Blind Detection of Median Filtering in Digital Images: A Difference Domain Based Approach

Chenglong Chen, *Student Member, IEEE*, Jiangqun Ni, *Member, IEEE*, and Jiwu Huang, *Senior Member, IEEE*

***Abstract*— Recently, the median filtering (MF) detector as a forensic tool for the recovery of images' processing history has attracted wide interest. This paper presents a novel method for the blind detection of MF in digital images. Following some strongly indicative analyses in the difference domain of images, we introduce two new feature sets that allow us to distinguish a median-filtered image from an untouched image or average-filtered one. The effectiveness of the proposed features is verified with evidence from exhaustive experiments on a large composite image database. Compared with prior arts, the proposed method achieves significant performance improvement in the case of low resolution and strong JPEG post-compression. In addition, it is demonstrated that our method is more robust against additive noise than other existing MF detectors. With analyses and extensive experimental researches presented in this paper, we hope that the proposed method will add a new tool to the arsenal of forensic analysts.**

***Index Terms*— Digital image forensics, median filter, statistics in difference domain, tampering detection.**

I. Introduction

With significant advances made in the development of digital photography, digital images have found wide applications in news media, military, and law enforcement. Meanwhile, due to the high popularity and widespread availability of digital image editing software, we can no longer take the authenticity of digital images for granted. In recent years, many image forgery detection techniques have been proposed, especially passive or blind forensic methods which do not require any additional information besides the image itself with undetermined authenticity. These methods work on the assumption that although digital forgeries may leave no visual clues that indicate tampering, they do alter, to some extent, the underlying statistics of an image [1]. By extracting features that capture such artifacts, tampering can be blindly distinguished from authentic data. In order to find out what kind of manipulation the image under investigation has undergone, the possible use of a wide variety of operations must be tested for. Existing image forensic works involve the detection of median filtering (MF) [2]–[5], resampling [6], [7], JPEG compression [8], amongst others.

This work concentrates on the blind detection of median filtering, which is commonly used in image denoising and smoothing. As a great variety of existing forensic methods, like the resampling detection scheme [6], rely on some kind of linearity assumption, blind detection of non-linear median filtering becomes especially important. Due to its non-linearity and complex statistical properties, counter-forensic techniques, the art and science of impeding or misleading forensic analyses of digital images, show special interest in this operation [9], [10]. As a result, the median filtering detector becomes an important forensic tool for the recovery of the processing history of an image.

In this paper, we provide a new approach in the difference domain for reliable MF detection in digital images. Based on the properties observed in the difference domain of images, two new feature sets in difference domain are constructed for MF detection [5]. The effectiveness of the proposed scheme is extensively evaluated with a composite image database of 9,000 images. Compared with existing detectors targeting the same problem, the proposed method yields better or at least comparable performance and is particularly more reliable in the case of low resolution and strong JPEG post-compression. Moreover, the proposed method is shown to be more robust against noise than other existing MF detectors.

The rest of this paper is organized as follows. In Section II, we briefly review existing works on median filtering detection. Two new feature sets in the difference domain are introduced for the detection of median filtering in Section III and IV, respectively. Section V details our experimental methodology, which is followed by the experimental results and analyses, including a comparison with previous arts in Section VI. The last section summarizes the work that has been done in this paper.

II. Background and Related Work

A. Median Filter

The median filter is a well-known non-linear filter based on order statistics. Given an $H \times W$ grayscale image $X_{n,m}$ with $(n, m) \in \{1, 2, \ldots, H\} \times \{1, 2, \ldots, W\}$, a 2-D median filter is defined as

$$\hat{X}_{n,m} = \text{median}\{X_{n',m'} : (n', m') \in \mathcal{W}(n, m)\} \quad (1)$$

where $\hat{X}_{n,m}$ is the output of the median filter at image coordinates (n, m), median$\{\cdot\}$ is the median operator, and $\mathcal{W}(n, m)$ is the 2D filter window centered at image coordinates

Manuscript received October 1, 2012; revised March 10, 2013 and June 7, 2013; accepted July 22, 2013. Date of publication August 8, 2013; date of current version September 27, 2013. This work was supported in part by the National Natural Science Foundation of China under Grants 61379156, 1135001 and 60970145, in part by the 973 Program under Grant 2011CB302204, in part by the National Research Foundation for the Doctoral Program of Higher Education of China under Grant 20120171110037, in part by the Key Program of Natural Science Foundation of Guangdong under Grant S2012020011114, and in part by the National Science & Technology Pillar Program under Grant 2012BAK16B06. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Chun-Shien Li.

