Identification of Motion-Compensated Frame Rate Up-Conversion Based on Residual Signal

Xiangling Ding, Gaobo Yang, Ran Li, Lebing Zhang, Yue Li and Xingming Sun, *Senior Member, IEEE*

***Abstract*—Motion-compensated frame rate up-conversion (MC -FRUC) is originally presented to increase the motion continuity of low frame rate videos by periodically inserting new frames, which improves the viewing experience. However, MC-FRUC can also be exploited to fake high frame rate videos or splice two videos with different frame rates for malicious purposes. A blind forensics approach is proposed for the identification of various MC-FRUC techniques. A theoretical model is firstly built for residual signal, which is exploited as tampering trace for blind forensics. The identification of various MC-FRUC techniques is then converted into a problem of discriminating the differences of residual signals among them. A pre-classifier is designed to suppress the side effects of original frames and static interpolated frames in candidate videos. Then, spatial and temporal Markov statistics features (ST-MSF) are extracted from the residual signals inside interpolated frames for MC-FRUC identification. Five open MC-FRUC softwares and six representative MC-FRUC techniques have been tested, and experimental results show that the proposed approach can effectively locate interpolated frames and further identify the adopted MC-FRUC technique for both uncompressed videos and compressed videos with high perceptual qualities.**

***Index Terms*—Blind video forensics, Motion-compensated frame rate up-conversion, Residual signal and Classification.**

I. INTRODUCTION

WITH the availability of inexpensive and portable video capture devices such as mobile phones, digital video is enriching our daily life. Meanwhile, the proliferation of powerful video editing tools makes it much easier than ever to produce faked videos without leaving any perceptible traces [1]. The doctored videos are very difficult, if not impossible, to be identified by naked eyes. This breaks our traditional concept of "seeing is believing" and brings serious crises in respect to public confidence. Video forensics, which attempts to verify the trustworthiness of digital videos, has attracted wide research interests in the field of information security. Especially, passive forensics approaches [1]-[4], which detect tampering traces without the aid of any *prior* axillary information such as digital watermark or signature, are extensively studied in recent years.

This work is supported in part by the National Natural Science Foundation of China (61572183, 61379143, 61232016, 61501393, U1405254), and the Specialized Research Fund for the Doctoral Program of Higher Education (SRFDP) under grant 20120161110014.

X. Ding, G. Yang, L. Zhang, and Y. Li are with the School of Information Science and Engineering, Hunan University, Changsha, 410082, China. (e-mail: xianglingding@163.com; yanggaobo@hnu.edu.cn).

R. Li is with the School of Computer and Information Technology, Xinyang Normal University, Xinyang, 464000, China. (e-mail: liran358@163.com).

X. Sun is with the School of Computer and Software, Nanjing University of Information Science and Technology, Nangjing, 210044, China. (e-mail: sunnudt@163.com).

Digital video can be regarded as a series of image frames. Almost all potential image editing operations can be applied to video frame. Thus, most image forensic methods are extended to video forensics in a direct or indirect way [3]. However, digital video has an extra temporal dimension. Video forgeries include both intra-frame and inter-frame operations, which are referred to as spatial-domain and temporal-domain forgeries, respectively. Frame-based manipulations such as frame adding/deleting, group of pictures (GOPs) reorganization are temporal forgeries specific to digital video. Until now, there are some specific methods to detect inter-frame forgeries including frame adding/deleting and frame duplication [5]-[9]. The clues exploited to expose these inter-frame forgeries include optical flow consistency [7], velocity field consistency [8] and Zernike opponent chromaticity moments [9].

Frame rate up-conversion (FRUC) is another special frame-based video manipulation, which interpolates new frames between two successive frames to increases the motion continuity of low frame rate videos [10]. Though FRUC is originally proposed to improve the visual quality of low frame rate videos [11], it can also be used for malicious video forgeries. First, since high bitrate videos are usually more preferred over the Internet, FRUC might be used to fake high bitrate videos from low bitrate ones by increasing their frame rates. Second, two videos with different frame rates can be spliced by up-converting the low frame rate video to match the higher one. Frame repetition and frame averaging are two simple FRUC approaches which disregard objects'motions. However, they often lead to motion jerkiness and ghosting artifacts for non-stationary videos, respectively [17]. In recent years, many advanced FRUC approaches, which are also known as motion-compensated FRUC (MC-FRUC), have been proposed by estimating motions as close as possible to true motions [12]-[20]. Specifically, various assumptions and strategies are exploited by MC-FRUC to achieve better motion estimation (ME) and motion compensated interpolation (MCI). Thus, more natural and realistic videos are obtained for non-stationary videos without leaving any visible artifacts.

Until now, there are still few works reported for the blind forensics of FRUC. Four existing works are summarized as follows. Bestagini *et al.* [21] firstly designed a detector capable of revealing the use of MC-FRUC method. The detector firstly computes an estimated motion vector (MV) for each frame from its neighboring frames, and further computes prediction errors between estimated frame and original one, which leads to a periodic signal. Then, the periodic signal is exploited to infer original frame rate. However, it is claimed that it can not locate interpolated frames. Bian *et al.* [22] presented

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2676162, IEEE Transactions on Circuits and Systems for Video Technology

a similarity-analysis-based detection approach. However, it only reports the detection results of simple frame repetition. Since MC-FRUC usually leads to edge discontinuity or over-smoothing artifacts, we presented a MC-FRUC detection approach by exploiting the periodicity of edge-intensity [23]. After computing the edge intensities of each frame, an adaptive threshold is defined by Kaufman adaptive moving average to differentiate interpolated frames from original ones. Recently, we proposed another blind MC-FRUC detection approach by exploiting average texture variation [24]. It firstly computes a curve of frame-level texture variation to indicate the existence of MC-FRUC and infer original frame rate. Then, an adaptive threshold is selected in a way similar to [23] to locate interpolated frames. These existing approaches achieve desirable detection results, but still can not identify the adopted MC-FRUC technique for suspicious videos. In practical forensics cases, users may want to further identify the specific MC-FRUC technique after knowing that a candidate video has been up-converted. This is an essential step towards estimating the key parameters of MC-FRUC techniques such as block size involved in ME and MCI, which is actually an in-depth goal of passive video forensics.

In this paper, we propose a blind forensics approach to identify the adopted MC-FRUC technique for suspicious videos. By theoretically modelling residual signals caused by various MC-FRUC techniques, we verify that residual signals mainly occur in motion and texture-rich regions of non-static interpolated frames and have different variances among various MC-FRUC techniques. Consequently, the correlations of adjacent pixels in interpolated frames are also disturbed to different extents. Inspired by the strong ability of Markov statistics in characterizing adjacent pixel correlation, spatial and temporal Markov statistics features (ST-MSF) are designed to capture the differences of residual signals. Then, an Error-Correcting Output Code (ECOC) [25] framework on the basis of Ensemble classifier [26] is adopted to identify the adopted MC-FRUC technique. Since static interpolated frames and original frames have less or null residual signals, a pre-classifier is designed to select interpolated frames with prominent residual signals. The contributions are three-folds: First, this is the first attempt to identify the adopted MC-FRUC techniques in passive video forensics. Second, residual signal is firstly exploited as forgery trace to expose MC-FRUC, and thus the identification of MC-FRUC techniques is converted into a problem of discriminating the differences of residual signals inside interpolated frames. Third, a pre-classifier, which includes Scene Change Detection (SCD), Static Scene Detection (SSD) and Multi-Loop Detection Method (MLDM), is designed to improve detection accuracy by suppressing the side effects of static interpolated frames and original frames.

The rest of this paper is organized as follows. Section II summarizes existing MC-FRUC techniques. Section III introduces the concept of residual signal and discusses its possibility to be exploited as forgery trace for forensics. Section IV presents the proposed blind forensics approach. Experimental results are provided in Section V, and we conclude this paper in Section VI. For ease of understanding, the acronyms and notations in this paper are listed in Table I.

TABLE I: List of Acronyms and Notations.

MC-FRUC	Motion-Compensated Frame Rate Up-Conversion
ST-MSF	Spatial and Temporal Markov Statistics
ME	Motion Estimation
MCI	Motion Compensated Interpolation
MV	Motion Vector
ECOC	Error-Correcting Output Code
SCD	Scene Change Detection
SSD	Static Scene Detection
MLDM	Multi-Loop Detection Method
OBMC	Overlapped Block Motion Compensation
TFDM	Temporal Frame Difference Matrix
STFDM	Spatial and Temporal Frame Difference Matrix
MI	Mutual Information
F^0	Available Original Frame
F^α	Interpolated Frame
F^β	Absent Original Frame corresponding to Interpolated Frame
R	Residual Signal
fps	Frame Per Second
SR	Success Rate

II. Preliminaries of MC-FRUC techniques

A. Notation

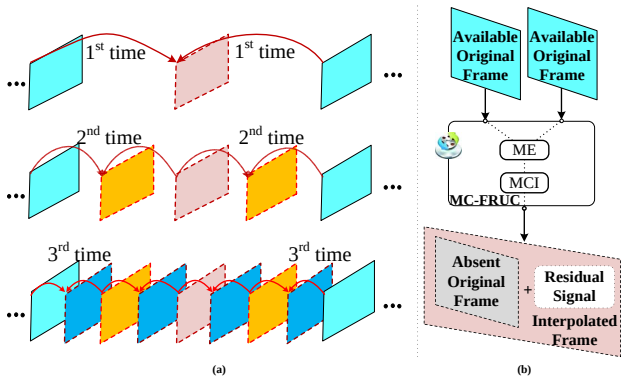


Fig. 1: (a) Frame interpolated by MC-FRUC ($\alpha = \frac{1}{8}$) ; (b) Residual signal.

Let $F(x, y, t) \in R^{m\times n\times k}$ be a video sequence, where (x, y) are spatial coordinates and t is temporal index, $x \in [1, ..., m]$, $y \in [1, ..., n]$, and $t \in [0, ..., k-1]$ [21]. Let α be an up-conversion factor ($0 < \alpha < 1$). That is, there are $(\frac{1}{\alpha} - 1)$ new interpolated frames between two original adjacent frames. Please note that these interpolated frames are generated from their adjacent frames by various MC-FRUC techniques. Thus, we assume that there exists an original frame corresponding to each interpolated frame. Since this kind of original frame does not actually exist, it is named as an absent original frame. Let $F^0(x, y, \alpha t)$ denote an available original frame, where αt is an integer instant. Let $F^\alpha(x, y, \alpha t)$ be an interpolated frame and $F^\beta(x, y, \alpha t)$ be the absent original frame corresponding to $F^\alpha(x, y, \alpha t)$. In the rest of the paper, we consider a block in the αt^{th} frame to be interpolated, where αt is a non-integer instant. Fig. 1(a) is an example of interpolated frames generated by MC-FRUC with $\alpha = \frac{1}{8}$. Apparently, MC-FRUC operations are repeated three times to generate seven interpolated frames, which are denoted by Pink, Orange, and Blue colors, respectively. There are also seven absent original frames corresponding to them.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2676162, IEEE Transactions on Circuits and Systems for Video Technology

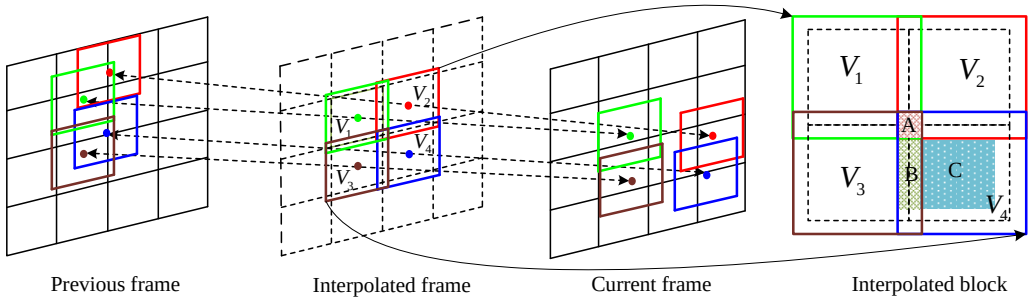


Fig. 2: The architecture of OBMC-based MC-FRUC. (Green, Red, Brown, and Blue denote reference blocks; Dashed box is interpolated block)

Then, residual signal is referred to as the difference between an interpolated frame and its absent original frame, as shown in Fig.1(b). That is,

$$R(x,y,\alpha t)=\begin{cases} F^{\beta}(x,y,\alpha t)-F^{\alpha}(x,y,\alpha t) & \alpha t \text{ is a noninteger} \\ null & \text{otherwise} \end{cases} \tag{1}$$

In brief, MC-FRUC is mathematically expressed as an estimation $F^{\alpha}(x,y,\alpha t)$ of absent original frame $F^{\beta}(x,y,\alpha t)$ by minimizing residual signal $R(x,y,\alpha t)$.

B. MC-FRUC techniques

MC-FRUC typically includes two key modules: ME and MCI. ME is to estimate spatial displacement, i.e., MV, of the same pixel in adjacent frames. MCI exploits the estimated MVs to construct an interpolated frame. Thus, the strategies of ME and MCI determine the performance of MC-FRUC, which also differentiate various MC-FRUC techniques.

There are four classes of ME strategies for MC-FRUC: (i) unidirectional ME (UME) [12], [13]; (ii) bidirectional ME (BME) [14]-[16]; (iii) unidirectional and bidirectional ME (UBME) [17]-[19]; and (iv) multiple hypotheses ME (MHME) [20]. Since there are great differences among these ME strategies, they will have distinct impacts on resultant videos.

