqrt{(2\pi)^{Nd} |\mathbf{K}_Y|^d}} \exp\left(-\frac{1}{2} \text{tr}(\mathbf{K}_Y^{-1} \mathbf{Y} \mathbf{Y}^T)\right) \quad (4)$$

where  $d$  is the dimension of the shape space and  $\beta$  denotes the kernel hyper-parameters used in the covariance matrix,  $\mathbf{K}_Y$ . It is worth noting that Equation (4) is similar to the objective function of GPLVM [19], and:

**Figure 3.** JVIM learning and shape interpolation, where one view-independent identity manifold and six identity-dependent view manifolds are color-coded according to the uncertainty of GP mapping. (Adapted from [24], with permission from Elsevier.)



$$p(\mathbf{X}|\mathbf{w}) = \frac{1}{Z} \exp \left( -\frac{1}{\sigma^2} \sum_{i=1}^N \left\| \mathbf{X}_i - \sum_{j=1}^N \mathbf{w}_{ij} \mathbf{X}_j \right\|^2 \right) \quad (5)$$

where  $\mathbf{w}$  is the set of LLE weights to reconstruct each latent point from its local neighboring points by minimizing  $f(\mathbf{w}) = \sum_{i=1}^N \left\| \mathbf{X}_i - \sum_{j=1}^N \mathbf{w}_{ij} \mathbf{X}_j \right\|^2$ ,  $Z$  is a normalization constant,  $\sigma^2$  represents a global scaling of the prior and  $N$  the number of training samples. Furthermore, the negative log operation is used to simplify the objective function as:

$$L_{JVIM} = -\log p(\mathbf{Y}|\mathbf{X}, \beta) p(\mathbf{X}|\mathbf{w}) = L_D + L_T + C \quad (6)$$

where  $C$  is a constant.

JVIM learning involves a gradient descent method to minimize the objective function defined in Equation (3) with respect to  $\mathbf{X}$  and  $\beta$ . With an  $O(N^3)$  operation required at each iteration, it is computationally prohibitive for a large training data set. The stochastic gradient descent proposed in [10] is adapted to be a local updating according to the unique structure of JVIM to approximate the gradients locally. At each iteration, the reference point,  $\mathbf{x}_r$ , is chosen randomly, and the derivatives w.r.t  $\mathbf{X}_R$  and  $\beta$  are calculated as:

$$\frac{\partial L_D}{\partial \mathbf{X}_R} \approx -(\mathbf{K}_R^{-1} \mathbf{Y}_R \mathbf{Y}_R^T \mathbf{K}_R^{-1} - d\mathbf{K}_R^{-1}) \cdot \frac{\partial \mathbf{K}_R}{\partial \mathbf{X}_R} \quad (7)$$

$$\frac{\partial L_{JVIM}}{\partial \beta} \approx -(\mathbf{K}_R^{-1} \mathbf{Y}_R \mathbf{Y}_R^T \mathbf{K}_R^{-1} - d\mathbf{K}_R^{-1}) \cdot \frac{\partial \mathbf{K}_R}{\partial \beta} \quad (8)$$

where  $\mathbf{X}_R$  is the neighborhood for a reference point,  $\mathbf{x}_r$ , of size  $M_1$ ,  $\mathbf{Y}_R$  is the corresponding shape data and  $\mathbf{K}_R$  ( $M_1 \times M_1$ ) is the kernel matrix of  $\mathbf{X}_R$ . The neighborhood for each training data can be pre-assigned according to the topology structure, and the gradients are estimated stochastically, locally and efficiently.

As a generative model, given an arbitrary latent point in  $\mathbf{X}$ , JVIM can generate the corresponding shape via Gaussian Process (GP) mapping. For real-time applications, shape interpolation must be

carried out efficiently, which is difficult for a large training data set with high dimensionality. Inspired by [25], a GP can be approximated by a set of local GPs, in JVIM-based shape interpolation, the kernel matrix is computed locally from a set of training data that are close to the given point. Given  $\mathbf{x}'$ , we first find its closest training point, which has a pre-assigned neighborhood,  $\mathbf{X}'$ , of size  $M_2$ ; then,  $\mathbf{X}'$  and the corresponding shape data  $\mathbf{Y}'$  are used to approximate the mean and variance of GP mapping as:

$$\hat{\boldsymbol{\mu}}_{x'} = \mathbf{k}_{x'X'}^T \mathbf{K}_{Y'}^{-1} \mathbf{Y}' \quad (9)$$

$$\hat{\sigma}_{x'}^2 = k(\mathbf{x}', \mathbf{x}') - \mathbf{k}_{x'X'}^T \mathbf{K}_{Y'}^{-1} \mathbf{k}_{x'X'} \quad (10)$$

where  $\mathbf{k}_{x'X'}$  is a vector made of  $k(\mathbf{x}', \mathbf{x}_i)$  ( $\mathbf{x}_i \in \mathbf{X}'$ ) and  $\mathbf{K}_{Y'}$  ( $M_2 \times M_2$ ) is the local covariance matrix computed from  $\mathbf{X}'$ . More detail about JVIM learning and inference can be found in our previous work [26], where explicit shape matching is involved. In the following section, we will introduce implicit shape matching by incorporating a shape-aware level set for target tracking and recognition, where target segmentation becomes a by-product.

## 5. Shape-Aware Level Set

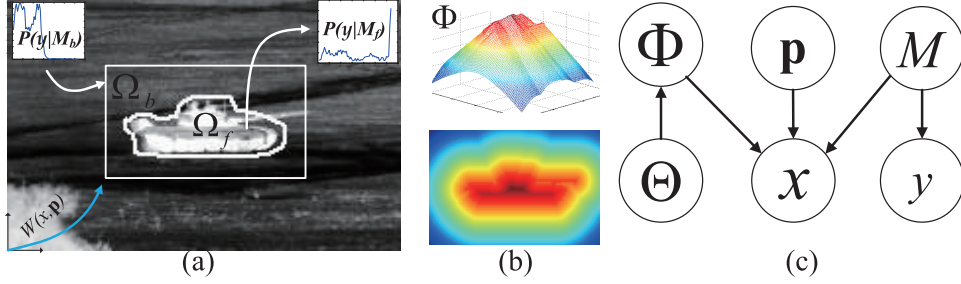
JVIM is used to provide a useful shape prior that can be further combined with the level set to define an energy function for implicit shape matching. This is called the shape-aware level set, which does not involve feature extraction or target pre-segmentation. The shape-aware level set in this work is distinct from that in [6,7,27] primarily in two aspects. Firstly, the shape generated model in [6,7], which was less structured with little semantic meaning and, was limited to object recognition/segmentation under the same view or human pose estimation for the same person along the same walking path. JVIM is a multi-view and multi-target shape model that has a well-defined semantic structure, which supports robust ATR for different targets under arbitrary view angles. Secondly, a gradient decent method was used for level set optimization in [6,7,27], which does not involve a motion model and makes it hard to track highly maneuverable targets. In this work, a 3D motion model is used to combine the position/pose priors into a sequential optimization model to improve the robustness and accuracy of ATR-Seg.

As shown in Figure 4a, we represent an image by  $\mathbf{I} = \{x_i, y_i\}$ , where  $1 \leq i \leq n$ ,  $n$  is the number of pixels in  $\mathbf{I}$  and  $x$  and  $y$  are the pixel 2D location and pixel intensity value, respectively. We introduce a parameter,  $M$ , to represent the foreground/background models  $M = \{M_f, M_b\}$ ; then, the original graphical model of ATR-Seg in Figure 1 will become the one in Figure 4c. which defines a joint distribution of all parameters for each pixel  $(x_i, y_i)$  as

$$p(x_i, y_i, \mathbf{p}, \boldsymbol{\Theta}, \Phi, M) = p(x_i | \mathbf{p}, \Phi, M) p(y_i | M) p(\Phi | \boldsymbol{\Theta}) p(M) p(\boldsymbol{\Theta}) p(\mathbf{p}) \quad (11)$$

where  $\Phi$  is a shape represented by the level set embedding function shown in Figure 4b and  $p(\Phi | \boldsymbol{\Theta})$  corresponds to JVIM-based shape interpolation via GP mapping. A histogram is used for foreground/background appearance model  $p(y_i | M)$ , and the number of bins is dependent on the size of the target and gray scale. In order to get the posterior,  $p(\mathbf{p}, \boldsymbol{\Theta}, \Phi, M | x_i, y_i)$ , which will be used

**Figure 4.** Shape-aware level set model for implicit shape matching. (a) Illustration of a target in an infrared image: foreground  $\Omega_f$  and background  $\Omega_b$ , foreground/background intensity models  $M$ , and the 3D-2D camera projection  $W(\mathbf{x}, \mathbf{p})$ . (b) The shape embedding function  $\Phi$ . (c) The graphical model for shape-aware level set, where  $\mathbf{p}$  is the target 3D location of a ground-vehicle, and  $\Theta$  is the shape parameter in JVIM.



to develop the objective function for ATR-Seg, we take the same strategy as in [27]. First, divide Equation (11) by  $p(y_i) = \sum_{j \in \{f,b\}} p(y_i|M_j)p(M_j)$ :

$$p(x_i, \mathbf{p}, \Theta, \Phi, M|y_i) = p(x_i|\mathbf{p}, \Phi, M)p(M|y_i)p(\Phi|\Theta)p(\Theta)p(\mathbf{p}) \quad (12)$$

where  $p(M|y_i)$  is given by:

$$p(M_j|y_i) = \frac{p(y_i|M_j)p(M_j)}{\sum_{k \in \{f,b\}} p(y_i|M_k)p(M_k)}, \quad j \in \{f, b\} \quad (13)$$

Upon dividing Equation (12) by  $p(x_i) = 1/n$  and marginalizing over the models,  $M$ , we obtain:

$$p(\mathbf{p}, \Theta, \Phi|x_i, y_i) = n \sum_{j \in \{f,b\}} p(x_i|\mathbf{p}, \Phi, M_j)p(M_j|y_i)p(\Phi|\Theta)p(\Theta)p(\mathbf{p}) \quad (14)$$

Assuming all pixels are independent, the posterior for all pixels in a frame is then given by:

$$\begin{aligned} p(x_i|\mathbf{p}, \Phi, M_f) &= H_\epsilon[\Phi(\mathbf{x}_i)]/\eta_f \\ p(x_i|\mathbf{p}, \Phi, M_b) &= \{1 - H_\epsilon[\Phi(\mathbf{x}_i)]\}/\eta_b \end{aligned} \quad (15)$$

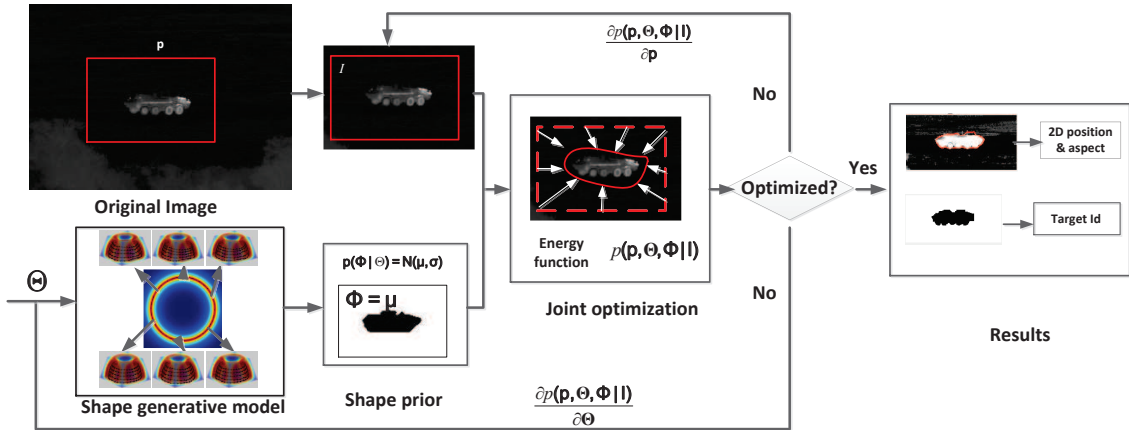
where  $H_\epsilon[\cdot]$  is the smoothed Heaviside step function,  $\eta_f = \sum_{i=1}^n H_\epsilon[\Phi(x_i)]$ ,  $\eta_b = \sum_{i=1}^n \{1 - H_\epsilon[\Phi(x_i)]\}$  and  $p(M_j) = \eta_j/n$  for  $j \in \{f, b\}$ .

Then, from Equations (2), (14) and (15), we have:

$$p(\mathbf{I}|\mathbf{p}, \Phi) \propto \prod_{i=1}^n \sum_{j \in \{f,b\}} p(x_i|\mathbf{p}, \Phi, M_j)p(M_j|y_i) \quad (16)$$

which evaluates how likely shape  $\Phi$  can segment a valid target from  $\mathbf{I}$  at position  $\mathbf{p}$ . The objective function in Equation (2) can be optimized through a gradient descent method similar to the one in [7], which is illustrated in Figure 5. As shown in Figure 5, JVIM is firstly used to generate a shape hypothesis,  $\Phi^0$ , given initial identity and view angle  $\Theta^0$ ; then,  $\Phi^0$  is used to initialize the objective function,  $p(\mathbf{p}, \Theta, \mathbf{I})$ , for initial position,  $\mathbf{p}^0$ . We take the derivative of  $p(\mathbf{p}, \Theta, \Phi|\mathbf{I})$  with respect to  $\Theta$  and  $\mathbf{p}$  to get  $\frac{\partial p(\mathbf{p}, \Theta, \Phi|\mathbf{I})}{\partial \Theta}$  and  $\frac{\partial p(\mathbf{p}, \Theta, \Phi|\mathbf{I})}{\partial \mathbf{p}}$ , which will be used to update  $\Theta$  and  $\mathbf{p}$  until the

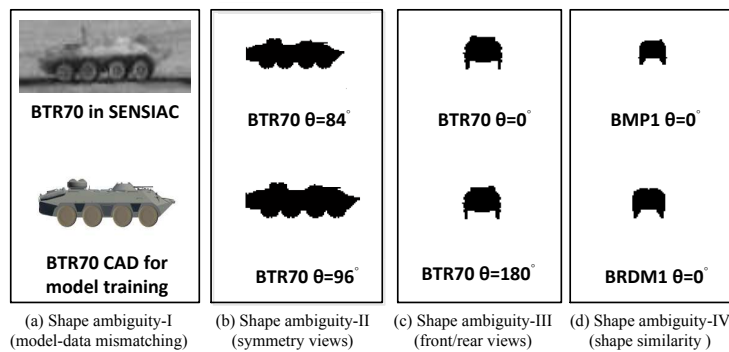
**Figure 5.** Optimization of ATR-Seg by a gradient descent method.



objective function converges. When  $p(\mathbf{p}, \Theta, \Phi | \mathbf{I})$  is maximized, we output the updated target's 2D position,  $\mathbf{p}^*$ , target identity and view angle  $\Theta^*$ , as well as the updated shape  $\Phi^*$  that can best segment the image.

This method works well on a single image when a good initialization is given in the latent space of JVIM. However, it may fail quickly when dealing with an image sequence with a highly maneuverable target, due to four possible cases of shape ambiguity, as shown in Figure 6, which makes data-driven optimization not reliable in practice. (1) The first is due to the possible shape mismatch between the CAD models and real targets, even for the same target type (Figure 6a). (2) The second is due to the symmetry property of a target's shape (Figure 6b), which means a target may present a similar shape at different (e.g., supplement) aspect angles, especially when the elevation angle is zero (Figure 6b). (3) The third is due to the ambiguity of the front/rear views when a target looks very similar (Figure 6c). (4) The fourth is similar to the previous one in which many targets look alike at the front/rear views (Figure 6d). These factors make the gradient-based approach not effective at dealing with a maneuvering target. A possible remedy is to introduce a dynamic motion model to support robust sequential shape inference based on JVIM, as to be discussed below.

**Figure 6.** Possible reasons for the failure of the gradient descent method.





## 6. Sequential Shape Inference

Essentially, the objective of ATR-Seg is to perform sequential shape inference from an image sequence by maximizing the posterior of  $p(\mathbf{p}_t, \Theta_t, \Phi_t | \mathbf{I}_t)$ . According to Figure 1 in Section 3,  $\Phi$  is only dependent on  $\Theta$ , so the objective function can be rewritten as:

$$p(\mathbf{p}_t, \Theta_t, \Phi_t | \mathbf{I}_t) = p(\mathbf{p}_t, \Theta_t | \mathbf{I}_t) p(\Phi_t | \Theta_t) \quad (17)$$

where  $p(\Phi_t | \Theta_t)$  is JVIM-based shape interpolation via GP mapping. Since  $p(\Phi_t | \Theta_t)$  is not related to the observation, so the main computational load is the maximization of  $p(\mathbf{p}_t, \Theta_t | \mathbf{I}_t)$ . For sequential ATR-Seg, the optimization of  $p(\mathbf{p}_t, \Theta_t | \mathbf{I}_t)$  has two stages: prediction and update. In the first stage (prediction), we use a motion model to predict  $p(\mathbf{p}_t, \Theta_t | \mathbf{I}_{t-1})$  from the previous result  $p(\mathbf{p}_{t-1}, \Theta_{t-1} | \mathbf{I}_{t-1})$  as:

$$p(\mathbf{p}_t, \Theta_t | \mathbf{I}_{t-1}) = \int \int p(\mathbf{p}_{t-1}, \Theta_{t-1} | \mathbf{I}_{t-1}) p(\mathbf{p}_t | \mathbf{p}_{t-1}) p(\Theta_t | \Theta_{t-1}) d\Theta_{t-1} d\mathbf{p}_{t-1} \quad (18)$$

where  $p(\mathbf{p}_t | \mathbf{p}_{t-1})$  and  $p(\Theta_t | \Theta_{t-1})$  are used to predict the position and identity/view of a moving target. They are related a motion model that characterizes the target's dynamics and kinematics. In the second stage (update stage), we use the Bayes' rule to compute the posterior as:

$$p(\mathbf{p}_t, \Theta_t, \Phi_t | \mathbf{I}_t) = \frac{p(\mathbf{p}_t, \Theta_t, \Phi_t | \mathbf{I}_{t-1}) p(\mathbf{I}_t | \mathbf{p}_t, \Phi_t)}{p(\mathbf{I}_t | \mathbf{I}_{t-1})} \quad (19)$$

where  $p(\mathbf{p}_t, \Theta_t, \Phi_t | \mathbf{I}_{t-1}) = p(\mathbf{p}_t, \Theta_t | \mathbf{I}_{t-1}) p(\Phi_t | \Theta_t)$  and we have  $p(\mathbf{I}_t | \mathbf{p}_t, \Phi_t, \Theta_t) = p(\mathbf{I}_t | \mathbf{p}_t, \Phi_t)$ . Hence, the objective function of the sequential ATR-Seg algorithm can be further rewritten as:

$$p(\mathbf{p}_t, \Theta_t, \Phi_t | \mathbf{I}_t) \propto p(\mathbf{I}_t | \mathbf{p}_t, \Phi_t) p(\Phi_t | \Theta_t) \iint p(\Theta_t | \Theta_{t-1}) p(\mathbf{p}_{t-1}, \Theta_{t-1}, \Phi_{t-1} | \mathbf{I}_{t-1}) p(\mathbf{p}_t | \mathbf{p}_{t-1}) d\Theta_{t-1} d\mathbf{p}_{t-1} \quad (20)$$

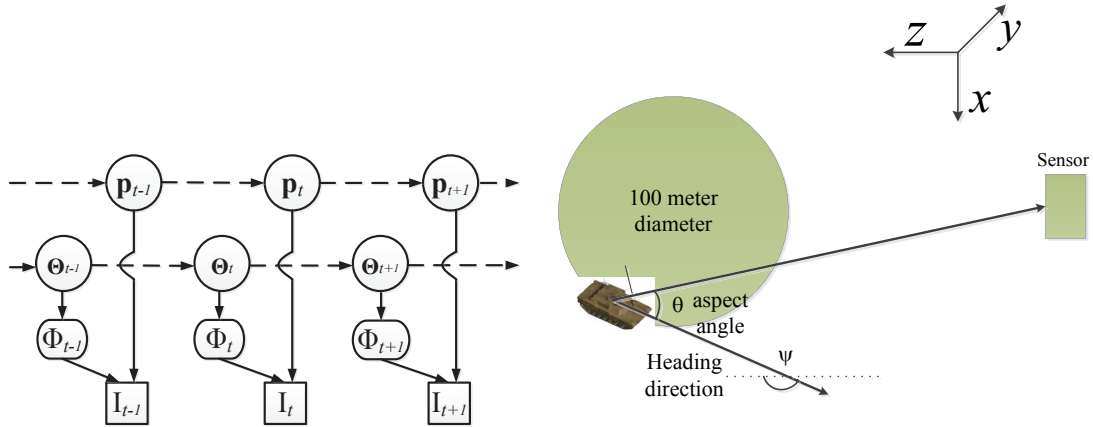
Due to the nonlinear nature of Equation (20), we resort to a particle filter-based inference framework [29] for sequential optimization, as represented by the graphic model in Figure 7 (left). Thanks to the compact and continuous nature of JVIM, we can draw samples from its latent space for efficient shape interpolation. In the inference process, the state vector is defined as  $\mathbf{Z}_t = [\mathbf{p}_t^T, v_t, \psi_t, \alpha_t]^T$ , where  $\mathbf{p}_t = [p_x^t, p_y^t, p_z^t]^T$  represents the target's 3D position, with the  $x - y - z$  axes denoting the horizon ( $x$ ), elevation ( $y$ ) and range ( $z$ ) directions, respectively (as shown in Figure 7 (right));  $v_t$  is the velocity along the heading direction,  $\psi_t$ . A 3D-2D camera projection,  $W(\mathbf{p})$ , is needed to project a 3D position to a 2D position in an image that is assumed to be unchanging for a stationary sensor platform. It is worth noting that we can compute  $\theta_t$  (the aspect angle) from  $\psi_t$  (the heading direction) or *vice versa*. As a matter of fact, the two angles are similar for distant targets when the angle between the line of sight and the optical axis along the range direction ( $z$ ) is very small. Because the target is a ground vehicle and to keep it general, a white noise acceleration model is used to represent the dynamics of  $\mathbf{Z}_t$ , where a simple random walk is applied on the heading direction,  $\psi_t$ , to represent arbitrary maneuvering. Moreover, we define the

dynamics of  $\alpha_t$  (target identity) to be a simple random walk along the identity manifold by which the estimated identity value normally quickly converges to the correct one.

$$\begin{cases} \psi_t = \psi_{t-1} + \zeta_t^\psi, \\ v_t = v_{t-1} + \zeta_t^v, \\ p_x^t = p_x^{t-1} + v_{t-1} \sin(\psi_{t-1})\Delta t + \zeta_t^x, \\ p_y^t = p_y^{t-1} + \zeta_t^y, \\ p_z^t = p_z^{t-1} + v_{t-1} \cos(\psi_{t-1})\Delta t + \zeta_t^z, \\ \alpha^t = \alpha^{t-1} + \zeta_t^\alpha, \end{cases} \quad (21)$$

where  $\Delta t$  is the time interval between two adjacent frames. The process noises associated with the target kinematics,  $\zeta_t^\psi$ ,  $\zeta_t^v$ ,  $\zeta_t^x$ ,  $\zeta_t^y$ ,  $\zeta_t^z$ , and  $\zeta_t^\alpha$ , are usually assumed to be a zero-mean Gaussian.

**Figure 7.** The graphical model representation of ATR-Seg and the 3D camera coordinate. (Reprint from [28] with permission from IEEE).



In a particle filter-based inference algorithm, samples were first drawn according to the dynamics of the state vector and the previous state value, and then, the implicit shape matching defined in Equation (16) was performed to assign a weight for each particle. The mean estimation of weighted samples produces the solution in the present frame. The pseudo-code for the ATR-Seg algorithm is given in Table 2. Thanks to the unique structure of JVIM, we can capture the continuous and smooth shape evolution during target tracking and recognition, where segmentation  $\Phi_t$  is also archived as a by-product via the shape-aware level set. We expect that the proposed ATR-Seg algorithm has some advantages over other methods that require pre-processing or feature extraction prior to ATR inference [8,26].

**Table 2.** Pseudo-code for ATR-Seg algorithm.

- 
- Initialization: Initialize the target position,  $\mathbf{p}_0$ , type  $\alpha_0$ , heading direction  $\psi_0$  and speed  $v_0$  according to the ground-truth and get the initial state,  $\mathbf{Z}_0$ . Draw  $\mathbf{Z}_0^j \sim N(\mathbf{Z}_0, 1)$ ,  $\forall j \in \{1, \dots, N_p\}$ ,  $N_p$  is the number of particles.
  - For  $t = 1, \dots, T$  (number of frames)
    1. For  $j = 1, \dots, N_p$ 
      - 1.1 Draw samples  $\mathbf{Z}_t^j \sim p(\mathbf{Z}_t^j | \mathbf{Z}_{t-1}^j)$  as in Equation (21).
      - 1.2 Generate the target shape according to the target state using Equations (9) and (10).
      - 1.3 Compute weights  $w_t^j = p(\mathbf{z}_t | \alpha_t^j, \mathbf{Z}_t^j)$  using Equation (16).
    - End
    2. Normalize the weights, such that  $\sum_{j=1}^{N_p} w_t^j = 1$ .
    3. Compute the mean estimates of the target state,  $\hat{\mathbf{Z}}_t = \sum_{j=1}^{N_p} w_t^j \mathbf{Z}_t^j$
    4. Set  $\mathbf{Z}_t^j = \text{resample}(\mathbf{Z}_k^j, w_k^j)$  to increase the effective number of particles [29].
  - End
- 

## 7. Experimental Results

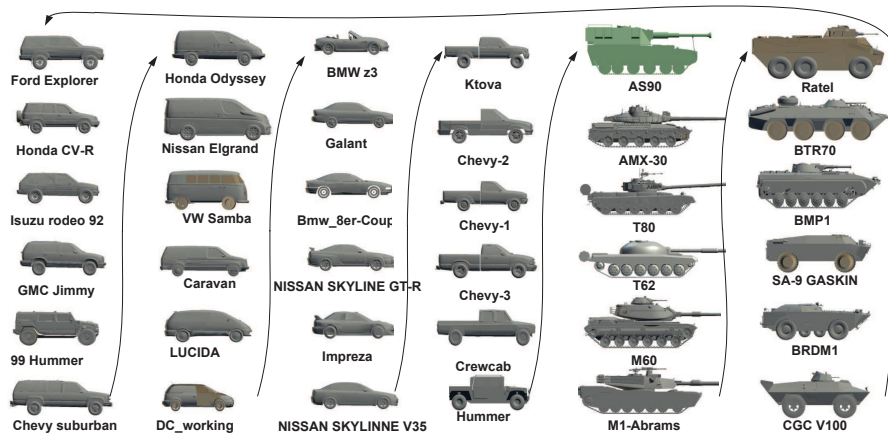
This experimental section provides a detailed evaluation of the ATR-Seg algorithm in six parts. First, we briefly talk about training data collection for JVIM learning with some qualitative results of shape interpolation. Second, we introduce the infrared ATR database used in this work and how different shape models are to be evaluated collectively and fairly. Third, we present the results of the particle filter-based infrared ATR algorithm, where four shape models (JVIM, CVIM, LL-GPLVM, nearest neighbor (NN)) are compared in the case of explicit shape matching. Fourth, we discuss the results of the proposed ATR-Seg algorithm, which involves JVIM-based implicit shape matching and is compared with the algorithms using explicit shape matching (with JVIM and CVIM). Fifth, we discuss the target segmentation results, which are the by-product of the ATR-Seg algorithm. We will also discuss some limitation of ATR-Seg along with some failed cases.

### 7.1. Training Data Collection

In our work, we considered six target classes as [8], *i.e.*, SUVs, mini-vans, cars, pick-ups, tanks and armored personnel carriers (APCs), each of which has six sub-classes, resulting in a total of 36 targets, as shown in Figure 8. These 36 targets were ordered along the view-independent identity manifold according to a unique topology optimized by the class-constrained shortest-closed-path method proposed in [8] (before training). We considered aspect and elevation angles in the ranges  $0 \leq \theta < 2\pi$  and  $0 \leq \phi < \pi/4$ , which are digitized in the interval of  $\pi/15$  and  $\pi/18$  rad, respectively. A total of 150 training viewpoints were used for each target; all training data are generated by their 3D CAD models. In order to reduce the data dimension, the DCT-based shape descriptor proposed in [7] was used here to represent all training shapes for manifold learning. We first detect the contour of a 2D shape ( $120 \times 80$ ) and then apply the signed distance transform to the contour image, followed

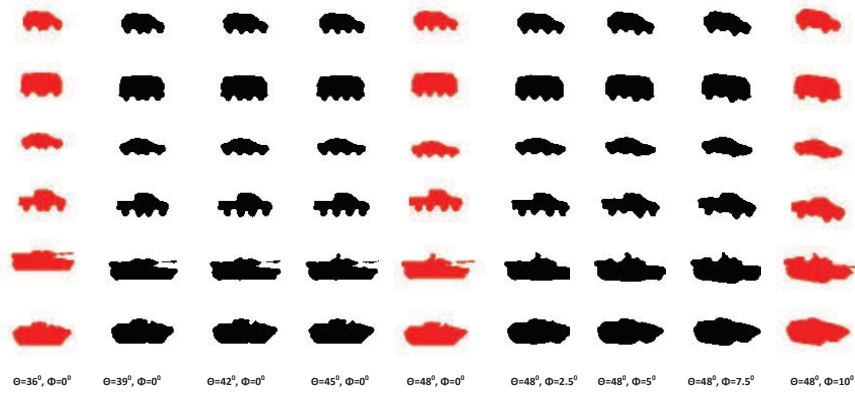
by the 2D DCT. Only about 10% DCT coefficients are used to represent a shape, which are sufficient for nearly lossless shape reconstruction. Another advantage of this shape descriptor is that we can do zero-padding prior to inverse DCT to accommodate an arbitrary scaling factor without additional zooming or shrinking operations.

**Figure 8.** All 36 CAD models used in this work, which are ordered according to the class-constrained shortest-closed-path [8]. (Reprint from [24], with permission from Elsevier.)

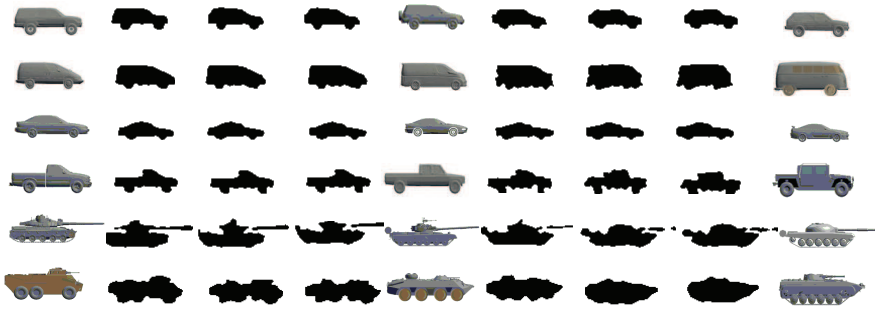


JVIM-based shape interpolation is demonstrated in Figure 9, which manifests its capability of handling a variety of target shapes with respect to viewpoint changes for a known target, as well as the generalization to previously unseen target types. In Figure 9a, we pick one target type from each of the six classes. For each target type, we can obtain an identity-specific view manifold from JVIM along which we can interpolate new target shapes of intermediate views (in black) between two training view-points. A smooth shape transition is observed across all interpolated shapes, despite the strong nonlinearity of training shapes. Figure 9b shows the shape interpolation results (in black) along the view-independent identity manifold for the same side view. Although the interpolated shapes are not as smooth as previous ones, most of them are still meaningful, with a mixed nature of two adjacent training target types along the identity manifold. Compared to CVIM in [8], which assumes that the identity and view manifolds are independent, JVIM shows better shape interpolation results by imposing a conditional dependency between the two manifolds and is also more computationally efficient due to local inference. A detailed comparison can be found in [26], where JVIM is found to be advantageous over CVIM and several GPLVM-based shape models, both qualitatively and quantitatively.

**Figure 9.** Qualitative analysis of JVIM shape interpolation: (a) along six identity-specific view manifolds. (b) along the view-independent identity manifold between two training target types. (Reprint from [24], with permission from Elsevier.)



(a)



(b)

## 7.2. Infrared ATR Database and Shape Models

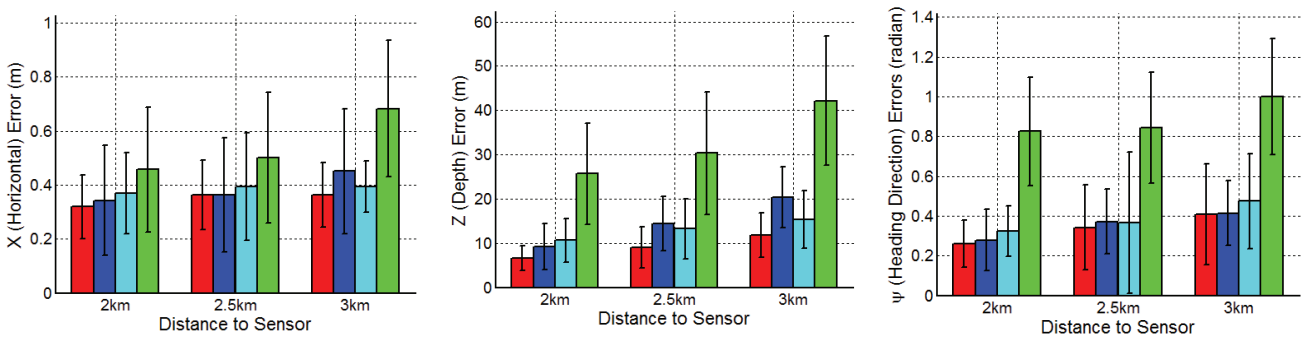
We have obtained a set of mid-wave IR sequences from the SENSIAC ATR database [11], which includes IR imagery of civilian and military ground vehicles maneuvering around a closed-circular path at ranges from 1–3 km. Forty sequences from eight different target types at ranges of 1.0 km, 1.5 km, 2.0 km, 2.5 km and 3 km were selected for this work. For each sequence, tracking was performed on 500 frames. Background subtraction [30] was applied to each frame for clutter rejection, which is needed for two competing algorithms involving explicit shape matching. For each tracking method, the particle filter was initialized with the ground-truth in the first frame. Similar to [8], the process noise of the heading direction  $\zeta_t^\psi$  is assumed to be a non-zero mean Gaussian to accommodate the circular moving trajectory which is necessary due to the ill-posed nature of image-based 3D tracking. This assumption can be relaxed if 3D pose estimation is not needed. Using the metadata provided with the database and a calibrated camera model, we computed the 3D ground-truth of position and aspect angle (in the sensor-centered coordinate system) for each frame. We refer the readers to [26] for more details about the ATR database.

In the following infrared ATR evaluation, we compare JVIM with LL-GPLVM [9] and CVIM [8], as well as the traditional nearest neighbor shape interpolation (NN). Both JVIM and CVIM treat shape factors (view and identity) continuously. To make a fair comparison, we learned a set of target-specific view manifolds by using LL-GPLVM, which involves a hemisphere as the topology constraint for manifold-based shape modeling. Then, we augment a “virtual” circular-shaped identity manifold (similar to that in JVIM and CVIM) for LL-GPLVM, where a NN method is used to “interpolate” arbitrary target types via training ones. Likewise, two “virtual manifolds” are introduced for the NN-based shape model, where we use the nearest neighbor to find the best matched training shapes. Thus, the two shape variables for four shape models can be inferred in a similar continuous way during ATR inference.

### 7.3. ATR with Explicit Shape Matching

We adopted the particle filter-based ATR algorithm used in [8], where JVIM, CVIM, LL-GPLVM and NN are evaluated in the case of explicit shape matching. In the CVIM-based ATR algorithm, two independent dynamical models are used. In JVIM-based tracking, the dynamic model is a two-stage one, where the first stage is along the view-independent identity manifold, while the second stage along the identity-dependent view manifold. For the LL-GPLVM-based ATR algorithm, one dynamic model is defined on each target-specific view manifold and one on the virtual identity manifold, where NN is used for identity interpolation. For the NN-based ATR algorithm, we employ two dynamic models on two virtual manifolds, like those in CVIM, where shape interpolation is done via NN (*i.e.*, just using the training shapes).

**Figure 10.** Comparison of the tracking errors of the horizontal position, slant range and heading direction. In each plot, the results for each method averaged over eight target types for each range. From left to right, the plot gives the results for JVIM (first, red), couplet of view and identity manifolds (CVIM) (second, blue), local linear (LL)-Gaussian process latent variable model (GPLVM) (third, cyan) and nearest neighbor (NN) (forth, green). (Reprint from [24], with permission from Elsevier).



The ATR performance of four shape models was evaluated with respect to three figures of merit: (1)  $p_x$  (horizontal) position error (in meters); (2)  $p_z$  (slant range) position error (in meters); and (3) heading direction error  $\psi$  (in rads). Quantitative tracking performance results are reported in

Figure 10, which give the horizontal, slant range and heading direction tracking errors, respectively, averaged over the eight target types for each range. It is shown that JVIM gains 9%, 10% and 35% improvements over CVIM, LL-GPLVM and NN along the horizontal direction, respectively, 35%, 31% and 72% along the slant range, respectively, and 5%, 13% and 62% along the heading direction, respectively. The results demonstrate that JVIM delivers better tracking performance with respect to all three figures of merit, with the advantage over CVIM, LL-GPLVM and NN being particularly significant for the range estimation.

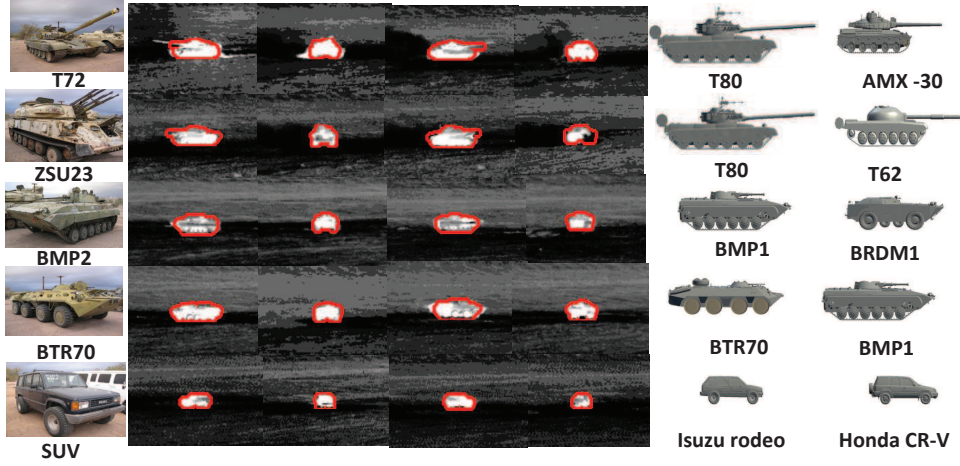
#### 7.4. ATR-Seg with Implicit Shape Matching

The proposed ATR-Seg algorithm (noted as Method I in the following) was tested against 15 SENSIAC sequences of five targets (SUV, BMP2, BTR70, T72 and ZSU23) under three ranges (1 km, 1.5 km and 2 km). Two more methods, Method II (JVIM with explicit shape matching, [23,26]) and Method III (CVIM [8]), were considered for comparison. All methods share a similar inference algorithm shown in Figure 7. Both Methods II and III involve explicit shape matching, and JVIM was used for both Methods I and II, while CVIM was used for method III. The tracking results are shown in Table 3. Results for tanks were averaged over T72 and ZSU23, and those for APCs averaged over BTR70 and BMP2. It is shown that Method I outperformed Methods II and III by providing lower tracking errors. More importantly, unlike Methods II and III, which require target pre-segmentation, Method I accomplishes target segmentation along with tracking and recognition as a by-product.

During tracking, the target identity is also estimated frame-by-frame by three methods, and the recognition accuracy is calculated as the percentage of frames where the target types were correctly classified in terms of the six target classes. The overall recognition results of three methods are shown in Table 4, where all methods perform well, and Method I (ATR-Seg) still slightly and moderately outperforms Methods II and III, respectively. Especially, when the range is large, e.g., 2 km, the advantage of Method I over Method III is more significant. This is mainly due to the fact that target segmentation is less reliable when the target is small.

The tracking, recognition and segmentation results of Method I (ATR-Seg) against five 1.5-km sequences were shown in Figure 11, where the two best matched target types are presented to show sub-class target recognition. As shown in Figure 11 (the fourth tracking result of ZSU23), part of ZSU23 is missing during tracking; the proposed method still can give an accurate segmentation and tracking result. ATR-Seg uses the intensity information from the present frame to build the energy term in Equation (20) that reduces the error accumulation over time and then evaluates how likely a hypothesized shape created by JVIM can segment a valid target at the predicted position. On the other hand, Method III in [8] uses the background subtraction results and involves an explicit shape comparison for evaluation, so the tracking and recognition results highly depend on the pre-segmentation results.

**Figure 11.** ATR-Seg results for five IR sequences. Column 1: truth target types. Columns 2–5: selected IR frames overlaid with the segmentation results. Columns 6–7: the two best matched training targets along the identity manifold. (Reprint from [28], with permission from IEEE.)



**Table 3.** Tracking errors for three ATR methods (Method I/Method II/Method III). (Reprint from [28], with permission from IEEE).

Range	Error in	Tank	APC	SUV	Total
1 km	$p_x$ (m)	0.22/0.25/0.22	0.22/0.25/0.18	0.16/0.17/0.19	0.21/0.23/ <b>0.20</b>
	$p_z$ (m)	5.06/8.67/7.53	4.19/4.48/5.14	9.03/8.36/10.95	<b>5.51</b> /6.93/7.26
	$\psi$ (rad)	0.13/0.17/0.18	0.15/0.32/0.15	0.11/0.24/0.22	<b>0.13</b> /0.24/0.18
1.5 km	$p_x$ (m)	0.24/0.19/0.18	0.15/0.21/0.20	0.16/0.56/0.60	<b>0.27</b> /0.27/0.27
	$p_z$ (m)	4.40/7.20/7.28	4.70/5.88/5.96	—NA—	<b>4.55</b> /6.54/6.26
	$\psi$ (rad)	0.16/0.22/0.24	0.18/0.53/0.51	0.11/0.32/0.35	<b>0.20</b> /0.36/0.37
2 km	$p_x$ (m)	0.31/0.27/0.28	0.23/0.19/0.36	0.13/0.17/0.35	0.24/ <b>0.22</b> /0.32
	$p_z$ (m)	8.68/10.6/8.58	8.95/9.28/7.95	5.35/8.09/14.25	<b>8.19</b> /9.55/9.46
	$\psi$ (rad)	0.19/0.38/0.26	0.41/0.18/0.38	0.08/0.41/0.31	<b>0.26</b> /0.31/0.32

**Table 4.** Recognition accuracy (%) for Methods I, II and III. (Reprint from [28], with permission from IEEE).

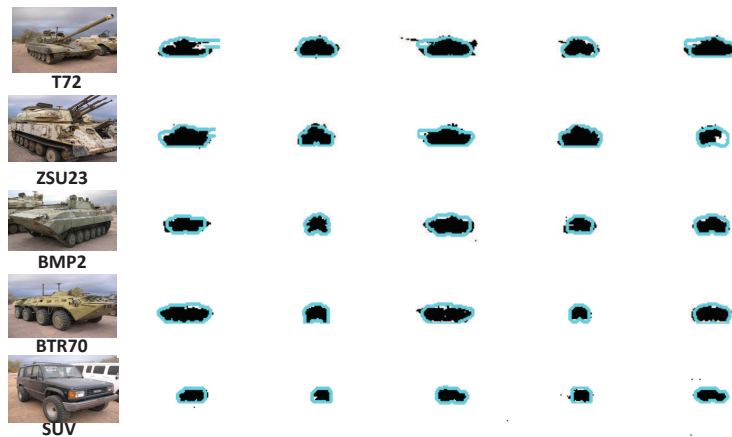
Targets	Tanks	APCs	SUV	Total
1 km	100/96/96	100/100/94	100/100/100	<b>100</b> /98/96
1.5 km	98/96/94	99/100/89	100/100/100	<b>99</b> /98/93
2 km	98/92/86	100/100/85	100/100/98	<b>99</b> /98/88



### 7.5. ATR-Seg Segmentation Results

We evaluated the segmentation performance of ATR-Seg using the metric of the overlap ratio. The ground-truth segmentation results were generated manually for five randomly selected frames in each of 15 sequences. For a comparison, we also computed the overlap ratios for background subtraction results, which are averaged around 81%. While those of ATR-Seg are averaged around 85%. It is worth noting that the segmentation results of ATR-Seg are essentially constrained by the training shapes created from the CAD models, and the training models may have some shape discrepancy with the observed targets in the SENSIAC data. Another source of segmentation errors is due to tracking errors. Some segmentation results of five targets at 1.5 km were shown in Figure 12, where we overlaid the ATR-Seg results (contours) over the ground-truth ones. Background subtraction is not easy for a moving platform and is susceptible to the occlusion problem, while ATR-Seg is more flexible and robust to the sensor ego-motion and has great potential for occlusion handling, due to the shape prior involved [6,7].

**Figure 12.** Segmentation results of five targets at the range of 1.5 km.

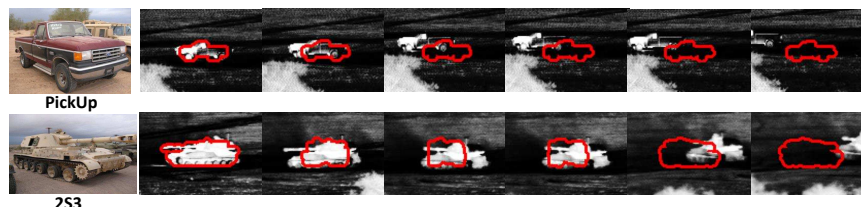


### 7.6. Limitation and Discussion

There are two limitations of ATR-Seg due to the unsupervised nature of the level set, where no prior is used for foreground/background pixel intensities, and the mismatching between the training targets and the test ones. Thus, when a major part of a target is occluded or part of a target is similar to the background, the shape-aware set will lose the sensitivity for segmentation evaluation, leading to tracking failure, as shown in Figure 13 (first row), which shows the failed results for the pick-up sequence at 1.5 km. The mismatching and the low-quality data are the main reasons for the tracking failure of 2S3 at a range of 1.5 km (second row in Figure 13). One possible remedy is to incorporate some pixel priors of background and foreground into the level set energy function. However, an online learning scheme may be needed to update the pixel priors that are usually necessary for a long infrared sequence [31]. It is worth emphasizing that the goal of this work is to test the potential of a “model-based” approach that only uses CAD models for training. It is a natural extension

to incorporate real infrared data for training that is likely to improve the algorithm robustness and applicability significantly.

**Figure 13.** Tracking failure for the pick-up and 2S3 sequences at 1.5 km.



## 8. Conclusion

A new algorithm, called ATR-Seg, is proposed for joint target tracking, recognition and segmentation in infrared imagery, which has three major technical components. First is a novel GPLVM-based shape generative model, the joint view-identity manifold (JVIM), which unifies one view-independent identity manifold and infinite identity-dependent view manifolds jointly in a semantically meaningful latent space. Second is the incorporation of a shape-aware level set energy function that evaluates how likely a valid target can be segmented by a shape synthesized by JVIM. Third, a particle filter-based sequential inference algorithm is developed to jointly accomplish target tracking, recognition and segmentation. Specifically, the level set energy function is used as the likelihood function in the particle filter that performs implicit shape matching, and a general motion model is involved to accommodate a highly maneuvering target. Experimental results on the recent SENSIAC ATR database manifest the advantage of ATR-Seg over two existing methods using explicit shape matching. This work is mainly focused on a shape-based approach. One possible future research issue is to involve other visual cues, such as pixel intensities or textures, to enhance the sensitivity and discriminability of the shape-aware level set energy function, which could mitigate the limitations of the ATR-Seg algorithm.

## Author Contributions

This work was supported in part by the U.S. Army Research Laboratory (ARL) and U.S. Army Research Office (ARO) under grant W911NF-04-1-0221 and the Oklahoma Center for the Advancement of Science and Technology (OCAST) under grant HR12-30.

## Conflicts of Interest

The authors declare no conflicts of interest.

## References

1. Yilmaz, A.; Javed, O.; Shah, M. Object tracking: A survey. *ACM Comput. Surv.* **2006**, *38*, 1–45.

2. Fan, X.; Fan, G.; Havilcek, J. Generative Models for Maneuvering Target Tracking. *IEEE Trans. Aerospace Electron. Syst.* **2010**, *46*, 635–655.
3. Srivastava, A. Bayesian filtering for tracking pose and location of rigid targets. *Proc. SPIE* **2000**, *4052*, 160–171.
4. Shaik, J.; Iftexharuddin, K. Automated tracking and classification of infrared images. In Proceedings of International Joint Conference on Neural Networks, Portland, OR, USA, 20–24 July 2003; Volume 2, pp. 1201–1206.
5. Khan, Z.; Gu, I.H. Tracking visual and infrared objects using joint Riemannian manifold appearance and affine shape modeling. In Proceedings of IEEE International Conference on Computer Vision Workshops (ICCVW), Barcelona, Spain, 6–13 November 2011.
6. Prisacariu, V.; Reid, I. Nonlinear shape manifolds as shape priors in level set segmentation and tracking. In Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Colorado Springs, CO, USA, 20–25 June 2011; pp. 2185–2192.
7. Prisacariu, V.; Reid, I. Shared shape spaces. In Proceedings of IEEE International Conference on Computer Vision (ICCV), Barcelona, Spain, 6–13 November 2011; pp. 2587–2594.
8. Venkataraman, V.; Fan, G.; Yu, L.; Zhang, X.; Liu, W.; Havlicek, J.P. Automated Target Tracking and Recognition using Coupled View and Identity Manifolds for Shape Representation. *EURASIP J. Adv. Signal Process.* **2011**, *124*, 1–17.
9. Urtasun, R.; Fleet, D.J.; Geiger, A.; Popović, J.; Darrell, T.J.; Lawrence, N.D. Topologically-constrained latent variable models. In Proceedings of International Conference on Machine learning (ICML), Helsinki, Finland, 5–9 July 2008; pp. 1080–1087.
10. Yao, A.; Gall, J.; Gool, L.; Urtasun, R. Learning Probabilistic Non-Linear Latent Variable Models for Tracking Complex Activities. In Proceedings of Annual Conference on Neural Information Processing Systems (NIPS), Granada, Spain, 12–14 December 2011; pp. 1–9.
11. Military Sensing Information Analysis Center (SENSIAC). Available online: <https://www.sensiac.org/> (accessed on 12 December 2012).
12. Yankov, D.; Keogh, E. Manifold Clustering of Shapes. In Proceedings of International Conference on Data Mining, Hong Kong, China, 18–22 December 2006.
13. Roweis, S.T.; Saul, L.K. Nonlinear dimensionality Reduction by Local Linear Embedding. *Science* **2000**, *290*, 2323–2326.
14. Etyngier, P.; Segonne, F.; Keriven, R. Shape Priors using Manifold Learning Techniques. In Proceedings of IEEE International Conference on Computer Vision (ICCV), Rio de Janeiro, Brazil, 14–20 October 2007.
15. Etyngier, P.; Keriven, R.; Segonne, F. Projection onto a Shape Manifold for Image Segmentation with Prior. In Proceedings of IEEE International Conference on Image Processing (ICIP), San Antonio, TX, USA, 16–19 October 2007; Volume 4, pp. 361–364.
16. He, R.; Lei, Z.; Yuan, X.; Li, S. Regularized active shape model for shape alignment. In Proceedings of IEEE International Conference on Automatic Face Gesture Recognition (FG), Amsterdam, the Netherlands, 17–19 September 2008.

17. Lüthi, M.; Albrecht, T.; Vetter, T. Probabilistic Modeling and Visualization of the Flexibility in Morphable Models. In Proceedings of IMA International Conference on Mathematics of Surfaces XIII, York, UK, 7–9 September 2009; Volume 5654, pp. 251–264.
18. Dambreville, S.; Rathi, Y.; Tannenbaum, A. A Framework for Image Segmentation Using Shape Models and Kernel Space Shape Priors. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *30*, 1385–1399.
19. Lawrence, N. Probabilistic Non-linear Principal Component Analysis with Gaussian Process Latent Variable Models. *J. Mach. Learn. Res.* **2005**, *6*, 1783–1816.
20. Elgammal, A.; Lee, C.S. Separating style and content on a nonlinear manifold. In Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Washington, DC, USA, 27 June–2 July 2004; Volume 1, pp. 478–485.
21. Lee, C.; Elgammal, A. Modeling View and Posture Manifolds for Tracking. In Proceedings of IEEE International Conference on Computer Vision, Rio de Janeiro, Brazil, 14–21 October 2007.
22. Li, X.R.; Jilkov, V. Survey of maneuvering target tracking. Part I. Dynamic models. *IEEE Trans. Aerospace Electron. Syst.* **2003**, *39*, 1333–1364.
23. Gong, J.; Fan, G.; Yu, L.; Havlicek, J.; Chen, D. Joint view-identity manifold for target tracking and recognition. In Proceedings of 2012 19th IEEE International Conference on Image Processing (ICIP), Orlando, FL, USA, 30 September–3 October 2012; pp. 1357–1360.
24. Gong, J.; Fan, G.; Yu, L.; Havlicek, J.P.; Chen, D.; Fan, N. Joint view-identity manifold for infrared target tracking and recognition. *Comput. Vis. Image Underst.* **2014**, *118*, 211–224.
25. Urtasun, R.; Darrell, T. Sparse probabilistic regression for activity-independent human pose inference. In Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Anchorage, AK, USA, 23–28 June 2008.
26. Gong, J.; Fan, G.; Yu, L.; Havlicek, J.P.; Chen, D.; Fan, N. Joint View-Identity Manifold for Infrared Target Tracking and Recognition. *Comput. Vis. Image Underst.* **2014**, *118*, 211–224.
27. Bibby, C.; Reid, I. Robust Real-Time Visual Tracking Using Pixel-Wise Posteriors. In Proceedings of the 10th European Conference on Computer Vision: Part II, Marseille, France, 12–18 October 2008; pp. 831–844.
28. Gong, J.; Fan, G.; Havlicek, J.P.; Fan, N.; Chen, D. Infrared Target Tracking, Recognition and Segmentation using Shape-Aware Level Set. In Proceedings of IEEE International Conference on Image Processing (ICIP), Melbourne, Australia, 15–18 September 2013.
29. Arulampalam, S.; Maskell, S.; Gordon, N.; Clapp, T. A Tutorial on Particle Filters for Online Non-linear/Non-Gaussian Bayesian Tracking. *IEEE Trans. Signal Process.* **2002**, *50*, 174–188.
30. Zivkovic, Z.; van der Heijden, F. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognit. Lett.* **2006**, *27*, 773–780.
31. Venkataraman, V.; Fan, G.; Havlicek, J.; Fan, X.; Zhai, Y.; Yeary, M. Adaptive Kalman Filtering for Histogram-based Appearance Learning in Infrared Imagery. *IEEE Trans. Image Process.* **2012**, *21*, 4622–4635.

