Journal of Physics: Conference Series

PAPER • **OPEN ACCESS**

Video scene change detection based on histogram analysis for hiding message

To cite this article: M Fuad *et al* 2021 *J. Phys.: Conf. Ser.* **1918** 042141

View the article online for updates and enhancements.

Video scene change detection based on histogram analysis for hiding message

M Fuad[*]**, F Ernawan, and L J Hui**

Faculty of Computing, College of Computing and Applied Sciences, Universiti Malaysia Pahang, 26600 Pekan, Pahang Darul Makmur, Malaysia

[*]Corresponding author: fuadbabdullah@gmail.com

Abstract. The rapid growth of internet technology has greatly increased the opportunity for secrecy communication. Most of communications implemented compression method in digital applications due to fast transferring data in the limited bandwidth. In addition, the information can be changed by a third party in communication. This paper proposed a new hiding technique by modifying DCT coefficients in the video frames. The scene changes of the video data are identified based on histogram analysis for hiding locations. Scene changes among video frames are detected by measuring the significant difference of histogram analysis. The proposed hiding technique in video is also evaluated against MPEG-4 compression. The experimental results show that the proposed scheme achieved high imperceptibility with minimum visual distortion on the video quality. The proposed hiding scheme also can recover the concealed data from the compressed video. The results show that the extracted hidden message is able to resistant against MPEG compression.

1. Introduction

Internet has always been the most popular used medium to transfer valuable information between users. The valuable information can be changed or lost by a third party in communication. The hiding data should provide a highly secured way without leaking to any authorized users or hackers, Video steganography is one of concealing techniques in communication by hiding valuable information in the video data [1]. Data hiding techniques can help users to embed secret data inside a cover object without the existence of secret data being realized. Video steganography techniques are developed to be used for transmission of data without being perceptible to others and withstand against attacks from any unauthorized parties [2]. Secret data such as text, audio, images or videos can be embedded for communication. Videos are one of the popular cover media that has been widely used in communication. The video is the combination of audio and a collection of sequence images; it has large capacity for hiding data. Video provides a large redundant area that can be used for hiding data. Therefore, video steganography becomes a popular technique for concealing messages.

Most video steganography techniques pay attention to produce high imperceptibility results. In video steganography, the message must be embedded in the specific frame without affecting the quality of the video [3]. Video steganography allows a large amount of messages to be embedded in the video, and the hidden data can't be noticed by human eyes [4-6]. Steganography can be performed in frequency and spatial domains. In spatial domain, the message was encoded by altering the pixel value directly. The hiding data based on spatial domain will produce high imperceptibility. In this technique, the hidden data in the stego object can be affected by adding the noise. Hiding data in the spatial domain will be destroyed due to compression method. In the frequency domain, the data was not embedded directly into the image pixels. This technique is less vulnerable to steganalysis attacks because it has high security due to its complexity of hiding the messages. Besides that, hiding data in the frequency domain has possibility to produce robust message, it allows the stego object will be compressed before transmitting the data. The most widely used of transform domain in the steganography is Discrete Cosine Transformation (DCT).

In addition, video transmission always been compressed before transferring the data. Most of current video steganography does not satisfy to achieve robustness under compression method [7]. Researchers investigate the scene change in the sequence of video frames for hiding secret-data [8]. The hidden data into scene change has potential to maintain the video quality. The video scene change has not been fully investigated that the hidden data can be secured and robust against compression method.

This paper presents a new hiding technique in scene change of video frames based on the histogram analysis. The difference histogram analysis between current and next video frame is calculated to identify the scene change. The frames that have significant difference value are selected for concealing message. The proposed hiding scheme modifies the selected DCT coefficients for hiding data. The scene change of video frame has a potential for concealing message and it provides high imperceptibility.

