



Video Steganography for Piracy Prevention in on-demand Internet Streaming Media

Usha B A¹, Dr. N K Srinath², Sangeetha K N³, Abhineet M Deshpande[#]

Assistant Professor, Department of CSE, R V College of Engineering, Bangalore, India¹

Professor and Dean PG Studies, Department of CSE, R V College of Engineering, Bangalore, India²

Assistant Professor, Department of E&C, JSSATE, Bangalore, India³

UG Final Year Students, Dept. of Computer Science & Engineering, R V College of Engineering, Bangalore, India[#]

Abstract: In this paper, a video steganography scheme is proposed to tackle video piracy in on-demand internet streaming media. Several companies offer movies for direct streaming, such as Netflix, Hulu, but video piracy is a huge problem plaguing this industry. While there is going to be technology to prevent direct screen capture of the video, there is need to prevent high quality recordings using an external device. Since the movie transaction record is kept by the streaming company, by embedding a transaction identifier directly into the video, the illegal recorder can be identified. By providing each viewer with a unique copy of a movie, the viewer who illegally records and uploads can always be tracked using the detection system and charged. Thus a steganography system is developed to embed a transaction identifier by varying frame luminance, and detection of that identifier through any recreation or recording of that video. The results of the system indicate that the proposed scheme is robust and reliable for videos of at least two minutes length.

Index Terms: Video Steganography, Luminance, video piracy, digital watermarking

I. INTRODUCTION

Video Steganography and Digital Watermarking deal with data confidentiality and aim to hide information inside a video, such that it is hidden in plain sight, but retrievable only by the intended user [1]. Thus they can be used for digital copyrighting of videos and tackling video piracy, which has become a huge problem in the entertainment industry. It is reported that the movie industry loses \$25 billion a year to it. There are several companies that offer on-demand internet video streaming such as Netflix, Hulu, which allow viewers to stream and view movies and TV shows on their websites. Video piracy is also preventing such companies from offering the simultaneous release of new movies with cinemas. While there is technology to prevent direct screen capture using video overlays, there is no check to prevent video recording using an external device. Since the paid websites like Netflix keep account and banking details of each movie transaction, it becomes easy to track the person buying the movie. Thus by embedding information unique to a transaction (a transaction identifier) into the video using video steganography techniques, and the retrieval of the same from any pirated or recreated copy of that video, it is possible to identify the video pirate.

II. BASIC PHILOSOPHY

The entire steganography system is to be incorporated into a video server pipeline as shown in in Fig 1.

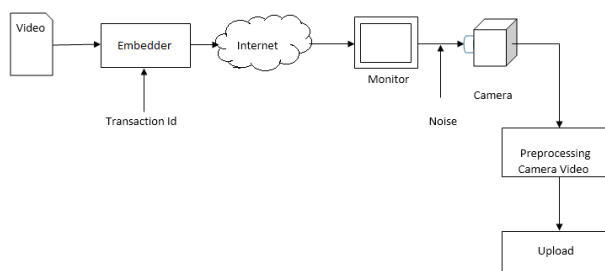


Fig. 1. Video Steganography System

The embedding scheme has to be robust to handle various steganalysis attacks, but also be imperceptible to any viewer [2]. It must satisfy the following conditions:

- The quality of the video should not noticeably degrade upon addition of a mark.
- Marks should be undetectable without secret knowledge.
- Removal of the mark should render the video useless.
- Independent of data format.
- Should be robust to content manipulation.

The information embedded must also persist through recordings. Thus there will be the introduction of noise, skewed angle recordings [3] or even compression and corruption of video. As the luminance of a video can persist through recordings, it is ideal for use in this embedding scheme. Thus information is embedded directly into the video by altering luminance of the frames of the video.

The embedding scheme should also be imperceptible to any viewer, without altering the high quality of the video [4]. Alteration of luminance values of sequences of frames can go unnoticed if the frames belong to the same scene in the video.