# Automatic Association of Chats and Video Tracks for Activity Learning and Recognition in Aerial Video Surveillance

Riad I. Hammoud, Cem S. Sahin, Erik P. Blasch, Bradley J. Rhodes and Tao Wang

**Abstract:** We describe two advanced video analysis techniques, including video-indexed by voice annotations (VIVA) and multi-media indexing and explorer (MINER). VIVA utilizes analyst call-outs (ACOs) in the form of chat messages (voice-to-text) to associate labels with video target tracks, to designate spatial-temporal activity boundaries and to augment video tracking in challenging scenarios. Challenging scenarios include low-resolution sensors, moving targets and target trajectories obscured by natural and man-made clutter. MINER includes: (1) a fusion of graphical track and text data using probabilistic methods; (2) an activity pattern learning framework to support querying an index of activities of interest (AOIs) and targets of interest (TOIs) by movement type and geolocation; and (3) a user interface to support streaming multi-intelligence data processing. We also present an activity pattern learning framework that uses the multi-source associated data as training to index a large archive of full-motion videos (FMV). VIVA and MINER examples are demonstrated for wide aerial/overhead imagery over common data sets affording an improvement in tracking from video data alone, leading to 84% detection with modest misdetection/false alarm results due to the complexity of the scenario. The novel use of ACOs and chat messages in video tracking paves the way for user interaction, correction and preparation of situation awareness reports.

Reprinted from *Sensors*. Cite as: Hammoud, R.I.; Sahin, C.S.; Blasch, E.P.; Rhodes, B.J.; Wang, T. Automatic Association of Chats and Video Tracks for Activity Learning and Recognition in Aerial Video Surveillance. *Sensors* **2014**, *14*, 19843–19860.

## 1. Introduction

Streaming airborne wide area motion imagery (WAMI) and full-motion video (FMV) sensor collections afford online analysis for various surveillance applications, such as crowded traffic scene monitoring [1]. In a layered sensing framework, such sensors may be used to simultaneously observe a region of interest to provide complimentary capabilities, including improved resolution for target discrimination, identification and tracking [2]. Typically, forensic analysis, including pattern-of-life detection and activity/event recognition, is conducted off line due to huge volumes of imagery. This big data outpaces users' available time to watch all videos in searching for key activity patterns within the data. To aid users in detecting patterns in aerial imagery, robust and efficient computer vision, pattern analysis and data mining tools are highly desired [3].

### *1.1. Multi-Source Data and Problem Statement*

For data collection and reporting, the aerial video is reviewed by humans (called hereafter reviewed FMV data), as the imagery is streamed down from an airborne platform. During a real-time FMV exploitation process, humans could call out significant AOIs, where a voice-to-text tool converts audible analyst call-outs (ACO) to text (see example in Figure 1), and a computer then saves the ACOs to storage disks along with the aerial imagery. Additional contextual information besides ACOs includes additional reviewers' (internal) chat, as well as discussions about the area of coverage of the overhead video from external sources. Together, the ACOs, internal discussions and external perspectives provide a collective set of "chat messages" [4].

However, these two data sources (chat messages and FMV) are not synchronized in time or in space. They are not recorded with corresponding time stamps. Furthermore, the called-out targets and activities are not marked in video frames with bounding boxes or with a start and an end of each activity. It is worth noting that the use of ACOs radically differs from a traditional video annotation paradigm that is typically done manually for training and/or benchmarking of computer vision algorithms. The incorporation of the user's ACO requires advances in automation, human-machine interaction and multimodal fusion. In addition, during the overhead imagery review process, there is no advanced equipment, such as an eye tracker [5] or touch screen employed to determine screen locations of the targets of interest (TOIs) corresponding to ACOs.

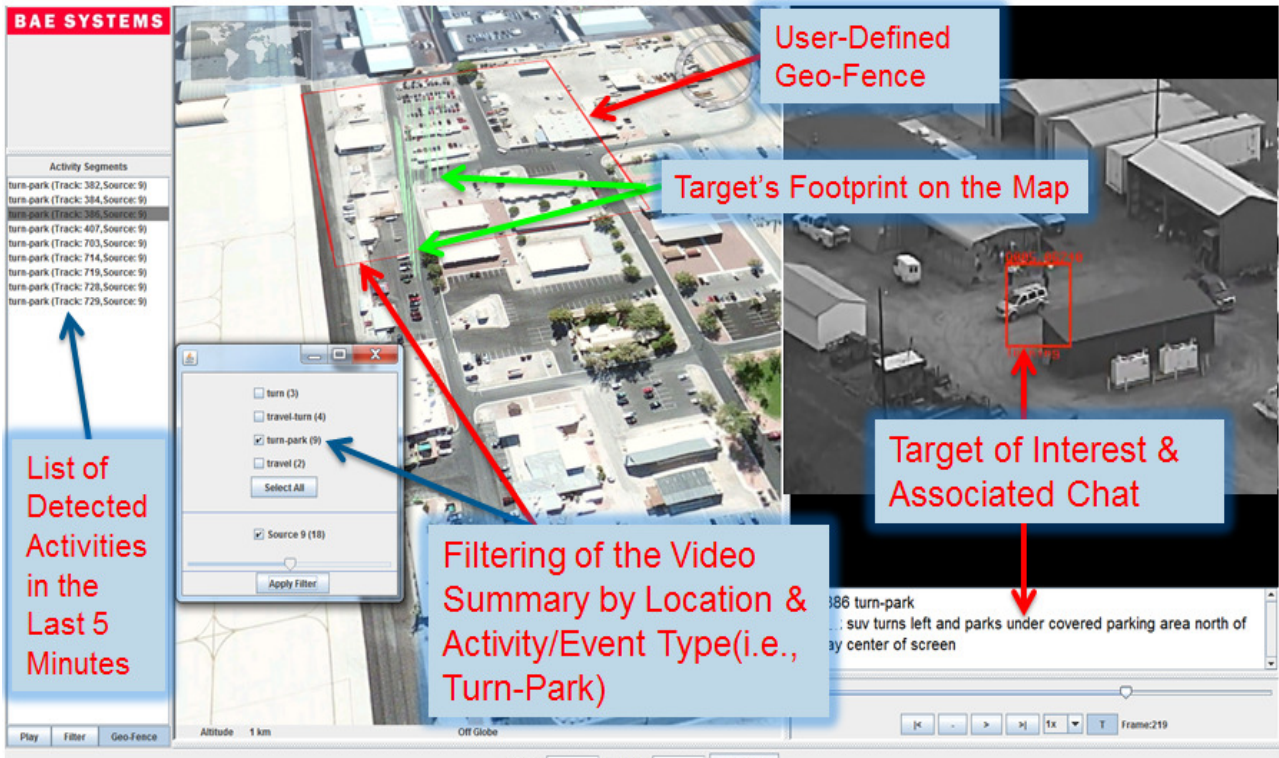
### *1.2. Paper Contributions*

The ACO messages present a rich source of information allowing for a fast retrieval of activities and providing a summary of events over FMV. ACOs can provide a reference ground truth of the AOIs that occur in the reviewed FMV data. Hence, correlating these two data sources would produce four novel products: (1) a video summary of AOIs/TOIs allowing nonlinear browsing of video content; (2) annotated text-over-video media where only TOIs are highlighted with bounding boxes and synchronized with chat messages; (3) an activities index where linked events are grouped together; and (4) adaptive data play back allowing for user-selected filtering by geographic location. For instance, the end user may submit a query like this: pull out all video segments of activity types "turn then stop" near this house on the map (see Figure 1).

---

"Non-Technical Data —Releasable to Foreign Persons."

**Figure 1.** Illustration of the event-report visualization interface (multi-media indexing and explorer (MINER)) allowing users to visualize and query correlated chats, pattern of life and activity-labeled track segments.



In this paper, we propose a multi-source probabilistic graph-based association framework to automatically: (1) identify TOIs corresponding to chat messages; (2) detect activity boundaries (*i.e.*, segmenting FMV tracks into semantic sub-tracks/segments); (3) learn activity patterns in low-level feature spaces using the reviewed FMV data; (4) index non-reviewed FMV data (*i.e.*, archived videos); as well as (5) assist FMV analysts with tools for fast querying and non-linear browsing of multi-source data.

Such an automatic linking process of multi-source data enhances data association by eliminating the tedious process of manually collecting and correlating the data. As a side benefit, pattern recognition typically requires training data for activity pattern learning; however, the chat messages provide a notional real-time training template. The need for activity training data has been well reported in the literature. For instance, [6] emphasizes the need to collect high-quality activity/event examples with minimal irrelevant pixels for the activity learning modules. Furthermore, during the manual annotation process, Oh *et al.* [6] define very specifically the start and end moments of activities to ensure proper learning on non-noisy data. Here, we demonstrate a paradigm shift in tracking and classification of imagery that does not require training data for real-world deployment of methods.

### 1.3. Paper Organization

Section 2 details related work. The following sections describe various components of our “video-indexed by voice annotations” (VIVA) system. Section 3 provides a video processing overview with extensions to our methods. Section 4 describes the mapping of a single FMV target track to multiple graphs of attributes. In Section 4.2, we describe our two-step algorithm to decompose a single track into semantic segments. Section 5 focuses on parsing of chat messages (or ACO) and their graphical representation. In Section 6, we present the multi-source graph-based association framework and the activity class assignment process. In Section 7, we briefly provide an overview of our approach for learning activity patterns from the reviewed FMV tracks (*i.e.*, training data) and querying the unlabeled FMV data. Sections 8 and 9 outline our multi-media indexing and explorer (MINER) interface and evaluate several scenarios to provide performance details of the proposed framework, respectively. We conclude the paper in Section 10.

## 2. Related Work

Visual activity recognition—the automatic process of recognizing semantic spatio-temporal target patterns, such as “person carrying” and “vehicle u-turn” from video data—has been an active research area in the computer vision community for many years [7]. Recently, the focus in the community has shifted toward recognizing activities/actions over large time scales, wide-area spatial resolutions [8] and multi-source multimodal frequencies in real-world operating conditions [9]. We assume here that a pattern is bounded by event changes, and target movement in between events is an “activity.” In such conditions, the major challenge arises from the large intra-class variations in activities/events, including variations in sensors (e.g., viewpoints, low resolution and scale), target (e.g., visual appearance, speed of motion) and environment (e.g., lighting condition, occlusion and clutter). The recognition of activities in overhead imagery poses many more challenges than from a fixed ground-level camera, mostly because of the imagery’s low resolution. Additionally, the need for video stabilization creates noise, tracking and segmentation difficulties for activity recognition.

The key algorithmic steps in visual activity recognition techniques are: (1) extracting spatio-temporal interest point detectors and descriptors [10]; (2) performing clustering (e.g., K-means) in the feature space (e.g., histogram of oriented gradients (HOG), histogram of flow (HOF), histogram of spatio-temporal gradients (3D-STHOG) and 3D-SIFT) to form codebooks after principal component analysis (PCA)-based dimension reduction; and (3) labeling tracks using a bag-of-words approach [6,11]. We follow a similar process when it comes to learning activity patterns from the reviewed FMV tracks. That being said, we first perform multi-source data association to generate training data from the reviewed FMV tracks where FMV tracks are assigned activity labels.

Xiey *et al.* [12] proposed a method for discovering meaningful structures in video through unsupervised learning of temporal clusters and associating the structures with metadata. For a news-domain model, they presented a co-occurrence analysis among structures and observed



that temporal models are indeed better at capturing the semantics than non-temporal clusters. Using data from digital TV news, Duygulu and Wactlar [13] proposed a framework to determine the correspondences between the video frames and associated text in order to annotate the video frames with more reliable labels and descriptions. The semantic labeling of videos enables a textual query to return more relevant corresponding images and enables an image-based query response to provide more meaningful descriptors (*i.e.*, content-based image retrieval). Our proposed activity recognition framework discovers meaningful activity structures (e.g., semantically-labeled events, activities, patterns) from overhead imagery over challenging scenarios in both reviewed and non-reviewed FMV data.

### 3. Multiple Target Tracking and Classification

Tracking multiple targets in aerial imagery requires first stabilizing the imagery and then detecting automatically any moving target. In this section, we briefly describe these techniques along with our automatic target recognition method.

#### 3.1. Video Stabilization

Our frame-to-frame stabilization module aligns successive image frames to compensate for camera motion [14]. There are several steps involved in our two-frame registration process: (1) extracting interest points from the previous image that possess enough texture and contrast to distinguish them from one another; and (2) matching the 2D locations of these points between frames using a robust correspondence algorithm. Establishing correspondences consists of two stages: (1) use “guesses,” or putative matches, established by correlating regions around pairs of feature points across images; and (2) performing outlier rejection with random sample consensus (RANSAC) to remove bad guesses.

The VIVA stabilization algorithm runs in real time on “commercial, off-the-shelf” (COTS) hardware, and it was specifically designed to be robust against large motions between frames. The enhanced robustness against large motion changes is essential, since analog transmission of electro-optical/infrared (EO/IR) airborne data to the ground can be corrupted, frames can be dropped, time-delays long and can vary in sample rates. As long as the two frames being registered have greater than a 35% overlap, we are usually able to establish enough correspondences for reliable stabilization.

#### 3.2. Target Detection and Tracking

Our moving target tracking algorithm, cluster objects using recognized sequence of estimates (COURSE), makes few assumptions about the scene content, operates almost exclusively in the focal plane domain and exploits the spatial and temporal coherence of the video data. It consists

---

“Non-Technical Data —Releasable to Foreign Persons.”

“Non-Technical Data —Releasable to Foreign Persons.”

of three processing steps. First, the frame-to-frame registration is used to find regions of the image where pixel intensities differ (this is done through frame differencing (see Figure 2)). Underlying frame differencing is the assumption that pixel intensity differences are due to objects that do not fit the global image motion model. Clearly, other effects, such as parallax, also cause false differences, but these false movers are filtered using subsequent motion analysis. Second, point features with a high pixel intensity difference are used to establish correspondences between other points in the previous frame, which produces a set of point-velocity pairs. Third, these point-velocity pairs are clustered into motion regions that we assume are due to individual targets. Regions that persist over time are reported as multiple target detections. The tracker provides two very important capabilities: (i) it removes false detections generated by the upstream target detection module; and (ii) extends detection associations beyond what can be accomplished by using only the image-based target detection module. COURSE achieves enhanced robustness by (i) removing isolated detections that are inconsistent with the presence of a moving object, and (ii) exploiting large time-event information to deal with brief interruptions caused by minor occlusions such as trees or passing cars. The COURSE tracker generates a mosaic tracking report (see Figures 3 and 4) to be used as input to our multi-source association framework.

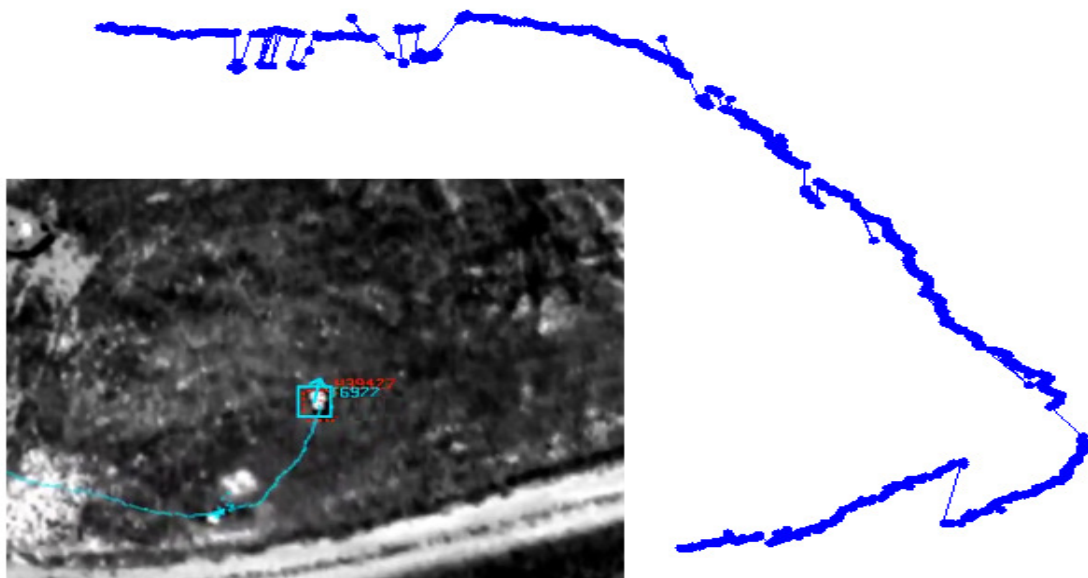
**Figure 2.** Video-indexed by voice annotations (VIVA)’s movement detection module. First, registered frames (**top left**) are differenced to produce a change detection image (**lower right**). That image is thresholded to detect changing pixels. Point correspondences within those detection pixels are established between the two frames and used to generate motion clusters (**right**).



**Figure 3.** Example of track profiles of vehicles generated by COURSE using sample videos from the VIRATAerial dataset (ApHill) [6].



**Figure 4.** Illustration of a noisy tracking trajectory of a single dismount (from the ApHill VIRAT aerial dataset) generated by COURSE. The track is broken into several segments (*i.e.*, several tracking labels) due to quick changes in motion direction, cluttered background and multiple stop-and-move scenarios.



### 3.3. Target Classification

In order to reduce the ambiguity in the multi-source association framework, we classify each target into “person,” “vehicle” or “others”. We employed support vector machines (SVM) with a radial basis function (RBF) to train models and classify unlabeled targets into these three categories. During training, we used a five-fold cross-validation process to find the best values for the radius of the RBF and the cost factor, which controls the importance of the training error with respect to the separation margin [15].

We extracted HOG and HOF to characterize the low resolution targets [10]. The HOG preserves some texture and local structure of the targets and is invariant to illumination changes. In contrast to HOG, the HOF features capture the motion information of the moving target. Once the supervised learning is completed, we classify every target track into one of the three categories using the majority vote from all individual frames of a track.

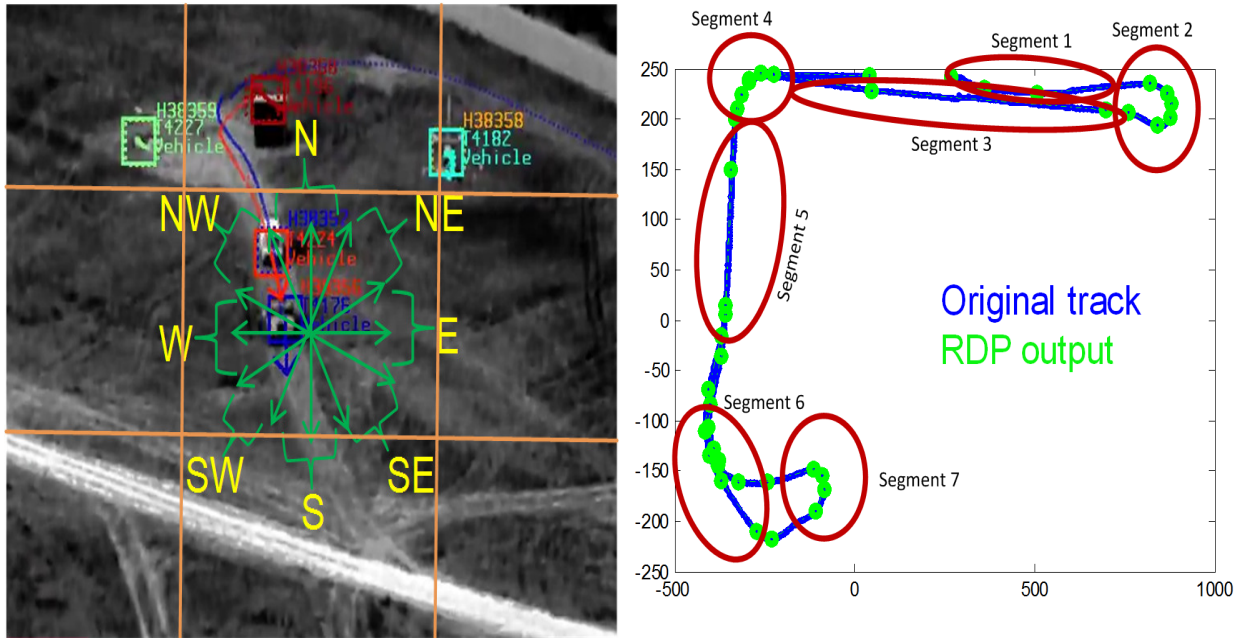
## 4. Multi-Graph Representation of a Single FMV Track

The multi-source association framework is based on a graph representation and matching of target tracks and chat messages. In this section, we describe how to build a graph-based model of a tracked target and how to divide long and informative tracks into semantic track segments and, hence, represent a single track with multiple graphs.

### 4.1. Mapping Tracks to Graphs

Each target track is cast by a combination of graphs where nodes represent the targets’ attributes and edges characterize the relationship between nodes. We divided attributes into common and uncommon based on their saliency over the lifetime of a target track. For instance, the color and shape of a vehicle remain unchanged, while direction and spatial location vary over time ( $t$ ). The TOIs are classified into “vehicle” *vs.* “human” (*i.e.*, actor attribute) based on motion, blob size and shape. The shape attribute is divided into “car” *vs.* support utility vehicle “SUV” *vs.* “truck” for vehicle, and “adult” *vs.* “child” for human actor/dismount [16]. Each actor is characterized with a unique color attribute (e.g., black truck, human with red-shirt, *etc.*) and a spatial location (*i.e.*,  $xy_s$  position on the screen and latitude/longitude on the geographic map). The location is mapped into gross zones (see Figure 5a) on the screen to match with gross locations in the chat messages. We divided the video frame into a  $3 \times 3$  grid (center screen, top left, *etc.*). The direction attribute is derived from the velocity vectors ( $V_x(t)$ ,  $V_y(t)$ ) at time  $t$ , such that  $\theta(t) = \arctan(\frac{V_y(t)}{V_x(t)})$ , which, in turn, is mapped to a geographical direction using the gross divisions of directions, as shown in Figure 5a. In order to reduce noise in the mapping of  $\theta$  and  $xy_s$  to gross direction and location zones, we applied a sliding window to smooth these values over time. The last attribute is mobility, which specifies whether the target is moving or stationary ( $m_t$ ).

**Figure 5.** Illustration of our assignment of tracking states into (a) direction and location zones (e.g., south-east direction, top-left screen zone, *etc.*) and (b) semantic segments based on changes in direction and speed using RDP.



#### 4.2. Dividing Tracks into Semantic Segments

When a track exhibits major changes in uncommon attributes, especially in direction, location and speed, it becomes necessary to break it down into multiple semantic segments, which results in multiple graphs in the association framework. Track segmentation into graphs is needed when multiple chats correspond to a single track generated by our video tracker. Figure 5b shows three minutes of a tracked vehicle moving toward the east, making a u-turn then moving toward the west. We apply a two-step algorithm to break down tracks into semantic segments:

- (1) Smooth the tracking locations  $(xy_s)$  using the Ramer–Douglas–Peucker (RDP) algorithm [17]. RDP will produce a short list of un-noisy position points  $(XY_s)$  (displayed as green points in Figure 5b).
- (2) Detect directional changes computed from  $XY_s(t)$  points. The beginning of a new semantic track segment is marked when a peak is detected. The end of a new semantic segment is flagged when the second derivative of  $XY_s(t)$  is near zero. Figure 5b illustrates the results of this step, where seven segments were detected.

## 5. Parsing and Graph Representation of Chats

In our data collection setup, the chat messages follow the following format for a target of type vehicle [18]:

```
At <time> <quantity> <color> <vehicle>
<activity> <direction> <location>
where:
<time> = 0000Z - 2359Z
<activity> = (travel | u-turn ...)
<direction> = (north | south ...)
<location> = screen (middle | left ...)
<color> = (red | black ...)
<shape> = (truck | car ...)
```

Basic search for keywords in a chat message is employed to extract relevant information, such as “activity type,” “direction,” and “location.” In our dataset, we have nine activities (vehicle turn, u-turn, human walking, running, *etc.*; see Section 9), eight direction zones (north, south, *etc.*) and nine location zones (middle, top-left screen zone, *etc.*; see Figure 5).

These chat messages represent an analyst calling out activities in the FMV, intra-viewer discussions or other related external discussions. In turn, a chat message is represented as a graph of attributes. However, more elaborated information extraction (IE) from a chat message (*i.e.*, micro-text) or a document (e.g., using Sphinx or Apache NLP) as an automated approach [19–22] could be employed to handle misspelled words and larger dictionaries.

**Figure 6.** Example of representation of a video track (a) and a chat message (b) as graphs.

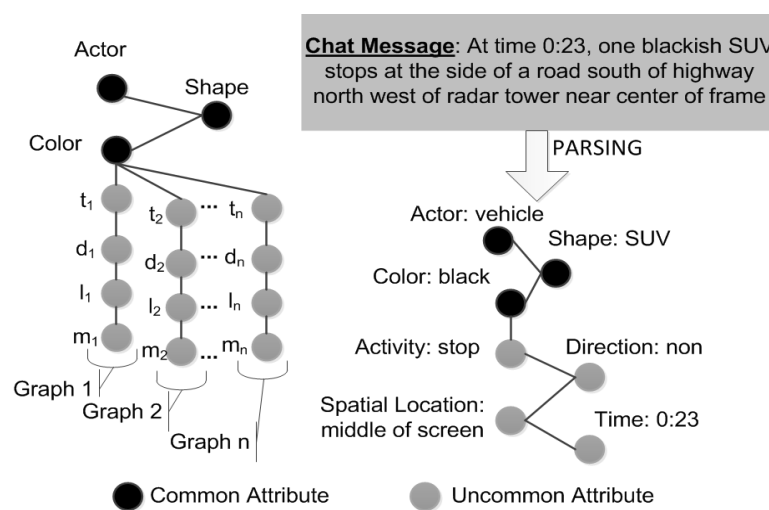


Figure 6b illustrates a chat message decomposed into multimodal attributes. An example can come from any modality (e.g., video, text, radar, *etc.*), so the goal is to decompose the data into these meaningful parts [4].

## 6. Multi-Source Graph Association and Activity Class Assignment

A mission goal includes allowing the image processing method to answer a user-defined query. The user calling out significant activities in the image would desire an automated processor to match the target being called out to that of a TOI. Within an image in a video stream, there could be many movers, targets and events happening. The system must choose the TOI among several tracked objects in the imagery that corresponds to a meaningful content (attributes) in the chat message by a user. Because users review FMV tracks from streaming airborne video, the callouts flag AOIs. Association between reviewed FMV tracks and chat messages can be achieved by performing probabilistic matching between graphs from both data sources. It is important to note that, as explained in the Introduction, a chat message is the only source to describe the true activity of the TOI. By performing multi-source graph-based association, the true activity of the TOI is mapped to a corresponding track segment from FMV.

The multi-source multimodal association framework consists of the following stages:

1. In a given time interval,  $[t - T, t + T]$  (with  $t$  the time stamp from a chat message and  $T$  : a predefined time window to search for the tracked objects), the chat message and all video tracks are extracted from the data sets.
2. Graph representations of video tracks and chat messages are generated as explained in Sections 4 and 5.
3. Graph matching uses a probabilistic distance measure (see Equation (1)) of ensemble similarity between a chat message ( $j$ ) and track segment ( $i$ ). There are three main reasons to use a probabilistic distance metric: (i) to associate the graphs even if there are missing attributes; (ii) to reduce the effects of errors coming from the video processor and chat messages (e.g., a user may assign a vehicle color as black, while a tracked object from the video processor might be marked as gray); and (iii) to impute the weights of attributes based on the quality of videos. The associated graphs with the highest likelihoods are assigned as a match.

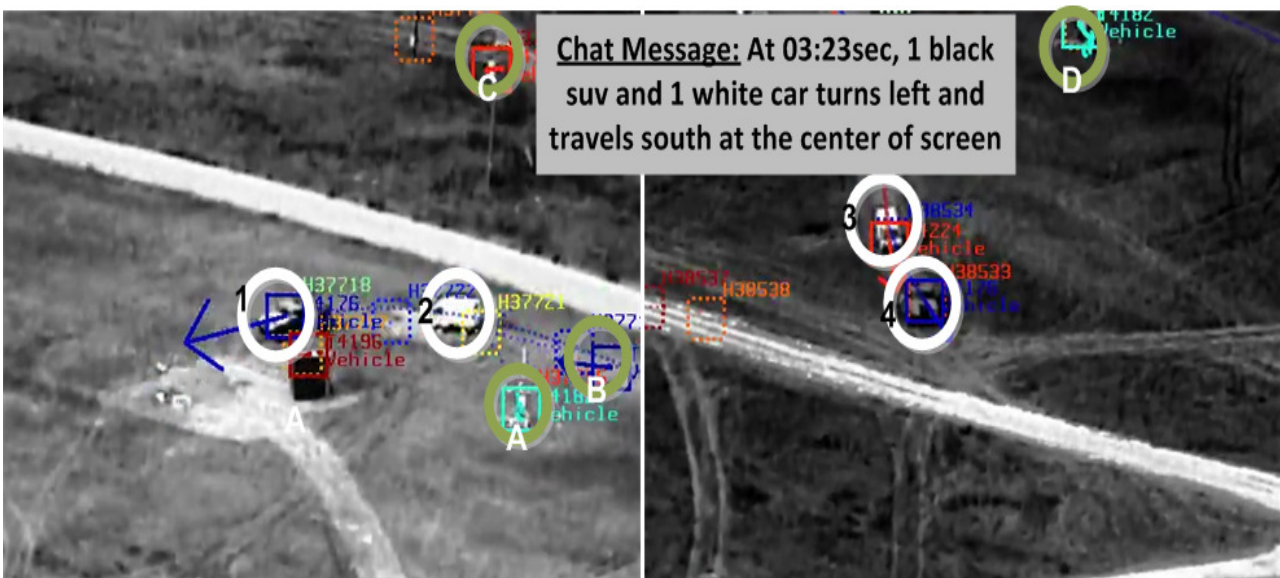
$$P(T_i|C_j, c_i) = w_a P_a + w_s P_s + w_t P_t + w_{cl} P_{cl} + w_d P_d + w_l P_l + w_{cn} P_{cn} + w_m P_m \quad (1)$$

where  $w_a$ ,  $w_s$ ,  $w_t$ ,  $w_{cl}$ ,  $w_d$ ,  $w_l$ ,  $w_{cn}$  and  $w_m$  are the user-defined weights of attributes for actor, shape, time, color, direction, spatial location, tracking confidence and target mobility, respectively.  $P_a$ ,  $P_s$ ,  $P_t$ ,  $P_{cl}$ ,  $P_d$ ,  $P_l$ , and  $P_m$  represent the probabilities of corresponding attributes, and  $P_{cn}$  is the track confidence value generated by COURSE.

An illustrative result of this framework is shown in Figure 7. This framework handles one-to-one, one-to- $N$ , and  $N$ -to- $M$  association cases. Furthermore, this framework not only marks the TOI, but

also the rendering of activities. Using labeled track profiles, the boundaries of each activity are determined by using the process described in Section 4.2. For example, after labeling each track segment by associating chat messages, track Segments 1, 3 and 5 are marked as travel, Segments 2 and 7 are u-turn and track Segments 4 and 6 are labeled as turn in Figure 5b.

**Figure 7.** Successful identifications of AOIs/TOIs in exemplar clips from the ApHill VIRAT aerial dataset using our multi-source association framework. (a) and (b) show multiple vehicle tracks and a single chat message being called-out; the tracks in white circles (1, 2, 3 and 4) were highly matched with the chat message graphs, while targets in green circles (A, B, C and D) scored low matching probabilities.



## 7. Learning Activity Patterns from Multi-Source Associated Data

The chat messages provide the pseudo ground truth of the AOIs occurring in the reviewed FMV video (see Section 6). These correlated data also serve as training data for activity pattern learning in aerial imagery. Here, we employ BAE Systems' Multi-intelligence Activity Pattern Learning and Exploitation (MAPLE) tool, which uses the hyper-elliptical learning and matching (HELM) unsupervised clustering algorithm [23] to learn activity patterns. This is done through extracting features from each labeled track segment, clustering features in each activity space and finally representing each track by a sequence of clusters (*i.e.*, chain code). In terms of features, we used simple descriptors for vehicles, including speed, heading relative to segment start and a position eigenvalue ratio. By measuring the change relative to a fixed starting value, rather than the instantaneous change, the heading feature is robust to variations in how quickly the targets turns from its initial course. The position eigenvalue ratio is a measure of the mobility of the target, which is the ratio of eigenvalues calculated from the target's position within a short time duration. As for



people tracking, we compute the histogram of motion flow and neighboring intensity variance, which describes the extent to which the target is moving toward or away from potential interaction sites.

The goal of the learning process is to be able to match an unlabeled track (*i.e.*, without chat) to the indexed pattern of life trajectories over large amounts of non-reviewed FMV data. First, we use HELM to classify each instance of a new track to one of the clusters of the index. Second, we use temporal gradient matching distance to obtain matches between the representation of the new track and the indexed learned patterns. The similarity score between a new track  $j$  and an index  $i$  is defined as follows:

$$\sigma_{ij} = \frac{t_{ij} + c_{ij}}{2} \quad (2)$$

where  $t_{ij}$  represents the similarity metric, which considers only the common clusters, and  $c_{ij}$  is the similarity score of temporal gradient for the cluster sequence.

## 8. Event/Activity Report Visualization and Querying by Activity Type and Geolocation

The VIVA and MINER framework produces three useful products for the end users to visualize in the same interface (see Figure 1): (1) a video summary of AOIs allowing nonlinear browsing of video content; (2) text-over-video media where only TOIs are highlighted with bounding boxes and synchronized with chat messages, which describe their activities; and (3) an index of activities. The benefit of the compiled index of videos is that a user (or machine) could find related content over a geographic location and text query. For instance, the end-user may submit a query like this: pull-out all video segments of activity types “turn then stop” near this house with specific latitude and longitude coordinates. We converted each track to geo-tracks using both meta-data and reference imagery. If the AOI is detected in archived video using the activity classification framework presented above, the chat panel in our MINER interface shows the automatically generated description (*i.e.*, target category and activity type).

## 9. Experimental Results

To validate the proposed framework, we used our own dataset consisting of EO/IRairborne videos (60 min long) and 100 chat messages. The activity list is limited to vehicle travel, stop, turn, u-turn, maintain-distance, accelerate and decelerate and human walking and running. The VIVA video tracker generated about 8000 different tracks. The percentage of false tracking is about 15 percent and could be reduced through fusion of both EO and IR [24]. This is mainly due to camera zoom-ins and zoom-outs, which happen frequently when the camera focuses on TOIs to enhance resolution.

Each object was automatically classified into one of the three categories: “person,” “vehicle” or “others.” These models were trained offline using 34,681 instances/frames of 514 tracked targets. We obtained 72.16% and 76.99% correct classification in HOF and HOG spaces, respectively. After majority voting, the track-wise classification accuracy increased to 86.58% and 92.61% in

HOF and HOG, respectively. The biggest confusion is between humans and “others”, due to the low resolution of human targets. Most of the false targets that should be classified as “others” are due to noisy registration, platform motion and inaccuracy of the tracking bounding boxes, which often included parts of the background.

Specific low-level features (see Section 4.1) were computed prior to running the proposed multi-source multimodal association framework. Table 1 summarizes the results of the association framework. Correct associations are marked when the tracks or sub-tracks (*i.e.*, semantic segments) in the overhead imagery are associated with their corresponding chat messages. A false association is flagged when the chat message is linked to the wrong target or semantic segment (see Section 4.2). This could occur when multiple targets are moving in the same area at the same time and in the same direction. A miss is defined as a chat message without an associated target in the overhead imagery. On this data set, we scored 83.77% correct association, 6.39% misses association and 9.84% wrong association (*i.e.*, false alarms). During these experiments, we set the time window in which to perform the multi-source associations to 15 seconds. Making this window shorter leads to less false alarms, but also a higher miss rate. Furthermore, we only used the target’s direction, location and speed as attributes, which do not include other useful content to reduce false alarms.

**Table 1.** Qualitative assessment of the multi-graph association and activity class assignment framework.

Detection	Miss	False
83.77%	6.39%	9.84%

The association framework for activity recognition handles complex scenarios with multiple tracked objects. Figure 7a and b shows eight different tracks (different track labels) of six moving objects and a single chat message called out within the same time window. The chat message is parsed automatically into four different graphs, which are matched to all ten graphs representing the video tracks. The additional two video graphs (initially, we got eight tracks) came out from the splitting process of a single track into semantic segments (or sub-tracks, as described in Section 4.2), due to changes in vehicle direction while traveling. The VIVA framework associated the four chat graphs to the correct four FMV semantic tracks due to strong matches between common attributes. Our approach was also challenged by broken tracks (e.g., the case of a dismount/TOI with three different tracking labels in Figure 4). In spite the fact that the same TOI is represented by three consecutive tracks, VIVA provides correct associations with event boundaries (*i.e.*, shorter and semantic track segments). Thus, it is robust to scenario variations.

These preliminary results are very promising. Both the direction and the location attributes play an important role in the association of chat messages to tracks. The list of potential matches is reduced drastically using these attributes. Nevertheless, in order to make a one-to-one association,

additional attributes, such as shape, color and size, and spatial relationships, such as a target near an identifiable landmark in the scene, would be very helpful to resolve association ambiguities. Due to the chat description, the extracted target’s direction and location are cast to gross zones (*i.e.*, middle screen region, northeast direction, *etc.*) rather than fine ranges, causing ambiguities in the association. Extracting buildings from available imagery [25] would greatly benefit the association, because the chats refer to such attributes when describing activities involving human-object interactions.

We used the multi-source associated data to learn activity patterns and then to index non-reviewed data (see Section 7). We tested this framework on 54 new track segments/activities and obtained 79.6% correct activity label assignment (see Table 2). Figure 8 shows a track segment from an unlabeled video correctly matched to a u-turn pattern model with a highest matching score  $\sigma_{q,uTurn} \approx 1.0$  compared to other models. It is worth mentioning that the automatically generated training data from the multi-source association framework is not noise-free with a near 10% false classification (see Table 1). Add to that the errors that come from the automatic target classification and event boundary detection. Further, misclassified targets as vehicles add noise to the pattern learning process. It is necessary to have the human in the loop to correct the automatically generated training data prior to the activity pattern learning process. Additionally, having more training data will ensure building reliable pattern activity models in challenging operating conditions with enough intra-class variations using high dimensional activity descriptors over a larger activity list.

**Table 2.** Qualitative assessment of the classification of unlabeled tracks into the seven learned activity patterns.

Correct	False
79.6%	20.4%

**Figure 8.** Illustration of an exemplar target track (from the ApHill VIRAT aerial dataset) being matched to the proper activity pattern model (a u-turn in this example) learned using the training data generated by the proposed multi-source association approach.



## 10. Conclusions

In this paper, we developed a novel concept for the graphical fusion of video and text data to enhance activity analysis from aerial imagery. We detailed the various components, including the VIVA association framework, the COURSE tracker, the MAPLE learning tool and the MINER visualization interface. Given the exemplar proof of concept, we highlighted the benefits for a user in reviewing, annotating and reporting on video content. Future work will explore the metrics and associations used to increase robustness and to reduce false alarms. However, it is noted that the end user can check the final results presented in our MINER interface to remove false alarms and effortlessly generate mission reports.

## Acknowledgments

This work was supported under contract number FA8750-13-C-0099 from the Air Force Research laboratory. The ideas and opinions expressed here are not official policies of the United States Air Force.

The authors would like to thank Adnan Bubalo (AFRL), Robert Biehl, Brad Galego, Helen Webb and Michael Schneider (BAE Systems) for their support.

## Author Contributions

Riad I. Hammoud served as Principal Investigator on this AFRL program. His contributions to this paper includes architecting the overall activity recognition and multi-source association framework, designing/implementing the FMV exploitation algorithms, and architecting the MINER interface.

Cem Sahin's contributions includes development of the graph-based association framework and the event break detection algorithm, and testing the activity pattern learning system on various data sets.

Erik Blasch's contributions includes selecting relevant features used in the multi-source association framework, designing the experiments and helping with the write-up of the paper.

Bradley Rhodes helped with the design of the activity pattern learning framework.

Tao Wang implemented the ATR module of the VIVA system.

## Conflicts of Interest

The authors declare no conflict of interest.

---

“Non-Technical Data —Releasable to Foreign Persons.”

“Non-Technical Data —Releasable to Foreign Persons.”

## References

1. Maurin, B.; Masoud, O.; Papanikolopoulos, N. Camera surveillance of crowded traffic scenes. In Proceedings of the ITS America 12th Annual Meeting, Long Beach, CA, USA, 29 April–2 May 2002; p. 28.
2. Brown, A.P.; Sheffler, M.J.; Dunn, K.E. Persistent Electro-Optical/Infrared Wide-Area Sensor Exploitation. *Proc. SPIE* **2012**, doi:10.1117/12.922167.
3. Fan, G.; Hammoud, R.I.; Sadjadi, F.; Kamgar-Parsi, B. Special section on Advances in Machine Vision Beyond the Visible Spectrum. *Comput. Vision Image Underst. J.* **2013**, *117*, 1645–1646.
4. Blasch, E.; Nagy, J.; Aved, A.; Pottenger, W.; Schneider, M.; Hammoud, R.; Jones, E.; Basharat, A.; Hoogs, A.; Chen, G.; *et al.* Context aided Video-to-text Information Fusion. In Proceedings of International Conference on Information Fusion, Salamanca, Spain, 7–10 July 2014; pp. 1–8.
5. Hammoud, R.I. *Passive Eye Monitoring: Algorithms, Applications and Experiments*; Springer: Berlin/Heidelberg, Germany, 2008.
6. Oh, S.; Hoogs, A.; Perera, A.; Cuntoor, N.; Chen, C.C.; Lee, J.T.; Mukherjee, S.; Aggarwal, J.K.; Lee, H.; Davis, L.; *et al.* A Large-scale Benchmark Dataset for Event Recognition in Surveillance Video. In Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 20–25 June 2011.
7. Liu, J.; Ali, S.; Shah, M. Recognizing human actions using multiple features. In Proceedings of the IEEE CVPR, Anchorage, AK, USA, 23–28 June 2008.
8. Gao, J.; Ling, H.; Blasch, E.; Pham, K.; Wang, Z.; Chen, G. Pattern of Life from WAMI Objects Tracking Based on Visual Context-aware Tracking and Infusion Network Models. *Proc. SPIE* **2013**, doi:10.1117/12.2015612.
9. Kahler, B.; Blasch, E. Sensor Management Fusion Using Operating Conditions. In Proceedings of IEEE National Aerospace and Electronics Conference, Dayton, OH, USA, 16–18 July 2008.
10. Dollar, P.; Rabaud, V.; Cottrell, G.; Belongie, S. Behavior recognition via sparse spatio-temporal features. In Proceedings of the 2nd Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VSPETS), Beijing, China, 15–16 October 2005.
11. Laptev, I.; Marszalek, M.; Schmid, C.; Rozenfeld, B. Learning realistic human actions from movies. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008.
12. Xiey, L.; Kennedy, L.; Changy, S.F.; Divakaranx, A.; Sunx, H.; Linz, C.Y. Discovering Meaningful Multimedia Patterns With Audio-Visual Concepts AND Associated Text. In Proceedings of the 2004 International Conference on Image Processing, ICIP, Singapore, 24–27 October 2004.

13. Duygulu, P.; Wactlar, H.D. Associating video frames with text. In Proceedings of the 26th Annual International ACM SIGIR Conference, Toronto, ON, Canada, 28 July–1 August 2003.
14. Seetharaman, G.; Gasperas, G.; Palaniappan, K. A piecewise affine model for image registration in nonrigid motion analysis. In Proceedings of the 2000 International Conference on Image Processing, Vancouver, BC, Canada, 10–13 September 2000; pp. 561–564.
15. Chang, C.C.; Lin, C.J. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol. (TIST)* **2011**, *2*, 1–27.
16. Blasch, E.; Ling, H.; Wu, Y.; Seetharaman, G.; Talbert, M.; Bai, L.; Chen, G. Dismount Tracking and Identification from Electro-Optical Imagery. *Proc. SPIE* **2012**, doi:10.1117/12.919025.
17. Prasad, D.K. Assessing Error Bound for Dominant Point Detection. *Int. J. Image Process.* **2012**, *6*, 326–333.
18. Blasch, E.; Wang, Z.; Ling, H.; Palaniappan, K.; Chen, G.; Shen, D.; Aved, A.; Seetharaman, G. Video-based Activity Analysis Using the L1 Tracker on VIRAT Data. In Proceedings of the IEEE Applied Imagery Pattern Recognition Workshop, Washington, DC, USA, 23–25 October 2013.
19. Nadeau, D.; Sekine, S. A Survey of Named Entity Recognition and Classification. *Linguisticae Investig.* **2007**, *30*, 3–26.
20. Cambria, E.; White, B. Jumping NLP Curves: A Review of Natural Language Processing Research. *IEEE Comput. Intell. Mag.* **2014**, *9*, 1–28.
21. Kulekci, M.O.; Oflazer, K. An Overview of Natural Language Processing Techniques in Text-to-Speech Systems. In Proceedings of the IEEE Conference on Signal Processing and Communications Applications, Aydin, Turkey, 28–30 April 2004.
22. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent Dirichlet Allocation. *J. Mach. Learn. Res.* **2003**, *3*, 993–1022.
23. Rhodes, B.J.; Bomberger, N.A.; Zandipour, M.; Stolzar, L.H.; Garagic, D.; Dankert, J.R.; Seibert, M. Anomaly Detection and Behavior Prediction: Higher-Level Fusion Based on Computational Neuroscientific Principles. *Sens. Data Fusion* **2009**, doi:10.5772/6585.
24. Leykin, A.; Hammoud, R.I. Pedestrian tracking by fusion of thermal-visible surveillance videos. *J. Mach. Vision Appl.* **2010**, *21*, 587–595.
25. Hammoud, R.I.; Kuzdeba, S.A.; Berard, B.; Tom, V.; Ivey, R.; Bostwick, R.; HandUber, J.; Vinciguerra, L.; Shnidman, N.; Smiley, B. Overhead-Based Image and Video Geo-localization Framework. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops, Portland, OR, USA, 23–28 June 2013; pp. 320–327.

# Small Infrared Target Detection by Region-Adaptive Clutter Rejection for Sea-Based Infrared Search and Track

Sungho Kim and Joohyoung Lee

**Abstract:** This paper presents a region-adaptive clutter rejection method for small target detection in sea-based infrared search and track. In the real world, clutter normally generates many false detections that impede the deployment of such detection systems. Incoming targets (missiles, boats, *etc.*) can be located in the sky, horizon and sea regions, which have different types of clutters, such as clouds, a horizontal line and sea-glint. The characteristics of regional clutter were analyzed after the geometrical analysis-based region segmentation. The false detections caused by cloud clutter were removed by the spatial attribute-based classification. Those by the horizontal line were removed using the heterogeneous background removal filter. False alarms by sun-glint were rejected using the temporal consistency filter, which is the most difficult part. The experimental results of the various cluttered background sequences show that the proposed region adaptive clutter rejection method produces fewer false alarms than that of the mean subtraction filter (MSF) with an acceptable degradation detection rate.

Reprinted from *Sensors*. Cite as: Kim, S.; Lee, J. Small Infrared Target Detection by Region-Adaptive Clutter Rejection for Sea-Based Infrared Search and Track. *Sensors* **2014**, *14*, 13210–13242.

## 1. Introduction

Sea-based infrared search and track (IRST) systems are wide field-of-view or omni-directional surveillance systems designed for autonomous search, detection, acquisition, track and designation of potential targets, as shown in Figure 1 [1,2]. The most important threats in sea-based IRST are incoming small targets, such as anti-ship sea-skimming missiles (ASSM) or asymmetric ships. In these applications, targets are typically unresolved and appear in the sky and sea backgrounds with a resolution of only a few pixels. Normally, a small infrared target's size is less than 100 pixels [3]. The important performance parameters of the target detection system consist of the radiant intensity of a target, detection distance, detection rate and false alarm rate. If the radiant intensity of a target and a minimal detection distance are determined, the detection algorithm should be able to detect true targets to satisfy the systems' detection rate and reject false targets as much as possible.

**Figure 1.** Operational concept of sea-based infrared search and track (IRST).

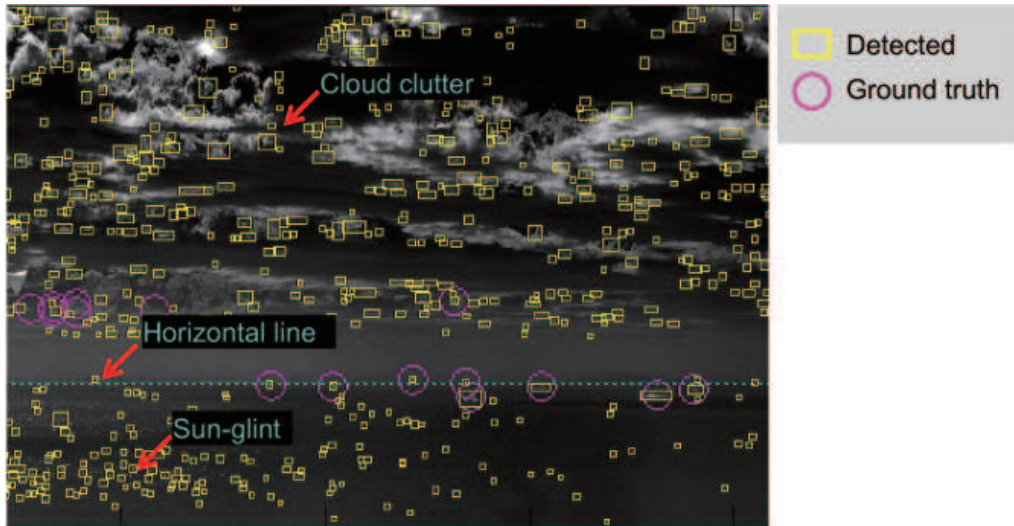


The detection of long-range, small targets is quite difficult, because of the small and dim target signal. The criteria of the detection rate can be achieved by lowering the detection threshold. On the other hand, such simple approaches lead to an increased number of false detections due to background clutters. Figure 2 shows the problems of the conventional small target detection method using the well-known modified mean subtraction filter (MMSF) [4]. The edge around cloud clutter can generate false detections. The horizontal edge line due to a heterogeneous background produces false detections. Finally, sun-glint has a similar shape (circular symmetry) to small targets and a high intensity value, which hinders true target detection. Such regional clutter produces many false alarms, which hinders true target detection.

This study examined how to make a small target detection method practical by reducing the number of false detections caused by different types of clutter, such as clouds in the sky, the edge line on the horizon and sun-glint in the sea surface region, in an integrated manner. According to geometric analysis, background images were segmented into the sky region, horizontal region and sea surface region. This paper proposes a region-adaptable clutter rejection scheme by careful observation and analysis of the clutter behavior. False detections around cloud clutter were removed by learning-based classification. The false detections around the horizon region were removed by subtracting the heterogeneous background. Finally, those around the sea surface region were removed by a temporal consistency filter. Therefore, the contributions from this study can be summarized as follows. The first contribution is the automatic region (sky-horizon-sea) segmentation by geometric analysis, which is an essential step in the clutter rejection system. The regions were segmented using the horizontal line estimated by the sensor pose-based prediction and image-based line fitting. The second contribution is the proposed region-adaptive false detection rejection scheme based on the analysis results. The third contribution is the demonstration of the proposed method using infrared test sequences by a comparison with the conventional detection method.



**Figure 2.** Problems of the conventional spatial filter-based, small target detection method. Many false detections are generated by regional clutter, such as clouds, horizon and sun-glint.



Section 2 reviews some related works on detecting small infrared targets focusing on the false alarm reduction aspect and analyzing the disadvantages of the related well-known methods of detecting small targets in heterogeneous backgrounds. Section 3 analyzes the target position in an infrared image based on the target type and incoming scenario. Section 4 introduces the overall system structure and presents the novel region adaptive clutter rejection methods. In Section 5, a range of performance evaluations and results are explained. Section 6 reports a discussion of the results with the conclusions.

## 2. Related Works in Terms of Clutter Rejection

Many studies have evaluated small infrared target detection methods over the past 20 years. This section reviews the related papers in terms of their use of information, such as target information, background information, visual context and decision information, to reduce the number of false alarms, as shown in Table 1, where the total sum of statistics is 100%. For example, a cause of false alarms due to clouds can be handled using the spatial information (14.2%) of the background cue and the shape information (5.8%) of the target cue. As a second example, a cause of false alarms due to sun-glint can be handled using motion information (3.5%) of the target cue, a high-level classifier (2.8%) of the decision cue, frequency information (2.2%) of target cue, multi-sensor fusion (2.1%) of the context cue or temporal information (1.5%) of the background cue. The following subsections introduce false alarm reducing methods and the related papers for the cloud clutter and sun-glint.

**Table 1.** Statistics of research papers in terms of the causes of false alarms and the methods of overcoming them (%).

Causes of False Alarms		Dim	Noise	Background Clutter			Similar Object	
				Cloud	Glint	Ground	Bird	Buoy
Target cue	Intensity	0	0	1.4	0	0	0	0
	Shape	0	0.7	5.8	0.7	1.4	0	0
	Motion	5.7	2.1	5.8	<b>3.5</b>	3.5	2.1	2.1
	Distance	0	0	1.4	0	0	1.5	0
	Freq.	0	0.7	1.4	2.2	0	0	0
Background cue	Spatial	0	0	<b>14.2</b>	0	14.2	0	0
	Temporal	0.7	0	0	1.5	1.4	0	0
Context cue	Region	0.7	0	0.7	0.7	2.1	0	0
	Fusion	4.3	0	0.7	2.1	2.1	0	0
Decision cue	Threshold	1.4	0	0.7	0.7	2.8	0	0
	Classifier	0.7	0.7	1.4	2.8	1.4	0	0

### 2.1. Related Studies on Cloud Clutter Rejection

Several studies have examined the removal or reduction of false detections caused by clouds. Their false alarm reduction strategies were strongly dependent on the situation. If there is any assumption, background subtraction can be a feasible approach. The background image can be estimated from an input image using spatial filters, such as the least mean square (LMS) filter [5–7], mean filter [8], median filter [9] and morphological filter (Top-hat) [10,11]. The LMS filter minimizes the difference between the input image and background image, which is estimated by the weighted average of the neighboring pixels. The mean filter can estimate the background by the Gaussian mean or simple moving average. The median filter is based on the order statistics. The median value can remove point-like targets effectively. The morphological opening filter can remove the specific shapes by erosion and dilation with a specific structural element. The mean filter-based target detection is computationally very simple, but sensitive to edge clutter. Target detection with non-linear filters, such as the median or morphology filter, shows low false alarms around the edge, but is computationally complex. Combinational filters, such as max-mean or max-median, can preserve the edge information of cloud and background structures [12]. A data fitting approach, which models the background as multi-dimensional parameters, has also been reported [13]. The super-resolution method is useful in a background estimation, which enhances small target detection [14]. The filtering process of localized directional Laplacian-of-Gaussian (LoG) filtering and the minimum selection can then remove false detection around cloud edges, maintaining a small target detection capability [15].

