Content Authentication of Halftone Video via Flickering as Sparse Signal

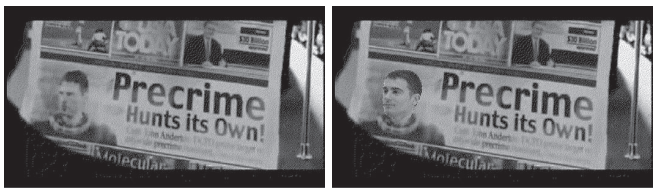
Chao-Yung Hsu,[1] *Chun-Shien Lu,*[1] *Soo-Chang Pei*[2]
[1]Institute of Information Science, Academia Sinica, Taipei, Taiwan, ROC
[2]Graduate Institute of Communication Eng., National Taiwan University, Taipei, Taiwan, ROC

***ABSTRACT*—We investigate the issue of content authentication for halftone videos transmitted over mobile devices. With an eye to the flickering that is the unique characteristic of halftone video and possesses the property of sparsity, a compressed sensing (CS)-based halftone video authentication method is presented. We show that the restricted isometry property (RIP) in CS can explain the principle of hash matching between two CS-based hashes. Promising results obtained from simulations demonstrate the feasibility of our method.**

keywords: Authentication, E-paper, Flickering, Halftone video, Hash, Sparsity

I. Introduction

A novel and interesting example appears in the film, "Minority Report," wherein there is a clip showing John Anderton (acted by Tom Cruise) boarding a train to hide himself in the crowd in order to escape from his partner. Before he enters the train, he is captured and identified by a surveillance camera. Soon, the focus of the picture moves from John's face to the newspaper shown on an electronic device (Fig. 1(a)) owned by a passenger who sits opposite John. It was just a movie in 2002, but it is becoming reality now.



(a) (b)

Fig. 1. (Cropped) halftone video frames: (a) original; (b) tampered with (in head).

One of the emerging mobile devices is electronic-paper (e-paper). Since most e-paper devices developed so far are 1 bit-depth, in order for video sequences to be displayed on e-paper, video halftoning [5], [9] (and the references therein) is required. Nevertheless, in view of the fact that the transmitted video may be tampered with (Fig. 1(b)) and no prior work explored this issue, this motivates us to focus on the security of halftone video transmission and to present a halftone video authentication method in this paper. Unlike watermarking-based image authentication [7], our approach presented here is based on media hashing; namely, a hash is extracted from each halftone video frame, and will be encrypted and transmitted, associated with the halftone video, to the receiver. At the receiver side, the image hash will be extracted again from the transmitted halftone video frame and compared with the received (and decrypted) hash to determine the authenticity of the received halftone video before display. As shown in Fig. 1, our authentication mechanism can on-line prevent the tampered halftone video frames from being displayed on the e-paper.

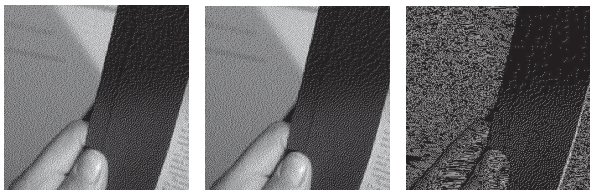
In our halftone video authentication method, we investigate how to use the inherent characteristic of video halftoning, *i.e.,* flickering flaws, to define the so-called halftone video hash. Thus, malicious tampering on the halftone video will change the resultant flickering flaws and content-preserving modifications (such as compression) will preserve flickering flaws to a certain extent.

II. Flickering as Sparse Signal in Halftone Video

We first explain what are the flickering flaws in a halftone video, then briefly introduce the temporal frequency of flickering-distortion (TFoFD) optimized video halftoning method [5], followed by describing how to utilize the flickering flaws as the sparsity cue for authenticating halftone videos.