The authors are with the School of Information Science and Technology, Sun Yat-sen University, Guangdong 510006, China (e-mail: c.chenglong@gmail.com; issjqni@mail.sysu.edu.cn; isshjw@mail.sysu.edu.cn).

Digital Object Identifier 10.1109/TIP.2013.2277814

(n, m). For the rest of our paper, we focus on filter with $w \times w$ ($w = 2r + 1$, $r = 1, 2, \ldots$) square windows as it is the most widely used form of median filter.

B. Detection of Median Filtering

Due to the non-linearity of the median filter, a theoretical analysis of the general relation between input and output distributions always seems highly non-trivial [2]. As a result, most prior methods on MF detection usually rely on some specific characteristics of median filter. In the following, we will briefly review some state-of-the-art MF detectors.

It is well-known that the median filter tends to produce regions of constant or nearly constant intensities. This effect, referred to as the *streaking artifacts* [11], was first exploited to detect median-filtered images by Kirchner and Fridrich [2]. Based on the first-order difference image, they presented a measure of the streaking artifacts. Denote h_0 and h_1 as the numbers of the first-order difference equal to 0 and 1, respectively. Streaking tends to increase particularly h_0 relative to h_1. For this reason, they suggested to use the ratio $\rho = h_0/h_1$ as a detection feature, where $\rho \gg 1$ indicates median filtering. This feature can be computed in four directions, i.e., horizontal, vertical, major diagonal, and minor diagonal directions, with the corresponding first-order difference calculated in the same direction. The experimental results indicated that this single feature works well in uncompressed images, whereas it is no longer reliable when the median-filtered images are JPEG post-compressed. To overcome that, the authors of [2] further employed the *subtractive pixel adjacency matrix* (SPAM) features [12], which are constructed as the sample transition probability matrix of a higher-order Markov model of the first-order difference image. It is shown that the second-order SPAM features with a threshold $T_s = 3$ (leading to 686-D features), can be used to detect median filtering in high to medium quality JPEG compressed images.

In another work, Cao *et al.* [3] measured the streaking artifacts via the probability of zero values on the first-order difference image in textured regions. The probabilities are calculated in the horizontal and vertical directions, and finally fused to form a scalar feature. The experimental results verified the effectiveness of this feature to detect median filtering in uncompressed images. However, it also suffers performance loss after JPEG compression as the single feature ρ in [2].

The 2D median filter affects either the order or the quantity of the gray levels in an image region. By exploring this fact, Yuan [4] presented the *median filtering forensics* (MFF) feature, as a combination of five feature subsets based on order statistics and the gray levels, to capture the local dependence artifact introduced by median filtering. The experimental results indicated that MFF-based approach can achieve comparable or better performance than SPAM-based method in the case of moderate JPEG post-compression and low resolution. However, as with the SPAM-based detector, the performance of MFF-based detector degrades as the JPEG quality factor decreases or the size of the analyzed image shrinks. Thus, there is a need to develop more reliable schemes to detect median filtering in the case of strong JPEG compression and low resolution, which is more desirable in practical applications.

III. Global Probability Feature Set

In this section, we present our first set of features, i.e., global probability feature set (GPF), which is based on the empirical cumulative distribution function (CDF) for the k-th ($k = 1, 2, \ldots$) order difference of the median-filtered image. The motivation of such features is backed with extensive experimental studies from three different image databases, i.e., BOWS2 [13], NRCS [14], and Dresden Image Database (DID) [15].[1] Such empirical evidence indicates that CDF in the difference domain varies considerably between different image sources, e.g., original, median-filtered, and average-filtered images, indicating an effective fingerprint for MF detection.

