

Article

Stego-STFAN: A Novel Neural Network for Video Steganography

Guilherme Fay Vergara ^{*,†} , Pedro Giacomelli [†] , André Luiz Marques Serrano [†] ,
Fábio Lúcio Lopes de Mendonça [†] , Gabriel Arquelau Pimenta Rodrigues [†] , Guilherme Dantas Bispo [†] ,
Vinícius Pereira Gonçalves [†] , Robson de Oliveira Albuquerque ^{*,†}  and Rafael Timóteo de Sousa Júnior [†] 

Department of Electrical Engineering, University of Brasília, Federal District, Brasília 70910-900, Brazil;
pedro.giacomelli2@gmail.com (P.G.); andrelms@unb.br (A.L.M.S.); fabio.mendonca@redes.unb.br (F.L.L.d.M.);
gabriel.arquelau@redes.unb.br (G.A.P.R.); guilherme.bispo@redes.unb.br (G.D.B.); vpgvinicius@unb.br (V.P.G.);
desousa@unb.br (R.T.d.S.J.)

* Correspondence: guilherme.vergara@redes.unb.br (G.F.V.); robson@redes.unb.br (R.d.O.A.)

[†] These authors contributed equally to this work.

Abstract: This article presents an innovative approach to video steganography called Stego-STFAN, as by using a cheap model process to use the temporal and spatial domains together, they end up presenting fine adjustments in each frame, the Stego-STFAN had a $PSNR_c$ metric of 27.03 and $PSNR_S$ of 23.09, which is close to the state-of-art. Steganography is the ability to hide a message so that third parties cannot perceive communication between them. Thus, one of the precautions in steganography is the size of the message you want to hide, as the security of the message is inversely proportional to its size. Inspired by this principle, video steganography appears to expand channels further and incorporate data into a message. To improve the construction of better stego-frames and recovered secrets, we propose a new architecture for video steganography derived from the Spatial-Temporal Adaptive Filter Network (STFAN) in conjunction with the Attention mechanism, which together generates filters and maps dynamic frames to increase the efficiency and effectiveness of frame processing, exploiting the redundancy present in the temporal dimension of the video, as well as fine details such as edges, fast-moving pixels and the context of secret and cover frames and by using the DWT method as another feature extraction level, having the same characteristics as when applied to an image file.

Keywords: video steganography; dynamic neural network; attention; DWT; deep learning; data security



Citation: Vergara, G.F.; Giacomelli, P.; Serrano, A.L.M.; Mendonça, F.L.L.d.; Rodrigues, G.A.P.; Bispo, G.D.; Gonçalves, V.P.; Albuquerque, R.d.O.; Sousa Júnior, R.T.d. Stego-STFAN: A Novel Neural Network for Video Steganography. *Computers* **2024**, *13*, 180. <https://doi.org/10.3390/computers13070180>

Academic Editor: Xiaochen Lu

Received: 21 June 2024

Revised: 12 July 2024

Accepted: 16 July 2024

Published: 19 July 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the increased number of people connected to the network, there has been a growing demand for online services, such as electronic commerce and e-government [1]. Therefore, the need for confidentiality in information exchanged between companies and users has increased. Unfortunately, the number of frauds and attacks on organizations, compromising the privacy of the users, also increased [2]. Furthermore, projects developed by the USA aimed to monitor all electronic content in traffic on the network, intending to filter suspicious messages from possible terrorists. These projects use cutting-edge technology to filter information exchanged via email, FTP, and HTTP content.

In addition to these facts, electronic commerce had its expansion boosted by the proliferation of mobile devices, such as cell phones, and there has also been an increase in the distribution of confidential multimedia material over the network. These facts highlight the need to create mechanisms that introduce due secrecy and access control to this information. It is important to raise the main aspects involved in the safe distribution of any material, especially multimedia, to guarantee digital security and prepare the market for growth in the area in the coming years. For such a level of security, it is necessary to create mechanisms that implement methods that add levels of Authenticity and Integrity to multimedia material. Authenticity provides the correct answer for identifying

system users, that is, the origin or destination are correctly identified. Integrity consists of protecting modifications without the explicit permission of the owner of the information. Modification includes writing, content changes, status changes, removal, creation, and delay of transmitted data.

In this way, the branch of computer science and mathematics focused on studying digital protection mechanisms called digital security, which uses efficient encryption and steganography algorithms. Cryptography aims to protect data by encoding or encrypting information, making it incomprehensible. Digital steganography seeks to hide information in various types of media, whether continuous or discrete, such as text, images, sounds, and videos (the object of this research). Therefore, the literature reports works that use steganography in images, such as [3]. However, the generalization of these techniques for use in digital videos has not been explored enough. The benefit of this generalization is related to the increase in the amount of information that can be hidden. Thus, creating a steganography mechanism in digital videos increases the amount of information inserted and provides privacy during the transmission of confidential data.

Video steganography is hiding secret information within video sequences to enhance data protection. Over recent years, the field has witnessed a proliferation of methods to improve covert communication [4]. The inherent characteristics of video sequences, including their high capacity and complex structure, render them a preferred choice for concealing information compared to other media types. One key aspect of video steganography is concealment in Video Frames, which involves hiding information within a video's frames. Each frame embeds bits of hidden data without causing noticeable changes to the visual or auditory quality [5].

Video steganography involves different techniques, including modifying the color values of pixels in the video frames, altering the least significant bits of pixel values, or manipulating the audio track's frequency components. LSB modification, a common technique, replaces the least significant bits of pixel values in the video frames with hidden data.

Video steganography, which is vital in fields like medical systems and law enforcement, employs various techniques to embed messages within video content. These techniques include modifying pixel color values, altering the least significant bits (LSB) of pixel values, or manipulating the audio track's frequency components [6]. Since the subtle changes affect only the least significant bits, they are less likely to be noticeable to human observers. Video steganography can operate in both spatial and temporal domains. Spatial domain techniques involve manipulating individual frames, while temporal domain techniques may include changes over time, such as modifying the timing or order of frames. In frequency domain steganography, changes occur to the frequency components of the audio or video signals. This method may involve hiding information in specific frequency bands or altering the phase of certain elements.

Three crucial factors in successful steganography are imperceptibility, robustness, and embedding capacity [6]. Imperceptibility ensures the hidden message is not visually detectable. Robustness measures the method's resilience against attacks. Embedding capacity defines the amount of data that can be securely embedded within the digital video.

The effectiveness of video steganography relies on the ability to hide information without raising suspicion. Detecting hidden information in videos poses a challenge, requiring specialized tools and techniques for analysis. Security concerns also arise when videos are shared or transmitted, as someone could expose hidden information if the video undergoes compression or other transformations.

Building on the studies in video steganography, there is a growing need for more sophisticated methods that seamlessly integrate with modern video processing techniques. This paper proposes a novel architecture that efficiently combines the Spatial-Temporal Filter Adaptive Network (STFAN) [7] and the attention mechanism, aiming to exploit temporal redundancy and spatial features in video steganography. By leveraging STFAN's ability to generate spatially adaptive filters for alignment and deblurring dynamically, the proposed method enhances the concealment and robustness of hidden information

within video sequences. Integrating attention mechanisms further refines the process by focusing on critical regions, improving video steganography's effectiveness and security. In addition, this model may enable control over the processes of recovering the secret by different users using the dynamic filters as a form of key. Despite these novel contributions, our model has the limitation of not providing state-of-the-art (SOTA) quality outputs both in Container and Restored Secret. Moreover, we could not prove the assumptions about the model's abilities because the output requires further improvements before we can validate these statements.

This paper is structured as follows. Section 2 reviews the literature regarding steganography and some adopted methods. Section 3 presents the methodology adopted in this work. Section 4 exposes the obtained results, which are discussed in Section 5. Ultimately, Section 6 concludes the paper.

2. Related Work

According to [8], a systematic review has three fundamental steps for its execution. The first step of the research consists of verifying the concept to be researched, enabling the creation of a search string. This string has the function of providing evidence about the topic of interest to be explored. After a search string has been defined, it is necessary to start searching for articles by defining the search methods used for this task. In the second step, an analysis and classification of articles found in the previous phase is done to determine which ones will be removed from the research. Also, at this stage, a form is filled out with data of interest and analysis for each article found. Finally, in the third step, we analyze the articles from the previous phase and look for ways to synthesize the data collected, concluding on the research results.

In this section, we review the existing literature related to video steganography and neural networks. We conducted a comprehensive search using Scopus and Web of Science databases with the query ((TI = (Video Steganography) AND TI = (neural Network)) OR ((AB = (Video Steganography) AND AB = (neural Network))). This search yielded 127 articles. Following a meticulous screening process, 26 articles were selected for further analysis.