The video obtained after recording as shown in Fig 1, and uploaded illegally on the internet still contains the transaction identifier. If any processing is performed by the pirate to remove or corrupt the embedded data, any random changes in the luminance of the video would render it unviewable.

The detection scheme makes use of the original video which is stored on the server for identifying the transaction identifier embedded within the recording of the video. This ensures that there are fewer false positives and complete accuracy of detection is achievable. Some preprocessing of the recorded video is performed in order to compare the luminance values of the recorded video with that of the original, to extract the embedded data.

III. EMBEDDING SCHEME

As mentioned earlier, companies like Netflix maintain information about users, and for each movie purchased a bill is generated which contains a transaction identifier (ID). A 64-bit number to represent a transaction ID is more than sufficient as an unsigned 64-bit integer has a range of 0 to 18 quintillion.

The embedding of a transaction identifier directly into the video cannot be performed at real time, thus a preprocessing method is chosen instead [5]. The original video is split into its respective frames and two additional copies are made of these frames, one for frames with bit-1 embedded within them and another with bit-0. While serving the video, the required transaction identifier can be created by joining the appropriate embedded one and zero frames, to create the required stego video. This method can be optimized for storage and joining speed by storing clips of video, rather than the frames.

While joining the video frames, a portion of the video at known locations has no embedded data to obtain a baseline for the detection scheme.

Embedding of bits in the frames is done by increasing or decreasing the average luminance of a frame based on whether the bit is a 1 or a 0 respectively. The average luminance of a frame is the sum of luminance values of all the pixels of the frame divided by the number of pixels in that frame as per Eq. 1 where luminance is calculated according to BT 709 standards [6]. The extent by which the average luminance of the frame is changed depends on a local and global scaling factor, which alter the values of all the pixels of that frame.

$$\text{Average luminance} = \frac{\sum_{i=0}^{\text{height}} \sum_{j=0}^{\text{width}} \text{luminance}(i,j)}{\text{height} * \text{width}} \quad \text{Eq. 1}$$

The Embedder makes use of the Eq. 2 to identify the amount to increase or decrease the luminance of each pixel to change the average luminance value of the frame. The image should be converted to YCbCr format from RGB first. The luminance values changed, and then converted back to RGB. The scheme is adaptive, and the local scaling factor V by which the luminance of the pixel is changed depends on a spatial factor (λ) and a motion factor (μ) [7]. V is the minimum of the spatial and motion factor, and it is also clipped if it exceed a maximum value V_{Max} . V should be large in moving areas of the video and low in frames that are of the same scene or shot. It makes use of a motion factor for when there is a change in the previous frame to the current frame, especially when there is a change in the scene of the video. If there is not a huge change in some frames (they belong to the same shot), then the spatial factor is utilized.

$$V = \begin{cases} \min(\lambda, \mu) & \text{if } \min(\lambda, \mu) < V_{\text{Max}} \\ V_{\text{Max}} & \text{if } \min(\lambda, \mu) > V_{\text{Max}} \end{cases} \quad \text{Eq. 2}$$

The spatial factor (λ) for each pixel x in frame n is given by the absolute value of the response of the Laplacian filter and multiplication with a global factor s as shown in Eq. 3. The Kernel to be used for the Laplacian filter F is shown in the equation as a matrix.

$$\lambda(x, n) = s * \left| F(x, n) * \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \right| \quad \text{Eq. 3}$$

The motion factor μ is calculated for every pixel x in the frame n using the equation shown in Eq. 4. It requires the same pixel from the previous frame as well, as it is the absolute difference between the luminance of the previous frame's pixel and the current pixel in the current frame. Y is the intensity of the pixel when it is in YCbCr format, and μMin is a the minimum value for μ .

$$\mu(x, n) = |Y(x, n) - Y(x, n - 1)| + \mu\text{Min} \quad \text{Eq. 4}$$

Thus the Embedder makes use of these for calculating the local factor V to increase or decrease the luminance of each pixel by, to alter the average luminance of the frame. By altering the average luminance of multiple frames, a bit-string can be embedded in the video.