2. Preliminaries

2.1. DCT

The Discrete Cosine Transformation (DCT) transformed video frame for non-overlapping blocks. The DCT coefficients consist of high-frequency, mid frequency and low-frequency sub-bands of the DCT coefficients. The DCT is defined by:

$$G_{pq} = \partial_p \beta_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos\frac{\pi(2m+1)_p}{2M} \cos\frac{\pi(2n+1)_q}{2N} \quad (1)$$

for $p = 0, 1, 2, \ldots, M$- 1 and $q = 0, 1, 2, \ldots, N$- 1 where

$$\partial_p = \begin{cases} \frac{1}{\sqrt{M}}, p = 0 \\ \frac{2}{\sqrt{M}}, p > 0 \end{cases} \qquad \beta_q = \begin{cases} \frac{1}{\sqrt{N}}, q = 0 \\ \frac{2}{\sqrt{N}}, q > 0 \end{cases} \quad (2)$$

The inverse of DCT is given by:

$$F_{mn} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \partial_p \beta_q R_{pq} \cos\frac{\pi(2m+1)_p}{2M} \cos\frac{\pi(2n+1)_q}{2N} \quad (3)$$

2.2. Entropy

Entropy is related to intensity variation and has great value for homogeneous regions and small values for selected regions [9] [10]. The entropy can be used to estimate the required bits of pixel to present it. The lowest entropy value has less image information to the human visual systems (HVS) [11] [12]. The entropy has been widely implemented to choose suitable location for hiding data, so it can achieve high imperceptibility [13] - [14]. The entropy represents the most significant texture of the image pixels. The entropy is defined by:

$$E = -\sum_{i=1}^{n} r_i \log(r_i) \quad (4)$$

where i= 1,2, …, n. E represents the entropy value, r_i represent the accurrence probability of an event i with $0 \le r \le 1$ and $\sum_{i=1}^{n} r_i = 1$.

3. Proposed Scheme

The experiments use five videos [15] with avi format to test the proposed scheme. The block diagram of the proposed hiding scheme is visualized in Figure 1. The scene change of video frame was detected by estimating each video frame's different histogram analysis as shown in Algorithm 1.

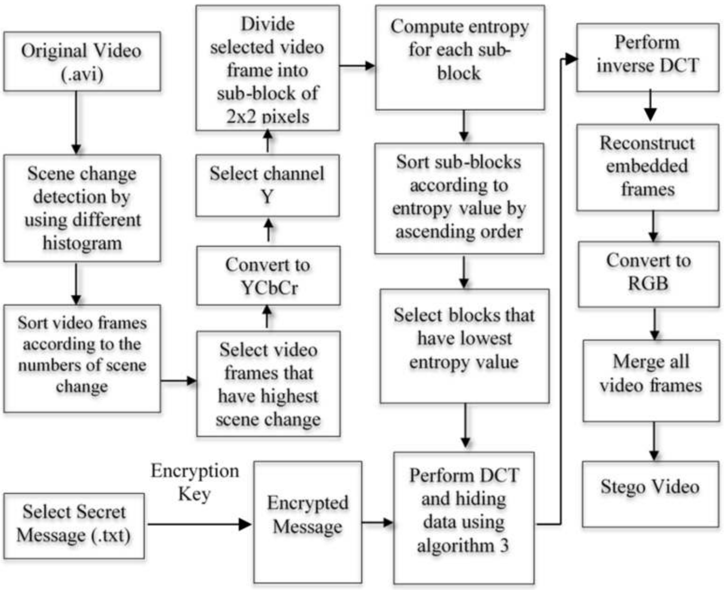
Original Video (.avi)
Scene change detection by using different histogram
Sort video frames according to the numbers of scene change
Select video frames that have highest scene change
Convert to YCbCr
Select channel Y
Divide selected video frame into sub-block of 2x2 pixels
Compute entropy for each sub-block
Sort sub-blocks according to entropy value by ascending order
Select blocks that have lowest entropy value
Perform DCT and hiding data using algorithm 3
Select Secret Message (.txt)
Encryption Key
Encrypted Message
Perform inverse DCT
Reconstruct embedded frames
Convert to RGB
Merge all video frames
Stego Video

