Proposed Video Watermarking Algorithm based on Edge or Corner Regions

Nidaa F. Hassan
Computer Science Department, University of Technology, Baghdad, Iraq.
nidaaalalousi_5@yahoo.com

Rusul N. Abbas
Computer Science Department, University of Technology, Baghdad, Iraq
rasulaalkhaffaji@yahoo.com

Received on: 08/12/2015
Accepted on: 23/02/2017

Abstract- *In this research, a watermark algorithm is proposed to embed a secret message in a digital video. The proposed algorithm exploits edges and corners regions in images, to be hosts for hiding secret bits. Embedding in these regions is consider optimal since these regions featuring with colors variation, so embedding will not effect on uniform distribution of colors, and on transparency requirement. The process of embedding and extracting watermarked massage is implemented by decomposition digital video to several images (frames), then selecting the edges and corners regions to be host locations, Least Significant Bit (LSB) techniques are used to embed watermarked message in images of digital video. Investigations results proved that number of hidden bits in corner region is small in comparison with edges regions, but it is harder to detect. Text message before embedded in video frames is encrypted by Advanced Encryption Algorithm (AES) to increase security and robustness of watermarking process.*

Keywords- *Video Watermarking, Edge Detection, Corner Detection and Statistical Attacks.*

How to cite this article: N.F. Hassan and R.N. Abbas, "Proposed Video Watermarking Algorithm based on Edge or Corner Regions" *Engineering and Technology Journal, Vol. 36, Part B, No. 1, pp. 25-32, 2018*

1. Introduction

In the recent years, there are many topics became more popular in the security field. one of these topics is watermarking technique which means protecting of data or information, this technique divides in two types :Visible watermarking and Non-visible watermarking, and both types are implemented by : embedding the logo inside video cover, and extracting process which extracted logo information from the watermarked video [1].

There are also watermarking techniques for audio, video, and text data. Digital watermarking techniques provide high security to digital content by allowing only authorized person to modify or detect [2].

Video watermarking is unlike image watermarking, due to the availability of an extra data that allows information to be more redundantly and reliably to embedding [3].

The goal of research preserves the hidden of secret messages, so that the attacker cannot discover or a doubt the existence of a message. There are some attacks (statistical attacks) that can discover watermark and detect it, this can be considered as the problem of this research. Therefore, an improved algorithm is proposed to avoid these statistical attacks.

2. Video Watermarking

Video watermarking is comparatively a new technology that has been projected to solve the problem of illegal manipulation and distribution of digital video. Video watermarking embeds data in the video for the purpose of identification, annotation and copyright. In various ways, a number of videos watermarking techniques have been used, in order to obtain a robust watermark and to maintain original video fidelity [4].

1. 3. Edge and Corner Regions

Boundaries are essential in image processing that characterized by edges, since edges commonly take place on the boundary between two different regions in an image. Edge detection allows user to recognize those image features, where there is a more or less intense change in gray level or texture, so edge illustrates the end of one region in the image and the beginning of another [5].

There many methods used for edge detection, the most important methods are: Sobel, Prewitt, Kirsch and Canny; these methods have been presented to identify changes in images [6].

Corners are characterized by being the points that have high curvature and lie in the junction of different brightness regions of images which give them an important local feature in the images, even with a variety of image features, corners are not influenced by illumination and having the property of rotational invariance. Extracting corner from images represents a lot of sensitive information and the most popular corner detectors is Harris corner detection algorithm [7].

https://doi.org/10.30684/etj.36.1B.4
2412-0758/University of Technology-Iraq, Baghdad, Iraq

Harris corners detector detect the common points (Interest Points) between the two images, these points (Corners) have a large intensity variations between the directions around [8].