In order to achieve robustness, the entire bitstring is repeated as well as each bit. To embed an identifier of 10, 1111000011110000 is actually being embedded as shown in Fig 2. Each bit repetition is called a bucket, and the number of times it repeats is the bucket size, which in the above example is 4. The bitstring is also being repeated twice, and thus the robustness count is 2.

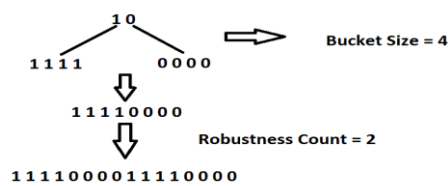


Fig. 2. Illustration of Embedding with Bucket Size and Robustness

The entire embedding process is performed on a GPU to speed up the processing.

IV. DETECTION SCHEME

The detection scheme utilizes the original video, to extract the transaction identifier embedded within the recording of the stego video. The recording could be a direct recording of the stego video, or using an external device, and that may have some processing performed on it to try and corrupt the embedded information.

The first step is to identify which video and what section of the video, the recording corresponds to. This can be done manually or by utilizing some sequence match algorithms for videos [8]. The recorded video frames are split into their frames, and the average luminance values of the frames are calculated using Eq. 1. The luminance values are modified to remove any blue tinges or any other effects added on due to the recording quality of the external device, by using the unembedded frames as a reference. The average luminance values of the original video frames are also obtained. The luminance values of the recording and the original have to be normalised, so that they can be compared. This is done by subtracting the overall mean of all the frames from both that of the recording, and that of the original video, or by taking the first differential. The exact synchronization point between the two lists is determined, and if the luminance of the recording is greater than the original, it represents a 1, and if it is lesser, it represents a 0. By knowing the bucket size, the embedded transaction identifier size and the robustness count, the original transaction identifier is reconstructed.

V. EXPERIMENTAL RESULTS

The embedding and detection scheme have been implemented for standard Full HD videos of resolution 1920x1080 and frame rate of 25 frames per second. The original frame and a recorded stego frame are shown in Fig. 3 and Fig. 4 respectively. The embedding was not easily detectable by viewers, when the bucket size was high (50 or greater). The values of the embedding parameters S , μMin , $V\text{Max}$ were set to be 0.1, 4 and 10 respectively.

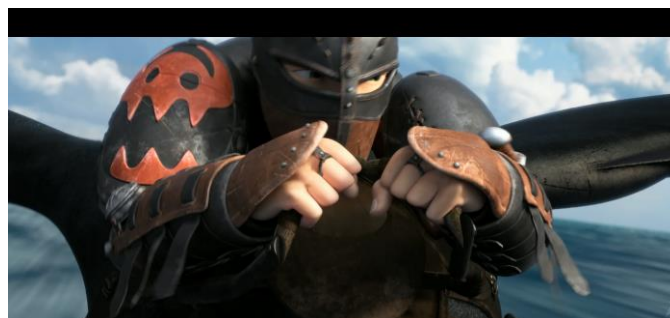


Fig. 3. Frame from the Original Video

Embedding of a 2 minute video (i.e. 3000 frames), embedding of both 1s and 0s to obtain 3 sets of frames (normal, one embedded and zero embedded) could be performed in 20 minutes on a HD5770 GPU using OpenCL by processing 20 frames at a time.

A DSLR camera with recording resolution of 1920x1080p and 25 fps was used for recording the stego video, and recording was performed for skewed angles and different lighting conditions. Stego video was also corrupted, and compressed and tested to see if the data embedded can be retrieved with perfect accuracy.



Fig. 4. Frame from the Recorded Stego Video

Two evaluation metrics were used to analyse the results i) accuracy_bitstring and ii) correspondence_without_robustness.