Figure 1. Block diagram of the proposed hiding scheme

A video is divided into a sequence of video frames. Then, the histogram analysis is performed to find scene change of video frames. The significant difference between histogram of the original video frame and the histogram of the next video frame. The video frames that have highest number of scene change on the sub-blocks are selected for hiding data. The selected video frames are computed by entropy to determine the hiding data location. The coordinate locations of the sub-blocks with lowest entropy are saved into a database. The sub-block of the video frame that have lowest entropy are computed by DCT to obtain DCT coefficients. The proposed hiding data is performed by following Algorithm 3. Thus, perform inverse DCT for the selected sub-block and merge the sub-block into video frames. Then, merge all video frames to obtain stego-video.

Algorithm 1: Scene change detection

```
Step 1: Read video file.
Step 2: Divide the video into a sequence of video frames, each component of video frame consists of
        red, green and blue channels.
Step 3: Find histogram difference between the current video frame and the next video frame.
        $dR$ = difference ($R_{i,1} - R_{i+1,1}$)
        $dG$ = difference ($R_{i,2} - R_{i+1,2}$)
        $dB$ = difference ($R_{i,3} - R_{i+1,3}$)
        $Dhist = dR + dG + dB$
        where $R_i$ denotes as the current video frame and $R_{i+1}$ represents the next video frame
Step 4: Compute the overall histogram difference from $i$ =1 to $i$ =$T$ - 1, where $T$ is the last video frame.
Step 5: Compute the mean and standard deviation of $Dhist$.
```

$$\text{Mean of Frame Difference } \overline{\mu} = \frac{1}{n}\sum_{i=1}^{n} D_{hist_i} \tag{5}$$

$$\text{Standard deviation } \sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}} \left(Dhist\ i\right) \tag{6}$$

$$\text{Threshold } = \overline{\mu} + \sigma * a\,, where\ a = 2 \tag{7}$$

```
Step 6: Compare threshold with Dhist.
        if Dhist > Threshold
                $R_{i+1}$ is marked as scene change frame
        End
Step 7: Sort the video frame that have scene change frame, highest histogram difference and save it
        into a database.
```

3.1. Hiding Data

The hiding data into the video is discussed in the following algorithm.

Step 1: Read video file.

Step 2: Select the video frame that have scene change frame, highest histogram difference from database

Step 3: Convert the selected frame into YCbCr, then select Y channel

Step 4: Divide the selected video frame into 2×2 pixels, then it is computed by entropy method

Step 5: Sort sub-block image according to the lowest entropy value, then save the coordinates of the selected sub-block into a database

Step 6: Perform DCT on each selected sub-block, embed the message bits into the selected sub-blocks using Algorithm 2.

Algorithm 2: Hiding data

Input: A= $G(1,2)$, B= $G(2,1)$, a

Output: Embedded sub-block

```
If message(i) ==1
   If abs(A) < abs (B)
      C = B
      B = A
      A = C + a
   else
      A = A + a
      B = B
   end
else if message(i) == 0
   If abs(A) < abs(B)
      A = A
      B = B + a
   end
   else
      C = A
      A = B
      B = C + a
end
```

where A represents DCT coefficients of $G(1,2)$, B denotes by DCT coefficients of $G(2,1)$ and a is a scaling factor obtained from Algorithm 3.

Step 7: Perform inverse DCT on the selected sub-block

Step 8: Convert embedded video frame into RGB channel.

Step 9: Merge all video frames to obtain stego-video.

Algorithm 3: Threshold

Input: A= $G(1,2)$, B= $G(2,1)$, T

Output: a

```
If A < 0
```

```
    a = -1 * T
else if A > 0
    a = 1 * T
end
if B < 0
    a = -1 * T
else if B > 0
    a = 1 * T
end
```

where T is a user-defined for scaling threshold, and the amount of T is investigated to achieve high video quality and prevent the embedded information in the DCT block.