4. Related Review

There are number of researchers has focused on watermarking in video using edge and corner regions, the most recent researches are explained as follow:

Ghosh et al. [9] proposed a novel Watermarking technique whether visible and invisible watermarks are embedded in the video by using Discrete Wavelet transform (DWT) which gives an extraction edge in the copyright protection. The suggested algorithm works properly on gray scale and on video of uncompressed (AVI format) that protected a video from illegal access by embedding watermarks into video, Peak Signal to Noise Ratio (PSNR) is calculated to measure efficiency of this method.

Ling & Zhang [10] proposed a novel semi-fragile watermarking video content authentication based on a hybrid feature, consisting of gray threshold and relative total variation edge feature. The watermark is encrypted and embedded invisibly into the principal diagonal of 8×8 DCT coefficients. At the receiver side, when a part of the watermarked video is tampered, the proposed approach can locate the manipulated areas. The experiment demonstrates that the proposed method can recognize malicious attacks from the common valid alterations. The proposed method is based on hybrid feature, which has higher recall, thus the method is fulfilled the requirements of video tampered areas.

Dutta [11] presented a novel region-based information theoretic approach in corner of image using the concept of region splitting, where corner detection algorithm is performed to only those regions containing more information with more intensity variation, the image is divided into a number of regions which are investigated for significant intensity variations and high information content. Corner detection algorithm is applied later to the reduced portion of the image and to improve computational efficiency by reducing the false positive corners in the image. The performance of the algorithm is heavily liable on the threshold that refers how much variations are allowed in the region containing corners and what is the minimum information content in a region.

Batra and Talwar [4] proposed a hybrid-watermarking scheme for digital videos based on Singular Value Decomposition (SVD) and multilevel Discrete Wavelet Transform (DWT), the video frames are distributed into layers (RGB) and DWT sub-band decomposition of host video, for providing copyright protection together with the reliability. This watermarking scheme stands on hybrid model using singular values from watermark image after resizing by using singular values of the wavelet decomposed frame's and also embedding the watermark key with watermarks decomposed orthogonal values in the 4-level decomposition of the selected low energy band of the decomposed band. Experimental results are show that the proposed scheme is able to resist a variety of video processing attacks as well as imperceptibility, mostly in geometrical attacks.

2. 5.The Proposed Watermarking Algorithm

The proposed Algorithm video watermarking consists of two modules, embedding and extracting modules. The following presented these two modules:

I. Embedding Module

In this module, a watermark is embedded into digital video; the host video is uncompressed video file with type AVI video file format, Figure 1 shows the general diagram of the embedding module. The following steps are explained the flow of embedding module:-

1-Apply AES algorithm to encrypt message.

2-Converting encrypted secret message into ASCII codes.

3-Conversion of ASCII codes into binary stream.

4-Open digital video format (AVI)

5-Embedding encrypted watermarked message into frames, the embedding is applied by one of two methods:

a-Embedding is done within selected edges regions using four edge detection methods (Sobel, Prewitt, Kirsch and Canny).

b-Embedding is done within selected corner regions using Harries corner detector.

6-Embedding Message (Add Watermark into Image)

The above steps are described briefly in the following sections:

1-Apply AES Algorithm to Encrypt Message

In this step, the plain text message is encrypted into cipher message by using AES Encryption algorithm. This method is applied to increase the security of embedded watermarked message by using 256-bit key.

