# ANTI-FORENSICS FOR FRAME DELETION/ADDITION IN MPEG VIDEO

*Matthew C. Stamm and K. J. Ray Liu*

Dept. of Electrical and Computer Engineering, University of Maryland, College Park

## ABSTRACT

Due to the ease with which digital information can be altered, many digital forensic techniques have recently been developed to authenticate multimedia content. One important digital forensic result is that adding or deleting frames from an MPEG video sequence introduces a temporally distributed fingerprint into the video can be used to identify frame deletion or addition. By contrast, very little research exists into anti-forensic operations designed to make digital forgeries undetectable by forensic techniques. In this paper, we propose an anti-forensic technique capable of removing the temporal fingerprint from MPEG videos that have undergone frame addition or deletion. We demonstrate that our proposed anti-forensic technique can effectively remove this fingerprint through a series of experiments.

***Index Terms***— Anti-Forensics, Digital Forensics, Video Compression

## I. INTRODUCTION

Several factors, such as the integration of digital video cameras into cell phones and laptops, as well as the increasing affordability of high quality digital video cameras, have caused digital video content to become pervasive throughout society. Digital video is commonly used by news organizations for reporting purposes, as well as evidence of specific events by law enforcement, legal institutions, and governmental organizations. Furthermore, many video surveillance systems record footage using digital video rather than film due to the ease with which digital video can be stored. Unfortunately, reliance on digital video for applications in which its authenticity is critical is complicated by the fact that digital video can easily be manipulated using editing software.

To verify the authenticity of digital video files, digital forensic techniques have been developed to detect video manipulation and identify digital video forgeries. Of particular importance is the detection of video frame deletion or addition and recompression. Frame deletion may be performed by a video forger who wishes to remove certain portions of a video sequence such as a person's presence in a surveillance video. Similarly, a forger may wish to falsify an event by inserting several new frames into a video segment. In prior work, Wang and Farid demonstrated that frame deletion or insertion followed by recompression introduces a forensically detectable fingerprint into MPEG video [1].

While existing digital forensic techniques are designed to identify digital forgeries even when the forgery is perceptually undetectable by humans, they do not consider the possibility that a forger may design and use *anti-forensic* operations to remove forensic evidence of their forgery. This is problematic because successfully designed anti-forensic algorithms will allow forgers to create forensically undetectable forgeries. Furthermore, if forensic examiners are unaware of the existence of certain anti-forensic operations, they may place too much trust in forensic results indicating that the digital content in question is authentic. Recent research has already demonstrated that anti-forensic operations are capable of decieving certain existing forensic techniques [2], [3].

To prevent digital forgers from gaining an upper hand, the digital forensics community must develop and study anti-forensic operations. By doing so, forensic investigators can be made aware of weaknesses in existing forensic techniques and know when to trust their results. Additionally, it is likely that anti-forensic operations leave behind evidence of their use just as digital editing operations do. If anti-forensic operations are developed and studied by digital forensic researchers, techniques capable of detecting the use of anti-forensic operations may be preemptively developed.

Recently, we proposed a set of anti-forensic operations capable of removing compression fingerprints from digital images [4], [5], [6] and showed how these operations could be used to fool a variety of existing digital image forensic techniques [3], [6]. In this paper, we further our study of compression-based anti-forensics by proposing a technique capable of hiding evidence of frame deletion or addition in MPEG video. We demonstrate the effectiveness of our anti-forensic technique by testing it against the state-of-the-art frame deletion and addition detection technique developed by Wang and Farid [1].

## II. VIDEO TAMPERING FINGERPRINTS

We begin with a brief discussion of the basics of MPEG video compression along with the forensically detectable fingerprints left in MPEG video when frames are deleted or inserted.

When a video sequence is captured, there is typically a great deal of redundancy between each frame of video. MPEG video compression exploits this redundancy by predicting certain frames in the video sequence from others, then encoding the residual error between the predicted frame and the actual frame. Because the prediction error can be compressed at a higher rate than a frame in its entirety, this leads to a more efficient compression scheme. Performing compression in this manner has its drawbacks, however, because error introduced into one frame will propagate into all frames predicted from it.