Various methodologies in video steganography have been proposed to bolster message concealment while ensuring resilience against detection. One notable strategy involves concealing secret data within the movements of objects within a video instead of embedding them solely in the background [9]. Additionally, reversible pipelines and invertible neural networks have been harnessed to enable the hiding and recovery of data across multiple videos, thereby enhancing both hiding capacity and flexibility [10]. Furthermore, techniques such as utilizing the least significant bits (LSB) in the raw domain and employing transform domain-based methods like discrete wavelet transform have been examined [4]. Moreover, there has been a surge in interest in end-to-end video steganography schemes, leveraging technologies such as Generative Adversarial Networks (GANs) and multi-scale deep learning networks, which have demonstrated remarkable visual quality, substantial embedding capacity, and resistance to video compression [11].

Another promising method is neural networks, which could be used to solve complex problems and generate concealments that are not traditional and could be difficult to detect, as explained by Gómez [12].

Early models, such as SRNET [13] and StegExpose [14], have laid the groundwork for video steganography but face notable constraints. SRNET, while effective in embedding, struggles with balancing the trade-off between embedding capacity and imperceptibility. High embedding capacities can lead to noticeable distortions in the cover video, making detection easier. StegExpose, designed for detecting steganographic content, highlights the vulnerabilities of existing methods by effectively identifying hidden messages, thus pushing for more sophisticated techniques.

Then, the first notable deep learning-based video steganographic network was VSteg-net [15], which addresses these challenges by leveraging the UCF 101 action recognition video dataset for training, which improves the robustness and imperceptibility of the

steganographic content. By optimizing the compression of the cover video and employing a GAN to divide it into frames, VStegnet enhances the embedding capacity while maintaining high quality.

Although VStegnet aims to enhance security through techniques like Arnold scrambling and encryption, it must continuously adapt to evolving detection and attack methods. Adversaries are constantly developing new ways to detect and disrupt steganographic content, requiring ongoing improvements to maintain security.

The interplay between steganography and steganalysis resembles a cat-and-mouse game, with advancements in one driving innovations in the other. As techniques to conceal information become more sophisticated, detection methods also evolve, prompting continuous development in both fields. This ongoing battle highlights the necessity of continuing research in steganography to stay ahead of increasingly advanced steganalysis techniques. So this work aims to integrate STFAN with attention mechanisms to potentially improve video steganography.

To better understand the Steganography process, Figure 1 explains the overall workflow of the steganography and steganalysis from Alice, Bob, and Eve's perspectives.

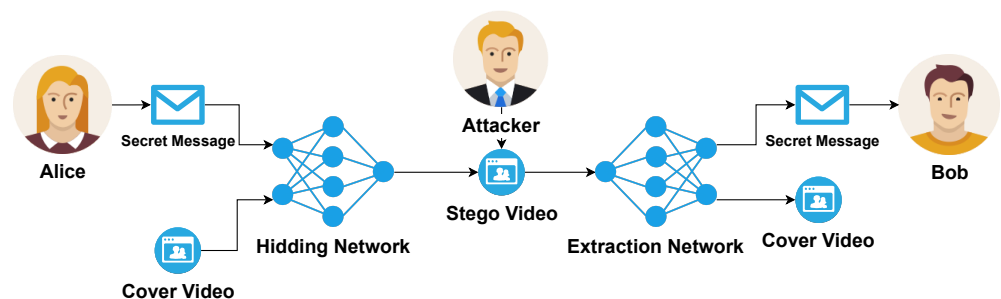


Figure 1. Steganography reduced to the prisoner's problem. Adapted from [4].

In Figure 1, we can see the encrypted message, which could be text, audio, or even another video from Alice then this message is steganographed using a cover video. This video is now sent to Bob, but the attacker is performing an attack in the network and intercepting the message, in this scenario, the attacker doesn't even know that the video has an encrypted message, and even the steganography method used to get the message. Finally, Bob can decode the message using the same steganography method and the secret key.

2.1. Attention

Attention mechanism, as the name suggests, helps the Network to pay more attention to some regions of the input. Formerly, an attention mechanism was developed for Natural Language Processing (NLP), which aims to process human language of any type. The machine needs to ensure that the specificities of the analyzed language are respected so that the interpretation occurs appropriately. Suppose a computer program takes a certain text in natural language exactly literally and ignores such nuances. In that case, it is possible that the written (or spoken) content will not be understood or well interpreted. There is also the issue of ambiguity; human languages change over time and are influenced by different means, such as contact with other languages or the advancement of the internet. For these reasons, NLP needs the AI set to function correctly and thus help the RNN [16,17] to better deal both with the length [18] and content of the text [19] being analyzed.

The most noticed mechanism developed was the Transformers Mechanism [19] introduced in 2017. With the development of the technique, the attention concept was transposed to images, making it possible to map the most critical areas of an image as a consequence of the attention. Next, the two attention strategies used in this article will be presented.

2.1.1. Non-Local Attention

First introduced a non-local Algorithm for the image [20] as a form of denoise the image which applies a weight to the average of a particular pixel's value according to all its neighbors in grey scale through a Gaussian kernel. Hence, the Algorithm can capture anomalies (peaks) in the image, configuring it as noise. Furthermore, in the middle of the calculation, the distance between two pixels is considered, which gives the advantage that edges will not be affected, given that an edge is, in simple terms, an abrupt change in the pixel value, but following the pattern that neighbor pixels will also differ, being possible to differentiate it from random noise.

With the development of the technique, ref. [21] adapted it to be used as an attention mechanism. Once averaging the values of a pixel, simplicity is the same as comparing its contextual meaning. Moreover, as a consequence of the relation between the pixel and its neighbors, when the temporal aspect, as done in [21], not only is it possible to capture the spatial features of the image but also its temporal importance, i.e., pixels that change its value will have different weights from its statical neighbors.

These characteristics are essential for video since they enable highlighting a specific area of the video that is important to the network, both in the spatial and temporal dimensions. In addition, ref. [21] demonstrated that the Algorithm also works with Dot Product instead of Gaussian Kernels (and its embedded variation). In other terms, ref. [21] developed a self-attention [19,22] mechanism that takes the spatial-time relation to the feature map.

Therefore, this layer works in conjunction with the Non-local algorithm. The attention is applied to the frames, not with a single task (to find the most relevant regions) but to guide how the regions highlighted by the Non-local algorithm influence the frames as a whole.

2.1.2. Convolutional Block Attention Module—CBAM

Proposed by [23], CBAM, which stands for “Convolutional Block Attention Module”, was created to serve as a cheap and efficient attention block to capture important features in the spatial dimension and between channels [24,25].

To do this, the Algorithm is divided into two parts: the first makes the channel's feature map, and the second maps the spatial attention of each channel. The channel part receives the input, makes both an average and a max pooling in parallel, and passes it in a shared MLP. Next, the result of the latter is summed and inserted into a Sigmoid, which is multiplied with the skip connection of the input. Then, the result is used as input of a convolutional layer (f) with max pooling followed by an average pooling, which passes in a Sigmoid and is multiplied again with the original input.

The equation for each part of the Attention model is the following:

$$F_{channel} = \sigma(MLP(AvgPool(X)) + MLP(MaxPool(X)))$$

$$F_{spatial} = \sigma(f([AvgPool(X); MaxPool(X)]))$$

The CBAM equation results in F'' . F is the input, and F' denotes the Attention Module's first part (channel) output.

$$F' = F_{channel}(F) \otimes F$$

$$F'' = F_{spatial}(F') \otimes F'$$

2.2. DWT

Discrete Wavelet Transform (DWT) is a technique to extract frequency bandwidths from a signal. When using it for images (through 2D-DWT), the frequencies extracted represent different information of the images, such as edges, texture, details, and noise [26]. The extract frequencies are divided into four groups: Low-Low (LL), Low-High (LH), High-

Low (HL), and High-High (HH). The former contains the most expressive part of the image, whereas the latter contains fine details and noise. The middle ones focus on extracting edges with horizontal and vertical directions. Due to these transformation characteristics, it was already used extensively for steganography [26–29].

3. Methodology

Posteriorly, the methodology was divided into five parts. First, a general description of the proposed architecture was done. Then, an explanation of STFAN was written to justify the motivation of such an algorithm in the presented architecture. Next, the motivation to use attention blocks (NLSA, NLCA, and CBAM) was given in detail. The next part briefly explains the DWT and its usage in the Network, followed by a subsection that describes the loss function used. Then, final considerations regarding the training and implementations were made.

3.1. General Architecture

The general architecture of Stego-STFAN consists of three main parts, which are, respectively, responsible for embedding, hiding, and restoring the frames.

In the embedding part, as seen in Figure 2, we create the stego frame (B_t) by converting the secret and cover to YUV, then a second-order extraction of all subbands to select the secret's most valuable features, after this, we embed the frames in Low-High (LH) and High-Low (HL), and at the end we perform the IDWT on Y and convert the frame back to RGB.