If a sensor platform is static, the information regarding the fast target motion is enhanced by removing the slowly moving cloud clutter. A well-known approach is the track-before-detect (TBD) method [16,17]. The concept is similar to that of the 3D matched filter. Dynamic programming (DP), which is a quick version of the traditional TBD method, achieves good performance in detecting dim targets [18,19]. The temporal profiles, including the mean and variance, at each pixel are

effective in the detection of moving targets in slowly moving clouds [20–23]. Recently, the temporal contrast filter (TCF)-based method was developed to detect supersonic small infrared targets [24]. Accumulating the detection results of each frame makes it possible to detect moving targets [25]. The wide-to-exact search method was developed to enhance the speed of 3D matched filters [26]. Recently, an improved power-law-detector-based moving target detection method was presented; it was effective for image sequences that occur in heavy clutter [27].

Cloud clutter can also be reduced using decision methods. These decision methods need to determine that a probing region is a target. The hysteresis method has two thresholds. The first threshold is a very low value and is used to identify the candidate target regions. The second threshold possesses a relatively high value that depends on the operational requirements [28]. As information regarding the size becomes available, it is possible to remove large sun-glint and other large objects. Similar results can be obtained by applying an iterative threshold [29]. Statistics-based adaptive threshold methods, such as the constant false alarm rate (CFAR), are useful in a severely cluttered background [30,31]. The simplest classification method is the nearest neighbor classifier (NNC) algorithm, which uses only feature similarity [32]. In addition to NNC, there are model-based the Bayesian classifier [33], learning-based neural network, and support vector machine (SVM) [34] methods. Classification information can be useful for removing various clutter points.

## *2.2. Related Works on Sun-Glint Clutter Rejection*

Sun-glint clutter can be rejected using the TBD methods mentioned above. These approaches, however, assume a high frame rate to reduce sun-glint. If the frame rate is approximately 1 Hz, a new approach should be developed.

On the other hand, frequency domain approaches can be useful for removing low frequency clutter. The 3D-FFT spectrum-based approach shows a possible research direction in the target detection [35]. The wavelet transform extracts the spatial frequency information in an image pyramid, which shows robustness in sun-glint environments [36–38]. The low-pass filter (LPF)-based approach can also be robust to sensor noise and sun-glint [39]. Recently, an adaptive high-pass filter (HPF) was proposed to reduce cloud and sun-glint clutter [40].

While the target is in motion, the previous frame is considered a background image. Therefore, a background estimation can be performed using a weighted autocorrelation matrix update using the recursive technique [41]. Static clutter can also be removed by the frame difference [42]. An advanced adaptive spatial-temporal filter derived by the multi-parametric approximation of clutter can achieve tremendous gain compared to that of the spatial filtering method [43]. Principal component analysis (PCA) for multi-frames can remove temporal noise, such as sun-glint [44].

The information fusion approach can be useful for reducing sun-glint. This includes the target-background context, multi-feature context, multi-band context and multi-classification context. Those visual contexts are implemented in the form of information fusion that leads to clutter reduction and high detection rates. The target-background context concomitantly enhances the target signature and reduces the background clutter, leading to a reduction of sun-glint clutter [45]. Multi-feature fusion can improve the detection rate of dim targets [46,47]. If spectral fusion, such as the ratio of mid-wave infrared and long-wave infrared or a combination of the detection results from

both bands, is used, the sun-glint can be removed easily [48,49]. The voting of various classifiers can enhance the dim target detection rates [50].

### 3. Location Analysis of Incoming Targets

How can the target distance from a project target pixel be calculated? The target distance is a very important system parameter of IRST. According to previous analysis, the projective relationship among the camera height ( $h$ ), target distance ( $D$ ), target height ( $H$ ) and target positioning angle ( $\theta$ ) can be simplified as shown in Figure 3. In this scheme, the camera elevation angle ( $\alpha$ ) is assumed  $0^\circ$ . The target positioning angle can be estimated by the camera height and target distance, as expressed in Equation (1). If it is assumed that the camera's field of view (FOV) is  $6^\circ$  and the size of the IR detector is 480, the projected target position ( $i - th$  image row) can be calculated using Equation (2). Because this study was interested in the relationship between the row image position and target distance, the final projective relation can be obtained as Equation (3), which is derived from Equations (1) and (2). If it is assumed that the camera height is 20 m, the ship height is 0 m and the minimal target detection range is 9000 m, the ship target is projected into 10 pixels just below the horizontal line, as shown in Figure 4. In the case of a sea-skimming missile, of which the whole normal flying height is 200 m, the projected image is located just 10 pixels above the horizontal line at the minimal 8000-m detection. If the height ( $H$ ) of the ASSM is lower than the camera height ( $h$ ), the target is located around the horizontal line. As it approaches the camera, it appears on the sea surface. From such geometrical analysis related to the target types, it can be concluded that the distant targets are located around the horizontal line ( $\pm 20$  pixels centered on the horizontal line at 5000-m detection), and relatively close targets exist in the sky region or sea surface region. Therefore, it is necessary to segment an input image into the sky region, horizontal region and sea surface region.

$$\theta = \tan^{-1} \left( \frac{h - H}{D} \right) \times \frac{180}{\pi} \quad (1)$$

$$i = (\theta + 3) \times \frac{480}{6} \quad (2)$$

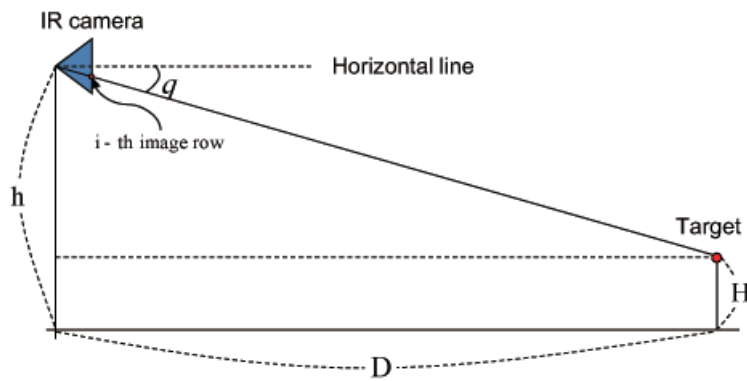
$$D = \frac{h - H}{\tan \left( \left( \frac{i}{80} - 3 \right) \times \frac{\pi}{180} \right)} \quad (3)$$

### 4. Proposed Small Target Detection with Region-Wise Clutter Rejection

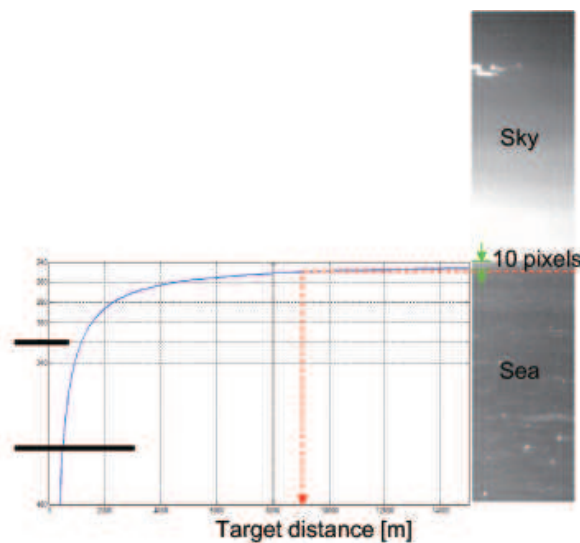
The proposed small target detection consists of background processing and target processing, as shown in Figure 5. The background processing module segments an input image into sky, horizon and sea region using the sensor pose information and image processing. The target processing module finds the candidate targets using a spatial filter and rejects any false alarms caused by background clutter using carefully-designed methods. The spatial filter (modified mean subtraction filter (MSF)) is commonly used in the entire region. Horizontal line clutter is estimated by a local directional background estimation (DBE) and removed. Small targets in the horizontal region are detected by the hysteresis threshold-based constant false alarm detector (H-CFAR). The candidate targets in the sky and sea regions are found by pre-detection. False detections in the sky region are generated by

clouds. Therefore, the target attribute-based classifier can reject false detections caused by cloud clutter. False detections by sea-glint in the sea region are rejected by a three-plot correlation and statistical filter. The following subsections introduce details of the region segmentation, removal of the horizontal line clutter in the horizon region, removal of cloud clutter in the sky region and removal of sea-glints in the sea region.

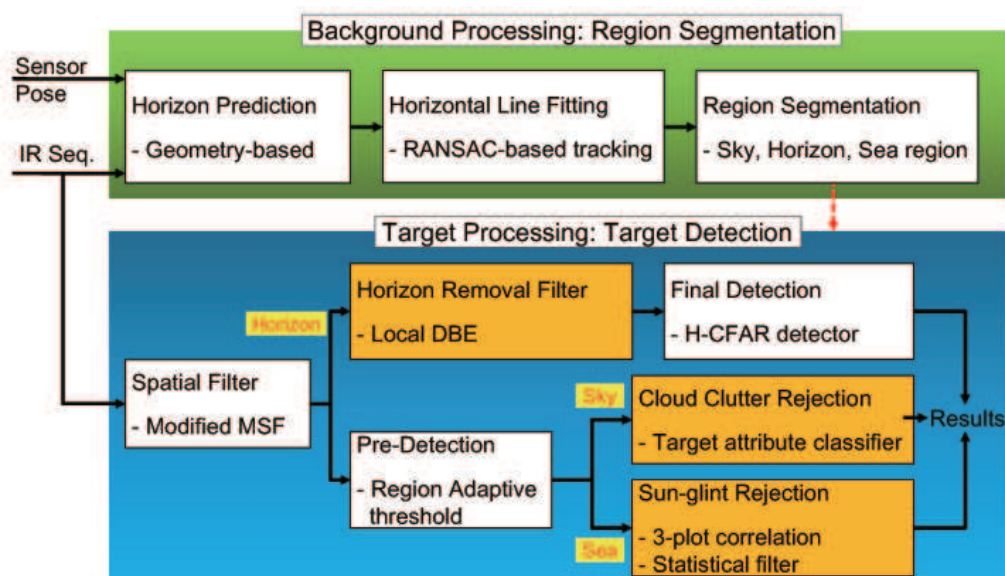
**Figure 3.** Simplified IRST projective geometry by assuming a flat surface. The triangle represents an IRST sensor, and the circular target is projected on the 1D infrared detector. The relationship between the target pixel position and target distance can be found using Equations (1)–(3).



**Figure 4.** Analysis results for the target distance ( $D$ ) and projected image position ( $i$ ). **(Left)** The left graph represents the relationship between the target distance and target pixel location using Equation (3); **(Right)** while the right image is the corresponding example of the IR scene. If the sensor height is 20 m, the target height is 0 m and the minimal detection range is 6000 m, then the ship target is located 15 pixels below the horizontal line.



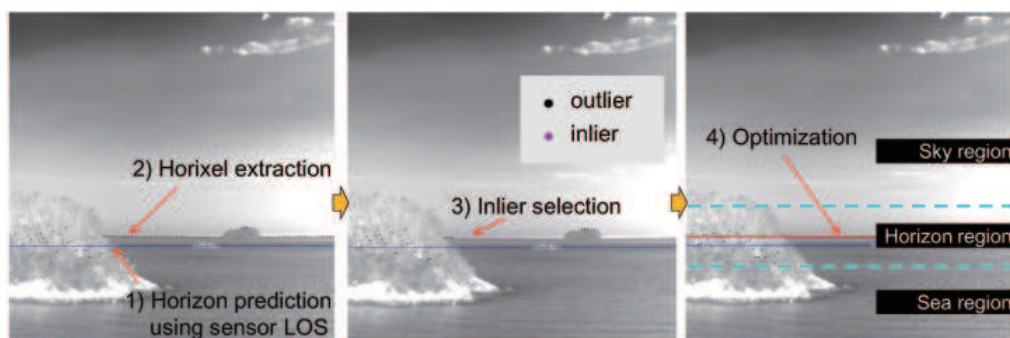
**Figure 5.** Overall small target detection flow based on region segmentation and region-specific clutter rejection.



#### 4.1. Geometry and Image-Based Region Segmentation

Horizontal information is very important, because it can provide a region segmentation cue. Therefore, region segmentation can be conducted in the following four steps: (1) horizon prediction using sensor LOS, (2) horizon pixel (horixel) extraction, (3) inlier selection and (4) horizon optimization and region segmentation, as shown in Figure 6. The horizontal location can be predicted using sensor pose information. The next step is the optimal horizon tracking in a video sequence. Given an input frame, the horixels are extracted using a column directional gradient and max selection. The inlier horixels are identified using the robust line fitting method of RANSAC [51]. The important role of RANSAC is to find the inlier indices of the true horixels. Based on the inlier index, the total least squares optimization can detect the final horizon stably. Because the inlier horixels are identified through the process, horizon tracking is conducted using horixel extraction and optimization. The inlier detection block is activated in the beginning and statistically to adapt to environmental changes.

**Figure 6.** Region segmentation flow by horizontal line prediction and optimization.



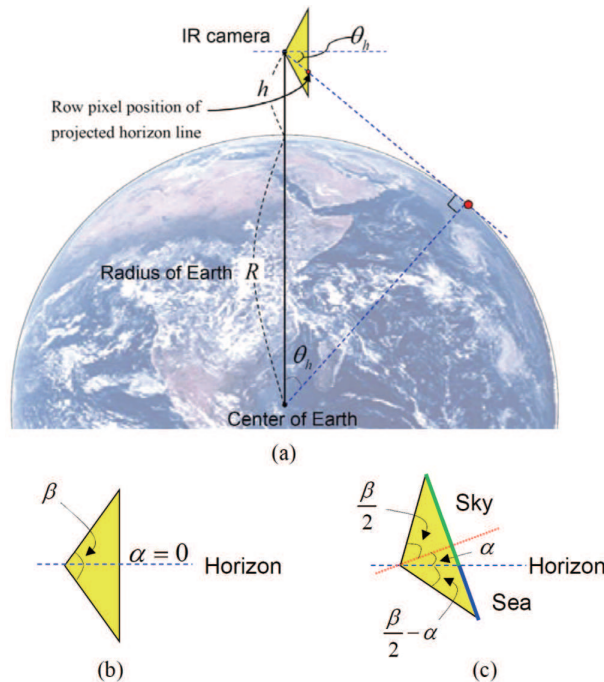
Sensor pose-based horizon prediction: If it is assumed that an IR camera has a height ( $h$ ), elevation angle ( $\alpha$ , assuming  $0^\circ$  for easy analysis) and Earth radius ( $R$ ), then the geometric relations can be depicted as shown in Figure 7a. The projected horizontal line in any image can be found by calculating the angle ( $\theta_H$ ), as shown in Equation (4). A real IRST sensor can change the elevation angle, which alters the location of the horizontal line in the image domain. If the elevation angle of a camera is given as  $\alpha$  and the field of view (FOV) of the sensor is given as  $\beta$ , then the angle of the sky region ( $\theta_{sky}$ ) is determined by Equation (5). If the elevation angle ( $\alpha$ ) is smaller than  $\theta_H - \beta/2$ , the sensor can only observe the sea region. Therefore, the angle of the sky region ( $\theta_{sky}$ ) is zero. Similarly, other cases can be analyzed. The angle of the sea region ( $\theta_{sea}$ ) is determined as,  $\theta_{sea} = \beta - \theta_{sky}$ . As the sky-sea region segmentation ratio is determined by  $\tan\theta_{sea}/\tan\theta_{sky}$ , the final horizontal line ( $H_{prior}$ ) is calculated using Equation (6). If it is assumed that the image height is 1280 pixels, the vertical field of view is  $20^\circ$ , the sensor height is 20 m and the elevation angle is  $5^\circ$ , then the prediction horizontal line ( $H_{prior}$ ) is located as shown in Figure 6 (the blue dotted line in the first image).

$$\theta_H = -\cos^{-1}\left(\frac{R}{R+h}\right) \quad (4)$$

$$\theta_{sky} = \begin{cases} 0 & \text{if } \alpha < \theta_H - \beta/2 \\ \beta & \text{if } \alpha > \theta_H + \beta/2 \\ \alpha - \theta_H + \beta/2 & \text{else} \end{cases} \quad (5)$$

$$H_{prior} = ImageHeight * \frac{\tan\theta_{sky}}{\tan\theta_{sky} + \tan\theta_{sea}} \quad (6)$$

**Figure 7.** Geometry of the sea-based IRST system. (a) Relationship between the sensor height and horizontal line; (b) camera geometry with the field of view and elevation angle ( $\alpha = 0$ ); (c) approximated position of the horizontal line when the elevation angle is  $\alpha$ .



Horixel extraction: Given a predicted horizon, as shown in Figure 6 (dotted blue line), a search boundary is set. The sampling interval is then defined to reduce the computational complexity. For each sample position, the column direction gradient filter is conducted using the derivative of the Gaussian kernel. The horixels close to a predicted horizon are then extracted by max selection. Figure 6 (dotted black line in the first image) shows the extracted horixels.

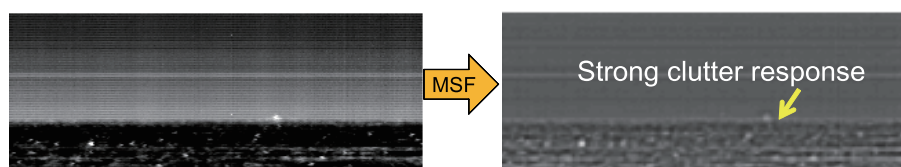
Inlier detection using RANSAC: In a sea environment, the horizon is occluded frequently by islands, coasts and clouds. Therefore, a robust horizon estimation method, such as RANSAC, is needed. Basically, the RANSAC algorithm chooses two horixels and predicts the horizon line. The algorithm then checks the line fitting and inliers. After a number of iterations, a horizon line parameter with the largest inliers is selected. Figure 6 (the second image) shows the inlier detection results using a RANSAC method. Note that the inliers and outliers are classified almost correctly. The inlier indices are used to optimize line fitting and horizon tracking.

SVD-based optimization and tracking: The last step is to refine horizon parameters using a total least squares fit of a given set of inlier horixels. The fitting process is as follows. First, the inlier horixels are normalized, and a singular value decomposition (SVD) is conducted [52]. The horizon direction is selected by an eigenvector with the smallest eigenvalue. Figure 6 (the last image) shows the horizon optimization results for an image occluded by near island and remote island. The horizontal area is enlarged to show the results. Horizon tracking is done by a horixel extraction and SVD-based optimization with the inlier indices. RANSAC-based initialization is activated statistically.

#### 4.2. Horizon Region: Removal of Horizontal Line Clutter

The mean subtraction filter (MSF)-based small target detection method is based on the 2D mean filter [8]. The 2D mean filter is used to estimate the local background with a window size of  $5 \times 5$  or  $7 \times 7$ . The MSF-based approach has been deployed in several countries, because of its simplicity and high detection capability of small targets [8,53,54]. A modified MSF (M-MSF) is used to enhance the signal-to-noise ratio using a pre-smoothing input image. On the other hand, the 2D local mean subtraction filter produces a strong response around the horizontal line, which prevents target detection or produces false detection, as shown in Figure 8. If a global threshold or constant false alarm rate (CFAR) detection are applied, the true target pixels are buried in the horizontal line pixel, which leads to the failure of horizontal target detection.

**Figure 8.** A 2D local mean subtraction filter (MSF) produces a strong response around the horizontal line where the heterogeneous regions exist.

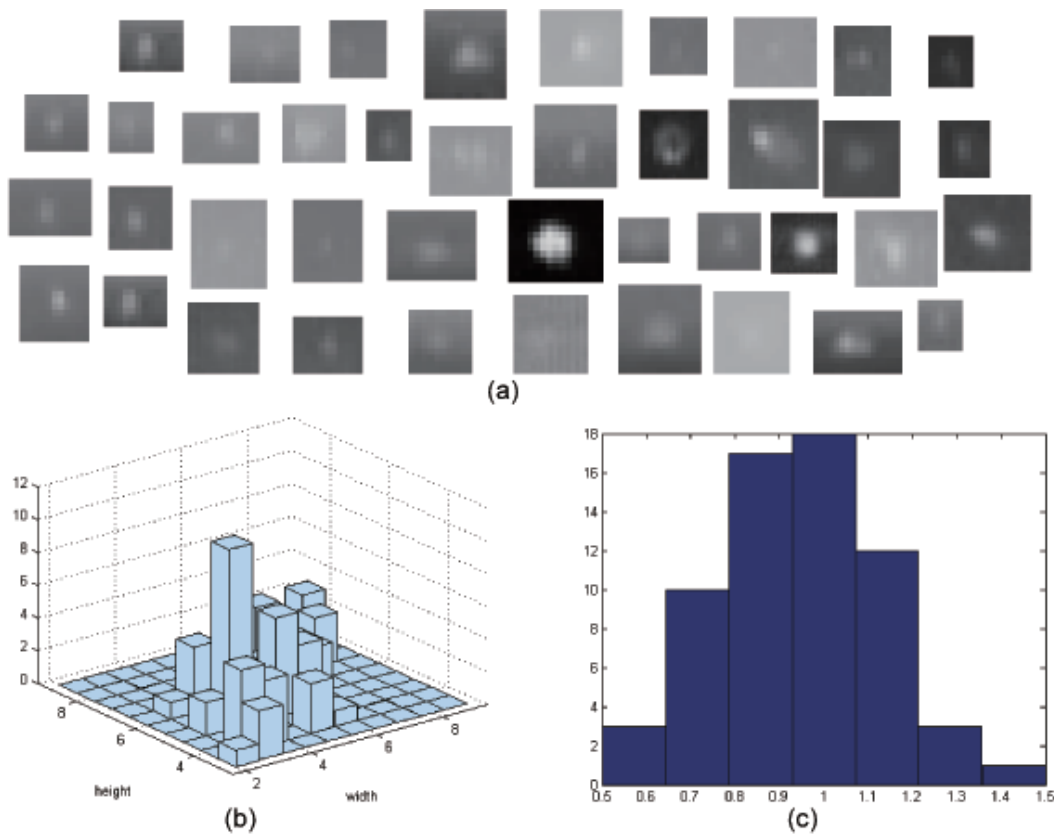


According to real target observations, the targets have Gaussian shapes, as shown in Figure 9. Figure 9 presents partial target examples, the distribution of the target size (width, height) and the



aspect ratio of observed targets, respectively. According to the statistics, the targets have blob-like structures (mean size: (width = 5.1 pixels, height = 5.4 pixel) with a standard deviation (width = 1.7, height = 1.4) and aspect ratio of  $\sim 1$ ). Note that the sizes include very low intensity pixels belonging to the target region. Therefore, a Gaussian-like filter is introduced. This idea is similar to the matched filter theory. If the filter coefficients are the same as the target shape, the the maximum signal-to-noise ratio is achieved. In this paper, the 2D Gaussian filter coefficients was set to  $G_{3 \times 3}(x, y) = [0.1 \ 0.11 \ 0.1; 0.11 \ 0.16 \ 0.11; 0.1 \ 0.11 \ 0.1]$ , which is generated by a 2D Gaussian function with a kernel size of three and a standard deviation of 1.4. The filter coefficients should be changed according the specific target applications.

**Figure 9.** Observations of real infrared targets. (a) Examples of small infrared targets; (b) histograms of target size in terms of the width and height; (c) histograms of the aspect ratio (height/width).



Therefore, the proposed M-MSF is conducted as follows (see Figure 10). An input image ( $I(x, y)$ ) is pre-filtered using the proposed filter coefficients ( $G_{3 \times 3}(x, y)$ ) to enhance the signal-to-clutter ratio (SCR), as shown in Equation (7) using the matched filter (MF). The SCR is defined as (max target signal—background intensity)/(standard deviation of background). Simultaneously, the background image ( $I_{BG}(x, y)$ ) is estimated by a  $7 \times 7$  moving average kernel ( $MA_{7 \times 7}(x, y)$ ), as expressed in Equation (8). The pre-filtered image is subtracted by the background image, which produces an image ( $I_{M-MSF}(x, y)$ ), as shown in Figure 9. The number of false detections is reduced with the same thresholds compared to that of the previous method. Therefore, the proposed M-MSF can improve the previous 2D local MSF in terms of false detections and the SCR of the true target.

$$I_{MF}(x, y) = I(x, y) * G_{3 \times 3}(x, y) \quad (7)$$

$$I_{BG}(x, y) = I(x, y) * MA_{7 \times 7}(x, y) \quad (8)$$

$$I_{M-MSF}(x, y) = I_{MF}(x, y) - I_{BG}(x, y) \quad (9)$$

The horizontal region should be processed further to remove the structural clutter, such as the horizontal line. After applying M-MSF, a SCR-improved image can be achieved. This suggests that the salt-and-pepper noise is reduced and the target signal is enhanced. The local directional background estimation (L-DBE) is applied directly to the horizontal region of the M-MSF result. In the scan-based sensor of IRST, the row pixels show similar responses, particularly around the horizontal region. Estimating the background along the scan direction for each row is reasonable. For each row, the number of target pixels is much smaller than that of the background pixels. The row directional background can be estimated based on this observation. The target pixel values are considered as outliers, whereas the background pixel values are regarded as inliers. The proposed L-DBE ( $I_{L-DBE}(x, y)$ ) is defined as Equation (10), where the tab size is  $2n + 1$ . A 1D local median filter is used to handle the image tilt error. Because a normal target size is approximately five pixels, the filter size ( $2n + 1$ ) should be five to 10 times larger than the target size to achieve a stable background estimation. In the test environment,  $n = 35$  to solve both the stable background estimation and image tilt problems.

**Figure 10.** Proposed small target detection and horizontal line clutter removal in the horizon region.

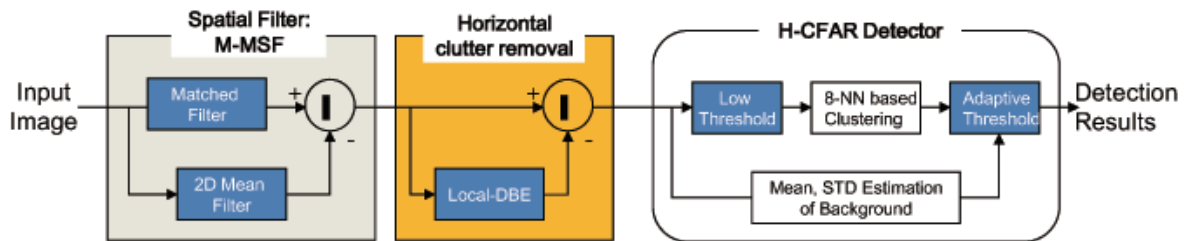
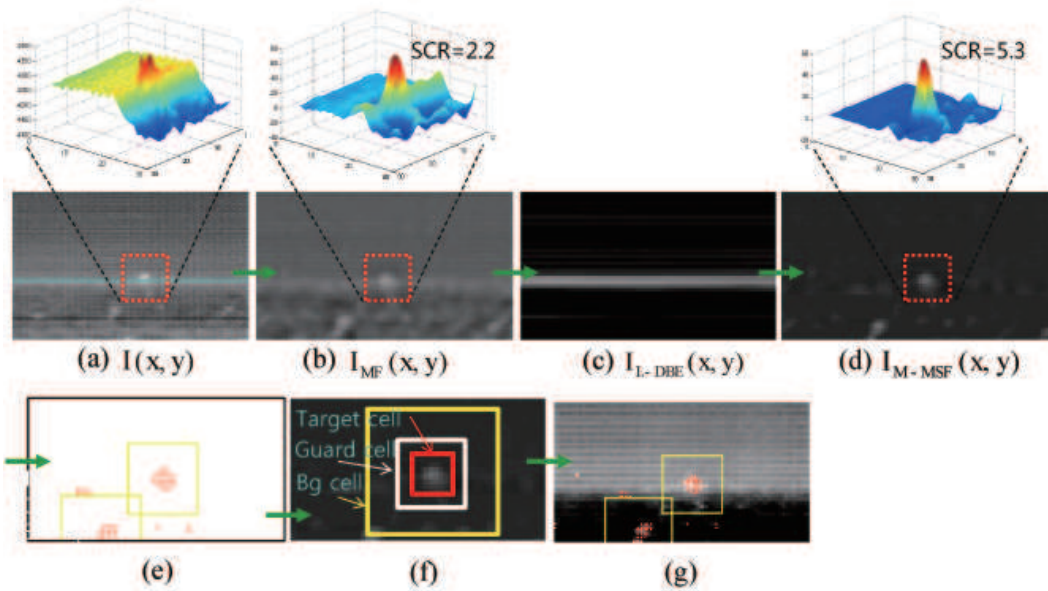


Figure 11 shows the overall procedures of the spatial filtering process for the horizontal region introduced in this section. The input of the L-DBRF is the output ( $I_{M-MSF}(x, y)$ ) of the previous filter stage from which the directional background ( $I_{L-DBE}(x, y)$ ) is estimated. The output ( $I_{L-DBRF}(x, y)$ ) of the consecutive filter can be calculated using Equation (11). Note the improvement of the SCR during the application of M-MSF and L-DBRF. Because the horizontal background clutter is estimated and removed in the L-DBRF stage, the clutter noise is reduced, leading to an enhancement of the SCR calculation.

$$I_{L-DBE}(x, y) = \text{median}\{I_{M-MSF}(x - n, y), I_{M-MSF}(x - n + 1, y), \dots, I_{M-MSF}(x, y), \dots, I_{M-MSF}(x + n - 1, y), I_{M-MSF}(x + n, y)\} \quad (10)$$

$$I_{L-DBRF}(x, y) = I_{M-MSF}(x, y) - I_{L-DBE}(x, y) \quad (11)$$

**Figure 11.** Visualization of horizontal clutter removal and detection flow: (a) input image; (b) matched filter; (c) horizontal line clutter estimation; (d) modified (M)-MSF results; (e) pre-thresholding; (f) signal-to-clutter ratio (SCR) computation region; and (g) final detection results using the SCR threshold.



The last step of small target detection in the horizon region is how to decide which pixels correspond to the target pixels. This paper proposes a new region hysteresis-threshold-based constant false alarm (H-CFAR) detector, as depicted in Figure 11e–g. A global threshold can be used to detect a possible target. On the other hand, it cannot work properly where different dense clutter exists. The global threshold-based detection scheme can be modified by incorporating the region segmentation information (sky, horizon, sea) to adapt to the properties of different backgrounds. In addition, a local background adaptive threshold, called the CFAR, can handle the clutter problem, because the threshold values are adaptive to the density of background clutter to produce constant false alarms. Directly applying the CFAR to each pixel is time consuming, because it needs to calculate the mean and standard deviation of the background pixels. The key idea is to use two region-adaptive thresholds in a hysteresis threshold framework (H-CFAR). As shown in Figure 11e, the pre-threshold is selected to be as low as possible. At the same time, the regional properties should be considered properly to find the candidate target region. The eight-nearest neighbor (8-NN)-based clustering method is used to group the detected pixels. The sizes of the possible targets can be estimated by 8-NN clustering. The probing region is divided into the target cell, guard cell and background cell, as depicted in Figure 11f. A target cell size is the same as the results of Threshold 1 with clustering. The background cell size is determined to be three- to four-times the size of the target cell. The guard cell is just a blank region that is not used in both regions and set as a two- or three-pixel gap. The second threshold ( $k_{region}$ ) in the CFAR can detect the final targets.  $\mu_{BG}$  and  $\sigma_{BG}$  represent the average and standard deviation of the background region, respectively.  $k_{region}$  denotes the region-dependent second threshold used to control the detection rate and false alarm rate.

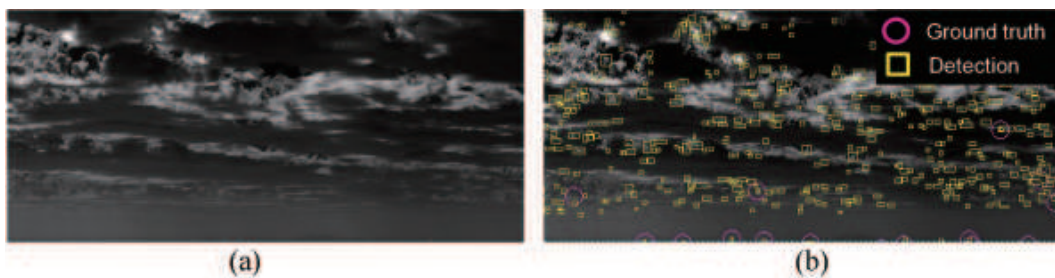
Normally, the threshold values have the following order:  $k_{horizon} < k_{sky} < k_{sea}$ . Figure 11g presents the final detection results (called plots in IRST) by applying Equation (12) to Figure 11d.

$$\begin{aligned} &\text{A probing region is a target if} \\ &SCR(x, y) = \frac{|T_{max} - \mu_{BG}|}{\sigma_{BG}} > k_{region} \end{aligned} \quad (12)$$

#### 4.3. Sky Region: Removal of Cloud Clutter

The detection results shown in Figure 12b can be obtained by applying the H-CFAR detector after spatial filtering to an IRST image, where many false detections caused by the strong cloud clutter exist for a given test image, as shown in Figure 12a. Machine learning approaches are applied to this problem. A classifier divides the correct targets and clutter points in the feature space. The simplest method is the nearest neighbor classifier (NNC) algorithm, which uses only the feature similarity [32]. In addition to NNC, there are the model-based Bayesian classifier [33], learning-based neural network and support vector machine (SVM) [34] methods. Classification information can be useful for removing various clutter points. On the other hand, it is difficult to apply these classification methods, because the targets are very small, resulting in little information being available. This paper proposes eight small target feature types and analyzes them in terms of discrimination. In this study, machine learning-based clutter rejection schemes were developed based on this feature analysis.

**Figure 12.** Problems of false alarms caused by cloud clutter: (a) original infrared image; (b) M-MSF + hysteresis threshold-based constant false alarm rate (H-CFAR) detection.



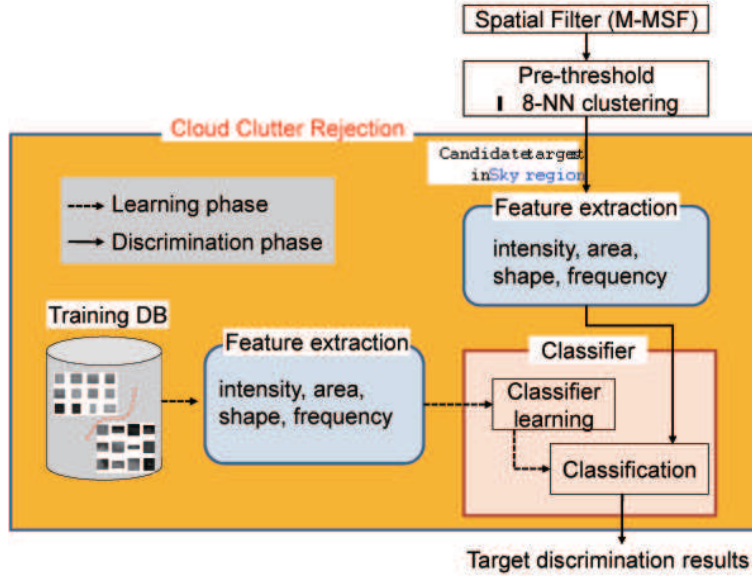
As shown in Figure 13, the cloud clutter rejection system consists of a learning phase and a discrimination phase. In the learning phase, a training database (DB) is prepared automatically using the target detection algorithm and ground truth information. The classifiers are learned using the extracted features. In the discrimination phase, the features are extracted by probing the target regions, which are obtained by the spatial filter (M-MSF) and 8-NN clustering after a pre-threshold; the final target discrimination is performed by the learned classifier.

Small infrared targets are normally small bright blobs of fewer than 100 pixels; extracting informative features from point-like target images is quite difficult. In this study, the standard deviation, ranked-fill-ratio, second-order moment, area, size ratio, rotational size variation, frequency energy and average distance methods were considered. In advance, a filtered database was considered to inspect the features.

The first feature (standard deviation) is a simple standard deviation of the image intensity for a considered region, as defined by Equation (13).  $I(i)$  denotes the intensity at the  $i$ -th pixels;  $N$  denotes the total number of pixels, and  $\mu$  is the average intensity.

$$\sigma = \sqrt{\frac{\sum_{i=1}^N (I(i) - \mu)^2}{N}} \quad (13)$$

**Figure 13.** Overall flow of the target discrimination.



The second feature (ranked-fill-ratio) considers the ratio between the  $K$  brightest pixels and the total intensity, as defined in Equation (14). The targets normally have higher values than the clutter, because targets are observed as a hot spot on a cold background.

$$\eta = \frac{\sum_{j \in K} I(j)}{\sum_i I(i)} \quad (14)$$

The third feature (second order moment) considers the second image moment as defined in Equation (15).

$$m_{22} = \frac{\sum_x \sum_y (x - \mu_x)^2 (y - \mu_y)^2 I(x, y)}{\sum_i \sum_j I(i, j)} \quad (15)$$

The following five features are basically extracted from the target region: In the fourth feature (area), a black and white target region is obtained by applying Otsu's method, which chooses the threshold to minimize the intraclass variance of the black and white pixels [55]. Given a gray image  $I(i)$ , the segmented target region is denoted as  $R(i)$ . This feature can be calculated using the following equation:

$$a = \sum_i R(i) \quad (16)$$

The fifth feature (Size Ratio) considers the target size ratio. If the target width is denoted as  $l_W$  and the target height is expressed as  $l_H$ , then the ratio can be defined as:

$$S_{ratio} = \frac{l_H}{l_W} \quad (17)$$

The sixth feature (rotational size variation) is based on the rotational size profile ( $L(i)$ ). A target size profile is generated by rotating the region. Therefore, the rotational size profile reflects the target shape. The profile is uniform if a small target has a circular blob, whereas it is similar to a cosine curve if it has a rectangular shape. The rotational size profile can be quantified using the standard deviation of the curve, as defined in Equation (18).

$$\sigma_L = \sqrt{\frac{\sum_{i=1}^N (L(i) - \mu_L)^2}{N}} \quad (18)$$

The seventh feature regards the frequency energy and is obtained by applying a fast Fourier transform ( $FFT$ ) to the rotational size profile ( $L(i)$ ):

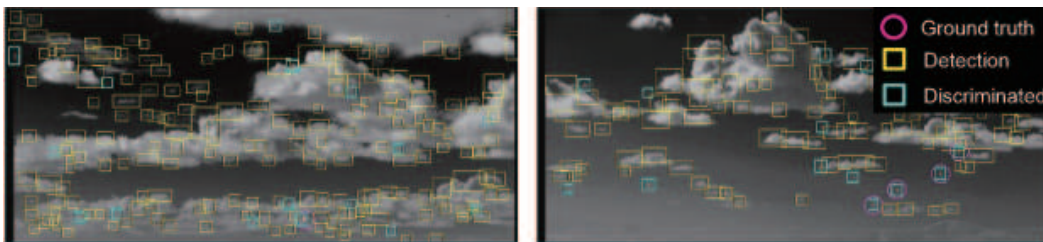
$$M(k) = FFT(L(i) - \mu_L),$$

$$f_{energy} = \sum_{k=1}^M \frac{|M(k)|^2}{M} \quad (19)$$

The last feature is the mean distance. If a region consists of  $N$  pixels and the region center is  $(\mu_x, \mu_y)$ , the average Euclidean distance can be calculated using the following equation:

$$d = \frac{\sum_{i=1}^N \sqrt{(x(i) - \mu_x)^2 + (y(i) - \mu_y)^2}}{N} \quad (20)$$

**Figure 14.** Cloud clutter rejection examples using the proposed feature and AdaBoost classifier.



This section thus far discussed the feature extraction methods to discriminate infrared small targets and cloud clutters. The remainder of the process is the selection of the optimal classifier. In this study, AdaBoost was chosen, because it can select the features suitable for discriminating true targets. The SVM method considers multi-dimensional feature vectors and finds the support vectors using a kernel recipe. AdaBoost, on the other hand, uses simple weak classifiers ( $h_i$ ), as well as the weighted sum of weak classifiers, which leads to a strong classifier, as expressed in Equation (21). In this study, the weak classifiers are just simple threshold-based binary decisions for individual

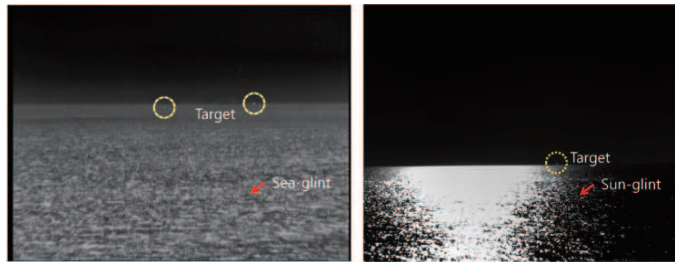
feature space. Figure 14 presents examples of cloud clutter rejection using the proposed method. Note that the proposed scheme can remove false detections by cloud clutter.

$$H_{strong}(\mathbf{x}) = \text{sign} \left( \sum_{i=1}^N \alpha_i h_i(\mathbf{x}) \right) \quad (21)$$

#### 4.4. Sea Region: Removal of Sea-Glint

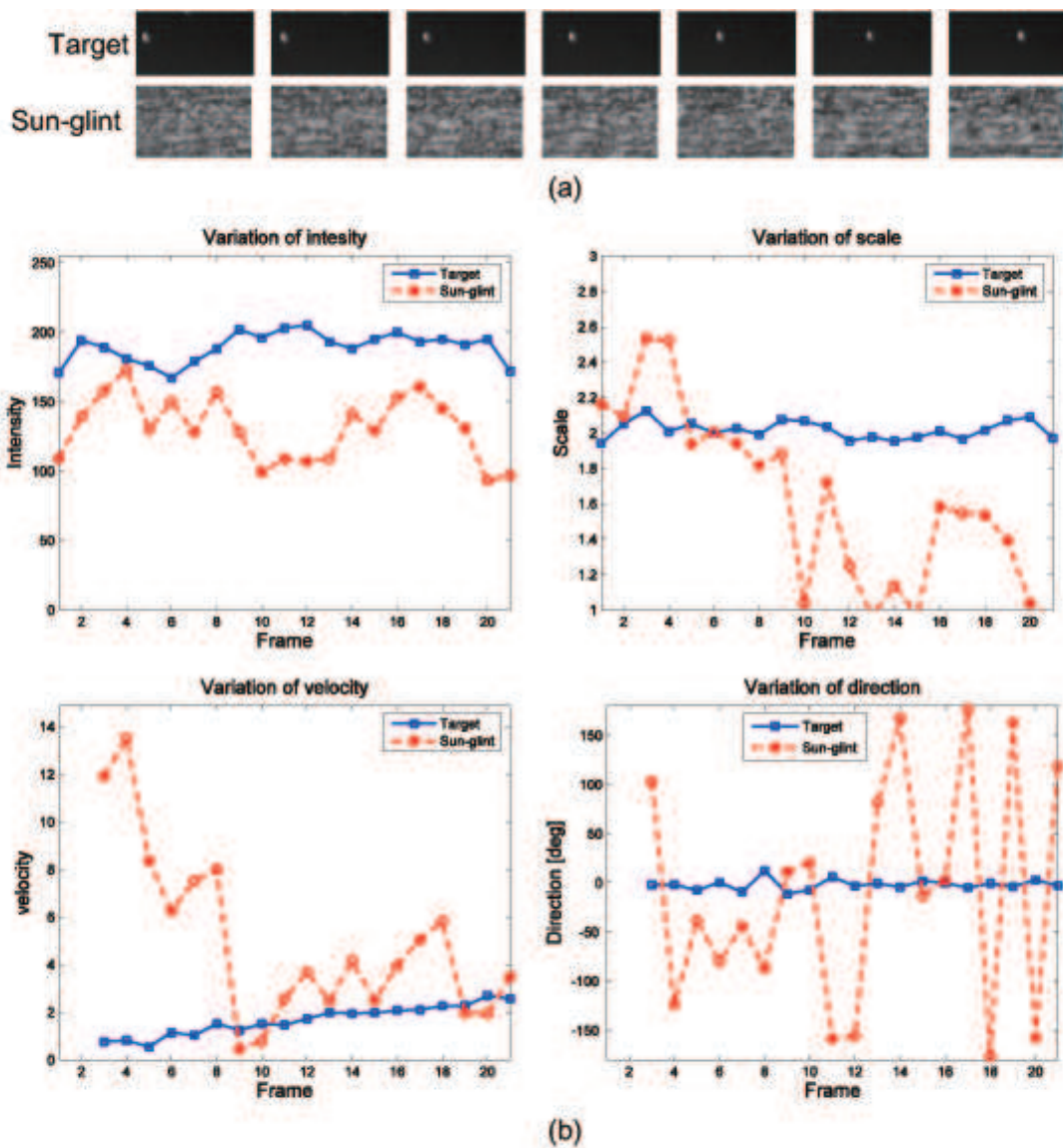
Sea-glint makes the detection of small targets in the sea region a challenging problem, as shown in Figure 15. The dotted circle indicates the true target, and the arrow indicates the sun-glint. The irradiated target energy is quite small, due to scattering and absorption through the atmosphere. This leads to a dim target, whose signal-to-noise ratio (SNR) is quite low. The dim targets are composed of 2–10 pixels. The target intensity level is similar to that of the neighboring pixels. Furthermore, sun-glint has a similar shape (circular symmetry), like small targets, and a high intensity value, which hinders true target detection.

**Figure 15.** Example of sun-glint in the infrared search and track system.



Why is the detection of a small target very difficult? If each frame is observed, as shown in Figure 16a, the targets and sun-glint have small bright spots. Therefore, spatial shape information cannot discriminate the true targets and sun-glint. On the other hand, if targets and sun-glint are observed in the temporal domain, observation results can be obtained in terms of intensity, scale, velocity and moving direction, as shown in Figure 16b. The key property is consistency. The targets show a consistent intensity, scale, velocity and direction compared to sun-glint.

**Figure 16.** Observation of a target and sun-glint: (a) sequence of a target and sun-glint; (b) observation results in terms of the intensity, scale, velocity and moving direction.

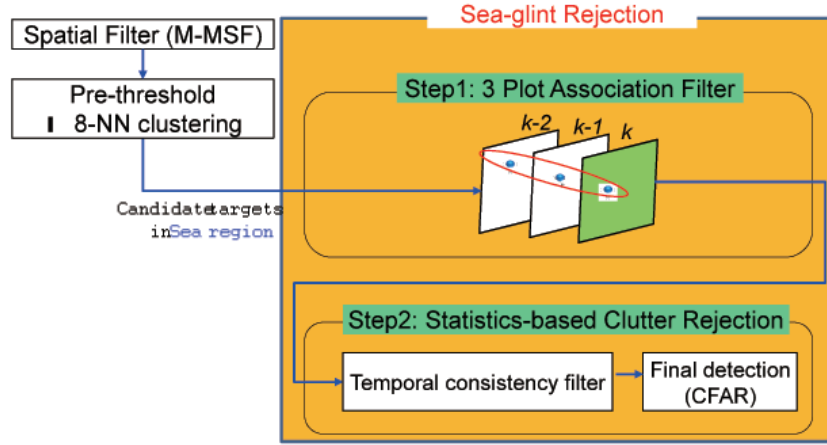


According to the survey, there have been few studies on small target detection in a dense sun-glint clutter environment. A single spatial filter cannot remove sun-glint clutter as the signatures of a true target, and sea-glint has a quite similar shape with circular symmetry. A conventional motion cue cannot be utilized, as the target may be stationary and the frame rate is very low. Therefore, this paper proposed a hybrid method by making a compromise for the spatial filter approach and temporal approach, known as the separate spatio-temporal filtering method based on an attribute-based plot association. The plot indicates only the candidate target in IRST. The underlying assumption is that a true target behaves like an outlier in both the spatial and temporal domains. The behavior of sun-glint is random, but that of the targets is consistent. Such a concept is used in the design of spatial and temporal filters. Figure 17 represents the proposed target system based on these concepts. The top component level consists of a plot association-based temporal filtering part and a statistics-based clutter rejection part, given the candidate targets extracted by pre-detection



using M-MSF and pre-thresholding. In the temporal filtering part, this paper proposes a three-plot association filter based on the target attributes for data association. After a three-plot association, the sea-glint clutter is reduced further using a temporal consistency filter and constant false alarm (CFAR) detection method.

**Figure 17.** Proposed small target detection system. The system consists of a geometric sea region extraction part, spatial filtering part, three-plot correlation-based temporal filter part and statistics-based clutter rejection part.



The next step is to produce a group of plots, called the three-plot correlation or association to remove sun-glint. In general, this can be considered a target tracking problem, like Bayesian filtering, shown in Equation (22).  $x_k$  denotes the target position to be estimated;  $z_k$  denotes the observed target position, and  $Z_k$  denotes the observation sequence data up to the  $k$ -th frame.  $p(x_k|Z_{k-1})$  acts as a prior target position estimated from the previous frames. Data association should be conducted to link a target track and an observation in measurement,  $p(z_k|x_k)$ . This approach is focused on estimating target position using a large amount of frame data.

$$p(x_k|Z_k) = \frac{p(z_k|x_k)p(x_k|Z_{k-1})}{p(z_k|Z_{k-1})} \quad (22)$$

where:

$$p(x_k|Z_{k-1}) = \int_{x_{k-1}} p(x_k|x_{k-1})p(x_{k-1}|Z_{k-1}) \quad (23)$$

In the target detection problem, the focus is on how to remove sun-glint within three frames (system requirement), but leave the tracking of the targets relatively unscathed. As mentioned earlier, the basic assumption is that targets behave as outliers compared to the sun-glint. This suggests that sun-glint behaves randomly, but true targets behave consistently. Therefore, the false alarms caused by the sun-glint can be removed through the three-plot correlation using a graphical model. Figure 18a shows the basic concept of a three-plot correlation using a graphical model. The white circle denotes the hidden variable, and the gray circle denotes the detected target data. The correlation is concerned only with a prior prediction and data association given in three consecutive frames. Figure 18b shows a corresponding three-plot correlation process. The first frame is used to

generate an initial plot, whose attribute is  $F_{t-2} = [row(r), column(c), height(h), width(w), area(a), intensity(i), 0, 0]_{prior}^{k-2}$ . Given this information, this plot can be associated with the new plot in the second frame. The association is conducted by finding the maximum target similarity using the previous attribute information. The feature distance measure that is proposed in Equation (24) is used. This can measure the shape distance by summing the differences in the heights, widths, areas and intensities between the associating targets. The target motion, such as moving distance ( $d$ ) and moving direction ( $\theta$ ), can be found during the consecutive association. The previous unassociated plot ( $k - 2$ ) is removed automatically, and the currently unassociated plot ( $k - 1$ ) generates a new plot. Given this attribute ( $F = [r, c, h, w, a, i, d, \theta]_{prior}^{k-1}$ ), the second plot can be associated with the third plot using the target attribute and the target motion prediction. If the three consecutive plot attributes are collected, a statistics-based clutter rejection is conducted, which is explained in the following subsection.

$$S_{dist}(F_{t-1}, F_t) = |h_{t-1} - h_t| + |w_{t-1} - w_t| + |a_{t-1} - a_t| + |i_{t-1} - i_t| \quad (24)$$

The previous three-plot correlation method checks only the shape similarity of the associating targets. If the temporal behavior, such as the intensity statistics and motion statistics, is considered, the sun-glint can be removed further for the three correlated plots (*correlation ID* = 3), as shown in Figure 16. Given the plot attributes, as shown in Figure 19, the intensity consistency filter ( $C_I$ ) and motion consistency filter ( $C_M$ ) can be applied using Equations (25) and (26), respectively.  $\sigma$  denotes the standard deviation, and  $d_{Th}$  denotes the distance threshold of the target motion. Although the number of data points is just three, these filters are powerful for rejecting sun-glint. The standard deviation of both the plot intensity and plot motion are used. On the other hand, the standard deviation of the motion direction is considered only if the motion is large enough to avoid the image noise effect (e.g.,  $d_{Th} > 2$  pixels).

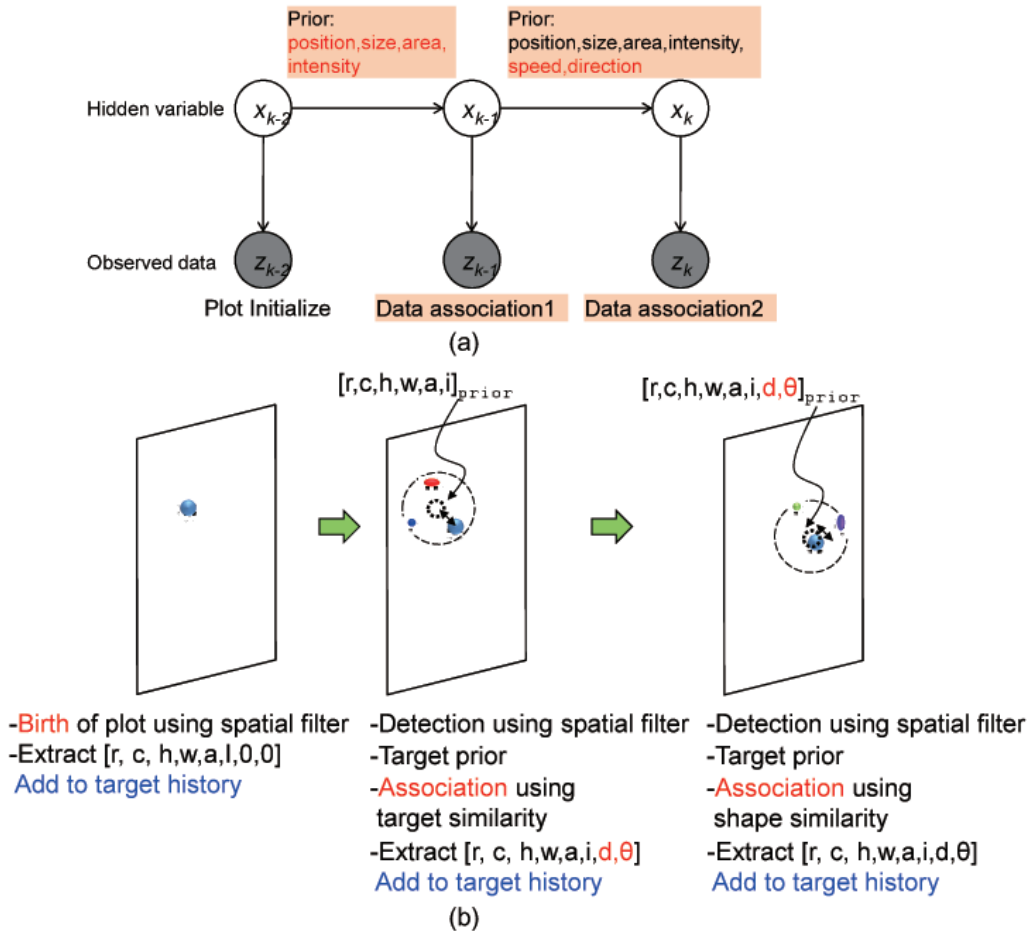
$$C_I = \sigma([i_{k-2}, i_{k-1}, i_k]) \quad (25)$$

$$if \ d_k > d_{Th} \quad (26)$$

$$C_M = \sigma([\theta_{k-2}, \theta_{k-1}, \theta_k])$$

To explain the proposed detection system depicted in Figure 17, this paper presents the overall processing flows with the related results for a standard test image, as shown in Figure 20. The test IR image (Figure 20a) has possible targets on the sea. Figure 20b represents the detection results using a three-plot correlation filter. The ID indicates the number of correlations. For this process, M-MSF and pre-thresholding are used for spatial candidate target detection. Figure 20c represents the results of a statistics-based temporal filtering. Figure 20d shows the targets finally detected using the H-CFAR method. Table 2 summarizes the clutter reduction rate for this test sequence. The proposed three-plot correlation filter can reduce 50% of clutters. Through the temporal filter and CFAR detection, we can achieve up to 97.7% of clutter rejection, while detecting the true targets.