To prevent error propagation, the video sequence is divided into segments, where each segment is referred to as a group of pictures (GOP), during MPEG video compression. Frame prediction is performed within each segment, but never across segments, thus preventing decoding errors in one frame from spreading throughout entire video sequence. Within each GOP, frames are divided into three types: intra-frames (I-frames), predicted-frames (P-frames), and bidirectional-frames (B-frames).

Each GOP begins with an I-frame, followed by a number of P-frames and B-frames. No prediction is performed when encoding I-frames, therefore each I-frame is encoded and decoded independently. During encoding, each I-frame is compressed through a lossy process similar to JPEG compression.

P-frames are predictively encoded through a process known as motion estimation. A predicted version of the current P-frame is obtained by first segmenting the frame into $16 \times 16$ pixel blocks known as macroblocks, then searching the previous P or I-frame, known as the anchor frame, for the macroblock that best matches each macroblock in the current P-frame. The locations of these macroblocks in the anchor frame are stored, along with how far each macroblock must be displaced to create the predicted frame. These displacements are referred to as motion vectors. The residual error between the predicted frame and and the current frame, known

This work is supported in part by AFOSR grant FA95500910179.

The authors can be reached by email at {mcstamm,kjrliu}@umd.edu.

978-1-4577-0539-7/11/$26.00 ©2011 IEEE

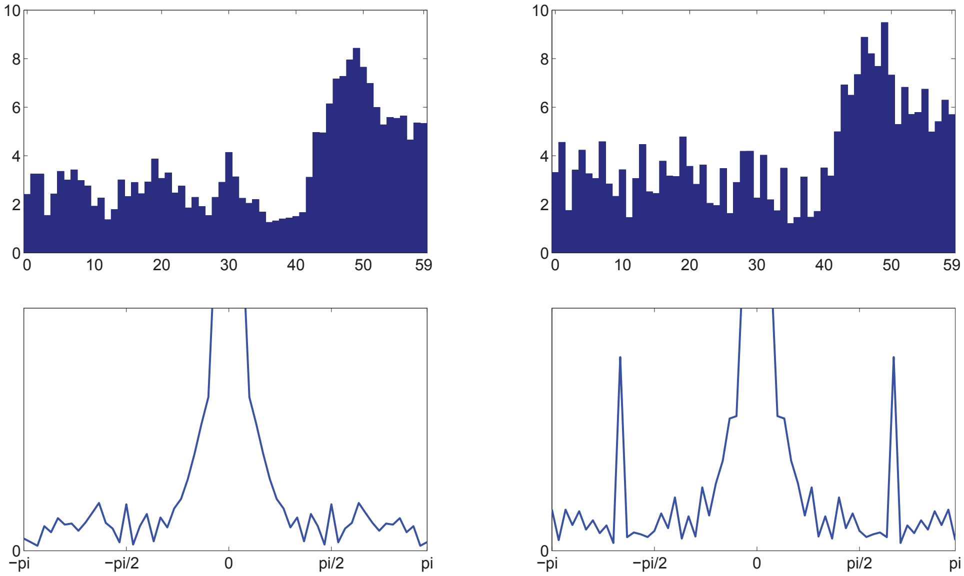

**Fig. 1**. P-frame prediction error sequence (top left) and the magnitude of its DFT (bottom left) obtained from an unedited, MPEG compressed version of the 'Carphone' video sequence along with the P-frame prediction error sequence (top right) and the magnitude of its DFT (bottom right) obtained from the same video after frame deletion followed by recompression.

as the prediction error, is then compressed using the same JPEG-like process that I-frames undergo.

Decompression of these frames is performed by first recreating each predicted frame using the decompressed anchor frame and the stored motion vectors. Next the prediction error is decompressed, then added to the predicted frame, thus reconstructing the frame. B-frames are compressed using a similar process, however each macroblock in a B-frame can be predicted from either the previous anchor frame, the next anchor frame, or both.

When frames are deleted from or added to an MPEG compressed video sequence, the tampered video that results must be recompressed for storage. Previous work has shown that recompression of MPEG video results in two distinct forensically detectable fingerprints; one spatial and the other temporal [1]. The spatial fingerprint can be observed within a single MPEG frame and is similar in nature to fingerprint left by double JPEG compression [7], [8]. The temporal fingerprint occurs in the sequence of P-frame prediction errors and occurs only if frames have been added to or deleted from the video sequence prior to recompression.