The second (hiding) part has five inputs as can be seen in the Figure 3: Non-local Self Attention (NLSA), Non-local Co-Attention (NLCA), previous container frame (HR_{t-1}), last frame (HB_{t-1}), and current stego-frame from DWT (HB_t). In the second (restoring) part, there are four inputs: container, previously predicted secret (RR_{t-1}), previous frame (RB_{t-1}), and current container (RB_t).

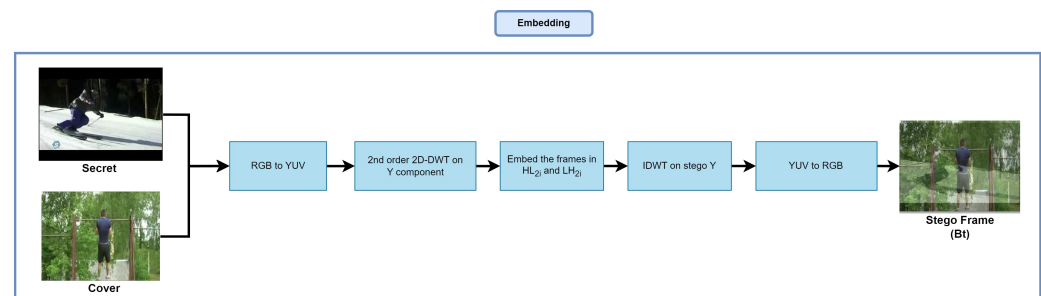


Figure 2. DWT embedding.

In this part of the Network, NLSA, and NLCA go into the extraction features block, which is responsible for learning the proper way to hide the secret inside the cover. The attention mechanism, given by the Non-local technique, yields a map that improves the learning of hard features like edges, fast-moving pixels (Latent Frame frames), and contextual characteristics of the frames. The second part of the hiding block focuses on building the adaptive filters, which improve the correlation between the spatial and temporal aspects of the frames and adjust the fine details according to the secret frame. In this part, there are 3 inputs: HR_{t-1} , HB_{t-1} , and HB_t . Last, the reconstruction part of the Network derives from the convolution (through FAC layers) between the previous two parts of the hiding block.

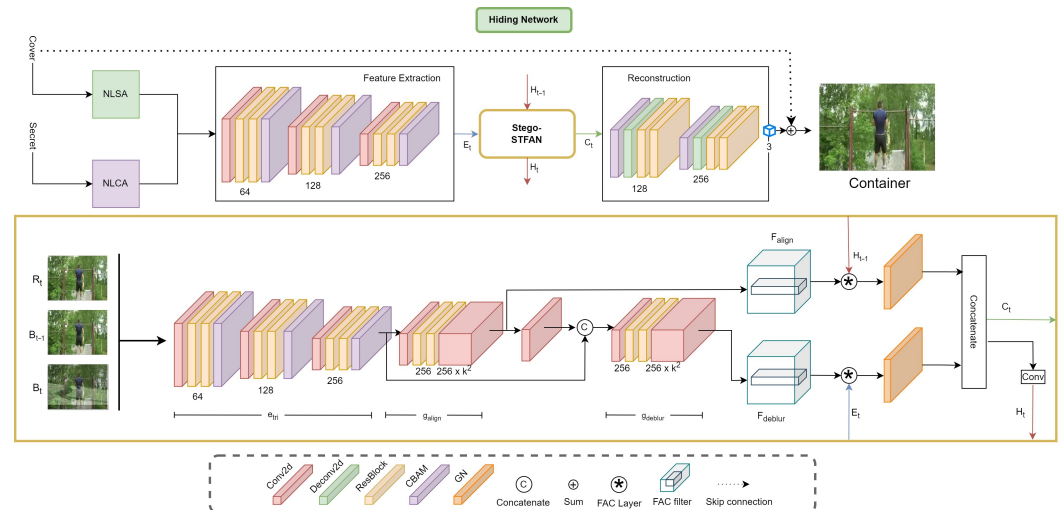


Figure 3. Hiding network.

In the restoring part of the Network (Figure 4), the container goes into the extraction features block, which is responsible for learning how to recover the secret message. The second part of the hiding block also utilizes the adaptive filters, which improve the correlation between the spatial and temporal aspects of the frames and adjust the fine details to restore the frames of the secret message better. This part has three inputs: RR_{t-1} , RB_{t-1} , and RB_t . Last, the reconstruction part of the Network derives from the convolution (through FAC layers) between the previous two parts of the restoring block.

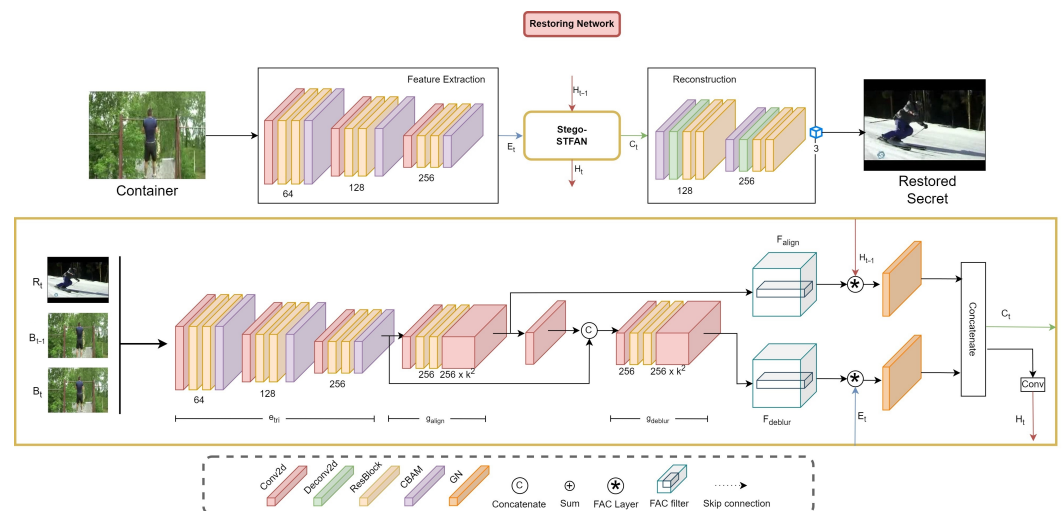


Figure 4. Restoring network.

3.2. Spatial-Temporal Filter Adaptive Network—STFAN

The proposed architecture was based in [7], which developed the STFAN algorithm for video deblurring. We adapted this network for steganography because it can dynamically process both spatial and temporal data simultaneously [30] for each frame, enabling a fine detail concealing of the message into the cover frame and restoring it. Also, the recurrent strategy design in STFAN allows an efficient way to insert more temporal samples (previous frames) cheaply, making it worthwhile to put more effort into the temporal aspect without sacrificing processing.

As mentioned before, the hiding part receives the previous container frame (HR_{t-1}), the last frame (HB_{t-1}), and the current stego-frame from DWT (HB_t). Due to the capacity of STFAN through the Filter Adaptive Convolutional (FAC) layer to dynamically mingle the information between frames, specific modifications for each frame can be done, which enables a more accurate container. The adaptations that it does come from the Concealing Filter Generator (g_{conc}) and Alignment Filter Generator (g_{align}). g_{conc} is yielded by joining HR_{t-1} and HB_t , which gives features of how to conceal the image through its spatial information better, but also making the current container (that is being built) more harmonic with the previous one, making more challenging to be steganalysis. Moreover, g_{align} is constructed by joining HB_{t-1} and HB_t that enables the extraction of temporal features to understand the alignment of the frames better and explore it to build an improved container with less redundant information and utilizing the dynamics portrayed in the video for valuable modifications. g_{conc} and g_{align} explore each frame and spatial characteristics of early and effective since the prediction will be dynamically updated to fit each frame.

Following the same reasoning, the restoring part receives the previously predicted secret (RR_{t-1}), the previous frame (RB_{t-1}), and the current container (RB_t). Since STFAN enables dynamic mingling of the information between frames, specific modifications for each frame can be done to build more reliable reconstructed secrets. As said before, g_{conc} considers the spatial information to hide the secret better. Similarly, Restoring Filter Generator (g_{rec}) is produced by joining RR_{t-1} and RB_t , which gives features of what the restored secret should be, making it easier to the Network exploring specific spatial information in the current frame, and also to build a more harmonic video based on the previous prediction. Adapting the g_{align} to the restoring part, this filter is generated by joining RB_{t-1} and RB_t that enables the extraction of temporal features, making it easier to spot the differences between the frames and guiding how to recover the secret more efficiently based on the previous frame.

Due to the characteristics shown and the recurrent and sample-specific aspects of STFAN, a cheap network that takes the temporal dimension is formed and predicts a new filter for each frame, enabling more accurate adaptation for concealing and recovering the message. Furthermore, since it can deal with blur more efficiently, it is expected to perform better in fast-moving videos and use the blur to hide the message better. Therefore, the proposed network differs from the models hitherto designed in how the spatial and temporal dimensions are processed. Usually, Networks designed for video steganography use 3D Convolutions, which are extremely expensive and may not accurately extract the important information and redundancy in the video as a network design with alignment and deblurring (together with attention mechanism) as the proposed network. In addition, with the dynamic filter, frame-specific modifications are easier to perform, which in other models may not be feasible due to restrictions from the architecture.