The metric accuracy_bitstring is measured as a percentage of bits that matched between the bit-string actually embedded, and the bit-string that was extracted. This has to be 100% for the Steganography system to be functioning correctly. Whereas correspondence_without_robustness is a metric that is used to identify the amount of match between the bit-string that is embedded several times across the video across various buckets and repetitions. This value, which is expressed as a percentage, should be as high as possible and is found to be around 80%. If this value drops below 60%, the output is rejected as it does not have the required assurance in the result, and is thus taken as an acceptance threshold.

Higher bucket size results in better results, for both imperceptibility and for robustness, and a bucket size of around 50 was suitable. The ideal bucket size would be adaptive and based on the frames that belong to the same scene in the video.

For any video of at least 2 minutes length, a bitstring of a least 32 bits can be embedded into a video and extracted from any recording of that video with 100% accuracy.

VI. CONCLUSION

The video steganography algorithm described in this paper can be used for identifying the video pirate, in order to try and prevent the video piracy problem plaguing digital cinema and on-demand internet streaming media. The imperceptibility and robustness requirements as well as the resistance to steganalysis, unintentional and intentional corruption were met by using the luminance properties of a video and a robust embedding scheme.

The system however deals only with full HD videos of 25 fps (which most modern encoding schemes use), but can be modified to deal with other frame rates. The embedding process should be performed on a GPU to meet the real time requirements of serving out videos to thousands of viewers, and maintaining pre-processed video clips instead of video frames would be advantageous. This would also be useful for meeting the severe imperceptibility requirements. Automation of synchronization between the original video and the recording of the video would also be necessary, and audio stream of the video can also be used for this synchronization.

This method is designed for a single account scenario where it is assumed that each person has access to a single account and streams media only through that account. The case where a user might merge several different copies and merge them into a single movie is not accounted for. It can however be dealt with by manual separation of videos. Future research will deal with improving the current embedding process, automation of detection, and benchmarking of the results for standard videos.

REFERENCES

- [1] Jonathan Cummins, Patrick Diskin, Samuel Lau, Robert Parlett, "Steganography and Digital Watermarking", [online] 2004, <http://www.cs.bham.ac.uk/~mdr/teaching/modules03/security/students/SS5/Steganography.pdf> (Accessed: 27 May 2014)
- [2] Norishige Morimoto, "Digital Watermarking Technology with Practical Applications", Informing Science Special Issue on Multimedia Informing Technologies, Part 1 vol. 2, (4), 1999, pp. 107 – 111.
- [3] Lintao Lv, Liang Hao, Hui Lv, "Resisting RST Watermarking Algorithm for Image Content Authentication", International Conference on Networks Security Wireless Communications and Trusted Computing, Wuhan, Hubei, 2010, pp. 361 - 364.
- [4] Xing Chang, Weilin Wang, Jianyu Zhao, Li Zhang, "A survey of digital video watermarking", Seventh International Conference on Natural Computation (ICNC) vol. 1, Shanghai, 2011, pp. 61 – 65.

- [5] Martin Steinebach, Enrico Hauer, Patrick Wolf Fraunhofer, "Efficient Watermarking Strategies", Third International Conference on Automated Production of Cross Media Content for Multi-Channel Distribution, AXMEDIS '07, Barcelona, 2007, pp. 65 – 71.
- [6] Charles A. Poynton, "Digital Video and HDTV: Algorithms and Interfaces", Morgan Kaufmann, 2nd Edition, 2003, ISBN: 9780123919267.
- [7] Jaap Haitzma and Ton Kalker, "A Watermarking Scheme for Digital Cinema", International Conference on Image Processing, Proceedings, vol. 2, Thessaloniki, 2001, pp. 487 – 489.
- [8] Sang Hyun Kim, Rae-Hong Park, "An efficient algorithm for video sequence matching using the modified Hausdorff distance and the directed divergence", IEEE Transactions on Circuits and Systems for Video Technology, vol. 12 , (7), 2002, pp. 592 – 596.