A. The CDF in Difference Domain

It is noted that neighboring pixels in median-filtered image are correlated to some extent because they originate from overlapping windows of the original image. With the inherent nature of the median filter, it is expected that pixels in median-filtered image are correlated to their neighbors in a specific way. In this work, we study the effect of median filtering in terms of the k-th ($k = 1, 2, \ldots$) order difference. For a 2D signal $X_{n,m}$, we define it as

$$\Delta_k^{(p,q)}(n, m) = \Delta_{k-1}^{(p,q)}(n, m) - \Delta_{k-1}^{(p,q)}(n + p, m + q) \quad (2)$$

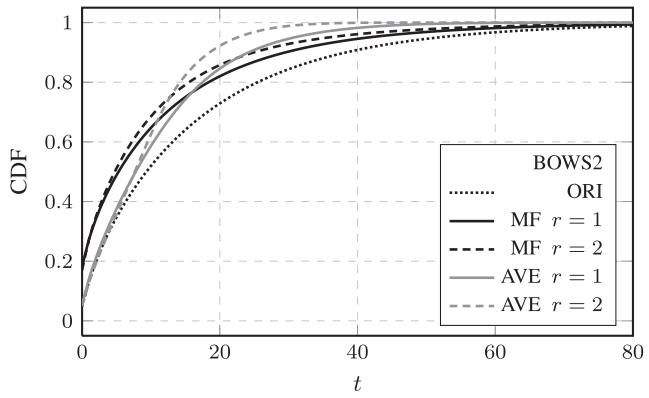
where $(p, q) \in \{-1, 0, 1\}^2$ with $|p| + |q| \neq 0$, and $|\cdot|$ is the operation to calculate the absolute value. Specially, we define $\Delta_0^{(p,q)}(n, m) = X_{n,m}$. Under the assumption that spatial statistics in natural images are symmetric with respect to mirroring and flipping [12], the CDF of $|\mathbf{\Delta}_k^{(p,q)}|$, i.e.,

$$F_k^{(p,q)}(r, t) = \Pr\left(|\Delta_k^{(p,q)}(n, m)| \leq t\right) \quad (3)$$

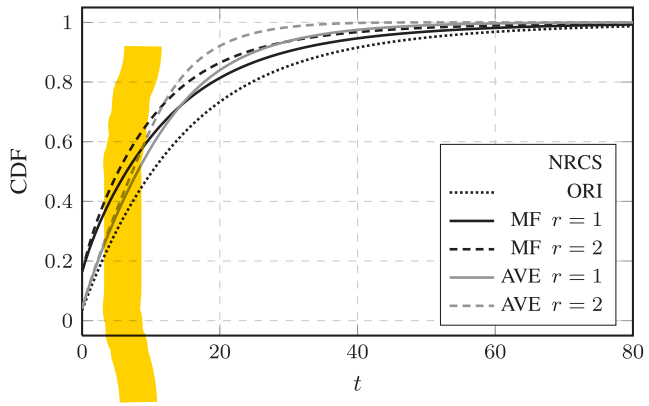
is of greater interest than that of $\mathbf{\Delta}_k^{(p,q)}$. In the above definition, r relates to the 2D $w \times w$ filter window as $w = 2r + 1$ (for original signal, we take $r = 0$) and t is an integer.

To show how different image sources vary in the difference domain, we report in Fig. 1 the empirical CDFs of $|\mathbf{\Delta}_1^{(0,1)}|$ estimated using images from three different image databases mentioned above. Each figure is obtained using $3,000$ 512×512 images from the corresponding database. Where necessary, images are converted to 8-bit grayscale. Specifically, all the empirical CDFs in Fig. 1 are estimated with 2D $w \times w$ ($w = 2r + 1$) windows (for filtered images) and by excluding untextured or smooth pixels satisfying $\sigma_u < \tau$, where σ_u is the standard deviation of local surrounding pixels in a square region of size $w_u \times w_u$ and τ is the selected threshold. These untextured pixels are excluded in a similar manner as prior work [3], since the definite statistical fingerprint of median filtering or average filtering is not necessarily assured in very smooth regions [3]. As we can see, effects for the CDFs of median-filtered and average-filtered images are very similar among different image databases. This indicates that such effects are due to the filtering operation instead of the image contents.

