IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 3, 2018, accepted February 7, 2019, date of publication February 11, 2019, date of current version March 1, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2898838

The Pixogram: Addressing High Payload Demands for Video Steganography

TAMER RABIE[1,2] AND MOHAMMED BAZIYAD[3]
[1]Department of Electrical and Computer Engineering, University of Sharjah, Sharjah 27272, United Arab Emirates
[2]Intelligent Transportation Systems Centre, University of Toronto, Toronto, ON M5S 1A4, Canada
[3]Research Institute of Sciences and Engineering, University of Sharjah, Sharjah 27272, United Arab Emirates

Corresponding author: Tamer Rabie (trabie@sharjah.ac.ae)

This work was supported in part by the College of Graduate Studies and Research, University of Sharjah, and in part by the Research Institute of Sciences and Engineering, University of Sharjah.

ABSTRACT This paper introduces the concept of a pixogram which makes possible a fresh approach to high payload video steganography. The pixogram allows for a new perspective by investigating the temporal changes that take place at the individual pixel level across frames of a video segment. Simply put, a pixogram has the property of converting highly uncorrelated spatial areas of individual frames of a video scene into highly correlated temporal segments by making use of the temporal correlation between the frames of the same scene in a given video segment, thus maximizing the redundant area suitable for hiding in the transform domain. Experimental results demonstrate the effectiveness of this new approach for increased payload capacity while maintaining visual fidelity of the stego-video as compared to competing video steganography schemes.

INDEX TERMS Pixogram, video steganography, segment-growing, temporal correlation.

I. INTRODUCTION

The rapid growth in processing power of current computer systems and communication devices, coupled with huge improvements in network speeds, has lead to increased popularity of video manipulation, streaming, and transmission over wireless networks in the digital world. Many companies and businesses are based on video streaming, such as "Youtube", "Instagram", and "Dailymotion", or involved in developing software for manipulating these digital signals such as "Adobe Premiere Pro" and "Final Cut Pro X".

A recent study by Cisco[1] predicted that in 2019 approximately 80% of the world's internet traffic will be video. The study predicts that by 2021 a million minutes of video content will cross the internet every second. Currently, IP video traffic acquires 73% of all consumer internet traffic. This percentage is expected to rise to 82% in 2021. Based on the study, live internet video will account for 13% of internet video traffic by 2021, with a 15-fold growth from 2016 to 2021. Internet video surveillance, which is an application of live internet video, has a traffic that grew 72% in 2016, with 883 PetaBytes per month. It is expected that internet video surveillance traffic will boost seven times between 2016 and 2021. Video surveillance will take 3.4% of all internet video traffic in 2021.

This increased demand for video streaming over the internet, paired with the critical nature of various video applications, has opened many important issues in different computing fields, such as networking and security. From the networking perspective, transmitting video over networks forms a classical bottleneck problem. In the last few years, the problem of optimizing network performance for video applications has attracted many researchers [1], [2]. The challenge of video transmission from a networking point of view is the large number of bits to transmit in a real-time manner.

Moreover, some video streaming applications, such as surveillance, require some security services such as integrity and confidentiality. Confidentiality can be defined as restricting access to authorized entities to view the data. On the other hand, integrity deals with achieving consistency, accuracy, and trustworthiness of the data. In other words, the data must not be altered during transmission, and techniques must be implemented to ensure that the data cannot be changed by unauthorized entities.

The associate editor coordinating the review of this manuscript and approving it for publication was Yan Huo.

[1] https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.html

Video compression is the solution to deal with the huge number of bits involved in video transmission. Since network resources are limited, a video is compressed before transmission or storage. However, for security purposes, steganography and watermarking techniques are used to achieve confidentiality and integrity respectively. Steganography is the art that deals with hiding a secret message into an innocent carrier medium forming a stego medium. This stego medium should not reveal the existence of the hidden secret message. Watermarking is another hiding strategy, but when the purpose is integrity and intellectual property of the information being exchanged [3], [4].

Both compression and data hiding techniques share the same basic concept, that of utilizing high-frequency bands representing redundant information in a signal. Perceptual experimental evidence has established that the human visual system is less sensitive to noise present in high-frequency areas [5]. Thus, in compression, these bits are removed to reduce the size of the signal, while data hiding techniques, such as steganography and watermarking, replace these bits with secret bits [6].

A video segment can be defined as a series of related images, correlated in time, and displayed sequentially at a constant rate. Thus, a video is a 3D signal which can be expressed as $\mathcal{V}(r, c, t)$, where r and c represent the spatial positions in the vertical and horizontal directions respectively, while t is the temporal dimension. As a result of the 3D nature of video signals, two types of information redundancies arise, spatial and temporal redundancies. Spatial redundancy is related to the inter-pixel relations of individual frames in the (r, c) space while fixing the time dimension (t). Similarly, temporal redundancy is the inter-pixel relations along the time dimension (t) while fixing the spatial dimensions (r, c).

There are four main attributes that researchers try to improve in steganography, namely the capacity, imperceptibility, robustness, and security. Capacity is the amount of secret information that can be embedded in the cover medium. Imperceptibility refers to the amount of degradation in the cover medium due to the hiding process. Moreover, robustness is defined as the degree of resistance against attacks, and whether the hidden data can still be extracted after the attack. While, security refers to an attacker's inability to extract the hidden data, or to detect the existence of the hidden data in the stego medium.

This paper proposes a novel *video-in-video* hiding technique that exploits the temporal redundancy in video segments. The proposed approach is based on the fact that regions in the video that have minimal or no movement will have redundant pixels along the time dimension. Thus, a temporal vector of pixels is extracted from each row, column (r, c) location of a video segment forming what is referred to throughout this paper as a *pixogram*. This pixogram is then segmented into homogeneous segments using a 1D segment-growing segmentation technique that is developed in this paper. Due to the homogeneity in each segment, the 1D-DCT of each segment will represent the segment in a small number of low-frequency-compacted significant DCT coefficients, leaving the larger high-frequency insignificant DCT coefficient portion for embedding. This allows the proposed Pixogram Adaptive Region (PixAR) technique to surpass other video-based steganography schemes in terms of embedding payload capacity as well as imperceptibility of the stego video.

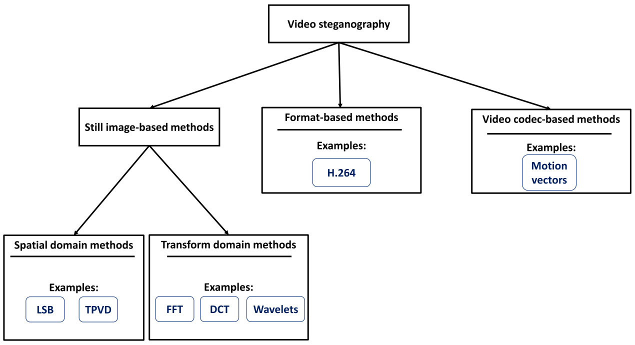


FIGURE 1. Categorizing video steganography methods.

The rest of the paper is organized as follows; section II presents background work in the area of video steganography. The embedding and extraction processes of the proposed PixAR scheme are discussed in section III, while section IV presents experimental results and comparative discussions. Final conclusions are presented in section V.

II. RELATED WORK

In general, video steganography methods can be categorized into three main categories, namely still image-based methods, format-based methods, and video codec-based methods [7], [8]. Figure 1 illustrates these different categories in a tree diagram. The still image-based methods treat video steganography as an extension of image steganography. Since video is simply a sequence of still images, these methods will operate on the basis of hiding the secret data frame by frame separately.

Still image-based methods can be further classified into spatial domain methods, and transform domain methods. In the spatial domain methods, LSB techniques is the classical and most popular method due to its simplicity and the low computational complexity [7]. LSB methods operate by replacing some LSBs of pixels from the cover video frames with the secret message bits. A hash-based LSB method for video steganography is proposed by Dasgupta *et al.* [9]. The position of insertion in LSB bits is located using a hash function. Then, each eight bits of the secret data are divided into 3,3,2 and embedded into the R,G,B pixel values of the cover frames respectively.

Dasgupta *et al.* [10] have made further improvements over basic LSB methods by using a Genetic Algorithm (GA). The proposed scheme was able to enhance the quality of the stego video while embedding at the same capacity rate. The scheme was able to reach an imperceptibility level measured at 39.374 dB with a capacity rate of 2.66 bits per bytes (bpB) for a gray-scale video, which is equivalent to hiding at 7.98 bits per pixel for an RGB video.

Another LSB based technique is proposed by Singh and Agarwal [11]. The proposed technique utilizes only the LSB of the cover frames to embed a single secret image in the video frame-by-frame. Furthermore, the algorithm embeds each row of the secret image pixels in the rows of multiple frames of the cover video. In other words, each row of pixels composed of 8 bits is embedded in the first rows of multiple frames of the cover. Thus, to hide one byte of secret image pixels, 8 bytes are needed from the cover video frames. It is clear that this technique is simple, and easy to develop. In addition, the imperceptibility level of the scheme is very high since only one LSB is used for embedding. However, as a result, the capacity is very low compared to other hiding techniques in the literature.

Another spatial domain method is the Tri-way Pixel-Value Differencing (TPVD) method proposed in [12]. The Tri-way Pixel-Value Differencing (TPVD) method is simply a modified version of the popular Pixel-Value Differencing (PVD) method. PVD embeds the secret data in the difference value of two adjacent pixels. In [13], a TPVD based video steganography system is proposed. The system uses TPVD method to embed in a compressed domain of the cover medium.

The second subcategory that falls under the still images methods is the transform techniques. Mainly, in the transform techniques, the cover medium is first transfered to the frequency domain using one of the transform functions. Then, the secret data is embedded by replacing some selected coefficients. At the end, the domain, with the modified coefficients, is transformed back to the spatial domain. Transform techniques include: Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT).

A Discrete Wavelet Transform (DWT) based scheme is proposed in [14]. This scheme encodes first the secret video using the Bose-Chaudhuri-Hocquenghem (BCH) codes, and then hides the encoded version of the secret message into the Discrete Wavelet Transform (DWT) coefficients of video frames. The scheme hides the secret data in all DWT regions except the low-frequency region, since this region has the significant coefficients of the image which must be preserved for proper reproduction of the frame. Thus, the scheme was able to embed with a higher capacity rate than other proposed schemes in the literature. The proposed scheme was also tested on cover videos that are compromised with fast and slow motion objects, and showed an improved performance in terms of the hiding capacity and the imperceptibility over other hiding techniques in the literature.

A novel method is proposed in [15]. The proposed method is a blind adaptive steganography scheme for video files where hiding takes place in human skin regions in an image. Embedding is performed in the red and blue channels using a wavelet quantization technique. The technique showed robustness against MPEG-4 compression techniques.

From the description of the previous techniques, it is clear that transform methods have improved security and robustness over LSB based methods. However, transform methods suffer from a traditional trade-off between capacity and imperceptibility. This relationship was exhaustively investigated by Rabie and Kamel [5], [16], [17] and Rabie *et al.* [18], [19] for image-based steganography, but never before investigated in-depth for video-based steganography schemes.