Existing MCI schemes can also be divided into several categories: direct MCI (DMCI) [10], overlapped block motion compensation (OBMC) [13], adaptive OBMC (AOBMC) [15], dual weighted based joint OBMC (MCMP) [17], and direction-select OBMC (DSME) [18], etc. Among them, OBMC is the most popular MCI technique, which is originally used in video encoder to alleviate blocking artifacts. For the OBMC-based MC-FRUC, each interpolated block is synthesized as a weighted average of reference blocks, as shown in Fig. 2. Let V_1, V_2, V_3 and V_4 be four neighboring blocks. Their MVs are (v_{1x},v_{1y}), (v_{2x},v_{2y}), (v_{3x},v_{3y}) and (v_{4x},v_{4y}), respectively. Interpolated blocks in region A, B and C, which overlap with four blocks (V_1, V_2, V_3 and V_4), two blocks (V_3 and V_4) or one block V_4, respectively. Each pixel of the interpolated block is obtained by

$$\begin{aligned} &F^{\alpha}(x,y,\alpha t)= \\ &\tfrac{1}{2}\sum_{i=1}^{\lambda}\omega_i(x,y)\sum_{j=5-i}^{4}[F^{0}(x+v_{jx},y+v_{jy},\alpha(t-1)) \\ &+F^{0}(x-v_{jx},y-v_{jy},\alpha(t+1))] \end{aligned} \tag{2}$$

where $\lambda\in\{1,2,4\}$, $\sum_{i=1}^{\lambda}\omega_i(x,y)=1$, and $\omega_i(x,y)$ is the weighting coefficient determined by the relative position (x,y) within an interpolated block. Other MCI methods can be regarded as the variations of OBMC with different weighting mechanisms. For instance, when $\lambda=1$ and $\omega_1(x,y)=1$, OBMC is turned into DMCI [10]. When it meets

$$\omega_i^{'}(x,y)=\frac{\Phi_{V_i}[v_{ix,iy}]\cdot\omega_i(x,y)}{\sum_{\lambda=1}^{4}\Phi_{V_i}[v_{ix+\lambda,iy+\lambda}]\cdot\omega_{i+\lambda}(x,y)}$$

$$\Phi_{V_i}[v_{ix,iy}]=\frac{SBAD[V_i,v_{ix,jy}]}{SBAD[V_i,v_{ix+\lambda,jy+\lambda}]},$$

where $SBAD[V_i,v_{ix,jy}]$ is the sum of bilateral absolute difference for current block V_i. It is actually AOBMC [15]. Thus, OBMC can serve as the unified form to represent existing MCI methods (Please refer to [12]-[20] for details).

III. Residual Signal and Its Forensics Role

Similar to block-based ME in video coding, there is also an assumption for the ME involved in MC-FRUC that all the pixels in an interpolated block share the same MV [10], [27]. However, it does not always hold because of non-translational motion or non-rigid object itself. This will lead to residual signal, which is quite similar to prediction residue in block-based video coding. Actually, residual signal is the difference of interpolated pixels when they are obtained by estimated MV and true MV, respectively. In this section, we firstly model residual signal, and then analyze its possibility as forgery clue for the identification of MC-FRUC techniques. For explanatory simplicity, we only consider 1-D signal, which can also be extended to 2-D signal in a straightforward way.

A. Modeling of Residual Signal in an Unified Form

We assume that $F^{0}(x,\alpha(t-1))$ and $F^{\alpha}(x,\alpha t)$ come from the same underlying intensity function with MV $v_{x,t-1}$ and additive noise [10]. Similarly, $F^{\alpha}(x,\alpha t)$ and $F^{0}(x,\alpha(t+1))$ have the same MV $v_{x,t+1}$ and additive noise. Thus, $F^{\alpha}(x,\alpha t)$ is correlated with $F^{0}(x,\alpha(t-1))$ and $F^{0}(x,\alpha(t+1))$ as

$$F^{\alpha}(x,\alpha t)=F^{0}(x+v_{x,t-1},\alpha(t-1))+n(x,\alpha(t-1)) \tag{12}$$

$$F^{\alpha}(x,\alpha t)=F^{0}(x-v_{x,t+1},\alpha(t+1))+n(x,\alpha(t+1)) \tag{13}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2676162, IEEE Transactions on Circuits and Systems for Video Technology

where $n(x,\alpha\nu)$ is additive noise with zero mean and variance $\sigma^2_{\alpha\nu}$, and $\alpha\nu \in [0,...,k-1]$. Besides, $n(x,\alpha\nu)$ is assumed to be independent of $F^\alpha(x,\alpha t)$, $F^0(x,\alpha(t-1))$ and $F^0(x,\alpha(t+1))$ [10].

First, let ξ be an arbitrary pixel within an interpolated block V_i. N is the block size and its motion vector is v_ξ, $0 \le \xi \le N-1$. Thus, $F^\alpha(\xi,\alpha t)$, $F^0(\xi+v_{N/2},\alpha(t-1))$ and $F^0(\xi-v_{N/2},\alpha(t+1))$ are the same except for their noise components. If v_ξ keeps stable and temporal symmetry in whole block, it is equal to $v_{N/2}$, but in general this does not hold. By expanding $F^0(\xi+v_{N/2},\alpha(t-1))$, and $F^0(\xi-v_{N/2},\alpha(t+1))$ with (1), (3), and (4) and omitting higher order terms, the first-order approximations of $R(\xi,\alpha t)$ between $F^\alpha(\xi,\alpha t)$ and $F^\beta(\xi,\alpha t)$ can be expressed as

$$\begin{aligned} R(\xi,\alpha t) &= F^\beta(\xi,\alpha t) - F^\alpha(\xi,\alpha t) \\ &\approx \tfrac{1}{2}\{(v_\xi - v_{N/2})\tfrac{\partial}{\partial\xi}F^0(\xi+v_{N/2},\alpha(t-1)) \\ &+(v_{N/2}-v_\xi)\tfrac{\partial}{\partial\xi}F^0(\xi-v_{N/2},\alpha(t+1)) \\ &+n(\xi,\alpha(t-1))+n(\xi,\alpha(t+1))\} \end{aligned} \tag{3}$$

To simplify computation, we further assume that $\frac{1}{2}\{n(\xi,\alpha(t-1))+n(\xi,\alpha(t+1))\} \approx n(\xi,\alpha t)$ [10]. The difference between motion at (ξ) and that at $(N/2)$ follows distribution of $N(0,c^2(\xi-N/2)^2)$, where c is constant that represents the amount of fluctuation of motion [27], [28]. Meanwhile, $\frac{\partial}{\partial\xi}F^0(\xi+v_{N/2},\alpha(t-1))$ and $\frac{\partial}{\partial\xi}F^0(\xi-v_{N/2},\alpha(t+1))$ are assumed to be statistically independent of v_ξ. Consequently, the mean and variance of $R(\xi,\alpha t)$ can be expressed as

$$E[R(\xi,\alpha t)]=0,\ E[R^2(\xi,\alpha t)] = \hbar(\xi-N/2)^2+\sigma^2_{\alpha t} \tag{4}$$

where $\hbar = \frac{1}{4}[c^2\rho^2_{t-1}+c^2\rho^2_{t+1}+2c^2\rho_{t-1}\rho_{t+1}\rho_{t-1,t+1}]$,

$$\rho^2_{t-1} = E\left[\left(\tfrac{\partial}{\partial\xi}F^0(\xi+N/2,\alpha(t-1))\right)^2\right],$$
$$\rho^2_{t+1} = E\left[\left(\tfrac{\partial}{\partial\xi}F^0(\xi-N/2,\alpha(t+1))\right)^2\right],$$

and $\rho_{t-1,t+1}$ is the correlation coefficient between $\frac{\partial}{\partial\xi}F^0(\xi+N/2,\alpha(t-1))$ and $\frac{\partial}{\partial\xi}F^0(\xi-N/2,\alpha(t+1))$.

After modeling the residual signal in an interpolated block, we further derive the residual signal of OBMC. Since OBMC can serve as the unified form of existing MCI methods, the derived results can be extended to other MCI methods.

Let V_1 and V_2 be two adjacent blocks in $F^\alpha(x,\alpha t)$, and $N/2$ and $3N/2$ be their center positions, respectively. ω_1 and ω_2 are the shifted versions of weighting coefficient $\omega(\xi)$, which are centered at $N/2$ and $3N/2$, respectively [13]. $\omega(\xi)$ is defined within $[-N,N]$. For $\xi \in [0,N]$, it satisfies

$$\begin{cases} 0 \le \omega(\xi) \le 1, \\ \omega(\xi)+\omega(N-\xi)=1, \\ \omega(-\xi)=\omega(\xi). \end{cases}$$

For a pixel $\xi \in [N/2,N]$ in the block V_1, its interpolated process by OBMC is firstly considered. When the estimated and the true MVs of V_1 and V_2 are used, the first-order approximations of residual signals are denoted by $R_1(\xi,\alpha t)$ and $R_2(\xi,\alpha t)$, respectively. Since $R_1(\xi,\alpha t)$ and $R_2(\xi,\alpha t)$ share the same derivations as $R(\xi,\alpha t)$, residual signal $R_{OBMC}(\xi,\alpha t)$ is also defined as the difference between $F^\beta(\xi,\alpha t)$ and $F^\alpha(\xi,\alpha t)$.

$$\begin{aligned} R_{OBMC}(\xi,\alpha t) &= F^\beta(\xi,\alpha t) - F^\alpha(\xi,\alpha t) \\ &= \omega_1(\xi)R_1(\xi,\alpha t)+\omega_2(\xi)R_2(\xi,\alpha t) \end{aligned} \tag{5}$$

$$\begin{aligned} R_1(\xi,\alpha t) &\approx \tfrac{1}{2}\{(v_\xi - v_{N/2})\times\tfrac{\partial}{\partial\xi}F^0(\xi+v_{N/2},\alpha(t-1)) \\ &+(v_{N/2}-v_\xi)\times\tfrac{\partial}{\partial\xi}F^0(\xi-v_{N/2},\alpha(t+1))\} \\ &+n(\xi,\alpha t) \\ R_2(\xi,\alpha t) &\approx \tfrac{1}{2}\{(v_\xi - v_{3N/2})\times\tfrac{\partial}{\partial\xi}F^0(\xi+v_{3N/2},\alpha(t-1)) \\ &+(v_{3N/2}-v_\xi)\times\tfrac{\partial}{\partial\xi}F^0(\xi-v_{3N/2},\alpha(t+1))\} \\ &+n(\xi,\alpha t) \end{aligned} \tag{6}$$

Similar to the derivation of (6), the means and the variances of $R_1(\xi,\alpha t)$ and $R_2(\xi,\alpha t)$ are derived as follows.

$$\begin{aligned} &E[R_1(\xi,\alpha t)]=0,\ \ E[R_2(\xi,\alpha t)]=0, \\ &E[{R_1}^2(\xi,\alpha t)] = \hbar(\xi-N/2)^2+\sigma^2_{\alpha t} \\ &E[{R_2}^2(\xi,\alpha t)] = \hbar(3N/2-\xi)^2+\sigma^2_{\alpha t} \end{aligned} \tag{7}$$

Further, the cross-correlation of $R_1(\xi,\alpha t)$ and $R_2(\xi,\alpha t)$ is given in (10), where ρ_v is the correlation coefficient between $(v_\xi - v_{N/2})$ and $(v_\xi - v_{3N/2})$, and ρ_{1f}, ρ_{2f}, ρ_{3f} and ρ_{4f} are respectively the correlation coefficients between
$\frac{\partial}{\partial\xi}F^0(\xi+v_{N/2},\alpha(t-1))$ and $\frac{\partial}{\partial\xi}F^0(\xi+v_{3N/2},\alpha(t-1))$,
$\frac{\partial}{\partial\xi}F^0(\xi+v_{N/2},\alpha(t-1))$ and $\frac{\partial}{\partial\xi}F^0(\xi-v_{3N/2},\alpha(t+1))$,
$\frac{\partial}{\partial\xi}F^0(\xi-v_{N/2},\alpha(t+1))$ and $\frac{\partial}{\partial\xi}F^0(\xi+v_{3N/2},\alpha(t-1))$,
$\frac{\partial}{\partial\xi}F^0(\xi-v_{N/2},\alpha(t+1))$ and $\frac{\partial}{\partial\xi}F^0(\xi-v_{3N/2},\alpha(t+1))$.