3.3. Attention

In computer vision, the attention mechanism allows us to handle different regions of an input image differently. When images are input into a standard convolutional neural network (CNN), the resulting feature map often contains redundant information across different channels [31]. Traditional CNNs treat each point on the feature map equally, but in reality, the importance of features varies across different channels and locations. Some features are crucial, and others are not.

3.3.1. Non-Local Attention

Stego-STFAN adopts attention mechanisms to enable the message hiding into video carriers with an improved performance. Since edges and fast pixels are essential to steganography, especially video steganography, more efforts must be made to embed a secret in these regions and not be detected by steganalysers.

Following this reasoning, ref. [32], through the Non-local Self-Attention (NLSA) block, adapted the Algorithm above to map the features of the context, edges, and fast-moving pixels of the cover video in a more explicit way to the Network.

Furthermore, ref. [32] adapted it to not only capture the relevant parts of the cover itself but also to capture the main spatial-time features of the secret video, i.e., detect the most difficult parts of the secret video to be hidden in the cover and highlight it to the Network give more importance to these areas. To perform the latter, ref. [32] changed the Algorithm of [21] to receive the secret as an input (check Figure 5b for more details).

To perform NLSA (Figure 5), the Algorithm is divided into three basic steps. First, the input is passed through 3 independent residual blocks (for NLCA, the cover passes in only one, and the others belong to secret video). Second, two outputs from the previous blocks perform a matrix multiplication, followed by a softmax activation. Adapting it to NLCA, the output from the cover residual block multiplies one output of the other two residual blocks. Finally, a matrix multiplication between the softmax output and the last residual block is performed, followed by an upsample (due to strides 2 in residual blocks) and a 1×1 convolution layer, which has its output added with the input of the attention block. Making it suitable for NLCA, the skip connection at the end of the block is the secret video.

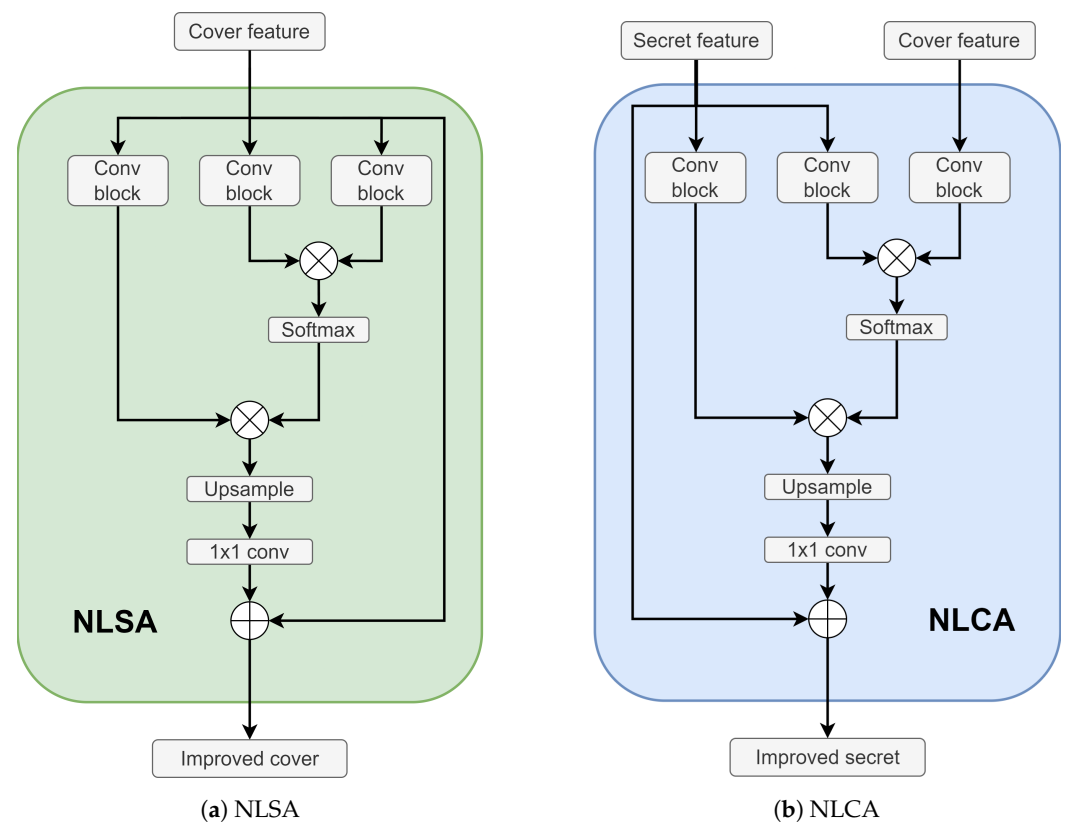


Figure 5. Non-Local Attention.

3.3.2. Convolutional Block Attention Module—CBAM

Transposing it to our Network, CBAM, as seen in the Figure 6 helps to (1) get the most important spatial information of the filters used in convolutional blocks, (2) highlight the channels' region (filters' region) that have more relation with each other, (3), since the time is passed in form of channel through different frames, it also captures the relevant information between the frames (of cover and secret) and makes it easier to the Network explore the inter-frame correlation.

To achieve this, sequentially apply channel and spatial attention, as seen in the Figure 6, so that each of the branches can learn ‘what’ and ‘where’ to attend in the channel and spatial axes, respectively. As a result, our module efficiently helps the information flow within the network by learning which information to emphasize or suppress. Studies were then conducted to quantitatively evaluate improvements in model interpretability. Better performance and better interpretability are possible at the same time using CBAM. Taking this into account, it is conjectured that the performance increase comes from accurate attention and noise reduction from irrelevant clutter. Finally, we validate the performance-improved object detection in datasets demonstrating the broad applicability of CBAM. As designed, if the module is lightweight, the parameter and computation overhead is insignificant in most cases. It is worth mentioning that no changes were made in CBAM architecture.

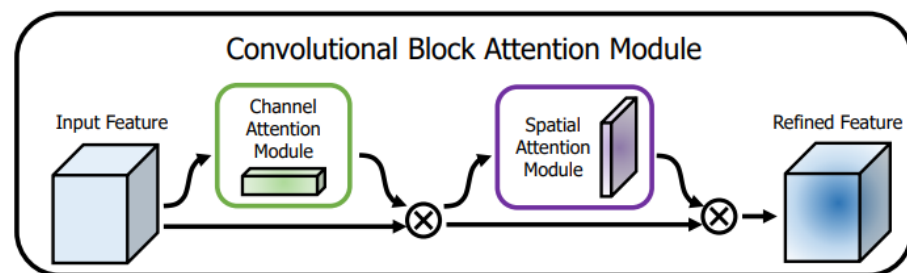


Figure 6. CBAM.

3.4. Group Normalization Block

Following the strategy of [32], a Group Normalization [33] was introduced to normalize the feature, avoiding problems in the convergence and numerical stability. This block differs from Batch Normalization (BN) in how normalization is done. While BN normalizes over the batch dimension, this layer uses the individual sample divided into areas (groups) as a reference. By doing that, the number of batches has a significant influence, as in BN, where small batch sizes have big detrimental effects on the training. Since BN layers can only be used in big batch sizes, Group Normalization was an excellent option for stabilizing the frames' values, especially after FAC blocks. In our model, 4 groups were chosen.

3.5. DWT

For video steganography, it is desirable to have good embedding and the embedded message in the frames imperceptible. For this reason, Haar-DWT helps detect the good parts of the frame and gives it special attention in processing. Following the work of [27,28,34], the embedding was done in the second-order extraction of all subbands to select the secret's most valuable features and not deteriorate the cover image. In the current Network, the stego-image (based on the fusion of the cover and secret DWT) was passed to the STFAN layer to help the Network identify the most relevant details and regions of the frame to be steganographed.

Considering the fact that human eyes perceive pixel perturbation differently in different video areas, we explore suitable coverage areas for hiding and corresponding hidden areas for attention by modeling spatiotemporal correlation in video steganography. To find suitable coverage areas for concealment, a Non-Local Self-Attention (NLSA) block was proposed to improve the representation of coverage features by calculating correlations between positions in spatial and temporal dimensions within coverage frames. The method models spatiotemporal correlation by leveraging information from multiple reference frames. The residue between the corresponding lid and the container frame shows that the approach changes the value of the pixels in the motion boundary areas. To focus on significant areas of the covert video, detailed representations of covert frames were developed based on the calculation of correlated items between cover frames and covert frames by a non-local co-attention (NLCA) block.