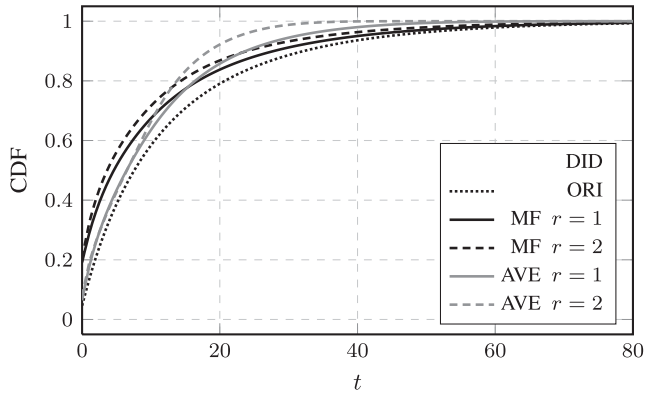
[1] A description of these image databases is given in Section V-A.



(a)



(b)



(c)

Fig. 1. CDFs of the first-order difference for different image sources, i.e., original (ORI), median-filtered (MF), and average-filtered (AVE) images. From (a) to (c): empirical CDFs of $|\boldsymbol{\Delta}_1^{(0,1)}|$ for each 3,000 512×512 images from BOWS2 [13], NRCS [14], and Dresden Image Database (DID) [15], respectively. All the empirical CDFs are estimated with 2D $w \times w$ ($w = 2r+1$) windows (for filtered images) and by excluding untextured pixels with $w_u = 7$ and $\tau = 15$.

B. Indications

In the following, we will highlight some indications, which follow directly from our experimental findings above.

1) Filtered Images vs. Original Images: The curves of CDF in Fig. 1 illustrate the principal effect of the considered filters. It is observed that, for median-filtered or average-filtered images, $|\boldsymbol{\Delta}_1^{(0,1)}|$ tends to take small values. Note that smaller and larger values in the first-order difference correspond to the low and high frequency components in spatial domain, respectively. In other words, some of the high frequency components in filtered images are removed, which is indeed the case when considering the low-pass property of such filters. Although the median filter, in a strict sense, is not a low-pass one, it is observed to exhibit low-pass effect to some extent [16].

2) Median-Filtered Images vs. Average-Filtered Images: Figure 1 also suggests a clear distinction between the CDFs of median-filtered and average-filtered images in difference domain. For median-filtered images and small t, e.g., $t \leq 10$, CDF is larger than that of original and average-filtered images. This is related to the inherent effect of the median filter known as *streaking artifacts* [11]. Streaking means that the median filter tends to produce regions of constant or nearly constant intensities, which leads to the larger probability in first-order difference for small t.

It is also observed in Fig. 1 that the curves of median-filtered images rise slowly whereas the curves of average-filtered images rise rapidly with increasing t. This indicates that, compared to average-filtered images, median-filtered images retain more high frequency components, such as image edges. This effect is also related to another inherent nature of median filter known as *good edge preservation* [17]. It is noted that the median filter preserves edges better than the average filter, owing to its statistical and robustness properties [16].

C. Construction of Feature Set

In general, we observe that the distribution of the k-th order difference array $\boldsymbol{\Delta}_k^{(p,q)}$ (or $|\boldsymbol{\Delta}_k^{(p,q)}|$) varies considerably between different image sources, indicating an effective fingerprint for MF detection. We herein construct our first feature set for MF detection based on the empirical CDF of $|\boldsymbol{\Delta}_k^{(p,q)}|$. By gathering the estimated probabilities satisfying $0 \leq t \leq T$ ($T \in \mathbb{Z}$ is a user-defined parameter), we obtain a vector $\boldsymbol{P}_k^{(p,q)}$ of length $T+1$ for each $|\boldsymbol{\Delta}_k^{(p,q)}|$. To reduce the dimensionality, we adopt the assumption in [12] that spatial statistics in natural images are symmetric with respect to mirroring and flipping. Thus, we separately average feature vectors with lags $|p| + |q| = i$ ($i = 1, 2$), i.e.,

$$\boldsymbol{P}_k^i = \frac{1}{4} \sum_{|p|+|q|=i} \boldsymbol{P}_k^{(p,q)} \tag{4}$$

to form a feature vector

$$\boldsymbol{P}_k = \left[\boldsymbol{P}_k^1, \boldsymbol{P}_k^2 \right]. \tag{5}$$

Finally, concatenating all the $\boldsymbol{P}_k$ ($k = 1, 2, \ldots, K$) leads to a $2(T+1) \times K$-D *global probability feature set* (GPF) to detect median filtering in digital images.