During the first application of compression, the lossy and predictive nature of MPEG compression causes the frames within each GOP to become correlated. When frames are added to or deleted from the video prior to the second application of MPEG compression, all subsequent frames occur in a different location in the video sequence and may be grouped into a new GOP. As a result, each GOP in the recompressed video sequence occurring after the frame addition or deletion contains contains frames that originally belonged to multiple different GOPs during the initial compression. During recompression, P-frames predicted from an anchor frame within the same initial GOP will result in less prediction error than those predicted from anchor frames belonging to a different GOP during the first application of compression.

If a fixed GOP structure is used, this increase in prediction error occurs periodically in the sequence of P-frame prediction errors. Wang and Farid have demonstrated that frame deletion or addition can be detected by visually inspecting the sequence

$$e(n) = \frac{1}{N_{xy}} \sum_{x} \sum_{y} |p_{x,y}(n)|, \tag{1}$$

where $N_{xy}$ is the number of pixels in each frame and $p_{x,y}(n)$ is the prediction error of the $n^{\text{th}}$ P-frame at pixel location $(x, y)$, for this periodic fingerprint. Alternately, the discrete Fourier transform (DFT) of this sequence $E(k) = \text{DFT}\{e(n)\}$ can be inspected for peaks resulting from the periodic fingerprint. This fingerprint can be seen in Fig. 1 which shows the P-frame prediction error sequence of 250 frames of an MPEG compressed version of the commonly used 'Carphone' video, along with the P-frame prediction error sequence of the same video after the first 6 frames have been deleted followed by recompression. The GOP structure used during compression was *IBBPBBPBBPBB*. We note that because the temporal fingerprint occurs in the prediction errors of P-frames, the prediction error sequence contains only only approximately 60 entries for this video segment.

## III. TEMPORAL FINGERPRINT REMOVAL

Because spatial fingerprints resemble double JPEG compression fingerprints, which we have shown how to prevent through anti-forensics in previous work [3], we focus exclusively on the removal of the temporal fingerprint. We assume, as Wang and Farid have, that a fixed GOP structure is used during both applications of compression. In order to develop our anti-forensic temporal fingerprint removal technique, we first identify a few simple properties of the temporal fingerprint not stated by Wang and Farid. We use these properties to construct a model of the temporal fingerprint, which we then use to generate a target P-frame prediction error sequence that does not contain the temporal fingerprint. Our anti-forensic operation is designed to be integrated into the MPEG encoding process so that the P-frame prediction error sequence of the anti-forensically recompressed video matches the target prediction error sequence.

### III-A. Temporal Fingerprint Properties

As was previously discussed, the temporal fingerprint corresponds to a repetitive pattern of increased P-frame prediction error as observed in the sequence $e(n)$. We define the period $T$ of the temporal fingerprint as the number of P-frames in one complete repetition of this pattern. The temporal fingerprint exhibits the following propeties:

**Property 1:** The temporal fingerprint's repetitive pattern corresponds to a disproportionate increase in the value of $e(n)$ exactly once within one period of the fingerprint.

**Property 2:** The period of the temporal fingerprint is equal to the number of P-frames within a GOP.

Because the initial application of compression increases the correlation amongst frames within each set, the P-frame prediction error will be lower when a P-frame is predicted from an anchor frame within the same set. Since P-frame prediction occurs across sets only once within each GOP, this increase in prediction error due to frame addition or deletion also occurs only once within a GOP. As a result, the period of the temporal fingerprint is equal to number of P-frames within a GOP and the increase in $e(n)$ due to frame deletion occurs only once per fingerprint period.

Additionally, we define the phase $\phi$ of the temporal fingerprint as the number of P-frames within a GOP before the increase in $e(n)$ due to frame deletion or addition is observed. The phase of the temporal fingerprint can be easily shown to obey the following property:

**Property 3:** The phase of the temporal fingerprint is determined by the equation $\phi = \lfloor |\mathcal{A}|/r \rfloor$, where $r$ is the number of P-frames within a GOP, $\mathcal{A}$ is the set of frames at the beginning of each GOP that belonged to the same GOP during the initial application of compression, $|\mathcal{A}|$ denotes the cardinality of $\mathcal{A}$, and $\lfloor \cdot \rfloor$ denotes the floor operation.