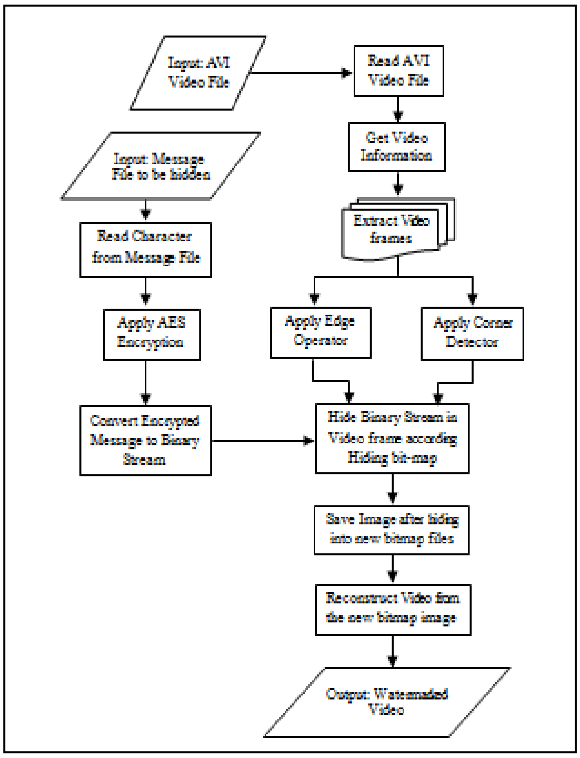

Figure 1: Block Diagram of Embedding Module.

2-Converting Encrypted Secret Message into ASCII Codes

Each encrypted character from pervious step is converted to ASCII code, which known as (American Standard Code for Information Interchange) code to be embedded as watermark message in digital video.

3-Conversion of ASCII codes into binary stream

In this step, the encrypted message is converted to binary stream; this stream of bits is embedded in pixels within video frames by using LSB technique.

4-Open digital video format (AVI)

In this step, the AVI video file is opened, where the full path of the file video is specified. Header information of AVI video is read to extract video frames from the video file, and then video frames are extracted into separate bitmap images files. Algorithm (1) illustrates this process.

5-Embedding encrypted watermarked message into frames

In the proposed algorithm, images are used to be host for watermark by exploiting edges regions, so in this step, edge detectors are applied on images (frames) to detect edges by using four different detection methods (Sobel, Prewitt, Kirsch, and Canny). The following are the common edges operators that used in embedding watermarked:

Algorithm (1): Read AVI Video File

Input: VideoFile: (AVI Video file)
Output: Frames: (Video Frames Saved as BMP images)

Step 1: Read Video Header
VideoHeader ← Read_Header(VideoFile)
Step 2: Create Folder to Save Video Frames
CreateFolder(DestinationFolder)
Step 3: Pass to all Video frames
- { for each frame in the Video}

For $i = 0$ to $VideoHeader.frameCount - 1$
Extract Frame from Video File and Save as BMP image
- {Automatically give name for each frame in the video}

$BmpFileName = Frame + i + ".bmp"$
- {Save frame as BMP Image}

$VideoFile.ExportBMP(i, BmpFileName)$
Continue for next frame
Next i
Step 4: END

a-Sobel, Prewitt , Kirsch and Canny Operators

Sobel operator is the simplest method that uses for edge detection. Embedding bits are done by applying the following steps:
1-Selecting video frames
2-Apply Sobel operator to first frame to detect all edges in it. This process is applied to all successive frames.
3- If the frame does not contain edges or fine edges then this frame is skipped or executed. Specify values of 3x3 filters for X axis and Y axis:

$$Sobel\_Filter\_X = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}$$

$$Sobel\_Filter\_Y = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix} \quad (1)$$

Prewitt operators, is one of edge detection which is used to detect the edges, it from Sobel detector differs by coefficients value. Kirsch operators are similar to the Sobel edge detection algorithm but it differs by number of filters and filter's coefficients value. Canny edge detector is consider as the best method for detecting edge area, where progress multi-stage algorithm to detect a wide range of edges in images from selected edge of the optimal detection. Prewitt, Kirsch and Canny are used as the same as Sobel detector.

b-Proposed Watermarking algorithm based on Corners

Harris Corner Algorithm is designed and implemented since it's an efficient method to detect best corner regions for robust watermarking process and make it difficult to discover message by the statistical attack. Algorithm (2) illustrates Harris Corner Detection Algorithm.