For a pixel $\xi \in [0,N/2]$, it can be computed by using V_1 and the MV of its left neighboring block. Consequently, Equations (7) and (10) are combined to derive (11).

$$\begin{aligned} &E[R_1(\xi,\alpha t)\cdot R_2(\xi,\alpha t)] \\ &= \sigma^2_{\alpha t}+\tfrac{1}{4}E[\{(v_\xi - v_{N/2})\tfrac{\partial}{\partial\xi}F^0(\xi+v_{N/2},\alpha(t-1))+(v_{N/2}-v_\xi)\tfrac{\partial}{\partial\xi}F^0(\xi-v_{N/2},\alpha(t+1))\} \\ &\quad\times\{(v_\xi - v_{3N/2})\tfrac{\partial}{\partial\xi}F^0(\xi+v_{3N/2},\alpha(t-1))+(v_{3N/2}-v_\xi)\tfrac{\partial}{\partial\xi}F^0(\xi-v_{3N/2},\alpha(t+1))\}] \\ &= \tfrac{1}{4}\{\rho_v c^2(\xi-N/2)(3N/2-\xi)\rho_{1f}\rho^2_{t-1}-\rho_v c^2(\xi-N/2)(3N/2-\xi)\rho_{2f}\rho_{t-1}\rho_{t+1} \\ &\quad-\rho_v c^2(\xi-N/2)(3N/2-\xi)\rho_{3f}\rho_{t-1}\rho_{t+1}+\rho_v c^2(\xi-N/2)(3N/2-\xi)\rho_{4f}\rho^2_{t-1}\}+\sigma^2_{\alpha t} \\ &= \tfrac{1}{4}\rho_v c^2(\xi-N/2)(3N/2-\xi)(\rho_{1f}\rho^2_{t-1}-\rho_{2f}\rho_{t-1}\rho_{t+1}-\rho_{3f}\rho_{t-1}\rho_{t+1}+\rho_{4f}\rho^2_{t-1})+\sigma^2_{\alpha t} \end{aligned} \tag{10}$$

$$\begin{aligned} &E[R_{OBMC}(\xi,\alpha t)]=0, and\ \ E[R^2_{OBMC}(\xi,\alpha t)] = E[(\omega_1(\xi)R_1(\xi,\alpha t)+\omega_2(\xi)R_2(\xi,\alpha t))^2] \\ &= E[(\omega_1(\xi)R_1(\xi,\alpha t))^2]+E[(\omega_2(\xi)R_2(\xi,\alpha t))^2]+2E[\omega_1(\xi)R_1(\xi,\alpha t)\omega_2(\xi)R_2(\xi,\alpha t)] \\ &= \hbar\omega^2_1(\xi)(\xi-N/2)^2+\hbar\omega^2_2(\xi)(N-|\xi-N/2|)^2+\sigma^2_{\alpha t} \\ &\quad+2\omega_1(\xi)\omega_2(\xi)\rho_v c^2(\xi-N/2)(N-|\xi-N/2|)(\rho_{1f}\rho^2_{t-1}-\rho_{2f}\rho_{t-1}\rho_{t+1}-\rho_{3f}\rho_{t-1}\rho_{t+1}+\rho_{4f}\rho^2_{t-1}) \end{aligned} \tag{11}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2676162, IEEE Transactions on Circuits and Systems for Video Technology

B. Residual signal as the clue for MC-FRUC forensics

It is well-known that the $R_{OBMC}(\xi, \alpha t)$ in interpolated frames of a static scene or interpolated blocks in texture-smooth region must be null because the whole block keeps steady motion (i.e., $c = 0$ or $v_\xi = v_{N/2}$). Thus, we need to exclude the negative effect of static interpolated frames.

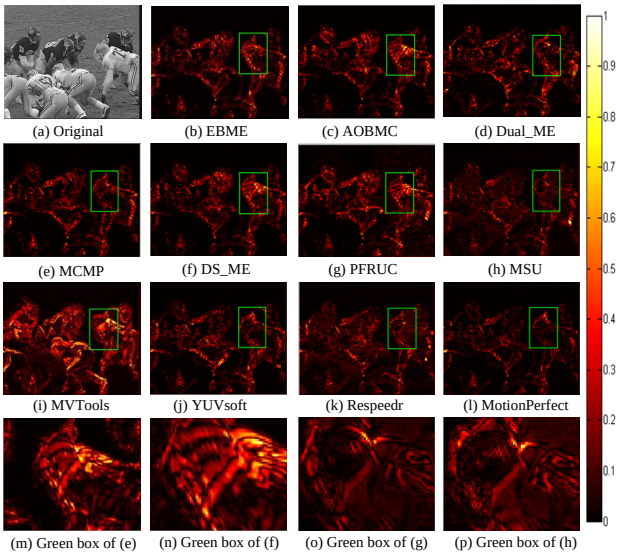


Fig. 3: The original 4^{th} frame of "*Football*" sequence and residual signals; (m)-(p) are highlighted versions of the marked green boxes in (e)-(h), respectively.

TABLE II: Comparisons of eleven MC-FRUC techniques.

MC-FRUC	Technical details
EBME [14]	Extended BME + OBMC.
AOBMC [15]	BME + AOBMC.
DualME [16]	Dual ME + OBMC.
MCMP [17]	Multi-channel mixed pattern UBME + DW-OBMC.
DSME [18]	Direction-select UBME + DS-OBMC.
PFRUC [19]	Hierarchical MVF construction + OBMC + Particle-based motion trajectory calibration.
MSU	Quarter pixel accuracy + Bilinear interpolation + The same sharpness as the original ones.
MVTools2	Block-matching ME + Half a pixel precision + Sharper Wiener interpolation + Pixel-based MC.
YUVsoft	Half a pixel precision+ Motion adaptation + Scene change detection.
Respeedr	Frame-blending + key-frame technologies.
MotionPerfect	MCI + Morphing algorithms.

For non-translational motion, non-rigid object and texture-rich regions (i.e., $v_\xi \neq v_{N/2}$), there must exist residual signals. As shown in Fig. 3, the interpolated frames are obtained by using the 3^{rd} and 5^{th} frame of "*Football*" sequence in CIF format with eleven MC-FRUC techniques. They include six most representative MC-FRUC techniques and five public-available FRUC softwares[1], summarized in Table II. Fig.3 (b)-(l) are the residual signals between interpolated frame and the 4^{th} frame, which are highlighted by using "*hot*" color map. From it, we observe that the dark areas are those relatively static regions and texture-smooth regions. There are also subtle variations in the bright regions, which correspond to motion and texture-rich regions. Especially from the boxes marked with Green, we observe from Fig.3 (m)-(p) that the residual signals are also different from each other. Therefore, we need to highlight motion and texture-rich regions of non-static interpolated frames for blind forensics.

[1]Available on http://www.wondershare.com/multimedia-tips/slow-motion-software.html.

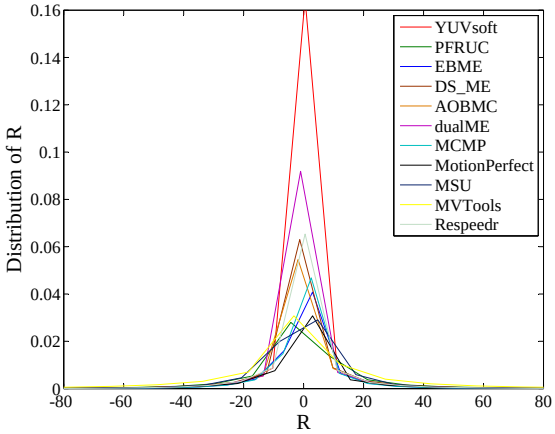


Fig. 4: Laplace distributions of residual signals.

For various MC-FRUC methods, their residual signals have different variances. Fig. 4 shows the distributions of residual signals. From it, residual signals follow Laplace distributions with different variances. This motivates us to exploit residual signal as forgery trace to identify various MC-FRUC techniques. However, the variance of residual signal can not be directly employed to identify the adopted MC-FRUC method because of the diversities of video content and the complexities of motion and texture. Instead, residual signal should be treated in a novel manner for blind forensics. Since the residual signals caused by various MC-FRUC methods have different variances, the inherent correlation among adjacent pixels in motion and texture-rich regions will be inevitably destroyed along temporal and spatial directions. Therefore, the Markov features are adopted to capture this correlation in this paper, the effectiveness of which will discussed in next Section.

IV. Proposed Blind MC-FRUC Forensics System

For blind forensics, the identification of tampering technique is a deeper goal than simple binary decision on whether a candidate video is forged or not. The proposed forensics system makes the first attempt to identify the adopted MC-FRUC technique, and its block diagram is shown in Fig.5. There are three key components: spatial and temporal Markov statistics feature (ST-MSF) extraction, pre-classifier and the identification of MC-FRUC techniques. As claimed in previous section, various MC-FRUC techniques introduce distinct residual signals, which destroy the inherent correlations among adjacent pixels of interpolated frames to some extent. This is a desirable property for the identification of MC-FRUC. Since Markov statistics has been proved to be simple yet effective in characterizing adjacent pixel correlation [29], [30], [31], it is straightforward that our proposed ST-MSFs are effective to model the differences of residual signals as

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2676162, IEEE Transactions on Circuits and Systems for Video Technology

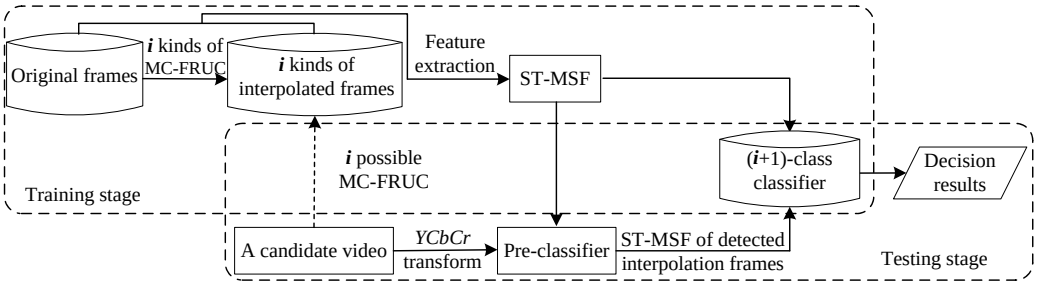


Fig. 5: Overview of the proposed blind MC-FRUC forensics system.

well. However, if ST-MSFs are directly extracted from any frame of a candidate video, the accuracy will be influenced because there exists static interpolated frames without residual signals. Therefore, a pre-classifier is proposed to exclude static interpolated frames so as to improve the accuracy. Finally, a multi-class classifier is exploited for the identification of various MC-FRUC techniques.

A. ST-MSF Extraction and its Effectiveness Analysis

As claimed in Section III-B, motion and texture-rich regions of non-static interpolated frames have prominent residual signals. Therefore, they should be highlighted by suppressing temporal and spatial redundancies of a candidate video.

1) Suppress spatial and temporal redundancies: Firstly, a temporal frame difference matrix (TFDM) is defined among three successive frames to suppress temporal redundancies.

$$\begin{aligned} \textit{TFDM}(x,y,t) = \tfrac{1}{2}[&(F(x,y,t) - F(x,y,t-1)) \\ &+(F(x,y,t) - F(x,y,t+1))] \end{aligned} \tag{12}$$

where t is the frame index. For the first and last frame of a candidate video, their *TFDMs* are computed between the first two frames and the last two frames, respectively.

Further, a spatial frame difference filter is defined to reduce spatial redundancies.

$$\textit{STFDM}^{\downarrow}(x,y,t) = \textit{TFDM}(x,y,t) - \textit{TFDM}(x+1,y,t) \tag{13}$$

where $\textit{STFDM}^{\downarrow}(x,y,t)$ is the up-to-down spatial difference.

To measure the information redundancy between $F(x,y,t)$ and $\textit{STFDM}^{\downarrow}(x,y,t)$, mutual information (MI) is exploited here because it is a basic concept in information theory which measures the statistical dependence between two random variables [32].

$$\begin{aligned} MI(A,B) &= H(A) + H(B) - H(A,B) \\ &= H(A) - H(A|B) \end{aligned}$$

where $H(A)$ and $H(B)$ are the entropies of A and B, respectively. $H(A,B)$ is the joint entropy, and $H(A|B)$ is the conditional entropy of A given B. For most videos[2], $MI(F(x,y-1,t),F(x,y,t))$ and $H(F(x,y-1,t))$ are about 1.8350 and 4.8929, respectively; and $MI(STFDM^{\downarrow}(x,y,t),F(x,y,t))$ and $H(STFDM^{\downarrow}(x,y,t))$ are about 0.2015 and 3.1974, respectively. Thus, when $F(x,y,t)$ is known, the entropy of $F(x,y-1,t)$ will decrease by 37.50%, but the entropy of $STFDM^{\downarrow}(x,y,t)$ will decrease by $\frac{0.2015}{3.1974} = 6.30\%$. That is, the dependence between $\textit{STFDM}^{\downarrow}(x,y,t)$ and $F(x,y,t)$ decreases to a fairly low level from the information theory point of view [32]. Therefore, temporal and spatial redundancies are effectively suppressed by *STFDM*, and thus residual signals are highlighted within motion and texture-rich regions along temporal and spatial directions.

2) ST-MSF Extraction: Residual signals destroy the inherent correlations among adjacent pixels. To differentiate the residual signals caused by various MC-FRUC techniques, we model such correlations with Markov chain, which are computed from empirical probability transition matrices. Specifically, the Markov features are extracted in spatial-temporal domain, which are denoted as *ST-MSF*. The steps of *ST-MSF* extraction are summarized as follows.

Step1: Compute $\textit{TFDM}(x,y,t)$ with (12) for each frame.

Step2: Compute $\textit{STFDM}(x,y,t)$ along eight directions. Eight direction-specific quantities are denoted by superscripts $\{\leftarrow,\rightarrow,\downarrow,\uparrow,\searrow,\nwarrow,\nearrow,\swarrow\}$, respectively. An example is the up-to-down spatial difference $\textit{STFDM}^{\downarrow}(x,y,t)$ as (13), and the other seven difference matrices are defined similarly.

Step3: Truncate $\textit{STFDM}^{\downarrow}(x,y,t)$ with a threshold T. If it is bigger than T or smaller than $-T$, it will be replaced with T or $-T$, respectively. The selection of T will be discussed in Section V-B.

Step4: Model $\textit{STFDM}^{\downarrow}(x,y,t)$ with the first-order Markov process along the vertical direction, and compute the empirical matrices as follows.

$$\begin{aligned} M^{\downarrow}_{u,v,t} = \frac{1}{m \times n} \sum_x \sum_y \Pr(&\textit{STFDM}^{\downarrow}(x+1,y,t) = u \\ &\left|\textit{STFDM}^{\downarrow}\,(x,y,t) = v\right.) \end{aligned}$$

where $u,v \in \{-T,\cdots,T\}$. If $\Pr(\textit{STFDM}^{\downarrow}(x,y,t)=v)=0$, then $M^{\downarrow}_{u,v,t} = 0$. The empirical matrices of the rest seven directions can be computed similarly.