### III-B. Anti-Forensic Temporal Fingerprint Removal

Our anti-forensic operation works by modifying the MPEG encoding process so that the P-frame prediction error sequence matches a target prediction error sequence that does not contain the temporal fingerprint. The value of $e(n)$ is increased to the target value $\hat{e}(n)$ for a given P-frame by changing the frame's predicted value in a manner that increases the prediction error. Since the anti-forensically modified video must be capable of being decompressed by a standard MPEG decoder, we accomplish this by selectively setting a specific number of the motion vectors used to construct the predicted frame to zero, then obtaining new prediction error values for the macroblocks whose motion vectors have been set to zero. We note that though the prediction error is increased for an anti-forensically modified P-frame, the decompressed P-frame remains essentially unchanged by anti-forensic modification because the new prediction error is stored during compression, then added back to the new predicted frame during decompression.

In order to obtain a target prediction error sequence that does not contain the temporal fingerprint, we first use properties 1 through 3 to model the effect of the temporal fingerprint on the P-frame prediction error sequence. Let $e_1(n)$ denote the P-frame prediction error sequence of an untampered video that has been compressed once and let $e_2(n)$ denote the prediction error sequence of the same video after frame addition or deletion has occurred, followed by recompression. We relate $e_1(n)$ and $e_2(n)$ using the following equation

$$e_2(n) = (\alpha + \beta\, \mathbb{1}((n-\phi) \bmod T = 0))e_1(n) \tag{2}$$

where the scalars $\alpha, \beta > 0$ adjust the relative strength of the prediction error, $\mathrm{mod}$ denotes the modulo operation and $\mathbb{1}(\cdot)$ denotes the indicator function.

Because our anti-forensic operation works by selectively increasing the prediction error for specific P-frames, the target prediction error sequence must obey the rule $\hat{e}(n) \geq e(n)$. Taking this into account, we obtain a target prediction error sequence by letting $\hat{e}(n) = e_2(n)$ if $(n-\phi) \bmod T = 0$, then determining the remaining values of $\hat{e}(n)$ through cubic spline interpolation. By choosing the target prediction error sequence in this manner, we prevent the prediction error from being scaled alternatingly by two different values, and raise the prediction error scaling constant to $(\alpha + \beta)$ for each P-frame.

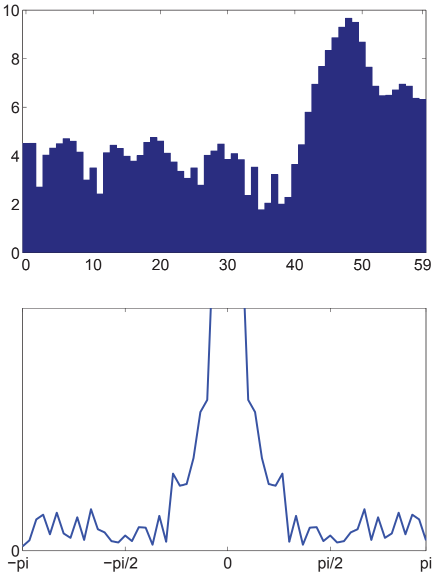

**Fig. 2**. P-frame prediction error sequence (top) and the magnitude of its DFT (bottom) obtained from the 'Carphone' video sequence after the first six frames were deleted and the video was recompressed while using our anti-forensic technique to remove the temporal fingerprint.

Once $\hat{e}(n)$ has been obtained for a parctiuclar P-frame, we determine which of its motion vectors should be set to zero in order to increase the prediction error to the desired level. By setting a macroblock's motion vector to zero, the predicted value of that macroblock is simply the macroblock at the same location in the anchor frame. To decide which motion vectors will be set to zero, we first determine the effect of changing each macroblock's motion vector to zero. We then zero out the motion vectors of the macroblocks whose prediction error is changed the least until the target prediction error level is reached.