IV. Local Correlation Feature Set

In this section, we present our second set of features, i.e., local correlation feature set (LCF), which is based on the correlation between adjacent difference pairs in the difference domain. To be more specific, we first present the motivation of these correlation features by discussing the behavior of

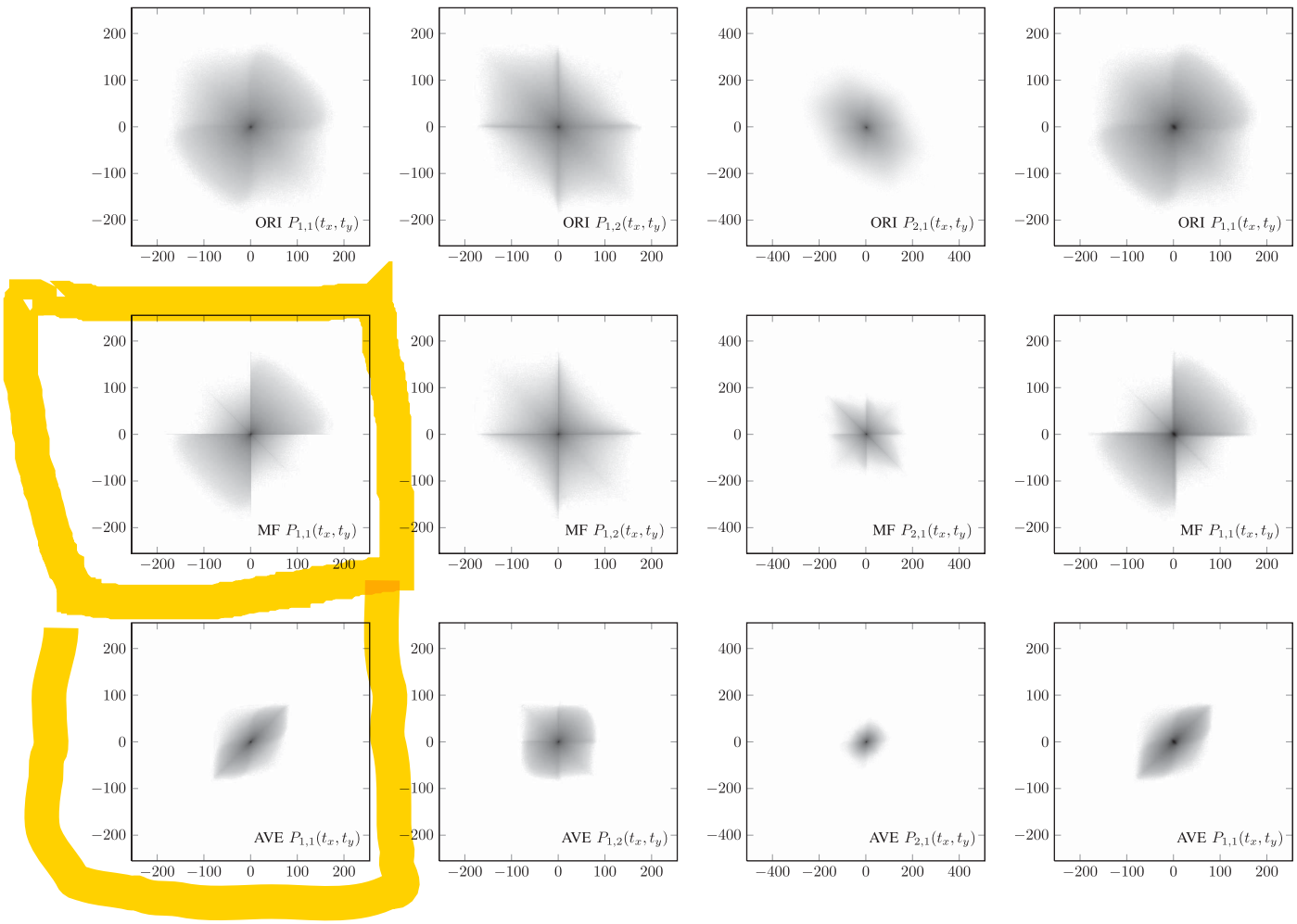


Fig. 2. Distribution of different adjacent difference pairs for different image sources using 3,000 512 × 512 images from the BOWS2 database. From top to bottom: original images (ORI), median-filtered images (MF), and average-filtered images (AVE). From left to right: $P_{1,1}(t_x,t_y)$, $P_{1,2}(t_x,t_y)$, and $P_{2,1}(t_x,t_y)$ for images without noise contamination, and $P_{1,1}(t_x,t_y)$ after adding zero mean Gaussian noise to each image, with resulting signal-to-noise ratio SNR = 40 dB. The median-filtered images and average-filtered images are obtained with 2D 3 × 3 square windows.

adjacent difference pairs. After that, we construct LCF features using a compact metric that can capture the property of the joint distribution of adjacent difference pair. Finally, we combine GPF and LCF to obtain the final global and local feature set (GLF) for the blind detection of median filtering in digital images.

A. The Behavior of Adjacent Difference Pairs

Besides the CDF, we also investigate the statistical characteristic in the difference domain using joint probability. Such a metric is suitable to elaborate the behavior of adjacent difference pairs. For the sake of brevity, we only take the joint probability of $\mathbf{\Delta}_k^{(0,1)}$, denoted as

$$P_{k,l}(t_x,t_y)=\Pr\left(\Delta_k^{(0,1)}(n,m)=t_x,\Delta_k^{(0,1)}(n,m+l)=t_y\right)$$

for example, where k is the difference order and $l\in\{1,2,\ldots\}$. The analysis of $\mathbf{\Delta}_k^{(p,q)}$ with other lags can be carried out in a similar manner. Figure 2 shows the empirical joint probability for different image sources, which are estimated using the same images from the BOWS2 database as above with 2D 3×3 windows (for filtered images). In these figures, the horizontal and vertical axes represent t_x and t_y, respectively. And the intensity values at (t_x,t_y) represent the probabilities at the logarithmic scale, whereas darker points refer to larger probabilities.