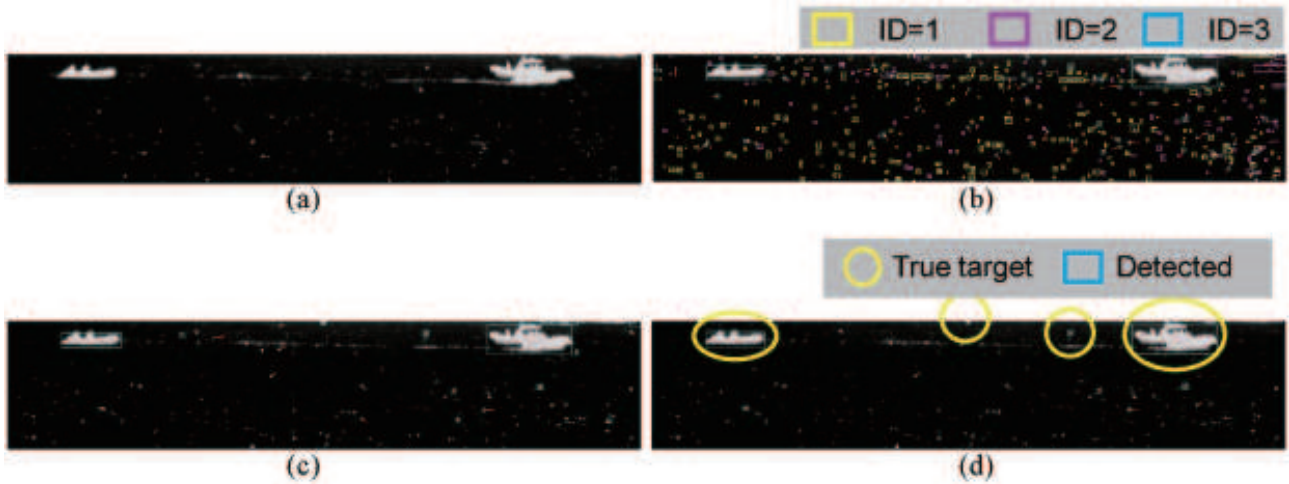
**Figure 18.** Concept of temporal filter using data association: (a) Graphical model-based representation of a three-plot correlation filter; (b) implementation procedures. The first frame is used to generate initial plots without prior knowledge. In the second frame, the prior target attribute is used for data association. In the third frame, the prior motion is also used during data association.



**Figure 19.** Attributes of the three-plot correlation and temporal behavior data of intensity and motion used for a statistics-based clutter rejection.

		Correlation ID: 3			# of plot correlation
		$k-2$	$k-1$	$k$	
		r=333	r=335	r=336	Plot attribute
		c=360	c=360	c=359	
		h=5	h=4	h=4	
		w=4	w=4	w=5	
		a=18	a=16	a=18	
		i=64	i=72	i=67	
Intensity		d=0	d=2	d=1.41	
Motion		theta=0	theta=1.57	theta=0	

**Figure 20.** Example of the target detection flow for a sea regional infrared image. (a) Test image; (b) three-plot correlation results (ID denotes the number of plot correlation); (c) temporal filter-based clutter reduction; (d) final detection using the H-CFAR method. The circles represent true targets, and squares represent detected targets.



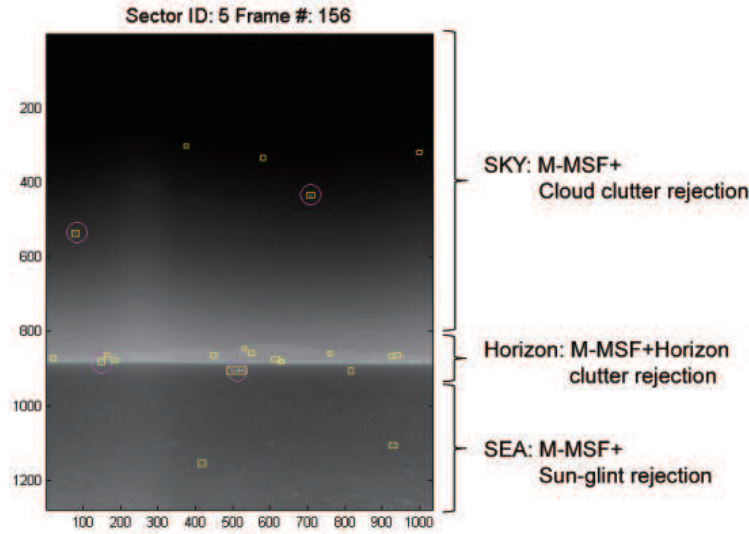
**Table 2.** Clutter reduction performance for each spatio-temporal processing module.

Processing	No. of Plots	Clutter Reduction Rate (%)
1-plot correlation	807	0.0
3-plot correlation	399	50.5
Temporal filter	289	64.2
CFAR detection	18	97.7

## 5. Experimental Results

This paper introduced details of the proposed region segmentation by horizon detection, horizontal line clutter rejection, cloud clutter rejection and sun-glint rejection, as shown in Figure 21. In this section, each proposed item was evaluated by comparing the conventional methods, and then, the integrated method was applied to test sequences.

**Figure 21.** Proposed region-adaptive, small target detection and clutter rejection scheme.



5.1. Evaluation of Horizontal Line Detection

Four kinds of test sequences were prepared, as shown in Figure 22, to validate the robustness of the proposed method. Set 1 is remote sea images occluded by a strong cloud. Set 2 is occluded by the island nearby, which occupies 1/3 of the horizon length. Set 3 has nearby islands and a remote island. The last one, Set 4, has a coast nearby, in which boats and buildings occlude the horizon.

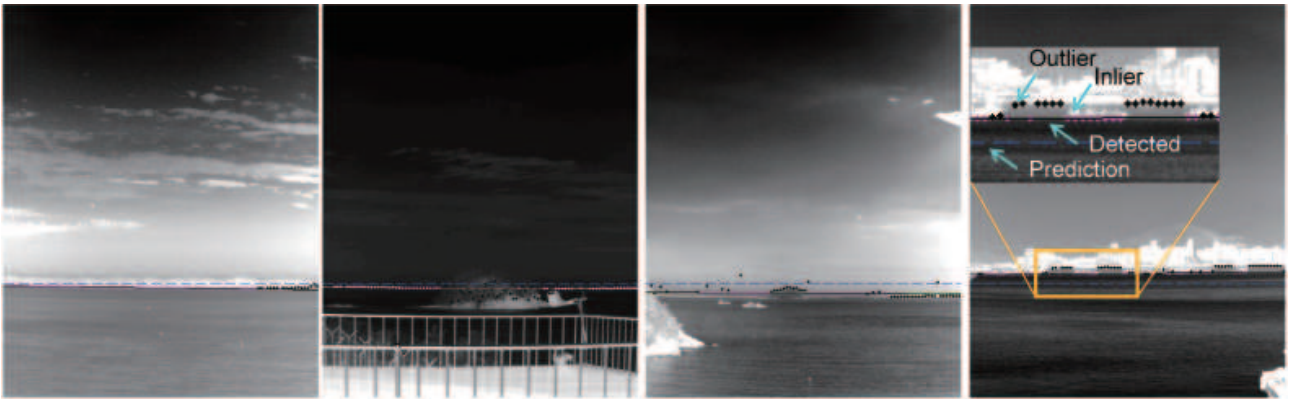
**Figure 22.** Composition of the test database for horizontal line detection.



A detected horizon is declared to be a correct detection if the line fitting error is within one pixel on average. The ground truth of the horizon location was prepared by a manual inspection. The original test sets had almost no sensor noise. Therefore, artificial sensor tilt noise and horizon location noise by the  $\pm 0.5^\circ$  and  $\pm 3.0$  pixels, respectively, were generated by the uniform distribution for that range. Table 3 lists the overall experimental results. The proposed method detected the horizons correctly for the noiseless sequence data. In the case of the noisy data, only one frame of Set 4 showed incorrect horizon detection. Figure 23 shows the sampled horizontal detection results for the noise-added sequences. The dotted blue lines denote the horizon prediction by sensor LOS. The solid black or white line denotes the optimal horizon. The magenta dots denote the inlier horixels extracted by RANSAC. Note that the horizon lines are detected robustly, regardless of the occlusion types under sensor noise.

**Table 3.** Detection rate (DR) of the horizon for the noiseless data and noisy data.

Test Set	DR w/o Noise (%)	DR with Noise
Set 1	100 (20/20)	100 (20/20)
Set 2	100 (35/35)	100 (35/35)
Set 3	100 (35/35)	100 (35/35)
Set 4	100 (30/30)	97 (29/30)

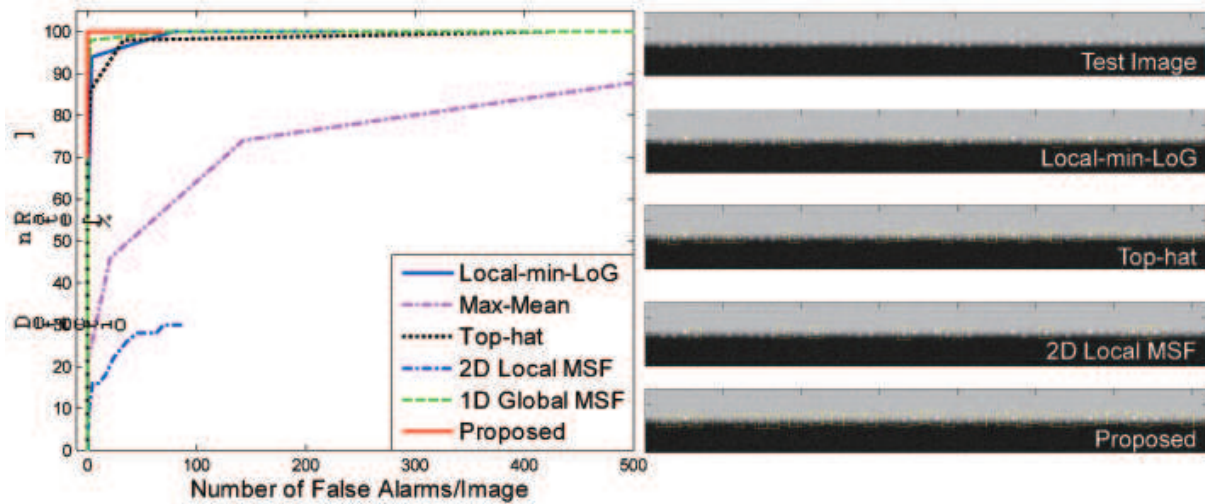
**Figure 23.** Examples of horizon detection for the test Sets 1–4.

### 5.2. Evaluation of Horizontal Clutter Rejection

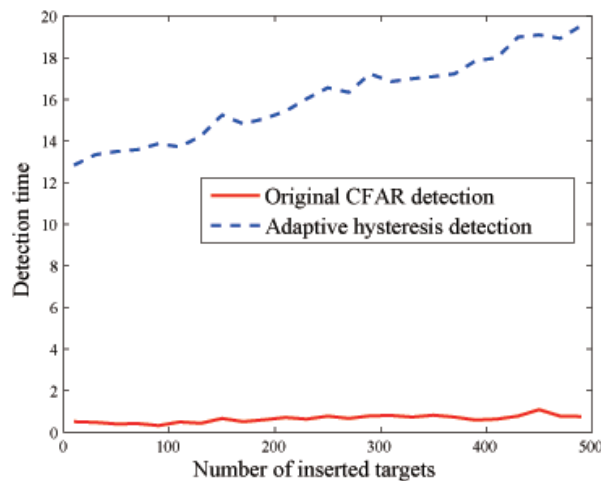
In an evaluation of horizontal clutter rejection, the detection rate and false alarms per image were compared to evaluate the detection performance according to the different spatial filter types. As initial experiments, a synthetic image was prepared by background modeling and target modeling. The background image had a sky region and a background region with an intensity difference of 100 gray values. The horizontal line was smoothed further column-wise using a Gaussian filter. Fifty targets were generated with different sizes and difference SCR values. Those targets were inserted around the horizontal line, as shown in the top of Figure 24b. The targets generated have a size range of  $(3 \times 3)$  to  $(10 \times 10)$  and an SCR range of 0.97 to 1.95. The ROC curve metric was used to evaluate the filtering method for this test image. The pre-threshold ( $Th_{pre}$ ) was set as five, and the H-CFAR threshold ( $k$ ) was changed from one to 20. Figure 24a shows the evaluation results. The results with a 2D Local MSF [8] show a very small ROC region and a relatively low detection rate. The max-mean filter [12] also produces a poor ROC area. The 1D Global MSF-based method showed much larger ROC region, but produced many false detections (more than 4000 false alarms with  $k = 1$ ) with a small threshold value. Recent methods, local-min-LoG and Top-hat filter, showed good performances [15,56]. In contrast, the proposed method (horizontal clutter rejection (L-DBRF) after M-MSF) showed an ideal ROC curve pattern. Note that the maximum number of false alarms was just 70 with  $k = 1$ . Figure 24b shows the target detection results using three types of spatial filters. The H-CFAR thresholds were tuned to make zero false alarms. The proposed method could detect all of the targets successfully.

In the next evaluation, the target decision methods were compared. The original CFAR detector probes all of the pixels above the noise level. On the other hand, the proposed decision method (H-CFAR) uses an adaptive hysteresis threshold consisting of a small threshold for candidate detection and a CFAR threshold for the final decision. A test image consists of a different number of synthetic targets from 10 to 490. Figure 25 presents the comparison results. The processing time of the original CFAR detection took approximately 16.1 s, which increased with increasing numbers of targets. In contrast, the processing time of the proposed detection method took approximately 0.65 s and increased slightly with increasing number of targets. Both decision methods showed similar detection results.

**Figure 24.** ROC curves and related detection examples. (a) ROC curves of three different spatial filters; (b) detection examples with thresholds of zero false alarms.



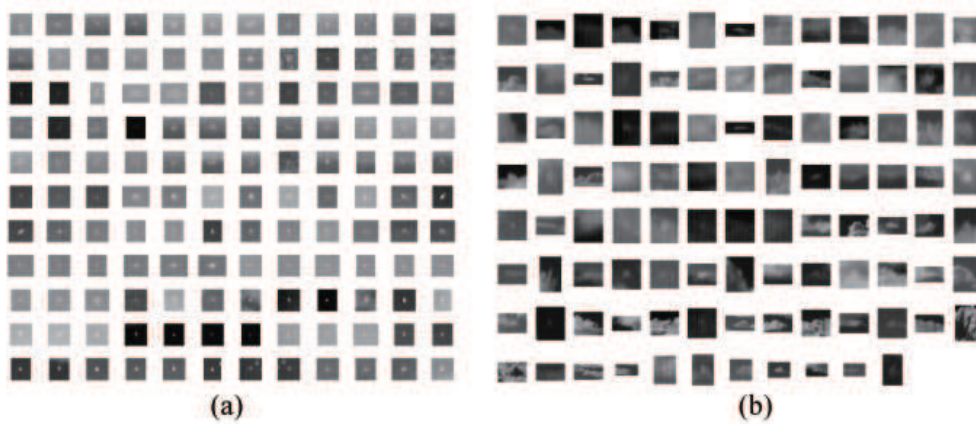
**Figure 25.** Processing time of the decision methods: CFAR vs. H-CFAR (adaptive hysteresis detection).



### 5.3. Evaluation of Cloud Clutter Rejection

A sufficiently large data set is important for ensuring successful learning for cloud clutter rejection. In this study, 136 real target images were collected using either a mid-wave infrared (MWIR) camera or a long-wave infrared (LWIR) camera. The target images were acquired by real airplanes, such as the KT-1, F-5 and F-16. The cloud clutter database was prepared using the detection algorithms introduced in the previous section. Figure 26 provides examples of the target and clutter images.

**Figure 26.** Target and clutter database for classifier learning: (a) target chips; (b) clutter chips.



The naive Bayes, SVM and AdaBoost classifiers were compared in the evaluation. The training samples were selected randomly, and the remaining samples were used for the test set. The average detection rate (DR) and false alarm rate (FAR) were evaluated over 100 iterations. Table 4 lists the results. Although the naive Bayes method produced a low FAR, it had a relatively low DR. The DR is more important in target discrimination, because true targets need to be detected. The SVM classifier produced an improved DR, but had a high FAR. The AdaBoost classifier (29 weak classifiers after learning) produced an improved DR with a lower FAR than that found for the SVM. Therefore, AdaBoost was selected as a classifier to reject cloud clutter in the sky region.

**Table 4.** Performance of the: (a) naive Bayes, (b) SVM and (c) AdaBoost classifiers in terms of the detection rate (DR) and false alarm rate (FAR).


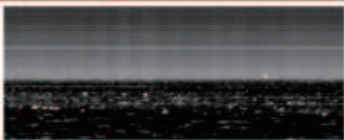


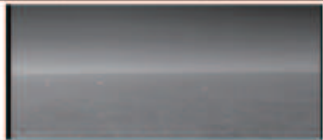
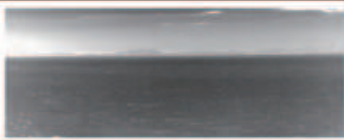

Measure	Naive Bayes [57]	SVM [58]	AdaBoost [59]
DR (%)	84.07	86.56	88.80
FAR (%)	6.03	13.58	8.70



#### 5.4. Evaluation of Sea-Glint Rejection

A set of sea-basedIRST images were prepared to test and evaluate the proposed method. Figure 27 summarizes seven kinds of test sequences that were acquired by mid-wave infrared (MWIR) cameras. Set 1 has weak sun-glints with an incoming ship scenario. Set 2 has strong sun-glint with ships passing by. Set 3 has strong sparse sun-glints with large ships near the coast. Set 4 has dense strong sun-glints with a synthetic incoming target and far away true targets. Set 5 has weak dense sun-glint with a synthetic incoming target and several real ships. Set 6 has strong sparse sun-glint with WIGships passing by in a remote coastal environment. Set 7 has strong dense sun-glint with WIG ships passing by. Each image set was used selectively, depending on the evaluation.

**Figure 27.** Composition of the test database.

ID	Sample Image	Information	ID	Sample Image	Information
1		-Bg.: weak sun-glint -Target: incoming ship	2		-Bg.: Strong sun-glint -Target: Passing by ship
3		-Bg.: Strong sparse sun-glint -Target: Passing by ships	4		-Bg.: Dense strong sun-glint -Target: synthetic + true ship
5		-Bg.: Weak dense sun-glint -Target: synthetic + real ships	6		-Bg.: Strong sparse sun-glint -Target: passing by wig ships
7		-Bg.: Strong dense sun-glint -Target: passing by wig ships			

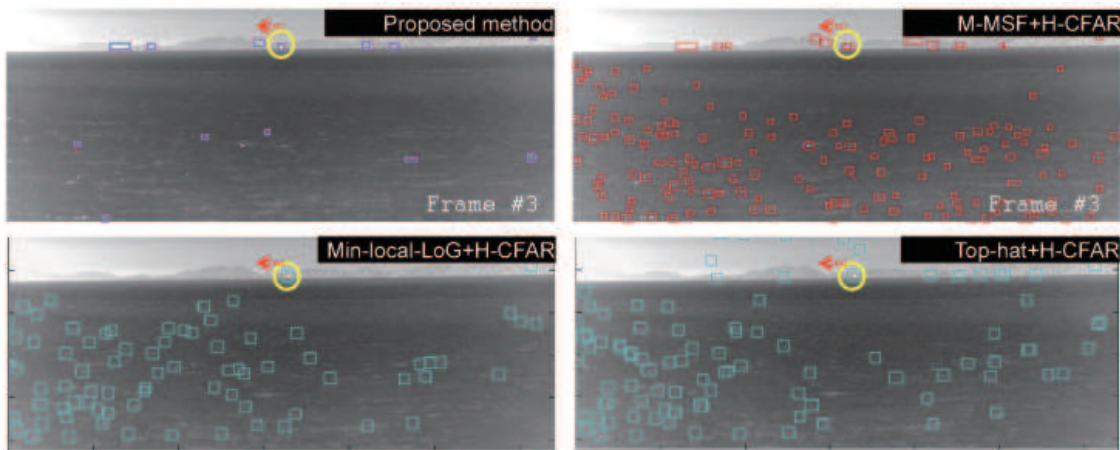
**Table 5.** Statistical performance comparisons of the proposed method and baseline method. DR denotes the detection rate, and FAR denotes the number of false alarms per image. DB, database.

Test DB	DR	FAR (number/image)			
		Proposed	M-MSF	Min-local-LoG[15]	Top-hat [56]
Set 3 (strong, sparse)	94.3 (83/88)	<b>12</b>	24	20	40
Set 4 (strong, dense)	97.6 (121/124)	<b>2</b>	8	13	16
Set 5 (weak, dense)	99.2 (119/120)	<b>1</b>	16	9	16
Set 6 (strong, sparse)	98.6 (72/73)	<b>22</b>	99	74	102
Set 7 (strong, dense)	94.7 (71/75)	<b>18</b>	75	70	95

The proposed sea-glint rejection method was compared with the baseline methods ((M-MSF, Min-local-LoG [15], Top-hat [56]) + H-CFAR detection) for five kinds of test sets (Set 4, Set 5, Set 6,

Set 7). The detection rate (DR) and number of false alarms (FAR) per image were used as the comparison measures. For a fair comparison, the detection rates were fixed for each data set by tuning threshold values. For each test set, the ground truths were prepared manually. Table 5 lists the overall performance results for the five different test sets in terms of false alarm rate (number of false detections per image). The proposed method showed the same detection rate as the baseline method, but it produced fewer (approximately one- to 16-times fewer) false alarms than the baseline methods. Figure 28 shows the detection results for test Set 6, which had a real target (WIG ship) passing by a remote coast. In the proposed method, the squares denote the final detection by removing the edge targets. Note that the baseline methods produced a large number of false alarms around the sun-glint. According to the results, the proposed method (3 plot correlation + attribute filter) could detect the true targets robustly and produce a small number of false alarms in the sea-glint region.

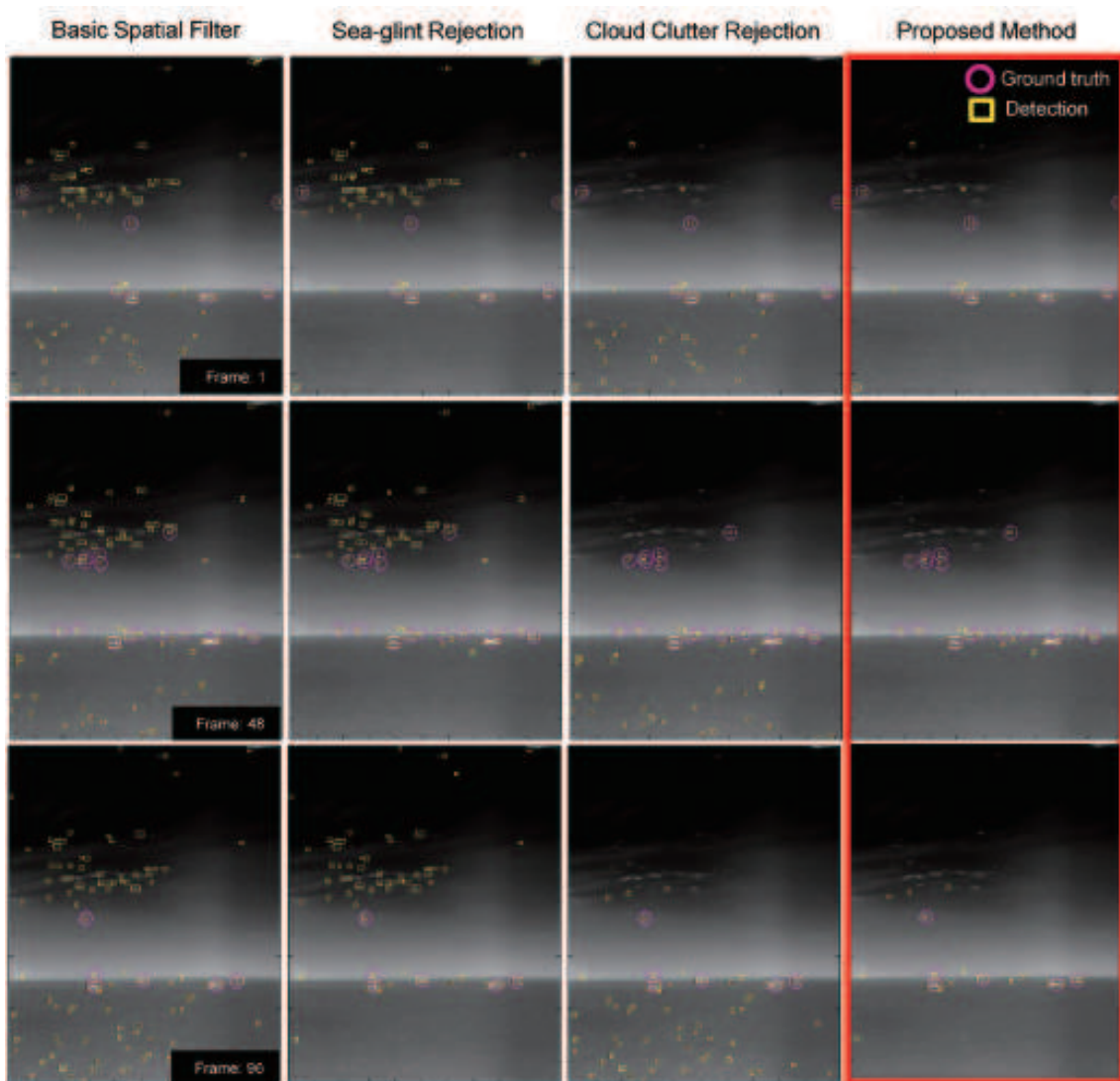
**Figure 28.** Target detection comparison between the proposed method and the baseline method for test Set 6.



### 5.5. Integrated Evaluation of the Proposed Method

As a final evaluation, the test sequence consisted of five sectors with 156 frames ( $1280 \times 1024$ ). A number of synthetic targets were generated using the method reported by Kim *et al.* [60]. The test sets consisted of cloud clutter and sea-glint. Table 6 lists the overall evaluation results depending on the clutter rejection schemes in terms of the detection rate and number of false alarms per frame. The basic spatial filter means the (M-MSF + L-DBRF) + H-CFAR detector. The basic one denotes M-MSF + pre-thresholding. The proposed method (region-wise clutter rejection) reduced the number of false detections by a factor of 2.5 to 9.4 per image, depending on the sector type by the clutter rejection schemes, with just a 0.1%–0.8% degradation in the detection rate. Figure 29 gives examples of the clutter rejection effects on the Sector 2 DB. Note that the false detections in the cloudy sky region and in the sea-glint region were removed almost completely by the proposed method, while still maintaining target detection.

**Figure 29.** System performance comparison results by applying the clutter rejection methods.



**Table 6.** Comparison of clutter rejection performance.

Test Set	Method	DR (%)	FAR (number/frame)
Sector 1	Basic spatial filter	99.8 (601/602)	272.0 (42,483/156)
	Basic + Cloud clutter reject	99.0 (596/602)	204.5 (31,909/156)
	Basic + Sun-glnt reject	99.8 (601/602)	154.7 (24,136/156)
	Proposed	<b>99.0</b> (596/602)	<b>87.2</b> (13,607/156)
Sector 2	Basic spatial filter	99.8 (1576/1478)	80.6 (12,589/156)
	Basic + Cloud clutter reject	99.2 (1467/1478)	39.3 (6146/156)
	Basic + Sun-glnt reject	99.8 (1576/1478)	49.8 (7769/156)
	Proposed	<b>99.2</b> (1467/1478)	<b>8.5</b> (1326/156)
Sector 3	Basic spatial filter	99.9 (1407/1422)	19.4 (3039/156)
	Basic + Cloud clutter reject	98.4 (1399/1422)	16.2 (2521/156)
	Basic + Sun-glnt reject	98.9 (1407/1422)	7.7 (1206/156)

**Table 6. Cont.**

Test set	Method	DR (%)	FAR (number/frame)
Sector 4	Proposed	<b>99.2</b> (1399/1422)	<b>4.4</b> (688/156)
	Basic spatial filter	99.4 (1356/1363)	24.4 (3816/156)
	Basic + Cloud clutter reject	99.3 (1353/1363)	21.6 (3376/156)
	Basic + Sun-glint reject	99.4 (1356/1363)	9.8 (1530/156)
Sector 5	Proposed	<b>99.3</b> (1353/1363)	<b>6.9</b> (1089/156)
	Basic spatial filter	99.7 (1079/1082)	32.0 (4999/156)
	Basic + Cloud clutter reject	99.3 (1074/1082)	30.4 (4745/156)
	Basic + Sun-glint reject	99.7 (1079/1082)	14.3 (2233/156)
	Proposed	<b>99.3</b> (1074/1082)	<b>12.6</b> (1979/156)

## 6. Conclusions

Reducing the number of false detections caused by clutter in small infrared target detection is quite challenging due to the point-like target nature. Clutters have different natures depending on the types, such as horizontal line clutter, cloud clutter in the sky and sea-glint in the sea. This paper presented a region segmentation method based on horizontal line detection using both the sensor pose information and image processing. In the horizontal region, the process of the local directional background removal filter (L-DBRF) after the modified mean subtraction filter (M-MSF) can reject the horizontal line clutter and achieve a high detection rate with few false alarms per image. In the sky region, the AdaBoost discriminative learning method was proposed to remove cloud clutter based on the target attribute feature, such as intensity, area, frequency, *etc.* According to the results of the AdaBoost-based target discrimination method on the test sequence, a false alarm reduction was achieved with only a small amount of degradation in the detection rate. In the sea region, separate spatio-temporal filtering was proposed to reject sea-glint. The temporal filter after a three plot correlation could reduce the sun-glint further. Through experimental comparisons, the proposed method was found to be robust for the detection of targets in a strong sun-glint environment using a low frame rate infrared camera, regardless of the target motion. In the final test, the proposed integrated clutter rejection scheme can effectively reduce the number of false detections by a factor of 2.5 to 9.4 with just 0.1%–0.8% degradation in the detection rate. Therefore, the proposed scheme is expected to be useful for sea-based infrared search and tracking systems.

## Acknowledgments

This work was supported by the 2013 Yeungnam University Research Grants.

## Author Contributions

The contributions were distributed between authors as follows: Sungho Kim wrote the text of the manuscript, programmed the target detection and clutter rejection methods, performed the in-depth discussion of the related literature and confirmed the accuracy experiments that are exclusive to this paper. Joohyoung Lee prepared the test database in various environments and pointed out the design parameters and clutter issues in the sea-based infrared search and track.

## Conflicts of Interest

The authors declare no conflict of interest.

## References

1. Campana, S.B. *The Infrared and Electro-Optical Systems Handbook*; SPIE Optical Engineering Press: Alexandria, VA, USA, 1993.
2. de Jong, A.N.IRST and perspective. *Proc. SPIE* **1995**, 2552, 206–213.
3. Wang, X.; Zhang, T. Clutter-adaptive infrared small target detection in infrared maritime scenarios. *Opt. Eng.* **2011**, 50, doi:10.1117/1.3582855.
4. Kim, S.; Lee, J. Double Layered-Background Removal Filter for Detecting Small Infrared Targets in Heterogenous Backgrounds. *J. Infrared Milli. Terahz Waves* **2011**, 32, 79–101.
5. Longmire, M.S.; Takken, E.H. LMS and matched digital filters for optical clutter suppression. *Appl. Opt.* **2003**, 27, 1141–1159.
6. Soni, T.; Zeidler, J.R.; Ku, W.H. Performance Evaluation of 2-D Adaptive Prediction Filters for Detection of Small Objects in Image Data. *IEEE Trans. Image Process.* **1993**, 2, 327–340.
7. Sang, H.; Shen, X.; Chen, C. Architecture of a configurable 2-D adaptive filter used for small object detection and digital image processing. *Opt. Eng.* **2003**, 48, 2182–2189.
8. Warren, R.C. Detection of Distant Airborne Targets in Cluttered Backgrounds in Infrared Image Sequences. Ph.D. Thesis, University of South Australia, Adelaide, Australia, 2002.
9. Sang, N.; Zhang, T.; Shi, W. Detection of Sea Surface Small Targets in Infrared Images based on Multi-level Filters. *Proc. SPIE* **1998**, 3373, 123–129.
10. Rivest, J.F.; Fortin, R. Detection of Dim Targets in Digital Infrared Imagery by Morphological Image Processing. *Opt. Eng.* **1996**, 35, 1886–1893.
11. Wang, Y.L.; Dai, J.M.; Sun, X.G.; Wang, Q. An efficient method of small targets detection in low SNR. *J. Phys.* **2006**, 48, 427–430.
12. Deshpande, S.D.; Er, M.H.; Venkateswarlu, R.; Chan, P. Max-Mean and Max-Median Filters for Detection of Small-targets. *Proc. SPIE* **1999**, 3809, 74–83.
13. van den Broek, S.P.; Bakker, E.J.; de Lange, D.J.; Theil, A. Detection and Classification of Infrared Decoys and Small Targets in a Sea Background. *Proc. SPIE* **2000**, 4029, 70–80.
14. Dijk, J.; van Eekeren, A.W.M.; Schutte, K.; de Lange, D.J.J. Point target detection using super-resolution reconstruction. *Proc. SPIE* **2007**, 6566, doi:10.1117/12.725074.
15. Kim, S. Min-local-LoG filter for detecting small targets in cluttered background. *Electron. Lett.* **2011**, 47, 105–106.
16. Reed, I.S.; Gagliardi, R.M.; Stotts, L.B. A Recursive Moving-target-indication Algorithm for Optical Image Sequences. *IEEE Trans. Aerospace Electron. Syst.* **1990**, 26, 434–440.
17. Rozovskii, B.; Petrov, A. Optimal Nonlinear Filtering for Track-before-Detect in IR Image Sequences. *Proc. SPIE* **1999**, 3809, 152–163.
18. Arnold, J.; Pasternack, H. Detection and Tracking of Low-Observable Targets through Dynamic Programming. *Proc. SPIE* **1990**, 1305, 207–217.
19. Chan, D.S.K. A Unified Framework for IR Target Detection and Tracking. *Proc. SPIE* **1992**, 1698, 66–76.

20. Caefer, C.E.; Mooney, J.M.; Silverman, J. Point Target Detection in Consecutive Frame Staring IR Imagery with Evolving Cloud Clutter. *Proc. SPIE* **1995**, *2561*, 14–24.
21. Silverman, J.; Mooney, J.M.; Caefer, C.E. Tracking Point Targets in Cloud Clutter. *Proc. SPIE* **1997**, *3061*, 496–507.
22. Tzannes, A.P.; Brooks, D.H. Point Target Detection in IR Image Sequences: A Hypothesis-Testing Approach based on Target and Clutter Temporal Profile Modeling. *Opt. Eng.* **2000**, *39*, 2270–2278.
23. Thiam, E.; Shue, L.; Venkateswarlu, R. Adaptive Mean and Variance Filter for Detection of Dim Point-Like Targets. *Proc. SPIE* **2002**, *4728*, 492–502.
24. Kim, S.; Sun, S.G.; Kim, K.T. Highly efficient supersonic small infrared target detection using temporal contrast filter. *Electron. Lett.* **2014**, *50*, 81–83.
25. Ronda, V.; Er, M.H.; Deshpande, S.D.; Chan, P. Multi-mode Algorithm for Detection and Tracking of Point-targets. *Proc. SPIE* **1999**, *3692*, 269–278.
26. Zhang, B.Z.T.; Cao, Z.; Zhang, K. Fast New Small-target Detection Algorithm based on a Modified Partial Differential Equation in Infrared Clutter. *Opt. Eng.* **2007**, *46*, doi:10.1117/1.2799509.
27. Wu, B.; Ji, H.B. Improved power-law-detector-based moving small dim target detection in infrared images. *Opt. Eng.* **2008**, *47*, doi:10.1117/1.2829771.
28. de Lange, H.B.D.J.J.; van den Broek, S.P.; Kemp, R.A.W.; Schwering, P.B.W. Automatic Detection of Small Surface Targets with Electro-Optical Sensors in a Harbor Environment. *Proc. SPIE* **2008**, *7114*, doi:10.1117/12.799813.
29. Bai, Z.; Zhou, F.; Jin, T.; Xie, Y. Infrared Small Target Detection and Tracking under the Conditions of Dim Target Intensity and Clutter Background. *Proc. SPIE* **2007**, *6786*, doi:10.1117/12.751691.
30. New, W.L.; Tand, M.J.; Er, M.H.; Venkateswarlu, R. New Method for Detection of Dim Point-Targets in InfraRed Images. *Proc. SPIE* **1999**, *3809*, 141–150.
31. Crosby, F. Signature Adaptive Target Detection and Threshold Selection for Constant False Alarm Rate. *J. Electron. Imaging* **2005**, *14*, doi:10.1117/1.1995710.
32. Khan, J.F.; Alam, M.S. Target Detection in Cluttered Forward-looking Infrared Imagery. *Opt. Eng.* **2005**, *44*, doi:10.1117/1.1950147.
33. Hubbard, W.A.; Page, G.A.; Carroll, B.D.; Manson, D.C. Feature Measurement Augmentation for a Dynamic Programming based IR Target Detection Algorithm in the Naval Environment. *Proc. SPIE* **1999**, *2698*, 2–9.
34. Shirvaikar, M.V.; Trivedi, M.M. A Neural Network Filter to Detect Small Targets in High Clutter Backgrounds. *IEEE Trans. Neural Netw.* **1995**, *6*, 252–257.
35. Kojima, A.; Sakurai, N.; Kishigami, J.I. Motion detection using 3D-FFT spectrum. In Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Minneapolis, MN, USA, 27–30 April 1993.
36. Strickland, R.N.; Hahn, H.I. Wavelet Transform Methods for Object Detection and Recovery. *IEEE Trans. Image Process.* **1997**, *6*, 724–735.
37. Boccignone, G.; Chianese, A.; Picariello, A. Small target detection using Wavlets. In Proceedings of International Conference on Pattern Recognition, Brisbane, Australia, 16–20 August 1998; pp. 1776–1778.
38. Ye, Z.; Wang, J.; Yu, R.; Jiang, Y.; Zou, Y. Infrared clutter rejection in detection of point targets. *Proc. SPIE* **2002**, *4077*, 533–537.

39. Zuo, Z.; Zhang, T. Detection of Sea Surface Small Targets in Infrared Images based on Multi-level Filters. *Proc. SPIE* **1999**, 3544, 372–377.
40. Yang, L.; Yang, J.; Yang, K. Adaptive Detection for Infrared Small Target under Sea-sky Complex Background. *Electron. Lett.* **2004**, 40, 1083–1085.
41. Soni, T.; Zeidler, R.; Ku, W.H. Recursive estimation techniques for detection of small objects in infrared image data. In Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), San Francisco, CA, USA, 23–26 March, 1992; Volume 3, pp. 581–584.
42. Watson, G.H.; Watson, S.K. The Detection of Moving Targets in Time-sequenced Imagery using Statistical Background Rejection. *Proc. SPIE* **1997**, 3163, 45–60.
43. Tartakovsky, A.; Blazek, R. Effective adaptive spatial-temporal technique for clutter rejection in IRST. *Proc. SPIE* **2000**, 4048, 85–95.
44. Lopez-Alonso, J.M.; Alda, J. Characterization of Dynamic Sea Scenarios with Infrared Imagers. *Infrared Phys. Technol.* **2005**, 46, 355–363.
45. Chen, Z.; Wang, G.; Liu, J.; Liu, C. Small Target Detection Algorithm based on Average Absolute Difference Maximum and Background Forecast. *Int. J. Infrared Milli. Waves* **2007**, 28, 87–97.
46. Peng, Z.; Zhang, Q. Dim Target Detection based on Nonlinear Multifeature Fusion by Karhunen-Loeve Transform. *Opt. Eng.* **2004**, 43, 2954–2958.
47. Chi, J.N.; Fu, P.; Wang, D.S.; Xu, X.H. A Detection Method of Infrared Image Small Target based on Order Morphology Transformation and Image Entropy Difference. In Proceedings of 2005 International Conference on Machine Learning and Cybernetics, Guangzhou, China, 18–21 August 2005; pp. 5111–5116.
48. Crosby, F. Glint Induced False Alarm Reduction in Signature Adaptive Target Detection. *Proc. SPIE* **2002**, 4726, 285–294.
49. Toet, A. Detection of Dim Point Targets in Cluttered Maritime Backgrounds through Multisensor Image Fusion. *Proc. SPIE* **2002**, 4718, 118–129.
50. Lim, E.T.; Shue, L.; Ronda, V. Multi-mode Fusion Algorithm for Robust Dim Point-like Target Detection. *Proc. SPIE* **2003**, 5082, 94–102.
51. Hartley, R.I.; Zisserman, A. *Multiple View Geometry in Computer Vision*, 2nd ed.; Cambridge University Press: Cambridge, UK, 2004.
52. Hanson, R.; Norris, M. Analysis of Measurements Based on the Singular Value Decomposition. *SIAM J. Sci. Stat. Comput.* **1981**, 2, 363–373.
53. Missirian, J.M.; Ducruet, L. IRST: a key system in modern warfare. *Proc. SPIE* **1997**, 3061, 554–565.
54. Maltese, D.; Deyla, O.; Vernet, G.; Preux, C. New generation of naval IRST: Example of EOMS NG. *Proc. SPIE* **2010**, 7660, doi:10.1117/12.850066.
55. Duda, R.O.; Hart, P.E.; Stork, D.G. *Pattern Classification*, 2nd ed.; Wiley-Interscience: New York, NY, USA, 2001.
56. Toet, A.; Wu, T. Small maritime target detection through false color fusion. *Proc. SPIE* **2008**, 6945, doi:10.1117/12.773279.
57. Csurka, G.; Dance, C.R.; Fan, L.; Willamowski, J.; Bray, C. Visual categorization with bags of keypoints. In Proceedings of Workshop on Statistical Learning in Computer Vision, Prague, Czech Republic, 15 May 2004; pp. 1–22.

58. Maji, S.; Berg, A.C.; Malik, J. Classification using intersection kernel support vector machines is efficient. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, Anchorage, AK, USA, 24–26 June 2008; pp. 1–8.
59. Torralba, A.; Murphy, K.P.; Freeman, W.T. Sharing Visual Features for Multiclass and Multiview Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 854–869.
60. Kim, S.; Yang, Y.; Choi, B. Realistic infrared sequence generation by physics-based infrared target modeling for infrared search and track. *Opt. Eng.* **2010**, *49*, doi:10.1117/1.3509363.



# Robust Pedestrian Tracking and Recognition from FLIR Video: A Unified Approach via Sparse Coding

Xin Li, Rui Guo and Chao Chen

**Abstract:** Sparse coding is an emerging method that has been successfully applied to both robust object tracking and recognition in the vision literature. In this paper, we propose to explore a sparse coding-based approach toward joint object tracking-and-recognition and explore its potential in the analysis of forward-looking infrared (FLIR) video to support nighttime machine vision systems. A key technical contribution of this work is to unify existing sparse coding-based approaches toward tracking and recognition under the same framework, so that they can benefit from each other in a closed-loop. On the one hand, tracking the same object through temporal frames allows us to achieve improved recognition performance through dynamical updating of template/dictionary and combining multiple recognition results; on the other hand, the recognition of individual objects facilitates the tracking of multiple objects (*i.e.*, walking pedestrians), especially in the presence of occlusion within a crowded environment. We report experimental results on both the CASIAPedestrian Database and our own collected FLIR video database to demonstrate the effectiveness of the proposed joint tracking-and-recognition approach.

Reprinted from *Sensors*. Cite as: Li, X.; Guo, R.; Chen, C. Robust Pedestrian Tracking and Recognition from FLIR Video: A Unified Approach via Sparse Coding. *Sensors* **2014**, *14*, 11245–11259.

## 1. Introduction

The capability of recognizing a person at a distance in nighttime environments, which we call remote and night biometrics, has gained increasingly more attention in recent years. Fast advances in sensor technology (e.g., infrared cameras) and biometric systems (e.g., video-based recognition) have facilitated the task of remote and night biometrics. Object tracking and recognition are two basic building blocks in almost all video-based biometrics systems, including forward-looking infrared (FLIR)-based ones. The literature of object detection/tracking, face recognition and visual surveillance is huge; for recent advances, please refer to [1–3] and their references; pedestrian detection and tracking from FLIR video has also been studied in [4–7]. However, the relationship between detection/tracking and recognition has not been well studied in the literature. To the best of our knowledge, joint tracking and recognition has been considered under the context of particle filtering [8] only and specifically in the scenario of face biometrics [9].

In this paper, we propose to tackle joint object tracking and recognition under a unified sparse coding-based framework. Sparse coding originated from the research on compressed sensing theory [10] and has been recently leveraged into the problems of robust object tracking [11–13] and robust face recognition [14,15]. For both tracking and recognition problems, the target patch/template of interest is sparsely represented in the space spanned by the dictionary (a collection of matching templates); and the final result is given by the candidate with the smallest projection

error. Such a similarity motivates us to cast the two problems under the same framework and solve them simultaneously, *i.e.*, unlike previous works assuming a dictionary of templates (e.g., face portions) already cropped from the original image/video, ours obtains this dictionary by dynamically tracking the target of interest (e.g., a walking pedestrian).

We argue that tracking and recognition can benefit from each other for the following reasons. On the one hand, robust tracking of an object under a particle filter framework [16] often involves the updating of the matching templates on-the-fly. Such a dynamical strategy of template updating helps overcome the difficulties with occlusion and the cluttered background, which are also common adversary factors to the task of robust recognition. Moreover, persistently tracking allows the system to temporally combine the recognition results across multiple frames for improved accuracy (since we know it is the same object that has been tracked) [17,18]. On the other hand, high-level vision tasks, such as recognition, often facilitates those at lower levels, including tracking, especially in the situation of multiple targets being involved [19]. More specifically, we suggest that the recognition result can be exploited by the template updating strategy to better fight against occlusion and a cluttered background. Such tracking-by-recognition offers some new insight to the challenging problem of multi-target tracking, which was often tackled by an energy optimization approach [20].

When applied to remote and night biometrics systems, the proposed approach has several advantages over other competing ones (e.g., gait-based [21] or silhouette-based [22]). First, previous approaches mostly count on image/video segmentation to extract relevant gait or silhouette information before recognition; consequently, segmentation errors have a significant impact on the accuracy of recognition [23]. By contrast, the proposed one directly works with image patches and does not involve any cropping or segmentation at all (note that in many previous works, such as [14], it is assumed that cropped image patches are already available). Second, it is widely known that occlusions and background clutters are often primary obstacles to various vision tasks, including tracking and recognition. Sparse coding has shown great potential in fighting against those adversary factors, thanks to the power of collaborative representation [15] (please refer to the Experimental Results section). Third, the unification of tracking and recognition allows us to jointly optimize these intrinsically connected components, which is highly desirable in the scenario of handling complicated cases, such as multi-target tracking in a crowd [24]. In other words, tracking and recognition can be viewed as two sides of the same coin: One helps the other and *vice versa*.

## 2. Background on Sparse Coding

In this section, we review the current state-of-the-art in sparse coding and its applications into object tracking/recognition [25]. The basic idea behind sparse coding is to approximate a signal of interest  $\mathbf{x} \in R^n$  by linear combination of a small number of atoms (elements in a dictionary  $\mathbf{A}_{m \times n}$ ); namely,  $\mathbf{x}_{m \times 1} = \mathbf{D}_{m \times n} \mathbf{a}_{n \times 1}$ , where  $\mathbf{a}$  is the vector of sparse coefficients. Ideally, the sparsity constraint is enforced about the total number of nonzero coefficients in  $\mathbf{a}$ , which gives rise to the following constrained optimization problem:

$$\min_{\mathbf{a}} \|\mathbf{a}\|_0 \text{ subject to } \|\mathbf{x} - \mathbf{A}\mathbf{a}\| \leq \epsilon \quad (1)$$

However, the above problem is known to be NP-hard [26], and it is often suggested that the original  $l_0$ -norm be replaced by its  $l_1$  counterpart. That is, one considers the following computationally tractable formulation:

$$\min_{\mathbf{a}} \|\mathbf{a}\|_1 + \lambda \|\mathbf{x} - \mathbf{A}\mathbf{a}\| \quad (2)$$

where  $\lambda$  is the Lagrangian multiplier converting the constrained optimization into an unconstrained one [27]. Various algorithms have been developed in recent years to solve this class of  $l_1$ -minimization problems (for a recent review, please refer to [28] and its references). Meanwhile, it is amazing to witness that many engineering problems across different disciplines can be reformulated into a variant of  $l_1$ -minimization problem. Within the scope of this paper, we opt to review two of them; namely, object tracking and object recognition.

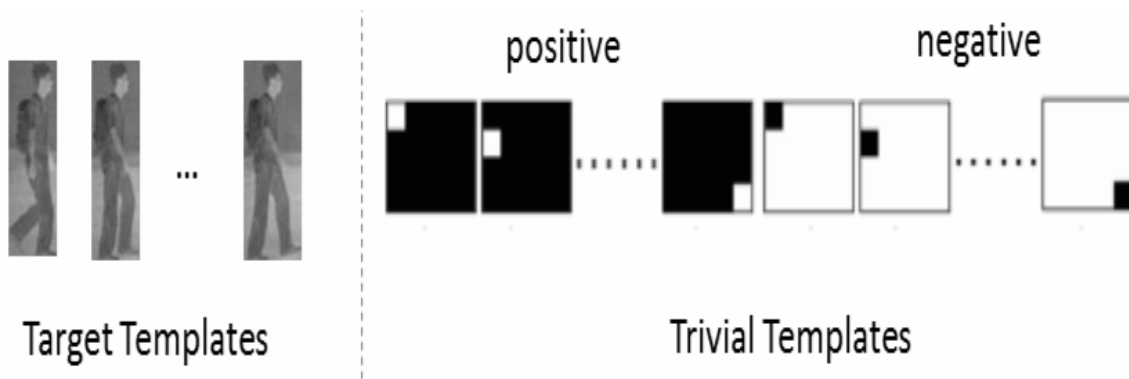
### 2.1. Sparse Coding for Object Tracking

The fundamental assumption for appearance-based object tracking is that the global appearance of an object, despite varying illumination and viewpoint conditions, is still characterized by a low-dimensional space. Under the context of appearance-based object tracking, dictionary  $\mathbf{A}$  is decomposed of target templates (image patches in  $R^m$ ), as well as a collection of trivial templates (to model occlusion and noise in the real-world observation data), as shown in Figure 1. If one writes  $\mathbf{A}$  as:

$$\mathbf{x}_{m \times 1} = [\mathbf{T} \mathbf{I} - \mathbf{I}][\mathbf{b} \mathbf{e}^+ \mathbf{e}^-]^t = \mathbf{A}_{m \times (n+2m)} \mathbf{a}_{(n+2m) \times 1} \quad (3)$$

where  $\mathbf{T} = [\mathbf{t}_1, \dots, \mathbf{t}_n]$  denotes  $n$  target templates (note that  $m \gg n$ ) and  $\mathbf{e}^+, \mathbf{e}^- \in R^m$  correspond to positive/negative trivial coefficient vectors, respectively.

**Figure 1.** Decomposition of a dictionary into target and trivial templates in sparse coding-based object tracking.



For a good target candidate, there are only a small number of nonzero coefficients in positive and negative trivial coefficients accounting for the noise and partial occlusion. Such an observation has led to the formulation of object tracking into a  $l_1$ -minimization problem, as proposed in [11,29–31]. The final tracking result is obtained by finding the smallest residual after projecting onto the subspace spanned by target templates, *i.e.*,  $\|\mathbf{x} - \mathbf{T}\mathbf{b}\|_2$ . Under a particle

filtering framework [16], such minimum-error tracking admits a maximum *a posteriori* probability interpretation. Further improvement on robustness tracking can be brought by the idea of template updating. More specifically, the  $l_2$ -norm of template  $\mathbf{t}_i$  intuitively indicates its significance to tracking; therefore, it is plausible to eliminate the template of the least weight and replace it by the newly-obtained successful tracking result.

## 2.2. Sparse Coding for Object Recognition

Based on a similar observation to tracking, one can assume that the appearance of each individual subject lies in a unique low-dimensional subspace, and the structure of this subspace can be exploited to distinguish the subject of interest from others [14]. Therefore, if we consider a collection of  $k$  subjects, each containing  $n$  templates  $\mathbf{t}_{i,j} \in R^m$  (again,  $m$  is the size of the template of the image patch), the dictionary  $\mathbf{A}_{m \times N}$  will consist of  $N = nk$  elements. For any given inquiry template  $\mathbf{x}$ , one can formulate the following sparse coding problem:

$$\min_{\mathbf{a}} \|\mathbf{a}\|_1 + \lambda \|\mathbf{x}_{m \times 1} - \mathbf{A}_{m \times N} \mathbf{a}_{N \times 1}\| \quad (4)$$

where sparse coefficients  $\mathbf{a}$  will be exploited to tell which subspace the inquiry is associated with. Ideally, the sparsest solution will associate the inquiry with the group of templates from a single subject class. However, due to noise and modeling errors, inference from other competing classes might arise; in other words, one might observe small nonzero entries associated with several subject classes. Therefore, it is often desirable to identify the subject by a twist of the above minimum-error strategy; namely, one can calculate the residual errors after projecting onto the subspace spanned by each class of target templates [14]:

$$E(i) = \|\mathbf{x}_{m \times 1} - \mathbf{A}_{m \times N} \delta_{(i)}(\mathbf{a}_{N \times 1})\| \quad (5)$$

where  $\delta_{(i)}(\mathbf{a})$  is the characteristic function that assigns ones to the entries associated with subject  $i$  in  $\mathbf{a}$ . Then, the identity of inquiry  $\mathbf{x}$  is obtained by  $Id = \operatorname{argmin}_i E(i), 1 \leq i \leq k$ .