Algorithm (2):Harris Corner Detection Algorithm

Input: Img: (Original bitmap image)
Threshold: (threshold value for selecting corner) k: (Harris parameter k. Default value is 0.04)
Sigma: (Gaussian smoothing parameters)
r: (Non-maximum suppression parameters)

Output: Corner-List: (corners points of an image)

Step 1: { Calculate partial differences of image pixels }

```
for (y = 1; y < height - 1; y++)
    for (x = 1; x < width - 1; x++)
// Retrieve the pixel neighborhood
  a11 = Img[x-1,y-1], a12 = Img[x,y-1],
  a13 = Img[x+1,y-1];
  a21 = Img[x-1,y],   /* a22   */
  a23 = Img[x+1,y];
  a31 = Img[x-1,y+1], a32 = Img[x,y+1],
  a33 = Img[x+1,y+1];
// Convolution with horizontal differentiation kernel mask
  float h = ((a11 + a12 + a13) - (a31 + a32 +
            a33)) * 0.166666667f;
// Convolution with vertical differentiation kernel mask
  float v = ((a11 + a21 + a31) - (a13 + a23 +
            a33)) * 0.166666667f;
// save values directly
 Diffx[x,y]= h*h;  diffy[x,y]=v*v;
 diffxy[x,y]=h*v;
```

Step 2: { Smooth the diff images }

```
// Convolve with Gaussian kernel
       convolve(diffx, Gaussian_kernel,Sigma);
       convolve(diffy, Gaussian_kernel,Sigma);
       convolve(diffxy,
Gaussian_kernel,Sigma);
```

Step 3: { Compute Harris Corner Map }

```
for ( y = 0; y < height; y++)
     for ( x = 0; x < width; x++)
            A = diffx[x,y];
            B = diffy[x,y];
            C = diffxy[x,y];
M = (A * B - C * C) - (k * ((A + B) * (A + B)));
if (M > threshold)  Map[x,y] = M;
```

Step 4: { Suppress non-maximum points }

```
// for each row
        for ( y = r; y < height – r ; y++)
           // for each column
           for (int x = r; x < width - r; x++)
              currentValue = map[x, x];
              // for each windows' row
  for ( i = -r; (currentValue != 0) && (i <= r); i++)
              // for each windows' pixel
                 for (j = -r; j <= r; j++)
    if (map[y + i, x + j] > currentValue)
                   {
                     currentValue = 0;
                     break;
                   }
// check if this point is really interesting
              if (currentValue != 0)
              {
     cornersList.Add(new Point(x, y));
              }
```

Step 5: Return (cornerList)

Step 6: END

6-Embedding Message (Add Watermark into Image)

Converted watermark message is established as mentioned previously, so as to be embedded into images (frames of video). Before embedding, message's length must estimate.

Therefore, length (*C-bits*) will be hided in the first group of hiding pixels in image afterward secret message (*N-bits*) itself will be hided.

C-bits in this paper is specified to 40 bits, because maximum length of message to be hided consist of 5-decimal digits is assumed.

Thereafter, message (its bits) is hided in next pixels (after its length). Whether length of message bits was larger than number of hiding pixels of the image (video frame), next video frames is used to hide remaining bits of message. Algorithm (3) illustrates the embedding watermark of the image.

Embedding watermark in edges or corners will release sender from sending hiding locations (hiding map) to the receiver to extract secret message. Instead, receiver will use same edges or corners detection method to find hiding map to extract watermark, this increase security of watermarking technique.