Step5: Reduce feature dimensions. Specifically, four horizontal or vertical matrices, and four diagonal matrices are separately averaged to form two feature subsets [30], [31]. Thus, the ST-MSF of a frame is defined as

$$S^t_{1,\ldots,s} = \frac{1}{4}(M^{\leftarrow}_t + M^{\rightarrow}_t + M^{\uparrow}_t + M^{\downarrow}_t) \tag{14}$$

$$S^t_{s+1,\ldots,2s} = \frac{1}{4}(M^{\nearrow}_t + M^{\swarrow}_t + M^{\searrow}_t + M^{\nwarrow}_t) \tag{15}$$

where M_t, $S^t_{1,\ldots,s}$, and $S^t_{s+1,\ldots,2s}$ have the same dimensions of $s = (2 \times T + 1)^2$.

[2]from the online video Databases (http://media.xiph.org/video/derf/#)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2676162, IEEE Transactions on Circuits and Systems for Video Technology

For a candidate video encoded with H.264/AVC or MPEG-2 codec, I frames and motion predicted residues are compressed by a lossy compression scheme. As pointed by [33], with similar dimensionality, the feature sets extracted from DCT domain are always better than those from pixel domain for compressed videos. It is also well-known that once the pixel values are modified, the corresponding coefficients in frequency domain are inevitably changed. That is, though residual signal is derived in pixel domain, it is also valid for frequency domain. Therefore, the ST-MSFs are extracted from DCT-domain for candidate compressed videos. There are two additional steps to be firstly executed as follows.

Step1, Apply 8×8 block Discrete Cosine Transform on the decoded Y components of a candidate encoded video, and the corresponding DCT coefficient array is obtained.

Step2, Take absolute value of the DCT coefficients. Then, the obtained arrays are used to replace frames in (12), and the rest operations remain the same.

3) Effectiveness analysis of ST-MSF: As claimed in Sections II-B and III-B, the performance of MC-FRUC depends on the strategies of ME and MCI, and the differences mainly exist in motion and texture-rich regions of non-static interpolated frames. In the following, we discuss the differences among various MC-FRUC techniques for these regions under two cases.

In the first case, we assume that various MC-FRUC techniques have the same MCI scheme but different searching patterns of ME. From Fig. 2, this may lead to block V_1 with different MVs (v_{1x}, v_{1y}). That is, block V_1 may come from different blocks in the previous and current frames. As a result, block V_1 may exhibit differences among various MC-FRUC techniques. Similarly, blocks V_2, V_3 and V_4 also exhibit differences among various MC-FRUC techniques. Then, the same MCI method is exploited to obtain each interpolated block from them. Thus, there inevitably exists some differences among interpolated blocks. In this case, the intra-frame and inter-frame correlations also have differences among various MC-FRUC techniques.

In the second case, we assume that various MC-FRUC techniques have the same ME pattern but different MCI strategies. Since they share the same ME pattern, blocks V_1, V_2, V_3, or V_4 keep unchanged. However, the weighting coefficients are different because of different MCI methods. That is, each interpolated block is obtained by the same reference blocks but averaged with different weighting coefficients. Thus, there are also differences among various MC-FRUC techniques for intra-frame and inter-frame correlations.

Actually, different MC-FRUC techniques may have distinct ME and MCI strategies simultaneously. Thus, each interpolated block may have different reference blocks and weighting coefficients. Fig. 6 shows the averaged ST-MSFs for eleven MC-FRUC methods. From it, the shapes and intensities of ST-MSFs are more or less different between original frames and interpolated frames. This implies the changes of intra-frame and inter-frame correlations. Furthermore, the shapes and intensities of ST-MSFs are slightly different from each other, especially in those regions marked with red boxes. This verifies that the devised ST-MSFs are effective to expose

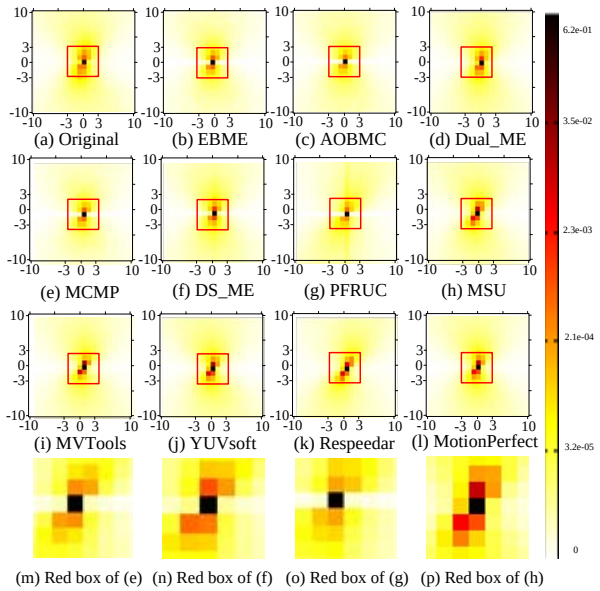


Fig. 6: Average ST-MSF of (a) original frame; (b)-(l) interpolated frames; (m)-(p) highlighted regions that marked with red boxes in (e)-(h).

interpolated frames and further identify the adopted MC-FRUC technique.

B. Pre-classifier

A pre-classifier is designed to restrain the negative effects of original frames and static interpolated frames by excluding them from a candidate video. The reasons behind this are two-folds. First, residual signals only exist inside interpolated frames, and there are less residual signals in static interpolated frames. Thus, interpolated frames with motion and rich-texture should be firstly differentiated from original frames and static interpolated frames. Second, the amount and positions of interpolated frames are not fixed in up-converted videos due to different up-conversion factors. Figure 7(a) is the flowchart of the proposed pre-classifier, which includes three components: Scene Change Detection (SCD), Static Scene Detection (SSD) and Multi-Loop Detection Method (MLDM).

1) SCD: For video frames with rapid scene change, there are prominent changes in motion regions, which might be wrongly regarded as residual signals. Thus, the first frame of a new scene and the last frame of previous scene might be wrongly decided as an interpolated frames. However, they do not contain any residual signals, which might lead to incorrect identification of MC-FRUC technique. Actually, when video frames with rapid scene change are up-converted by MC-FRUC, the first frame of a new scene is usually replicated as an interpolated frame to alleviate severe blocking artifacts. This is the reason that SCD is involved in most commercial MC-FRUC software. To avoid this kind of erroneous judgment, the SCD approach [34] is adopted to divide a candidate video into several scene segments. Meanwhile, when the structural similarity index between the first two frames of each scene is larger than 0.995 [22], the first frame is also labelled as an interpolated frame.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2676162, IEEE Transactions on Circuits and Systems for Video Technology

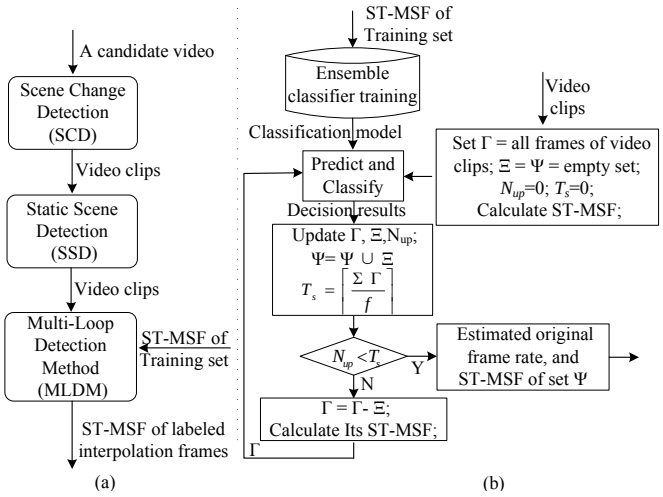


Fig. 7: Flowchart of (a) pre-classifier; (b) MLDM.

2) SSD: When an interpolated frame is obtained from two successive static frames, its residual signal is null. This will invalidate the following identification of MC-FRUC. Thus, if the percentage of zero values in *TFDM* is more than 99.5% for a frame, it is labelled as a static frame.

3) MLDM: After removing interpolated frames with null residual signal by SCD and SSD, there might still exist some original frames due to the diversities of up-conversion factors and MC-FRUC technique. Thus, a MLDM scheme is presented to choose true forgery frames. The flowchart of MLDM is shown in Fig. 7(b). It is an iterative approach motivated by the facts that interpolated frames must locate between two reference frames and the intermediate frame of three-successive detected frames is also an interpolated frame. The steps of MLDM are summarized as follows.

Initialization: Let Γ, Ξ, and Ψ be the frame sets of candidate video, detected interpolated frames in each loop, and all the detected interpolated frames, respectively. Their initial values are the whole frames of candidate video, $\emptyset$ and $\emptyset$, respectively. $\emptyset$ stands for an empty set.

Let N_{up} and T_s be the number of interpolated frames in each loop and the duration time of the updated Γ, respectively. Their initial values are 0.

Step1: Extract the ST-MSFs from training samples, which are then trained to construct classification model by the Ensemble classifier.

Step 2: Calculate the ST-MSFs of Γ. Then, Then, decision results are obtained after the classification model is used to test on these ST-MSFs by the Ensemble classifier.

Step 3: Update Γ as the scope of frames from the first detected frame to the last one in the decision results. Intermediate frames of each detected 3-successive frames are grouped into Ξ, and its frame number is recorded in N_{up}, Ψ is modified to be the union set of Ψ and Ξ, and $T_s = \lceil \frac{\Sigma \text{ updated } \Gamma}{f} \rceil$, where Σ stands for the numbers of set elements, f is the frame rate of a candidate video, and $\lceil \cdot \rceil$ denotes the ceiling operator.

Step 4: If it meets $N_{up} < T_s$, goto step5; Else update Γ as the difference set of Γ and Ξ; Goto Step2;

Step 5: Terminate and output the original frame rate as $\frac{\Sigma \Gamma}{T}$ fps where $T = \frac{\Sigma(\Gamma+\Psi)}{f}$, and the ST-MSFs of Ψ.

In summary, there are three key issues for MLDM in each iteration: the scope of frames to be detected, how to choose interpolated frames, and the condition of termination. For the first issue, the whole frames of candidate video is used as an initial estimate, and then it is updated as the scope of frames which are involved from the first detected frame to the last one in each loop. Thus, it covers all true interpolated frames and their left and right adjacent frames. Meanwhile, this can avoid early termination for spliced videos because the proportion of up-converted video clips to the unaltered ones may be fairly small. For the second issue, interpolated frames are gradually selected by deciding the intermediate frame as an interpolated frame in each slide window of three-successive detected frames, in which only one frame overlaps with each slide window. For the condition of termination, it is determined by the repeated times of MC-FRUC operations. However, the repeated times is unknown in practical forensics cases. Thus, whether N_{up} is larger than T_s in each loop iteration is used as an alternative. That is, when there is no interpolated frame in an one-second video clip, MLDM is terminated.

Actually, MLDM is an inverse process of generating interpolated frames. Figure 1(a) is also an example of MLDM with three loops, where the first, second and third loop correspond to select those interpolated frames generated in the third, second and first time of MC-FRUC operation, respectively.

C. The identification strategy

After pre-classification, ST-MSFs are extracted from those detected interpolated-frames for the identification of various MC-FRUC techniques. In this paper, an Error-Correcting Output Code (ECOC) strategy [25] based on Ensemble classifier [26] with its default settings is exploited to turn a multi-class problem into binary sub-problems because the Ensemble classifier compromises well between computational complexity and detection accuracy, and the ECOC strategy is an excellent multi-class categorization tool as well. In the experiments, there are eleven MC-FRUC techniques. Thus, a twelve-class (including the original video without MC-FRUC as a special class) classifier is designed, and a strategy of pairwise coupling [35] is adopted. For the ECOC strategy, a discrete decomposition matrix (codematrix) is firstly defined for the twelve-class classification problem. Then, this problem is decomposed into $N = 11 \times 12/2 = 66$ binary sub-problems, i.e., dichotomies, according to the sequence of 0s and 1s of columns of the codematrix. After training the Ensemble classifier on these dichotomies, the ST-MSFs from pre-classifier are tested to output a binary vector. The final type is assigned to the class with the smallest Hamming distance between this vector and the codewords.

V. Experimental results and discussion

A. Experimental Setup

To verify the proposed approach[3], a series of experiments are done on a personal computer (64-bit AMD A8-5500 CPU

[3]The source codes are publicly available at https://drive.google.com/open?id=0Bw98LY740lfsMGQzckJpSllhT1k

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2676162, IEEE Transactions on Circuits and Systems for Video Technology

3.2GHz, 8.0GB RAM) with MATLAB R2014a Until now, there is still no open video database for MC-FRUC forgery detection and identification. Thus, we build a test video database by ourselves. Table III reports the used parameters. Fifteen videos in CIF format (352×288) and Ten videos in HD720p format (1280×720) are selected[4], which have different contents, diverse motion and texture complexities. These original videos are in YUV4:2:0 format and their original frame rate (fps_1) is 15 fps. Nine specified target frame rates (fps_2) are employed here. According to the repeated times of MC-FRUC operations, nine target frame-rates are divided into three classes: once ($20fps$, $24fps$, $25fps$, $30fps$); twice($36fps$, $45fps$, $60fps$) and thrice ($90fps$, $120fps$). Eleven MC-FRUC techniques and software are summarized in Table II. Please note that to achieve a low up-conversion factor, videos are repeatedly conducted with multiple MC-FRUC operations, as shown in Fig. 1(a). Meanwhile, optimized parameters are adopted for these MC-FRUC techniques to obtain desirable resultant videos. Please note that open tools YUVsoft and MSU only support integer FRUC. To prove the robustness of the proposed detector against lossy compression, JM18.6[5] and MPEG-2 codec[6] are used to encode these video sequences with the configurations summarized in the last two rows of Table III. For H.264/AVC codec, three configurations ($conf_1$, $conf_2$, and $conf_3$)are used under different QPs or GOP lengths. Two similar configurations ($conf_4$, and $conf_5$) are adopted for MPEG2 codec.