It is noted that both median filter and average filter employ sliding windows across the whole image. This introduces local correlations to the filtered image with respect to the position where the window is centered. This also holds for the difference domain of filtered images. As shown in Fig. 2, $P_{1,l}(t_x,t_y)$ varies considerably between different image sources for specific adjacent pairs. Moreover, Fig. 2 illustrates that not only the distribution of adjacent difference pairs differs between filtered and original images, but also between median-filtered and average-filtered images. For instance, adjacent difference pairs (t_x,t_y) of $P_{1,1}(t_x,t_y)$ tend to cluster in the first and third quadrants with high probability, which becomes more evident for median-filtered images. Since difference pairs with large value $(|t_x|,|t_y|)$ of $P_{1,1}(t_x,t_y)$ in the second and fourth quadrants are mainly related to impulse noise, the distribution $P_{1,1}(t_x,t_y)$ of median-filtered images in the second and fourth quadrants can be explained by the effectiveness of median filter to remove such noise [16].

While median filter smooths out impulse noise from images, it maintains the integrity of edges as discussed in Section III-B. Thus, the distribution $P_{1,1}(t_x, t_y)$ in the first and third quadrants is not heavily degraded from that of original images. To understand this, one can consider the difference of a 1D slope edge signal. It is most likely that t_x shares the same sign with t_y, and thus adjacent difference pairs (t_x, t_y) lie in the first or third quadrant. Due to the good edge preservation property of median filter, the positions of these adjacent pairs slightly deviate from the original ones after median filtering. As for average filter, it not only removes noise, but also tends to smooth edges as indicated in Fig. 1, resulting in a distribution which is more compact than other image sources.

A closer inspection of the distribution $P_{1,1}(t_x, t_y)$ of median-filtered images reveals that high probabilities occur along the t_x and t_y axes, which is different from original and average-filtered images. This is not surprising as median filtering tends to increase particularly the probability of zero values in the first-order difference, i.e., the streaking artifact, as shown in Fig. 1. Therefore, adjacent difference pairs (t_x, t_y) of $P_{1,1}(t_x, t_y)$ are likely to cluster along t_x axis (i.e., $t_y = 0$) and t_y axis (i.e., $t_x = 0$). Similar effects are also observed for other distribution $P_{1,l}(t_x, t_y)$ of median-filtered images.