Rabie and Kamel [5] studied the trade-off between hiding capacity and imperceptibility by implementing a fixed-block adaptive-region (FBAR) DCT embedding approach. The aim of this work was to investigate the embedding limits of DCT techniques. In [16], the fixed-block-size globally adaptive-region (FB-GAR) approach was proposed, which was an improvement over the FBAR method. The new technique was successfully able to hide at higher capacities and improved imperceptibility.

Rabie and Kamel [17] introduced the quad-tree adaptive region (QTAR) DCT hiding scheme. This hiding scheme is based on the fact that DCT of highly correlated images can be expressed using a few coefficients in the top left corner of the transform, leaving a significantly large area to hide in. Thus, the system partitions the cover image into non overlapping blocks for maintaining stationarity of the image regions using a quad-tree partitioning algorithm. The system was able to achieve a very high capacity at reasonably high levels of imperceptibility.

The novel work by Rabie *et al.* [18] has shown that it was possible to brake the barrier between hiding capacity and imperceptibility. Instead of hiding in a squared region area, the idea was to fully exploit the embedding region in the transform domain by developing a curve-fitting technique to maximize both the embedding payload capacity as well as the stego perceptual quality.

The second category involve format-based techniques. These methods are designed for a specific video format, by exploiting the structure and compression strategy of the format. H.264/AVC is an example of a standard video format [20]. This compression standard can achieve very high compression ratios, as well as being widely used for network transmission [21]. A video steganography scheme based on the H.264 standard is proposed in [22]. The technique is based on the Context Adaptive Variable Length Coding (CAVLC) which is utilized in H.264 baseline entropy coding. Embedding secret data is performed by altering trailing ones sign flag and levels' words in CAVLC.

Another H.264-based method is proposed in [23]. This algorithm embeds secret data in the high-frequency coefficients information (Trailing Ones) of CAVLC coding. The scheme has the advantage of low computationally complexity compared to other H.264/AVC-based hiding techniques, and thus, it can be implemented in a real-time manner. However, the scheme was not designed to embed with large hiding capacity; it only embeds 250 bits in a 352 × 288 video composed of 30 frames.

The third category is the video codec-based methods. These methods try to exploit the 3D nature of videos, and utilizing the third dimension which is the time dimension t in embedding. This additional dimension introduces some additional properties such as motion vectors, and motion components [7]. In [24], an adaptive steganography scheme hides data in a compressed video stream using temporal and spatial features of the cover video and human visual system concept.

Another hiding scheme based on discrete wavelet transform (DWT) and discrete cosine transform (DCT) is proposed in [25]. The scheme relies on a multiple object tracking (MOT) algorithm and error correcting codes in the embedding process. First, the "Hamming and Bose" algorithm along with the "Chaudhuri and Hocquenghem" algorithm are performed on the secret data for encoding purposes. Then, the motion-based MOT algorithm is executed on the cover video to manifest the area of interest in moving objects. In the final stage, and based on the foreground mask, the secret message is embedded into the DWT and DCT coefficients of all motion regions in the video. The scheme was able to embed a secret message with a size of 3.40% of the cover video, with a stego quality measured at 49.0 dB.

In [26], a robust video watermarking scheme is proposed to hide a watermark image in digital video frames using the variable-temporal length 3-D DCT technique. The technique selects a number of successive 8x8 blocks, and applies the 3D-DCT to these blocks. Embedding takes place in the mid-range coefficients of individual blocks.

Since the proposed PixAR scheme utilizes the temporal redundancy in a video segment, this proposed PixAR technique may be categorized as a video codec-based method. The proposed PixAR scheme differs, however, from other video codec-based algorithms by extracting a temporal 1D vector (pixogram), and then applying 1D-DCT to each homogeneous section of the vector, rather than taking the 3D-DCT of sequential temporal blocks as in [26], or exploiting moving objects as in [25]. The proposed PixAR scheme has shown improved results both in terms of capacity and imperceptibility over other video codec-based schemes and other state-of-the-art schemes as will be shown in the experimental section (section IV-A).

III. THE PROPOSED SCHEME

The proposed PixAR scheme aims to exploit video redundancy along the temporal dimension. This temporal redundancy is related to scene motion in a video segment. Areas in the video that have slow or zero-motion, will have identical or similar inter-frame pixel values, which gives rise to increased temporal redundancy between frames.

This suggests extracting a temporal 1D vector (a *pixogram*) consisting of pixels from sequential frames at the same spatial location (r, c). It is expected that such a vector will have many highly-correlated segments especially if the location (r, c) represents low or minimal motion throughout the frames of this video segment. Since this *pixogram* may have repeated values, transforming it using the 1D-DCT will compact the energy of this vector in the first few low-frequency coefficients, leaving a large number of redundant high-frequency coefficients that may be used for hiding.

To fully exploit temporal redundancy, a pixogram is first segmented into homogeneous segments using a 1D segment-growing strategy, resulting in each segment being expressed most of the time by only a single significant DCT coefficient, namely the DC coefficient, leaving the rest of the DCT segment suitable for embedding.

The proposed PixAR scheme was able to achieve high payload capacity levels that reached 22.3 bpp, at acceptable PSNR stego-video quality levels of approximately 40 dB, as will be shown in section IV.

A. PIXOGRAM SEGMENT-GROWING

The first step in the proposed PixAR scheme is to extract a *pixogram* from each row and column location (r, c) in the cover video. A pixogram, $\mathcal{P}(r, c)$, for a given cover video $\mathcal{V}$,

$$\mathcal{P}(r, c) = \mathcal{V}_{i=1:n}^{r,c}, \tag{1}$$

is a vector of n pixel values at spatial location (r, c), where n represents the total number of frames in this cover video. Each individual spatial position in the cover video (every (r, c) position) will have a pixogram associated with that position. The content of a pixogram is simply the values of pixels at (r, c) along the time dimension.

As an example, for spatial location $(r = 1, c = 1)$ (first row, first column), $\mathcal{V}_i^{1,1}$ is the value of the pixel located at $(r = 1, c = 1)$ in the i^{th} frame; $\mathcal{V}_1^{1,1}$ is the value extracted from the first frame, $\mathcal{V}_2^{1,1}$ is the value extracted from the second frame, $\mathcal{V}_n^{1,1}$ is the value extracted from the last frame.

To optimally benefit from the merits that temporal redundancy provides, the pixogram is segmented further into homogeneous segments using a 1D segment-growing technique. This segmentation process is a 1D version of the image-based 2D region-growing technique, which is a class of bottom-up image segmentation algorithms used to segment an image into homogeneous regions based on similarity of neighboring gray-level values. Image segmentation methods such as the top-down quad-tree and bottom-up region-growing techniques are useful tools that have been utilized in a variety of image processing applications such as image steganography [17], image compression [27], and object detection [28].

The pixogram segmentation process starts by grouping the first two pixels of the pixogram, which are simply the pixels of the first and second frames for a certain (r, c) location, and then examines the homogeneity between the pixels of this segment. The 1D segment-growing algorithm measures the homogeneity by evaluating the absolute difference between the highest and lowest gray-level value in the current segment. If the difference value is less than a chosen threshold $\mathcal{T}_{seg}$, then it is considered that the homogeneity

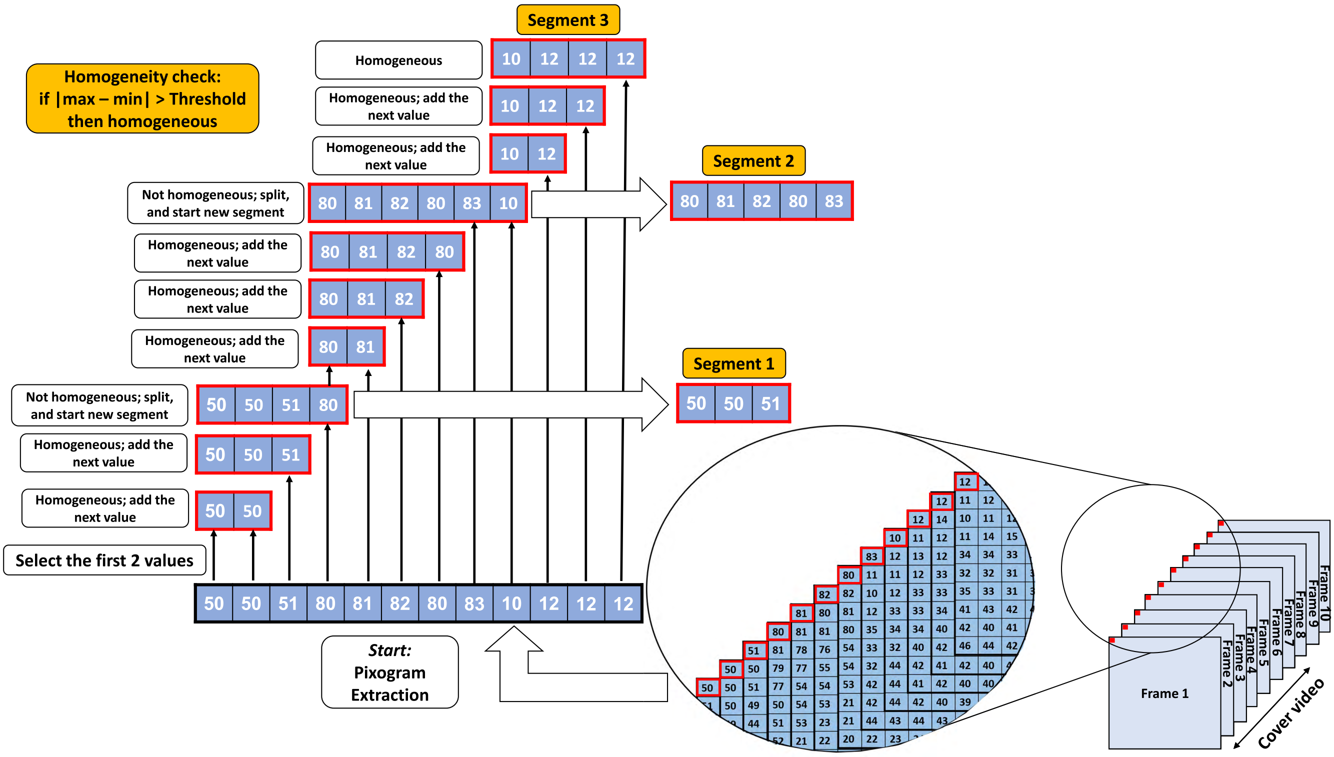

Segment 3
Homogeneity check:
if |max – min| > Threshold
then homogeneous
Homogeneous
10 12 12 12
Homogeneous; add the next value
10 12 12
Homogeneous; add the next value
10 12
Segment 2
Not homogeneous; split, and start new segment
80 81 82 80 83 10
80 81 82 80 83
Homogeneous; add the next value
80 81 82 80
Homogeneous; add the next value
80 81 82
Homogeneous; add the next value
80 81
Segment 1
Not homogeneous; split, and start new segment
50 50 51 80
50 50 51
Homogeneous; add the next value
50 50 51
Homogeneous; add the next value
50 50
Select the first 2 values
50 50 51 80 81 82 80 83 10 12 12 12
Start:
Pixogram
Extraction
Frame 1
Frame 2
Frame 3
Frame 4
Frame 5
Frame 6
Frame 7
Frame 8
Frame 9
Frame 10
Cover video

FIGURE 2. An illustration of the proposed segment-growing algorithm.

criterion is satisfied, and an additional pixel from the next frame in the pixogram is added to the current segment.