Because it is a cheap operation and serves as another “attention” mechanism, passing it through STFAN and not through the other attention mechanisms proposed here, Non-Local Self-Attention and Non-Local Co-Attention (NLCA and NLSA). was presented, as it would require the insertion of more dimensions, and having another perspective of the framework (through DWT) is also beneficial for the Network and its convergence. DWT was already used with deep learning [WaveNet, compression networks], specifically for steganography [35] for similar purposes.

In addition, following the work in [27,28,34], it is clear that embedding in the Y component is an efficient and cheap way of hiding the secret. Hence, the same was done in this work. Instead of doing a DWT on each channel, the frames were transformed from RGB to YUV, and then the Y component was selected as the embedding channel.

3.6. Loss Function

To develop a good video steganographic technique, the stego-image and the recovered secret should be of good quality. Therefore, the loss function has to consider these two outcomes of the Network. Following the same strategy from [32], our loss function sums the MSE from both the stego-image and the secret image, giving a β weight to the secret recovered. The same β value (0.75) used in [32] was used here for simplicity.

$$L_c = \text{MSE}(\text{Cover} - \text{Container})$$

$$L_s = \text{MSE}(\text{Secret} - \text{Restored Secret})$$

Moreover, it is also expected that the model does not produce an unbalanced loss to favor one loss over the other. For this reason, it was also added a loss to control this difference and help the model to converge:

$$L_a = \text{Abs}(L_c - L_s) \cdot \text{Abs}(\text{PSNR}_c - \text{PSNR}_s) \cdot \frac{1}{\alpha}$$

where α is a hiper-parameter and was set to 3 a value too high reduces the penalization from deviation in PSNR from cover and secret, and too small puts more effort on making them similar instead of learning to complement each other. Therefore, the final loss is as follows:

$$L_{\text{loss}} = L_c + \beta \cdot L_s + L_a$$

3.7. Final Considerations

The Network was implemented in PyTorch and trained in Google Colab using L4 GPU (22Gb) and high RAM (53 Gb). The Convolutional block is a residual layer composed of a 3×3 convolution stride = 1 + Leaky ReLU ($\alpha = 0.01$) + 3×3 convolution stride = 1. Then, the sum of the input with the last convolution output should be performed, followed by another Leaky ReLU ($\alpha = 0.01$). In total, 5700 frames (144×180) were used from datasets UCF-101 and Weizmann Action, diving 80% for training and the rest for validation. Patience was set to 5 since it represented approximately 10% of the maximum quantity of epochs planned to train (60 epochs). Learning Rate Scheduler was used to adjust the learning for each phase of the training. Additionally, a noise injection ($\mathcal{N}(0, 0.015)$) was used as regularization and also to simulate variations that may occur in practical situations. The pseudocode could be seen in the Algorithm 1

Algorithm 1 Stego STFAN Training Algorithm

```

1: Initialize variables
2: hiding_network = STFAN_hiding()           ▷ Initialize the hiding network
3: restoring_network = STFAN_restoring()      ▷ Initialize the restoring network
4: patience = 5                               ▷ Patience for early stopping
5: best_loss = ∞                               ▷ Initialize best loss to infinity
6: for each epoch do
7:     epoch_loss = 0                           ▷ Initialize epoch loss
8:     lr_scheduler.step()                     ▷ Update learning rate scheduler
9:     for each batch do
10:        total_loss = 0                       ▷ Initialize total loss for the batch
11:        for each frame do
12:            Container, Restored_Secret, HHt-1, RHt-1 = my_df[current_frame]
13:            Container, HHt = hiding_network(HRt, HBt-1, HBt, HHt-1)
14:            Restored_Secret, RHtr = restoring_network(RRt, RBt-1, RBt, RHt-1)
15:            my_df[current_frame] = [Container, Restored_Secret, HHt, RHt]
16:            lossc = MSE(Cover, Container)      ▷ Compute loss for cover
17:            losss = MSE(Secret, Restored_Secret) ▷ Compute loss for secret
18:            lossa = abs(lossc - losss)      ▷ Compute absolute difference loss
19:            loss = lossc + β · losss + lossa    ▷ Compute total loss
20:            total_loss += loss                ▷ Accumulate total loss
21:        end for
22:        total_loss.backward()                ▷ Backpropagate total loss
23:        optimizer.step()                    ▷ Update model parameters
24:        epoch_loss += total_loss             ▷ Accumulate epoch loss
25:    end for
26:    if epoch_loss < best_loss then
27:        best_loss = epoch_loss               ▷ Update best loss
28:        patience = 5                         ▷ Reset patience
29:    else
30:        patience -= 1                       ▷ Decrement patience
31:    end if
32:    if patience == 0 then
33:        break                               ▷ Early stopping if patience is zero
34:    end if
35: end for
36: Finalize process                           ▷ Any final steps

```

4. Results

In this section, the behavior of the Network to different inputs will be evaluated. Also, some ablations studies are performed to test the assumptions made before (I) the preference for blurred regions [7,20,21,23,32], (II) improved quality of steganography derived from the blur [7,20,21,23,32], and (III) the leverage caused by the addition of more frames as input [36].

4.1. Network Training

During the training, the network sees very slow convergence. The literature for not GAN-like models proposes two losses: an MSE between the cover and the original frame and another MSE between the restored secret and the original secret frame (the same used in Section 3.6). However, due to the antagonism in the actions performed by the network (one tries to hide, and the other to recover the secret), it was seen that one loss was suppressing the other instead of working as complementary. Therefore, another loss was used to control this divergence between the two losses (L_c and L_s) by taking the absolute difference between L_c and L_s . It controlled the divergence of the two losses, leading to more balanced results between the container and restored frames.

Despite this improvement, the model converged even slower. In fact, the divergence between L_c and L_a almost disappeared, but the model struggled to produce a good output because it was putting more weight into making the container and restoring the secret with identical losses instead of truly hiding and restoring the secret, even if it implies on increasing the loss temporarily (until the model improves). Hence, the new loss L_a was proposed to overcome this difficulty, which was accomplished successfully.

During the first attempts of training a slow convergence was seen. To deal with it, the batch size was changed from 1 (as suggested in similar works) to 4, which gave a huge increase in performance and stability during training as well as another way of regularizing the network. In the first attempts, huge variations were seen, suggesting a divergence, what is clear that did not happen in the second model (with batch size = 4) due to more restrictions in how the gradients should be to increase the learning process (instead of just improving the model to each frame without generalization). From the PSNR graphs, it is seen the same effect: with batch size = 1, the values are changing for the same reasons explained before. Also, with batch size = 4, a stable PSNR was found and also with better results.

4.2. Model Evaluation

As a result of the training, low-quality frames, compared to state-of-the-art models, were obtained in the container in nearly all cases. Despite the good quality of the Container, residuals (black lines) from dynamic filters are seen, which greatly reduces the PSNR, and it still is human-detectable as a steganographed frame. Moreover, the Restored Secret considerably differs from the original secret having as a result blurred images, but similar to the original ones.

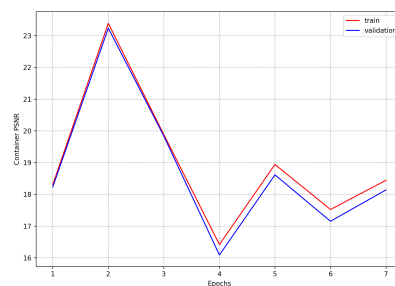
Table 1 compares the performance of our model with the current ones [15,32,37–39]. In terms of cost-benefit, our model is the second option since DRANet has fewer parameters and better performance both in Container and Restored Secret. In addition, the Container of Stego-STFAN is in third, only behind the VStegNET by a few points. However, in terms of Restored Secret PSNR, our model requires major improvements since 23 dB is low in quality and is behind most of the recent models. However, except for DRANet, the other models use 3D convolution, making our model not only lighter but faster. Moreover, only DRANet and Stego-STFAN use the Attention Mechanism. None of the other models use techniques like deblurring and alignment for steganography. Regarding visual evaluation, our Container is characterized by a different type of residual, that is, the black strips, which causes our Container to be slightly darker. The models mentioned before are characterized by having big spots due to the different techniques used. Moving to Restored Secret, our is similar to models that produce that are in the same quality level.

Table 1. Comparison with state of art models. Adapted from [32].

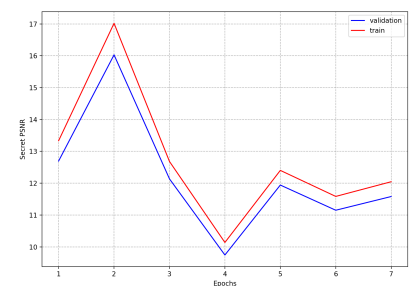
Model	Parameters (M)	$PSNR_c$	$PSNR_s$
HliPS	0.49	17.5363	16.6804
StegNet	0.15	20.8553	15.4662
SteganoGAN	0.06	12.3950	14.7580
HCCVS	314.16	26.7777	27.4274
VStegNET	122.38	27.7055	23.1677
DRANet	16.54	31.1122	28.4507
Stego-STFAN (Ours)	43.91	27.0164	23.0853

Due to the model's poor performance, no statements can be made regarding the hypothesis about the redundancy in the pixels or the exploration of the frame's temporal-spatial dependencies, like preference for blurred regions. The assumption of adding more frames is still unconfirmed since no difference was seen in the outputs using more or fewer frames. Furthermore, the influence of the DWT as extra context to the network cannot be affirmed for the same reasons as before.