II-A. Flickering flaws in video halftoning

A general video halftoning method consists of spatial error diffusion and temporal error diffusion, both of which create the flickering phenomena. Flickering is defined as the change of halftone values (either from black to white or from white to black) in the display of consecutive video frames that will be easily perceived by human eyes. For temporal error diffusion, this procedure will cause the pixels located at the same positions of neighboring video frames to have different halftone values due to the introduced diffused temporal errors, in particular, when the pixels have the same or similar gray values. For spatial error diffusion, the diffused spatial errors will affect the halftoning results of the subsequent gray-scale pixels. If the area of gray-scale pixels located at the same positions of neighboring video frames is affected by different diffused spatial errors, then the resultant halftone values may be different, leading to flickering flaws. This situation occurs with higher probabilities for pixels with gray-scale values close to the quantization threshold (*e.g.*, 128) of halftoning. Fig. 2 shows the effect of flickering flaws. Specifically, the white dots in Fig. 2(c) indicate the changes of halftone values, which will make the human eye feel uncomfortable when displaying Figs. 2(a) and (b) successively.



(a) (b) (c)

Fig. 2. Flickering flaws: (a) and (b) show the neighboring halftone video frames, and (c) shows the differences, illuminated with white dots (*i.e.*, flickering flaws), between (a) and (b).

II-B. Proposed Video Halftoning Method

Our video halftoning method with group of pictures (GOPs) being taken into consideration is summarized as follows (see [5] for details), where a reference or I-frame is generated after flickering sensitivity-based reference frame generation is conducted on a GOP. The reference frame, R_F, is then halftoned by spatial error diffusion to generate the halftone reference frame R_H. Each of the subsequent video frames or P-frames is compared with its reference frame during the subsequent halftoning process. If the gray level difference between two pixels located at the same position is smaller than a flickering reduction threshold [5], then the halftone value of the pixel in the non-reference frame is assigned to be the same as that in R_H; otherwise, the halftone value is determined via a quantization process in error diffusion.

II-C. Flickering as Sparse Signal

Unlike the example shown in Fig. 2(c), our video halftoning method can significantly reduce the flickering flaws, as shown in Fig. 3. We can also observe that the flickering flaws are quite limited in the object edges of our halftone video frames so that the flickering flaws form a sparse signal no matter the video contains small motions or not. Furthermore, we can also observe from Table I of [5] that the average temporal frequency of flickering obtained for the video halftoning method (except error diffusion) equipped with flickering reduction is always below 0.04. These results undoubtedly indicate a symptom of sparsity for flickering flaws.

Due to this unique sparsity property, the data stored for a halftone P-frame are the differences between it and its corresponding halftone reference frame in order to reduce storage overhead. For example, storing Fig. 3 is more storage economic than storing Fig. 2(c) since Fig. 3 contains more 0's due to the proposed reference-based video halftoning scheme.

In addition, the flickering will be exploited in compressed sensing as a sparse signal to extract hashes for the purpose of content authentication.

III. Content Authentication for Halftone Video Based on Perceptual Hashing

In our video halftoning method, a video sequence is composed of GOPs, each of which is composed of 1 I-frame and several P-frames. Thus, the proposed authentication mechanism is conducted on the I- and P-frames, respectively, by extracting perceptual hashes from the contents of I- and P-frames.

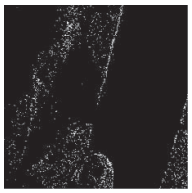


Fig. 3. Difference of halftone values between Figs. 2 (a) and (b) via our method [5].

III-A. Perceptual Hash for I-Frames

Since I-frames preserve more information, well-developed image-based hashing methods can be employed to extract I-frame hash if inverse filtering is applied on halftone I-frames. The I-frame hash extraction strategy is briefly described below.

Step 1: Apply inverse filtering (see, for example, [1]) on I-frames to attain gray-level frames.

Step 2: Conduct Scale Invariant Feature Transform (SIFT) [6] on each resultant gray-level I-frame. SIFT is mainly composed of (1) multiscale Different-of-Gaussian (DoG) filtering; (2) SIFT keypoint detection; (3) keypoint descriptor extraction.

When the keypoints are available, we want to compute a descriptor vector for each keypoint and generate a media hash from the descriptor for reference frame authentication. As done in [6], an orientation assignment is executed for each detected feature point. Then, a normalized 16×16 region expanded from the region covering the derived orientation is built from which feature descriptors are obtained as follows. An SIFT feature descriptor is established for the 16×16 region, which is further divided into sixteen 4×4 blocks, around a feature point. For each 4×4 block, the gradient magnitude and orientation are, respectively, computed for each position (x, y) as:

$$m(x,y) = \sqrt{(Diff_X)^2 + (Diff_Y)^2}, \tag{1}$$

$$\theta(x,y) = tan^{-1}\frac{Diff_X}{Diff_Y}, \tag{2}$$

where $Diff_X$ and $Diff_Y$, respectively, denoting the gradient magnitudes along the X and Y axes in the context of SIFT, are obtained from the difference between two Gaussian convoluted blocks at different scales. Then, the histogram of weighted magnitudes defined on a number of restrictive directions is derived based on Eqs. (1) and (2).