The distribution $P_{k,l}(t_x, t_y)$ also varies considerably between different image sources for high-order differences, e.g., $P_{2,1}(t_x, t_y)$ as shown in the third column of Fig. 2. Moreover, an interesting and important observation here is that, the distribution "pattern" still remains even in the additive noise environment (c.f. the last column of Fig. 2) and also varies considerably between different image sources. Thus, such distribution "pattern" can be exploited to construct more robust features as will be discussed in the following subsection.

B. Construction of Feature Set

In the following, we will construct the second set of features for MF detection based on the normalized cross correlation (NCC). Consider two general random variables $\boldsymbol{x}$ and $\boldsymbol{y}$. The NCC coefficient $\gamma(\boldsymbol{x}, \boldsymbol{y})$ of $\boldsymbol{x}$ and $\boldsymbol{y}$ is defined as

$$\gamma(\boldsymbol{x}, \boldsymbol{y}) = \frac{\mathrm{cov}(\boldsymbol{x}, \boldsymbol{y})}{\sqrt{\mathrm{cov}(\boldsymbol{x}, \boldsymbol{x})\mathrm{cov}(\boldsymbol{y}, \boldsymbol{y})}} \tag{6}$$

where $\mathrm{cov}(\boldsymbol{x}, \boldsymbol{y}) = E[(\boldsymbol{x} - E[\boldsymbol{x}])(\boldsymbol{y} - E[\boldsymbol{y}])]$ is the covariance and $E[\cdot]$ is the expectation operator. It is obvious that $\gamma(\boldsymbol{x}, \boldsymbol{y})$ relies on the joint probability of $\boldsymbol{x}$ and $\boldsymbol{y}$. Therefore, we believe that the NCC coefficient can be adopted as a compact feature to capture the property of the distribution "pattern" (i.e., joint probability of adjacent difference pair) as shown in Fig. 2.

To construct the correlation features on the 2D difference array $\boldsymbol{\Delta}_k^{(p,q)}$, we first scan (*overlapping* scan) it with a 1D window of width B along the same direction as it is computed, and rearrange all B pixels of the i-th window into the i-th *column* of a matrix $\boldsymbol{Y}_k^{(p,q)}$. Note that $\boldsymbol{Y}_k^{(p,q)}$ is a matrix of B *rows*. The number of columns in $\boldsymbol{Y}_k^{(p,q)}$ depends on the size of considered image, scanning window length B, and the scanning method, i.e., overlapping or not. After eliminating the columns with all zero values in $\boldsymbol{Y}_k^{(p,q)}$, we combine matrices $\boldsymbol{Y}_k^{(p,q)}$ with lags $|p| + |q| = 1$ to obtain

$$\boldsymbol{Y}_k^1 = \left[\boldsymbol{Y}_k^{(0,1)}, \boldsymbol{Y}_k^{(0,-1)}, \boldsymbol{Y}_k^{(1,0)}, \boldsymbol{Y}_k^{(-1,0)}\right]^T. \tag{7}$$

Due to the transpose operation $(\bullet^T)$ above, matrix $\boldsymbol{Y}_k^1$ has B *columns* in total.

We then treat each column of $\boldsymbol{Y}_k^1$ as a variable which is related to the position in the filter window, and each row as a sample. Denoting $\gamma_k^1(i, j)$ (i and $j \in \mathbb{Z}$) as the NCC coefficient of the i-th and j-th columns of $\boldsymbol{Y}_k^1$, we combine all the coefficients $\gamma_k^1(i, j)$ $(i > j)$ to yield a $(B^2 - B)/2$-D vector $\boldsymbol{C}_k^1$. Another vector $\boldsymbol{C}_k^2$ is obtained from $\boldsymbol{Y}_k^{(p,q)}$ with lags $|p| + |q| = 2$ in a similar manner, resulting in a feature vector $\boldsymbol{C}_k$, i.e.,

$$\boldsymbol{C}_k = \left[\boldsymbol{C}_k^1, \boldsymbol{C}_k^2\right] \tag{8}$$

for the k-th order difference arrays. Finally, concatenating all the $\boldsymbol{C}_k$ $(k = 1, 2, \ldots, K)$ together leads to a $(B^2 - B) \times K$-D *local correlation feature set* (LCF) for the detection of median filtering.