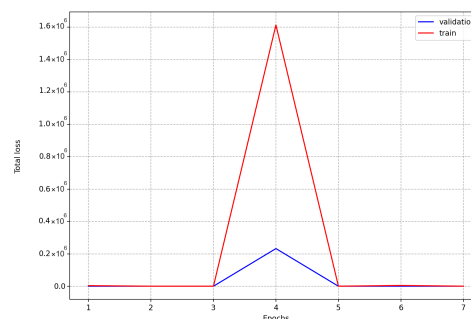
The Figure 7a–c show the learning curve using the loss (MSE) and PSNR. The results using batch size = 1 are unstable due to its high stochasticity in the parameter's update, which generates big variations from one epoch to the other. This results in loss convergence since it is harder to achieve generalization. This is present in the figures by checking loss values through the epochs, especially the total loss, which came from hundreds to millions in a single epoch. However, a good point is the similarity in performance in training and validation, which suggests that from the few features it was able to learn, it was done concisely (without overfitting). The first graph shows the Learning Curve, which did not stabilize and was briefly diverging. Also, the model struggles to maintain a consistent convergence even after the peak. The second shows the PSNR of the Container. Although they are close in quality, no statement can be made regarding the quality of the hiding due to the lack of a satisfactorily restored secret. Furthermore, the continuous variation in PSNR suggests insufficient features were incorporated to generalize. The last graph presents the PSNR curve of the Restored Secret. Validation and training performance are behaving in the same way (varying heavily), showing that no stable convergence was achieved.



(a) Container PSNR batch size = 1



(b) Restored Secret PSNR batch size = 1



(c) Total Loss batch size = 1

Figure 7. Results from model with batch size = 1.

Figure 8 is divided into four columns: Container (steganographed frame), the Original Cover, the Restored Secret, and the Original Secret with batch size = 4. The Restored Secret is indeed similar to the original one, although very blurry. In addition, there is no residual from the Container in the restored frame, which may suggest a different way of hiding the images that were not performed in previous works. In fact, looking closely at the Container, vertical black strips are present as the message, and not squared spots with different colors, as is seen in compared models [15,32,37–39].

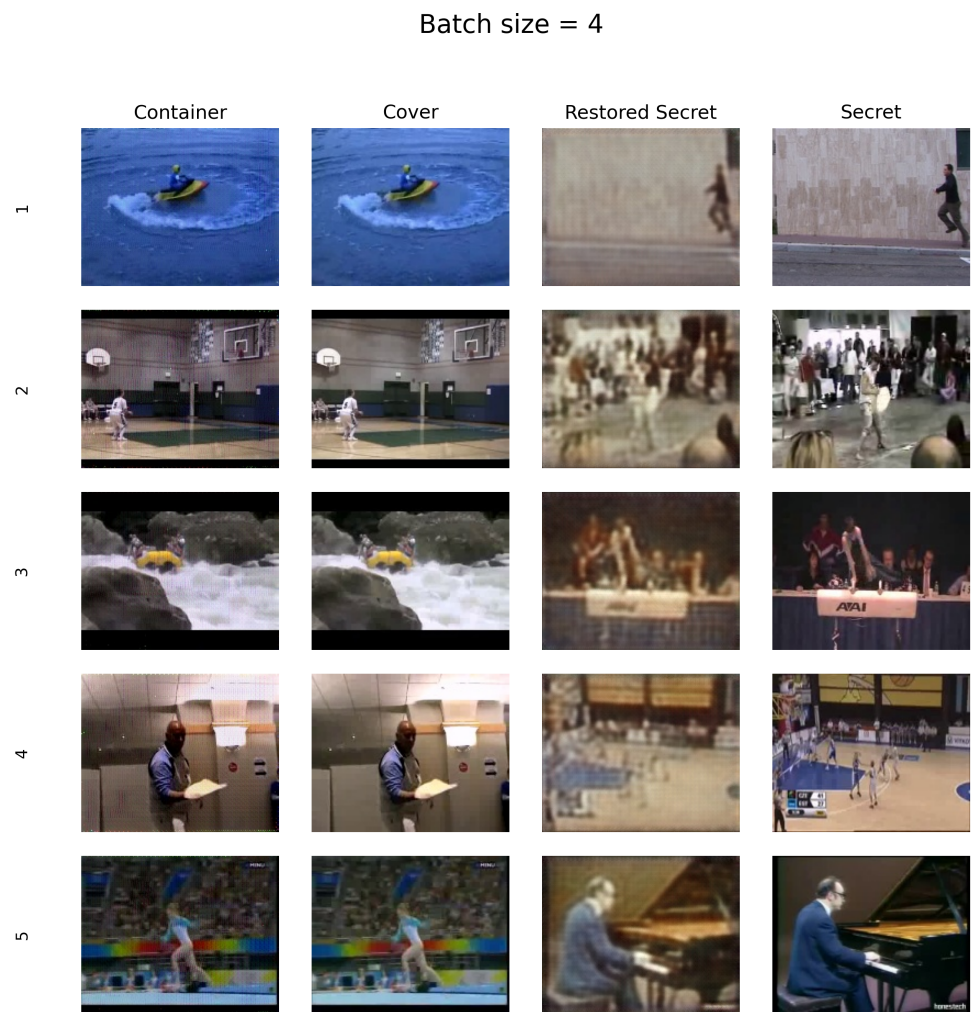
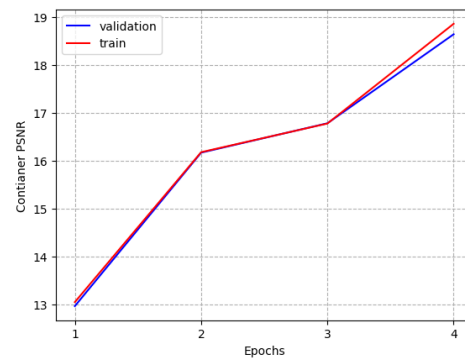


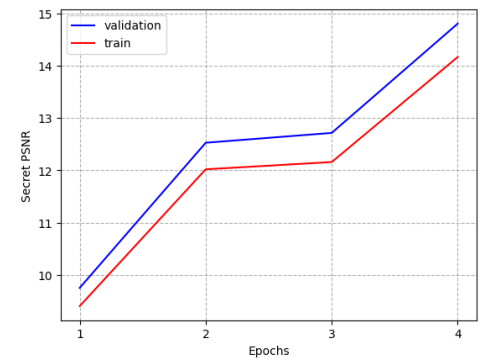
Figure 8. Results from model with batch size = 4.

Similar to Figure 7, Figure 9 shows the learning curve using the loss (MSE) and the PSNR using batch size = 4. Compared to the previous one, the training was far more stable and consistent, achieving better results using fewer resources, suggesting that the model learned to generalize easier than the previous one. Regarding the quality of the learning, the same can be said: from the few features incorporated in the model, it managed to generalize well. The first graph shows the Learning Curve that fell and then decreased consistently, as expected. The second shows the PSNR of the Container.

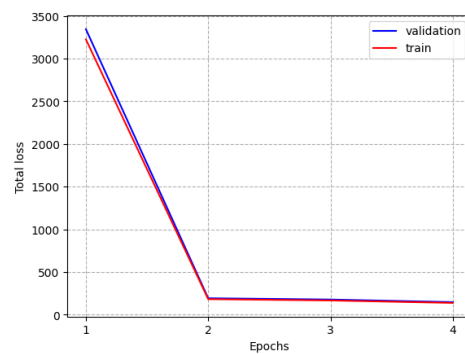
The performance of could be seen in the Figure 10. It is clear that the model indeed learned to steganograph. The learning curves increased as expected, deserving special attention to the PSNR of the Container, where Training and Validation frames were in almost all epochs together, suggesting that the model centered the learning on hiding. Also, this is confirmed by the PSNR absolute value. In the Validation dataset, the Container reached 27.03 dB, whereas the Restored Secret reached 23.09 dB, causing a huge gap between both tasks. However, since during and after finishing the training, the PSNR between the Container and Restored Secret is small (1.37 dB), the loss proposed indeed correctly penalized big differences between them both. Moreover, since the plateau was not reached in Container's loss, some improvements may still be done without losing performance in the restoration phase.