TABLE III: List of parameters used to build the dataset.

Name	Values
Videos	CIF: Akiyo, Bowing, Bus, Coastguard, Container, Flower, Football, Hall, Highway, Mobile, Mother-daughter, News, Silent, Tempete, Waterfall. HD720p: Ducks_take_off, Fourpeople, In_to_tree, Johnny, Mobcal, Old_town_cross, Park_joy, Parkrun, Shields, Stockholm.
MC-FRUC	See Table II.
$fps_1(fps)$	15.
$fps_2(fps)$	once: 20, 24, 25,30; twice: 36, 45, 60; thrice: 90, 120.
H.264	$conf_1$: QP 12; GOP 11, 17, 24, 30. $conf_2$: GOP 8; QP 12, 30, 42. $conf_3$: GOP 8; QP 12, 14, ..., 22.
MPEG2	$conf_4$: Q_Scales 8; GOP: 5,10. $conf_5$: GOP 10; Q_Scales: 2, 4, 6.

TABLE IV: The Tampered video datasets and parameters

Name	Important parameters
DB1	CIF videos; All fps_2; Uncompressed.
Fast	Bowing, Bus, Coastguard, Container, Flower, Football, Highway, Mobile, Silent, Tempete, Waterfall; fps_2: 30.
Static	Akiyo, Hall, bridge-close, bridge-far; fps_2: 30.
DB2	CIF videos; fps_2: 30, 60 and 120; H.264: $conf_1$ and $conf_2$; MPEG2: $conf_4$ and $conf_5$.
DB3	CIF videos; fps_2: 30, 60 and 120; H.264: $conf_3$.
DB4	CIF videos; fps_2: 30; Uncompressed; FRUC methods: OBMC [13] and MHME [20].
DB5	HD720 videos; fps_2:30, 60, and 120; Uncompressed; H.264: $conf_2$; MPEG2: $conf_4$.

[4]from the online video Databases (http://media.xiph.org/video/derf/#)
[5]Available on http://iphome.hhi.de/suehring/tml/
[6]Available on http://www.mpeg.org/MPEG/video

The tampered video datasets and their parameters are summarized in Table IV. For each original CIF video, the first tampered dataset, which is denoted as **DB1**, is constructed by directly up-converting it to each target frame rate with eleven MC-FRUC techniques, respectively. Since there are nine target frame rates, this dataset contains nine subsets. For example, one of the subsets is DB1 with $fps_2 = 30fps$. Next, DB1 with $fps_2 = 30fps$ is further divided into two subsets to verify the performance improvement by SSD. The first subset originates from four YUV sequences with static scene, which is denoted as **Static**. The second subset originates from eleven YUV videos with fast motions, denoted as **Fast**.

Then, DB1 with $fps_2 = 30fps, 60fps$ and $120fps$ are encoded with different configurations. That is, $conf_1$ and $conf_2$ for H.264/AVC, whereas $conf_4$ and $conf_5$ for MPEG2. We denote this tampered dataset as **DB2**. Moreover, the trend of accuracy is further verified when estimating the original frame-rate with H.264/AVC ($conf_3$) under different QPs for these subsets, denoted as **DB3**.

To assess the generalization ability of MC-FRUC identification, the original CIF videos are further up-converted with fps_2=$30fps$ by two unknown MC-FRUC techniques including OBMC [13] and MHME [20]. The tampered dataset is denoted as **DB4**. Note that OBMC is a simplified version of AOBMC without adaptive weighting mechanism, whereas MHME is an advanced MC-FRUC technique which exploits multiple hypothesis Bayesian to produce more realistic videos.

Ten HD720p videos are also up-converted with fps_2 including $30fps$, $60fps$ and $120fps$. Then, two encoding configurations ($conf_2$ for H.264/AVC and $conf_4$ for MPEG2) are adopted to encode these up-converted videos. We denote this whole HD720p tampered videos including uncompressed videos and compressed videos as **DB5**. Please note that since YUVsoft is limited to spatial resolution of 720×576, YUVsoft is not used here.

In each group of experiments, the ST-MSF features are extracted from pixel domain when candidate videos are uncompressed, otherwise directly from DCT domain. And, the Ensemble classifier [26] with its default settings is used for classification. Videos are randomly divided into two categories: 50% for training and the rest 50% for testing. The training and testing are repeated for 10 times, and the average results are reported as final detection accuracies.

Comparisons are made among the proposed approach and three state-of-the-art approaches including P. Bestagini *et al.* [21], Yao *et al.* [23] and Xia *et al.* [24]. Interpolated frames and original frames are denoted as positive samples (S_p) and negative samples (S_n), respectively. A widely-accepted metric F_1 [33], [36], which focuses on positive samples, is adopted for performance evaluation.

$$F_1 = \begin{cases} \frac{(\gamma^2+1)\cdot Precision \times Recall}{\gamma^2 \cdot Precision + Recall} & if \ \Sigma \ S_{tp} > 0 \\ 0 & if \ \Sigma \ S_{tp} = 0 \end{cases} \quad (16)$$

where

$$Precision = \frac{\Sigma \ S_{tp}}{\Sigma \ S_{tp} + \Sigma \ S_{fp}}, \ Recall = \frac{\Sigma \ S_{tp}}{\Sigma \ S_p}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2676162, IEEE Transactions on Circuits and Systems for Video Technology

and γ controls the balance between $Precision$ and $Recall$. Normally, γ is set to 1. S_{tp} and S_{fp} are true positive and false positive samples, respectively.

A threshold T_h is used to judge whether the proposed approach successfully estimates original frame rate or not. If it meets $F_1 \geq T_h$, the original frame rate is successfully estimated. If the estimated frame rate fluctuates about 5% around the true one, it is regarded that the original frame rate is correctly estimated [22]. Since $F_1 \geq T_h$ means $T_h/(2-T_h) \leq Recall \leq 1$ and $0 \leq |S_{fp}|\,/\,|S_{tp}| \leq 2/T_h - 1/Recall - 1$, T_h is calculated to be 94%. Thus, the success rate (*SR*) of frame rate estimation is defined as follows.

$$SR = \frac{\sum_{i=1}^{K} \delta_{F_1(i) \geq T_h}}{K} \tag{17}$$

where K is the amount of videos in an evaluation dataset, and if $F_1(i) \geq T_h$, $\delta(\cdot)$ equals 1 or else 0.

B. Choice of Truncated Threshold T

For the proposed method, there is only one parameter, namely truncated threshold T, to be determined. The dimensionality of the devised *ST-MSF* is $2 \times (2 \times T + 1)^2$, which is directly determined by T. If T is too small, the correlations between adjacent pixels in motion and texture-rich regions, which are captured by the Markov process, might be insufficient to distinguish interpolated frames from original ones. On contrary, a too big T means a very high dimensionality of feature vectors, leading to intensive computational costs. Actually, the choice of T is a trade-off between the value of F_1 and computational complexity.

TABLE V: Experiment results with different threshold T.

T	Dimensionality	$F_1(\%)$
1	18	90.95
2	50	93.29
3	98	96.49
4	162	96.65
5	242	96.71
6	338	97.03

To assess the influence of T, some experiments are conducted on the subset of **DB1** with $fps_2 = 30fps$. Table V reports the experiment results. As we expect, the value of F_1 also increases with the increment of T. Moreover, there is no dramatic rise of F_1 when T is increased from 3 to 6. Thus, T=3 is adopted in the following experiments.

C. Exposing single MC-FRUC operation

For our earlier works including Yao *et al.* [23] and Xia *et al.* [24], we follow the same parameters to obtain their

TABLE VI: The F_1 on tampered dataset DB1.(%)

MC-FRUC	Forensics methods	$fps_2(fps)$ 20	24	25	30	36	45	60	90	120
EBME	[23]	68.42	64.66	66.84	60.29	75.54	85.42	88.26	83.94	80.48
	[24]	62.48	69.77	67.61	69.49	57.67	58.97	61.64	58.43	55.89
	ours	**97.52**	**99.64**	**99.16**	**99.44**	**99.67**	**99.45**	**100.00**	**100.00**	**100.00**
AOBMC	[23]	65.78	60.22	61.13	60.17	76.23	82.87	87.55	80.94	79.66
	[24]	65.22	68.15	63.09	70.65	57.39	62.35	62.59	59.32	57.12
	ours	**99.87**	**99.91**	**100.00**	**99.91**	**100.00**	**100.00**	**100.00**	**100.00**	**99.95**
DualME	[23]	58.69	60.38	61.13	59.64	75.81	80.62	84.52	81.23	76.89
	[24]	62.53	70.48	68.88	71.38	66.75	62.54	64.41	60.02	58.28
	ours	**89.45**	**90.92**	**89.99**	**90.15**	**92.39**	**94.42**	**95.22**	**95.89**	**98.04**
MCMP	[23]	61.88	62.26	66.20	60.30	77.03	84.55	89.14	85.22	81.24
	[24]	61.44	63.33	62.23	68.48	52.69	57.06	60.07	56.56	54.74
	ours	**98.64**	**99.46**	**99.24**	**99.81**	**99.80**	**99.85**	**99.88**	**100.00**	**100.00**
DSME	[23]	53.60	59.13	58.44	59.89	75.27	82.97	85.84	80.68	78.39
	[24]	64.77	70.95	69.69	71.54	50.51	56.14	63.60	60.49	57.84
	ours	**99.33**	**99.46**	**99.92**	**99.86**	**99.93**	**99.90**	**100.00**	**100.00**	**100.00**
PFRUC	[23]	56.75	60.31	59.81	60.26	70.71	82.23	88.62	85.73	80.58
	[24]	61.27	66.59	63.50	68.93	55.68	58.79	60.75	57.57	54.97
	ours	**97.40**	**97.98**	**97.88**	**98.17**	**98.29**	**99.35**	**99.97**	**99.95**	**100.00**
MSU	[23]	-	-	-	62.14	-	84.98	88.20	80.79	78.01
	[24]	-	-	-	72.64	-	62.80	65.78	60.93	58.84
	ours	-	-	-	**95.55**	-	**100.00**	**96.90**	**100.00**	**98.12**
MVTools2	[23]	60.08	63.51	61.42	61.64	76.72	83.57	87.69	83.55	76.56
	[24]	66.36	70.79	69.47	74.24	54.28	65.85	66.39	60.98	56.73
	ours	**94.33**	**95.22**	**94.73**	**97.84**	**97.78**	**97.69**	**98.34**	**98.15**	**98.87**
YUVsoft	[23]	-	-	-	64.34	-	83.14	88.11	77.93	76.19
	[24]	-	-	-	73.15	-	63.20	66.33	55.22	52.25
	ours	-	-	-	**94.30**	-	**95.35**	**97.67**	**97.81**	**97.85**
Respeedr	[23]	63.22	61.25	63.81	62.74	74.67	82.30	94.05	87.48	83.72
	[24]	64.33	72.69	68.15	71.56	53.02	66.44	73.51	64.12	61.38
	ours	**92.78**	**93.64**	**93.66**	**93.80**	**94.93**	**95.90**	**96.88**	**96.90**	**96.92**
MotionPerfect	[23]	56.06	62.75	55.32	61.94	74.86	83.69	88.78	78.26	77.54
	[24]	67.81	75.83	72.22	74.87	53.53	64.05	66.38	58.34	55.43
	ours	**92.19**	**92.82**	**92.41**	**92.53**	**94.53**	**95.00**	**95.96**	**96.67**	**96.84**
Average	[23]	60.50	61.61	61.57	61.21	75.20	83.30	88.25	82.34	79.02
	[24]	64.02	69.84	67.20	71.54	55.72	61.65	64.68	59.27	56.68
	ours	**95.61**	**96.67**	**96.22**	**96.49**	**97.48**	**97.90**	**98.26**	**98.67**	**98.78**

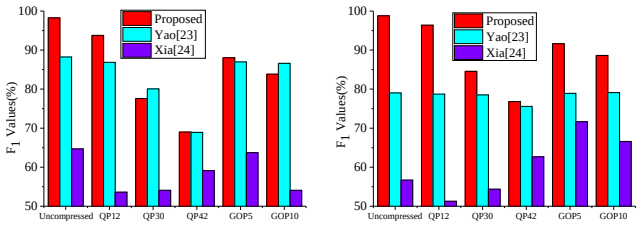
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2676162, IEEE Transactions on Circuits and Systems for Video Technology

experimental results. For the approach by P. Bestagini *et al.* [21], it can estimate original frame rate, but can not localize interpolated frames [37]. Thus, our approach is only compared with method [21] in terms of *SR*. To comprehensively evaluate the performance, several groups of experiments are conducted.