Finally, a hash is defined from the histogram in each 4×4 block. A typical technique (like [8]) is to sort the magnitude histogram composed of 8 entries, wherein the entries with the first 4 largest magnitudes are assigned hash bits 1 and others are 0. Since there are in total $4 \times 4 = 16$ histograms, the hash sequence has 128 bits per keypoint.

As for the similarity criterion, since the objective here is to authenticate halftone videos, partial matching is considered. In this paper, two frames are regarded to be similar if the

number of matched local feature or hash pairs is larger than a threshold. Moreover, it is said that a pair of local features/hashes is matched if the bit error rate (BER) between them is smaller than a threshold. These two parameters can be found statistically [8]. It should be noted that, for other applications relying local features, the proposed local feature-based hashing is useful.

III-B. Perceptual Hash for P-Frames

Different from the reference frames, a P-frame only consists of the flickering information, which are the differences between the halftone P-frame and its corresponding halftone I-frame. As described previously, the P-frame is a highly sparse signal in that only a few of its samples are nonzeros. With an eye to the unique characteristic, compressive sensing is exploited to extract the perceptual hash of a P-frame.

III-B1. Brief of Review of Compressive sensing (CS)

Let x denote a K-sparse signal of length N to be sensed, let ϕ of dimensionality $M \times N$ represent a sampling matrix, and let y be the measurement of length M. At the encoder, a signal x is simultaneously sensed and compressed via random projection and the obtained samples are called measurements y. They are related via random projection as $y = \phi x$. The measurement rate is defined as $0 < \frac{M}{N} < 1$, which indicates the compression ratio (without quantization). At the decoder, the original signal x can be perfectly recovered by means of convex optimization or greedy algorithms if a certain relationship between M and K is satisfied. In CS [2], the constraint of sparsity enables the possibility of sparse signal recovery from using the number of measurements (far) fewer than the original signal length.

III-B2. Hash Generation via CS

In the context of P-frame hash extraction, we let a P-frame be denoted as f_P and its corresponding measurement be denoted as h_P; thus, we have the formulation of random projection as: $h_P = \Phi f_P$. More specifically, given an $M \times N$ sampling matrix Φ, $M < N$, controlled by a secret key, the P-frame f_P (treated as an 1D signal) of size N is randomly projected via random projection to attain a measurement vector with size M, where each vector component will be further quantized to form the final "hash vector," $h_P = [h_P(1), h_P(2), ..., h_P(M)]^T$. Here, a non-uniform quantizer (designed by the k-means algorithm) can be adopted [4].

III-B3. Hash Comparison via CS

To compare two hash vectors, h_P and h'_P, we simply calculate the MSE between them via

$$MSE(h_P, h'_P) = \frac{1}{M} \sum_{m=1}^{M} (h_P(m) - h'_P(m))^2. \quad (3)$$

To estimate the distortion between f_P and f'_P, *i.e.*, $MSE(f_P, f'_P)$, from $MSE(h_P, h'_P)$, which has also been mentioned in [4], [10], we derive the relationship between them for our hash scheme as follows. Without taking the quantization operation on measurements into account, we have:

$$\|h_P - h'_P\|_2^2 = \|\Phi(f_P - f'_P)\|_2^2 = \|\Phi e\|_2^2, \quad (4)$$

where $e = f_P - f'_P$ denotes the frame difference. Based on the assumption that Φ obeys the restricted isometry property (RIP) [2], we have

$$(1 - \delta_K)\|e\|_2^2 \leq \|\Phi e\|_2^2 \leq (1 + \delta_K)\|e\|_2^2, \quad (5)$$

where δ_K is the isometry constant of Φ for all K-sparse vectors e, and δ_K is not too close to 1. Based on Eqs. (4) and (5), we have:

$$\|h_P - h'_P\|_2^2 = \|\Phi e\|_2^2 \approx \|e\|_2^2 = \|(f_P - f'_P)\|_2^2, \quad (6)$$

which implies $MSE(f_P, f'_P) \approx MSE(h_P, h'_P)$. Based on RIP and Eq. (6), we can claim that comparing the MSE between two hash vectors is approximately equivalent to comparing the MSE of the two corresponding frames. To decide whether f'_P is manipulated from f_P based on their hash comparison, we have the rule: if $MSE(h_P, h'_P) \leq \tau_{mse}$, where τ_{mse} is a predefined threshold, f'_P can be authentic; otherwise, f'_P is unauthentic.