Once the segmentation algorithm adds a new pixel to the current segment, it re-evaluates the homogeneity of this updated segment (with the newly added pixel). The moment that the homogeneity check fails (the segment becomes no longer homogeneous), this newly added pixel is removed from the current segment and is now used to initiate a new segment. This process is repeated iteratively until the end of the pixogram is reached, which results in several homogeneous segments for this pixogram. Figure 2 illustrates the working strategy of the proposed pixogram segment-growing scheme.

B. THE EMBEDDING PROCESS

Figure 3 illustrates the general idea behind the proposed PixAR video steganography scheme.

Once pixograms have been homogenized using the segment-growing scheme described in the previous section, the 1D-DCT is applied to each homogeneous pixogram segment. The DCT has the property of strong energy compaction for stationary signals, where most of the signal information tends to be concentrated in a few low-frequency coefficients of the DCT [29]. Since a segmented pixogram will have many redundant values, most of the pixogram's information will be compacted in the first few coefficients of each segment's DCT, leaving the large number of insignificant coefficients in the DCT high-frequency region to be utilized for hiding.

Moreover, decreasing the gray-level homogeneity threshold value $\mathcal{T}_{seg}$, used in the pixogram segment-growing process, will decrease the variance of a segment. Thus, the number of significant DCT coefficients will decrease further, allowing for an increase in payload capacity. A threshold value $\mathcal{T}_{seg}$ of 0.1 (for gray-levels in the range [0,1], which is equivalent to gray level 25 for the range [0,255]) results in having a single significant coefficient for each segment's DCT most of the time. Decreasing this threshold to a value lower than 0.1 increases the chance of having a single significant coefficient in each segment's DCT.

After that, each pixogram's DCT-magnitude is separated from its phase, since embedding will take place in the DCT-magnitude of each pixogram segment (figure 3-step 3). The reason behind this magnitude-phase separation step is because it has been shown that the DCT-phase conveys significant amounts of information about its associated signal. On the other hand, the DCT-magnitude provides much less significant information about the signal [30]–[32]. Thus, it is important for compression and steganography DCT-based schemes to retain the DCT-phase intact, and only modify the DCT-magnitude, in order to maintain high perceptual quality [33].

A quantization step is then applied to distinguish between critical and insignificant coefficients of the DCT-magnitude. The quantization process is performed by dividing the elements of the DCT-magnitude of a pixogram segment by

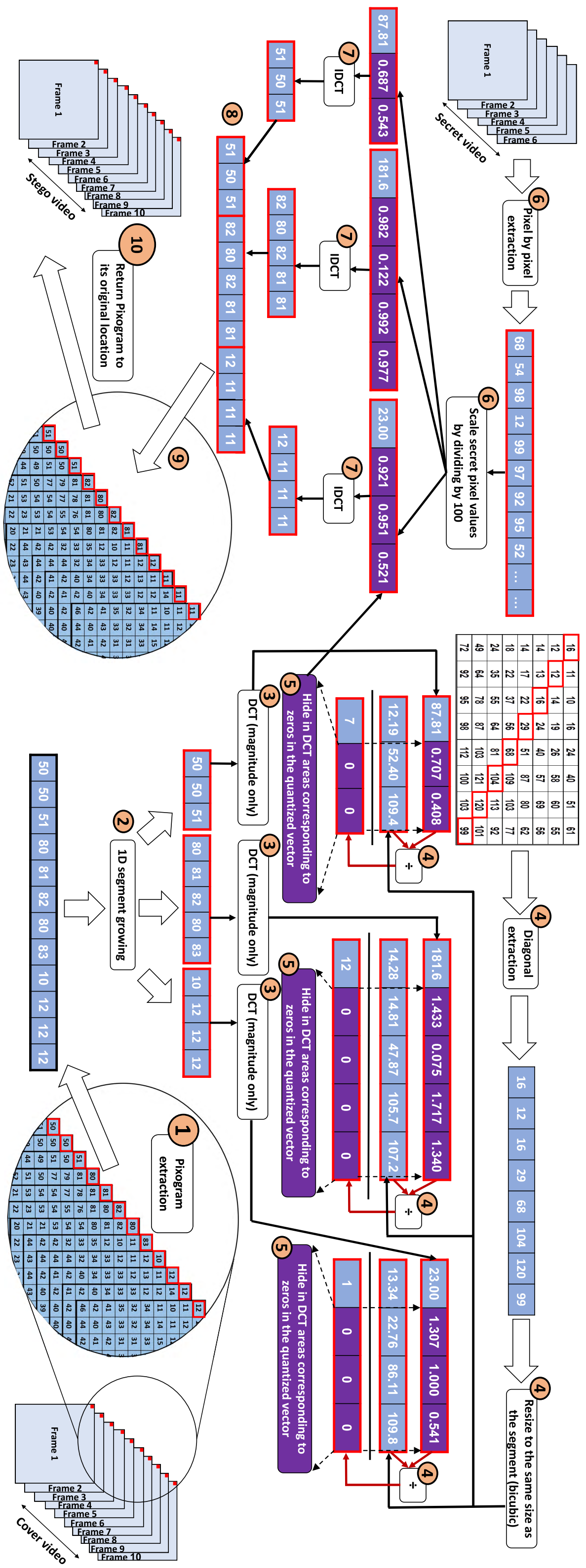

Cover video
Frame 1
Frame 2
Frame 3
Frame 4
Frame 5
Frame 6
Frame 7
Frame 8
Frame 9
Frame 10
1
Pixogram extraction
2
1D segment growing
3
DCT (magnitude only)
4
5
Hide in DCT areas corresponding to zeros in the quantized vector
4
Diagonal extraction
4
Resize to the same size as the segment (bicubic)
Secret video
6
Pixel by pixel extraction
6
Scale secret pixel values by dividing by 100
7
IDCT
8
9
10
Return Pixogram to its original location
Stego video

FIGURE 3. A generalized illustration of the proposed PixAR video steganography scheme.

the elements of a quantization vector of the same length. The base quantization vector values used are shown in figure 3-step 4, and are simply chosen to be all the diagonal elements of one of the standard quantization matrices used in JPEG compression.

The quantization process will result in a new quantized vector. The locations of insignificant coefficients in the DCT-magnitude of the pixogram segments have the same corresponding indices as the zero locations in this new vector. Pixels from the secret video are first rescaled, and then embedded in these insignificant locations as is clear in figure 3-steps 4, 5 and 6.

To properly choose the secret video size that can fit into these insignificant DCT locations, an initial scan on the whole cover video must be processed before embedding takes place (figure 3-steps 1 to 5, repeated for all pixograms of the cover video). This allows calculating the total number of insignificant DCT coefficients of the cover video that can be safely replaced with the secret video pixels. The secret video is then resized so that the total number of secret video pixels fits in the total number of insignificant DCT coefficients of the cover video that were marked for replacement after the initial scan. This can be described by the following equation:

$$H \approx x \times y \times f, \tag{2}$$

where H is the total number of insignificant DCT coefficients of the cover video, x and y are the width and height of the secret video in pixels (the resolution) and f is the number of frames of the secret video. In the proposed PixAR scheme, the resolution parameters (x, y) are selected to be the original resolution of the secret video to be hidden. The only unknown left in equation (2) is the number of frames (f) of the rescaled secret video. As an example, consider the number of insignificant DCT coefficients (H) is equal to $500,000$, the number of frames (f) to hide will equal 50, if the secret video resolution was 100×100. The secret video is then resized into a 1D vector, and embedding takes place pixel-by-pixel in the insignificant DCT coefficients of each homogenous segment of each pixogram extracted from the cover video.

After the embedding process takes place, the inverse DCT is applied on the modified segment's DCT (after multiplying it by its corresponding DCT-phase), and the segment is placed back into its location in the pixogram (figure 3-steps 7, 8, and 9). These quantization-embedding steps are performed on every segment in a pixogram. Once embedding in all segments of a pixogram is complete, the pixogram is returned to its original location (r, c) (figure 3-step 10).

C. THE EXTRACTION PROCESS

There are two approaches to extracting the secret video from the received stego video. The first is a blind approach where the receiver only needs the stego video and the homogeneity threshold value $\mathcal{T}_{seg}$ used in the pixogram segment gowing process. The second extraction approach requires sending the location of each pixogram segment and the index of the first insignificant coefficient in each segment along with the stego video, to the receiver side.

TABLE 1. Details of the video segments used as cover and secret videos.

Video Name	Resolution	No. of Frames	Frames used by Motion			Time (sec.)
			slow	medium	fast	
Gravity.avi	512x512	18048	37-87	3181-3230	8053-8064	752
Train.avi	256x256	2256	-	1-2256	-	94
Chappie.avi	512x512	28800	1008-1058	1-50	1632-1682	1200
WW2.avi	512x512	1104	24-74	504-554	729-842	46

TABLE 2. Comparative results expressed as the best VQM valuess while hiding using the maximum capacity for the various methods. Low VQM values are emphasized in a bold font.

Method	VQM
Chen 2009 *et al.* [36]	1.3
Chung 2010 *et al.* [37]	3.6
Xu 2012 *et al.* [38]	2.75
Xu 2014 *et al.* [39]	2.75
PixAR	**0.48**

This latter approach is useful when tampering of the stego video is to be expected, since the pixel values would be modified. In this case the extraction process takes place in the reverse order, where the pixograms are extracted from all (r, c) locations of the stego video. Then the DCT-magnitude of each segment is computed, and the secret pixels are extracted from each segment by knowing apriori the index of the first insignificant coefficient in that segment; the secret data will occupy all locations starting from the first insignificant coefficient in a segment till the end of the segment. This is then repeated for all segments.

The former blind approach assumes the stego video has not been modified or tampered with, such as when the transmission channel is guaranteed to be secure. In this case the received homogeneity threshold value $\mathcal{T}_{seg}$ will be used to sgement each pixogram of the secret video and the same quantization process (figure 3-steps 4 and 5) is applied on the DCT-magnitude of each segment to regenerate the zero-coefficient locations in the quantized vectors. These are the exact locations that will be used to extract the secret pixel values.

Every extracted pixel is added to a 1D vector. The resolution of the secret video (width x and height y) is also transmitted along with the stego video. Thus, the recovered 1D vector can be reformatted into a sequence of frames with a size of $x \times y$, which reconstructs the recovered hidden video. This recovered secret video is then rescaled from its current range to the original intensity range [0,255] per color channel.

IV. EXPERIMENTAL RESULTS

This section presents comparative results of the proposed PixAR scheme versus other video-based steganography techniques which have been recently published in the literature. Robustness of the proposed PixAR scheme is examined against various stego attacks, and a detailed discussion on the effects on the quality of the recovered secret video is presented. Detailed results of the proposed PixAR scheme are also shown when applying it on different cover host videos, and embedding with different secret video segments. The details of the cover and secret videos used are shown in table 1.