1) Localization accuracy of interpolated frame for uncompressed tampered videos: Comparisons are made among the proposed approach and existing methods [23], [24] for **DB1** tampered dataset. Table VI reports the average localization accuracies of interpolated frame, in which "-" means the test sequences are not constructed by MSU and YUVsoft. From it, the proposed pre-classifier achieves the best performance. Meanwhile, we also observe that with the increase of repeated times of MC-FRUC operations, the accuracies of the proposed pre-classifier increase steadily, whereas the accuracies of method [23] firstly increase and then decrease. For method [24], the accuracies get worse. The reasons behind this are summarized as follows. First, residual signals inside interpolated frames are accumulated, which are beneficial for the proposed pre-classifier. Second, when there are much more interpolated frames than original ones, the periodicity of edge intensity [23] and average texture variation [24] might be destroyed, leading to poor localization performance. Third, existing methods [23], [24] do not consider the possible rapid scene change and consecutive static scene, but this case is specially treated by our approach, which will be discussed in next subsection. Forth, the localization accuracy of method [24] depends on whether the estimated frame-rate completely matches the original one. However, when the target frame-rate is greater than $30fps$, there exists the fluctuation of estimated frame-rate around the original one. This gradually degrades the localization accuracies of interpolated frames.

2) Localization accuracy of interpolated frame against lossy compression with different configurations: Experimental results on the dataset **DB2** are reported in Fig. 8 and Table VII. The results of method [24] are not reported here because they are not as well as method [23]. With the increase of QP, GOP length or Q_scale, the proposed pre-classifier has a gradual degradation of detection accuracy due to the following reasons: (i) for compressed videos, motion and texture-rich regions become smoother, which inevitably has side effects on residual signals; (ii) From ME and MC points of view, there are some similarities among MC-FRUC, H.264/AVC and MPEG-2 compression. Thus, it is more difficult to discriminate interpolated frames from original frames when they are highly compressed. However, the F_1 values of method [23] have small fluctuation for the robustness of edge-detector. Furthermore, the estimated interpolated-frames are not always correct, which inevitably does harm to the estimation of original frame rate and further identification of various MC-FRUC approaches.

The proposed pre-classifier still achieves better performance for compressed videos with different configurations and different fps_2 in most cases. With the increment of fps_2, it achieves similar trend of location accuracy with uncompressed videos. For the tampered dataset encoded with configuration



(a) (b)

Fig. 8: The F_1 on subset of DB2 under $conf_2$ and $conf_4$. (a) with fps_2=60fps; (b)with fps_2=120fps.

TABLE VII: The F_1 on the subset of DB2 with ($fps_2 = 30fps$).(%)

MC-FRUC	Forensics methods	H.264							MPEG2				
		$conf_1$: QP=12				$conf_2$: GOP=8			$conf_4$: Q_scale=8		$conf_5$: GOP=10		
		GOP11	GOP17	GOP24	GOP30	QP12	QP30	QP42	GOP5	GOP10	Q_scale2	Q_scale4	Q_scale6
EBME	[23]	59.48	59.09	57.96	58.20	60.82	60.53	57.02	60.18	60.75	60.19	62.88	58.66
	ours	**90.64**	**90.51**	**90.21**	**90.09**	**90.76**	**65.14**	**57.08**	**84.81**	**77.03**	**87.85**	**85.74**	**84.35**
AOBMC	[23]	59.08	59.46	57.78	57.89	60.70	60.04	56.62	60.05	60.72	59.76	61.95	58.64
	ours	**91.37**	**91.25**	**90.75**	**90.11**	**91.50**	**68.81**	**61.03**	**89.59**	**83.48**	**89.30**	**87.44**	**86.84**
DualME	[23]	58.88	58.36	57.01	57.16	59.62	58.74	55.69	59.56	60.22	58.71	61.40	57.63
	ours	**93.31**	**92.84**	**91.84**	**91.07**	**93.69**	**75.23**	**62.92**	**78.84**	**72.98**	**80.27**	**75.50**	**73.12**
MCMP	[23]	58.66	58.12	57.03	57.20	61.01	61.16	**57.56**	59.87	60.56	59.89	61.70	58.64
	ours	**90.59**	**90.43**	**90.15**	**89.71**	**90.82**	**63.66**	55.60	**84.92**	**77.63**	**85.44**	**81.52**	**83.53**
DSME	[23]	58.34	57.88	56.67	56.93	60.03	59.21	56.57	59.81	60.51	59.04	62.80	58.85
	ours	**92.45**	**92.32**	**91.83**	**91.35**	**92.74**	**68.78**	**58.81**	**84.93**	**80.04**	**88.88**	**85.94**	**84.23**
PFRUC	[23]	59.70	60.25	59.15	59.16	60.93	60.84	57.55	60.07	60.75	59.75	62.35	58.41
	ours	**96.05**	**95.73**	**95.23**	**94.47**	**96.10**	**75.78**	**65.94**	**89.99**	**84.47**	**89.45**	**87.21**	**85.06**
MSU	[23]	62.75	62.59	63.44	61.75	62.27	61.09	57.22	62.14	62.95	64.11	63.57	61.95
	ours	**81.60**	**81.11**	**80.45**	**79.47**	**82.19**	**62.52**	**58.09**	**84.40**	**79.79**	**83.36**	**82.15**	**81.01**
MVTools2	[23]	62.69	62.31	61.91	61.22	62.10	60.73	57.56	62.16	62.82	63.94	63.55	62.35
	ours	**92.48**	**91.97**	**91.53**	**90.93**	**92.75**	**61.29**	**61.35**	**82.49**	**77.29**	**82.92**	**82.76**	**79.15**
YUVsoft	[23]	63.98	63.29	62.75	62.04	63.36	60.56	56.65	64.37	65.60	64.18	63.14	63.08
	ours	**89.17**	**88.81**	**87.37**	**86.93**	**89.49**	**66.34**	**62.03**	**83.23**	**76.78**	**81.26**	**79.41**	**78.78**
Respeedr	[23]	63.89	63.37	62.42	61.70	63.45	59.66	56.74	63.24	64.37	64.28	63.61	62.87
	ours	**88.97**	**88.78**	**87.88**	**86.97**	**89.55**	**62.90**	**58.59**	**83.48**	**78.45**	**83.11**	**82.07**	**80.50**
MotionPerfect	[23]	63.15	62.78	62.12	61.30	62.83	60.55	56.91	62.11	63.03	63.41	62.79	61.98
	ours	**87.47**	**86.94**	**85.95**	**84.56**	**87.83**	**62.69**	**57.58**	**87.40**	**86.23**	**90.62**	**89.83**	**88.18**
Average	[23]	60.96	60.68	59.84	59.50	61.56	60.28	56.92	61.23	62.03	61.57	62.70	60.28
	ours	**90.37**	**90.06**	**89.38**	**88.70**	**90.67**	**66.65**	**59.91**	**84.92**	**79.74**	**85.68**	**83.60**	**82.25**

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2676162, IEEE Transactions on Circuits and Systems for Video Technology

$conf_1$, when the GOP value is the same, the F_1 values are on average 25% higher than method [23]. In the case of $conf_2$, the difference is also significant when the QP value is relatively small. However, the difference dramatically reduces with the increase of QP value. In the cases of $conf_4$ or $conf_5$ (encoded with MPEG-2 codec), the F_1 values are on average 10.93% higher than method [23]. Thus, our proposed pre-classifier degrades faster than method [23] when the strength of lossy compression increases. Meanwhile, the performance increase of our proposed method is less than method [23] when $fps2$ increases from $30fps$ to $60fps$. Thus, there still exist some results of method [23] are better than ours in the case of fps_2=$60fps$ encoded by H.264 with QP30 or MPEG-2 with GOP10, which can be seen from Fig. 8(a).

In summary, no matter interpolated frames come from uncompressed videos or compressed videos with high perceptual quality, our proposed pre-classifier achieves more desirable location accuracy than existing works.

TABLE VIII: *SRs* on tampered CIF video Dataset(%).

MC-FRUC	Forensics method	Uncompressed	H.264: conf2 QP12	QP30	QP42	MPEG2: conf4 GOP5	GOP10
EBME	[21]	84.33	72.22	55.56	44.45	61.11	55.55
	[24]	93.33	80.66	76.66	66.66	78.67	75.11
	ours	**100.00**	**88.67**	-	-	-	-
AOBMC	[21]	86.66	55.55	61.11	44.45	50.00	38.89
	[24]	92.66	76.33	74.66	65.33	73.56	68.67
	ours	**100.00**	**84.33**	-	-	23.80	-
DualME	[21]	85.33	33.34	38.89	33.33	33.33	33.34
	[24]	87.55	85.66	80.66	73.33	82.67	79.67
	ours	**94.00**	**92.33**	-	-	-	-
MCMP	[21]	86.66	72.22	61.11	50.00	66.66	61.11
	[24]	95.00	88.45	81.23	71.66	83.33	78.45
	ours	**100.00**	**92.22**	-	-	-	-
DSME	[21]	88.66	83.33	66.67	44.44	66.67	55.56
	[24]	98.50	78.66	75.33	70.56	76.67	74.56
	ours	**100.00**	**85.33**	-	-	-	-
PFRUC	[21]	80.45	72.22	72.22	61.11	83.33	72.22
	[24]	94.66	89.50	80.33	75.33	85.33	78.54
	ours	**100.00**	**100.00**	-	-	16.67	-
MSU	[21]	90.54	44.44	50.00	38.89	50.00	38.89
	[24]	96.70	82.00	76.33	69.67	78.45	74.56
	ours	**100.00**	**84.33**	-	-	-	-
MVTools2	[21]	95.45	33.33	50.00	22.22	38.89	33.33
	[24]	93.30	83.33	80.45	75.56	81.33	77.67
	ours	**96.00**	**88.67**	-	-	-	-
YUVsoft	[21]	93.33	66.67	55.56	50.00	66.67	66.67
	[24]	97.50	83.33	78.56	68.56	79.67	79.67
	ours	**100.00**	**85.66**	-	-	-	-
Respeedr	[21]	94.54	38.54	33.33	16.67	41.67	33.33
	[24]	95.66	76.66	70.33	66.67	73.56	69.67
	ours	**96.54**	**84.82**	-	-	-	-
Motion Perfect	[21]	92.56	66.67	55.56	38.89	44.45	38.89
	[24]	88.50	82.33	78.66	65.33	79.67	77.45
	ours	**94.66**	**85.56**	-	-	-	-
Average	[21]	90.65	62.25	49.50	45.46	54.80	47.98
	[24]	93.94	82.81	**77.56**	**70.24**	**79.36**	**75.82**
	ours	**98.29**	**88.36**	-	-	3.68	-

3) SRs of estimating original frame rate: In this experiment, the subset of **DB2** with two configurations $conf_2$ and $conf_4$ are evaluated. Then, the average *SRs* of estimating original frame rate are compared among existing methods [21], [24] and the proposed pre-classifier. The average *SRs* are summarized in Table VIII. Apparently, our proposed pre-classifier outperforms existing methods [21], [24] for both uncompressed videos and compressed videos with QP12. Except for the above-mentioned cases, our proposed method fails to estimate the original frame-rate, which are marked with "-" in Tabel VIII. Further, we evaluate the performance for the dataset **DB3** and simultaneously calculate the structural similarity index (SSIM) between compressed videos and original videos. The results are shown in Fig.9. From it, when the QP value increases, the average *SR* of our proposed method decreases dramatically and became zero when QP=22. Meanwhile, we observe that SSIM also decreases from 0.9946 for QP12 to 0.9521 for QP22. This implies that lossy video compression degrades perceptual video quality and weakens the residual signals left by various MC-FRUC techniques as well, leading to the decrease of F_1 values and the decrease of *SRs*. Besides, the SSIMs for MPEG-2 with GOP5 and GOP10 are 0.9424 and 0.9395, respectively. It is reasonable to infer that their average *SRs* are also less than that of QP22 because the perceptual video quality in both cases are worse than QP22. Therefore, we conclude that from the degradation of perceptual video quality point of view, our proposed method fails to estimate the original frame-rate for H.264 with QP30 and QP42, and MPEG-2 with GOP5 and GOP10. Actually, the average *SRs* of our proposed pre-classifier seriously depends on the values of F_1, and their average F_1 values are below 94% in these cases. The reasons have been discussed in above subsection. Please note that though the proposed pre-classifier fails to estimate original frame rate in some cases, suspicious videos can still be decided as the up-converted ones since there are lots of correctly located interpolated-frames.

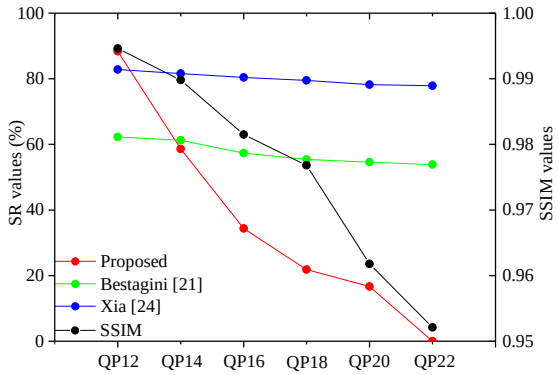


Fig. 9: *SRs* on DB3 and changes of SSIM.

Moreover, from Table VIII, the average *SRs* of method [24] are about 75% because it focuses on estimating original frame-rate, in which some pre-processing or post-processing such as high-pass filter and threshold decision are adopted to improve the SRs. From the results of method [24], the estimated frame-rates fluctuate about ± 1 or ± 2 frames around the original frame-rate, leading to worse localization accuracy. Method [21] achieves the worse performance, and the reasons are two-folds: (i) when the up-converted videos are highly compressed, the prediction errors become more serious resulting in incorrect periodical signal; (ii) when candidate videos are produced by multiple times of MC-FRUC operations, especially for fps_2=$60fps$ and 120 fps, there is aliasing artifact of the estimated frame rate.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2676162, IEEE Transactions on Circuits and Systems for Video Technology

4) Performance improvement by excluding static interpolated frames with SSD: The **Static** and **Fast** datasets are chosen for experiments. Before excluding static interpolated frames, SSD is firstly used to label static original frames in the original videos of **Static**. Unfortunately, though there are abundant static scene in **Static**, no frames satisfy the requirements of SSD. That is, no frames are labelled as static original frames. Then, SSD is employed to select static interpolated frames from the **Static** and **Fast** datasets, and the detection results are reported in Table IX. From it, we observe that there are some static interpolated frames in these datasets, even in the **Fast**. Besides, we also observe that commercial MC-FRUC software avoid the emergence of abundant static interpolated frames. Please note that DualME is an executable software. Maybe because it has post-processing operation, there are less static interpolated frames as well.