Let $m_{i,j}(n)$ denote of sum of the absolute value of the prediction error in the macroblock at location $(i, j)$ in the $n^{\text{th}}$ P-frame when motion prediction is used and let $\hat{m}_{i,j}(n)$ be the sum of the absolute value of the prediction error in the same location when the macroblock's motion vector has been set to zero. We define the difference in macroblock prediction errors as

$$q_{i,j}(n) = \hat{m}_{i,j}(n) - m_{i,j}(n). \tag{3}$$

We note that $q_{i,j}(n) \geq 0$ because the zero motion vector is included in the search space for the optimal motion vector during compression.

Next, we define $\mathcal{Q}^{(l)}(n)$ as the set of indices of the macroblocks that result in the $l$ smallest prediction error differences when their motion vectors are set to zero. More explicitly, $\mathcal{Q}^{(l)}(n)$ is defined as

$$\mathcal{Q}^{(l)}(n) = \left\{(i,j) | q_{i,j}(n) \leq q^{(l)}(n)\right\}, \tag{4}$$

where $q^{(l)}(n)$ is the $l^{\text{th}}$ smallest entry of $q(n)$.

The total absolute prediction error $g_n(l)$ in the $n^{\text{th}}$ frame that results from setting the motion vectors of each macroblock whose indices are in $\mathcal{Q}^{(l)}(n)$ to zero is given by the equation

$$g_n(l) = \sum_{(i,j) \in \mathcal{Q}^{(l)}(n)} \hat{m}_{i,j}(n) + \sum_{(i,j) \notin \mathcal{Q}^{(l)}(n)} m_{i,j}(n). \tag{5}$$

The value of $l$ that minimizes the absolute distance between the target prediction error level and the actual prediction error level is

$$l^* = \arg\min_l |g_n(l) - \hat{e}(n)| \tag{6}$$

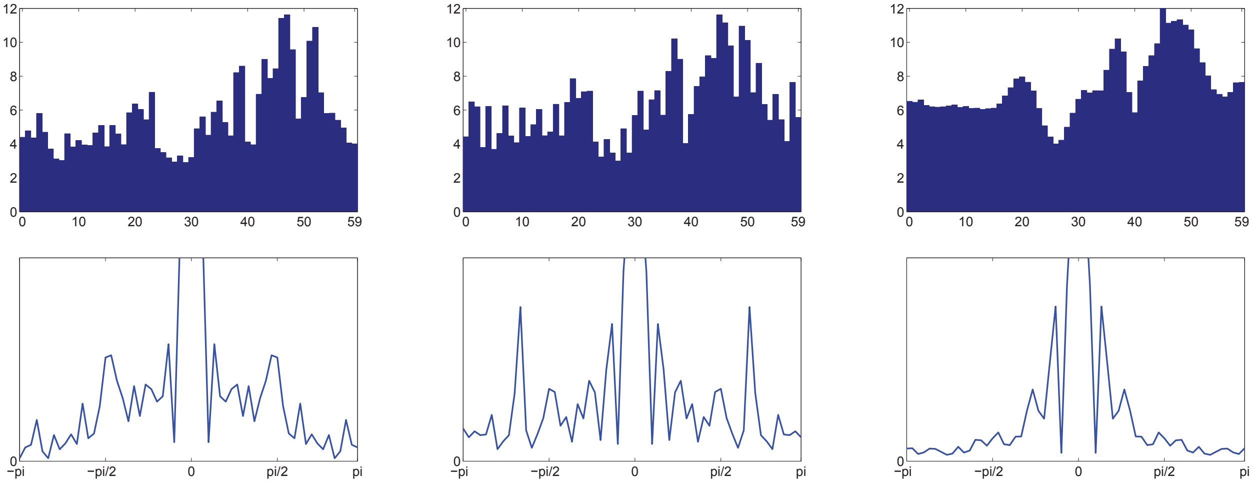

**Fig. 3**. P-frame prediction error sequences (top row) and the magnitudes of their respective DFTs (bottom row) obtained from an untampered MPEG compressed version of the 'Foreman' video (left column), as well as from the the same video after the first six frames were deleted followed by recompression without anti-forensic modification (middle column) and with the use of our proposed anti-forensic technique (right column).

To remove the temporal fingerprint from the $n^{\text{th}}$ P-frame of the recompressed video, we set the motion vectors of each macroblock whose indices are in $\mathcal{Q}^{(l^*)}(n)$ to zero and recompute the prediction error at these macroblock locations during recompression. Due to the relatively small number of macroblocks in each frame, we find $l^*$ for each frame through an exhaustive search.