IV. Experimental Results

In the experiments, a tested video clip, which was excerpted from the movie "Minority Report," with a size of 480×640 was reported here for authentication of reference frames and P-frames. The malicious tampering conducted here was to change the head by means of copy and paste.

IV-A. Reference Frame Authentication

An example of reference frame authentication is shown in Fig. 4. Fig. 4(a) shows an original reference frame that was transmitted from the sender to the receiver. During transmission, the original frame was maliciously tampered with by pasting in another head to form the modified version, as shown in Fig. 4(c). Since the newly added object, shown in Fig. 4(b), destroys the original local features and their corresponding hashes, the areas (containing a number of features) that contain the pasted object can be located. After our authentication scheme was performed, the areas that were detected as having been maliciously tampered with were those (hash mismatches) indicated in circles in Fig. 4(d).

IV-B. P-Frame Authentication

To evaluate our compressed sensing-based P-frame hashing scheme, the $400 \times (480 \times 640)$ scrambled block Hadamard ensemble (SBHE) matrix[1], where $M = 400$ and $N = 480 \times 640$,

[1]Usually, CS-based image manipulation incurs large sampling matrix problem. One popular way to conquer this problem is to employ block-based CS. Recently, we developed an even better method, 2D CS sensing [3], that can achieve extremely fast computation and approximate recovery.

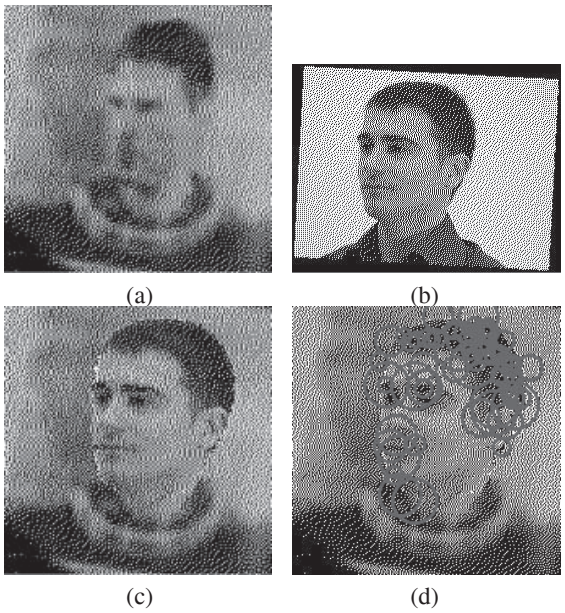


Fig. 4. Reference frame authentication: (a) an original reference frame; (b) a frame used for tampering; (c) a tampered frame with the head changed; (d) detected tampering indicated in circles.

was adopted as the measurement matrix Φ to randomly project a P frame f_P to get the measurement vector. Next, the measurement vector was quantized to form the final hash vector $h_P = [h_P(1), ..., h_P(400)]^T$. Hence, the hash length for a P-frame is $400 \times 8 = 3200$ bits. Thus, the measurement rate in the context of CS or the ratio of hash size to image size is about 0.13%. Fig. 5 shows an example of P-frame authentication. An original reference frame and a P-frame were, respectively, shown in Fig. 5(a) and Fig. 5(b). Fig. 5(c), which was reconstructed from the received measurement via CS, approximates to the difference between Fig. 5(a) and Fig. 5(b), and shows the flickering stored for Fig. 5(b). During the authentication process, the tampered halftone frame Fig. 5(d) instead of Fig. 5(b) is received. The measurement is then extracted and recovered via CS to obtain Fig. 5(e), which denotes the difference between Figs. 5(a) and (d), and shows the flickering flaws extracted from the tampered P-frame. After our scheme was performed, the tampered regions (with white dots) can be detected, as shown in Fig. 5(f), which is the difference between Fig. 5(c) and Fig. 5(e).