Algorithm (3): Embed Watermark into Frame

Input: ImageFileName: (An input image name (cover image))
EdgeCornerFilterType: (Filter to detect edges or corners to hide watermark)
Message: (Message to be hided in the cover image)
Output: WaterMarkedImage: (Image that hold watermark)

Step 1: Read Image into 2-D array of colors
Image ← ReadImage(ImageFileName)
Step 2: Extract Binary bits of Message that need to hide it
BinaryMessage ← Extract_Binary(Message)
Step 3: Find length of message and hide it in image
MessageLength ← Length(Message)

Step 4: Find Binary of message length and hide it in the first colors in the image
- {Find Binary of Message Length and extended into 40-bits length}

Bin_MessageLength ← ExtractBinary_40bit(MessageLength)
- {Hide 40-bit message length in the first 40-bytes in the image colors}

For i = 0 to Image.Width − 1
For j = 0 to Image.Height − 1

−if (Bin MessageLength = Empty) Goto Step 5
- Hide bit from Bin-Message Length in Color (i,j), Red
- Hide bit from Bin-Message Length in Color (i,j), Green
- Hide bit from Bin-Message Length in Color (i,j), Blue
- Put Color after Hiding in New Color(i,j) that belong to watermarkedImage

−Remove 3 − bits from Bin_MessageLength
Next j
Next i

Step 5: Find Edges or Corners of the Image according specified filter to find hiding Map
CImage ← ClearLSB(*Image*)

EdgeCornerImage ← ApplyFilter(Image, EdgeCornerFilterType)
Step 6: Hide Binary Message in Image According Edges Map
For i = 0 to Image.Width − 1
For j = 0 to Image.Height − 1
- If (this byte belongs to first 40 bytes from image colors)

Continue to next byte
- // check this is edge pixel or not

if (EdgeCornerImage.Getpixel(i,j) !=White Color)
Continue to next color;
- Hide 3bits from Binary Message in Color(i,j) in Red, Green, Blue
- Put color after Hiding in new Color(i,j) that belong to WatermarkedImage
- if (Binary Message is Empty)Goto Step 7
- Remove 3 − bits from Binary message

Next j
Next i

Step 7: Return (WaterkedImage)
Step 8: END

II. Extraction Module

In this module, the watermarked secret bits are extracted from video frames according to bitmap image files, the general diagram of extraction module as shown in the Figure 2. The following steps are explained the flow of this process module:

1-Open watermarked video file.
2-Extract binary stream from host images by using one of two methods:
a-Extraction is done within selected edges using edge detection methods (Sobel, Prewitt, Kirsch and Canny) are similar used in embedding module.
b-Extraction is done within selected comer regions using Harries corner detectors.
3-Binary stream of extract message bits is converted into ASCII codes.
4-Apply AES algorithm to decrypt ASCII message.

To extract message from watermarked-image (video frame), firstly edge or corner filter is applied to find Hiding Map. Thereafter, extract C-bits of message length from first 14 pixels of hiding map (40-bits/3 =13.3=14). Find secret message length (from these bits) and depends on it as the actual message bits length.

An important note must be mentioned in this step, Hiding Map; that extracted from edges or corners detectors, in extraction stage may different from Hiding map of hiding stage. This is due to values of pixels are changed during hiding process. This was a challenge in this paper because different hiding map causes incorrect message.

To solve this problem, Image's LSB-Clearing, so hiding process (changing LSB) will not effect on extracting hiding map (in extracting process). This clearing LSB process is applied in hiding and extracting processes. Algorithm (4) illustrates the extraction of the all bits of watermark image.