As articulated in [15], it is the idea of collaborative representation—namely, the formulation of joint dictionary  $\mathbf{A}$ —that contributes to the good performance of Equation (5) in robust face recognition. It has been shown that replacing  $l_1$ -norm by its  $l_2$ -counterpart achieves comparable recognition performance, even though the computational complexity of the solution algorithm can be dramatically reduced (since the regularized least-square problem admits the analytical solution). When compared against previous  $l_2$ -based approaches (e.g., eigen-face [32]), we note that it is collaborative representation that enforces the global constraint on the collection of appearance subspaces spanned by individual subjects. In other words, the competition among sparse coefficients  $a_i$  contributes to the effectiveness of the winner-take-all strategy, and therefore, it is possible to obtain robust recognition by searching for the smallest projection errors.

Despite the use of sparse coding in both object tracking and recognition, it should be emphasized that the relationship between them has not been studied in the open literature. To the best of our knowledge, joint tracking-and-recognition has only been addressed in two isolated scenarios: one is to embed them into a single particle filtering framework [8], and the other is to integrate tracking with recognition specially for the class of face biometrics [9]. The apparent similarity between

Equations (3) and (5) inspires us to explore a unified sparse coding-based approach toward joint tracking-and-recognition. The primary objective of this paper is to demonstrate that such a joint approach can offer several new insights into the design of robust vision systems and find niche applications in challenging environments, such as remote and night biometrics using FLIR data.

### 3. Joint Tracking-and-Recognition: A Unified Approach via Sparse Coding

In this paper, we formally define a joint tracking-and-recognition problem as follows. Given an inquiry FLIR video  $X$  containing walking pedestrians and a database of  $k$  subjects each associated with  $n$  video segments (training samples), establish the identity of the inquiry video. Note that unlike previous studies, [8] and [9], in which only one subject is considered, tracking and recognition are more tightly twisted in our multi-subject formulation (*i.e.*, one has to simultaneously track and recognize multiple subjects). At first sight, the interference among multiple subjects (e.g., one person could become occluded due to another person's presence) makes the joint tracking-and-recognition problem a lot more challenging than the single-subject scenario. To overcome this difficulty, we propose to gain a deeper understanding between tracking and recognition in this section.

#### 3.1. Tracking-for-Recognition: Exploiting Temporal Redundancy

We first consider a simplified scenario where only one walking pedestrian is present in the inquiry video. When no interference is present, tracking a single pedestrian is a solved problem, and the recognition subproblem can be solved by sparse coding in a similar fashion to face recognition [14]. A more interesting question is: how can tracking help recognition? Here, we present a Bayesian interpretation of sparse coding-based recognition [14], which facilitates the exploitation of temporal redundancy arising from tracking a target template in the inquiry video. The key observation behind tracking-for-recognition lies in the fact that if it is known as *a priori* that multiple templates are associated with the same identity, such information can be exploited by the recognition system to improve the accuracy. Each template can be viewed as an independent classifier, and accordingly, the idea of combining classifiers [33] can be easily implemented under the sparse coding framework.

Following the same notation used above, we consider a dictionary  $\mathbf{A}_{m \times N}$  consisting of  $k$  subjects each containing  $n$  templates  $\mathbf{t}_{i,j} \in R^m$  ( $N = nk$ ). The subspace constraint of the appearance model for subject  $i$  ( $1 \leq i \leq k$ ) implies that a target template  $\mathbf{x}$  associated with subject  $i$  can be best approximated by the following sparse coding strategy:

$$\mathbf{x} \approx \mathbf{A}\delta_{(i)}(\mathbf{a}), \quad (6)$$

where  $\delta_{(i)}(\mathbf{a})$  is a binary vector in  $R^N$ , whose only nonzero elements are located at  $j = (i - 1) * n + 1, \dots, i * n$  (*i.e.*, those associated with subject  $i$ ). If the approximation error is given by  $E(i) = \mathbf{x} - \mathbf{A}_{m \times N}\delta_{(i)}(\mathbf{a}_{N \times 1})$  and assumed to observe an i.i.d. Gaussian model  $N(0, \sigma_w^2)$ , then the likelihood function of observing a template  $\mathbf{x}_i$  given subject  $i$  (denoted by  $w_i$ ) can be written as:

$$p(\mathbf{x}|w_i) \approx \exp\left(-\frac{\|E(i)\|_2^2}{2\sigma_w^2}\right) \quad (7)$$

Now, it follows from the Bayesian formula that the maximum *a posteriori* (MAP) classification of a given template  $\mathbf{x}$  can be obtained from:

$$\max_i p(w_i|\mathbf{x}) = \max_i \frac{p(\mathbf{x}|w_i)p(w_i)}{p(\mathbf{x})} \quad (8)$$

which implies the equivalence between the MAP strategy in the Bayesian classifier and the minimum-distance classifier of Equation (5) used in SCR. Such a connection allows us to conveniently exploit the temporal redundancy of an inquiry video under the framework of combining classifiers, as we will elaborate next.

Similar to the setup in [33], we use  $\{w_1, \dots, w_k\}$  to denote  $k$  different classes of subjects/identities and  $\{\mathbf{x}_1, \dots, \mathbf{x}_l\}$  the collection of measurement vectors. Given an inquiry FLIR video  $X$ , those measurement vectors are obtained by tracking a single target template  $\mathbf{x}$  across multiple frames. Therefore, a Bayesian classifier works by assigning the label  $Id = \max_i p(w_i|\mathbf{x}_1, \dots, \mathbf{x}_l)$ , which, in turn, can be written as:

$$\max_i p(w_i|\mathbf{x}_1, \dots, \mathbf{x}_l) = \max_i \frac{p(\mathbf{x}_1, \dots, \mathbf{x}_l|w_i)p(w_i)}{p(\mathbf{x}_1, \dots, \mathbf{x}_l)} \quad (9)$$

Under the assumption that all measurement vectors are conditionally statistically independent, we have:

$$p(\mathbf{x}_1, \dots, \mathbf{x}_l|w_i) = \prod_{j=1}^l p(\mathbf{x}_j|w_i) \quad (10)$$

Substituting Equations (7) and (8) into Equation (10), we can obtain the so-called feature-level fusion strategy:

$$p(\mathbf{x}_1, \dots, \mathbf{x}_l|w_i) \approx \exp\left(-\frac{\sum_{j=1}^l \|E_j(i)\|_2^2}{2\sigma_j^2}\right) \quad (11)$$

Therefore, the MAP decision boils down to a generalized minimum-distance classifier defined with respect to the group of measurement vectors. Alternatively, as suggested in [33], one can combine the decision outcomes instead of posterior probabilities, e.g., the final decision can be made by either sum rule  $Id = \operatorname{argmin}_i \sum_{j=1}^l \|E_j(i)\|_2^2$ ,  $1 \leq i \leq k$  or majority-vote rule  $Id = \operatorname{mode}\{Id_1, \dots, Id_l\}$ , where  $Id_j$  is the label returned by applying the minimum-distance classifier of Equation (8) to measurement vector  $\mathbf{x}_j$ . Even though the benefit of combining classifiers has been well-established in the literature (e.g., refer to [34]), the relationship between the number of classifiers  $l$  and performance gain is not. As we will show in the Experimental Results, even a small number of  $l$  ( $<10$  frames) measurement vectors can dramatically boost the recognition accuracy.

### 3.2. Tracking-by-Recognition: Nonlocal Template Updating

Now, let us consider the more general situation: a multi-subject extension of the above joint tracking-and-recognition problem. In the literature, the problem of multi-object tracking is often addressed under the framework of energy minimization (e.g., refer to [35,36] and their references). Two common technical challenges with tracking multiple objects is that the space of all possible trajectories is large and the appearance of a target might vary dramatically, due to the presence

of occlusion or illumination variations. Consequently, it often requires special attention to design an appropriate cost function and a fast search strategy to solve the multi-object tracking problem. By contrast, we propose to cast multi-object tracking under the framework of sparse coding and explore the question of how the recognition result could help a multi-object tracking algorithm fight against adversary factors, such as occlusion and illumination variations. The basic assumption behind our tracking-by-recognition approach is that as long as the problem of multi-object tracking can be solved in a robust fashion, the recognition of multiple objects becomes straightforward (e.g., based on what we have discussed in the previous subsection on tracking-for-recognition).

The key observation behind our tracking-by-recognition is that one person's appearance along the moving trajectory behaves like the noise to the tracking of another person. For this reason, only the person of interest (that has been recognized) contributes to the formation of dictionary  $\mathbf{A}$  in sparse coding-based tracking; all others can be handled the same way as background clutter. In other words, recognition facilitates the multi-object tracking problem by recognizing that for each appearance subspace of an individual subject, all other subjects, as well as the background can be modeled by the outliers. Such an observation leads us to rethink the template updating strategy proposed in [11], where the least-important template is eliminated from the dictionary and  $\omega_i = \|\mathbf{t}_i\|_2$  is adopted to quantify the importance of a template  $\mathbf{t}_i$ . Empirical studies have shown that such strategy is highly sensitive to occlusion, due to the reasons listed above. Instead, we propose a nonlocal alternative strategy of template updating; based on the recognition result, one can switch to a default set of templates upon the suspicion of occlusion. One way of implementing such a strategy is to save a copy of templates that have been recognized to be the same person (but likely in the distant history or even in the training set).

It is enlightening to appreciate the advantage of the above tracking-by-recognition formulation for multi-object tracking over existing energy minimization approaches. In energy minimization approaches, occlusion handling is often a thorny issue to address when coming up with an appropriate energy term for multi-object tracking. For example, a sophisticated global occlusion reasoning strategy is studied in [36], where a principled modeling of occlusion remains elusive, due to the complex dependency between a target's visibility and other targets' trajectories. By contrast, we argue that if the ultimate objective of the surveillance system is to recognize walking pedestrians, one can get around the tricky occlusion issue by stopping the tracker. In other words, the continuity of motion trajectory is unnecessary for the task of recognition; what matters is only the accumulated group size of measurement vectors (occlusion will reduce this size, but there is no need for accurate occlusion detection). In other words, tracking and recognition are essentially two sides of the same coin: tracking where a target template goes in the next frame is conceptually equivalent to recognizing whether a new hypothesized template in the next frame still belongs to the same class as the target one. With the recognition result available, tracking can always rely on a more trustworthy source (e.g., nonlocal rather than local) for template updating.

## 4. Experimental Results

### 4.1. Experimental Setup

In this section, we report our experimental results with two FLIR pedestrian databases: one is collected by CASIA (Dataset C in the CASIA Gait Database, Publicly available at <http://www.cbsr.ia.ac.cn/english/Databases.asp>), and the other is collected at the WVU Erickson Alumni Center (not publicly available, but it can be requested from [http://www.citer.wvu.edu/biometric\\_dataset\\_collections](http://www.citer.wvu.edu/biometric_dataset_collections)). The CASIA Dataset C contains 153 subjects, each of which contains 11 video clips acquired by an FLIR camera. Each subject passes through the scene with and without carrying a bag, as well as at varying walking speeds; although silhouettes of those 153 subjects are supplied, we have found that they are error-prone, and therefore, we do not utilize them in our approach. The WVU dataset contains 30 subjects (18 males and 12 females) walking at three planned camera distances: 20, 25 and 30 m. In addition to the bag carrying option, the protocol includes both single-person and double-person scenarios. In the latter, two persons walk toward each other, one carrying a bag and the other empty-handed; when they meet halfway, the bag will be handed to the other; then, they walk away from each other. Both occlusion and carrying a bag are adversary factors to pedestrian tracking and recognition in this setup.

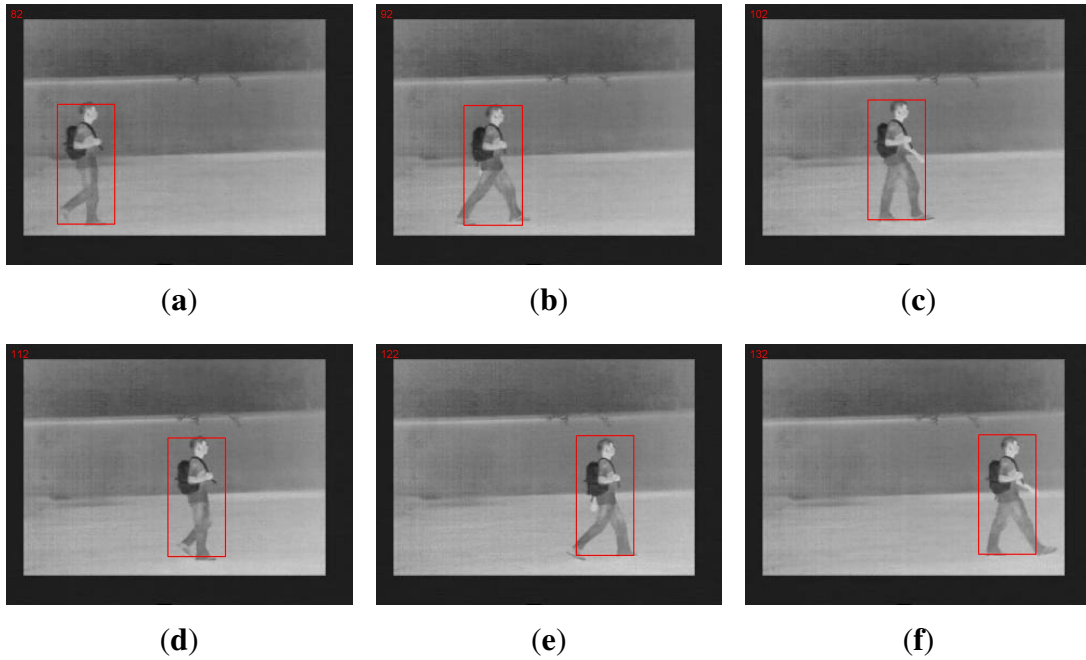
To promote reproducible research, the source codes and saved experimental results accompanying this research can be accessed at <http://www.csee.wvu.edu/xin/code/FLIR.zip>. In our MATLAB-based implementation, we have built upon two previous releases of sparse coding for tracking and recognition. The source codes of sparse coding for  $l_1$ -based tracking and recognition have been obtained from [http://www.dabi.temple.edu/hbling/code\\_data.htm#L1\\_Tracker](http://www.dabi.temple.edu/hbling/code_data.htm#L1_Tracker) and <http://www.eecs.berkeley.edu/yang/software/l1benchmark/>. More specifically, the dictionary needed for sparse coding-based recognition is obtained from the tracking result; we simply normalize the cropped templates to a common size. For the CASIA Dataset C, the following parameter setting is adopted:  $k = 153, n = 40$ .

### 4.2. Single-Object and Multi-Object Tracking

We first demonstrate the tracking result for single-object tracking. Figure 2 shows a collection of sample frames obtained from one typical FLIR video of CASIA Dataset C by  $l_1$ -based tracking. Since the background is relatively simple and only one pedestrian is present, the tracking is not a challenging issue for this data set. The new insight supplied by this experiment lies in that  $l_1$ -based tracking offers an automatic and robust cropping tool to obtain matching templates; *i.e.*, the elements of dictionary  $\mathbf{A}$ . Note that the length of even a short video segment is a few seconds, which implies that at least dozens (or even hundreds) of matching templates can be cropped from the video clip. We note that this fact suggests that there is a significant amount of temporal redundancy that can be exploited by the recognition component.

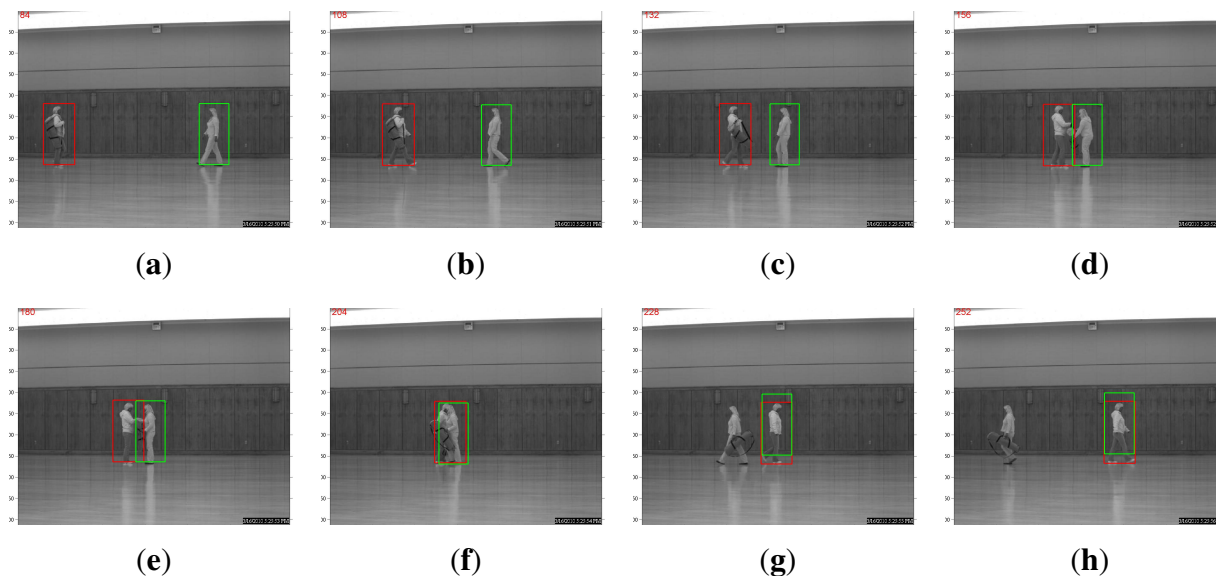


**Figure 2.** Sample tracking results for the forward-looking infrared (FLIR) video (red boxes highlight the locations of the walking pedestrian).

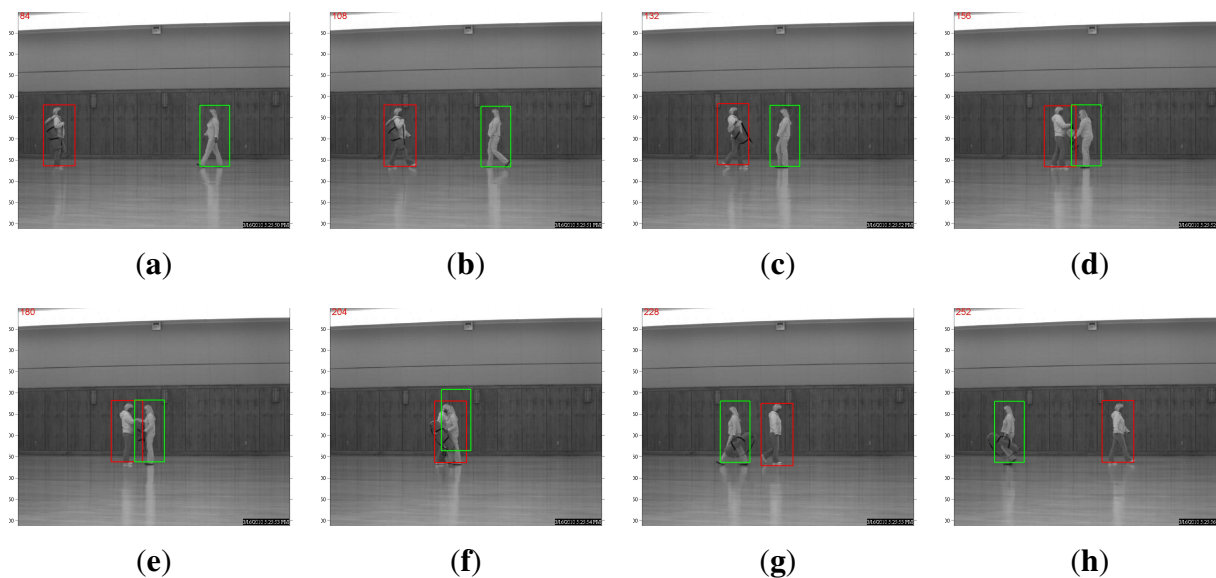


A more interesting comparison result is in the scenario of multi-object tracking. For example, the WVU dataset contains test sequences in which two persons walk toward each other. When the two pedestrians meet, one hands the bag to the other, and then, they continue walking away from each other. Such a protocol dictates that occlusion is present for a relatively long period of time. As shown in Figure 3, the straightforward application of the  $l_1$ -based tracking algorithm in [11] expectedly fails at the occlusion. The algorithm will be confused by the overlap of target templates associated with two pedestrians. By contrast, a recognition-based, nonlocal, template-updating strategy proposed in the previous section can produce robust and accurate tracking, even after one person hands the bag to the other (note that there are significant variations in terms of appearance), as shown in Figure 4. This is because when occlusion occurs, the recognition-based strategy will update the template stored from a distance past (in other words, nonlocal becomes more trustworthy than the local temporal neighborhood). Such experimental results justify the effectiveness of our tracking-by-recognition approach.

**Figure 3.** Tracking failure result obtained by [11] due to occlusion (after the two persons pass by each other, the tracking algorithm got confused; both red and green boxes get attached to the pedestrian walking to the right).



**Figure 4.** Joint tracking-and-recognition is capable of persistently tracking both pedestrians regardless of the occlusion and bad exchange (both red and green boxes are correctly associated with the correct identity).

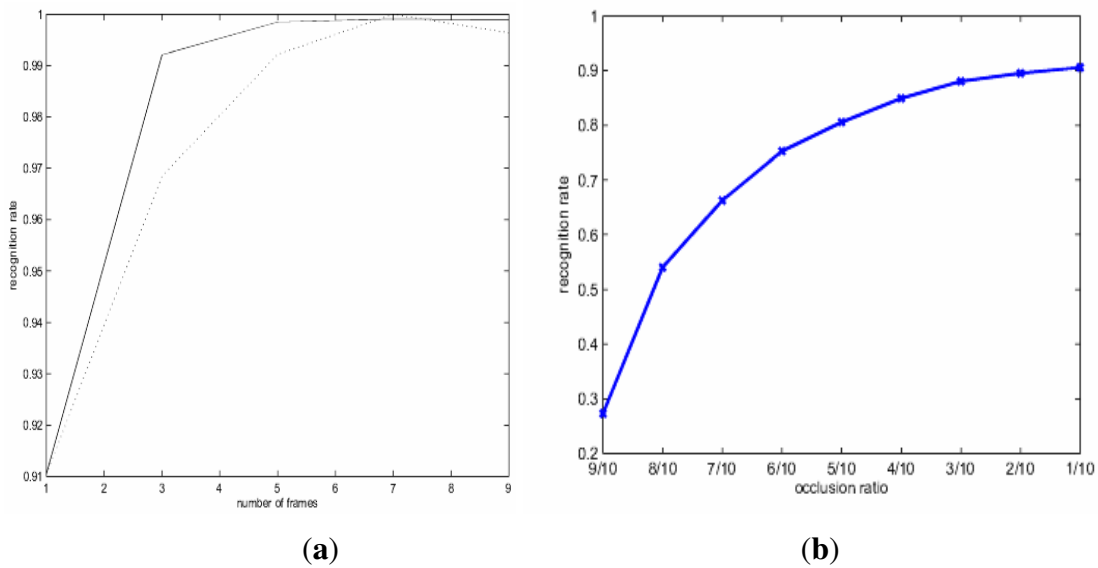


#### 4.3. Robust Pedestrian Recognition from FLIR Video

Next, we report our experimental results with sparse coding-based recognition. In particular, we want to explore the gain brought by exploiting temporal redundancy (through combining classifiers) and the impact of occlusion on recognition performance. In the first experiment, we change the

parameter  $l$ —the size of measurement vectors or the total number of frames for which we have successfully tracked for the inquiry video  $X$ . Two rules of combining the classification results have been implemented: sum *vs.* majority vote. Figure 5a shows how the accuracy of recognition evolves as  $l$  varies: it can be observed that the gain improves rapidly as  $l$  increases and quickly saturates. Therefore, even when a small number of measurement vectors (e.g.,  $l = 9$  or  $< \frac{1}{3}$  second for 30 fps of video) is available, highly accurate recognition (close to 100%) is possible thanks to the power of temporal fusion. By contrast, we note that the best recognition performance reported for this data set is 96% in the open literature (e.g., gait energy image based [21]). Such a finding seems to suggest that video-based biometrics has a lot more potential than image-based, thanks to the blessing of redundancy.

**Figure 5.** The recognition performance of sparse coding-based recognition: (a) exploiting temporal redundancy improves the recognition accuracy (solid: sum rule; dashed: majority voting); (b) the recognition performance gracefully degrades as the occlusion ratio increases (no temporal fusion involved  $l = 1$ ).



In the second experiment, we artificially mask a certain percentage of the inquiry template (e.g., to simulate how the lower part of human body is occluded by bushes or deep grass in a real-world scenario) and test the performance of sparse coding-based recognition (no fusion is involved, *i.e.*,  $l = 1$ ). Figure 5b includes the result for the masking percentage varying from 10 to 90. It can be observed that sparse coding-based recognition is indeed insensitive to occlusion to some degree: about 30% occlusion degrades the recognition performance by about 5%. This is not surprising, because the lower part of the human body is not as discriminating as the upper part (more theoretical justifications can be found in the paper [14]). Combined with the result in Figure 5a, we conclude that when spatial clue becomes less reliable (e.g., due to occlusion), it is plausible to exploit temporal ones by a strategy, such as tracking-for-recognition.

Finally, we use experimental results to clarify the importance of obtaining a good dictionary for sparse coding-based recognition. One basic assumption behind sparse coding-based recognition is that the dictionary contains a densely sampled representation of appearance subspace; such an assumption is not always valid in practical situations. For instance, if the training set and testing set are significantly different (e.g., without and with a bag), the accuracy of recognition will be affected. Table 1 includes the experimental results of sparse coding-based recognition on CASIA Dataset C for a variety of different training/testing set situations. It shows that the walking speed of the pedestrian has a minor impact on the recognition performance; while the effect of carrying a bag or not is substantial. This is in contrast to what we have observed for the tracking experiments, where handing a bag over does not affect the result much. Nevertheless, the recognition accuracy achieved by SCR (even in the situation of no fusion being involved) is at least comparable to the template-matching-based approach, as reported in [37]. One can expect that much better recognition performance can be obtained by temporal fusion, as we have shown above.

**Table 1.** The recognition performance of the baseline algorithm for the training/testing data of different conditions.

Training	Testing	This Work	[37]
Normal	Normal	91.05%	94%
Normal	Slow	84.05%	85%
Normal	Fast	88.35%	88%
Slow	Normal	81.24%	-
Fast	Normal	83.70%	-
with bag	with bag	93.56%	-
w/o bag	w/o bag	92.94%	-
w/o bag	with bag	57.61%	51%
with bag	w/o bag	49.75%	-

## 5. Conclusions

In this paper, we studied a unified approach toward robust pedestrian tracking and recognition from FLIR video via sparse coding. Under the joint tracking-and-recognition framework, tracking helps recognition by generating matching templates needed for the dictionary and by facilitating the exploitation of temporal redundancy; recognition helps multi-object recognition by supplying a nonlocal template updating strategy instead of a local one. The main contributions of this work include: (1) an automatic night biometrics system capable of tracking and recognizing pedestrians from infrared video; and (2) an extension of sparse coding-based tracking from a single target to multiple targets, enabled by the proposed recognition-based template updating strategy. We have reported our experimental results on two FLIR video data sets: the CASIA gait database and the WVU Infrared Pedestrian database. On the former, we show how joint tracking-and-recognition can improve the accuracy and robustness of sparse coding-based recognition; on the latter, we

demonstrate that the nonlocal template updating strategy based on the recognition result is capable of boosting the performance of sparse coding-based tracking in the presence of occlusion.

### Acknowledgments

We want to acknowledge the authors of [11] and [28] for making their MATLAB codes available.

### Conflicts of Interest

The authors declare no conflict of interest.

### References

1. Yilmaz, A.; Javed, O.; Shah, M. Object tracking: A survey. *ACM Comput. Surv.* **2006**, *38*, 13.
2. Zhao, W.; Chellappa, R.; Rosenfeld, A.; Phillips, P. Face recognition: A literature survey. *ACM Comput. Surv.* **2003**, *35*, 399–458.
3. Hu, W.; Tan, T.; Wang, L.; Maybank, S. A survey on visual surveillance of object motion and behaviors. *IEEE Trans. Syst. Man Cybern. Part C: Appl. Rev.* **2004**, *34*, 334–352.
4. Nanda, H.; Davis, L. Probabilistic template based pedestrian detection in infrared videos. *IEEE Intell. Veh. Symp.* **2002**, *1*, 45–52.
5. Xu, F.; Liu, X.; Fujimura, K. Pedestrian detection and tracking with night vision. *IEEE Trans. Intell. Transp. Syst.* **2005**, *6*, 63–71.
6. Suard, F.; Rakotomamonjy, A.; Bensrhair, A.; Broggi, A. Pedestrian detection using infrared images and histograms of oriented gradients. In Proceedings of the IEEE Intelligent Vehicles Symposium, Tokyo, Japan, 2006; pp. 206–212.
7. Dai, C.; Zheng, Y.; Li, X. Pedestrian detection and tracking in infrared imagery using shape and appearance. *Comput. Vis. Image Underst.* **2007**, *106*, 288–299.
8. Zhou, S.K.; Chellappa, R.; Moghaddam, B. Visual tracking and recognition using appearance-adaptive models in particle filters. *IEEE Trans. Image Process.* **2004**, *13*, 1491–1506.
9. Lee, K.-C.; Ho, J.; Yang, M.-H.; Kriegman, D. Visual tracking and recognition using probabilistic appearance manifolds. *Comput. Vis. Image Underst.* **2005**, *99*, 303–331.
10. Candès, E.J.; Romberg, J.K.; Tao, T. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory* **2006**, *52*, 489–509.
11. Mei, X.; Ling, H. Robust visual tracking using l1 minimization. In Proceedings of the IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 27 September–4 October 2009; pp. 1436–1443.
12. Li, H.; Shen, C.; Shi, Q. Real-time visual tracking using compressive sensing. In Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 20–25 June 2011; pp. 1305–1312.

13. Zhang, S.; Yao, H.; Sun, X.; Lu, X. Sparse coding based visual tracking: Review and experimental comparison. *Pattern Recognit.* **2013**, *46*, 1772–1788.
14. Wright, J.; Yang, A.; Ganesh, A.; Sastry, S.; Ma, Y. Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *31*, 210–227.
15. Zhang, L.; Yang, M.; Feng, X. Sparse representation or collaborative representation: Which helps face recognition? *IEEE Int. Conf. Comput. Vis.* **2011**, 471–478.
16. Van Der Merwe, R.; Doucet, A.; De Freitas, N.; Wan, E. The unscented particle filter. *NIPS* **2000**, 584–590.
17. Lee, K.-C.; Ho, J.; Yang, M.-H.; Kriegman, D. Video-based face recognition using probabilistic appearance manifolds. In Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2003), Madison, WI, USA, 16–22 June 2003; pp. 313–320.
18. Lee, K.-C.; Kriegman, D. Online learning of probabilistic appearance manifolds for video-based recognition and tracking. *IEEE Conf. Comput. Vis. Pattern Recognit.* **2005**, *1*, 852–859.
19. Zhang, L.; Li, Y.; Nevatia, R. Global data association for multi-object tracking using network flows. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 24–26 June 2008; pp. 1–8.
20. Milgram, S. The small world problem. *Psychol. Today* **1967**, *2*, 60–67.
21. Han, J.; Bhanu, B. Individual recognition using gait energy image. *IEEE Trans. Pattern Anal. Mach. Intell.* **2006**, *28*, 316–322.
22. Wang, L.; Tan, T.; Ning, H.; Hu, W. Silhouette analysis-based gait recognition for human identification. *IEEE Trans. Pattern Anal. Mach. Intell.* **2003**, *25*, 1505–1518.
23. Zhang, H.; Fritts, J.E.; Goldman, S.A. Image segmentation evaluation: A survey of unsupervised methods. *Comput. Vis. Imag. Underst.* **2008**, *110*, 260–280.
24. Zhan, B.; Monekosso, D.N.; Remagnino, P.; Velastin, S.A.; Xu, L.-Q. Crowd analysis: A survey. *Mach. Vis. Appl.* **2008**, *19*, 345–357.
25. Wright, J.; Ma, Y.; Mairal, J.; Sapiro, G.; Huang, T.S.; Yan, S. Sparse representation for computer vision and pattern recognition. *Proc. IEEE* **2010**, *98*, 1031–1044.
26. Zibulevsky, M.; Elad, M. L1-l2 optimization in signal and image processing. *IEEE Signal Process. Mag.* **2010**, *27* 76–88.
27. Shoham, Y.; Gersho, A. Efficient bit allocation for an arbitrary set of quantizers. *IEEE Trans. Acoust. Speech Signal Proc.* **1988**, *36*, 1445–1453.
28. Yang, A.; Sastry, S.; Ganesh, A.; Ma, Y. *Fast l1-minimization Algorithms and an Application in Robust Face Recognition: A Review*; Technical Report No. UCB/EECS-2010-13; EECS Department University of California: Berkeley, CA, USA, 5 February 2010.
29. Mei, X.; Ling, H. Robust visual tracking and vehicle classification via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 2259–2272.
30. Zhang, S.; Yao, H.; Sun, X.; Liu, S. Robust visual tracking using an effective appearance model based on sparse coding. *ACM Trans. Intell. Syst. Technol.* **2012**, *3*, 43.

31. Zhang, S.; Yao, H.; Zhou, H.; Sun, X.; Liu, S. Robust visual tracking based on online learning sparse representation. *Neurocomput* **2013**, *100*, 31–40.
32. Pentland, A. Fractal-based description of natural scenes. *IEEE Trans. Pattern Anal. Mach. Intell.* **1984**, *6*, 661–674.
33. Kittler, J.; Hatef, M.; Duin, R.; Matas, J. On combining classifiers. *IEEE Trans. PAMI* **1998**, *20*, 226–239.
34. Kuncheva, L. *Combining Pattern Classifiers: Methods and Algorithms*; Wiley-Interscience: New York, NY, USA, 2004.
35. Berclaz, J.; Fleuret, F.; Turetken, E.; Fua, P. Multiple object tracking using k-shortest paths optimization. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 1806–1819.
36. Schindler, K. Continuous energy minimization for multi-target tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*.
37. Tan, D.; Huang, K.; Yu, S.; Tan, T. Efficient night gait recognition based on template matching. In the Proceedings of the 18th International Conference on Pattern Recognition, Hong Kong, China, 20–24 August 2006; pp. 1000–1003.

# Sparse Representation for Infrared Dim Target Detection via a Discriminative Over-Complete Dictionary Learned Online

Zheng-Zhou Li, Jing Chen, Qian Hou, Hong-Xia Fu, Zhen Dai, Gang Jin, Ru-Zhang Li and Chang-Ju Liu

**Abstract:** It is difficult for structural over-complete dictionaries such as the Gabor function and discriminative over-complete dictionary, which are learned offline and classified manually, to represent natural images with the goal of ideal sparseness and to enhance the difference between background clutter and target signals. This paper proposes an infrared dim target detection approach based on sparse representation on a discriminative over-complete dictionary. An adaptive morphological over-complete dictionary is trained and constructed online according to the content of infrared image by K-singular value decomposition (K-SVD) algorithm. Then the adaptive morphological over-complete dictionary is divided automatically into a target over-complete dictionary describing target signals, and a background over-complete dictionary embedding background by the criteria that the atoms in the target over-complete dictionary could be decomposed more sparsely based on a Gaussian over-complete dictionary than the one in the background over-complete dictionary. This discriminative over-complete dictionary can not only capture significant features of background clutter and dim targets better than a structural over-complete dictionary, but also strengthens the sparse feature difference between background and target more efficiently than a discriminative over-complete dictionary learned offline and classified manually. The target and background clutter can be sparsely decomposed over their corresponding over-complete dictionaries, yet couldn't be sparsely decomposed based on their opposite over-complete dictionary, so their residuals after reconstruction by the prescribed number of target and background atoms differ very visibly. Some experiments are included and the results show that this proposed approach could not only improve the sparsity more efficiently, but also enhance the performance of small target detection more effectively.

Reprinted from *Sensors*. Cite as: Li, Z.-Z.; Chen, J.; Hou, Q.; Fu, H.-X.; Dai, Z.; Jin, G.; Li, R.-Z.; Liu, C.-J. Sparse Representation for Infrared Dim Target Detection via a Discriminative Over-Complete Dictionary Learned Online. *Sensors* **2014**, *14*, 9451–9470.

## 1. Introduction

The distance between man-made satellites and ground-based EO sensors is usually more than 30,000 km, and the angle between the object and ground-based EO sensor is so small that the target on a EO sensor is a small blob with only several pixels. Meanwhile, the energy of the object decays greatly for long distance propagation, and it is usually submerged in noise and clutter. This causes great difficulty for infrared dim small target detection and tracking [1,2]. The problem of how to effectively distinguish dim small targets from clutter has been widely studied over the past years, and a number of dim target detection algorithms have been developed and they can approximately be classified into two categories, namely, detection before track (DBT) and track before detection (TBD) [3–5]. Image filtering and content learning are the two basic methods of DBT-based target



detection algorithms. The image filtering-based detection algorithms such as Top-Hat [6], TDLMS [7] and wavelet transform [8,9] usually whiten the image signal, and then determine whether there is a target or not in every scan by the amplitude threshold using some criteria, such as constant false alarm ratio (CFAR). The content learning-based target detection algorithms such as principal component analysis (PCA) compare the similarity of the image and the template pre-learned by knowledge [10]. The TBD-based algorithms jointly process more consecutive scans and declare the presence of a target and its track by searching the candidate trajectory using an exhaustive hypothesis. Temporal cross product (TCP) is presented to extract the characteristics of temporal pixels by using a temporal profile in infrared image sequences, and it could effectively enhance the signal-to-clutter ratio (SCR) [11]. Higher detection probability and lower false alarm probability in every scan could not only facilitate analysis of characteristics, including movement analysis, but also simplify the computational complexity for TBD.

The sparse representation decomposed on an over-complete dictionary is a newly-developed content learning-based target detection algorithm strategy [12–14]. In the over-complete dictionary, there are large number of atoms representing target and even background [15]. Gaussian [16,17] and Gabor [18] are the representative functions used to construct structural over-complete dictionaries offline. It is difficult for these structural over-complete dictionaries to suit the target and background with non-structure shape, so their representation coefficient vectors would be not sparse enough to distinguish targets from background clutter [19]. The adaptive morphological component over-complete dictionary is constructed according to the image content, and it could enhance the sparsity of the representation coefficient vector. However, the atoms representing target and background are mixed together, and the sparsity of the representation coefficient vector is usually too lower to detect target signals [20], therefore, it is necessary to discriminate the atoms representing targets from the ones describing background clutter. The existing techniques to discriminate atoms usually choose background clutter to train a background over-complete dictionary, and manually select target signals to build a target over-complete dictionary. The discriminative over-complete dictionary trained manually could greatly improve the sparsity of the representation coefficient vector and also improve the performance of dim target detection. Nevertheless, its serious limitation is that the atoms couldn't adapt to moving targets and changing backgrounds effectively, so it is necessary to distinguish the atoms automatically for the discriminative over-complete dictionary in order to further enhance the capability of dim target detection.

An infrared dim target detection approach based on sparse representation over discriminative over-complete dictionary learned online is proposed in this paper. An adaptive morphological over-complete dictionary is built according to infrared image content by a K-singular value decomposition (K-SVD) algorithm [21], and then a target over-complete dictionary is discriminated automatically from a background over-complete dictionary by the criteria that the atoms representing dim target signals could be decomposed more sparsely over a Gaussian over-complete dictionary than the one in a background over-complete dictionary. The remainder of the paper is organized as follows: the adaptive over-complete dictionary is trained in Section 2. It is further divided into target over-complete dictionary and background over-complete dictionary in Section 3. The sparsity-driven dim target detection approach is presented in Section 4. Some experiments

are included in Section 5 to evaluate the performance of the discriminative over-complete dictionary and Gaussian over-complete dictionary, and the results show that the target detection performance achievable by the proposed approach is significantly enhanced. Conclusions are drawn in Section 6.

## 2. Adaptive Morphological Component Dictionary

Infrared dim target images consists of target, background and noise, and can be modeled as [5]:

$$\begin{cases} H_1: f = f_t + f_b + n & \text{target present} \\ H_0: f = f_b + n & \text{target absent} \end{cases} \quad (1)$$

where  $f$  is the infrared image,  $f_t$ ,  $f_b$ , and  $n$  are the target, background and noise, respectively. Dim target detection is a two-category classification problem, *i.e.*, pixels are labeled as target (target present) or background (target absent) based on their diverse characteristic differences. The dim target concentrates itself relatively in a small other than pixel-sized object region with uniform amplitude, and could be described by point spread function:

$$f_t(x, y, k) = a(k) \cdot \exp \left\{ -\frac{1}{2} \left[ \left( \frac{x - x_t(k)}{\delta_x(k)} \right)^2 + \left( \frac{y - y_t(k)}{\delta_y(k)} \right)^2 \right] \right\} \quad (2)$$

where,  $a(k)$  is target intensity amplitude;  $(x_t, y_t)$  denotes the target location at instant  $k$ , and  $x_t$  and  $y_t$  represent the horizontal and vertical direction, respectively;  $\delta_x(k)$  and  $\delta_y(k)$  are the extent parameters at horizontal and vertical direction, respectively, and they are usually several pixels for their angle is very small when the distance between the man-made satellite and ground-based EO receiver is 30,000 km. Moreover, the signal-to-noise (SNR) is always also low because the target's energy decays greatly when it propagates in noise and clutter.

The K-singular value decomposition (K-SVD) algorithm is adopted to learn the image content and train adaptive over-complete dictionary  $\mathbf{D}$  from a large number of infrared dim images. The dictionary is trained by the following formula [22]:

$$(\gamma, \mathbf{D}) = \arg \min_{\mathbf{D}, \gamma} \left( \sum \|\gamma\|_0 + \sum \|\mathbf{D}\gamma - f\|_2^2 \right) \quad (3)$$

where  $\|\cdot\|_0$  denotes  $\ell_0$ -norm, which is defined as the number of nonzero entries in the vector, and  $\|\cdot\|_2$  denotes  $\ell_2$ -norm, which is defined as the energy of the vector. The formula means the infrared image  $f$  could be decomposed on the over-complete dictionary  $\mathbf{D}$ , and the representing coefficient vector  $\gamma$  has the least number of nonzero entries under the constraint that the residual  $\|f - \mathbf{D}\gamma\|_2^2$  is the minimum. Obviously, the sparsity of the representation coefficients would be dominated by the over-complete dictionary  $\mathbf{D}$ .

The two terms  $\mathbf{D}$  and  $\gamma$  should be solved simultaneously in Equation (3). The construction of the adaptive over-complete dictionary is an iterative process, and there are two stages in every iteration, namely, sparse coding and dictionary update [23]:

(1) Sparse coding. In this stage, assuming that  $\mathbf{D}$  is fixed; the sparse representation  $\gamma$  is updated by solving the following formula:

$$\arg \min \|\gamma\|_0 \quad s.t. \quad \|f - \mathbf{D}\gamma\|_2^2 \leq \varepsilon \quad (4)$$

It means that the sparsest representation vector  $\gamma$  is searched by an orthogonal matching pursuit (OMP) algorithm under the constraint that the residual  $\|f - \mathbf{D}\gamma\|_2^2$  would be less than the error tolerance  $\varepsilon$  [24,25].

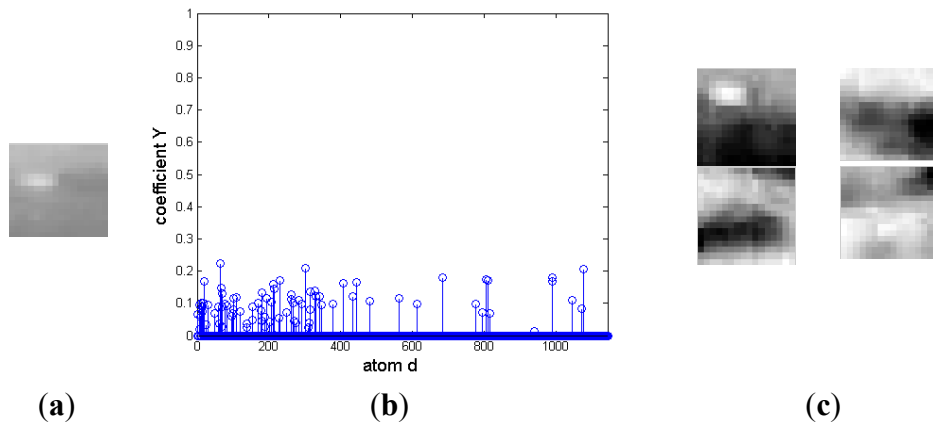
(2) Dictionary  $\mathbf{D}$  update. During this stage, only one atom  $d_k$  in the dictionary  $\mathbf{D}$  is updated at every iteration, and then the residual is estimated by:

$$E_k = \left\| f - \sum_k d_k \gamma_k \right\|_2^2 \quad (5)$$

Therefore, an approximate solution set  $(d_k, \gamma_k)$  would be optimized by the SVD algorithm. Repeating Equation (5), every atom  $d_k$  in the dictionary  $\mathbf{D}$  will be updated until the residual  $\left\| f - \sum_k d_k \gamma_k \right\|_2$  is less than the error tolerance  $\varepsilon$ . The signal representation error decays exponentially with increasing iteration number, and the adaptive dictionary would be constructed after a few iterations. The final version dictionary  $\mathbf{D}$ , which called adaptive morphological over-complete dictionary, would be compatible with the content of infrared image.

In the morphological over-complete dictionary  $\mathbf{D}$ , the atoms representing target and background are mixed together. This could induce two difficulties for the signal is decomposed based on not only the target over-complete dictionary, but also the background over-complete dictionary. One is that the representing coefficients mightn't be sparse. The other is that the representing coefficients are irregular and couldn't easily discriminate target from background. To better illustrate the challenge, an example is shown in Figure 1. There is a target in Figure 1a. Its representation coefficient decomposed on the over-complete dictionary  $\mathbf{D}$  is shown in Figure 1b. The dictionary contains 1,144 atoms. There are many nonzero coefficients on atoms representing target and background, and the four atoms corresponding to the largest four nonzero entries are shown in Figure 1c. The first atom represents the target, and the other three atoms describe the background. Therefore, it is necessary to discriminate the atoms representing the target from the ones describing background clutter.

**Figure 1.** Decomposition of a target image block on adaptive morphology over-complete dictionary. (a) Target image block. (b) Representation coefficient. (c) Four atoms with maximum coefficients.



### 3. Discriminative Over-Complete Dictionary Constructed Online

The sparse representation model assumes that the signal could be reconstructed with the same type of over-complete dictionary and corresponding representation coefficients [26]. For the background signal  $f_b$ , it can be represented by a linear combination of the background atoms:

$$\begin{aligned} f_b &\approx \alpha_1 d_1^b + \alpha_2 d_2^b + \dots + \alpha_{N_b} d_{N_b}^b \\ &= [d_1^b, d_2^b, \dots, d_{N_b}^b] [\alpha_1, \alpha_2, \dots, \alpha_{N_b}]^T = \mathbf{D}_b \boldsymbol{\alpha} \end{aligned} \quad (6)$$

where  $\mathbf{D}_b$  is the background over-complete dictionary,  $N_b$  denotes the account of atoms in the dictionary  $\mathbf{D}_b$ , and  $\boldsymbol{\alpha}$  is the representing coefficient vector whose entries are the abundances of the corresponding atoms in  $\mathbf{D}_b$ , *i.e.*, sparse vector or a vector with only few nonzero entries.

Similarly, for the target signal  $f_t$ , it can also be sparsely represented as a linear combination of the target atoms:

$$\begin{aligned} f_t &\approx \beta_1 d_1^t + \beta_2 d_2^t + \dots + \beta_{N_t} d_{N_t}^t \\ &= [d_1^t, d_2^t, \dots, d_{N_t}^t] [\beta_1, \beta_2, \dots, \beta_{N_t}]^T = \mathbf{D}_t \boldsymbol{\beta} \end{aligned} \quad (7)$$

where  $\mathbf{D}_t$  is the target over-complete dictionary,  $N_t$  is the number of atoms in the target dictionary  $\mathbf{D}_t$  and  $\boldsymbol{\beta}$  is the sparse representing coefficient vector. An infrared image lies in the union of the background dictionary and target dictionary, *i.e.*,

$$f = \mathbf{D}_b \boldsymbol{\alpha} + \mathbf{D}_t \boldsymbol{\beta} = \underbrace{[\mathbf{D}_b \quad \mathbf{D}_t]}_{\mathbf{D}} \underbrace{\begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{bmatrix}}_{\boldsymbol{\gamma}} = \mathbf{D} \boldsymbol{\gamma} \quad (8)$$

where  $\mathbf{D} = [\mathbf{D}_b \quad \mathbf{D}_t]$  is a matrix consisting of both background and target atoms, and  $\boldsymbol{\gamma} = [\boldsymbol{\alpha} \quad \boldsymbol{\beta}]$  is a  $(N_t + N_b)$  – dimensional vector consisting of the two vectors  $\boldsymbol{\alpha}$  and  $\boldsymbol{\beta}$  associated with the two dictionaries.

Since background clutter and the target signal usually consist of different materials, they have distinct signatures and thus lie in different over-complete dictionaries. If an infrared image is a target signal, it ideally can be represented by a target dictionary, but can't be represented by the background atoms. In this case,  $\boldsymbol{\alpha}$  is a zero vector and  $\boldsymbol{\beta}$  is a sparse vector. Therefore, the infrared image can be sparsely represented by the union of background over-complete dictionary and target over-complete dictionary, and the location of nonzero entries in the sparse vector  $\boldsymbol{\gamma}$  actually contains critical information about the class of the infrared image.

The existing techniques to discriminate atoms usually choose background clutter to train a background over-complete dictionary, and manually select target signals to build a target over-complete dictionary offline. Nevertheless, the serious limitation of such an offline discriminative over-complete dictionary is that the atoms couldn't effectively adapt to moving targets and changing backgrounds, which would induce the discrimination between target and clutter to be too small to distinguish targets from background clutter, so it is necessary to distinguish automatically the atoms online for the discriminative over-complete dictionary in order to further enhance the capability of dim target detection.

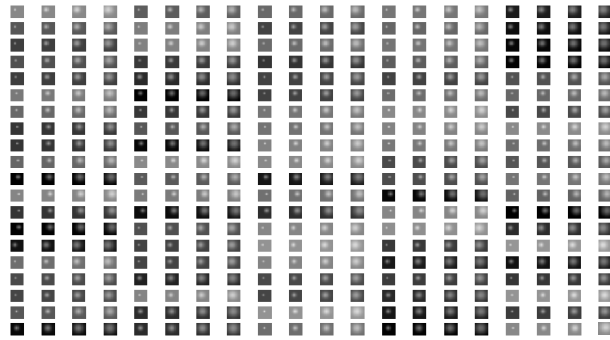
As the Introduction discusses, the distance between a geosynchronous satellite and a ground-based EO receiver is more than 30,000 km, and the angle is so small that the target on the EO sensor is a small blob with only several pixels with Gaussian distribution, so dim small targets usually are described by a two-dimensional Gaussian intensity model (GIM), which is widely used to describe infrared dim small targets:

$$I(i, j) = I_{\max} \exp\left(-\frac{1}{2}\left[\frac{(i-i_0)^2}{\sigma_x^2} + \frac{(j-j_0)^2}{\sigma_y^2}\right]\right) \quad (9)$$

where  $(i_0, j_0)$  is the the center of the blob,  $I(i, j)$  is the intensity at the position  $(i, j)$ ,  $I_{\max}$  is the intensity of the peak,  $\sigma_x^2$  and  $\sigma_y^2$  are the extent parameters at horizontal and vertical directions, respectively.

The extent parameters control the target intensity spread degree, and it represents a small point when they are very little, and denote a flat block when they are very large. In content learning-based target detection algorithms, the correlation between infrared image and GIM is usually measured to decide whether there is dim target or not. In this paper, the GIM-based structural over-complete dictionary is adopted to test the atoms of adaptive morphological component dictionary, and automatically discriminate the atoms representing target from the ones describing background clutter. There are four parameters in GIM, namely, center position  $(i_0, j_0)$ , peak intensity  $I_{\max}$ , horizontal and vertical extent parameters  $\sigma_x^2$  and  $\sigma_y^2$ , and they are adjusted to generate a large number of diverse atoms with different position, brightness and shape. Figure 2 is a part of the atoms of Gaussian over-complete dictionary  $\mathbf{D}_{\text{gaussian}}$ , and each atom has  $7 \times 7$  pixels.

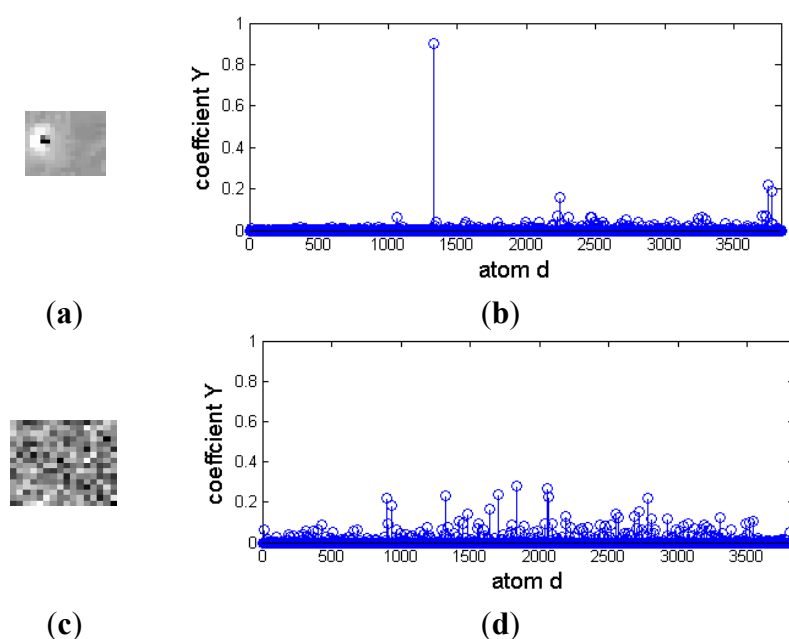
**Figure 2.** Gaussian over-complete dictionary.