V. Conclusions

A content authentication scheme has been presented for halftone video transmitted over e-paper. In particular, sparsity is employed in this paper to characterize the flickering flaws that is unique to halftone video so that compressed sensing-based image hashing is studied.

Acknowledgment: This work was supported by National Science Council, Taiwan, ROC, under grants NSC 100-2631-H-001-013 and NSC 101-2631-H-001-007.

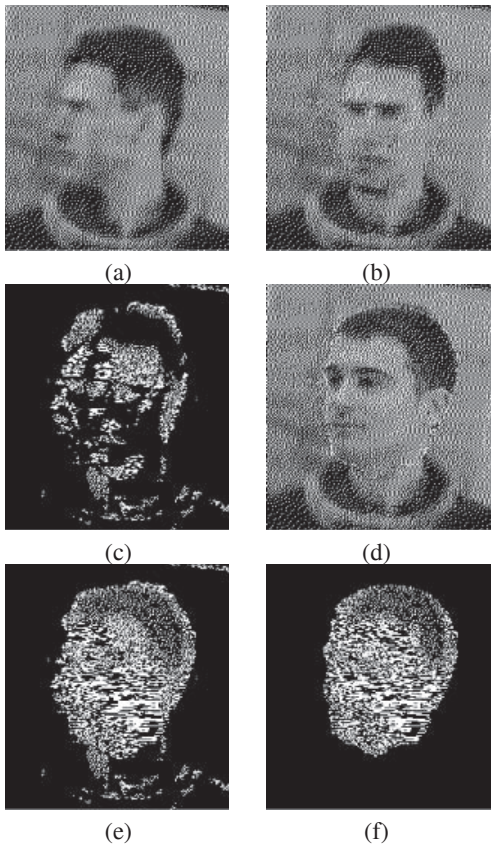


Fig. 5. P-frame authentication: (a) an original halftone reference frame; (b) an original halftone P-frame; (c) reconstructed flickering corresponding to (a) and (b); (d) tampered halftone P-frame; (e) extracted flickering corresponding to (a) and (d); (f) detected tampered regions indicated in while dots.

References

[1] O. C. Au *et al.*, "Hybrid inverse halftoning using adaptive filtering," *ISCAS*, Vol. 4, pp. 259-262, 1999.

[2] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, Vol. 25, No. 2, 2008.

[3] H.-W. Chen, C.-S. Lu, and S.-C. Pei, "Fast Compressive Sensing Recovery with Transform-based Sampling," *Proc. Workshop on Signal Processing with Adaptive Sparse Structured Representations*, 2011.

[4] L.-W. Kang and C.-S. Lu, "Compressive Sensing-based Image Hashing," *Proc. IEEE Int. Conf. on Image Processing*, pp. 1285-1289, 2009.

[5] C.-Y. Hsu, C.-S. Lu, and S.-C. Pei, "Temporal Frequency of Flickering-Distortion Optimized Video Halftoning for Electronic Paper," *IEEE Trans. on Image Processing*, Vol. 20, No. 9, pp. 2502-2514, 2011.

[6] D. Lowe, "Distinctive Image features from Scale Invariant Keypoints," *Int. Journal of Computer Vision*, Vol. 60, pp. 91-110, 2004.

[7] C.-S. Lu and H.-Y. Mark Liao, "Multipurpose Watermarking for Image Authentication and Protection," *IEEE Trans. on Image Processing*, Vol. 10, No. 10, pp. 1579-1592, 2001.

[8] C.-S. Lu and C.-Y. Hsu, "Geometric Distortion-Resilient Image Hashing Scheme and Its Applications on Copy Detection and Authentication," *ACM Multimedia Systems Journal*, special issue on Multimedia and Security, Vol. 11, No. 2, pp. 159-173, 2005.

[9] Z. Sun, "Video Halftoning," *IEEE Trans. on Image Processing*, Vol. 15, No. 3, pp. 678-686, 2006.

[10] M. Tagliasacchi, G. Valenzise, and S. Tubaro, "Hash-based identification of sparse image tampering," *IEEE Trans. on Image Processing*, Vol. 18, pp. 2491-2504, 2009.