A. COMPARATIVE RESULTS

In this section, the proposed PixAR scheme is compared to other video steganography schemes published in the literature. Table 2 compares the proposed PixAR scheme using the Video Quality Metric (VQM) described in [34]. For VQM, the lower the value, the better the quality. Based on [35], VQM values lower than 1 reflect an excellent video quality where degradations will not be visible. Although these competitive schemes are classified as watermarking schemes, and thus have much lower capacity rates than the proposed PixAR scheme, the proposed PixAR scheme was able to achieve better VQM scores.

Table 3 shows a detailed comparison with other video steganography schemes. To make a fair comparison for each scheme to be compared, the length of the cover video was selected to be equal to the compared scheme. Since the proposed PixAR scheme has achieved a much higher capacity rate than other video steganography schemes, two experiments were performed; one with the highest possible capacity achievable by the proposed PixAR scheme, and the other with a similar capacity achieved by the competing scheme.

It is clear from table 3, that the proposed PixAR scheme exceeds other video steganography schemes presented in the table in terms of payload capacity. Furthermore, the quality of the stego video was also a challenge for competing schemes. Actually, in one case, the stego quality achieved by the proposed PixAR scheme has surpassed the competing scheme when the PixAR payload capacity was higher.

This is clear when inspecting the results achieved by Dasgupta *et al.* [10] in table 3. Even though the proposed PixAR scheme embeds with a much higher capacity rate of 16.64 bpp versus 7.98 bpp, the proposed PixAR scheme is able to achieve a better stego quality (imperceptibility) of 42.34 dB versus 34.37 dB.

For all other schemes, when hiding with approximately the same or slightly higher capacities, the imperceptibility achieved by the proposed PixAR scheme is higher than the competing scheme.

For all experiments in this paper, the capacity was calculated using bits-per-pixel (bpp) as follows:

$$bpp = \frac{P_{secret} \times b}{P_{steg}}, \tag{3}$$

where P_{secret} is the total number of secret pixels to be hidden, P_{steg} is the total number of stego video pixels used, and b is the number of bits to represent a pixel (for gray-scale $b = 8$, and for color $b = 24$).

B. ROBUSTNESS

This section discusses the robustness of the proposed PixAR scheme. The scheme was able to resist many types of attacks with high quality extraction. Table 4 shows the performance

TABLE 3. Comparative results expressed as maximum Capacity/PSNR/SSIM values for the various methods. Highest Capacities and PSNR values are emphasized in a bold font.

Method	Number of Cover Frames	Capacity (bpp)	Average PSNR (dB)	Average SSIM
Ke 2012 *et al.* [23]	30	2.4e-4	43.15	-
Sadek 2017 *et al.* [15]	150	0.008	46	-
Dasgupta 2012 *et al.*[9]	182	7.98	44.34	-
PixAR	182	**22.3**	37.38	0.9930
PixAR	182	8.4	**52.92**	0.9996
Dasgupta 2013 *et al.*[10]	107	7.98	34.37	-
Xu 2014 *et al.*[40]	100	7.8e-5	38.33	0.9825
PixAR	107	**16.64**	42.34	1.000
PixAR	107	9.6	**52.16**	0.9996
Ke & Weidong 2013 [22]	300	1.73	42.03	-
PixAR	300	**17.35**	37.78	0.9998
PixAR	300	2.00	**48.28**	1.000
Ramadhan J 2017 *et al.* - DWT [25]	795	0.2720	48.67	-
Ramadhan J 2017 *et al.* - DCT [25]	795	0.2768	49.01	-
Ramadhan J & Khaled Elleithy 2016 [41]	413	5.256	34.66	-
PixAR	795	**16.85**	32.83	0.9949
PixAR	795	1.0	**50.53**	1.000
Ramadhan J and Elleithy 2015 [14]	150	1.2	45.68	-
Hu and KinTak [42] 2011	150	**21.92**	29.15	-
PixAR	150	**16.51**	38.99	1.000
PixAR	150	1.5	**51.44**	1.000

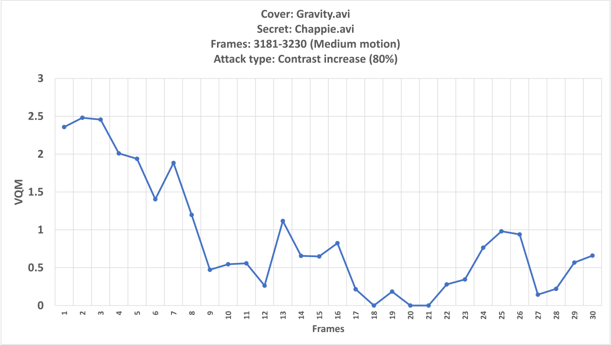


FIGURE 4. Although the stego video was attacked by increasing the brightness by 80%, the VQM value between the recovered and the secret video is lower than 1 for most frames.

of the proposed method against ‘‘Salt & Pepper’’ noise, ‘‘Possion’’ noise, and brightness-increase attacks. The scheme was able to recover the secret video with minimal VQM values. Figure 4 shows that the VQM values between the extracted frames and the original secret frames is approximately zero for all frames even though the increase in the brightness is 80%.

The reason behind this high quality extraction, especially for the ‘‘Salt & Pepper’’ attack, is due to embedding in pixograms rather than embedding spatially. When embedding spatially, any change in even a single coefficient effects the spatial appearance of the image. However, in a pixogram, a change in a single coefficient will not affect the neighborhood pixels in a frame. The effect will be only in a single pixogram which will not be shown clearly since it is a single pixel in each frame.

The second attack to be tested is the tampering attack. Tampering will cause loss in the secret data embedded in the tampered region of the stego video. This will cause a large black area in the recovered video. To prevent having such areas, the proposed PixAR scheme suggests randomizing the secret video frames before embedding. Randomization will scatter and disperse the noise in the extracted frames, which improves the quality of the recovered video.

TABLE 4. Recovering the secret video “Chappie.avi” after attacking the stego video “Gravity.avi” using various attacks.

Attack	Motion	Density	Capacity	VQM		
				MAX	MIN	AVG
Salt & pepper	Slow	0.01	23.02	27.95	22.36	25.34
		0.001	22.78	13.03	4.36	8.4
		0.0001	22.78	8.33	2.46	4.28
	Medium	0.01	16.11	24.09	17.95	20.86
		0.001	16.11	10.89	2.87	6.52
		0.0001	16.11	6.55	0.78	3.07
	Fast	0.01	14.71	20.45	13.88	17.8
		0.001	14.71	7.72	1.49	3.74
		0.0001	14.71	4.26	0	0.73
Brightness	Slow	+30%	23.02	2.87	1.12	1.87
		+50%	22.78	2.87	1.12	1.87
		+80%	22.78	2.87	1.12	1.87
	Medium	+30%	16.11	2.48	0	0.88
		+50%	16.11	2.48	0	0.88
		0.0001	16.11	+80%	0	0.88
	Fast	+30%	14.71	3.41	0	0.11
		+50%	14.71	3.41	0	0.11
		+80%	14.71	3.41	0	0.11
Poisson	Slow	-	23.02	5.80	2.22	3.77
	Medium	-	16.11	5.70	0.65	2.63
	Fast	-	14.71	3.42	0.0	0.12

For randomization, the Linear congruential generator (LCG) technique is utilized. The LCG randomization technique is a class of chaotic mapping transformations used extensively in cryptography. This LCG transformation allows encryption of a signal and the recovery of the original signal back. Recovery depends on knowing three key variables used in the initial LCG transformation, and therefore, can be considered as a secret encryption key without which the receiver will not be able to descramble the extracted secret video frames, thus improving the security of the secret hidden information. More details about LCG randomization can be found in [43] and [44].

TABLE 5. Recovering the secret video "Chappie.avi" with and without using the LCG randomization technique after tampering the "Gravity.avi" stego video with a white box of different sizes. C1 corresponds to hiding with capacity 13.55 bpp (fast motion). C2 corresponds to hiding with capacity 15.33 bpp (medium motion). C3 corresponds to hiding with capacity 22.32 bpp (slow motion). N corresponds to the number of frames with VQM=0 for hiding with C1 capacity rate.

White Box %	Extraction without LCG									
	MAX VQM			MIN VQM			AVG VQM			
	C1	C2	C3	C1	C2	C3	C1	C2	C3	N
20%	7.7	10.0	12.3	0	2.6	2.5	1.63	4.7	5.1	22
40%	12.8	15.3	17.0	0	2.6	2.6	4.97	8.0	8.6	16
60%	15.1	18.5	19.8	0	2.7	2.6	9.01	11.9	12.8	10
80%	18.6	21.2	22.1	0	2.7	2.7	14.3	17.0	17.7	3
White Box %	Extraction with LCG									
	MAX VQM			MIN VQM			AVG VQM			
	C1	C2	C3	C1	C2	C3	C1	C2	C3	N
20%	6.07	7.4	8.2	0	5.8	4.5	3.2	6.6	6.4	1
40%	10.4	14.8	15.1	7.4	11.2	8.7	8.4	12.7	11.8	0
60%	15.6	20.0	23.1	12.0	16.5	14.2	13.7	18.1	17.7	0
80%	21.5	24.8	31.0	18.0	21.1	18.8	19.4	23.0	23.5	0

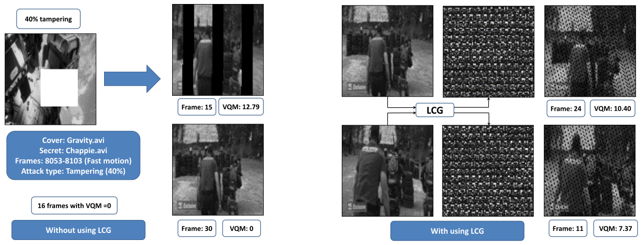


FIGURE 5. Recovering the secret video "Chappie.avi" with and without using LCG randomization technique after tampering the "Gravity.avi" stego video with a white box that covers 40% of each of the stego video frames.

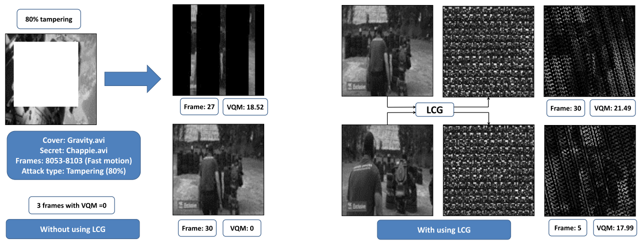


FIGURE 6. Recovering the secret video "Chappie.avi" with and without using LCG randomization technique after tampering the "Gravity.avi" stego video with a white box that covers 80% of each of the stego video frames.

Table 5 shows detailed results when simulating data-loss tampering of the "Gravity.avi" stego video by replacing regions in each frame with white square areas of different sizes, when using LCG and without using LCG. Figures 5 and 6 show detailed tampering experiments when using LCG and without. It is clear that without using LCG, the extracted video will have a black area where secret data is lost. However, an advantage of not using LCG is that many frames can have an VQM value of zero; there is no loss in secret data for those frames. That is because, these secret data are hidden in pixograms located in areas without tampering. This zero VQM extraction would not happen if the scheme was spatially oriented, or hides frame by frame.

TABLE 6. Recovering the secret video "Chappie.avi" after compressing the stego video "Gravity.avi" using various compression techniques under different compression ratios.