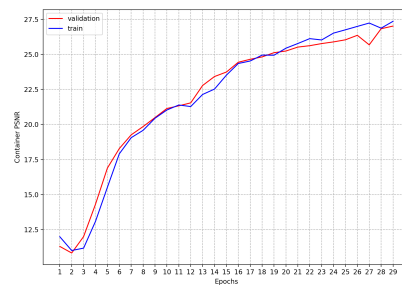
(a) Container PSNR batch size = 4



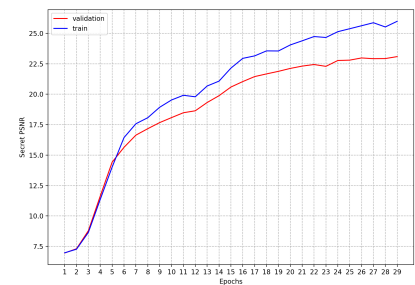
(b) Restored Secret PSNR batch size = 4.



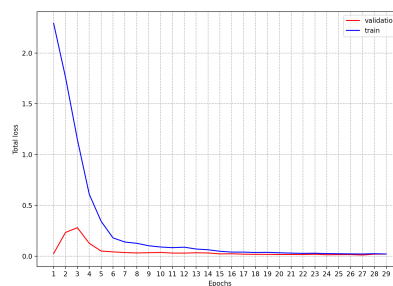
(c) Total Loss PSNR batch size = 4

Figure 9. Results from model with batch size = 4.

(a) Container PSNR



(b) Restored Secret PSNR



(c) Total Loss

Figure 10. Complete training performance graphs.

Regarding the black lines in the Container, the Figure 11 shows what these strips are. In the second column, the residual between the Container and the Cover forms these black strips (here called residual). It is clear that they approximately shape the secret since the restoration is made using them as information. Except in image 5, where the picture is big

and fairly simple, the shape does not give much information about what the secret image is, neither its color. For clearance, the “random” colors in the top and bottom of the images are caused by the abrupt differences that have in the secret and cover, and, when multiplied by a large value ($\times 4$), extrapolate the pixel value.

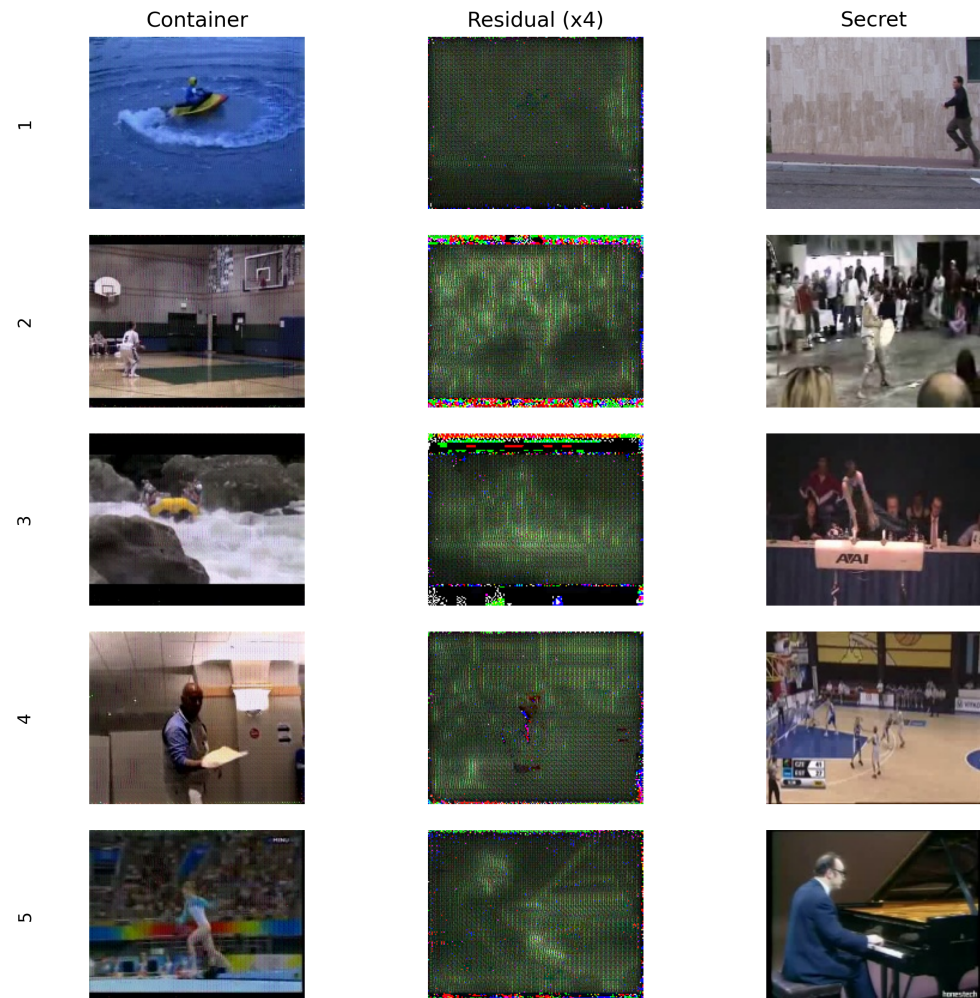


Figure 11. Residual from the container.

5. Discussion

From the results, it is possible to say that, despite the poor convergence, the model learned to steganograph videos, even though its quality is below the state-of-the-art models. However, its uniqueness may guarantee more security in recuperation by a third party than current models since the kernel produced in the FAC layer (dynamic filters) imposes a restriction on how the frame is processed both in hiding and restoration. If needed, these filters may be used as a key to allow only certain parties to retrieve the secret, making the model also secure during particular cases where some level of encryption is required. It is worth mentioning that the kernel of the next frames depends on the previous one. Therefore, it is only necessary to encrypt the first kernel (first frame) so that the model will produce the others normally. However, the model may correct itself even if not decrypting the kernel because the other parts of the network will recover but with considerable discrepancy from the correct kernel. Therefore, the model may suffer some changes to adequate to this usage.

Compared to state-of-the-art networks, our model is still behind in performance, especially in the restoration phase. However, we believe that putting more effort into the second phase using attention mechanism and other feature extraction techniques (like DWT) may help the model to perform equally with SOTA models. Moreover, the Container

has good quality, being necessary improvements in feature extraction during the restoration part to enable the network to hide messages (black strips) in the Container without losing information about the Secret to be restored.

Thus, studies to address the low convergence are required and may follow the suggestions made before improving the recovery. On account of these results, the hypothesis could not be proved (or dismissed) since the outputs are considerably brute to motivate studies about regions that the network may choose or performance in blurred videos, as well as the addition of more frames.

6. Conclusions

From the analysis of this research, it was found that steganography is a fascinating area of study in science as a whole. Even so, it can act as a double-edged sword because, due to its principle of hiding the existence of information, its use is extremely valuable for illicit purposes, as was the case when a drug trafficking control network and execution were debunked after finding more than 200 images of Hello Kitty that contained steganographed text and voice information. Therefore, with regard to existing techniques in steganography, it can be said that they are fragile in terms of integrity, as a simple compression can destroy hidden information, but strong in terms of information security since the complexity and high computational cost would make it unfeasible to develop steganalysis algorithms that can extract the hidden message, and not just identify its existence.

Given the results, it is clear that the model did not work as expected. However, it is early to state that the proposed hypotheses are false since results with consistent accuracy were not observed in accordance with the literature. Therefore, more studies are needed to reach a conclusion about the real behavior of the algorithm. Improving the training strategy is highly suggested as a first step, establishing more apparent objectives (losses) than those used. Therefore, exploring more attention techniques like Transformers [36] can be vital to achieving better results since the attention mechanism, in simple words, is designed to highlight essential features of the image, which is precisely necessary to direct the training of the network to help improve the convergence result which in this research was quite low.

Furthermore, the evaluation of the spatio-temporal aspects of the network still needs to be confirmed, as the results do not allow any statement. Thus, the effectiveness of the attention used, especially CBAM, could not be demonstrated, except in [32]. Therefore, the effectiveness of attention blocks with dynamic filters, which target the pixel level (from their adaptability to the frame), could not be documented as effective.

For future studies, in-depth studies of compression techniques in both images and videos are relevant, it can be concluded that the treatment for steganography in videos must be carried out respecting the characteristics of temporal compression. More specifically, the need to explore the characteristics of movement vectors for the possible insertion of hidden content was verified. Another important point that could be explored together with temporal compression will be the adaptation of spatial compression techniques to insert hidden data within videos. Another important factor verified was the lack of classifications for steganography techniques and modes, so researchers respect the taxonomy according to regional characteristics.

Author Contributions: Conceptualization and writing, G.F.V. and P.G.; methodology, formal analysis, reviewing and supervision, A.L.M.S.; review and validation, F.L.L.d.M., G.A.P.R. and G.D.B.; original draft preparation and methodology, V.P.G.; methodology and review, R.d.O.A. and R.T.d.S.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: This paper uses two publicly available datasets. UCF101—Action Recognition Data Set and Weizmann Human Action Dataset.

Acknowledgments: The authors would like to thank the Brazilian National Confederation of Industry (CNI) for partially supporting this project and for their support and collaboration throughout this research project. Also we would like to thank the Laboratory of Technologies for Decision Making—LATITUDE, University of Brasilia.