TABLE IX: Performance improvement (%) by SSD

MC-FRUC	**Static**		**Fast**	
	Numbers	Improvement	Numbers	Improvement
EBME	281	7.38	3	0.24
AOBMC	267	6.41	3	0.13
DualME	1	0.06	0	0.00
MCMP	53	3.65	5	0.32
DSME	11	1.14	2	0.23
PFRUC	74	5.1	42	1.79
MSU	0	0.00	0	0.00
MVTools2	0	0.00	0	0.00
YUVsoft	2	0.45	0	0.00
Respeedr	2	0.21	0	0.00
MotionPerfect	7	0.54	0	0.00

To evaluate the interference of static interpolated frames on the F_1 values, the proposed pre-classifier is tested on these datasets by using SSD or not, respectively. Then, we calculate the differences of theirs F_1 values. The experimental results are listed in Table IX. From it, the improvements vary from 0.06% to 7.38% for the **Static** dataset. Meanwhile, there is subtle improvement for the **Fast** dataset.

D. Identification of various MC-FRUC techniques

1) Identification accuracy of known MC-FRUC techniques: The subset of **DB1** with fps_2=30fps, 60fps, and 120fps and the subset of **DB2** with configurations including $conf_2$ for H.264/AVC, and $conf_4$ for MPEG2 are chosen for experiments. Meanwhile, "mixed" means the mixture of tampered datasets with fps_2=30fps, 60fps and 120fps. Since there are eleven known MC-FRUC operations involved, this experiment is a 12-class (including original videos as a special class) classification problem.

Table X reports the average identification accuracies, which are the elements of confusion matrix along diagonal direction. Apparently, the accuracies also increase with the increment of repeated times of FRUC operations. Especially, the accuracies are quite desirable for uncompressed videos and compressed videos with QP12. Meanwhile, we notice that the accuracies are relatively low for videos with slow motion or less texture. Actually, this is in accordance with our expectation. On one hand, MC-FRUC can achieve much better results for videos with slow motion or less texture. On the other hand, there are less residual signals in the tampered videos, and the differences of residual signals are very subtle among various MC-FRUC techniques. For highly compressed videos, the accuracies drop significantly. By experiments, we found the reason behind this is the incorrect classification of original frames as interpolated frames in the pre-classifier stage. This verifies the importance of a low false positive rate of the proposed pre-classifier, as discussed in Section IV-B. In addition, the loss of accuracy is also caused by strong compression, since it smoothes motion and texture-rich regions.

Table XI and XII show the accuracies for "mixed" tampered Dataset including uncompressed videos and compressed videos with QP12, respectively. From them, the proposed approach can effectively identify the adopted MC-FRUC technique for tampered videos either uncompressed or compressed with QP12. However, if two MC-FRUC techniques have only slight differences in motion search pattern or weighted average mechanism, it is extremely difficult to distinguish them by inspecting residual signal. This is a limitation for the proposed

TABLE X: The average identification accuracies (%) along the diagonal direction in the corresponding confusion matrices.

Status	Uncompressed				QP12				QP30				QP42				GOP5				GOP10			
fps_2	30fps	60fps	120fps	mixed	30fps	60fps	120fps	mixed	30fps	60fps	120fps	mixed	30fps	60fps	120fps	mixed	30fps	60fps	120fps	mixed	30fps	60fps	120fps	mixed
Accuracy	86.93	87.39	88.17	81.27	67.20	67.90	74.23	70.29	38.80	44.52	50.96	44.76	30.30	33.04	43.44	39.85	45.83	53.11	62.21	54.61	40.59	50.12	58.24	51.65

TABLE XI: Confusion matrix on "mixed" tampered Dataset under uncompressed status. The asterisks "$*$" denote that the corresponding values are below 1%

	Classified as											
MC-FRUC	Respeedr	YUVsoft	MotionPerfect	MVTools	MSU	dualME	DSME	AOBMC	EBME	PFRUC	MCMP	Pristine
Respeedr	77.25	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$
YUVsoft	$*$	67.95	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$
MotionPerfect	$*$	$*$	78.24	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$
MVTools	$*$	$*$	$*$	88.31	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$
MSU	$*$	$*$	$*$	$*$	60.76	$*$	$*$	$*$	$*$	$*$	$*$	$*$
dualME	$*$	$*$	$*$	$*$	$*$	91.74	$*$	$*$	$*$	$*$	$*$	$*$
DSME	$*$	$*$	$*$	$*$	$*$	$*$	95.06	$*$	$*$	$*$	$*$	$*$
AOBMC	$*$	$*$	$*$	$*$	$*$	$*$	$*$	74.37	$*$	$*$	$*$	$*$
EBME	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	83.11	$*$	$*$	$*$
PFRUC	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	96.06	2.32	$*$
MCMP	$*$	$*$	$*$	7.52	$*$	$*$	4.64	23.46	11.17	1.99	71.93	11.17
Pristine	$*$	33.64	22.35	5.75	40.61	9.84	1.88	3.76	$*$	3.54	27.33	90.42

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2676162, IEEE Transactions on Circuits and Systems for Video Technology

TABLE XII: Confusion matrix on "mixed" tampered Dataset under compressed QP12 status. The asterisks "$*$" denote that the corresponding values are below 1%

	Classified as												
MC-FRUC	Respeedr	YUVsoft	MotionPerfect	MVTools	MSU	dualME	DSME	AOBMC	EBME	PFRUC	MCMP	Compressed	Pristine
Respeedr	71.53	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$
YUVsoft	$*$	53.4	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$
MotionPerfect	$*$	$*$	65.67	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$
MVTools	$*$	$*$	$*$	74.63	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$
MSU	$*$	$*$	$*$	$*$	52.85	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$
dualME	$*$	$*$	$*$	$*$	$*$	91.54	$*$	$*$	$*$	$*$	$*$	$*$	$*$
DSME	$*$	$*$	$*$	$*$	$*$	$*$	66.45	$*$	$*$	$*$	$*$	$*$	$*$
AOBMC	$*$	$*$	$*$	$*$	$*$	$*$	$*$	59.15	$*$	$*$	$*$	$*$	$*$
EBME	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	63.79	$*$	$*$	$*$	$*$
PFRUC	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	76.18	$*$	$*$	$*$
MCMP	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	58.04	3.98	$*$
Compressed	5.97	42.23	30.18	22.22	46.43	7.62	30.07	37.70	27.08	37.70	37.70	83.25	4.2
Pristine	$*$	5.85	5.63	4.64	8.29	2.32	4.97	4.64	10.61	2.87	5.74	14.26	97.29

TABLE XIII: Test results for unknown MC-FRUC. The asterisks "$*$" denote that the corresponding values are below 1%

	Classified as											
unknown MC-FRUC	Respeedr	YUVsoft	MotionPerfect	MVTools	MSU	dualME	DSME	AOBMC	EBME	PFRUC	MCMP	Pristine
OBMC [13]	$*$	$*$	$*$	$*$	$*$	12.12	9.31	62.12	10.52	3.25	2.68	$*$
MHME [20]	18.45	25.72	21.06	26.36	7.16	$*$	$*$	$*$	$*$	$*$	$*$	1.25

approach.

2) Identification of unknown MC-FRUC techniques: To assess the generalization ability of the proposed forensics system, it is also used to identify the additional dataset **DB4**. Specifically, the forgery videos produced by OBMC [13] and MHME [20] are not used in the training stage, but they are used in the testing stage. That is, the classifier has not any information about two unknown MC-FRUC techniques. The subset of DB1 with fps_2=30fps is used for training. Table XIII reports the experimental results. For those tampered videos by MHME, the proposed system can effectively decide that they are forgery videos, even though they are identified as open MC-FRUC software produced. For candidate videos produced by OBMC, they are effectively detected as the up-converted ones, and their types are also assigned to the OBMC-based technique. Especially, most videos are identified as the type of AOBMC.

The proposed system can correctly detect the samples of unknown MC-FRUC technique as the forgery ones, but it still can not identify the unknown MC-FRUC method with totally different ME or MCI strategies as a new type. This is the so-called open-set recognition problem [38], in which the testing sample corresponds to a new class that is not included in the training stage. Recently, a few approaches [38], [39], [40] have been proposed based on the Extreme Value Theory (EVT). In this paper, the newest algorithm [40] is used to identify unknown MC-FRUC technique. We firstly take the ST-MSF features and their labels of train videos from the DB1 with fps_2=30fps as training samples, and the ST-MSF features and their labels of DB4 as testing samples. Next, the EVT is used to model the tail distributions of matched reconstruction errors and the sum of non-matched reconstruction errors so as to simplify the open set recognition problem into a two hypothesis testing problem. Then, the reconstruction errors of a test sample are calculated and the confidence scores based on the two tail distributions are fused to identify the type of test samples. The detection accuracies of OBMC and MHME are 63.59% and 76.44%, respectively.

E. Evaluation on HD720p tampered video dataset

The proposed forensics system is further evaluated on tampered dataset **DB5**. Table XIV shows the average accuracies of locating interpolated frames, and Table XV reports the average identification accuracies. From them, the performance is much better than that of tampered CIF Dataset, but they have similar variation trends. With the increase of repeated times of MC-FRUC operations, the localization and identification accuracies increase steadily. Actually, with the increase of spatial resolution, there are more residual signals inside interpolated frames, even when the tampered video are highly compressed.

TABLE XIV: The F_1on tampered HD720p video Dataset

MC-FRUC	Uncompressed				$conf_2$												$conf_4$							
					QP12				QP30				QP42				GOP5				GOP10			
	30fps	60fps	120fps	mixed	30fps	60fps	120fps	mixed	30fps	60fps	120fps	mixed	30fps	60fps	120fps	mixed	30fps	60fps	120fps	mixed	30fps	60fps	120fps	mixed
EBME	99.58	100.00	100.00	99.93	99.98	99.81	100.00	99.87	95.95	99.26	100.00	98.78	76.55	88.29	98.35	90.31	97.78	99.91	99.60	99.64	97.47	99.25	99.84	99.31
AOBMC	100.00	100.00	99.84	100.00	99.43	100.00	100.00	99.84	95.48	99.63	100.00	97.89	75.59	89.43	99.69	90.75	98.61	99.91	99.90	99.80	99.17	99.81	99.85	99.54
Dual_ME	87.07	97.66	100.00	95.39	92.22	98.59	100.00	96.37	95.45	99.72	100.00	98.27	92.51	99.45	100.00	97.47	85.46	93.35	99.84	94.62	83.54	91.07	99.60	93.40
MCMP	99.72	100.00	100.00	99.93	99.72	99.72	100.00	99.31	95.84	99.07	99.76	98.23	72.28	90.55	97.52	89.16	98.08	99.91	99.84	99.11	97.37	99.35	99.52	99.18
DS_ME	99.86	100.00	100.00	99.87	98.75	100.00	100.00	99.93	97.50	99.72	99.92	98.88	77.64	92.83	99.92	91.82	99.58	99.91	99.91	99.93	99.16	99.82	99.84	99.74
PFRUC	98.18	100.00	100.00	99.57	100.00	100.00	100.00	99.93	96.45	99.16	99.92	99.34	81.01	92.48	99.44	91.83	99.58	99.54	99.85	99.87	99.86	99.35	99.80	99.80
MSU	98.06	99.91	99.76	98.49	97.44	99.72	100.00	98.68	95.80	97.29	99.52	97.28	91.67	87.72	92.84	82.95	94.27	99.91	99.60	98.31	94.42	99.53	99.28	97.86
MVTools2	95.66	97.17	100.00	98.61	96.61	97.04	100.00	98.51	95.51	96.07	100.00	97.64	94.41	91.22	100.00	95.28	96.28	94.57	99.52	97.45	95.68	95.76	99.76	96.65
Respeedr	100.00	100.00	100.00	98.95	99.45	100.00	100.00	99.80	91.77	99.45	100.00	98.05	73.40	92.81	99.12	91.00	97.07	99.17	99.36	98.05	97.49	98.03	98.85	97.89
Motion Perfect	98.18	100.00	99.84	100.00	98.19	99.34	99.92	99.19	96.64	98.78	98.96	97.34	81.01	85.47	92.28	82.53	92.32	99.72	99.76	99.11	95.35	99.54	98.88	98.62
Average	97.64	99.46	99.94	99.08	98.18	99.42	99.99	99.14	95.64	98.82	99.81	98.17	81.61	91.03	97.92	90.31	95.90	98.59	99.72	98.59	95.95	98.15	99.52	98.20

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2676162, IEEE Transactions on Circuits and Systems for Video Technology

TABLE XV: The average identification accuracies (%) along the diagonal direction in the corresponding confusion matrices.