Although a dim moving target is always polluted by environmental noise, it approximately affords a two-dimensional Gaussian model. Otherwise, different background clutter has diverse and abundant morphology. For example, cirrocumulus cloud is composed of small spherical clouds, which arrange in rows or groups; stratocumulus cloud is generally larger and looser, and its thickness and shape are also different. Figure 3 is the representation coefficients of target signal and background noise decomposed based on a Gaussian over-complete dictionary. The target signals and background noise are from a deep space image. Figure 3a and 3b are the target image block and its representation coefficients, respectively. The target signal is a bright blob with a black noise at its center, and it is very similar to a two-dimensional Gaussian function. There are only several nonzero representation coefficients, and the target signal could be decomposed sparsely by

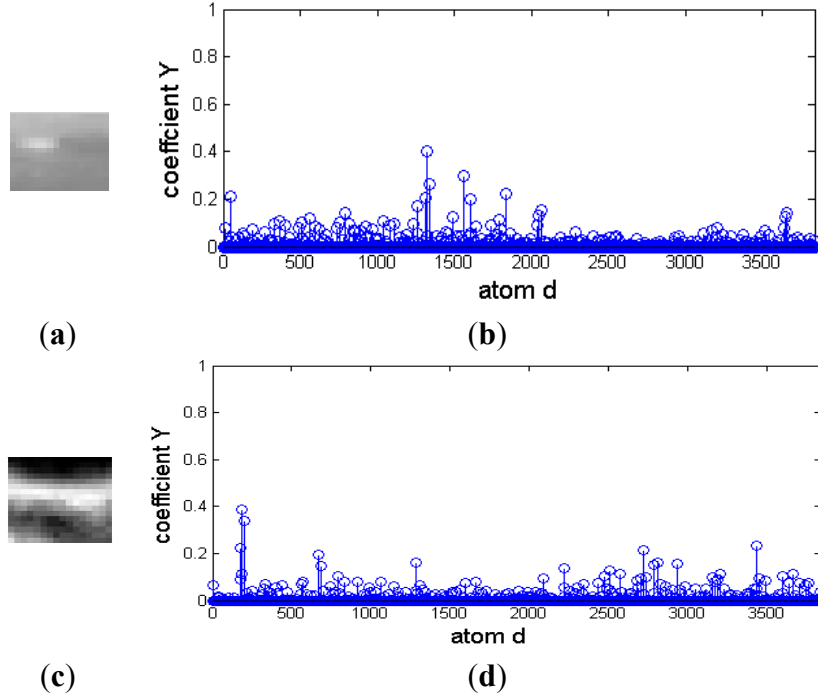
a Gaussian over-complete dictionary. Figure 3c and 3d are background noise and its representation coefficients, respectively. Much of its representation coefficients are nonzero, and the background noise should be reconstructed by many Gaussian atoms.

**Figure 3.** Representation coefficient of a deep space image decomposed based on a Gaussian over-complete dictionary. (a) and (b) are target signal and its sparse coefficient, respectively; (c) and (d) are background noise and its sparse coefficient, respectively.



A small infrared target in a cloud background and their representation coefficients decomposed based on a Gaussian over-complete dictionary are shown in Figure 4. Figure 4a and 4b are the target signal and its representation coefficients, respectively. The target signal is a bright rectangle, and it has more nonzero representation coefficients than that of the target signal similar to the two-dimensional Gaussian function shown in Figure 3. Therefore, a target signal like a bright rectangle could be reconstructed by a few of Gaussian atoms with maximum nonzero representation coefficients. Figure 4c and 4d are the cloud background and its representation coefficients, respectively. It is shown that much of these coefficients are nonzero, and it couldn't be decomposed sparsely. Moreover, the representation coefficient difference between target signal and background is so small that it is hard to distinguish the target from background clutter.

**Figure 4.** Representation coefficient of cloud image on Gaussian over-complete dictionary. (a) and (b) are target signal and its sparse coefficient, respectively. (c) and (d) are cloud background and its sparse coefficient, respectively.



Therefore, compared with the background atom in the adaptive morphological component dictionary, the target atom could be reconstructed from a lesser amount of Gaussian atoms. In other words, with the same number of Gaussian atoms with maximum sparse coefficients, the residual of a background atom would be much greater than that of a target atom. Based on this idea, the atom in the adaptive morphology over-complete dictionary  $\mathbf{D}$  could be classified as target over-complete dictionary  $\mathbf{D}_t$  and background over-complete dictionary  $\mathbf{D}_b$ . The atom  $d_k$  is decomposed on Gaussian over-complete dictionary  $\mathbf{D}_{gaussian}$  as follows:

$$\hat{\mathbf{a}} = \arg \min_{\mathbf{a}} \left\| d_k - \mathbf{D}_{gaussian} \mathbf{a} \right\|_2^2 \quad s.t. \quad \|\mathbf{a}\|_0 = k \quad (10)$$

where  $d_k$  is an atom of  $\mathbf{D}$ . After decomposing for the fixed  $k$  times, the residual  $r(d_k)$  of the atom  $d_k$  is defined as:

$$r(d_k) = \left\| d_k - \mathbf{D}_{gaussian} \hat{\mathbf{a}} \right\|_2 \quad (11)$$

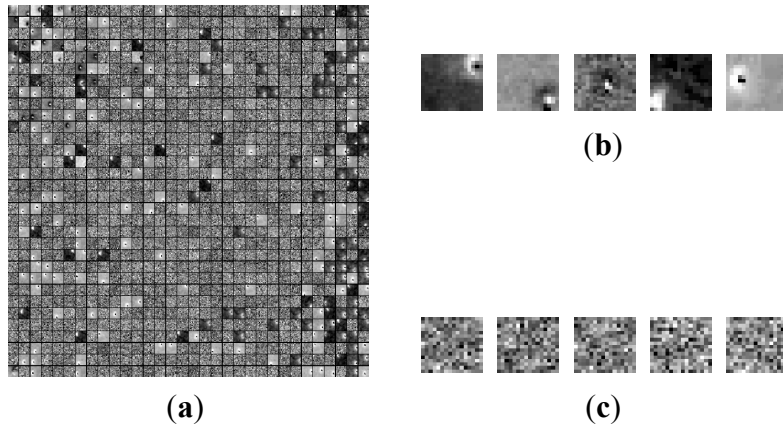
Whether the atom  $d_k$  is target atom or not could be decided by comparing the residual  $r(d_k)$  with a threshold  $\delta$ :

$$\begin{cases} d_k \in \mathbf{D}_t & r(d_k) \leq \delta \\ d_k \in \mathbf{D}_b & r(d_k) > \delta \end{cases} \quad (12)$$

The threshold  $\delta$  usually is proportional to the size of the atom. Every atom of  $\mathbf{D}$  could be identified, then the adaptive background over-complete dictionary  $\mathbf{D}_b$  and target over-complete dictionary  $\mathbf{D}_t$  could be constructed online and automatically. Figure 5 is an example of the

discriminative over-complete dictionary learned online. Figure 5a is a part of adaptive morphological dictionary  $\mathbf{D}$  for space target image. Figure 5b shows five target atoms. The target atoms have abundant shapes, and they can more really reflect the morphological characteristics of the original dim target than Gaussian atom. Five background atoms are listed in Figure 5c, and all of them are noise.

**Figure 5.** Discriminative over-complete dictionary for space image. (a) Dictionary  $\mathbf{D}$ , (b) Target atom of  $\mathbf{D}_t$ , (c) Background atom of  $\mathbf{D}_b$ .



#### 4. Dim Target Detection Criteria

Once the target and background over-complete dictionaries  $\mathbf{D}_t$  and  $\mathbf{D}_b$  are learned and constructed online through these above procedures, the image  $f$  could be decomposed on these two dictionaries with the error tolerance  $\sigma$ , respectively:

$$\hat{\boldsymbol{\beta}} = \arg \min \|\boldsymbol{\beta}\|_0 \quad s.t. \quad \|\mathbf{D}_t \boldsymbol{\beta} - f\|_2 \leq \sigma \quad (13)$$

$$\hat{\boldsymbol{\alpha}} = \arg \min \|\boldsymbol{\alpha}\|_0 \quad s.t. \quad \|\mathbf{D}_b \boldsymbol{\alpha} - f\|_2 \leq \sigma \quad (14)$$

These formulas are approximately solved by greedy pursuit algorithms such as orthogonal matching pursuit (OMP) or subspace pursuit (SP). The target can be sparsely decomposed over the target over-complete dictionary, yet it can't be sparsely decomposed on the background over-complete dictionary. For target signals, the residual reconstructed by target atoms with maximum representation coefficients would be less than that reconstructed by the same number of background atoms with maximum representation coefficients. Let us define the residual reconstructed by target over-complete dictionary as  $r_t(f)$ :

$$r_t(f) = \sqrt{\sum_{i=1}^m \|f - d_i^t \beta_i\|^2} \quad (15)$$

where  $\beta_i$  denotes the recovered sparse coefficient for  $f$  associated with the target over-complete dictionary, and  $m$  is a prescribed constant, such as five. Similarly, background clutter can be sparsely decomposed on the background over-complete dictionary, yet it can't be sparsely decomposed on the target over-complete dictionary. For background clutter, the residual reconstructed by target atoms with maximum representation coefficients would be larger than that



reconstructed by the same number of background atoms with maximum representation coefficients. Let us define the residual reconstructed by background over-complete dictionary as  $r_b(f)$ :

$$r_b(f) = \sqrt{\sum_{i=1}^m \|f - d_i^b a_i\|^2} \quad (16)$$

where  $a_i$  denotes the recovered sparse coefficient for  $f$  associated with the background over-complete dictionary.

The sparsity-based dim target detector can be done by comparing the difference of residuals with a prescribed threshold, *i.e.*,

$$D(f) = r_b(f) - r_t(f) \quad (17)$$

If  $D(f) > \eta$ ,  $f$  would be labeled as target; otherwise, it would be labeled as background clutter.  $\eta$  is a prescribed threshold.

## 5. Experimental Results and Analysis

The following experiments have been implemented in MATLAB language on personal computer with a Pentium dual-core CPU E5900. Figure 6 shows the low contrast infrared images, which are captured outfield by an EO imaging tracking system. Figure 6a is the deep space sequence image, Figure 6b is the cloud sequence image, and Figure 6c is the multi-target image. Noise and cloud are the background clutter of deep space images, multi-target images and cloud images, respectively. The target is the brighter maculous form at the center of rectangle box. Their signal-to-noises (SNRs) are about 2.3 and 3.5 in Figure 6a and 6b, respectively. In Figure 6c, three targets with different scale are marked, and their SNRs are different.

**Figure 6.** Original infrared image. (a) Deep space image. (b) Cloud image. (c) Multi-target image.

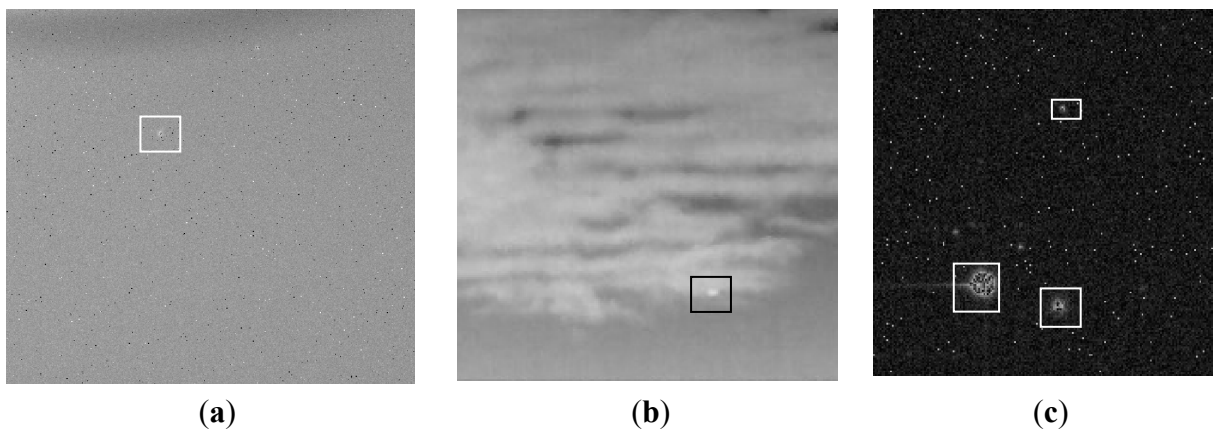
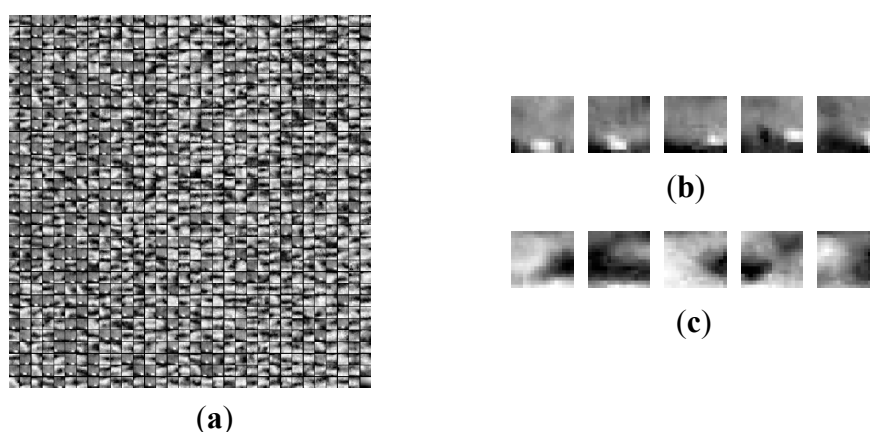


Figure 5, 7 and 8 are the morphological over-complete dictionary for the deep space images, cloud images and multi-target images, respectively. Compared with the Gaussian over-complete dictionary shown in Figure 2, the adaptive morphological over-complete dictionary has more diverse and abundant morphology, and it would be more suitable to represent original images with less atoms. Every atom is  $7 \times 7$  pixels.

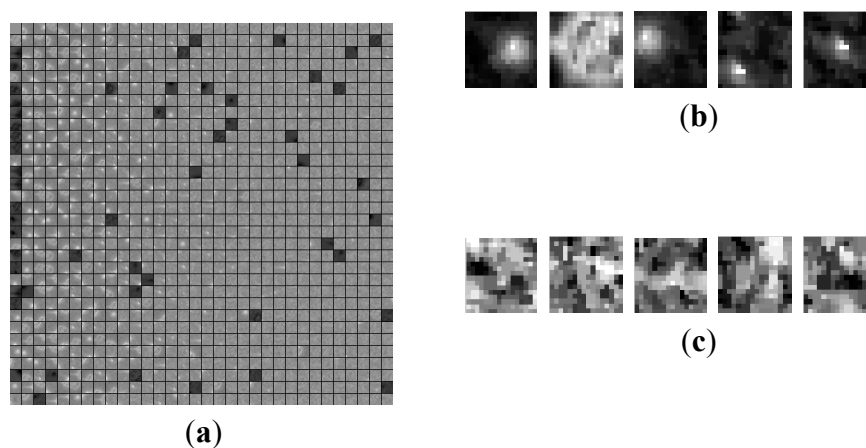
The space image and cloud image are decomposed on a Gabor over-complete dictionary (GD), Gaussian model over-complete dictionary (GMD), and adaptive morphological over-complete dictionary (AMCD), and then they are reconstructed by the five atoms with maximum representation coefficients. The first two are structural dictionaries, and the last one is a non-structural dictionary. The residual energy between the original image and the reconstructed image is introduced to evaluate the capability of sparse representation.

Obviously, the smaller the residual energy is, the more powerful the sparse representation would be. Ten target image blocks and ten background image blocks in space image and cloud image are decomposed and reconstructed, and their residual energy (not normalization) are shown in Figure 9. For the twenty image blocks, the residual energy of AMCD is the minimum, and that of GD is the maximum. This figure indicates that structural dictionary AMCD, which is trained according to image content, could more effectively describe the morphological component than these non-structural dictionaries GD and GMD, and its sparse representation ability is the most powerful.

**Figure 7.** Adaptive over-complete dictionary for cloud image. (a) Dictionary. (b) Target atom. (c) Background atom.

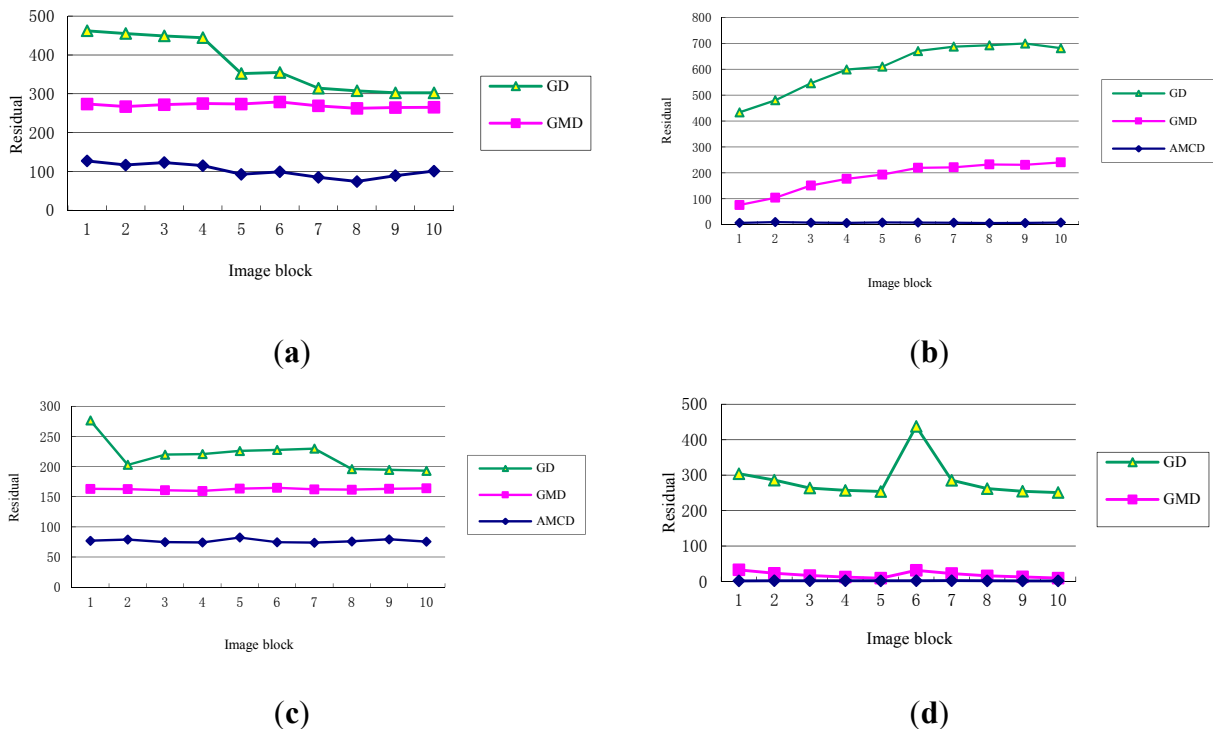


**Figure 8.** Adaptive over-complete dictionary for Multi-target image. (a) Dictionary. (b) Target atom. (c) Background atom.

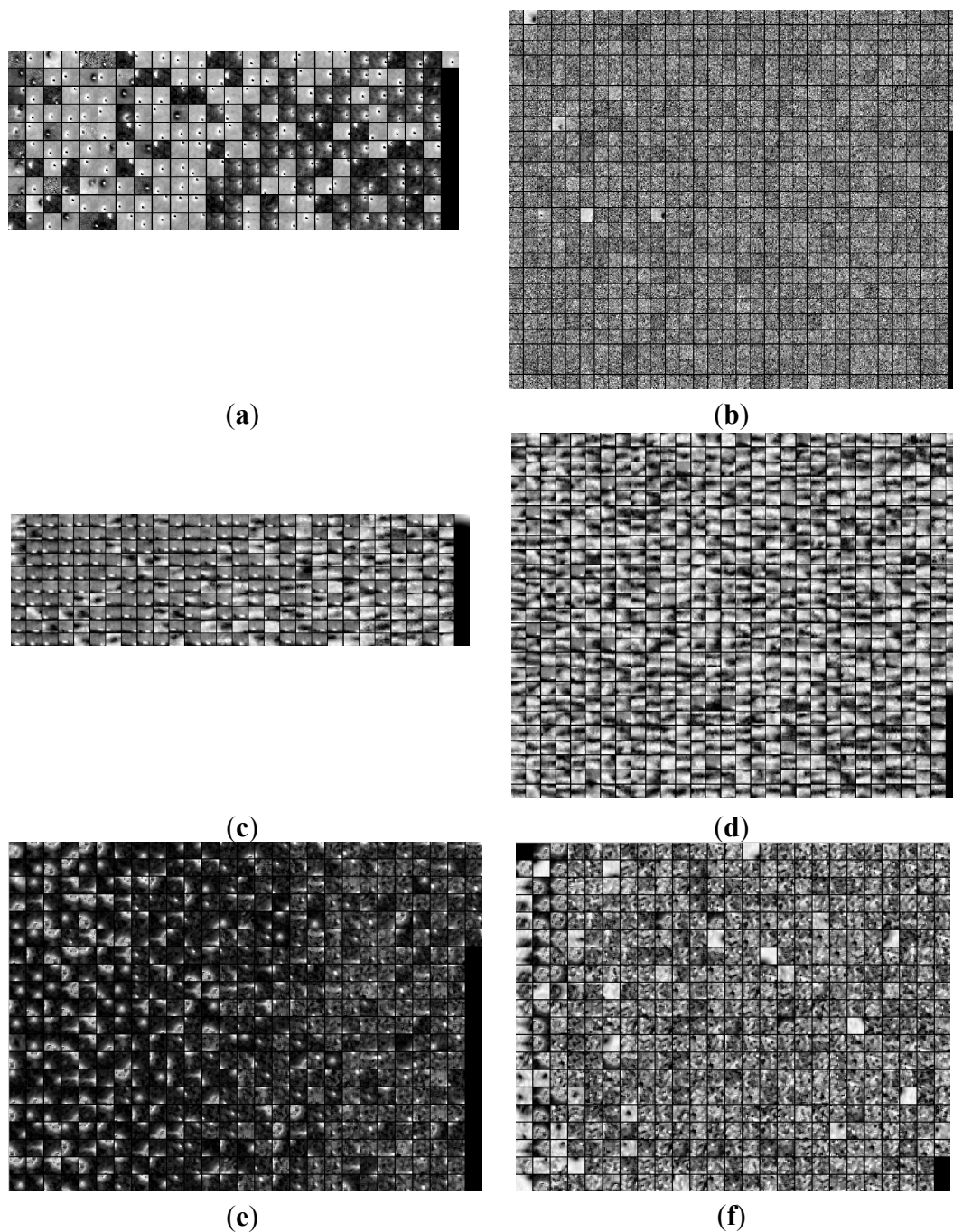


The discriminative over-complete dictionaries for space image, cloud image and multi-target image are shown in Figure 10. Figure 10a and 10b are the target over-complete dictionary and background over-complete dictionary for space image, respectively; Figure 10c and 10d are the target over-complete dictionary and background over-complete dictionary for cloud image, respectively; Figure 10e and 10f are the target over-complete dictionary and background over-complete dictionary for multi-target images, respectively. The threshold  $\delta$  used to distinguish target over-complete dictionary and background over-complete dictionary is equal to three multiplied by size of the atom, and the parameter  $k$  is equal to five. The target atoms are bright points, which are located at various positions of the image blocks with diverse shapes. The background atoms are noise and cloud clutter for space image, cloud image, and multi-target image respectively. There are 250 target atoms and 894 background atoms for deep space image, 280 target atoms and 884 background atoms for cloud image, and 526 target atom and 498 background atoms for multi-target image. For the space image, three background atoms and two target atoms are wrongly identified as target atom and background atoms, respectively; for cloud image, there are 22 background atoms and nine target atoms are wrongly classified as target atom and background atom, respectively; for multi-target image, there are seven background atoms and six target atoms are wrongly classified as target atom and background atom, respectively. The correct probabilities for space image, cloud image and multi-target image are more than 98%, 91%, and 95%, respectively, yet, depending on the complex degree of background clutter, the detection probability by the criteria based on Gaussian over-complete dictionary fluctuates.

**Figure 9.** Residual energy after image reconstruction. (a) and (b) are target blocks in space image and cloud image, respectively. (c) and (d) are background blocks in space image and cloud image, respectively.



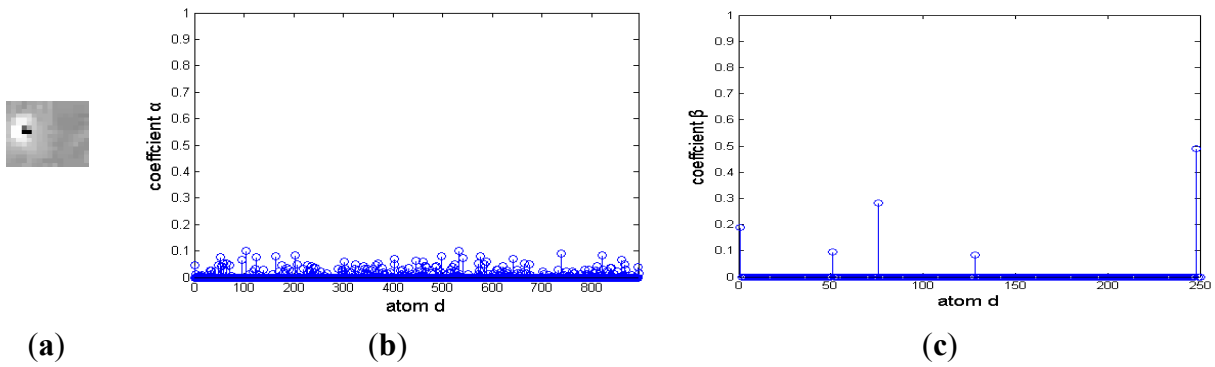
**Figure 10.** Discriminative over-complete dictionary. (a) and (b) are target dictionary and background dictionary for space image, respectively; (c) and (d) are target dictionary and background dictionary for cloud image, respectively; (e) and (f) are target dictionary and background dictionary for multi-target image, respectively.



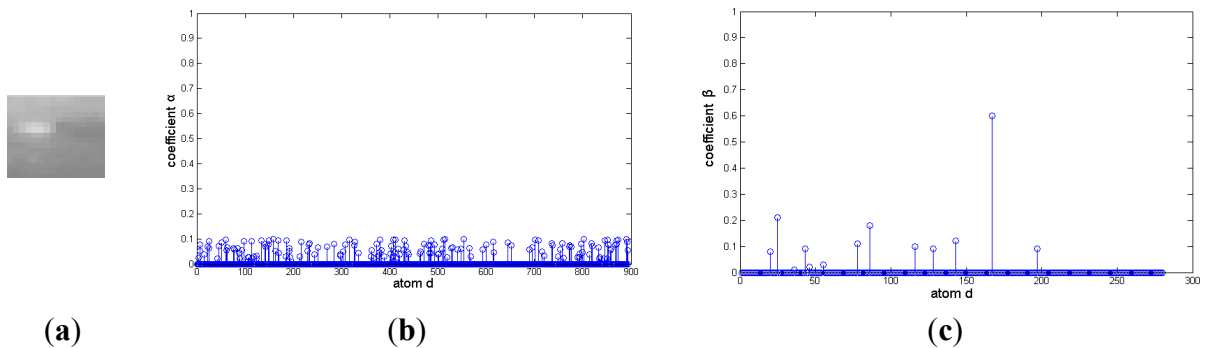
The representation coefficients of the target image blocks decomposed on this discriminative over-complete dictionary are shown in Figures 11–13. In Figures 11–13, Figures 11a–13a are target signals, and Figures 11b–13b and Figures 11c–13c are the coefficients on corresponding background over-complete dictionary and corresponding target dictionary, respectively. There are many nonzero coefficients in Figures 11b–13b of Figures 11, 12 and 13, and this means that the signal couldn't be sparsely represented by background over-complete dictionary. In Figures 11–13,

Figures 11c–13c have less nonzero coefficients than that of Figures 11b–13b, and it indicates the target image blocks could be sparsely represented by target over-complete dictionary. Moreover, the residual reconstructed by background over-complete dictionary is much than that of target over-complete dictionary with the same  $m$  atoms, Here,  $m$  is equal to five. The target signal could be reconstructed by five target atoms with maximum nonzero coefficients in the corresponding target over-complete dictionary, and the residuals of deep space image, cloud image and multi-target image are very small, about 0.13, 0.17 and 0.09 (normalization), respectively. Otherwise, their residuals after reconstructed using five background atoms with maximum nonzero coefficients in corresponding background over-complete dictionary are very big, and they are 0.94, 0.91 and 0.89, respectively. The residuals reconstructed by corresponding target over-complete dictionary  $r_t(f)$  are less than that constructed by corresponding background over-complete dictionary  $r_b(f)$ , and the image would be correctly labeled as target.

**Figure 11.** Representation coefficient of target signal in deep space image decomposed on discriminative over-complete dictionary. (a) Target signal. (b) Sparse coefficient on background dictionary. (c) Sparse coefficient on target dictionary.



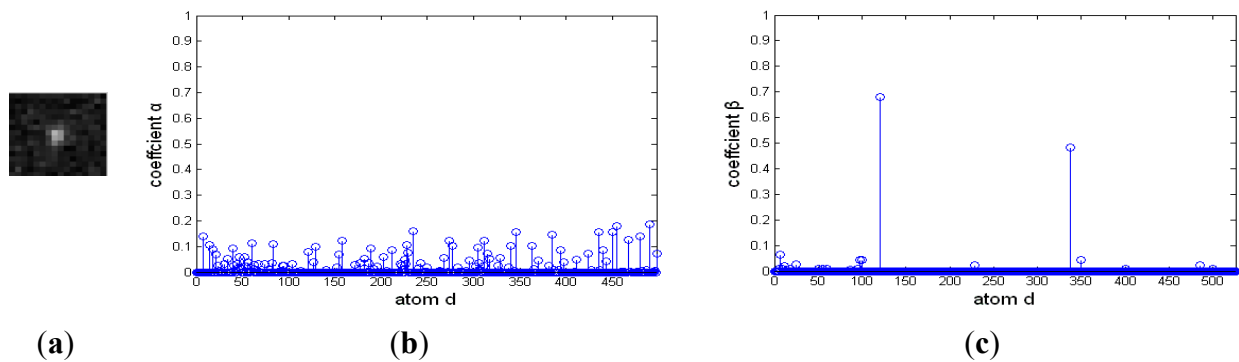
**Figure 12.** Representation coefficient of target signal in cloud image decomposed on discriminative over-complete dictionary. (a) Target signal. (b) Sparse coefficient on background dictionary. (c) Sparse coefficient on target dictionary.



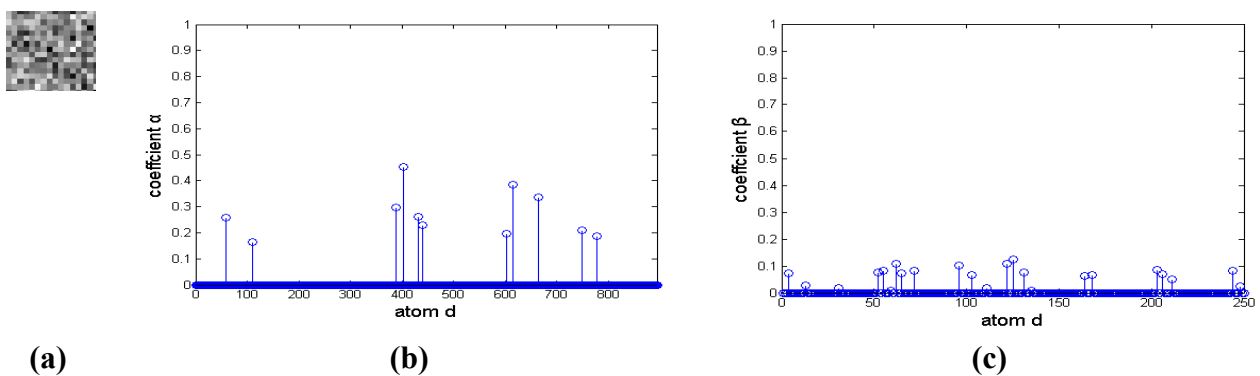
The representation coefficient of the background image blocks decomposed based on this discriminative over-complete dictionary are shown in Figures 14–16. In Figures 14–16, Figures 14a–16a are background noise, and Figures 14b–16b and Figures 14c–16c are the representation coefficients based on the corresponding background over-complete dictionary and

corresponding target over-complete dictionary, respectively. In Figures 14, 15 and 16, There are some nonzero coefficients on atoms in Figures 14b–16b and Figures 14c–16c, and Figure 14b–16b has less nonzero coefficients than that of Figures 14c–16c, and it indicates the background image blocks could be represented based on the corresponding background over-complete dictionary more sparsely than based on the corresponding target over-complete dictionary. Moreover, the residual reconstructed by the background over-complete dictionary is less than that of the target over-complete dictionary with the same atoms. The background image could be reconstructed by five background atoms with maximum nonzero coefficients in the corresponding background over-complete dictionary, and the residuals of deep space image, cloud image and multi-target image are about 0.17, 0.28 and 0.10 (normalization), respectively. The residuals reconstructed using five target atoms with maximum nonzero coefficients in corresponding target over-complete dictionary is 0.64, 0.83 and 0.75, respectively. Their residuals are  $r_t(f) > r_b(f)$ , and the image should be labeled as background.

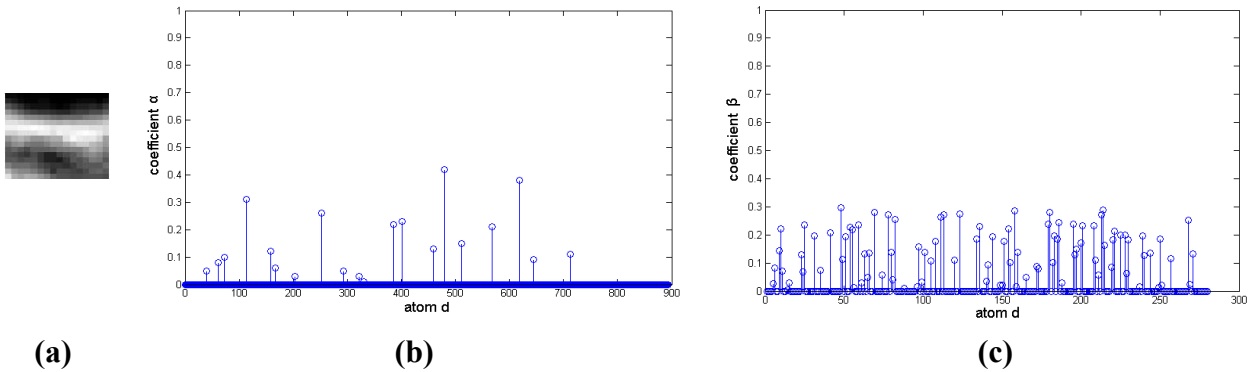
**Figure 13.** Representation coefficient of target signal in multi-target image decomposed on discriminative over-complete dictionary. (a) Target signal. (b) Sparse coefficient on background dictionary. (c) Sparse coefficient on target dictionary.



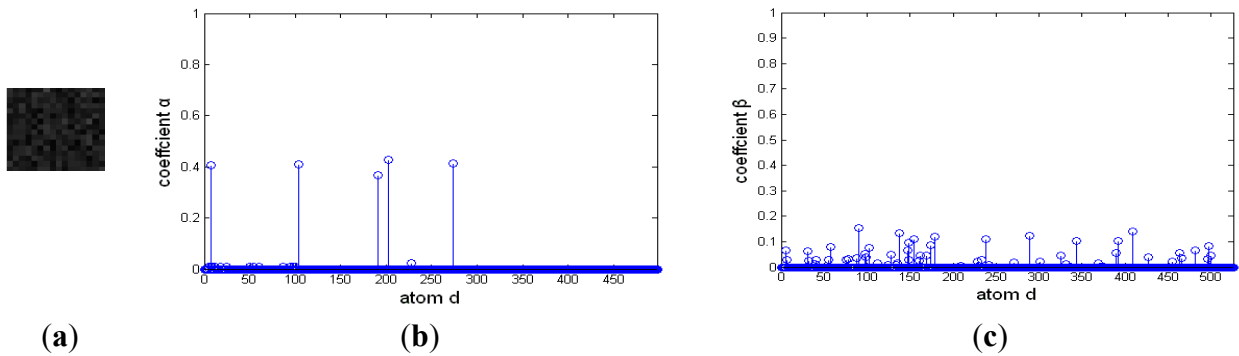
**Figure 14.** Representation coefficient of background block in deep space image decomposed on discriminative over-complete dictionary. (a) Sparse coefficient on background dictionary, (b) Sparse coefficient on target dictionary.



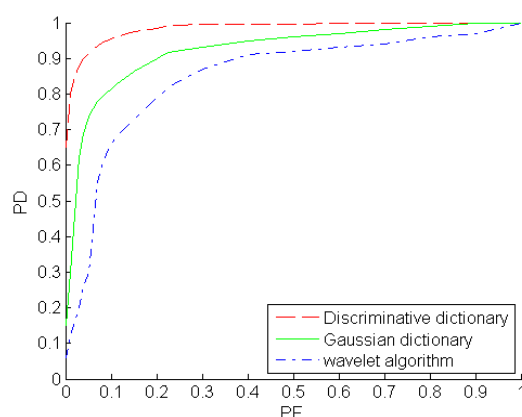
**Figure 15.** Representation coefficient of background block decomposed in cloud image on discriminative over-complete dictionary. (a) Sparse coefficient on background dictionary, (b) Sparse coefficient on target dictionary.



**Figure 16.** Representation coefficient of background block in multi-target image decomposed on discriminative over-complete dictionary. (a) Sparse coefficient on background dictionary, (b) Sparse coefficient on target dictionary.



The receiver operating characteristic (ROC) curves of the Gaussian over-complete dictionary, discriminative over-complete dictionary and wavelet algorithm are shown in Figure 17. The ROC curve describes the probability of detection (PD) as a function of the probability of false alarms (PF). The PF is calculated by the number of false alarms (background pixels determined as target) over the number of background samples, and the PD is the ratio of the number of hits (target pixels determined as target) and the total number of true target samples. The extent parameters  $\sigma_x^2$  and  $\sigma_y^2$  in the two dimensional Gaussian model are extended to some degree, background clutter, even flat background, could be represented sparsely by the Gaussian over-complete dictionary. Hence, PD and PF would be increased with the extent parameters increasing. From the ROC plots, the discriminative over-complete dictionary outperforms the Gaussian over-complete dictionary, *i.e.*, the PD of the former is larger than that of the latter when their PFs are same. Meanwhile, the wavelet algorithm is the worst one among the three algorithms for the example images.

**Figure 17.** ROC curves of target detection.

## 6. Conclusions

This paper proposed an infrared dim target detection approach based on a sparse representation on a discriminative over-complete dictionary. This non-structural over-dictionary adaptively learns the content of infrared images online and is further divided into a target over-complete dictionary and a background over-complete dictionary automatically. The target over-complete dictionary could describe target signals, and the background over-complete dictionary would represent background clutter. This discriminative over-complete dictionary not only can capture significant features of background clutter and dim targets better than a structural over-complete dictionary, but also can efficiently strengthen the sparse feature difference between background clutter and target signals better than a discriminative over-complete dictionary learned offline and classified manually. The experimental results show that this proposed approach could effectively improve the performance of small target detection.

When SNR is lower than one, a target signal would be submerged in the strong noise, and its shape would be polluted and be not represented simply by a Gaussian model. Future work would focus on how to more effectively distinguish the target over-complete dictionary and background over-complete dictionary from adaptive morphological over-complete dictionary for diverse target signals and background clutter even in low SNR. Moreover, the computation time of the discriminative over-complete dictionary to detect target signal is about 6.4 s for a frame image, and this proposed algorithm should be optimized to decrease the computation complexity.

## Acknowledgments

This research was supported by the National Natural Science Foundation of China under Grant No.61071191, Natural Science Foundation of Chongqing under grant No. CSTC 2011BB2048, and Fundamental Research Funds for the Central Universities under Grant No. 106112013CDJZR160007, and China Postdoctoral Science Foundation under Grant No. 2014M550455. And we are also grateful to the reviewers for their suggestion.



## Author Contributions

Zheng-Zhou Li contributed conception, Jing Chen and Qian Hou designed the discriminative over-complete dictionary, Hong-Xia Fu and Zhen Dai designed the image reconstruction experimental work based on over-complete dictionary, Gang Jin designed the target detection experimental work based on discriminative over-complete dictionary, and Ru-Zhang Li and Chang-Ju Liu designed the target detection experimental work based on wavelet and evaluated the target detection performance. The authors jointly prepared the manuscript.

## References

1. Liou R.J.; Azimi-Sadjadi, M.R. Dim target detection using high order correlation method. *IEEE Trans. Aerosp. Electron. Syst.* **1993**, *29*, 841–856.
2. Grossi, E.; Lops, M.; Venturino, L. A Novel Dynamic Programming Algorithm for Track-Before-Detect in Radar Systems. *IEEE Trans. Signal. Process.* **2013**, *61*, 2608–2619.
3. Grossi, E.; Lops, M. Sequential along-track integration for early detection of moving targets. *IEEE Trans. Signal. Process.* **2008**, *56*, 3969–3982.
4. Orlando, D.; Venturino, L.; Lops, M. Track-before-detect strategies for STAP radars. *IEEE Trans. Signal. Process.* **2010**, *58*, 933–938.
5. Li, Z.Z.; Qi, L.; Li, W.Y.; Jin, G.; Wei, M. Track initiation for dim small moving infrared target based on spatial-temporal hypothesis testing. *J. Infrared Millim. Terahertz Waves* **2009**, *30*, 513–525.
6. Bai, X.Z.; Zhou, F.G.; Jin, T. Enhancement of dim small target through modified top-hat transformation under the condition of heavy clutter. *Signal. Process.* **2010**, *90*, 1643–1654.
7. Cao, Y.; Liu, R.M.; Yang, J. Small target detection using two-dimensional least mean square (TDLMS) filter based on neighborhood analysis. *Int. J. Infrared Millim. Waves* **2008**, *29*, 188–200.
8. Wang, T.; Yang, S.Y. Weak and small infrared target automatic detection based on wavelet transform. *Intell. Inform. Technol. Appl.* **2008**, *2008*, 609–701.
9. Davidson, G.; Griffiths, H.D. Wavelet detection scheme for small targets in sea clutter. *Electr. Lett.* **2002**, *38*, 1128–1130.
10. Panagopoulos, S.; Soraghan, J.J. Small-target detection in sea clutter. *IEEE Trans. Geosci. Remote Sens.* **2004**, *42*, 1355–1361.
11. Tae-wulk, B. Small target detection using bilateral filter and temporal cross product in infrared images. *Infrared Phys. Technol.* **2011**, *54*, 403–411.
12. Bai, X.Z.; Zhou, F.G.; Xie, Y.C.; Jin, T. Enhanced detectability of point target using adaptive morphological clutter elimination by importing the properties of the target region. *Signal Process.* **2009**, *89*, 1973–1989.
13. Pillai, J.K.; Patel, V.M.; Chellappa, R. Sparsity inspired selection and recognition of iris images. In Proceedings of the IEEE 3rd International Conference on Biometrics: Theory, Applications, and Systems, Washington, DC, USA, 28–30 September 2009; pp. 1–6.

14. Hang, X.; Wu, F.X. Sparse representation for classification of tumors using gene expression data. *J. Biomed. Biotech.* **2009**, doi:10.1155/2009/403689.
15. Chen, Y.; Nasrabadi, N.M.; Tran, T.D. Sparse representation for target detection in hyperspectral imagery. *IEEE J. Selected Top. Signal. Process.* **2011**, *5*, 629–640.
16. Zhao, J.J.; Tang, Z.Y.; Yang, J.; Liu, E.-Q.; Zhou, Y. Infrared small target detection based on image sparse representation. *J. Infrared Millim. Waves* **2011**, *30*, 156–166.
17. Zheng, C.Y.; Li, H. Small infrared target detection based on harmonic and space matrix decomposition. *Opt. Eng.* **2013**, *52*, 066401.
18. Bi, X.; Chen, X.D.; Zhang, Y.; Liu, B. Image compressed sensing based on wavelet transform in contourlet domain. *Signal Process.* **2011**, *91*, 1085–1092.
19. Chen, J.; Wang, Y.T.; Wu, H.X. A coded aperture compressive imaging array and its visual detection and tracking algorithms for surveillance systems. *Sensors* **2012**, *12*, 14397–14415.
20. Donoho, D.L.; Elad, M.; Temlyakov, V.N. Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Trans. Inform. Theory* **2006**, *52*, 6–18.
21. Aharon, M.; Elad, M.; Bruckstein, A. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. Signal Process.* **2006**, *54*, 4311–4322.
22. Donoho, D.; Huo, X. Uncertainty principles and ideal atomic decomposition. *IEEE Trans. Inform. Theory* **2001**, *47*, 2845–2862.
23. Wright, J.; Yang, A.Y.; Ganesh, A.; Sastry, S.S.; Ma, Y. Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *31*, 210–227.
24. Elad, M.; Aharon, M. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. Image Process.* **2006**, *15*, 3736–3745.
25. Tropp, J.A.; Gilbert, A.C. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans. Inform. Theory* **2007**, *53*, 4655–4666.
26. Dai, W.; Milenkovic, O. Subspace pursuit for compressive sensing signal reconstruction. *IEEE Trans. Inform. Theory* **2009**, *55*, 2230–2249.

# Relevance-Based Template Matching for Tracking Targets in FLIR Imagery

Gianluca Paravati and Stefano Esposito

**Abstract:** One of the main challenges in automatic target tracking applications is represented by the need to maintain a low computational footprint, especially when dealing with real-time scenarios and the limited resources of embedded environments. In this context, significant results can be obtained by using forward-looking infrared sensors capable of providing distinctive features for targets of interest. In fact, due to their nature, forward-looking infrared (FLIR) images lend themselves to being used with extremely small footprint techniques based on the extraction of target intensity profiles. This work proposes a method for increasing the computational efficiency of template-based target tracking algorithms. In particular, the speed of the algorithm is improved by using a dynamic threshold that narrows the number of computations, thus reducing both execution time and resources usage. The proposed approach has been tested on several datasets, and it has been compared to several target tracking techniques. Gathered results, both in terms of theoretical analysis and experimental data, showed that the proposed approach is able to achieve the same robustness of reference algorithms by reducing the number of operations needed and the processing time.

Reprinted from *Sensors*. Cite as: Paravati, G.; Esposito, S. Relevance-Based Template Matching for Tracking Targets in FLIR Imagery. *Sensors* **2014**, *14*, 14106–14130.

## 1. Introduction

Detection and tracking of objects and people represent an important research topic in computer vision. The ever-increasing need for automatic, fast and reliable solutions for extracting information from video flows through image processing techniques are dictated by the large domain of vision-based applications. In fact, nowadays, a growing number of applications are envisioned to analyze the motion of pedestrians or moving objects in several scenarios, such as driver assistance (e.g., warning drivers about obstacles on the road, helping with the piloting of aircraft), surveillance and human activity recognition (e.g., locating pinpointing sources of ignition during firefighting operations, the control of servo-motor cameras in security areas), *etc.* From this point of view, image processing techniques are useful, among others, for tracking both vehicles (e.g., in automatic traffic monitoring tools) and people (e.g., for the detection of potentially dangerous situations).

In general, the use of color cameras working in the visible domain represents the most investigated and widespread solution in the above-mentioned applications. However, visible light sensors suffer from illumination issues that make this solution not capable of correctly accomplishing its tasks in all-day and all-weather conditions [1]. The advances in infrared sensor technology laid the ground for introducing the use of forward-looking infrared (FLIR) cameras to overcome some limitations due to the use of traditional color sensors. In fact, thermal-infrared images present a key advantage with respect to images produced by sensing visible light. Since the intensity values are determined by the temperature and radiated heat of objects in the field of view, lighting conditions and object properties, such as material color or texture features, do not influence the generation of the image.

However, often, the visible light and the infrared spectrums have been used together in sensor fusion approaches to exploit the benefits of both domains, at the cost of a generally increased system and computational complexity. Moreover, additional challenges are posed when dealing with real-time applications and limited-resource environments. For example, mobile platforms can be equipped with on-board sensors to perform autonomous navigation and tasks [2]; in this scenario, a target-following task is usually implemented by enabling the corresponding actuators after the target detection and tracking phases. Data coming from on-board sensors can be processed and analyzed both locally and remotely [3]. It is worth noticing that, in the case of remotely-processed video flows, the real-time requirements are not only affected by the computational load of the image processing procedure; indeed, the overall latency might be affected by the performance of the communication network. However, proper network technologies exist that guarantee real-time delivery over packet-switched networks. For example, it has been shown that pipeline forwarding technology can offer deterministic delivery over wireless [4], wired [5] or even all-optical [6] networks.

This paper deals with target tracking in forward-looking infrared image sequences; in this context, the thermal imprint of a target is a distinctive feature with respect to background and clutter. Generally, target tracking applications are based on three consecutive steps: first, the detection of stationary or moving objects of interest; then, the tracking of these objects frame by frame; lastly, the classification of the target motion through the analysis of the objects' tracks to recognize their behavior. In this paper, the focus is on the second phase. In particular, the key contribution of this paper is the design of a novel strategy for improving the computational speed of template-matching-based algorithms, the computational domain of which is reduced by selecting a subset of pixels to be analyzed according to a relevance-based strategy.

The designed technique has been compared both to algorithms based on template-matching steps and several other traditional algorithms in target tracking applications. With the aim of assessing the devised technique on different working conditions, experimental tests have been carried out on FLIR image sequences from various datasets; for this reason, both object (vehicle) and pedestrian tracking scenarios have been considered. Results have been gathered both in terms of computational speed and precision. The results obtained by the proposed algorithm indicate an improvement in computational speed by maintaining precision comparable to that achievable by reference algorithms based on traditional template-matching implementation.

The remainder of this paper is organized as follows. Section 2 provides a review of the main target tracking techniques in FLIR imagery. Section 3 focuses on the aspects pertaining to reference algorithms used as the basis for the current work, whereas the proposed solution is presented in Section 4. A detailed theoretical analysis evaluating the use of resources and an evaluation of the tracking performance are illustrated in Section 5. Finally, conclusions are drawn and future research directions are sketched in Section 6.

## 2. Background

Visual tracking is an important topic in computer vision due to the ever-growing number of applications and systems that benefit from its integration, such as traffic monitoring [7] and video

surveillance [8]. In spite of many efforts, some challenges remain to be faced in order to build accurate and reliable tracking systems, such as dealing with occlusions, the alternating appearance of objects, illumination issues, and so on. Different techniques have been proposed to cope with different situations. Basically, target tracking can be realized by using traditional color camera sensors, infrared cameras and exploiting data-fusion techniques. Visual tracking with color cameras has been widely investigated. Among the most recent works in this field, studies of interest encompass detection by classification techniques to deal with adaptability for occlusion, appearance and illumination changes [9]. New schemes to account for appearance variation are considered in [10]. Recently, sparse representation has been widely applied to visual tracking as a solution to illumination changes and occlusions [11–14]. However, these techniques are not able to deal with sequences characterized by poor illumination.

Among the numerous algorithms for target tracking, only a limited amount of them is specifically designed to address the particular issues of target tracking in FLIR imagery. Target tracking in infrared images presents several issues. First of all, they present a low signal-to-noise ratio and are affected by the sensors' ego-motion [15,16]. Several techniques have been devised to cope with such a scenario [17–19]. Traditionally, target tracking has been based on two phases: a target detection (TD) step and a target tracking phase based on spatio-temporal correlation, like the mean-shift algorithms [20–22]. Target detection should be realized using very fast techniques, such as the intensity variation function (IVF) [17]. However, some conditions in FLIR images can invalidate the traditional mean-shift algorithms, for instance the assumptions that they are based on might not be true in the case of sensor ego-motion. Moreover, the presence of similar target signatures or noise represent conditions that lead the TD to fail in FLIR images, returning wrong results; hence the necessity of a strategy for the activation of a recovery phase [23].

To cope with a low signal-to-noise ratio and sensor ego-motion, target correlation approaches have been explored. Among target correlation-based techniques, the one presented in [17] has the peculiarity of using a small and compact target signature for fast frame-to-frame target tracking through IVF, using a larger template only to recover from IVF failures through a template matching (TM) technique. The higher reliability of TM is in terms of resources usage, which is much more intense in the TM phase with respect to the TD phase. In other words, TM is slower than TD, even though it is able to generate more precise results.

The reference algorithms selected for this work, presented in [17,18], are based on the TM technique. Although, while in [17], the IVF failure detection was based on a Cartesian distance metric, in [18], a motion prediction-based metric is presented, and it showed better tracking performances than the Cartesian metric. The techniques presented in [17,18] are analyzed more in detail in Section 3, since they constitute the base layer of the proposed algorithmic improvement.

### 3. Reference Algorithms

The target tracking procedure followed in this work puts down its roots in the techniques proposed in [17,18], in the following, referred to as ATT (automatic target tracking) and PATT (predictive

automatic target tracking), respectively. Both of them have been employed as a reference for evaluating the performance of the proposed solution in Section 5.

Both ATT and PATT use a target detection (TD) phase and a possible target recovery phase; the TD phase is based on the IVF algorithm, and the eventual recovery phase is based on a template matching algorithm. TM is triggered when false alarms are detected during the TD phase. In particular, the detection of IVF false alarms is performed through two different strategies. In [17], a Cartesian distance metric approach is used, while in [18], a motion prediction-based metric and a probabilistic evaluation is introduced.

In the following paragraphs, the main concepts concerning the TD and recovery phases are reviewed. These concepts will be recalled in Section 4 during the exposition of the proposed algorithm.

### 3.1. IVF-Based Target Detection

The detection of targets is based on the analysis of their thermal signature. This phase exploits a local maximum window extracted from the previous frame to compute IVF and uses IVF results to find a new local maximum representing the candidate target position for the current frame. Computations are limited to a sub-frame to avoid non-target objects from the background being identified as potential targets by the algorithm. This simplification clearly assumes that the target motion among frames is confined within the sub-frame. IVF is defined as follows.

$$F^n(v, z) = \frac{1}{\Lambda} \sum_{j=0}^l \sum_{i=0}^k [S^n(i + v, j + z) - \omega^{n-1}] \quad (1)$$

In Equation (1),  $\Lambda = k \times l$  is the area of the target window,  $v$  and  $z$  are the coordinates in the sub-frame,  $\omega^{n-1}$  is the local maximum matrix in the previous frame ( $n - 1$ ),  $S^n$  is the target window centered at  $(i + v, j + z)$  and  $F^n(v, z)$  is the IVF computed in  $(v, z)$  for the current frame  $n$ . A correlation output plane (COP) is built starting from IVF, and it is defined as follows.

$$C(v, z) = e^{-\lambda F^n(v, z)} \quad (2)$$

In Equation (2),  $\lambda$  is an arbitrary parameter, selected to ensure a satisfactory enhancement of IVF results. The position of the candidate target is associated with the position of the highest peak on the correlation output plane. In fact, the maximum on the COP is by definition the point in the sub-frame most similar to the local maximum in the previous frame; for this reason, it is considered the best candidate to represent the target in the current frame.