3.2. Extracting of Hidden Data

The extracting of hidden message is discussed in Algorithm 4 as follows:

Algorithm 4: Extracting data

```
Input: Stego-video, coordinates of the selected frames and sub-block
Output: message recovery
Step 1: Divide a video into a sequence of video frames
Step 2: Select video frames according by referring to the database
Step 3: Select sub-block images based on the coordinates of selected sub-
        blocks in the database
Step 4: The selected sub-block is transformed by two-dimensional DCT
Step 5: Compare the DCT coefficients of G(1,2) and G(2,1) as follows:
            If abs(G(1,2)) < abs (G(2,1))
            Message_recovery(i) = 0
            else
            Message_recovery(i) = 1
            end
```

If the DCT coefficients of $G(1,2)$ is lower than $G(2,1)$, the message bit is 0, else the message bit is 1.

3.3. Evaluation of stego-video

The stego-video is evaluated by Structural SIMilarity index (SSIM) and True Acceptance Rate (TAR) to measure the imperceptibility and message recovery under compression method. SSIM index is defined by:

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\mu_x\mu_y + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\mu_x^2 + \mu_y^2 + c_2)} \tag{8}$$

$$MSSIM = \frac{1}{S}\sum_{k=1}^{S} SSIM(k) \tag{9}$$

where S represents the sequence number of selected video frames. μ_x is the mean over a window in image x, μ_y is the mean over a window in image y and c_1, c_2 are constants. TAR is used to estimate the correctness of the message recover compared to the original message. TAR is utilized to quantify the correctness of the characters that have been hidden into the video under compression method. TAR can be defined by:

$$TAR = \frac{Corrected\_bits}{N} \tag{10}$$

4. Experimental Results

The experiments were conducted to conceal the message with different amounts of message size. The proposed scheme was tested with the amount of message which are 500 characters, 1000 characters and

2000 characters. The proposed scheme was compared to the existing hiding data which are the Randomized Video Frames Using LSB (RC-LSB) and Scene Change Frame Detection and LSB (SCD-LSB). The experimental results of the proposed hiding scheme is listed in Table 1.

Table 1. The comparison of SSIM value from the stego-video using the schemes by RC-LSB, SCD-LSB and the proposed scheme.

Message	Scheme by RC-LSB	Scheme by SCD-LSB	Proposed Scheme
	SSIM	SSIM	SSIM
500 characters	0.999996	0.999998	0.998278
1000 characters	0.999992	0.999993	0.996104
2000 characters	0.999984	0.999983	0.992393

According to Table 1, the proposed scheme achieved slightly lower of SSIM value than other schemes. The proposed video steganography is designed to robust under video compression. MPEG-4 compression is selected to test the proposed hiding message. MPEG-4 has been widely implemented in multimedia applications. The experiments investigate the embedding strength for hiding message in order to achieve an optimal balancing between video quality and resistance against compression method. The experiments increment the embedding strength *T* one at a time under compression method. The optimal embedding strength for hiding message is shown in Figure 2.

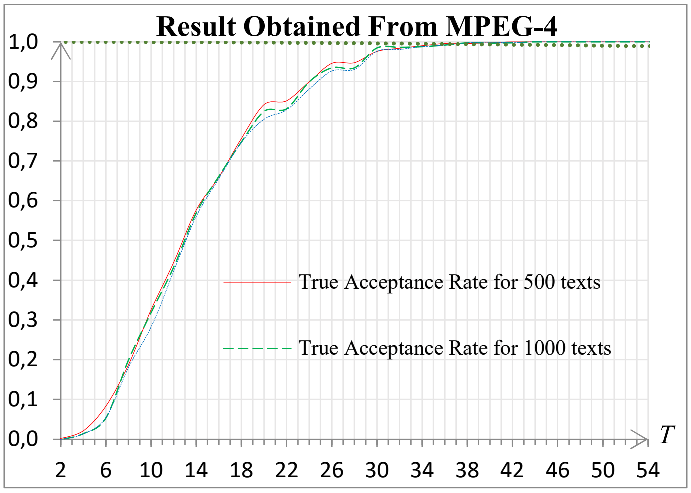

Figure 2. TAR obtained from different amount of hidden message under MPEG-4 compression