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

References

1. Di Giulio, M.; Vecchi, G. Implementing digitalization in the public sector. Technologies, agency, and governance. *Public Policy Adm.* **2023**, *38*, 133–158. [\[CrossRef\]](#)
2. Rodrigues, G.A.P.; Serrano, A.L.M.; Vergara, G.F.; Albuquerque, R.d.O.; Nze, G.D.A. Impact, Compliance, and Countermeasures in Relation to Data Breaches in Publicly Traded US Companies. *Future Internet* **2024**, *16*, 201. [\[CrossRef\]](#)
3. Rustad, S.; Andono, P.N.; Shidik, G.F. Digital image steganography survey and investigation (goal, assessment, method, development, and dataset). *Signal Process.* **2023**, *206*, 108908.
4. Kunhoth, J.; Subramanian, N.; Al-Maadeed, S.; Bouridane, A. Video steganography: Recent advances and challenges. *Multimed. Tools Appl.* **2023**, *82*, 41943–41985. [\[CrossRef\]](#)
5. Narayana, V.L.; Malleswari, K.; Divyanjali, M.; Nandini, S.; Purnima, G. Video Frame based Prompt Compression Model with Steganography for Secure Data Transmission. In Proceedings of the 2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS), Coimbatore, India, 2–4 February 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 373–377.
6. Liu, Y.; Liu, S.; Wang, Y.; Zhao, H.; Liu, S. Video steganography: A review. *Neurocomputing* **2019**, *335*, 238–250. [\[CrossRef\]](#)
7. Zhou, S.; Zhang, J.; Pan, J.; Xie, H.; Zuo, W.; Ren, J. Spatio-temporal filter adaptive network for video deblurring. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 2482–2491.
8. Bispo, G.D.; Vergara, G.F.; Saiki, G.M.; Martins, P.H.d.S.; Coelho, J.G.; Rodrigues, G.A.P.; Oliveira, M.N.d.; Mosquera, L.R.; Gonçalves, V.P.; Neumann, C.; et al. Automatic Literature Mapping Selection: Classification of Papers on Industry Productivity. *Appl. Sci.* **2024**, *14*, 3679. [\[CrossRef\]](#)
9. Jebur, M.H.; Joda, F.A.; Naser, M.A. Video Steganography Technique Based on Enhanced Moving Objects Detection Method. *J. Univ. Babylon Pure Appl. Sci.* **2023**, *31*, 270–295. [\[CrossRef\]](#)
10. Mou, C.; Xu, Y.; Song, J.; Zhao, C.; Ghanem, B.; Zhang, J. Large-capacity and flexible video steganography via invertible neural network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 22606–22615.
11. Xu, S.; Li, Z.; Zhang, Z.; Liu, J. An End-to-End Robust Video Steganography Model Based on a Multi-Scale Neural Network. *Electronics* **2022**, *11*, 4102. [\[CrossRef\]](#)
12. Gómez, A.; Muñoz, A. Deep Learning-Based Attack Detection and Classification in Android Devices. *Electronics* **2023**, *12*, 3253. [\[CrossRef\]](#)
13. Zeng, A.; Sun, X.; Huang, F.; Liu, M.; Xu, Q.; Lin, S. Srnet: Improving generalization in 3d human pose estimation with a split-and-recombine approach. In Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part XIV 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 507–523.
14. Boehm, B. Stegexpose-A tool for detecting LSB steganography. *arXiv* **2014**, arXiv:1410.6656.
15. Mishra, A.; Kumar, S.; Nigam, A.; Islam, S. VStegNET: Video Steganography Network using Spatio-Temporal features and Micro-Bottleneck. In Proceedings of the BMVC 2019, Cardiff, UK, 9–12 September 2019.
16. Zhang, S.; Wu, Y.; Che, T.; Lin, Z.; Memisevic, R.; Salakhutdinov, R.R.; Bengio, Y. Architectural complexity measures of recurrent neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; Volume 29.
17. Zaremba, W.; Sutskever, I.; Vinyals, O. Recurrent neural network regularization. *arXiv* **2014**, arXiv:1409.2329.
18. Cho, K.; Van Merriënboer, B.; Bahdanau, D.; Bengio, Y. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv* **2014**, arXiv:1409.1259.
19. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Volume 30.
20. Buades, A.; Coll, B.; Morel, J.M. A non-local algorithm for image denoising. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05), San Diego, CA, USA, 20–25 June 2005; IEEE: Piscataway, NJ, USA, 2005; Volume 2, pp. 60–65.
21. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7794–7803.
22. Bravo-Ortiz, M.A.; Mercado-Ruiz, E.; Villa-Pulgarin, J.P.; Hormaza-Cardona, C.A.; Quiñones-Arredondo, S.; Arteaga-Arteaga, H.B.; Orozco-Arias, S.; Cardona-Morales, O.; Tabares-Soto, R. CVTStego-Net: A convolutional vision transformer architecture for spatial image steganalysis. *J. Inf. Secur. Appl.* **2024**, *81*, 103695. [\[CrossRef\]](#)
23. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.

24. Li, L.; Yuan, R.; Lv, Y.; Xu, S.; Hu, H.; Song, G. An efficient robotic-assisted bolt-ball joint looseness monitoring approach using CBAM-enhanced lightweight ResNet. *Smart Mater. Struct.* **2023**, *32*, 125008. [\[CrossRef\]](#)
25. Sabeena, M.; Abraham, L. Convolutional block attention based network for copy-move image forgery detection. *Multimed. Tools Appl.* **2024**, *83*, 2383–2405. [\[CrossRef\]](#)
26. Singh, A.K.; Dave, M.; Mohan, A. Hybrid technique for robust and imperceptible multiple watermarking using medical images. *Multimed. Tools Appl.* **2016**, *75*, 8381–8401. [\[CrossRef\]](#)
27. Dalal, M.; Juneja, M. Steganalysis of DWT Based Steganography Technique for SD and HD Videos. *Wirel. Pers. Commun.* **2023**, *128*, 2441–2452. [\[CrossRef\]](#)
28. Suresh, M.; Sam, I.S. Optimal wavelet transform using Oppositional Grey Wolf Optimization for video steganography. *Multimed. Tools Appl.* **2020**, *79*, 27023–27037. [\[CrossRef\]](#)
29. Zear, A.; Singh, A.K.; Kumar, P. A proposed secure multiple watermarking technique based on DWT, DCT and SVD for application in medicine. *Multimed. Tools Appl.* **2018**, *77*, 4863–4882. [\[CrossRef\]](#)
30. Han, Y.; Huang, G.; Song, S.; Yang, L.; Wang, H.; Wang, Y. Dynamic neural networks: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 7436–7456. [\[CrossRef\]](#)
31. Djenouri, Y.; Belhadi, A.; Yazidi, A.; Srivastava, G.; Lin, J.C.W. Artificial intelligence of medical things for disease detection using ensemble deep learning and attention mechanism. *Expert Syst.* **2024**, *41*, e13093. [\[CrossRef\]](#)
32. Cui, J.; Zheng, L.; Yu, Y.; Lin, Y.; Ni, H.; Xu, X.; Zhang, Z. Deeply-Recursive Attention Network for video steganography. *CAAI Trans. Intell. Technol.* **2023**, *8*, 1507–1523. [\[CrossRef\]](#)
33. Wu, Y.; He, K. Group normalization. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
34. Dalal, M.; Juneja, M. A robust and imperceptible steganography technique for SD and HD videos. *Multimed. Tools Appl.* **2019**, *78*, 5769–5789. [\[CrossRef\]](#)
35. Duan, X.; Wu, G.; Li, C.; Li, Z.; Qin, C. DUIANet: A double layer U-Net image hiding method based on improved Inception module and attention mechanism. *J. Vis. Commun. Image Represent.* **2024**, *98*, 104035. [\[CrossRef\]](#)
36. Pan, J.; Xu, B.; Bai, H.; Tang, J.; Yang, M.H. Cascaded Deep Video Deblurring Using Temporal Sharpness Prior and Non-Local Spatial-Temporal Similarity. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 9411–9425. [\[CrossRef\]](#) [\[PubMed\]](#)
37. Zhang, K.A.; Cuesta-Infante, A.; Xu, L.; Veeramachaneni, K. SteganoGAN: High capacity image steganography with GANs. *arXiv* **2019**, arXiv:1901.03892v2.
38. Weng, X.; Li, Y.; Chi, L.; Mu, Y. High-capacity convolutional video steganography with temporal residual modeling. In Proceedings of the International Conference on Multimedia Retrieval, Ottawa, ON, Canada, 10–13 June 2019.
39. Jaiswal, A.; Kumar, S.; Nigam, A. En-vstegnet: Video steganography using spatio-temporal feature enhancement with 3d-cnn and hourglass. In Proceedings of the International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1–8.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.