**Figure 1.** Processing of the intensity variation function (IVF) algorithm in a sample frame from the OTCBVS (Object Tracking and Classification Beyond the Visible Spectrum) dataset [24] (sequence otcbvs 03-11s2ir-4). (a) Frame 57; (b) correlation output plane (COP) of Frame 57.

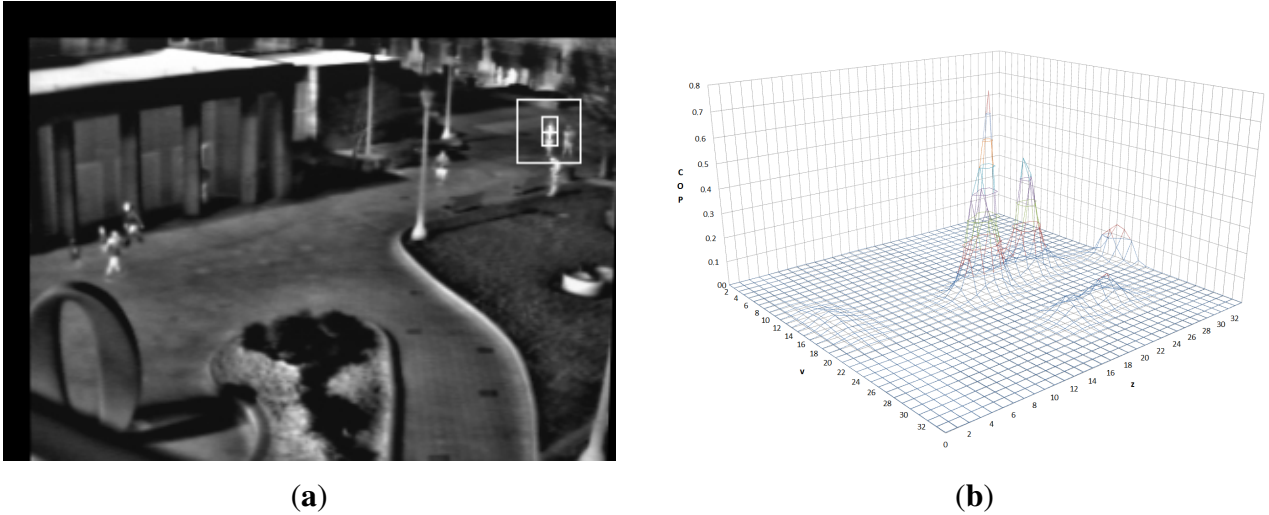


Figure 1 shows an example of the correlation output plane generated by IVF computed on a sample frame extracted from the OTCBVS (Object Tracking and Classification Beyond the Visible Spectrum) dataset [24]. As shown in this example, more peaks with a different local maximum can coexist in the COP. Despite the similarity between local maximum matrices in consecutive frames, it is not guaranteed that the highest peak in the plane (the one selected by IVF) is the real target. In this case, the activation strategy later described determines the need for the execution of a recovery phase. In general, IVF is a fast and reliable algorithm when the sequence is not affected by severe sensor ego-motion and target feature changes are not too swift. Nonetheless, it may be misled by a non-target object included in the sub-frame whose features are similar to those of the target. Moreover when ego-motion is dramatic, the chances of gathering correct results from the IVF algorithm alone are quite low because, due to its small target window and to the low signal-to-noise ratio of the images, a significant sensor ego-motion is very likely to be introduced into the sub-frame of an object with a higher IVF value than the target. Finally, changes in frame features may result in target feature changes, such that IVF is not able to determine the correct position of the target in the current frame. To solve these issues, the detection strategy described in the following section is used to decide when to launch a TM phase able to recover from this type of error.

### 3.2. Cartesian Distance Metric

Despite the good results and proven efficiency of IVF, the low signal-to-noise ratio and occasional sensor ego-motion can lead to wrong matches; a strategy to detect and correct false alarms is therefore mandatory in target tracking for FLIR images. In [17], an approach based on Cartesian distance was used. The algorithm evaluates the distance between the IVF candidate target ( $p_{IVF}^n$ ) and the previous position of the target  $p^{n-1}$ . Whenever the distance exceeds a threshold  $\beta$ , the value of which

depends on the sequence features (e.g., the sensor's ego-motion), the TM recovery phase is activated. The distance is computed as follows:

$$d_{IVF} = \sqrt{(x_{p_{IVF}^n} - x_{p^{n-1}})^2 + (y_{p_{IVF}^n} - y_{p^{n-1}})^2} \quad (3)$$

Even though this approach is rather simple and efficient, because little overhead is added for the activation strategy, it might not be effective enough. Indeed, an optimal  $\beta$  value is difficult to determine, because it hugely depends on sequence features, such as the motion of the target or the ego-motion of the sensor itself.

### 3.3. Motion Prediction-Based Metric

The strategy proposed in [18] for the activation of the recovery step is based on the target history. Information on the target position in previous frames is stored to generate a motion vector and to elaborate a prediction for the current frame using a position estimator. The candidate target position  $\bar{p}_{IVF}^n$ , computed by IVF, is associated with a motion vector  $(p^{n-1}, \bar{p}_{IVF}^n)$ , and it is compared to the predicted motion vector  $(p^{n-1}, \hat{p}^n)$ , where  $\hat{p}^n$  is the target position in the current frame estimated by a linear predictor. The reliability of the IVF result is then evaluated using a conditioned probability approach based on the distance and angle of the motion vectors. In particular, the probability that the result of the target detection phase is the correct target in the current frame is computed as follows:

$$P(\bar{p}_{IVF}^n) = P(d_{IVF} \cap \alpha_{IVF}) = P(d_{IVF}) \times P(\alpha_{IVF}|d_{IVF}) \quad (4)$$

where  $d_{IVF}$  is the IVF motion vector length and  $\alpha_{IVF}$  is the angle it describes with the predicted motion vector. In Equation (4),  $P(d_{IVF})$  and  $P(\alpha_{IVF}|d_{IVF})$  are defined as follows:

$$P(d_{IVF}) = \begin{cases} 1 - \frac{\hat{d} - d_{IVF}}{\hat{d}} & \text{if } d_{IVF} < \hat{d}, \\ 1 - \frac{d_{IVF} - \hat{d}}{d_{max} - \hat{d}} & \text{if } \hat{d} < d_{IVF} < d_{max} \\ 1 & \text{if } d_{IVF} = \hat{d} \\ 0 & \text{if } d_{IVF} = d_{max} \end{cases} \quad (5)$$

$$P(\alpha_{IVF}|d_{IVF}) = \begin{cases} \frac{|\alpha_{IVF} - 180^\circ|}{180^\circ} \times \left(1 - \frac{d_{max} - d_{IVF}}{d_{max}}\right) & \text{if } \alpha_{IVF} \neq 0, d_{IVF} > 0 \\ 1 & \text{otherwise} \end{cases} \quad (6)$$

Both in Equations (5) and (6),  $d_{max}$  is the maximum distance at which the target can be found given a certain sub-frame size. In Equation (5),  $\hat{d}$  is the predicted motion vector length; moreover, the probability is defined, so that it is highest if the length of the IVF motion vector is the same as the one of the predicted motion vector; on the other hand, it decreases as the difference between the motion vectors increases, until it is set to zero when distance  $d_{max}$  is reached. In Equation (6), the angular contribution is computed modulo  $180^\circ$ , and it is weighted to minimize its impact on short motion vectors. This probability value is then compared to a confidence level  $\mu$  to decide whether the IVF position should be selected as the new target position (when  $P(\bar{p}_{IVF}^n) > \mu$ ) or if a recovery algorithm should be invoked to recover from an error condition (when  $P(\bar{p}_{IVF}^n) \leq \mu$ ).



### 3.4. Template Matching

Whenever an error condition is detected through the aforementioned metrics (Sections 3.2 and 3.3), a TM phase is necessary to recover from the error in the TD phase and to find the correct target position on the current frame. The algorithm used in [17,18] is very similar to the described IVF, and it is defined as follows.

$$T^n(v, z) = \frac{1}{\Phi} \sum_{j=0}^p \sum_{i=0}^m [S^n(i + v, j + z) - W^{n-1}] \quad (7)$$

In Equation (7),  $T^n(v, z)$  is the computed TM value for the point of coordinates  $(v, z)$ ,  $\Phi = (p \times m)$  is the area of the target window, while  $W^{n-1}$  is the target window in the previous frame. All other symbols have the same meaning as in Equation (1). With reference to Equations (1) and (7), usually  $p > l$  and  $m > k$ , so  $\Phi > \Lambda$ ; hence the greater resource demand of TM over IVF. As in IVF, the value resulting for each point of the sub-frame is used to build a COP as follows.

$$C_{TM}(v, z) = e^{-\lambda T^n(v, z)} \quad (8)$$

As in the case of IVF,  $\lambda$  is an arbitrary parameter. The highest peak on the COP is taken as the best candidate for the target position on the current frame. Even though the use of  $W$  instead of the smaller  $\omega$  matrix increases the computational complexity of TM with respect to IVF, it also guarantees the use of information on target shape and surrounding background, allowing one to better discriminate between target and non-target objects. In this way, TM can recognize the target, even when IVF fails. Indeed, TM has a better capability of finding the target at the cost of a considerably higher computational time. This is due to the bigger size of the matrices used in TM computation with respect to the size of the matrices used in IVF computation. In [18], the position  $\bar{p}_{TM}^n$  obtained by the TM phase is then used to build a new motion vector  $(p^{n-1}, \bar{p}_{TM}^n)$ , which is, in turn, compared to the predicted motion vector  $(p^{n-1}, \hat{p}^n)$ , as described in Section 3.3. The probability  $P(\bar{p}_{TM}^n)$  is then compared to the probability  $P(\bar{p}_{IVF}^n)$  previously computed. The point with the highest probability in this comparison is finally selected as the position of the target in the current frame. This step is necessary to avoid drifting issues that can be introduced by TM [18].

## 4. Proposed Algorithm

The computational complexity of the template matching step could undermine the applicability in real-time systems of algorithms using the approach so far described. Indeed, the TM step, as described in Section 3.4, necessarily considers a larger domain of computations leading to long execution times. Therefore in this section, an approach based on the relevance of the sampling points in the sub-frame is proposed with the aim of reducing the computational complexity of the TM step.

The proposed solution for tracking targets in FLIR imagery combines the algorithms presented in Section 3.4 and improves the computational speed of the template matching step. The overall tracking algorithm is presented in Algorithm 1. For each new frame of a sequence, the IVF algorithm computes a candidate target; based on the history of the locations of the target under analysis, an expected target position based on a predictive step is computed, and a probability score is thus associated with the candidate target of the IVF step. Since the IVF algorithm has a small footprint,

if the result coming from this step satisfies a minimum confidence level, the location of the candidate target is considered reliable, and it is designated as the current position of the target. Otherwise, additional steps are required to solve the ambiguity. In this case, the template matching procedure involving the analysis of the sub-frame should be activated to gather more accurate results. The correlation output plane is thus analyzed to select an adaptive and suitable threshold used to restrict the computational domain of the TM step. The selected subset of the correlation output plane identifies the areas where the correlation between the searched target and the pixel areas of the current frame is strongest. Within only the selected subset, a score is computed for each possible candidate point, as well as the associated TM value and probability value; as will be explained in more detail later in this section, the score is based both on the TM value and on the likelihood associated with the prediction step for the candidate point under analysis. The higher the threshold, the greater the computational savings. However, it is imperative to avoid too restrictive results; this is the reason why the threshold is designed to be also dynamic: it starts from a high value and decreases as needed to accommodate adequate results (*i.e.*, the scores should satisfy minimum requirements in terms of quality). The three results within the selected subset maximizing the designed score, the TM value and the probability value are finally evaluated with a weighted comparison to choose the new position of the target. Deeper details are given in the remainder of this section. Given the considerable number of symbols cited in the text, for a quick reference, the interested reader can find a digest of them in Table 1.

The traditional TM phase (described in Section 3.4) computes the TM values for each point of the sub-frame, regardless of the likelihood of that point belonging to the target area. The proposed algorithmic improvement takes into account the relevance, *i.e.*, the likelihood of belonging to the target, of a point with coordinates  $(v, z)$  in the sub-frame before computing  $T^n(v, z)$  with Equation (7). The relevance of a point is evaluated by comparing the value associated with the point on the COP computed by IVF in the TD phase, using Equation (2), with a threshold  $\delta$ . The threshold is designed to be adaptive on a frame-by-frame basis, and it is dependent on the maximum value on the same COP. A point within the sub-frame is labeled as relevant ( $\hat{p}^n$ ) if its value on the COP is above the  $\delta$  threshold. Therefore, the TM function described in Equation (7) is computed only for relevant points of the COP; a detailed discussion about the savings in computational complexity is provided in Section 5.3.

---

**Algorithm 1** The proposed algorithm (see Table 1 for symbols meaning). Details on weighted comparisons are given in Equations (10)–(13).

---

**Input:**  $\omega^{n-1}, p^{n-1}, history$

**Output:**  $p^n$

```

1:  $COP \leftarrow ComputeCOP(p^{n-1}, \omega^{n-1})$ 
2:  $\bar{p}_{IVF}^n \leftarrow position\ of\ max\{COP\}$ 
3:  $\hat{p}^n \leftarrow LinearPrediction(history)$ 
4:  $P_{IVF} \leftarrow P(\bar{p}_{IVF}^n)$ 
5: if  $P_{IVF} > confidence\ level$  then
6:    $p^n \leftarrow \bar{p}_{IVF}^n$ 
7: else
8:    $\delta \leftarrow 0.95$ 
9:   repeat
10:    for all  $p : C^n(p) \geq \delta \times max\{COP\}$  do
11:      if  $maxscore < \psi(p)$  then
12:         $maxscore \leftarrow \psi(p)$ 
13:      end if
14:      if  $maxTM < T^n(p)$  then
15:         $maxTM \leftarrow T^n(p)$ 
16:         $\dot{p}_{TM}^n \leftarrow p$ 
17:      end if
18:      if  $maxP < P(p) \vee maxPScore < \psi(p)$  then
19:         $maxP \leftarrow P(p)$ 
20:         $\dot{p}_P^n \leftarrow p$ 
21:      end if
22:    end for
23:    if  $WeightedComparison\_P\_TM(\dot{p}_P^n, \dot{p}_{TM}^n)$  then
24:       $p^n \leftarrow \dot{p}_P^n$ 
25:    else
26:      if  $WeightedComparison\_IVF\_TM(\bar{p}_{IVF}^n, \dot{p}_{TM}^n)$  then
27:         $p^n \leftarrow \bar{p}_{IVF}^n$ 
28:      else
29:         $p^n \leftarrow \dot{p}_{TM}^n$ 
30:      end if
31:    end if
32:     $\delta \leftarrow \delta - 0.2$ 
33:  until  $maxscore \geq \epsilon \vee \delta \leq 0$ 
34: end if

```

---

**Table 1.** List of symbols.

Symbol	Significance
$n$	current frame number
$m$	number of activations of TM phase in a sequence
$p^{n-1}$	position of the target at the previous frame
$\hat{p}^n$	predicted location of the target of interest
$\dot{p}^n$	point of coordinates (v,z) belonging to the sub-frame marked as relevant
$\dot{p}_P^n$	point with maximum probability value
$\dot{p}_{TM}^n$	point with maximum template matching value
$\bar{p}_{IVF}^n$	point with maximum intensity variation function value
$T^n(p)$	template matching value for a point $p$ ; see Equation (7)
$P(p)$	probability value for a point $p$ ; see Equation (4)
$\psi(p)$	score associated to the point $p$ ; see Equation (9)
$\delta$	adaptive threshold for restricting the computational domain of the TM step
$\epsilon$	minimum score $\psi(p)$ to be reached by at least one relevant point
$\alpha$	weight for the TM value
$\beta$	weight for the probability value
$\Phi$	target window area in the computation of $T^n(p)$
$C^n(p)$	correlation output plane value for a point $p$ ; see Equation (2)
$\Lambda$	target window area during the target detection phase (Section 3.1)
$S_{\{P_e\}}$	size of the domain of evaluated points

For each relevant point  $\dot{p}^n$ , a motion vector  $(p^{n-1}, \dot{p}^n)$  is computed in order to be compared with the predicted motion vector  $(p^{n-1}, \hat{p}^n)$  following Equation (4). The likelihood value  $P(\dot{p}^n)$  resulting from the comparison of motion vectors is used along the value resulting from the computation of the template matching value  $T^n(\dot{p}^n)$  for the same point with the aim of defining a score as follows:

$$\psi(\dot{p}^n) = P(\dot{p}^n) \times T^n(\dot{p}^n) \quad (9)$$

where  $\psi(\dot{p}^n)$  is the score associated with a relevant point  $\dot{p}^n$ . To ensure that a significant number of relevant points is found for a specific frame, *i.e.*, the subset is not too small, a minimum score  $\epsilon$  is required. If no point in the current subset reaches the required minimum score  $\epsilon$ , the threshold  $\delta$  is lowered, so that other relevant points can be added to the subset. Once the set of points is considered large enough, *i.e.*, when at least one of the points in the subset reaches the required minimum score  $\epsilon$ , the algorithm should decide which of these points represents the target in the current frame. The point with the highest probability  $\dot{p}_P^n$  and the point with the highest TM value  $\dot{p}_{TM}^n$  are subjected to a weighted comparison to decide whether to choose  $\dot{p}_P^n$  or not. The Boolean function performing the decision is represented in the following:

$$a \times (bc + \bar{b}d + e) \quad (10)$$

In Equation (10), a weight  $\alpha$  is used for the TM value, and a weight  $\beta$  is used for the probability one. In particular,

$$\begin{aligned}
a &= T^n(\dot{p}_P^n) + \alpha > T^n(\dot{p}_{TM}^n) \\
b &= P(\dot{p}_P^n) < 0.9 \\
c &= P(\dot{p}_P^n) - \beta > P(\dot{p}_{TM}^n) \\
d &= P(\dot{p}_P^n) > P(\dot{p}_{TM}^n) \\
e &= P(\dot{p}_{TM}^n) + \eta < P(\bar{p}_{IVF}^n)
\end{aligned} \tag{11}$$

When the Boolean function (10) returns a true value,  $\dot{p}_P^n$  is selected as the position for the target in the current frame; otherwise  $\dot{p}_{TM}^n$  is compared to the IVF candidate position  $\bar{p}_{IVF}^n$ . Experimental results in Section 5 have been gathered using  $\eta$  equal to  $\alpha$ ; moreover, the required level of confidence in the probability metric was set to 0.9.

Likewise, the following Boolean function has been designed to choose between  $\dot{p}_{TM}^n$  and  $\bar{p}_{IVF}^n$ :

$$f \times (g + h) \times i \tag{12}$$

where:

$$\begin{aligned}
f &= \psi(\bar{p}_{IVF}^n) > 0 \\
g &= \psi(\bar{p}_{IVF}^n) + \iota > \psi(\dot{p}_{TM}^n) \\
h &= T^n(\dot{p}_{TM}^n) - T^n(\bar{p}_{IVF}^n) < 0.15 \\
i &= P(\bar{p}_{IVF}^n) + \xi > P(\dot{p}_{TM}^n)
\end{aligned} \tag{13}$$

In Equation (12),  $\xi$  is numerically equivalent to  $\iota$  in the performed tests. In this case, when the Boolean function (12) returns a true value,  $\bar{p}_{IVF}^n$  is selected as the position of the target in the current frame; otherwise  $\dot{p}_{TM}^n$  is selected. Weights in the above equations are assigned so that the correlation is a main criterion for the selection of the target position; on the other hand, also the motion prediction-based metric is taken into account to make sure that the best possible choice is made.

Moreover, the candidate points obtained from the IVF step are always reconsidered against the TM preferred points as in [18]; this is to avoid the TM being subject to drifting and losing the target, as happens when only TM routines are used, without an IVF-based target detection phase. Overall, the logic represented in Equations (10) and (12) showed a satisfactory robustness at the cost of using some thresholds. These levels of confidence depend on sequence features and on the desired precision of the algorithm.

## 5. Results and Discussion

The performance of the proposed algorithm in terms of tracking speed has been evaluated, both from a theoretical and experimental point of view. In fact, the primary objective of the relevance-based algorithm is to enhance the computational efficiency by discarding useless computations in areas with a low probability of finding a target. A preliminary theoretical analysis of the improvements introduced by adopting the devised technique has been carried out with respect to

the reference algorithms ATT [17] and PATT [18], on which this work is based. For this purpose, a set of metrics has been designed in Section 5.1 to enable a fair comparison between these algorithms. Moreover, the analysis of the tracking speed has been widened by taking into account several alternative and faster algorithms according to a recent benchmark on online tracking [25].

### 5.1. Assessment Criteria

With the aim of evaluating the computational efficiency of the devised technique, this section introduces the metrics designed to make a comparison with the reference tracking methods [17,18].

As previously described, all the considered algorithms share the computation of the template matching Function (7), which is activated when tracking error conditions are met. In reference algorithms, the template matching function  $T^n(P_e)$  is computed for each evaluation point  $P_e(v, z)$  with coordinates  $(v, z)$  lying inside the sub-frame. For each evaluation point  $\Phi$  subtractions,  $\Phi - 1$  additions, one division and one exponentiation are required, where  $\Phi$  is the target window area, as defined in Section 3.4. The complexity of the implementation is therefore  $O(\Phi)$  for reference algorithms.

The devised technique proposes to execute the  $T^n(P_e)$  function on a subset of points of the sub-frame. The operations executed on each relevant point are:  $\Phi$  subtractions,  $\Phi - 1$  additions, one division, one multiplication and one computation of probability. Since the algorithm implementing the probability computation has a complexity  $O(1)$ , the overall complexity of the proposed implementation is also  $O(\Phi)$ ; from these considerations, it follows that the size  $S_{\{P_e\}}$  of the domain of evaluated points  $P_e$  can be assumed as a valid comparison metric to evaluate the complexity savings of the proposed algorithm with respect to the reference algorithms.

In the reference algorithms, the size of the domain of evaluated points directly depends on the number of template matching activations  $m$ , and it is proportional to the size of the target window during the target detection phase:

$$S_{\{P_e\}} = \Lambda \times m \quad (14)$$

On the other hand, the size of the domain of evaluated points with the relevance-based algorithm is defined as follows:

$$S_{\{P_e\}} = |\{p : C^n(p) \geq \delta\}| \quad (15)$$

The comparison of the domain size of the template matching function is useful for giving an idea of the boost in performance, as discussed in Section 5.3. However, since it relates to only a portion of the overall tracking algorithm, it is necessary to introduce also another metric able to take into account the most relevant parts of the target tracking algorithms. With this intent, it is worth defining the number of operations required by the different algorithms, including IVF execution from the first to the last frame of a sequence. Since sum and subtraction operations are dominant in both the reference and proposed implementations, the designed metric is based on the number of these operations. IVF requires a number of operations dependent on the size of its target window  $\Lambda$ ; in particular, it requires  $\Lambda$  subtractions and  $\Lambda - 1$  additions. Similarly, the TM phase requires  $\Phi$  subtractions and  $\Phi - 1$  additions; thus, the number of operations of this kind performed by reference algorithms is computed as follows:

$$\Theta(n, m) = n \times [\Psi \times (2\Lambda - 1)] + m \times [\Psi \times (2\Phi - 1)] \quad (16)$$

where  $n$  is the number of frames in the sequence,  $m$  identifies the number of activations of the TM phase in the sequence,  $\Psi$  is the sub-frame area and  $\Lambda$  is the IVF target window area. The number of operations for the reference algorithms can be computed with the same Equation (16), because it is not dependent on the activation strategy, and the same TD and TM phases are used by both algorithms.

Similarly, the number of operations performed by the proposed algorithm is computed as follows:

$$\Omega(n, p) = n \times [\Psi \times (2\Lambda - 1)] + p \times (2\Phi - 1) \quad (17)$$

In Equation (17),  $p$  is the number of relevant points found and analyzed throughout the sequence, whereas the other symbols have the same meaning as Equation (16). It is worth noticing that, in both formulas, a double contribution is considered: the first product takes into account IVF operations, whereas the second product is related to the TM phase.

### 5.2. Analysis of Computational Complexity

The proposed algorithm, in the following referred to as RATT (relevance-based ATT), and the reference ones have been tested on a set of FLIR sequences to measure the designed metrics and to perform an analysis of their computational complexity. Sequences from various public datasets have been considered to take into account different target shapes, background scenarios and sensor and image characteristics (such as resolution). The considered datasets include the OTCBVS 03/OSU (Ohio State University) Color and Thermal Database [24], the Army Missile Command (AMCOM) FLIR dataset and the AIC (Adaptive Information Cluster) Thermal/Visible Nighttime Database [1,26]. The first and the latter concern the tracking of pedestrians, whereas the second one represents a database of military sequences.

**Figure 2.** Excerpts from the considered dataset. (a) OTCBVS (Object Tracking and Classification Beyond the Visible Spectrum) sequence; (b) Army Missile Command (AMCOM) sequence; (c) AIC (Adaptive Information Cluster) Thermal Database sequence.

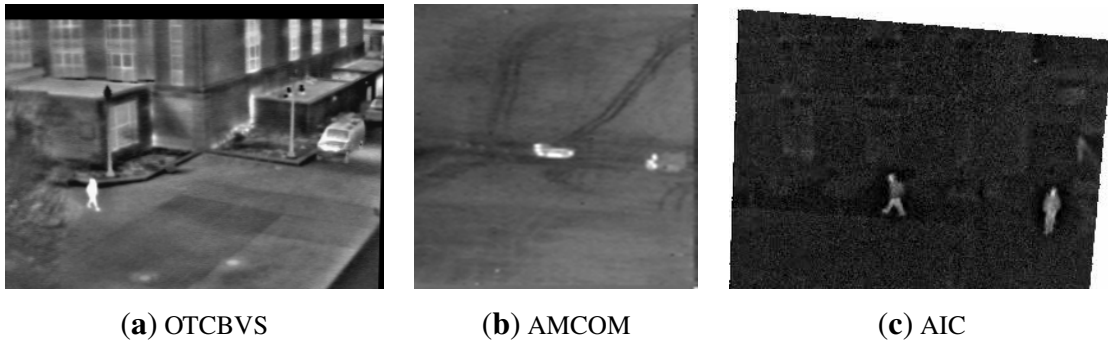


Figure 2 shows some excerpts taken from the aforementioned datasets. Sequences from the OTCBVS 03/OSU Color and Thermal Database have a resolution equal to  $320 \times 240$  pixels.

the AMCOM sequences are provided at a resolution equal to  $120 \times 120$  pixels, and the AIC Thermal/Visible Nighttime frames have a resolution of  $640 \times 480$  pixels. In all of the considered datasets, test sequences have been extracted in such a way that target losses do not occur by using reference algorithms. In particular, twelve sequences have been extracted, both from the OTCBVS and AMCOM datasets, and two sequences have been considered for the AIC database. For each sequence, one or more particular subset of frames has been identified to isolate the appearance and disappearance of targets; in fact, the coordinates and size of the target have been provided to the algorithms for the first frame of each sequence.

Both the reference and the proposed algorithms have been executed on the datasets using the same common parameters. In particular, all of the three algorithms share the same sub-frame size, target window size and the initial target position. The target window size is different for each sequence, since it depends on the shape of the target itself; on the other hand, the sub-frame has been kept constant with a size equal to  $33 \times 33$  pixels, as indicated in [17] for the first two datasets (OTCBVS and AMCOM). A slightly wider sub-frame ( $44 \times 44$  pixels) has been used with the AIC dataset for enabling a complete encapsulation of the target window inside the sub-frame. Instead, concerning only the comparison with the PATT algorithm, the same level of confidence  $\mu$  has been used to trigger the activation of the TM phase.

Table 2 reports the results for all of the above-mentioned sequences concerning the number of activations of the TM phase and the number of evaluated points. The first three columns provide the identifier of the sequence (Seq.), its length  $L$  (expressed in frames) and the size of the target window  $S_{TW}$ , respectively. The fourth and fifth column show the number of activations  $m$  of the TM phase and the size of the domain of evaluated points  $S_{\{P_e\}}$  for the ATT algorithm. Similarly, the next four columns provide the same information for the PATT algorithm and the proposed one, respectively. Finally, the last two columns give an indication of the behavior of the proposed algorithm with respect to ATT (second to last column) and PATT (last column), in terms of the variation of the size of the domain of evaluated points. More specifically, they indicate the percentage of points for which the template matching function  $T^n(P_e)$  is evaluated by using Equation (7) for the proposed algorithm with respect to the ones evaluated by the reference techniques.  $S_{\{P_e\}}$  is computed by using Equations (14) and (15).

The theoretical analysis shows that, in general, it is possible to hugely reduce the size of the function domain despite a higher number of TM activations that are triggered by the algorithm. In fact, in most cases, the number of evaluated points is a small percentage of the size of the original domains. For example, let us consider sequence `otcbvs 03-12s6ir-3`: a very high number of activations occur using the proposed algorithm (template matching is triggered 200-times, about two thirds of the length of the sequence); on the other hand, ATT requires only eight activations, and PATT requires 124 activations. Nevertheless, the proposed technique really evaluates the template matching function on 15.86% of the points with respect to ATT and only on 1.02% of the points with respect to PATT.



**Table 2.** Comparison of the number of activations  $m$  of the TM phase and the number of evaluated points  $S_{\{P_e\}}$  among the proposed algorithm and the reference ones, ATT (automatic target tracking) [17] and PATT (predictive automatic target tracking) [18]. O, OTCBVS dataset; A, AMCOM dataset; AI, AIC dataset; RATT, relevance-based ATT.

Seq.	Dataset		ATT [17]		PATT [18]		RATT		$\Delta$ ATT [17]	$\Delta$ PATT [18]
	$L$	$S_{TW}$	$m$	$S_{\{P_e\}}$	$m$	$S_{\{P_e\}}$	$m$	$S_{\{P_e\}}$	$S_{\{P_e\}}\%$	$S_{\{P_e\}}\%$
O 03-11s1ir-1	154	$16 \times 30$	41	44,649	23	25,047	60	846	1.89%	3.38%
O 03-11s1ir-2	637	$7 \times 15$	4	4,356	4	4,356	F	F	-	-
O 03-11s2ir-1	557	$10 \times 22$	27	29,403	16	17,424	69	648	2.20%	3.72%
O 03-11s2ir-2	339	$10 \times 20$	112	121,968	92	100,188	105	635	0.52%	0.63%
O 03-11s2ir-3	96	$9 \times 18$	11	11,979	4	4,356	45	1,548	12.92%	35.54%
O 03-11s2ir-4	24	$12 \times 30$	0	0	0	0	8	18	>100%	>100%
O 03-11s3ir-1	84	$11 \times 24$	31	33,759	29	31,581	37	238	0.70%	0.75%
O 03-11s3ir-2	787	$8 \times 15$	33	35,937	0	0	83	373	1.04%	>100%
O 03-11s3ir-3	448	$10 \times 24$	4	4,356	40	43,560	6	26	0.60%	0.06%
O 03-12s4ir-1	270	$11 \times 30$	45	49,005	61	66,429	67	2,080	4.24%	3.13%
O 03-12s6ir-1	323	$15 \times 28$	8	8,712	124	135,036	200	1,382	15.86%	1.02%
A 14-15-mantruck	281	$10 \times 10$	6	6,534	3	3,267	19	96	1.47%	2.94%
A 16-08-m60	290	$13 \times 7$	60	65,340	18	19,602	32	2,788	4.27%	14.22%
A 16-08-apc	80	$14 \times 8$	3	3,267	3	3,267	5	11	0.34%	0.34%
A 16-18-apc	300	$11 \times 8$	15	16,335	1	1,089	29	161	0.99%	14.78%
A 16-18-m60	103	$14 \times 8$	2	2,178	0	0	1	1	0.05%	>100%
A 17-02-mantruck	221	$9 \times 9$	2	2,178	16	17,424	39	1,511	69.38%	8.67%
A 17-02-bradley	185	$9 \times 9$	5	5,445	9	9,801	6	132	2.42%	1.35%
A 18-13-m60	227	$9 \times 9$	25	27,225	8	8,712	34	550	2.02%	6.31%
A 18-16-m60	162	$12 \times 12$	79	86,031	5	5,445	17	150	0.17%	2.75%
A 19-06-apc	208	$10 \times 10$	6	6,534	7	7,623	41	367	5.62%	4.81%
A 21-17-apc	360	$12 \times 12$	0	0	1	1,089	51	229	>100%	21.03%
AI ir11-1	263	$21 \times 40$	93	180,048	110	212,960	147	8,789	4.88%	4.13%
AI ir11-2	155	$25 \times 37$	101	195,536	109	211,024	103	2,993	1.53%	1.42%

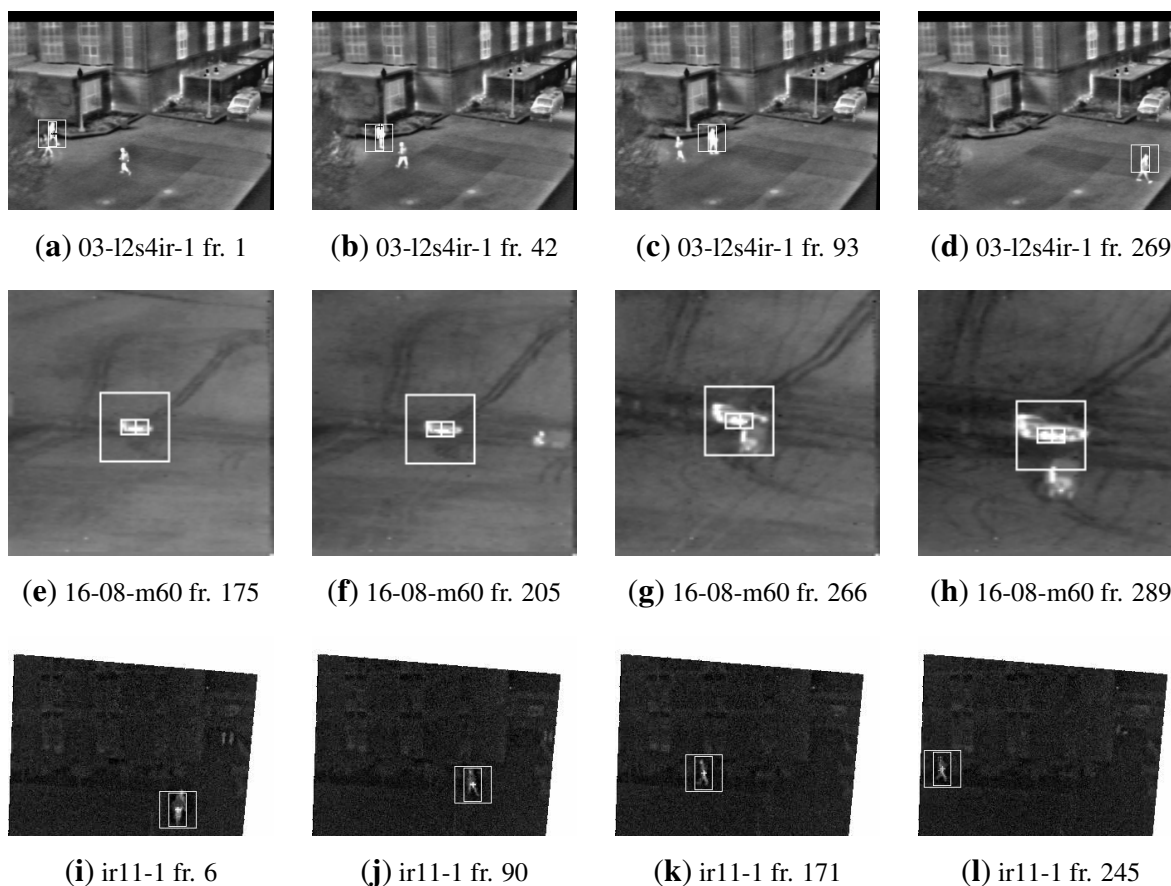
The different number of observed activations among algorithms is due to the different results given by the respective template matching processes. In this way, the history of target positions slightly changes and different probabilities are computed, which, in turn, are used to determine the activation of the TM phase. In some cases, reference algorithms never trigger any recovery phase.

Figure 3 visually shows the tracking results by running the proposed algorithm on a sequence for each dataset. Since the behavior of ATT and PATT from the point of view of the tracked position is analogous to RATT, their frames are omitted. The OTCBVS dataset is represented by the frames extracted from the sequence 03-12s4ir-1 in Figure 3a–d, where pedestrians are tracked throughout the sequence. Vehicles are considered in the AMCOM dataset; an excerpt of these tests is provided by sequence 16-08-m60 in Figure 3e–h. Finally, Figure 3i–l concerns again pedestrian tracking (sequence ir11-1). The smallest rectangle represents the bounding box (*i.e.*, the target window) of the target of interest; the widest one represents the search area (*i.e.*, the sub-frame).

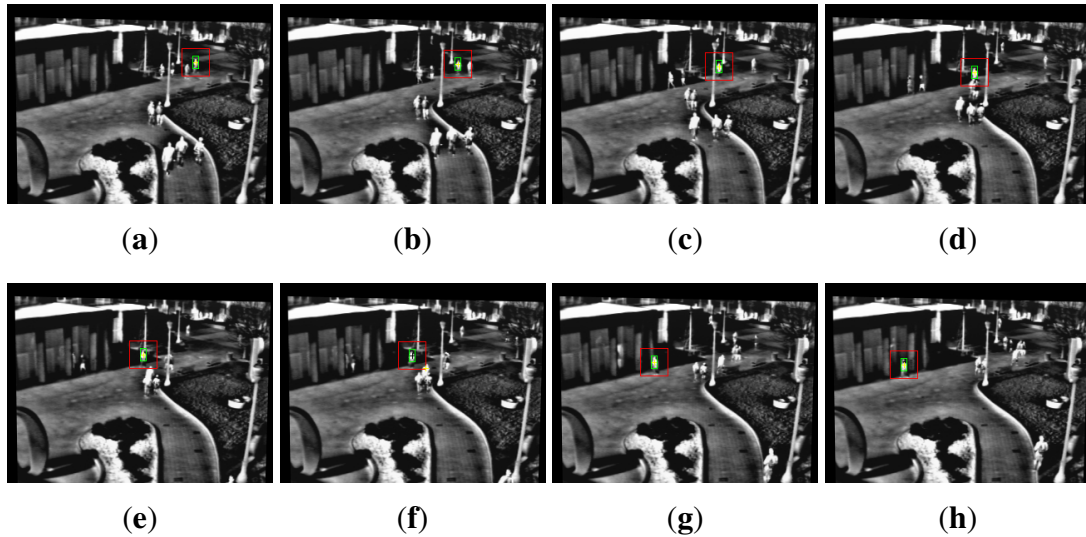
It is worth pointing out that, in some cases, the proposed computational savings can come at the expense of the tracking robustness. As anticipated before, sequences have been selected in such a way that target losses do not occur with the reference algorithms; in Table 2,  $F$  indicates a failure in the tracking algorithm. Considering the extended dataset, the proposed algorithm gets into a target loss for the sequence otcbvs 03-11s1ir-12; more in detail, in this case, only 64% of the sequence

has been correctly tracked. For this reason, the comparison of the number of evaluated points is not meaningful; therefore, in Table 2, it is not reported. On the other hand, all of the other sequences are tracked successfully. Figures 4 and 5 point out significant frames for the sequence otcbvs 03-11s1ir-12 using the ATT and the proposed algorithm, respectively. This sequence represents a challenging situation, due to the presence of similar targets in the scene. Indeed, though the behavior of the original algorithm is correct (Figure 4), a tracking failure occurs with the proposed technique (Figure 5); in this case, from (a) to (e), the tracking is correct; from (f) to (h), the algorithm selects an improper peak in the correlation output plane, thus giving rise to the failure.

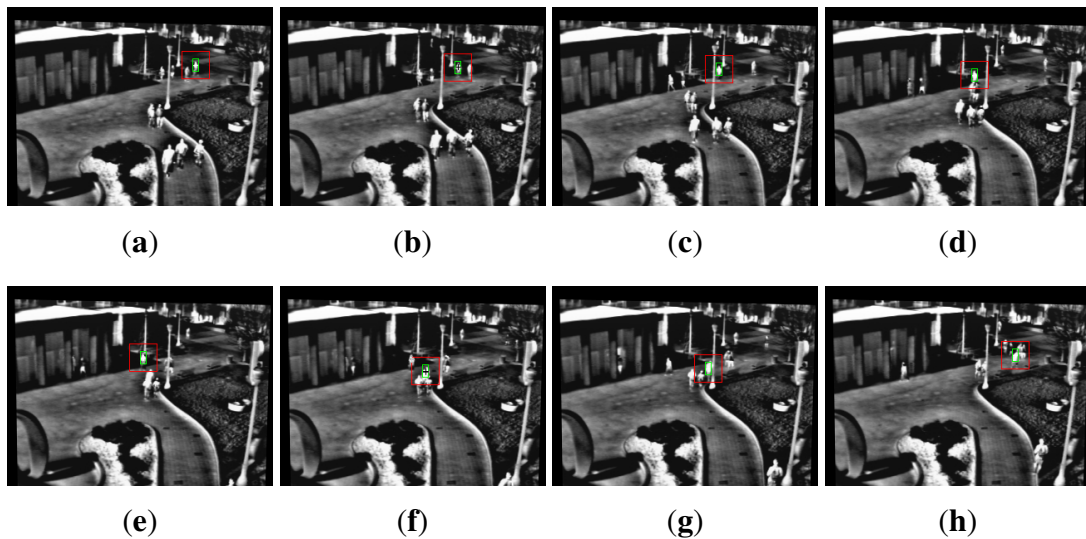
**Figure 3.** Tracking results by running the proposed algorithm on sample sequences extracted from each considered dataset. **(a–d)** OTCBVS sequence 03-12s4ir-1; **(e–h)** AMCOM sequence 16-08-m60; **(i–l)** AIC ir11-1.



**Figure 4.** Tracking results with the ATT algorithm for sequence otcbvs 03-11s1ir-12. (a) Frame 142; (b) Frame 188; (c) Frame 249; (d) Frame 333; (e) Frame 391; (f) Frame 407; (g) Frame 498; (h) Frame 556.



**Figure 5.** Tracking results with the proposed algorithm for sequence otcbvs 03-11s1ir-12. (a) Frame 142; (b) Frame 188; (c) Frame 249; (d) Frame 333; (e) Frame 391; (f) Frame 407; (g) Frame 458; (h) Frame 556.



The first designed metric gives only an indication of the possibility of reducing the computations. With the aim of better evaluating the complexity of the considered algorithms, Tables 3 and 4 summarize the results of the theoretical analysis, introducing the comparison of the number of estimated operations among the same algorithms considered so far. Moreover, the theoretical analysis is complemented with real average speed at run-time. The dominant operations in all considered algorithms are the sums and subtractions performed by IVF and TM; thus, Table 3 shows the number of such operations performed by all considered algorithms as described in Section 5.3. Due to the different kinds of parameters involved, the number of operations  $\Theta$  with ATT and PATT are estimated

by using Equation (16) and the number of operations  $\Omega$  of the reference algorithm are computed with Equation (17). After listing the number of operations and the average time per frame  $T_A$  for each algorithm, the four columns of Table 4 provide a comparison of the proposed algorithm with ATT and with PATT. In both cases, Table 4 provides the percentage of the number of operations needed by using the proposed algorithm with respect to the number of operations needed by using the respective reference algorithm (first and second to last columns). Overall, the theoretical analysis concludes that in most cases, it is possible to reduce the number of operations. In particular, as could be expected, the ratio is more significant on sequences with a relevant number of TM process activations, since the proposed algorithm realizes a real performance gain only acting on this phase. Nonetheless, the algorithm shows an intrinsic capacity to obtain a performance gain in sequences with a relatively low signal-to-noise ratio, thanks to the mechanism used to determine a sufficient set of relevant points. For instance, in sequence *otcbvs 03-11s3ir-1*, the number of activations is comparable among the algorithms, but the gain in terms of reducing the number of operations by using the proposed one is quite significant; in fact, less than 5% of the operations are performed with respect to ATT and PATT, even though the proposed algorithm activates the TM phase a higher number of times than the reference algorithms. Nonetheless, the performance gain becomes less significant when the percentage  $S_{Pe}$  of evaluated points between the reference and proposed algorithms increase. As can be observed, in sequence *amcom 17-02-mantruck*, where  $S_{Pe}$  is high, the savings in the performed operations is rather low in percentage terms (only 86.32% operations with respect to ATT).

As anticipated, since the theoretical analysis presented in Tables 3 and 4 is based on a series of assumptions (Section 5.1), the computation time per frame has been gathered for each algorithm to be able to compare the real performance (third, fifth and seventh column of the same table). The measured running time inherently includes all of the algorithmic details and, obviously, depends on the implementation of the algorithm. Experiments have been carried out by using a 2.13-GHz Intel Core 2 CPU. Similarly to the theoretical comparison of the number of operations, the third to last and last columns show the ratio between the average time per frame for the proposed algorithm and for the respective reference one (expressed in percentage terms). Furthermore, in this case, percentages smaller than 100% indicate savings in running time. In general, the proposed algorithm shows firm improvements with a growing number of activations. For example, when the reference algorithms do not need any TM activation (e.g., in the OTCBVS dataset in sequences *otcbvs 03-11s2ir-4* and *otcbvs 03-11s3ir-2*) or this number is very low (e.g., in the AMCOM dataset in sequences *amcom 14-15-mantruck*, *amcom 16-08-apc*, *amcom 19-06-apc* and *amcom 21-17-apc*), the speed of the proposed algorithm is comparable to those of the reference ones. On the other hand, it is possible that the low signal-to-noise ratio of the sequences or the presence of similar targets in the scene induces a considerable number of activations; in these cases, e.g., in the AIC dataset in sequences *aic ir11-1* and *aic ir11-2* or in the OTCBVS dataset in sequence *otcbvs 03-11s2ir-2*, the proposed algorithm is able to noticeably boost the performance of the target tracking application.

**Table 3.** A comparison of the number of estimated operations ( $\Theta$  and  $\Omega$ ) and the real average time per frame  $T_A$  among the proposed algorithm and the reference ones ATT [17] and PATT [18]. O, OTCBVS dataset; A, AMCOM dataset; AI, AIC dataset.

Dataset Seq.	ATT [17]		PATT [18]		RATT	
	$\Theta$	$T_A$ (ms)	$\Theta$	$T_A$ (ms)	$\Omega$	$T_A$ (ms)
O 03-11s1ir-1	44,327,745	1.38	25,529,427	0.6	1,566,894	0.195
O 03-11s1ir-2	7,153,641	0.105	7,153,641	0.075	F	F
O 03-11s2ir-1	18,367,074	0.705	13,108,293	0.615	5,489,448	0.33
O 03-11s2ir-2	51,987,771	1.515	43,297,551	1.41	3,364,434	0.135
O 03-11s2ir-3	4,810,113	0.195	2,347,884	0.165	955,431	0.06
O 03-11s2ir-4	235,224	0.015	235,224	0.09	240,976	0.015
O 03-11s3ir-1	18,614,277	0.69	17,466,471	0.975	842,783	0.225
O 03-11s3ir-2	16,302,330	0.525	7,713,387	0.375	7,733,224	0.42
O 03-11s3ir-3	6,477,372	0.15	25,256,088	0.87	4,393,722	0.135
O 03-12s4ir-1	34,940,565	1.065	46,422,981	1.365	2,690,423	0.3
O 03-12s6ir-1	10,475,091	0.3	116,460,927	4.11	3,333,523	0.24
A 14-15-mantruck	4,054,347	0.21	3,404,214	0.165	2,757,862	0.17
A 16-08-m60	14,668,830	0.42	6,390,252	0.24	2,848,082	0.20
A 16-08-apc	1,512,621	0.18	1,512,621	0.09	785,195	0.09
A 16-18-apc	5,798,925	0.255	3,130,875	0.21	2,945,375	0.16
A 16-18-m60	1,495,197	0.075	1,009,503	0.075	1,009,726	0.06
A 17-02-mantruck	2,516,679	0.15	4,971,285	0.225	2,172,300	0.14
A 17-02-bradley	2,689,830	0.21	3,391,146	0.18	1,814,151	0.17
A 18-13-m60	6,608,052	0.63	3,627,459	0.15	2,230,301	0.19
A 18-16-m60	26,278,659	1.23	3,150,477	0.15	1,592,641	0.14
A 19-06-apc	3,338,874	0.165	3,555,585	0.165	2,046,767	0.17
A 21-17-apc	3,528,360	0.225	3,840,903	0.165	3,542,997	0.24
AI ir11-1	306,883,104	4.88	362,142,352	5.18	4,829,325	0.39
AI ir11-2	364,246,784	6.99	392,884,096	5.5	2,891,167	0.24

**Table 4.** A comparison of the number of estimated operations ( $\Delta_{\%}O$ ) and real average time per frame  $\Delta_{\%}T_A$  with respect to the reference algorithms, ATT [17] and PATT [18]. O, OTCBVS dataset; A, AMCOM dataset; AI, AIC dataset.

Dataset Seq.	RATT/ATT [17]		RATT/PATT [18]	
	$\Delta_{\%}O$	$\Delta_{\%}T_A$	$\Delta_{\%}O$	$\Delta_{\%}T_A$
O 03-11s1ir-1	3.53%	14.13%	6.14%	32.50%
O 03-11s1ir-2	-	-	-	-
O 03-11s2ir-1	29.89%	46.81%	41.88%	53.66%
O 03-11s2ir-2	6.47%	8.91%	7.77%	9.57%
O 03-11s2ir-3	19.86%	30.77%	40.69%	36.36%
O 03-11s2ir-4	102.45%	100.00%	102.45%	16.67%
O 03-11s3ir-1	4.53%	32.61%	4.83%	23.08%
O 03-11s3ir-2	47.44%	80.00%	100.26%	112.00%
O 03-11s3ir-3	67.83%	90.00%	17.40%	15.52%
O 03-12s4ir-1	7.70%	28.17%	5.80%	21.98%
O 03-12s6ir-1	31.82%	80.00%	2.86%	5.84%
A 14-15-mantruck	68.02%	80.02%	81.01%	101.84%
A 16-08-m60	19.42%	48.03%	44.57%	84.06%
A 16-08-apc	51.91%	51.22%	51.91%	102.43%
A 16-18-apc	50.79%	62.59%	94.08%	76.00%
A 16-18-m60	67.53%	80.00%	100.02%	80.00%
A 17-02-mantruck	86.32%	93.33%	43.70%	62.22%
A 17-02-bradley	67.44%	80.02%	53.50%	93.35%
A 18-13-m60	33.75%	30.68%	61.48%	128.88%
A 18-16-m60	6.06%	11.61%	50.55%	95.17%
A 19-06-apc	61.30%	101.84%	57.56%	101.84%
A 21-17-apc	100.41%	108.39%	92.24%	147.80%
AI ir11-1	1.57%	7.99%	1.33%	7.53%
AI ir11-2	0.79%	3.43%	0.74%	4.36%

### 5.3. Tracking Speed vs. Tracking Robustness

After having analyzed the tracking speed of the proposed approach with respect to the reference algorithms, it is worthwhile to extend this analysis to the state-of-the-art algorithms in target tracking scenarios. For this purpose, a set of experimental tests has been carried out by considering several alternative techniques. In particular, they have been selected from a recently implemented benchmark on online tracking [25]. The benchmark is composed of a rather heterogeneous set of target tracking algorithms, but only the most relevant ones for the scope of this work have been chosen for comparison, *i.e.*, the fastest techniques, based on a study presented in [27]. In particular, nine of 29 algorithms have been selected, and all of the them have been tested using their default parameters, like in [25]. The terminology used for identifying the algorithms in this manuscript directly follows the one used in [25]. Moreover, each sequence has been evaluated also in terms of tracking failures in order to find a trade-off between tracking speed and robustness among the various approaches.

Furthermore, in this case, various datasets have been considered to test the algorithms in different working conditions.

Results gathered using the considered algorithms are summarized in Table 5. The first two columns identify the name of each sequence and its length. Then, for each considered technique, two columns show the measurements: the first represents the number of tracked frames  $Tf$  for a given sequence (expressed as a percentage value with respect to the length of the sequence), and the latter represents the average tracking speed of the algorithm expressed in frames per second (fps). The first three algorithms are ATT [17], PATT [18] and the proposed one (abbreviated as RATT). Each subsequent technique is identified by the same acronyms used in [25]. Sequences are grouped by dataset, and their average results are highlighted in bold, just as an indication of the performance of the techniques on different datasets. The percentage of tracked frames  $Tf$  for each dataset is computed as the ratio between the sum of the number of correctly tracked frames and the total number of frames in a given dataset ( $L$ ). Similarly, the average speed ( $S_A$ ) for each dataset is computed as the ratio between the sum of the average frames per second achievable in each sequence of the dataset itself and the total number of frames in the same dataset.

The performance in terms of achievable frames per second for each technique is quite consistent within each dataset. Indeed, the tracking speed for each technique depends both on parameters that are in common within the dataset (such as image resolution) and on parameters that can be set separately for tracking each sequence (e.g., the target window size, which is indicated in Table 2). As expected, the fastest algorithm is KMS [28]. By considering separately the three datasets, the proposed algorithm is second to only KMS, except for the OTCBVS dataset. In fact, in this case, the CSK [29] algorithm provides a higher frame rate than RATT. On the other hand, RATT is faster than KMS on the AMCOM and AIC datasets. The reason resides in the different behavior of the two algorithms. More in detail, CSK is an efficient algorithm that exploits the redundancy that characterize the targets in the process of sampling their features. Generally, OTCBVS sequences are characterized by well-defined target shapes that generate a high contrast with the background. Conversely, the AMCOM and AIC sequences are characterized by a low signal-to-noise ratio; indeed, these sequences present a lot of noise that changes the background around the target frame by frame. This fact indicates that CSK should be preferred in sequences where the target shape and background do not considerably change, such as the ones in the OTCBVS scenario.

**Table 5.** A comparison of the percentages of tracked frames and the speed for different tracking algorithms applied to three datasets. TM, template matching.