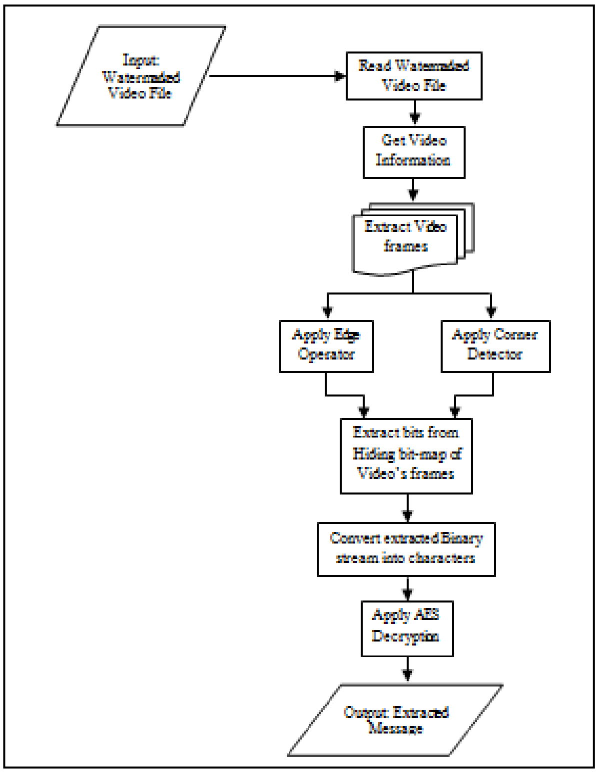

Figure 2: Block Diagram of Extraction Module

Algorithm (4): Extract Watermark from Image

Input: WMImageFileName: (Watermarkedima image name)
EdgeCornerFilterType: (Filter to detect edges or corners to extract watermark)

Output: Message: (message to be extracted from watermarked image)

Step 1: Read Watermarked Image into 2-D array of colors
$Image \leftarrow ReadImage(WMImageFileName)$

Step 2: Extract message length from first colors in the image

- {Extract Binary of Message Length from first 40-bits length}

$Bin\_MessageLength \leftarrow$ "“
$For\ i = 0\ to\ Image.Width - 1$
$For\ j = 0\ to\ Image.Height - 1$
$-if\ (Bin\ MessageLength\ =$
$40)\ Goto\ Step\ 3$
$-Extract\ bit\ from\ \ Color(i,j).Red$

$-Extract\ bit\ from\ B\ Color(i,j).Green$
$-Extract\ bit\ from\ \ Color(i,j).Blue$
$-Add\ 3 - bits\ to\ Bin\_MessageLength$
$Next\ j$
$Next\ i$

Step 3: {Find Message Length}
$MessageLength$
$\leftarrow Convert2Decimal(Bin\_MessageLength)$

Step 4: Find Edges or Corners of the Image according specified filter to Extract Message
CImage $\leftarrow$ ClearLSB$(Image)$
EdgeCornerImage $\leftarrow$
ApplyFilter$(Image, EdgeCornerFilterType)$

Step 5: Extract Binary Message from Watermarked Image According Edges of Filter
$For\ i = 0\ to\ Image.Width - 1$
$For\ j = 0\ to\ Image.Height - 1$

- $if\ (this\ byte\ belongs\ to\ first\ 40 -$
$bytes\ from\ image\ colors)$
$Continue\ to\ next\ byte$
- // check this is edge-corner pixel or not
if (EdgeCornerImage .Getpixel(i,j) ! White Color)
$Continue\ to\ next\ pixel$
- $Extract\ 3 - bits\ from\ Color(i,j)$
$from\ Red, Green\ and\ Blue$
- $Put\ the\ extracted\ 3 - bits\ in$
$Binary\ Message$

$if\ (Binary\ Message/8\ =$
$MessageLength\ \ )Goto\ Step\ 6$
$Next\ j$
$Next\ i$

Step 6: Find Message from Extracted Binary bits of Message
$Message \leftarrow$
$ConstructMessage(BinaryMessage)$

Step 7: Return $(Message)$
Step 8: END

6. Results and Discussion

Evaluation of the proposed watermark in video is accomplished by estimation of distortion measures. Three digital video are evaluated before and after embedding three watermark secret messages.
The Mean Square Error (MSE), Mean Absolute Error (MAE) and Peak to Signal Noise Ratio (PSNR) values are calculated to each frame in video, as shown in Table 1, 2 and 3.

Table 1: Mean Square Error (MSE) for video by using blind, edges and corner regions.