Figure 2 shows the TAR with different amount of hidden message by incrementing the embedding strength *T* from 2 to 54. Based on the results, the author defines the optimum embedding strength *T*=44. The results of TAR value under MPEG-4 compression is shown in Table 2.

Table 2. Comparisons of TAR value for different amount of hidden message under MPEG-4 compression

	Scheme by RC-LSB	Scheme by SCD-LSB	Proposed Scheme
Text file (.txt)	TAR	TAR	TAR
500 Texts	0.006	0.000	**1.000**
1000 Texts	0.004	0.003	**1.000**
2000 Texts	0.003	0.000	**1.000**

Referring to Table 2, it shows a significant difference between the proposed scheme and other scheme especially LSB methods. The hiding data using LSB method does not resistant against compression. The hidden message has been destroyed when the stego-video was compressed by MPEG-4. The proposed scheme demonstrates that it can achieve good imperceptibility with SSIM value of about 0.992 for hiding message 2000 character and resistance against compression with TAR value of about 1.

5. Conclusion

This paper proposed a new hiding message into the video scene change based on histogram analysis. The message has been concealed by examining selected DCT coefficients. The video frames that have scene change were selected based on the different histogram analysis. The proposed scheme has been tested under MPEG-4 compression. The scheme was evaluated by hiding message with different amount of hiding data. The experimental results prove that the proposed concealing message achieve high robustness under MPEG compression. The extracted message under MPEG-4 compression achieved high acceptance rate of recovered message. The proposed scheme is able to maintain the video quality with SSIM value of about 0.992.

Acknowledgments

This work was supported by Fundamental Research Grant Scheme (FRGS), Ministry of Education, Malaysia under Grant No. FRGS/1/2018/ICT04/UMP/02/2 and RDU190117

References

[1] Sahar A 2016 *Comput. Electr. Eng.* **70** 380
[2] Wafaa MA, Abdul MS, Rahmab A-S KP 2014 *Comput. Electr. Eng.* **40** 1390
[3] Ng K-H, Liew S-C, Ernawan F 2020 *Int. J. Adv. Comput. Sci. Appl.* **11** 222
[4] Anderson R J and Petitcolas F A P 1998 *IEEE J. Sel. Areas Commun.* **16** 474
[5] Dasgupta K Mondal J K and Dutta P 2013 *Procedia Technol.* **10** 131
[6] Nasreen S M, Jalewal G and Sutradhar S 2015 *Int. J. Comput. Eng. Res.* **10** 2250
[7] Fuad M and Ernawan F 2020 *Bull. Electr. Eng. Inform.* **9** 1015-1023
[8] Fuad M and Ernawan F 2019 *Int. J. Sci. Technol. Res.* **8** 761
[9] Sravanthi M, Devi M Riyazoddin S and Reddy M 2012 *Glob. J. Comput.* **12** 1
[10] Swetha V, Prajith V, Kshema V 2015 *Int. J. Comput. Sci. Eng. Technol.* **5** 206
[11] Ernawan F, Kabir M N, Fadli M and Mustaffa Z 2017 *2nd Int. Conf. Sci. Technol.-Comput.* 6
[12] Ernawan F 2019 *Int. J. Electr. Comput. Eng.* **9** 1850
[13] Ernawan F and Ariatmanto D 2019 *Int. J. Electr. Comput. Eng.* **9** 2185
[14] Ernawan F and Kabir MN 2018 *IEEE 14th Int. Colloq. Signal Process. Appl.* 221
[12] Video test media (derf's collection) 2018. Available: https://media.xiph.org/video/derf/