Motion	Capacity	Technique	Density	VQM		
				MAX	MIN	AVG
Slow	23.02	M-JPEG	95%	26.59	21.17	23.99
			96%	25.54	20.41	22.75
			97%	23.16	19.54	21.3
			98%	20.32	16.03	18.63
			99%	15.61	13.05	14.60
		M-J2	95%	21.18	18.05	19.59
			96%	17.54	14.88	16.13
			97%	12.26	9.9	11.26
			98%	8.15	5.46	6.92
			99%	6.06	3.1	4.72
		MPEG-4	95%	28.23	22.1	24.0
			96%	28.20	22.0	23.98
			97%	28.0	21.9	23.96
			98%	27.70	21.85	23.92
			99%	27.66	21.77	23.2

TABLE 7. Recovering the secret video "Chappie.avi" after compressing the stego video "Gravity.avi" using various compression techniques under different compression ratios.

Motion	Capacity	Technique	Density	VQM		
				MAX	MIN	AVG
Medium	16.11	M-JPEG	95%	29.54	22.47	26.65
			96%	28.1	20.17	24.77
			97%	25.69	16.85	20.31
			98%	21.41	3.72	11.43
			99%	15.31	1.81	11.28
		M-J2	95%	28.22	25.27	28.55
			96%	22.86	25.24	28.25
			97%	16.98	25.34	28.21
			98%	10.49	21.15	23.46
			99%	6.63	7.01	15.17
		MPEG-4	95%	29.25	26.68	27.97
			96%	29.42	26.11	27.77
			97%	29.38	26.45	27.91
			98%	29.99	26.26	28.12
			99%	29.82	26.75	28.28

The decision whether to use LCG or not depends on the importance of the secret video content. If the content of the whole video is critical, and thus is important to prevent having the black areas (data-loss) in the secret video, then it is recommended to randomize the secret video using LCG before embedding. However, if it is important to recover the secret video with the most number of unaffected frames (having VQM=0), and the content of the video is not very critical, then it is recommended to embed directly without LCG randomization.

The robustness of the proposed PixAR scheme is also examined against compression attacks. For this experiment, three compression techniques have been tested on the stego video generated from the "Gravity.avi" cover video sequence. The three techniques are the Motion JPEG (M-JPEG), Motion JPEG 2000 (MJ2), and MPEG-4 compression techniques. Tables 6, 7, and 8 shows the VQM results obtained after compressing the stego video. These compression techniques exploit temporal redundancies, and remove them to reduce the size of the video file. Due to this fact, the extraction process of the proposed PixAR scheme would be affected, since the proposed scheme hides in temporal redundancies rather than removing them. Thus, it is

TABLE 8. Recovering the secret video "Chappie.avi" after compressing the stego video "Gravity.avi" using various compression techniques under different compression ratios.

Motion	Capacity	Technique	Density	VQM		
				MAX	MIN	AVG
Fast	14.71	M-JPEG	95%	31.66	28.32	29.76
			96%	29.23	25.98	27.64
			97%	26.51	23.02	24.81
			98%	21.99	18.94	20.39
			99%	15.31	13.75	14.58
		M-J2	95%	32.84	30.67	31.84
			96%	31.32	28.53	29.69
			97%	25.27	23.59	24.31
			98%	13.17	11.82	12.67
			99%	3.51	2.25	2.46
		MPEG-4	95%	33.57	32.36	32.96
			96%	33.57	32.36	32.96
			97%	33.26	32.6	32.93
			98%	33.29	32.45	32.87
			99%	32.98	32.37	32.68

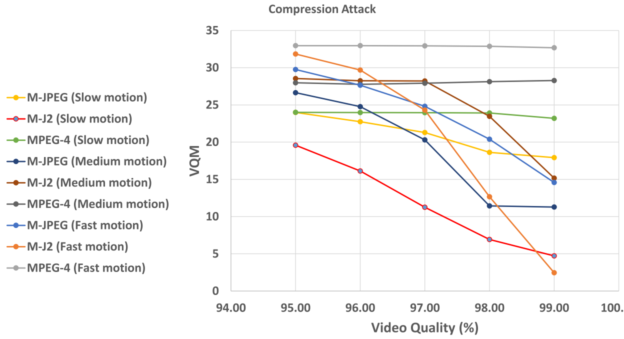
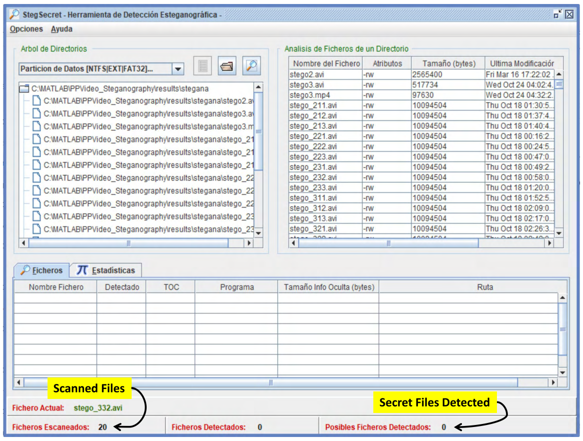


FIGURE 7. The VQM values between the recovered secret video "Chappie.avi" and the stego video "Gravity.avi" after compressing the stego video "Gravity.avi" using various compression techniques under different compression ratios.

expected that compressing the stego video will destroy the secret video. However, from tables 6, 7, and 8, the VQM results obtained indicate that the proposed PixAR technique was able to resist these types of compressions to some degree. A reasonable explanation is that the proposed PixAR scheme hides in temporal 1D pixograms rather than temporal blocks as in the former compression schemes. Figure 7 summarizes the compression results in a graph. The graph clearly shows that the VQM value of the extracted video increases with decreasing video quality (more compression).

Since PixAR is a video steganography technique, the proposed PixAR scheme must be tested for detectability of the secret hidden information using a security-aspect-based test. In this experiment the security level of the proposed PixAR scheme is examined by checking whether the existence of the secret video frames can be detected or not. To apply this experiment, 20 stego videos with different motion speeds have been examined using "StegSecret" software.[2] Figure 8 shows the output of the program. The figure shows that the program could not find any traces of hidden secret video frames in the PixAR stego videos tested.


StegSecret - Herramienta de Detección Esteganográfica -

Opciones
Ayuda
Arbol de Directorios
Partición de Datos [NTFS|EXT|FAT32]...
Analisis de Ficheros de un Directorio
Ficheros
Estadisticas

C:\MATLAB\PPVideo_Steganography\results\stegana
C:\MATLAB\PPVideo_Steganography\results\stegana\stego2.av
C:\MATLAB\PPVideo_Steganography\results\stegana\stego3.av
C:\MATLAB\PPVideo_Steganography\results\stegana\stego3.m
C:\MATLAB\PPVideo_Steganography\results\stegana\stego_21
C:\MATLAB\PPVideo_Steganography\results\stegana\stego_21
C:\MATLAB\PPVideo_Steganography\results\stegana\stego_21
C:\MATLAB\PPVideo_Steganography\results\stegana\stego_22
C:\MATLAB\PPVideo_Steganography\results\stegana\stego_22
C:\MATLAB\PPVideo_Steganography\results\stegana\stego_22
C:\MATLAB\PPVideo_Steganography\results\stegana\stego_23
C:\MATLAB\PPVideo_Steganography\results\stegana\stego_23

Nombre del Fichero	Atributos	Tamaño (bytes)	Ultima Modificaciór
stego2.avi	-rw	2565400	Fri Mar 16 17:22:02 ...
stego3.avi	-rw	517734	Wed Oct 24 04:02:4...
stego3.mp4	-rw	97630	Wed Oct 24 04:32:2...
stego_211.avi	-rw	10094504	Thu Oct 18 01:30:5...
stego_212.avi	-rw	10094504	Thu Oct 18 01:37:4...
stego_213.avi	-rw	10094504	Thu Oct 18 01:40:4...
stego_221.avi	-rw	10094504	Thu Oct 18 00:16:2...
stego_222.avi	-rw	10094504	Thu Oct 18 00:24:5...
stego_223.avi	-rw	10094504	Thu Oct 18 00:47:0...
stego_231.avi	-rw	10094504	Thu Oct 18 00:49:2...
stego_232.avi	-rw	10094504	Thu Oct 18 00:58:0...
stego_233.avi	-rw	10094504	Thu Oct 18 01:20:0...
stego_311.avi	-rw	10094504	Thu Oct 18 01:52:5...
stego_312.avi	-rw	10094504	Thu Oct 18 02:09:0...
stego_313.avi	-rw	10094504	Thu Oct 18 02:17:0...
stego_321.avi	-rw	10094504	Thu Oct 18 02:26:3...

Nombre Fichero	Detectado	TOC	Programa	Tamaño Info Oculta (bytes)	Ruta

Scanned Files

Fichero Actual: stego_332.avi

Ficheros Escaneados: 20

Ficheros Detectados: 0

Secret Files Detected

Posibles Ficheros Detectados: 0

FIGURE 8. Detectability test using StegSecret software.

TABLE 9. Recovering the secret video "Chappie.avi" after attacking the stego video "Gravity.avi" using different temporal-based attacks.

Attack	Motion	Percentage	Capacity	VQM			BFM
				MAX	MIN	AVG	
Re-Sampling	Slow	-50%	23.02	2.92	1.76	2.3	1.97
		-75%	22.78	5.38	3.39	4.51	2.06
		-85%	22.78	6.87	4.25	5.44	2.19
	Medium	-50%	16.11	2.7	1.76	2.36	0.49
		-75%	16.11	5.33	3.39	4.57	0.49
		-85%	16.11	6.14	4.25	5.43	0.63
	Fast	-50%	14.71	2.7	1.76	2.36	0.56
		-75%	14.71	5.33	3.39	4.58	0.51
		-85%	14.71	6.14	4.25	5.43	0.70
Drop Frame	Slow	10%	23.02	11.84	6.37	8.32	2.33
		25%	22.78	17.22	10.35	12.53	2.23
		50%	22.78	22.61	15.01	17.33	2.54
		75%	22.78	27.86	19.12	21.26	1.69
	Medium	10%	16.11	27.61	19.51	24.52	4.61
		25%	16.11	28.7	22.39	26.3	4.25
		50%	16.11	28.55	21.37	25.3	2.95
		75%	16.11	28.87	21.7	26.32	2.88
	Fast	10%	14.71	32.81	29.69	31.44	5.85
		25%	14.71	33.88	30.19	31.88	5.51
		50%	14.71	32.02	28.06	30.53	4.10
		75%	14.71	31.41	28.36	30.04	2.8

Figure 9 illustrates visually maximum and minimum VQM values for an extracted secret video segment from an MJ2 compressed stego video segment. It is clear from the figure that for this MJ2 compression technique the PixAR scheme was able to extract the hidden video with acceptable quality at 99% compression. The MPEG-4 compression technique, on the other hand, was able to destroy the extracted secret video, as is clear from the extremely high MPEG-4 VQM values shown in tables 6, 7, and 8.

Finally, being a steganography scheme, the robustness of the technique against some video temporal-based attacks must also be examined. Table 9 shows the results obtained after attacking with "Re-Sampling" and "Drop-Frame" attacks. In addition to VQM, another metric was used to evaluate the robustness in this experiment which is the BFM metric.