Video Name	Message Size	Classical Watermark	Proposed Watermarked				
		Blind Watermark	Sobel Edge Detection	Prewitt Edge Detection	Kirsch Edge Detection	Canny Edge Detection	Harries Corner Detection
Superimpose	Msg_3KB	0.07728	0.01177	0.01157	0.01361	0.01018	0.00083
Test _ video	Msg_10KB	0.05926	0.03002	0.02991	0.06004	0.03001	0.00064
Video1	Msg_13KB	0.30574	0.04376	0.02354	0.07632	0.03392	0.00107

Table 2: Mean Absolute Error (MAE) for video by using blind, edges and corner regions.

Video Name	Message Size	Classical Watermark	Proposed Watermarked				
		Blind Watermark	Sobel Edge Detection	Prewitt Edge Detection	Kirsch Edge Detection	Canny Edge Detection	Harries Corner Detection
Superimpose	Msg_3K	0.07728	0.01177	0.01157	0.01361	0.01018	0.00083
Test _ video	Msg_10K	0.05926	0.03002	0.02991	0.06004	0.03001	0.00064
Video1	Msg_13K	0.30574	0.04376	0.02354	0.07632	0.03392	0.00107

Table 3: Peak to Signal Noise Ratio (PSNR) for video by using blind, edges and corner regions.

Video Name	Message Size	Classical Watermark	Proposed Watermarked				
		Blind Watermark	Sobel Edge Detection	Prewitt Edge Detection	Kirsch Edge Detection	Canny Edge Detection	Harries Corner Detection
Superimpose	Msg_3K	59.2498	69.6239	69.0761	68.7715	69.7583	78.9968
Test _ video	Msg_10K	60.4031	65.7235	64.3118	60.3461	69.1884	80.0443
Video1	Msg_13K	53.2771	61.7297	64.9831	59.6863	63.1278	78.5285

To evaluate the efficient of robustness of the proposed watermark, watermarked video is subjected to three types of attacks (Laplace Formula, RS Analysis and Sample-Pair Analysis). The results of applying these attacks are shown in Table 4, 5 and 6.

Table 4: The Statistical Attack of the Laplace Formula

Video Name	Message Size	Classical Watermark	Proposed Watermarked				
		Blind Watermark	Sobel Edge Detection	Prewitt Edge Detection	Kirsch Edge Detection	Canny Edge Detection	Harries Corner Detection
Superimpose	Msg_3K	437.926	23.0509	22.2711	27.9696	34.0638	1.92505
Test _ video	Msg_10K	622.756	166.27	150.545	249.505	223.521	4.68438
Video1	Msg_13K	258.215	18.0265	9.41155	52.052	15.1316	1.46601

Table 5: The Statistical Attack of RS Analysis

Video Name	Message Size	RS of original video	Classical Watermark	Proposed Watermarked				
			Blind Watermark	Sobel Edge Detection	Prewitt Edge Detection	Kirsch Edge Detection	Canny Edge Detection	Harries Corner Detection
Superimpose	Msg_3K	0.01637	0.01669	0.016627	0.01663	0.016628	0.01677	0.01676
Test _ video	Msg_10K	0.01289	0.01322	0.01312	0.01309	0.01317	0.01316	0.01285
Video1	Msg_13K	0.01481	0.01549	0.01493	0.01485	0.01519	0.01491	0.01479

Table 6: The Statistical Attack of Sample-Pair Analysis

Video Name	Message Size	Sample-Pair of Original Video	Classical Watermark Blind Watermark	Proposed Watermarked				
				Sobel Edge Detection	Prewitt Edge Detection	Kirsch Edge Detection	Canny Edge Detection	Harries Corner Detection
Superimpose	Msg_3K	0.01236	0.01274	0.012688	0.012684	0.012683	0.01282	0.01290
Test _ video	Msg_10K	0.00632	0.00669	0.00658	0.00655	0.00661	0.00659	0.00664
Video1	Msg_13K	0.00074	0.00147	0.00112	0.00105	0.00126	0.00112	0.00093

The effects of some factors and the performance results of the watermarking in video are discussed below:

1-MSE and MAE results are similar, and this due to their equations which they give the same result if the difference of 0 or 1. LSB embedding lead to the biggest difference between the original image and watermarked image is 1.