Sequence	L		AITT [17]		PAITT [18]		RAITT		KMS [28]		CSK [29]		TM		PD		VR		RS		CPF [30]		MS [31]		SMS [20]		
	Tf (%)	SA (fps)	Tf (%)	SA (fps)	Tf (%)	SA (fps)	Tf (%)	SA (fps)	Tf (%)	SA (fps)	Tf (%)	SA (fps)	Tf (%)	SA (fps)	Tf (%)	SA (fps)	Tf (%)	SA (fps)	Tf (%)	SA (fps)	Tf (%)	SA (fps)	Tf (%)	SA (fps)	Tf (%)	SA (fps)	Tf (%)
otcbvs 03-11s1ir-1	154	100 322	100 430	100 521	35 3,998	100 207	100 81	100 74	100 74	100 65	100 76	100 71	19 60	19 51													
otcbvs 03-11s1ir-2	637	100 546	100 556	64 442	6 6,217	27 1,191	100 98	26 99	25 79	7 69	13 71	2 65	1 35														
otcbvs 03-11s2ir-1	557	100 412	100 427	100 487	33 5,189	41 637	45 98	55 93	100 82	49 61	58 48	28 64	13 26														
otcbvs 03-11s2ir-2	339	100 309	100 319	100 538	3 5,895	34 1,073	100 99	89 86	0 82	2 62	4 91	2 64	2 16														
otcbvs 03-11s2ir-3	96	100 521	100 529	100 560	6 5,764	100 1,044	100 100	100 86	100 77	100 61	76 90	6 64	12 46														
otcbvs 03-11s2ir-4	24	100 575	100 551	100 575	32 3,890	100 209	100 96	36 68	0 65	32 57	100 84	60 61	20 32														
otcbvs 03-11s3ir-1	84	100 414	100 370	100 513	59 5,133	100 537	100 99	100 75	21 73	100 59	100 78	20 63	7 65														
otcbvs 03-11s3ir-2	787	100 444	100 476	100 466	4 6,023	1 1,083	3 91	46 83	46 87	4 62	21 79	3 64	1 39														
otcbvs 03-11s3ir-3	448	100 533	100 385	100 538	4 5,343	96 698	33 80	100 86	100 85	3 68	100 82	73 63	3 134														
otcbvs 03-12s4ir-1	270	100 358	100 324	100 494	26 4,720	100 561	100 91	100 81	100 92	100 75	100 77	25 63	100 82														
otcbvs 03-12s6ir-1	323	100 494	100 171	100 509	91 4,924	100 570	100 90	100 76	17 92	100 60	20 80	80 61	6 27														
<b>OTCBVS TOTAL</b>	<b>3719</b>	<b>100%450</b>	<b>100%417</b>	<b>94%492</b>	<b>21%5,523</b>	<b>51%850</b>	<b>63%93</b>	<b>68%87</b>	<b>57%83</b>	<b>35%65</b>	<b>46%74</b>	<b>25%64</b>	<b>12%49</b>														
amcom 14-15-mantruck	281	100 1,626	100 1,754	100 1,745	1 6,663	53 864	75 496	100 474	100 436	100 280	5 124	1 379	1 19														
amcom 16-08-m60	290	100 1,220	100 1,550	100 1,648	1 6,250	59 821	100 588	100 422	41 463	100 334	41 115	1 399	14 16														
amcom 16-08-apc	80	100 1,709	100 2,000	100 2,011	4 7,557	12 805	100 555	74 416	15 451	62 326	62 110	4 393	9 11														
amcom 16-18-apc	300	100 1,515	100 1,626	100 1,771	1 6,902	1 830	100 571	27 385	23 430	37 327	10 128	1 384	20 19														
amcom 16-18-m60	103	100 2,083	100 2,083	100 2,128	4 7,529	100 816	100 416	46 399	13 426	100 324	78 102	3 364	6 7														
amcom 17-02-mantruck	221	100 1,802	100 1,587	100 1,818	1 7,578	41 833	100 463	100 478	100 409	52 335	54 116	2 406	2 13														
amcom 17-02-bradley	185	100 1,626	100 1,709	100 1,745	3 7,506	11 770	81 503	43 470	43 426	38 329	59 110	2 390	38 14														
amcom 18-13-m60	227	100 966	100 1,802	100 1,671	63 7,301	13 791	65 548	9 452	2 451	57 324	2 102	4 402	57 23														
amcom 18-16-m60	162	100 612	100 1,802	100 1,826	22 7,114	31 805	77 497	100 468	74 415	56 321	56 103	9 389	93 9														
amcom 19-06-apc	208	100 1,754	100 1,754	100 1,745	10 7,609	29 604	100 535	38 473	100 436	59 326	96 110	2 391	3 15														
amcom 21-17-apc	360	100 1,587	100 1,754	100 1,541	1 6,768	1 763	1 536	100 420	100 410	1 322	1 112	1 392	1 15														
<b>AMCOM TOTAL</b>	<b>2,417</b>	<b>100%1,477</b>	<b>100%1,725</b>	<b>100%1,737</b>	<b>9%7,045</b>	<b>29%792</b>	<b>76%527</b>	<b>70%441</b>	<b>62%431</b>	<b>56%322</b>	<b>34%114</b>	<b>2%391</b>	<b>20%16</b>														
aic ir11-1	263	100 151	100 145	100 472	100 3,496	100 162	100 73	100 72	100 72	100 60	100 81	38 65	23 2														
aic ir11-2	155	100 115	100 138	100 508	100 3,769	100 196	100 72	100 85	100 68	100 50	100 67	12 62	19 2														
<b>AIC TOTAL</b>	<b>418</b>	<b>100%138</b>	<b>100%142</b>	<b>100%485</b>	<b>100%3,597</b>	<b>100%175</b>	<b>100%73</b>	<b>100%77</b>	<b>100%71</b>	<b>100%56</b>	<b>100%76</b>	<b>28%64</b>	<b>22%2</b>														



The KMS algorithm [28] is the fastest. However, in this case, the speed comes at the cost of a general decrease of robustness. In fact, the KMS algorithm was one of the algorithms with the lowest performance in terms of the number of correctly tracked frames. The only exception is the result in the AIC dataset, where KMS continues to outperform the proposed algorithm in terms of speed, and it is also able to achieve the same result in terms of robustness. In this case, it is worth considering that the test sequences do not present particular challenges in terms of possible failures, due to the static and rather uniform nature of the background; in fact, most algorithms (all, but two) have been able to correctly track all of the frames of the sequences.

Nonetheless, it is worth noticing that in AIC, the relative improvements concerning the speed of the RATT algorithm *versus* the speed of other algorithms are significantly higher than in the other two datasets. In fact, RATT performance in terms of average speed is better, but still comparable to, e.g., the performance of ATT and PATT (in OTCBVS, on average, 492 fps are achieved by RATT, whereas ATT and PATT reach 450 fps and 417 fps, respectively; in AMCOM, on average, 1737 fps are achieved by RATT, whereas ATT and PATT reach 1477 fps and 1725 fps). Instead, in AIC, the improvement in average speed is much better in relative terms than other algorithms; e.g., in RATT, on average, 485 fps can be achieved, whereas only 138 fps and 142 fps can be reached by ATT and PATT, respectively (tripling the improvement in speed with respect to the other datasets). In fact, as indicated in Table 2, the AIC dataset is characterized by a high number of TM activations ( $m$ ), which is the main reason for the computational time savings.

## 6. Conclusions

This paper presented a novel algorithm for improving the speed performance in target tracking applications making use of template matching (TM) techniques in forward-looking infrared images (FLIR). The template matching algorithm is improved by selecting a representative group of points on which it has to be executed, thus reducing both execution time and resources usage. The selection strategy is based on dynamic thresholding and on the results of the target detection (TD) phase. After analyzing the theoretical impact, the paper discusses the results obtained by comparing the proposed technique and the reference implementations on different datasets. Moreover, several alternative techniques are evaluated and included in the performance analysis.

The proposed algorithm showed significant computational performance improvements with respect to reference algorithms, although this came at the cost of introducing some more parameters. Besides target window size and sub-frame size, the two reference algorithms depend only on a probability threshold and on the value of a parameter, whereas the new implementation requires also weights for computing probability and TM values to compare relevant points and a minimum score threshold to get a set of relevant points large enough to be representative of the whole sub-frame without losing essential information.

A different weighting strategy might be devised in the future to improve the precision of the algorithm and to reduce its dependency on arbitrary parameters that may be a hindrance to its use in real-time automatic target tracking applications. For instance, a strategy based on frame features might be used to determine, in a frame-by-frame fashion, the minimum score needed to ensure

that the relevant points are a representative set for the current frame. With a similar strategy, a mechanism to automatically set the weights on probability and TM values could be determined in a frame-by-frame fashion. Even though such a strategy might increase the computational complexity of the algorithm, the performance gain found in this paper should be enough to cover the increased complexity.

### Author Contributions

Gianluca Paravati coordinated the writing of the manuscript, made substantial contribution in the research of the related work and coordinated the preparation of the testing phase. Stefano Esposito designed and implemented the target tracking algorithm.

### Conflicts of Interest

The authors declare no conflict of interest.

### References

1. Conaire, C.O.; O'Connor, N.E.; Smeaton, A.F. Thermo-visual feature fusion for object tracking using multiple spatiogram trackers. *Mach. Vis. Appl.* **2008**, *19*, 483–494.
2. Sanna, A.; Pralio, B.; Lamberti, F.; Paravati, G. A Novel Ego-Motion Compensation Strategy for Automatic Target Tracking in FLIR Video Sequences taken from UAVs. *IEEE Trans. Aerosp. Electron. Syst.* **2009**, *45*, 723–734.
3. Paravati, G.; Pralio, B.; Sanna, A.; Lamberti, F. A reconfigurable multi-touch remote control system for teleoperated robots. In Proceedings of the 29th IEEE International Conference on Consumer Electronics (ICCE2011), Las Vegas, NV, USA, 9–12 January 2011; pp. 153–154.
4. Baldi, M.; Giacomelli, R.; Marchetto, G. Time-driven access and forwarding for industrial wireless multihop networks. *IEEE Trans. Ind. Inform.* **2009**, *5*, 99–112.
5. Baldi, M.; Marchetto, G.; Ofek, Y. A scalable solution for engineering streaming traffic in the future Internet. *Comput. Netw.* **2007**, *51*, 4092–4111.
6. Baldi, M.; Corrà, M.; Fontana, G.; Marchetto, G.; Ofek, Y.; Severina, D.; Zadedyurina, O. Scalable fractional lambda switching: A testbed. *J. Opt. Commun. Netw.* **2011**, *3*, 447–457.
7. Cao, X.; Lan, J.; Yan, P.; Li, X. Vehicle detection and tracking in airborne videos by multi-motion layer analysis. *Mach. Vis. Appl.* **2012**, *23*, 921–935.
8. Zhu, J.; Lao, Y.; Zheng, Y. Object tracking in structured environments for video surveillance applications. *IEEE Trans. Circuits Syst. Video Technol.* **2010**, *20*, 223–235.
9. Fang, J.; Wang, Q.; Yuan, Y. Part-Based Online Tracking With Geometry Constraint and Attention Selection. *IEEE Trans. Circuits Syst. Video Technol.* **2014**, *24*, 854–864.
10. Yang, F.; Lu, H.; Yang, M.-H. Robust Superpixel Tracking. *IEEE Trans. Image Process.* **2014**, *23*, 1639–1651.
11. Bai, Y.; Tang, M. Object Tracking via Robust Multitask Sparse Representation. *IEEE Signal Process. Lett.* **2014**, *21*, 909–913.

12. Zhang, S.; Yao, H.; Sun, X.; Lu, X. Sparse coding based visual tracking: Review and experimental comparison. *Pattern Recognit.* **2013**, *46*, 1772–1788.
13. Zhang, S.; Yao, H.; Zhou, H.; Sun, X.; Liu, S. Robust visual tracking based on online learning sparse representation. *Neurocomputing* **2013**, *100*, 31–40.
14. Zhang, S.; Yao, H.; Zhou, H.; Sun, X.; Liu, S. Robust Visual Tracking Using an Effective Appearance Model Based on Sparse Coding. *ACM Trans. Intell. Syst. Technol.* **2012**, *3*, 1–18.
15. Dawoud, A.; Alam, M.S.; Bal, A.; Loo, C. Target tracking in infrared imagery using weighted composite reference function-based decision fusion. *IEEE Trans. Image Process.* **2006**, *15*, 404–410.
16. Braga-Neto, U.; Choudhary, M.; Goutsias, J. Automatic target detection and tracking in forward-looking infrared image sequences using morphological connected operators. *J. Electron. Imaging* **2004**, *13*, 802–813.
17. Alam, M.S.; Bal, A. Automatic Target Tracking in FLIR Image Sequences Using Intensity Variation Function and Template Modeling. *IEEE Trans. Instrum. Meas.* **2005**, *54*, 1846–1852.
18. Lamberti, F.; Sanna, A.; Paravati, G. Improving Robustness of Infrared Target Tracking Algorithms Based on Template Matching. *IEEE Trans. Aerosp. Electron. Syst.* **2011**, *47*, 1462–1480.
19. Paravati, G.; Sanna, A.; Pralio, B.; Lamberti, F. A Genetic Algorithm for Target Tracking in FLIR Video Sequences Using Intensity Variation Function. *IEEE Trans. Instrum. Meas.* **2009**, *58*, 3457–3467.
20. Collins, R.T. Mean-shift blob tracking through scale space. In Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Madison, WI, USA, 18–20 June 2003; IEEE Press: Piscataway, NJ, USA, 2003; pp. 234–240.
21. Yilmaz, A.; Shafique, K.; Lobo, N.; Li, X.; Olson, T.; Shah, M.A. Target-tracking in FLIR imagery using mean-shift and global motion compensation. In Proceedings of the IEEE Workshop Computer Vision beyond Visible Spectrum, Kauai, HI, USA, 14 December 2001; pp. 54–58.
22. Yilmaz, A.; Shafique, K.; Shah, M. Tracking in airborne forward looking infrared imagery. *Image Vis. Comput.* **2003**, *21*, 623–635.
23. Paravati, G.; Sanna, A.; Lamberti, F. An image feature descriptors-based recovery activation metric for FLIR target tracking. In Proceedings of the IADIS International Conference on Computer Graphics, Visualization, Computer Vision and Image Processing (CGVCVIP 2011), Rome, Italy, 24–26 July 2011; pp. 67–74.
24. Davis, J.; Sharma, V. IEEE OTCBVS WS Series Bench; Background-Subtraction Using Contour-Based Fusion of Thermal and Visible Imagery. *Comput. Vis. Image Underst.* **2007**, *106*, 162–182.
25. Wu, Y.; Lim, J.; Yang, M.H. Online object tracking: A benchmark. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; IEEE Press: Piscataway, NJ, USA, 2013; pp. 2411–2418.

26. Conaire, C.O.; O'Connor, N.E.; Cooke, E.; Smeaton, A.F. Comparison of fusion methods for thermo-visual surveillance tracking. In Proceedings of the IEEE International Conference on Information Fusion, Florence, Italy, 10–13 July 2006; IEEE Press: Piscataway, NJ, USA, 2006; pp. 1–7.
27. Lamberti, F.; Sanna, A.; Paravati, G.; Belluccini, L. IVF<sup>3</sup>: Exploiting Intensity Variation Function for high performance pedestrian tracking in FLIR imagery. *Opt. Eng.* **2014**, *53*, 1–15.
28. Comaniciu, D.; Ramesh, V.; Meer, P. Kernel-based object tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2003**, *25*, 564–577.
29. Henriques J.; Caseiro, R.; Martins, P.; Batista, J. Exploiting the circulant structure of tracking by detection with kernels. In Proceedings of the 12th European Conference on Computer Vision, Florence, Italy, 7–13 October 2012; Springer: Berlin/Heidelberg, Germany, 2012; pp. 702–715.
30. Perez, P.; Hue, C.; Vermaak, J.; Gangnet, M. Color-based probabilistic tracking. In Proceedings of the 7th European Conference on Computer Vision, Copenhagen, Denmark, 28–31 May 2002; Springer: Berlin/Heidelberg, Germany, 2002; pp. 661–675.
31. Collins, R.T.; Liu, Y.; Leordeanu, M. Online selection of discriminative tracking features. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1631–1643.

# Feature Point Descriptors: Infrared and Visible Spectra

Pablo Ricaurte, Carmen Chilán, Cristhian A. Aguilera-Carrasco, Boris X. Vintimilla and Angel D. Sappa

**Abstract:** This manuscript evaluates the behavior of classical feature point descriptors when they are used in images from long-wave infrared spectral band and compare them with the results obtained in the visible spectrum. Robustness to changes in rotation, scaling, blur, and additive noise are analyzed using a state of the art framework. Experimental results using a cross-spectral outdoor image data set are presented and conclusions from these experiments are given.

Reprinted from *Sensors*. Cite as: Ricaurte, P.; Chilán, C.; Aguilera-Carrasco, C.A.; Vintimilla, B.X.; Sappa, A.D. Feature Point Descriptors: Infrared and Visible Spectra. *Sensors* **2014**, *14*, 3690–3701.

## 1. Introduction

Recent advances in imaging technologies have increased the usage of cameras working at different spectral bands. As a result, novel solutions to classical problems have been proposed improving the results that can be obtained when only the visible spectrum images are considered (e.g., [1,2]). Infrared imaging represents one of the examples of such novel technologies. These images cover the spectral band from 0.75  $\mu\text{m}$  to 15  $\mu\text{m}$ , which is split up into the following categories: Near-Infrared (NIR: 0.75–1.4  $\mu\text{m}$ ), Short-Wave Infrared (SWIR: 1.4–3  $\mu\text{m}$ ), Mid-Wave Infrared (MWIR: 3–8  $\mu\text{m}$ ) or Long-Wave Infrared (LWIR: 8–15  $\mu\text{m}$ ). Images from each one of these categories have a particular advantage for a given application; for instance, NIR images are generally used in gaze detection and eye tracking applications [3]; the SWIR spectral band has shown its usage in heavy fog environments [4]; MWIR is generally used to detect temperatures somehow above body temperature in military applications; finally, LWIR images have been used in video surveillance and driver assistance (e.g., [5,6]). Recently, a personal thermal imaging device has been developed (FLIR ONE (<http://www.flir.com/flirone/>)) to be used with smartphones for applications such as security, home repairs, and outdoor activities. The current work is focused on the LWIR domain, which corresponds to the infrared spectral band farthest from the visible spectrum.

Like in visible spectrum image processing, different algorithms must be envisaged to handle images from the infrared domain (e.g., [7–9]). Actually, in order to tackle the applications mentioned above we have to address classical computer vision problems such as feature selection and tracking, image registration, pattern recognition, just to mention a few. The easiest way is to adopt classical tools from the visible spectrum to this new domain. One of these tools is the feature point detection and description, which has been a very active research topic during the last decade in the computer vision community. Feature detection and description in the LWIR spectral band is especially attractive in motion related applications, where lighting conditions are prone to change more rapidly than temperature (e.g., SLAM [10], egomotion [11], remote sensing [12]). Due to the large amount of contributions on this topic there were several works on the literature evaluating

and comparing their performance in the visible spectrum case (e.g., [13–16]). However, to the best of our knowledge, there are no studies in the literature considering other spectral bands.

The current work proposes to study the performance of feature point descriptors when they are used in the far infrared domain (LWIR), and at the same time compare the results with those obtained in the visible domain (VS). The evaluation is performed using a data set from a cross-spectral stereo rig; hence a similar image is used to evaluate the performance in the two domains. Since there is a large amount of algorithms in the literature, we decided to select the most representative and recent ones. Hence, our study includes: SIFT [17], SURF [18], ORB [19], BRISK [20], BRIEF [21] and FREAK [22]. Although each descriptor has its own advantages and disadvantages, coarsely speaking they can be classified into two categories: (i) those based on image derivatives (e.g., SIFT, SURF) and (ii) those based on image intensities (e.g., ORB, BRISK, BRIEF, FREAK). Since images from the LWIR spectrum have less texture than those from the VS spectrum a lower number of features will be detected in the LWIR domain. However, it is difficult to predict whether this lack of texture would affect the performance of the different approaches when used with LWIR images.

The remainder of the paper is organized as follows: the evaluation methodology used for studying the performance in both spectral bands is presented in Section 2. Experimental results on a cross-spectral data set are presented in Section 3. Finally, conclusions and discussions are given in Section 4.

## 2. Evaluation Framework

The performance of different descriptors has been evaluated using the framework proposed by Khvedchenia [23]. This framework has been proposed for evaluating the performance of feature descriptors in the visible spectrum. It is intended to find the best approach for the correspondence problem when common image transformations are considered: rotation in the image plane, changes in the image size, blur and presence of noise in the images. In order to take into account all these possible changes, the given images are modified; then the different descriptors are applied and the matching with those points in the given images are considered as a ground truth. A brute force strategy is used for finding the matching, together with a L2 norm or Hamming distance, as detailed in Table 1. The brute force matching finds the closest descriptor in the second set by trying all the possible combinations. The percentage of correct matches between the ground truth image and the modified one is used as a criterion for the evaluation (Section 4). The transformations applied to the given images are detailed below:

**Rotation:** the study consists in evaluating the sensibility to rotations of the image. The rotations are in the image plane spanning the 360 degrees; a new image is obtained every 10 degrees.

**Scale:** the size of the given image is changed and the repeatability of a given descriptor is evaluated. The original image is scaled in between 0.2 to 2 times its size with a step of 0.1 per test. Pixels of scaled images are obtained through a linear interpolation.

**Blur:** the robustness with respect to blur is evaluated. It consists of a Gaussian filter iteratively applied over the given image. At each iteration the size of the kernel filter ( $K \times K$ ) used to blur the image is update as follows:  $K = 2n + 1$ , where  $n = \{1, 2, \dots, 9\}$ .

**Noise:** this final study consists in adding noise to the original image. This process is implemented by adding to the original image a personalized image. The value of the pixels of the personalized image are randomly obtained following a uniform distribution with  $\mu = 0$  and  $\sigma = t$ , where  $t = \{0, 10, 20, \dots, 100\}$ .

**Table 1.** Algorithms evaluated in the study.

Feature Descriptor Algorithm	Matcher Norm Type
SIFT	L2 Norm
SURF	L2 Norm
ORB	Hamming Distance
BRISK	Hamming Distance
BRIEF (SURF as a detector)	Hamming Distance
FREAK (SURF as a detector)	Hamming Distance

In the original framework proposed by Khvedchenia, lighting changes were also considered, since that study was only intended for images in the visible spectrum. In the current work, since images from the LWIR spectrum are considered, changes in the intensity values won't follow the same behavior all through the image (like lighting changes in the visible spectrum). Intensity values in LWIR images are related with the material of the objects in the scene. In summary, a study similar to the lighting changes is not considered in the current work. Figure 1 shows an illustration of a couple of cross-spectral images (visible spectrum: VS and long-wave Infrared: LWIR images) together with their corresponding transformed images. The current work does not include comparisons on the execution time performance since execution time is an intrinsic characteristic of the descriptors; hence, independently of the spectral band the same performance will be obtained. Evaluations of the execution time performance for the different descriptors can be found in [23].

**Figure 1.** Illustration of a pair of images from the evaluation dataset ((**top**) LWIR and (**bottom**) VS) together with their corresponding transformed images: (**a**) original ones; (**b**) rotation; (**c**) scale; (**d**) blur; (**e**) noise.

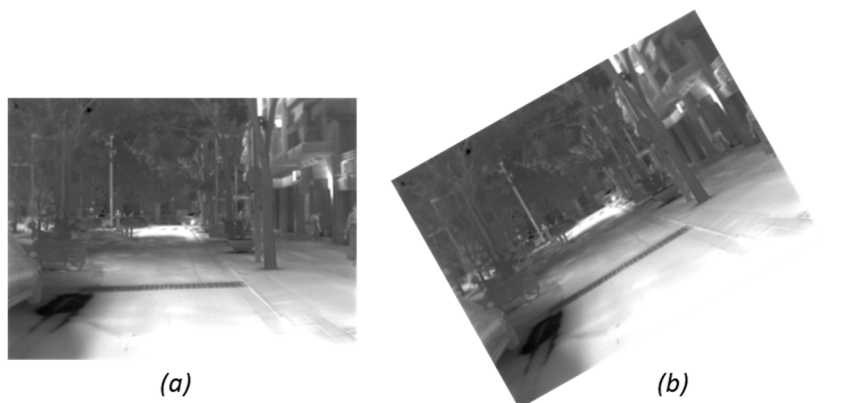
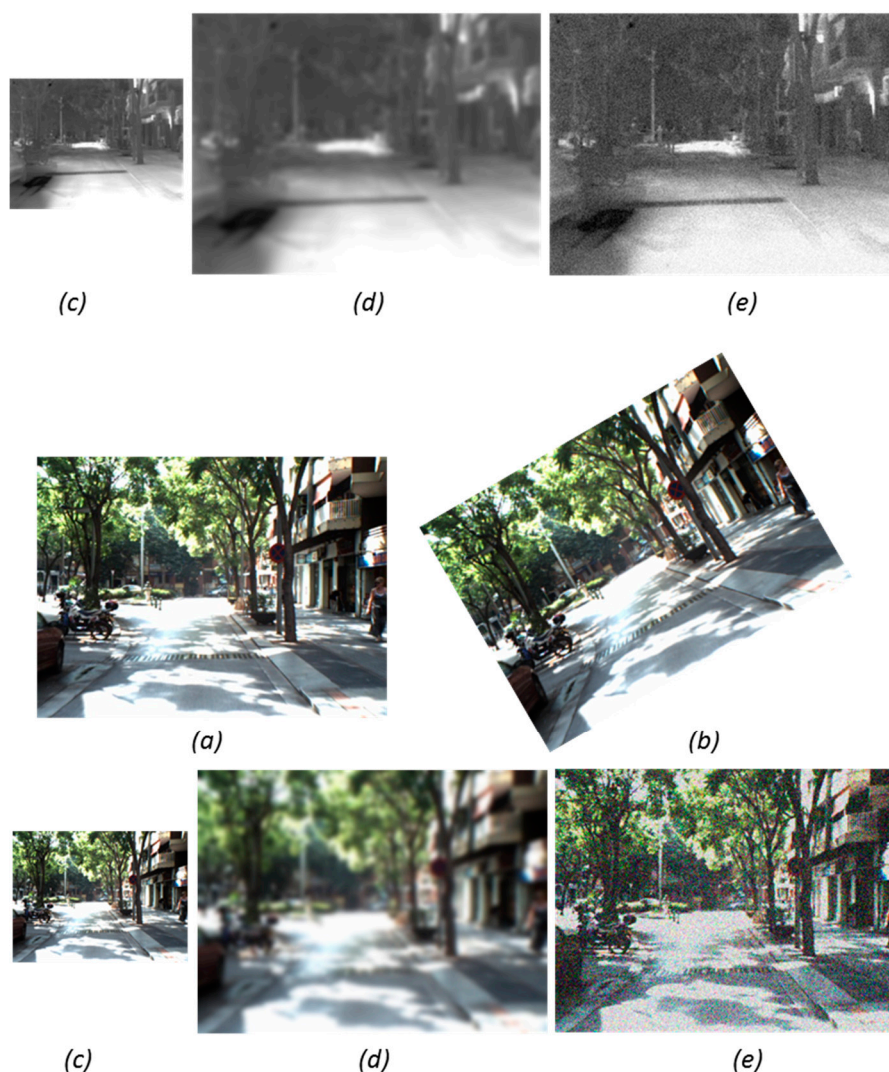


Figure 1. *Cont.*

### 3. Experimental Results

The framework presented above has been used to evaluate the performance of different feature descriptor algorithms in a cross-spectral data set consisting of 40 outdoor images (VS and LWIR). The images were obtained with a multispectral stereo head consisting of a pair of cameras working in different spectral bands. The VS images were obtained with an ACE camera, from Basler, with a resolution of  $658 \times 492$  pixels; while the LWIR images were obtained with a Gobi-640-GigE camera, from Xenixs. Both cameras are synchronized using an external trigger. Camera focal lengths were set so that pixels in both images contain similar amount of information from the given scene. This particular set up allows us to have images from different spectral bands of the same scenario. Note that the only preprocessing applied to the cross-spectral images is the color conversion of VS images to grey levels; there is no additional preprocessing or enhancement to highlight features or increase contrast.



**Figure 2.** Pairs of cross-spectral images contained in the data set.



There are some recent works on the infrared image modeling and filtering (e.g., [24,25]) but this kind of study is out of the scope of current paper. Figure 2 presents some of the cross-spectral images contained in the dataset (<http://www.cvc.uab.es/adas/projects/simeve/>).

For each algorithm and transformation the number of correct matches, with respect to those in the original image, is computed and used for measuring the performance. In order to take into account the amount of points correctly detected by each of the tested algorithms, the results from SIFT are used as a reference. This allows us to measure the performance in each of the test and at the same time to compare the results with those obtained by other approaches. The proposed performance measure is computed as follows:

$$performance = \frac{\#correct\ matches\ (Alg.i,\ Transf.j)}{\#correspondences\ (SIFT,\ Given\ image)} \quad (1)$$

Note that this performance measure can give values higher than one, which means that the evaluated algorithm obtains more features than those computed by SIFT in the given image. The

algorithms evaluated in the current work are presented in Table 1. In the cases of BRIEF and FREAK the SURF algorithm is used as a detector. In ORB, BRISK, BRIEF and FREAK the Hamming distance is used, instead of L2 norm, for speeding up the matching. For each transformation (Section 2) a set of images is obtained; for instance, in the rotation case 36 images are evaluated.

**Figure 3.** Performance in the rotation case: (a) visible spectrum; (b) LWIR spectrum.

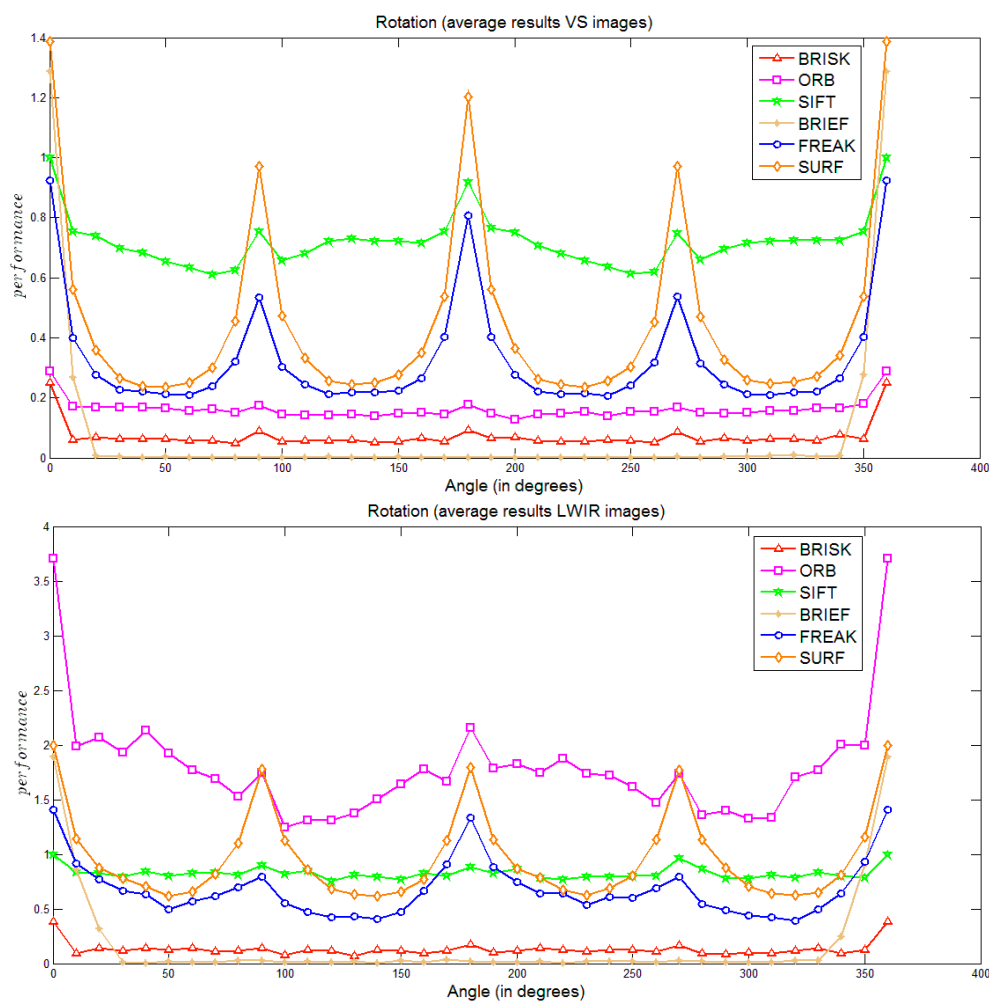


Figure 3 depicts average results obtained when the given images are rotated between 0 and 360 degrees. It can be observed that in both cases (VS and LWIR) the most robust algorithm is SIFT. It can be appreciated that its performance remains almost constant along the different rotations (in particular in the LWIR case); it only decreases at the beginning ( $\pm 10$  degrees) but then does not change so much. On the other hand, the BRIEF algorithm (using SURF as a detector) is the most sensitive to rotations; actually, its performance drop to zero after applying a rotation of just 20 degrees in the VS case and after a rotation of 30 degrees in the LWIR case. In the case of SURF and FREAK, a slightly better performance was appreciated in the LWIR case where the performance does not decrease as much as in the VS case. Using the number of points detected by SIFT as a reference allows us to visualize that ORB has a considerably larger amount of points when used in the LWIR case. In spite of its performance is not as good as in the VS case, showing

a large decrease just after a rotation of 10 degrees. Finally, BRISK shows a poor performance in both domains.

In the scale study, on average the algorithms have a better performance in the LWIR domain than in the VS one. In both cases BRISK shows the worst performance followed by BRIEF. The algorithms SIFT, FREAK and SURF are the most stable with respect to scale changes. Similarly to in the previous case ORB is able to detect a large number of points in the LWIR spectrum. Even though its performance decay considerably, most of the times is the algorithm with most detected points. On the contrary, it is among the algorithm with less detected points in the VS domain. Its performance in the VS domain is quite stable. Figure 4 shows these results.

**Figure 4.** Performance to changes in scale: (a) visible spectrum; (b) LWIR spectrum.

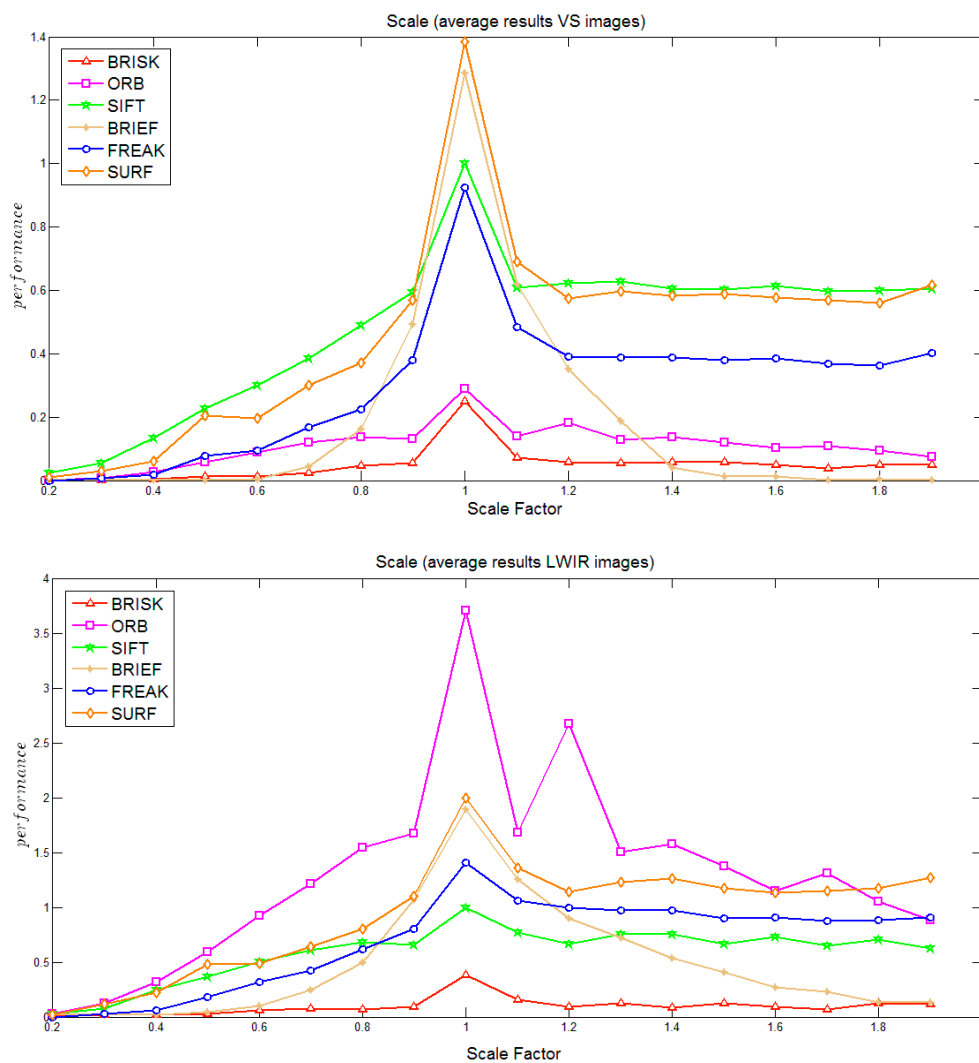
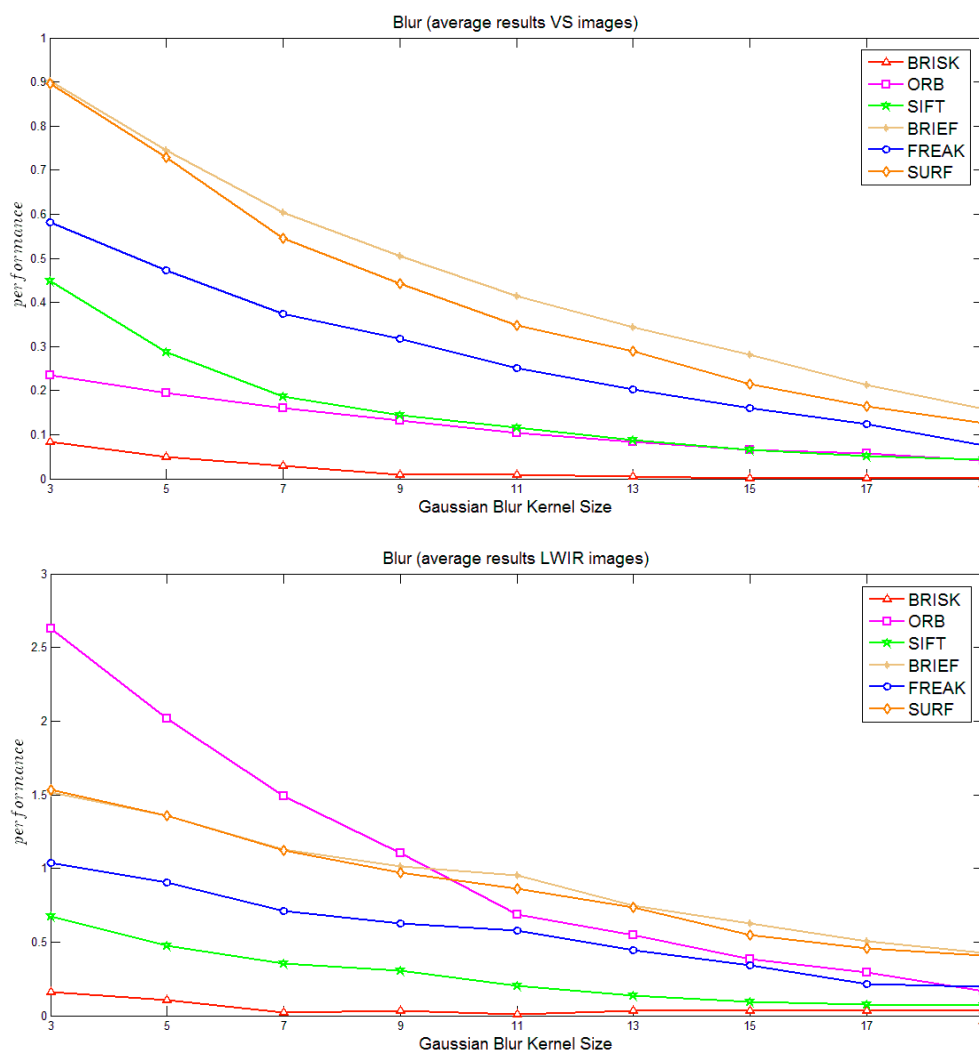
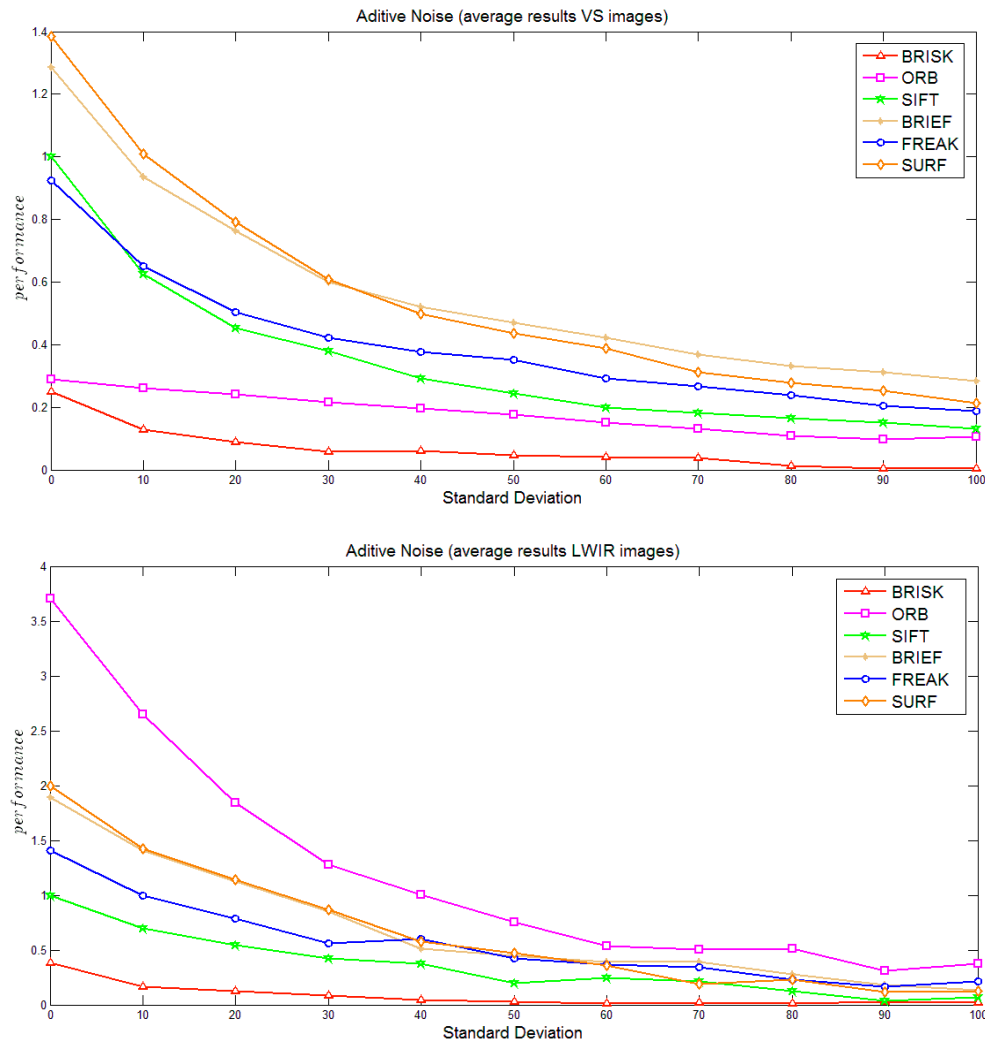


Figure 5 presents the study of robustness of the different algorithms when the given images are degraded using a Gaussian filter of increasing size. In general all the algorithms in both spectrums are equally affected showing a decrease in performance with the increase of kernel size. In the particular case of LWIR, ORB shows the worst performance; in other words it seems to be the most sensitive to blur. This fact can be appreciated in the fast decrease in performance.

**Figure 5.** Performance to image degradation (blur): (a) visible spectrum; (b) LWIR spectrum.



Finally, Figure 6 shows the curves obtained when additive noise is considered. As expected the performance of all the algorithms is degraded with noise. Similarly to in the case of blur, the performance of all the algorithms decreases with a similar behavior. In the VS spectrum ORB is one of the most robust algorithms, while its performance in the LWIR is this worst. This bad performance and sensitivity to noisy data can be explained by the nature of images (low contrast) together with the way this algorithm detect feature points (based on FAST, which uses an intensity threshold between the center pixel and those in a circular ring). On the contrary, in the VS domain, although ORB is affected by noisy data like all the other algorithms, it is not as evident in the LWIR case because of greater contrast noise in the images.

**Figure 6.** Noise case study: (a) visible spectrum; (b) LWIR spectrum.

#### 4. Discussion

As mentioned in Section 1, the descriptors considered in the current work can be coarsely classified as: (i) those based on gradient information; (ii) those based on intensity information. We study whether it is possible to find some correlation between the descriptor's family and the improvement or the drop in performance in the different experiments. The lack of texture in the LWIR domain was one of the focuses of our study. Since it is one of the characteristics of LWIR images we tried to see how it affects the performance mainly on those descriptors based on the usage of gradient information. As mentioned above, the images used to study the algorithm performance (LWIR and VS) are the ones provided by the cameras. There is no preprocessing to filter or improve their contrast.

Looking at the results presented in the previous plots we can conclude that the algorithm ORB is the one that detects most of the features in the LWIR domain. This conclusion is related with the lack of texture and low contrast of LWIR images that affects those algorithms based on gradient information. In order to unveil additional conclusions we propose to compute the recall, similar to [14], for each experiment with the different transformation. It is computed as follows:

$$recall = \frac{\#correct\ matches}{\#correspondences} \quad (2)$$

where  $\#correspondences$  represents the number of features detected/described in the given image by the algorithm being tested (used as a ground truth), and  $\#correct\ matches$  are the matches obtained after transforming the image and detecting/describing feature points with the algorithm being tested. This recall is computed for the different combinations of algorithms and transformations ( $recall_i^j$ , where  $i = \{\text{blur, rotation, noise, scale}\}$  and  $j = \{\text{BRISK, ORB, SIFT, BRIEF, FREAK, SURF}\}$ ) and for every set of images ( $recall\_LWIR_i^j, recall\_VS_i^j$ ). Finally, we propose to compute the average of differences between ( $recall\_LWIR_i^j, recall\_VS_i^j$ ); this will be referred to as *ARD: Average Recall Difference*. This value can be used to compare the performance of an algorithm in each spectral band (a negative value means its performance is better in the visible spectrum than in the infrared one):

$$ARD_i^j = \frac{\sum_{k=1}^n recall\_LWIR_{i_k}^j - recall\_VS_{i_k}^j}{n} \quad (3)$$

where  $n$  depends on the transformation, for instance in the rotation case it consists of 36 elements (see Figures 3–6 for more details). These *ARDs* are presented in Table 2; since the pairs of cross-spectral images contained in the data set correspond to the same scenario, the *ARD* gives an idea of the difference in performance for each transformation. Since this study is focused on the LWIR spectrum we identify the algorithms with best behavior in this domain. On average, the algorithm SURF has the best behavior in the LWIR in the blur transformation; while in the case of rotation SIFT seems to be the best one, which somehow corresponds with the results presented in Figure 3b where SIFT is the most stable one (it was also mentioned on page 6). In the case of noise all the algorithms have a bad performance in the LWIR spectrum, being BRISK the less sensitive one. Finally, in the case of changes in scale the algorithm SIFT has a better behavior in LWIR domain than in VS, this can also be noted by comparing curves in Figure 4, where SIFT (followed by FREAK) shows a quite robust behavior with respect to changes in scale in the LWIR spectrum.

## 5. Conclusions

This work presents an empirical evaluation of the performance of the state of the art descriptors when they are used in the LWIR domain and compared against the results from those obtained in the visible spectrum. Although it is difficult to make a conclusion about which is the best feature detector and descriptor algorithm, since it would depend on different factors and according to the Table 2 there is not a clear winner, we can say that SIFT is among the best ones, showing good performance in most of the experiments. As a future work we will explore the usage of image preprocessing to enhance LWIR before feature detection and description. Due to the nature of LWIR images, and according with recent works, it seems that results could be improved with some image preprocessing.

**Table 2.** Average Recall Difference for the algorithms evaluated with the framework presented in Section 3 (bold values correspond to the algorithm that has the best relative performance in LWIR for the tested transformation).

<i>ARD</i>	<b>BRISK</b>	<b>ORB</b>	<b>SIFT</b>	<b>BRIEF</b>	<b>FREAK</b>	<b>SURF</b>
Blur	0.0442	-0.1323	0.1064	0.1149	0.0904	0.1425
Rotation	0.0450	-0.0762	0.0726	0.0109	0.0584	0.0013
Noise	-0.0427	-0.2921	-0.0764	-0.1266	-0.1273	-0.1106
Scale	0.0598	-0.0250	0.1564	0.0853	0.1271	0.1126

## Acknowledgments

This work has been partially supported by the Spanish Government under Research Project TIN2011-25606 and PROMETEO Project of the “Secretaría Nacional de Educación Superior, Ciencia, Tecnología e Innovación de la República del Ecuador”. Cristhian A. Aguilera-Carrasco was supported by a grant from “Universitat Autònoma de Barcelona”. The authors would like to thanks to Ievgen Khvedchenia for providing them with the evaluation framework.

## Author Contributions

The work presented here was carried out in collaboration between all authors. A.S. defined the research topic. P.R. and C.C. carried out the experiments and interpreted the results. C.A. provided the data set. A.S. wrote the paper. P.R., C.C., C.A. and B.X.V. reviewed and edited the manuscript. All authors read and approved the manuscript.

## Conflicts of Interest

The authors declare no conflict of interest.

## References

1. Gerónimo, D.; López, A.; Sappa A.D. Computer Vision Approaches to Pedestrian Detection: Visible Spectrum Survey. In Proceedings of the 3rd. Iberian Conference on Pattern Recognition and Image Analysis, Girona, Spain, 6–8 June 2007; pp. 547–554.
2. Gerónimo, D.; Sappa, A.D.; López, A.; Ponsa, D. Pedestrian Detection using Adaboost Learning of Features and Vehicle Pitch Estimation. In Proceedings of the International Conference on Visualization, Imaging, and Image Processing, Palma de Mallorca, Spain, 28–30 August 2006.
3. Coyle, S.; Ward, T.; Markham, C.; McDarby, G. On the suitability of near-infrared (NIR) systems for next-generation braincomputer interfaces. *Physiol. Meas.* **2004**, *25*, 815–822.
4. Hansen, M.P.; Malchow, D.S. Overview of SWIR Detectors, Cameras, and Applications. In Proceedings of the SPIE 6939, Thermosense, Orlando, FL, USA, 16–20 March 2008.
5. Krotosky, S.; Trivedi, M. On color-, infrared-, and multimodal-stereo approaches to pedestrian detection. *IEEE Trans. Intell. Transp. Syst.* **2007**, *8*, 619–629.

6. Krotosky, S.; Trivedi, M. Person surveillance using visual and infrared imagery. *IEEE Trans. Circuits Syst. Video Technol.* **2008**, *18*, 1096–1105.
7. Aguilera, C.; Barrera, F.; Lumbreras, F.; Sappa, A.D.; Toledo, R. Multispectral image feature points. *Sensors* **2012**, *12*, 12661–12672.
8. Barrera, F.; Lumbreras, F.; Sappa, A.D. Multimodal stereo vision system: 3d data extraction and algorithm evaluation. *IEEE J. Sel. Top. Signal Process.* **2012**, *6*, 437–446.
9. Barrera, F.; Lumbreras, F.; Sappa, A.D. Multispectral piecewise planar stereo using Manhattan world assumption. *Pattern Recognit. Lett.* **2013**, *34*, 52–61.
10. Magnabosco, M.; Breckon, T. Cross-spectral visual simultaneous localization and mapping (SLAM) with sensor handover. *Robot. Auton. Syst.* **2013**, *61*, 195–208.
11. Jung, S.; Eledath, J.; Johansson, S.; Mathevon, V. Egomotion Estimation in Monocular Infrared Image Sequence for Night Vision Applications. In Proceedings of IEEE Workshop on Applications of Computer Vision, Austin, TX, USA, 21–22 February 2007.
12. Laliberte, A.; Goforth, M.; Steele, C.; Rango, A. Multispectral remote sensing from unmanned aircraft: Image processing workflows and applications for rangeland environments. *Remote Sens.* **2011**, *3*, 2529–2551.
13. Miksik, O.; Mikolajczyk, K. Evaluation of Local Detectors and Descriptors for Fast Feature Matching. In Proceedings of the 21st International Conference on Pattern Recognition, Tsukuba, Japan, 11–15 November 2012; pp. 2681–2684.
14. Mikolajczyk, K., Schmid, C. A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1615–1630.
15. Bauer, J.; Snderhauf, N.; Protzel, P. Comparing Several Implementations of Two Recently Published Feature Detectors. In Proceedings of the International Conference on Intelligent and Autonomous Systems, Toulouse, France, 3–5 September 2007.
16. Schmid, C.; Mohr, R.; Bauckhage, C. Evaluation of interest point detectors. *Int. J. Comput. Vis.* **2000**, *37*, 151–172.
17. Lowe, D.G. Object Recognition from Local Scale Invariant Features. In Proceedings of the IEEE International Conference on Computer Vision, Kerkyra, Greece, 20–27 September 1999; pp. 1150–1157.
18. Bay, H.; Tuytelaars, T.; Gool, L.J. SURF: Speeded up Robust Features. In Proceedings of the 9th European Conference on Computer Vision, Graz, Austria, 7–13 May 2006; pp. 404–417.
19. Rublee, E.; Rabaud, V.; Konolige, K.; Bradski, G.R. ORB: An Efficient Alternative to SIFT or SURF. In Proceedings of the IEEE International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 2564–2571.
20. Leutenegger, S.; Chli, M.; Siegwart, R. BRISK: Binary Robust Invariant Scalable Keypoints. In Proceedings of the IEEE International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 2548–2555.
21. Calonder, M.; Lepetit, V.; Özuysal, M.; Trzcinski, T.; Strecha, C.; Fua, P. BRIEF: Computing a local binary descriptor very fast. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 1281–1298.



22. Alahi, A.; Ortiz, R.; Vandergheynst, P. FREAK: Fast retina keypoint. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 510–517.
23. Feature Descriptor Comparison Report. Available online: <http://computer-vision-talks.com/articles/2011-08-19-feature-descriptor-comparison-report/> (accessed on 18 February 2014).
24. Morris, N.; Avidan, S.; Matusik, W.; Pfister, H. Statistics of Infrared Images. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA, 18–23 June 2007.
25. Tendero, Y.; Gilles, J. ADMIRE: A Locally Adaptive Single Image Non-Uniformity Correction and Denoising Algorithm. Application to Uncooled IR Camera. In Proceedings of the SPIE Defense, Security and Sensing, Baltimore, MD, USA, 23–27 April 2012.

MDPI AG

Klybeckstrasse 64

4057 Basel, Switzerland

Tel. +41 61 683 77 34

Fax +41 61 302 89 18

<http://www.mdpi.com/>

*Sensors* Editorial Office

E-mail: [sensors@mdpi.com](mailto:sensors@mdpi.com)

<http://www.mdpi.com/journal/sensors>





MDPI • Basel • Beijing • Wuhan • Barcelona  
ISBN 978-3-03842-053-8  
[www.mdpi.com](http://www.mdpi.com)