The proposed PixAR scheme deals with "Drop-Frame" attacks by interpolating the missing frames. If the communication is done through an insecure or unreliable channel,

[2] http://stegsecret.sourceforge.net/

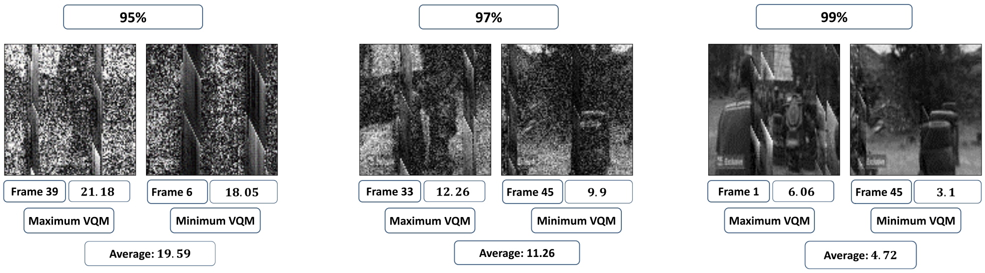


FIGURE 9. Recovering the secret video "Chappie.avi" after compressing the stego video "Gravity.avi" using Motion JPEG 2000 (MJ2).

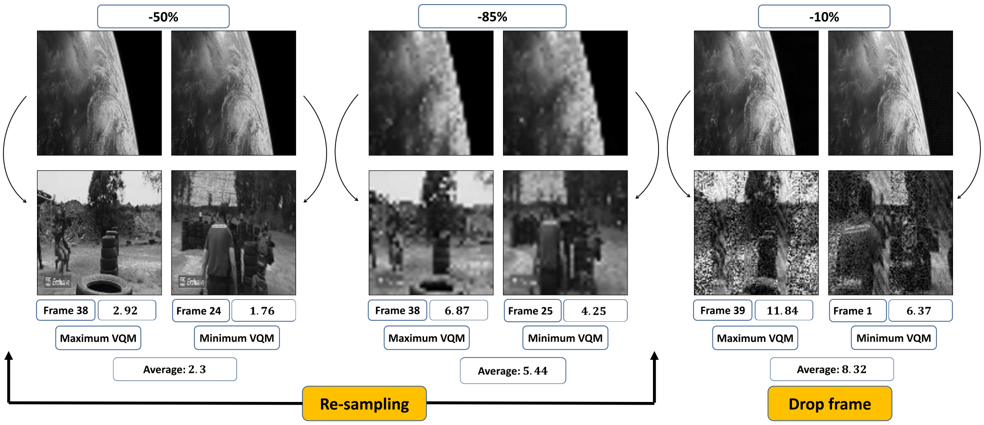


FIGURE 10. Recovering the secret video "Chappie.avi" after attacking the stego video "Gravity.avi" using re-sampling and drop frame attacks.

then it is recommended to send the "frame order value" along with the frame itself. This will help the receiver to estimate the location of the dropped frames if any. The proposed PixAR scheme replaces each missing frame with a frame with pixels having (-1) values. Then, instead of simply interpolate the missing frames using neighboring frames, linear interpolation is done within each pixogram segment containing an (-1) value independently. This insures having better interpolation since values within a pixogram segment are homogeneous and easy to interpolate. Without this "Drop-Frame" attack treatment, the proposed PixAR scheme would not be able to extract the secret video. However, figure 10 shows that the proposed scheme was able to extract the secret video for some dropped-frame percentages.

C. DETAILED EVALUATION OF THE PROPOSED SCHEME

Table 10 shows detailed experimental tests using some segments of "Gravity.avi", "Chappie.avi", and "WW2.avi" videos used as the cover video. The secret video to be embedded are some segments from "Chappie.avi", "WW2.avi", and "Train.avi". The threshold value was set to be $\mathcal{T}_{seg} = 26$. Each cover-secret pair have three experiments based on the motion presented in the cover video (slow, medium, fast).

For video sequences, and from the perspective of a pixogram, motion can be considered as the temporal correlation (i.e. the correlation of the gray-levels over time). For instance, slow motion can be thought of as a high temporal correlation component, since pixels do not change much over time, and the variance of their gray-level value will be low in the pixogram. On the other hand, for a pixogram, rapid motion in the video can be considered as a low temporal correlation component.

Three different segments are used from the "Gravity.avi" cover video representing low, mid, and high motion segments. Frames from [37-87] represents a slow motion segment, frames [3181-3230] represents a medium motion segment, and a fast motion segment is represented by frames [85053-8103].

It is clear from the results shown in table 10 that the slow motion segments have consistantly achieved the best capacity rates as well as the best stego quality values, mainly due to the inherent strong energy compaction property of the DCT for highly correlated signals. Payload capacities have reached an embedding rate of 22.32 bpp for the "Gravity.avi" cover video, which is the highest payload capacity reached across other cover video segments. The stego quality (imperceptibility) is related to the secret video used. The "Train.avi" secret video achieved the best average VQM value of 0.086 when using "Chappie.avi" as the cover video.

Slow and medium motion segments of "Chappie.avi" have achieved the highest payload capacity of 22.10 bpp and 19.57 bpp, and the highest visual stego quality, respectively.

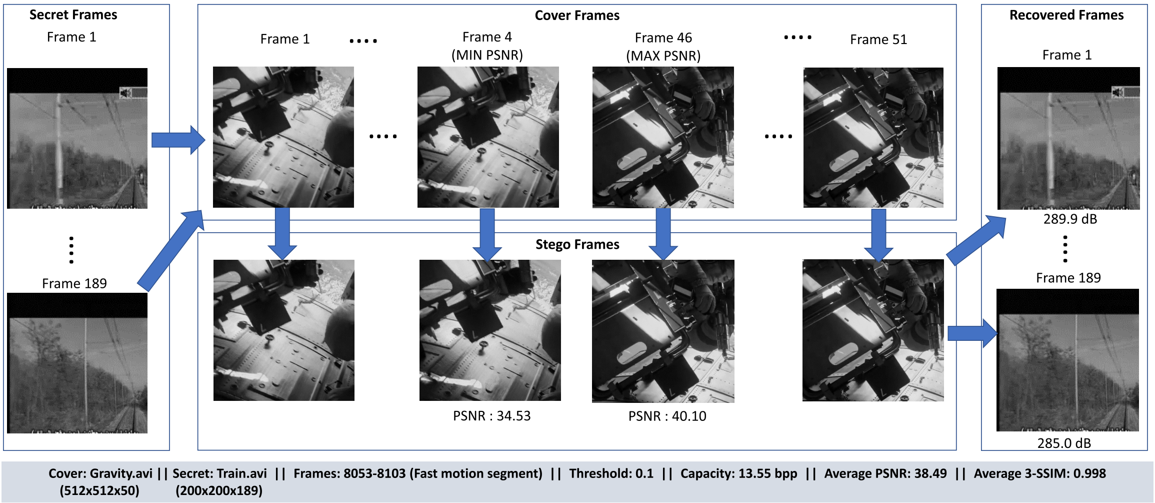

Secret Frames
Frame 1
Frame 189
Cover Frames
Frame 1
Frame 4
(MIN PSNR)
Frame 46
(MAX PSNR)
Frame 51
Stego Frames
PSNR : 34.53
PSNR : 40.10
Recovered Frames
Frame 1
289.9 dB
Frame 189
285.0 dB
Cover: Gravity.avi (512x512x50) || Secret: Train.avi (200x200x189) || Frames: 8053-8103 (Fast motion segment) || Threshold: 0.1 || Capacity: 13.55 bpp || Average PSNR: 38.49 || Average 3-SSIM: 0.998

FIGURE 11. Showing the cover and stego frames of the highest and lowest PSNR when using a fast motion segment from "Gravity.avi". The secret and recovered frames of "Train.avi" are also shown.

TABLE 10. Application of the proposed PixAR steganography scheme showing capacities, VQM, 3-SSIM, BFM, and PSNR of various Stego video segments.

Cover	Secret	Motion	Frames	Capacity	VQM			3-SSIM			BFM			PSNR
					MAX	MIN	AVG	MAX	MIN	AVG	Before	After	Delta	AVG
Gravity	*Chappie*	Slow	37-87	**22.32**	1.149	0.237	0.504	0.999	0.995	0.9986	0.0698	0.59	0.5202	38.1
		Medium	3181-3230	15.33	1.179	0.458	0.628	0.996	0.994	0.9967	0.32	0.63	0.31	35.57
		Fast	8053-8103	13.55	0.462	0.107	0.212	0.999	0.997	0.997	0.55	0.59	0.04	37.63
	WW2	Slow	37-87	**22.32**	1.24	0.492	0.71	0.999	0.994	0.998	0.03	0.84	0.81	36.76
		Medium	3181-3230	15.33	1.241	0.492	0.723	0.998	0.994	0.9964	0.3	0.67	0.37	33.1
		Fast	8053-8103	13.55	0.434	0.103	0.209	0.999	0.997	0.997	0.55	0.59	0.04	33.1
	Train	Slow	37-87	**22.32**	1.330	0.227	0.480	0.999	0.996	0.999	0.08	0.68	0.6	36.76
		Medium	3181-3230	15.33	2.39	0.0156	0.660	0.999	0.995	0.998	0.10	0.60	0.50	33.39
		Fast	8053-8103	13.55	1.05	0.240	0.550	0.999	0.996	0.998	0.10	0.77	0.67	38.49
Chappie	*Chappie*	Slow	1008-1058	**22.10**	0.326	0.08	0.13	0.999	0.998	0.999	0.87	0.87	0	37.67
		Medium	1-50	19.57	0.123	0.062	0.087	1.000	0.999	1.000	0.93	0.93	0	39.66
		Fast	1632-1682	15.89	1.01	0.107	0.330	0.998	0.986	0.995	5.46	5.46	0	37.63
	WW2	Slow	1008-1058	**22.10**	0.139	0.0513	0.770	1.00	1.00	1.00	0.840	0.840	0	37.47
		Medium	1-50	19.57	0.299	0.0498	0.125	1.00	0.999	1.00	0.74	0.74	0	39.60
		Fast	1632-1682	15.89	0.154	0.081	0.11	0.999	0.999	0.999	0.460	0.460	0	29.5
	Train	Slow	1008-1058	**22.10**	0.326	0.704	0.13	0.999	0.998	0.999	0.87	0.87	0	37.43
		Medium	1-50	19.57	0.121	0.061	0.086	1.000	0.999	1.000	0.93	0.93	0	39.66
		Fast	1632-1682	15.89	1.01	0.108	0.336	0.998	0.986	0.995	5.46	5.47	0.01	29.51
WW2	*Chappie*	Slow	24-74	17.32	0.775	0.313	0.47	0.999	0.997	0.998	1.43	1.36	-0.07	35.58
		Medium	504-554	14.00	0.709	0.236	0.440	0.999	0.995	0.997	3.26	3.19	-0.07	36.65
		Fast	729-842	13.77	0.686	0.225	0.35	0.999	0.998	0.999	0.850	0.910	0.06	34.86
	WW2	Slow	24-74	17.32	0.734	0.311	0.450	0.999	0.998	0.999	1.43	1.37	-0.06	35.41
		Medium	504-554	14.00	0.745	0.252	0.470	0.999	0.996	0.998	3.26	3.16	-0.10	36.91
		Fast	729-842	13.77	0.723	0.262	0.420	0.999	0.998	0.999	0.850	0.900	0.05	34.78
	Train	Slow	24-74	17.32	0.795	0.311	0.43	0.999	0.997	0.998	1.43	1.35	-0.08	35.52
		Medium	504-554	14.00	0.701	0.268	0.45	0.999	0.996	0.997	3.26	3.15	0.11	34.35
		Fast	729-842	13.77	0.604	0.264	0.38	0.999	0.998	0.999	0.850	0.850	0	34.92

Embedding "Train.avi" in "Chappie.avi" for medium motion segments recorded the best PSNR value of 39.66 dB. Figure 11 shows example "Gravity.avi" cover and stego frames of the highest and lowest PSNR values when using fast motion segments of the "Gravity.avi" cover video and embedding the "Train.avi" secret video.