2-The values of PSNR (measure the rate of deformation in frames and video) are high, this proves that video with watermark is reserved transparent.

3-The proposed corner-based video watermarking shows that it has lowest distortion. This due to the amount of storing data in each video frame of because number of corners is small compared with edges and whole image.

4-To increase probability of undetectable watermark, corner-based watermark is proposed in this research because corners located in regions of high-color variation and this will make it hard to detect (imperceptible) as shown in the results.

7. Conclusion

The proposed algorithm improve and increase the secrecy of watermark, by embedding in regions where the change in the color is high in order to avoid the attackers , so corners are proposed to be optimal regions to achieve this improvement. Therefore, embedding in the corners is the least discovery than embedding in edges, and embedding in edges is better than blind embedding.

Reference

[1] L. Lakshmipathi and S Srinivasa, "DWT Video Watermarking Based On SVD," International Journal of Embedded & Vlsi System, Vol. 3, pp. 1-5, 2014.

[2] H.B. Abdul Wahab and S.F. Amir, "Efficient Digital Watermark key Generation Using Hexagonal Structure and parametric Lagrange Curve," Eng. & Tech. Journal, Vol. 33, No.2, pp.192-203, 2015.

[3] G. V. Mane and G.G. Chiddarwar, "Review Paper on Video Watermarking Techniques," International Journal of Scientific and Research Publications, Vol. 3, pp. 1-5, 2013.

[4] S. Batra and R. Talwar, "Blind Video Watermarking based on SVD and Multilevel DWT," European Journal of Advances in Engineering and Technology, Vol.2, No. 1, pp. 80-85, 2015.

[5] R.M. Kumar and R. Saxena, "Algorithm And Technique On Various Edge Detection: A Survey," Signal & Image Processing An International Journal (SIPIJ) Vol.4, No.3, pp. 65-75, 2013.

[6] S. Dang and J. Sidhu, "Dual level Image watermarking using SVD and Edge detection technique," International Journal of Advanced Research in Computer Engineering & Technology, Vol. 3, No. 11, pp. 3649- 3652, 2014.

[7] J. Chen, L. Zou, J. Zhang and L. Dou, "The Comparison and Application of Corner Detection Algorithms," Journal of Multimedia, Vol. 4, No. 6, pp. 435-441, 2009.

[8] A.A. Kareem and S.S. Abood, "Real-Time Panoramic Video Construction Using Harris Corners Detector," Eng. & Tech. Journal, Vol.33, No.1, pp. 127-140, 2015.

[9] P. Ghosh, R. Ghosh, S. Sinha, U. Mukhopadhyay, D. Kole and A. Chakroborty, "A Novel Digital Watermarking Technique for Video Copyright Protection," Natarajan Meghanathan, et al. (Eds): SIPM, FCST, ITCA, WSE, ACSIT, CS & IT 06, pp. 601–609, 2012.

[10] C. Ling and W. Zhang, "Semi-fragile Watermarking for Video Content Authentication Based on Hybrid Feature," Journal of Computers, Vol. 25, No. 2, pp. 48-57, 2014.

[11] A. Dutta, "Local Information Based Approach to Corner Detection," International Journal of Application or Innovation in Engineering & Management (IJAIEM), Vol. 4, pp. 186-190, 2015.

Author's biography

Assist. Prof. Dr. Nidaa F. Hassan received the M.Sc. and PhD. in Computer Science from University of Technology, Baghdad, Iraq, 1996 and 2005 respectively. She has around 21 years of teaching experience. Her areas of interests are computer security and image processing.

Rusul Neamah Abbas received the M.Sc. in Computer Science from University of Technology, Baghdad, Iraq, 2016. Her areas interests are systems software, image processing and computer security.