To reiterate, our anti-forensic temporal fingerprint removal technique can be simply summarized as follows:

1) After frame deletion or addition, construct a target P-frame prediction error sequence that does not contain the temporal fingerprint.
2) Increase the prediction error for each P-frame to the target value by setting the motion vectors of several macroblocks to zero, then recalculating the prediction error for each of these macroblocks.

## IV. SIMULATION AND RESULTS

To evaluate the performance of our proposed anti-forensic technique, we simulated the MPEG compression and decompression process in Matlab and used this to obtain the P-frame prediction error sequence from a number of test videos. In each experiment, the twelve frame GOP structure *IBBPBBPBBPBB* was used. We then deleted a number of frames from the beginning of each MPEG compressed video and applied our anti-forensic technique during recompression to remove temporal fingerprints from the altered videos.

Fig. 2 displays the P-frame prediction error sequence obtained after deleting the first six frames from an MPEG compressed version of the 'Carphone' video sequence, then anti-forensically removing the temporal fingerprint during recompression using our proposed technique. We note that the Fig. 1 shows the P-frame prediction error sequences from the video both before frame deletion and after frame deletion without the use of our anti-forensic operation. As can be seen in Fig. 2, the anti-forensically modified video does not contain a temporal frame deletion fingerprint.

Fig. 3 shows the P-frame prediction error sequence taken from an untampered MPEG compressed version of the 'Foreman' video, as well as the P-frame prediction error sequences obtained after deleting the first six frames then recompressing the video with and without applying our anti-forensic temporal fingerprint removal technique. The temporal fingerprint features prominently in the prediction error sequence of the video in which frames are deleted without the use of our anti-forensic technique, particularly in the frequency domain. By contrast, these fingerprints are absent from the prediction error sequence when our anti-forensic technique is used to hide evidence of frame deletion.

## V. CONCLUSIONS

In this paper, we have proposed an anti-forensic operation capable of removing the temporal fingerprint that arises in MPEG video sequences when frames are added or deleted followed by recompression. We have identified properties of the temporal fingerprint and used these to model the effect of frame deletion or addition on the P-frame prediction error sequence. Our proposed anti-forensic technique operates by selectively increasing the prediction error in certain P-frames of the video so that the P-frame prediction error sequence approximates a target prediction error sequence obtained using our model. The prediction error in each P-frame is increased by setting the motion vectors of certain macroblocks within that frame to zero, then recalculating the prediction error for the frame. Experimental results demonstrate that our proposed anti-forensic technique is capable of removing the temporal fingerprint from MPEG videos that have undergone frame deletion or addition.

## VI. REFERENCES

[1] W. Wang and H. Farid, "Exposing digital forgeries in video by detecting double MPEG compression," in *ACM Multimedia and Security Workshop*, Geneva, Switzerland, 2006.

[2] M. Kirchner and R. Bohme, "Hiding traces of resampling in digital images," *IEEE Trans. Inf. Forensics and Security*, vol. 3, no. 4, pp. 582–592, Dec. 2008.

[3] M. C. Stamm, S. K. Tjoa, W. S. Lin, and K. J. R. Liu, "Undetectable image tampering through JPEG compression anti-forensics," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010.

[4] M. C. Stamm, S. K. Tjoa, W. S. Lin, and K. J. R. Liu, "Anti-forensics of JPEG compression," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Mar. 2010, pp. 1694 – 1697.

[5] M. C. Stamm and K. J. R. Liu, "Wavelet-based image compression anti-forensics," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010.

[6] M. C. Stamm and K. J. R. Liu, "Anti-forensics of digital image compression," *to appear in IEEE Trans. Inf. Forensics and Security*, vol. 6, no. 2, Jun. 2011.

[7] T. Pevny and J. Fridrich, "Detection of double-compression in JPEG images for applications in steganography," *IEEE Trans. Inf. Forensics and Security*, vol. 3, no. 2, pp. 247–258, Jun. 2008.

[8] A. C. Popescu and H. Farid, "Statistical tools for digital forensics," in *6th Int. Workshop on Information Hiding*, Toronto, Canada, 2004.