Figure 12 shows the PSNR value between the cover and stego video for all frames, when hiding the "Train.avi" secret video in the "Gravity.avi" cover video. The lowest and highest PSNR values (34.53 dB and 40.10 dB respectively) are emphasized with a red circle. The average PSNR was calculated to be 38.49 dB.

The proposed scheme is also evaluated using some video-based quality metrics such as Video Quality Metric (VQM) [34], 3-Component SSIM Index (3-SSIM) [45], and Brightness Flickering Metric (BFM) [46].

For VQM, the lower the value, the better the quality. Based on table 10, all of the average VQM values obtained are less than 1, with a highest value of 0.770. For maximum VQM values, only 33% of the values are above 1. The worst VQM value obtained was 2.39 when hiding "Train.avi" in "Gravity.avi" for medium motion segment. Nevertheless, this *worst* VQM obtained in table 10, using the proposed PixAR scheme, is better than the *best* VQM values for most

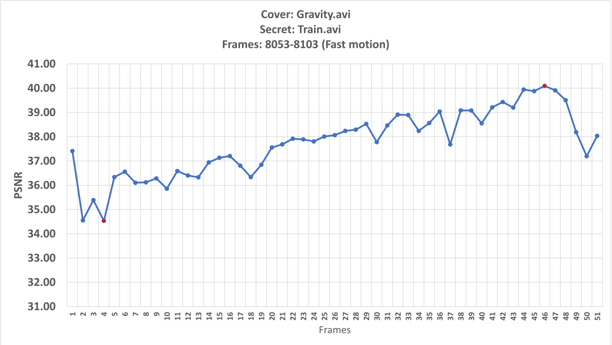


FIGURE 12. Showing the PSNR value between the cover and stego video for all frames, when hiding the "Train.avi" secret video in the "Gravity.avi" cover video. The lowest and highest PSNR values (34.53 dB and 40.10 dB respectively) are emphasized with a red circle. The average PSNR was calculated to be 38.49 dB.

other techniques mentioned in table 2, except for the technique proposed by Chen and Leung [36] with a VQM value of 1.3.

The 3-SSIM index is geared towards the human visual system model in its calculations rather than pure mathematical error calculations between two signals, and thus it is a preferable quality metric tool for video signals. It is clear from table 10, that the proposed PixAR scheme was able to achieve high 3-SSIM values. The lowest average value was 0.9964 while three entries have an average 3-SSIM value of 1. A 3-SSIM value of 1 indicates that the two video signals are identical.

The reason for these high 3-SSIM values, when hiding with high capacity ratios, is that the proposed PixAR scheme hides in large high-frequency DCT areas of homogeneous segments of a pixogram, segmented by the 1D segment-growing algorithm described earlier. Hiding in these areas results in stego media with very low degradation, which can be easily tolerated by the human visual system.

The BFM is a no-reference metric made to measure the flickering quantity between adjacent frames of the video. The value is the modulus of difference between the mean brightness values of current and previous frames. In table 10, the "Before" column is the BFM value of the cover video segment and the "After" column is the BFM value of the stego video segment. The "Delta" column is simply the difference between the BFM value of the stego video segment and the BFM value of the cover video segment. This "Delta" value can be thought of as the flicker-noise quantity produced due to the hiding process. A noticeable fact from the table is that 14 delta results (out of 27) have a value of zero or less. This is an indication that the technique was able to hide without adding any flickering noise.

D. COMPARATIVE RESULTS WITH IMAGE-BASED SCHEMES

TABLE 11. Comparative results expressed as maximum Capacity/PSNR values for the various extended image-based methods using "Gravity.avi" as the cover video, and "Chappie.avi" as secret video. The segment used in the experiment was the slow segment (frames 37-87). Maximum Capacity/PSNR values achieved by each scheme are reported and highest capacities and PSNR values are emphasized in a bold font.

Method	Capacity	PSNR
Rabie & Kamel (2016) [16] FB-GAR	**21.27**	26.6
Rabie & Kamel (2016) [17] QTAR	19.54	29.54
Rabie *et. al.* (2017) [18] CF-FB-GAR	21.45	33.52
Rabie *et. al.* (2017) [18] CF-QTAR	**21.30**	34.07
Rabie & Baziyad (2017) [44] CF-LPAR 3 points	11.02	**38.85**
Rabie & Baziyad (2017) [44] CF-LPAR 5 points	13.43	37.18
PixAR	**22.32**	**38.1**

The proposed PixAR scheme demonstrated very high payload capacities compared to other video-based steganography schemes, as was shown in section IV-A. The purpose of this section is to evaluate the proposed scheme against very high payload capacity image-based steganography schemes.

To perform this comparison, the high payload image-based schemes reported in [16]–[18] and [44] were extended to work as video steganography schemes. The idea is to adapt these schemes for hiding the secret video frames in each individual frame of a cover video separately. Table 11 presents a comparison with the extended versions of these high capacity image steganography schemes. Although these competitive methods are classified as top hiding capacity schemes in the literature, the proposed PixAR scheme has surpassed these methods in both hiding capacity and imperceptibility when testing all methods on the same cover video.

Many movies have segments where there is no motion. This can be a movie title at the start of a video for instance. In special cases, most of the video will have no motion at all, as in surveillance videos. Thus, zero-motion video segments should also be addressed as there are many scenarios where the video will have many of its segments with zero-motion. Therefore, this section investigates the viability of such videos as a cover video, and tests the performance of the proposed PixAR scheme against high payload image-based steganography schemes recently published in the literature.

These image-based schemes are extended to work as zero-motion video steganography schemes. Table 12 presents a detailed experiment showing highest bpp and PSNR values obtained by each scheme when using zero-motion cover video segments. The cover videos used in this comparison were the same cover images used by these image-based schemes repeated 100 times to form zero-motion videos.

Table 13 investigates the hiding capacity limits of the proposed PixAR scheme for highly uncorrelated cover images in comparison to these high payload capacity image-based hiding schemes. To perform the comparison, the video to be used as a cover video is composed of a 100 repeated frames of the "Zebras" cover image used in [18]. The reason for selecting the "Zebras" image is that many high capacity techniques have used it as a cover image, and the results are available. The other reason is that this image is a challenging cover image for many DCT image-based steganography schemes since this image has a highly uncorrelated content. The secret video to be used is the "Chappie.avi" video.

TABLE 12. Comparative results of the proposed PixAR scheme with various image-based methods for zero-motion cover videos of a 100 frames in length. Maximum Capacity/PSNR values achieved by each scheme are reported and highest capacities and PSNR values are emphasized in a bold font.

Method	Capacity	PSNR
Rabie & Kamel (2015) [5] FBAR	**20.22 bpp**	25 dB
Rabie & Kamel (2016) [16] FB-GAR	**20.83 bpp**	27 dB
Rabie & Kamel (2016) [17] QTAR *(Max. PSNR)*	15.17 bpp	35 dB
Rabie & Kamel (2016) [17] QTAR *(Max. Capacity)*	**21.01 bpp**	27 dB
Rabie *et. al.* (2017) [18] CF-FB-GAR *(Max. PSNR)*	**19.54 bpp**	35.03 dB
Rabie *et. al.* (2017) [18] CF-FB-GAR *(Max. Capacity)*	**22.43 bpp**	28.49 dB
Rabie *et. al.* (2017) [18] CF-QTAR *(Max. PSNR)*	**19.88 bpp**	35.02 dB
Rabie *et. al.* (2017) [18] CF-QTAR *(Max. Capacity)*	**22.70 bpp**	28.15 dB
Rabie & Baziyad (2017) [44] CF-LPAR 3 points *(Max. PSNR)*	18.1 bpp	**44.6 dB**
Rabie & Baziyad (2017) [44] CF-LPAR 3 points *(Max. Capacity)*	**19.3 bpp**	29.1 dB
Rabie & Baziyad (2017) [44] CF-LPAR 5 points *(Max. PSNR)*	**18.8 bpp**	**45.2 dB**
Rabie & Baziyad (2017) [44] CF-LPAR 5 points *(Max. Capacity)*	**19.5 bpp**	32.0 dB
PixAR	**23.76 bpp**	**54.91 dB**

TABLE 13. Comparative results expressed as maximum Capacity/PSNR values for the various image-based methods using 100 repeated frames of the "Zebras" image as the cover video. Highest Capacities and PSNR values are emphasized in a bold font.

Method	Capacity	PSNR
Rabie & Kamel (2016) [16] FB-GAR	15.24	23.31
Rabie & Kamel (2016) [17] QTAR	**15.61**	27.95
Rabie *et. al.* (2017) [18] CF-FB-GAR	15.4	28.5
Rabie *et. al.* (2017) [18] CF-QTAR	**16.1**	28.27
Rabie & Baziyad (2017) [44] CF-LPAR 3 points	10.0	**40.0**
Rabie & Baziyad (2017) [44] CF-LPAR 5 points	**17.2**	25.3
PixAR	**23.76**	**54.91**

It is clear from tables 12 and 13 that the proposed PixAR scheme has performed optimally, and surpassed other schemes in terms of hiding capacity and imperceptibility when the cover video was a 100 frames of zero-motion. The capacity reached 23.76 bpp (for RGB color cover videos), which is equivalent to hiding a secret video with a size of 99% of the cover video. The imperceptibility level also reached 54.91 dB.

Table 13 clearly manifests the advantages of the proposed schemes. Since the cover video has spatially highly uncorrelated content, all high capacity schemes presented in the table have achieved lower capacity and imperceptibility values. This is mainly because these schemes are based on the 2D-DCT, and they are able to achieve these high capacities due to the energy compaction property of the DCT, but only for highly correlated cover images. The lower the spatial correlation in a cover image, the less the number of insignificant (redundant) DCT coefficients that will be available for embedding, which leads to lower capacity.

However, in the proposed PixAR scheme, the spatial frequency content of individual frames of a video is not an issue. Simply put, a pixogram has the property of converting highly uncorrelated (non-stationary) spatial areas of individual frames of a video scene into highly correlated (statistically stationary) temporal segments by making use of the temporal correlation between frames of the same scene in a given video segment.