Status	Uncompressed				QP12				QP30				QP42				GOP5				GOP10			
fps_2	$30fps$	$60fps$	$120fps$	mixed	$30fps$	$60fps$	$120fps$	mixed	$30fps$	$60fps$	$120fps$	mixed	$30fps$	$60fps$	$120fps$	mixed	$30fps$	$60fps$	$120fps$	mixed	$30fps$	$60fps$	$120fps$	mixed
Accuracy	97.69	98.23	97.98	95.45	90.74	95.45	97.54	93.08	76.27	87.27	94.69	85.24	53.36	64.97	89.35	70.37	77.08	84.26	93.98	85.95	77.78	85.34	92.23	84.78

Moreover, we observe that the proposed pre-classifier achieves desirable results, in which most accuracies are above 95%. Thus, the proposed pre-classifier has higher *SRs* on the DB5 because when it meets $F_1 \geq 94\%$, the original frame-rate is successfully estimated, as discussed in Section V-A.

VI. Conclusions and Future Work

In this paper, interpolated frame generated by MC-FRUC is mathematically modeled as the sum of absent original frame and residual signal. The identification of various MC-FRUC techniques is thus converted into the problem of identifying the differences of residual signals among them. To this end, a two-stage blind forensics system is proposed for the identification of MC-FRUC techniques. Firstly, a pre-classifier is proposed to choose interpolated frames with prominent residual signals. Seondly, ST-MSFs are extracted as feature vectors for multi-class classification. Experimental results have shown that for up-converted videos both in uncompressed format and compressed with high perceptual quality, the proposed system can not only effectively locate interpolated frames, but also identify the adopted MC-FRUC technique. Meanwhile, it is more effective to detect interpolated frames obtained by repeated MC-FRUC operations. We believe the proposed approach is a positive step towards estimating key parameters in some specific MC-FRUC technique for deeper blind forensics. In future work, we will investigate the EVT and more discriminative classifier [41]-[43] to better address the open-set scenario. Furthermore, how to improve the identification robustness for highly compressed videos is worthy of investigation.

Acknowledgment

We would like to thank the anonymous reviewers for providing valuable comments and suggestions. We also appreciate Dr. He Zhang in Rutgers University, USA for permission to use their codes in our experiments. Prof. Gaobo Yang is the corresponding author.

References

[1] S. Milani, M. Fontani, P. Bestagini, M. Barni, A. Piva, M. Tagliasacchi, and S. Tubaro, "An overview on video forensics," *APSIPA Trans. Signal Inf. Process.*, vol. 1, pp. 1-18, 2012.

[2] A. Rocha, W. Scheirer, T.E. Boult, and S. Goldenstein, "Vision of the unseen: Current trends and challenges in digital image and video forensics," *ACM Comput. Surveys.*, vol. 43, pp. 1-42, 2011.

[3] J. Wang, T. Li, Y. Q. Shi, et al, "Forensics feature analysis in quaternion wavelet domain for distinguishing photographic images and computer graphics", *Multimedia Tools Appl.*, DOI: 10.1007/s11042-016-4153-0, 2016

[4] J. Li, X. Li, B. Yang, and X. Sun, "Segmentation-based image copy-move forgery detection scheme," *IEEE Trans. Inf. Forensics Security.*, vol. 10, no. 3, pp. 507-518, 2015.

[5] A. Gironi, M. Fontani, T. Bianchi, A. Piva, and M. Barni, "A video forensic technique for detecting frame deletion and insertion," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, pp. 6226-6230, 2014.

[6] M.C. Stamm, W. Lin, and K.J. Liu, "Temporal forensics and anti-forensics for motion compensated video," *IEEE Trans. Inf. Forensics Security.*, vol. 7, no. 4, pp. 1315-1329, 2012.

[7] J. Chao, X. Jiang, and T. Sun, "A novel video inter-frame forgery model detection scheme based on optical flow consistency" In: *Digital Forensics and Watermaking. Springer Berlin Heidelberg*, pp. 267-281, 2012.

[8] Y. Wu, X. Jiang, T. Sun, and W. Wang, "Exposing video inter-frame forgery based on velocity field consistency," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, pp. 2674-2678, 2014.

[9] Y. Liu, and T. Huang, "Exposing video inter-frame forgery by Zernike opponent chromaticity moments and coarseness analysis," *Multimedia Sys.*, pp. 1-16, 2015.

[10] Y. Dar, and A.M. Bruckstein, "Motion-Compensated Coding and Frame Rate Up-Conversion: Models and Analysis," *IEEE Trans. Image Process.*, vol. 24, no. 7, pp. 2051-2066, 2015.

[11] R. Feghali, F. Speranza, D. Wang, and A. Vincent, "Video quality metric for bit rate control via joint adjustment of quantization and frame rate," *IEEE Trans. Broadcast.*, vol. 53, no. 1, pp. 441-446, 2007.

[12] U.S. Kim, and M.H. Sunwoo, "New frame rate up-conversion algorithms with low computational complexity," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 3, pp. 384-393, 2014.

[13] M.T. Orchard, and C.J. Sullivan, "Overlapped block motion compensation: an estimation-theoretic approach," *IEEE Trans. Image Process.*, vol. 3, no. 9, pp. 693-699, 1994.

[14] S.J. Kang, K.R. Cho, and Y.H. Kim, "Motion compensated frame rate up-conversion using extended bilateral motion estimation," *IEEE Trans. Consum. Electron.*, vol. 53, no. 4, pp. 1759-1767, 2007.

[15] B.D. Choi, J.W. Han, C.S. Kim, and S.J. Ko, "Motion-compensated frame interpolation using bilateral motion estimation and adaptive overlapped block motion compensation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 4, pp. 407-416, 2007.

[16] S. J.Kang, S. J. Yoo, and Y. H.Kim, "Dual motion estimation for frame rate up-conversion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 12, pp. 1909-1914, 2010.

[17] R. Li, Z. Gan, Z. Cui, G. Tang, and X. Zhu, "Multi-channel mixed-pattern based frame rate up-conversion using spatio-temporal motion vector refinement and dual weighted overlapped block motion compensation," *J. Disp. Technol.*, vol. 10, no. 12, pp. 1010-1023, Dec. 2014.

[18] D.G. Yoo, S.J. Kang, and Y.H. Kim, "Direction-select motion estimation for motion-compensated frame rate up-conversion," *J. Display Technol.*, vol. 9, no. 10, pp. 840-850, 2013.

[19] T.H. Tsai, and H.Y. Lin, "High visual quality particle based frame rate up conversion with acceleration assisted motion trajectory calibration," *J. Display Technol.*, vol. 8, no. 6, pp. 341-351, 2012.

[20] H.B. Liu, R.Q. Xin, D.B. Zhao, S. W. Ma, and W. Gao, "Multiple hypotheses bayesian frame rate up-conversion by adaptive fusion of motion-compensated interpolations," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 8, pp. 1188-1198, 2012.

[21] P. Bestagini, S. Battaglia, S. Milani, M. Tagliasacchi, and S. Tubaro, "Detection of temporal interpolation in video sequences," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2013, pp. 3033-3037.

[22] S. Bian, W. Luo, and J. Huang, "Detecting video frame-rate upconversion based on periodic properties of inter-frame similarity," *Multimedia Tools Appl.*, vol. 72, no. 1, pp. 437-451, 2014.

[23] Y. Yao, G. Yang, X. Sun, and L. Li. "Detecting video frame-rate up-conversion based on periodic properties of edge-intensity," *Journal of Inf. Security and Appl.*, vol. 26, pp. 39-50, 2016.

[24] M. Xia, G. Yang, L. Li, R. Li, and X. Sun, "Detecting video frame rate up-conversion based on frame-level analysis of average texture variation," *Multimedia Tools Appl.*, pp. 1-23, 2016.

[25] T.G. Dietterich, and G. Bakiri, "Solving multiclass learning problems via error correcting output codes," *J. Artif. Intell. Res.*, vol. 2, pp. 263-286, 1995.

[26] J. Kodovsky, J. Fridrich, and V. Holub, "Ensemble classifiers for steganalysis of digital media," *IEEE Trans. Inf. Forensics Security.*, vol. 7, no. 2, pp. 432-444, 2012.

[27] W. Zheng, Y. Shishikui, M. Naemura, Y. Kanatsugu, and S. Itoh, "Analysis of space-dependent characteristics of motion-compensated frame differences based on a statistical motion distribution model," *IEEE Trans. Image Process.*, vol. 11, no. 4, pp. 377-386, 2002.

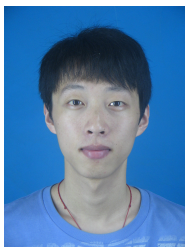
[28] J.Y. Wang, and E. H. Adelson, "Representing moving images with

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2676162, IEEE Transactions on Circuits and Systems for Video Technology

layers, *IEEE Trans. Pattern. Anal. Machine Intell.*, vol. 13, pp.625-638, 1994.
[29] T. Pevny, P. Bas, and J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," *IEEE Trans. Inf. Forensics Security.*, vol 5, no. 2, pp. 215-224, 2010.
[30] Z. He, W. Lu, W. Sun, and J. Huang, "Digital image splicing detection based on markov features in DCT and DWT domain," *Pattern Recogn.*, vol. 45, no. 12, pp. 4292-4299, 2012.
[31] Z. Xia, X. Wang, X. M. Sun, et al. "Steganalysis of least significant bit matching using multi-order differences," *Security and Comm. Netw.*, vol. 7, no. 8, pp. 1283-1291, 2014
[32] T.M. Cover and J. A. Thomas, Elements of Information Theory. New York: Wiley, 1991.
[33] S. Chen, S. Tan, B. Li, and J. Huang, "Automatic Detection of Object-based Forgery in Advanced Video," *IEEE Trans. Circuits Syst. Video Technol.*, 2015. DOI: 10.1109/TCSVT.2015.2473436.
[34] S.J. Kang, S.I. Cho, S. Yoo, and Y.H. Kim, "Scene change detection using multiple histograms for motion-compensated frame rate upconversion," *J. Display Technol.*, vol.8, no.3, pp.121-126, 2012.
[35] T. Hastie, and R. Tibshirani, "Classification by pairwise coupling", *Ann. Statist.*, vol. 26, no. 2, pp. 451-471, 1998.
[36] C.J.V. Rijsbergen, *Information Retrieval*. Newton, MA, USA: Butterworth-Heinemann, 1979.
[37] P. Bestagini, S. Milani, M. Tagliasacchi, and S. Tubaro, "Local tampering detection in video sequences," in *Proc. 15th IEEE Int. Workshop Multimedia Signal Process.*, 2013, pp. 488-493.
[38] W.J. Scheirer, A. Rocha, A. Sapkota, and T.E. Boult, "Toward open set recognition," *IEEE Trans. Pattern. Anal. Machine Intell.*, vol 35, no. 7, pp. 1757-1772, 2013.
[39] W.J. Scheirer, L.P. Jain, and T.E. Boult, "Probability models for open set recognition," *IEEE Trans. Pattern. Anal. Machine Intell.*, vol 36, no. 11, pp. 2317-2324, 2014.
[40] H. Zhang, and V. Patel, "Sparse Representation-based Open Set Recognition," *IEEE Trans. Pattern. Anal. Machine Intell.*, 2016.
[41] B. Gu, V.S. Sheng, K.Y. Tay, W. Romano, and S. Li, "Incremental support vector learning for ordinal regression," *IEEE Trans. Neural Netw. Learn. Syst.*, vol 26, no. 7, pp. 1403-1416, 2015.
[42] B. Gu, V.S. Sheng, Z. Wang, D. Ho, S. Osman, and S. Li, "Incremental learning for v-support vector regression," *Neural Netw.*, vol 67, pp. 140-150, 2015.
[43] B. Gu, X. Sun, and V.S. Sheng, "Structural Minimax Probability Machine," *IEEE Trans. Neural Netw. Learn. Syst.*, DOI: 10.1109/TNNLS.2016.2544779, 2016.

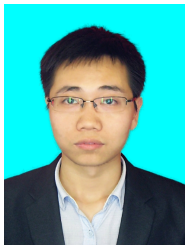
Xiangling Ding received the B.S. and M.S. degrees from Hunan Normal University in 2003 and 2006, respectively. He is pursuing the Ph.D. degree in Hunan University, China. He is also an assistant professor with Huaihua University, Huaihua, China. His current research interests include digital media forensics, and image processing.

Gaobo Yang is a professor in Hunan University, China. He is also a key member of Hunan Provincial Key Laboratory of Networks and Information Security. He received the Ph.D. degree in Communication and Information System from Shanghai University in 2004. He is the PI of several projects such as Natural Science Foundation of China (NSFC), Special Prophase Project on National Basic Research Program of China (973) and program for New Century Excellent Talents (NCET) in university. Currently, his research interests are in the area of digital media forensics, image and video signal processing.



Ran Li received Ph.D. degrees from the School of Communication and Information Engineering, Nanjing University of Posts and Telecommunications,Nanjing, China, in2014. He currently works as assistant professor with the School of Computer and Information Technology, Xinyang Normal University, Xinyang, China. His current research interests include frame-rate up-conversion, compressed sensing.

Lebing Zhang received the B.S. and M.S. degrees from Hunan Normal University in 2003 and 2009, respectively. He is pursuing the Ph.D. degree in Hunan University, China. His current research interests include image information security, and image processing.



Yue Li received the B.S. and M.S. degrees from Hengyang Normal University and Central South University in 2010 and 2013, respectively. He is pursuing the Ph.D. degree in Hunan University, China. His current research interests include video coding, image processing.

Xingming Sun is currently a professor with the School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, China. He received the B.S. degree in mathematics from Hunan Normal University, Hunan, China, in 1984, the M.S. degree in computing science from the Dalian University of Science and Technology, Dalian, China, in 1988, and the Ph.D. degree in computer science from Fudan University, Shanghai, China, in 2001. His research interests include network and information security, digital watermarking, cloud computing security, and wireless network security.