V. CONCLUSIONS

In this paper the concept of a *pixogram* was introduced and implemented into a high payload video steganography scheme named "Pixogram Adaptive Region" (PixAR). The proposed PixAR scheme allows for a new perspective for video steganography by investigating the temporal changes that take place at the individual pixel level across frames of a video segment. The advantage of this pixogram based technique lies in the fact that a pixogram has the property of converting highly uncorrelated spatial areas of individual frames of a video scene into highly correlated temporal segments by making use of the temporal correlation between frames of the same scene in a given video segment. Experimental results have demonstrated the effectiveness of this new approach for increased payload capacity while maintaining visual fidelity of the stego-video as compared to competing video steganography schemes. The robustness of the proposed PixAR scheme was also tested against several spatial and temporal-based attacks with acceptable performance, although facing particular difficulty resisting compression attacks.

REFERENCES

[1] S. D. Malaby *et al.*, "Cross platform application control in an interactive, multi-platform video network," U.S. Patent 8 789 124 B1, Jul. 22, 2014.

[2] J. P. L. Velasco, A. Authier, D. J. Bermejo, J. M. M. García, and N. S. Almodóvar, "Blind quality algorithm to analyze streaming video contents in 5g networks," in *Proc. Int. Conf. WWW/Internet Appl. Comput. (IADIS)*, 2017.

[3] X.-L. Liu, C.-C. Lin, and S.-M. Yuan, "Blind dual watermarking for color images' authentication and copyright protection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 5, pp. 1047–1055, May 2016.

[4] M. Asikuzzaman and M. R. Pickering, "An overview of digital video watermarking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 9, pp. 2131–2153, Sep. 2017.

[5] T. Rabie and I. Kamel, "On the embedding limits of the discrete cosine transform," *Multimedia Tools Appl.*, vol. 75, no. 10, pp. 5939–5957, 2016.

[6] F. Y. Shih, *Multimedia Security: Watermarking, Steganography, and Forensics*. Boca Raton, FL, USA: CRC Press, 2012.

[7] M. M. Sadek, A. S. Khalifa, and M. G. Mostafa, "Video steganography: A comprehensive review," *Multimedia Tools Appl.*, vol. 74, no. 17, pp. 7063–7094, 2015.

[8] M. Shirali-Shahreza, "A new method for real-time steganography," in *Proc. 8th Int. Conf. Signal Process.*, vol. 4, 2006. doi: 10.1109/ICOSP.2006.345954.

[9] K. Dasgupta, J. Mandal, and P. Dutta, "Hash based least significant bit technique for video steganography (HLSB)," *Int. J. Secur., Privacy Trust Manage.*, vol. 1, no. 2, pp. 1–11, 2012.

[10] K. Dasgupta, J. K. Mondal, and P. Dutta, "Optimized video steganography using genetic algorithm (GA)," *Procedia Technol.*, vol. 10, pp. 131–137, 2013. doi: 10.1016/j.protcy.2013.12.345.

[11] S. Singh and G. Agarwal, "Hiding image to video: A new approach of LSB replacement," *Int. J. Eng. Sci. Technol.*, vol. 2, no. 12, pp. 6999–7003, 2010.

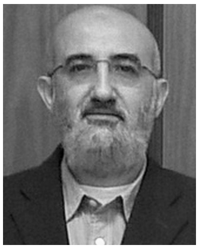
[12] K.-C. Chang, C.-P. Chang, P. S. Huang, and T.-M. Tu, "A novel image steganographic method using tri-way pixel-value differencing," *J. Multimedia*, vol. 3, no. 2, pp. 37–44, 2008.

[13] A. Sherly and P. Amritha, "A compressed video steganography using TPVD," *Int. J. Database Manage. Syst.*, vol. 2, no. 3, pp. 764–766, 2010.

[14] R. J. Mstafa and K. M. Elleithy, "A high payload video steganography algorithm in DWT domain based on BCH codes (15, 11)," in *Proc. Wireless Telecommun. Symp. (WTS)*, Apr. 2015, pp. 1–8.

[15] M. M. Sadek, A. S. Khalifa, and M. G. Mostafa, "Robust video steganography algorithm using adaptive skin-tone detection," *Multimedia Tools Appl.*, vol. 76, no. 2, pp. 3065–3085, 2017.

[16] T. Rabie and I. Kamel, "High-capacity steganography: A global-adaptive-region discrete cosine transform approach," *Multimedia Tools Appl.*, vol. 76, no. 5, pp. 6473–6493, 2017.
[17] T. Rabie and I. Kamel, "Toward optimal embedding capacity for transform domain steganography: A quad-tree adaptive-region approach," *Multimedia Tools Appl.*, vol. 76, no. 6, pp. 8627–8650, 2017.
[18] T. Rabie, I. Kamel, and M. Baziyad, "Maximizing embedding capacity and stego quality: Curve-fitting in the transform domain," *Multimedia Tools Appl.*, vol. 77, no. 7, pp. 8295–8326, 2018.
[19] T. Rabie, M. Baziyad, and I. Kamel, "Enhanced high capacity image steganography using discrete wavelet transform and the Laplacian pyramid," *Multimedia Tools Appl.*, vol. 77, no. 18, pp. 23673–23698, 2018.
[20] T. Stutz and A. Uhl, "A survey of H.264 AVC/SVC encryption," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 3, pp. 325–339, Mar. 2012.
[21] Y. Tew and K. Wong, "An overview of information hiding in H.264/AVC compressed video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 2, pp. 305–319, Feb. 2014.
[22] N. Ke and Z. Weidong, "A video steganography scheme based on H.264 bitstreams replaced," in *Proc. 4th IEEE Int. Conf. Softw. Eng. Service Sci. (ICSESS)*, May 2013, pp. 447–450.
[23] K. Liao, S. Lian, Z. Guo, and J. Wang, "Efficient information hiding in H.264/AVC video coding," *Telecommun. Syst.*, vol. 49, no. 2, pp. 261–269, 2012.
[24] J. Mansouri and M. Khademi, "An adaptive scheme for compressed video steganography using temporal and spatial features of the video signal," *Int. J. Imag. Syst. Technol.*, vol. 19, no. 4, pp. 306–315, 2009.
[25] R. J. Mstafa, K. M. Elleithy, and E. Abdelfattah, "A robust and secure video steganography method in DWT-DCT domains based on multiple object tracking and ECC," *IEEE Access*, vol. 5, pp. 5354–5365, 2017.
[26] V. K. Agrawal, "Perceptual watermarking of digital video using the variable temporal length 3D-DCT," IIT, Kanpur, India, Tech. Rep., 2007.
[27] V. N. Manju and A. L. Fred, "AC coefficient and K-means cuckoo optimisation algorithm-based segmentation and compression of compound images," *IET Image Process.*, vol. 12, no. 2, pp. 218–225, Feb. 2017.
[28] M. Wang, C. Luo, B. Ni, J. Yuan, J. Wang, and S. Yan, "First-person daily activity recognition with manipulated object proposals and non-linear feature fusion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 10, pp. 2946–2955, Oct. 2017.
[29] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Trans. Comput.*, vol. C-100, no. 1, pp. 90–93, Jan. 1974.
[30] J. Bracamonte, M. Ansorge, F. Pellandini, and P.-A. Farine, "Low complexity image matching in the compressed domain by using the DCT-phase," in *Proc. 6th COST*, vol. 276, 2004, pp. 88–93.
[31] J. Bracamonte, M. Ansorge, F. Pellandini, and P.-A. Farine, "Efficient compressed domain target image search and retrieval," in *Image and Video Retrieval*. Berlin, Germany: Springer, 2005, pp. 154–163.
[32] I. Ito and H. Kiya, "DCT sign-only correlation with application to image matching and the relationship with phase-only correlation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, vol. 1, Apr. 2007, pp. I-1237–I-1240.
[33] T. Rabie, "Color-secure digital image compression," *Multimedia Tools Appl.*, vol. 76, no. 15, pp. 16657–16679, 2017.
[34] F. Xiao *et al.*, "DCT-based video quality evaluation," Final Project EE392J 769, 2000.
[35] K.-H. Lee, S. T. Trong, B.-G. Lee, and Y.-T. Kim, "QoS-guaranteed IPTV service provisioning in home network with IEEE 802.11e wireless LAN," in *Proc. IEEE Workshop Netw. Oper. Manage.*, Apr. 2008, pp. 71–76.
[36] S. Chen and H. Leung, "A temporal approach for improving intra-frame concealment performance in H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 3, pp. 422–426, Mar. 2009.
[37] K.-L. Chung, Y.-H. Huang, P.-C. Chang, and H.-Y. M. Liao, "Reversible data hiding-based approach for intra-frame error concealment in H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 11, pp. 1643–1647, Nov. 2010.
[38] D. Xu, R. Wang, and J. Wang, "Prediction mode modulated data-hiding algorithm for H.264/AVC," *J. Real-Time Image Process.*, vol. 7, no. 4, pp. 205–214, 2012.
[39] D. Xu, R. Wang, and Y. Q. Shi, "An improved reversible data hiding-based approach for intra-frame error concealment in H.264/AVC," *J. Vis. Commun. Image Represent.*, vol. 25, no. 2, pp. 410–422, 2014.
[40] D. Xu, R. Wang, and Y. Q. Shi, "Data hiding in encrypted h.264/AVC video streams by codeword substitution," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 4, pp. 596–606, Apr. 2014.
[41] R. J. Mstafa and K. M. Elleithy, "A video steganography algorithm based on Kanade-Lucas-Tomasi tracking algorithm and error correcting codes," *Multimedia Tools Appl.*, vol. 75, no. 17, pp. 10311–10333, 2016.
[42] S. D. Hu and K. T. U, "A novel video steganography based on non-uniform rectangular partition," in *Proc. IEEE 14th Int. Conf. Comput. Sci. Eng. (CSE)*, Aug. 2011, pp. 57–61.
[43] C. Fontaine, "Linear congruential generator," in *Proc. Encyclopedia Cryptogr. Secur.*, 2011, p. 721.
[44] T. Rabie and M. Baziyad, "Visual fidelity without sacrificing capacity: An adaptive Laplacian pyramid approach to information hiding," *J. Electron. Imag.*, vol. 26, no. 6, p. 063001, 2017. doi: 10.1117/1.JEI.26.6.063001.
[45] C. Li and A. C. Bovik, "Three-component weighted structural similarity index," *Proc. SPIE*, vol. 7242, p. 72420Q, Jan. 2009. doi: 10.1117/12.811821.
[46] R. Dantu, "Measuring vital signs using smart phones," Ph.D. dissertation, Dept. Comput. Sci. Eng., Univ. North Texas, Denton, TX, USA, 2010.



TAMER RABIE received the M.Sc. degree from the Department of Electrical and Computer Engineering, University of Calgary, in 1993, and the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Toronto, in 1999. He is currently an Associate Professor of computer engineering with the Electrical and Computer Engineering Department, University of Sharjah. His current research interests include digital image processing, computer vision, image/speech watermarking, and intelligent robotic systems.

MOHAMMED BAZIYAD received the bachelor's degree in network engineering from the Canadian University of Dubai, in 2015. He is currently a Research Assistant with the Autonomous Robotics and Active Vision Research Group, University of Sharjah. He is currently involved in the research projects related to robotics. His research interests include information security, steganography, active computer vision integration, nonlinear robotic control, robot tracking/path planning, simulating human emotion for robotics, and stereo and color analysis for dynamic navigation.