C. Global and Local Feature Set (GLF)

Combining GPF and LCF features, we obtain the final *global and local feature set* (GLF) with $K[2(T+1)+(B^2-B)]$ elements for MF detection. While GPF relates to the estimated CDF in the difference domain (first-order statistics), LCF captures the correlation of adjacent difference pairs (second-order statistics). Thus, these two feature sets well capture the statistical artifacts introduced by median filter and other filters in a complementary way. As for practical implementation, the parameters of our feature sets, i.e., $\{T, B, K\}$, should be properly determined to trade off good detection capability and manageable computational complexity. A discussion about the parameter settings is presented in Section VI-B.

V. Experimental Methodology

This section describes the experimental methodology to verify the effectiveness of the proposed new features in the difference domain. Such a test setup is similar to that of Yuan [4].

A. Image Database

It is a well-known fact that the accuracy of forensics may vary significantly across different cover sources. If the adversary does not know any prior knowledge about the cover source, her best strategy is to train the forensic analyzer on as diverse image database as possible [12]. For this reason, our experiments make use of the following three widely used image databases, specifically:

- 3,000 images randomly selected from the BOWS2 database. This database was used for the BOWS2 contest [13], and contains downsampled and cropped natural grayscale images of fixed size 512×512.
- 3,000 images randomly selected from the NRCS Photo Gallery [14]. This database is provided by the Department of Agriculture, United States, and includes scanned images from a variety of film and paper sources.

- 3,000 images randomly selected from the Dresden Image Database (DID) [15]. This database is a collection of more than 14,000 images from 73 different digital cameras. All the used images are direct camera JPEG outputs which are captured with various camera settings.

Moreover, images were converted to 8-bit grayscale and only the 512×512 central parts of the images were used. This results in a composite database D^{ORI} of 9,000 images.

B. Training–Testing Pairs

Based on the above image database D^{ORI}, we prepared 9 training–testing pairs as follows:

1) Process all images in the original database D^{ORI} to obtain 5 image sets, i.e., D^{MF3}, D^{MF5}, D^{AVE}, D^{GAU} and D^{RES}, which are generated with a 3×3 and a 5×5 median filter, an average filter with the filter window randomly and uniformly set to 3×3 or 5×5, a Gaussian low-pass filter of size 3×3 with the standard deviation σ randomly and uniformly set to 0.5 or 0.8, and a rescale operation that is randomly and uniformly composed of nearest or bilinear interpolation and scaling factors 1.1 or 1.2, respectively. These randomized parameter settings resemble the scenarios in practical applications to some extent. Note that the images in D^{RES} were further cropped to the original size, i.e., 512×512.
2) Separate the above 5 image sets into 8 training–testing pairs, i.e., training sets $\{D^{\mathrm{MF}}(I), D^{\mathrm{ONE}}(I)\}$ and testing sets $\{D^{\mathrm{MF}}(\bar{I}), D^{\mathrm{ONE}}(\bar{I})\}$, where MF $\in \{\mathrm{MF3}, \mathrm{MF5}\}$, ONE $\in \{\mathrm{ORI}, \mathrm{AVE}, \mathrm{GAU}, \mathrm{RES}\}$, I is a randomly selected subset of the image indexes and $\bar{I}$ is its complement.
3) Randomly select 50% from each of the median-filtered image sets (D^{MF3} and D^{MF5}) to obtain D^{MF35}, and 25% from each of the 4 non-median-filtered image sets to obtain D^{ALL}. Partition the image sets D^{MF35} and D^{ALL} into a training sets $\{D^{\mathrm{MF35}}(I), D^{\mathrm{ALL}}(I)\}$ and a testing sets $\{D^{\mathrm{MF35}}(\bar{I}), D^{\mathrm{ALL}}(\bar{I})\}$. This training–testing pair is considered since it is likely that an investigator will need to distinguish between median filtering and a pool of other operations rather than a specific type of manipulation in practical settings [4].

For all the constructed training–testing pairs described above, the size of the training set is set to 40% of the database size. Based on these 512×512 training–testing pairs, we further obtained training–testing pairs of different image resolutions by cropping the central part of the corresponding image size from each image. Without loss of generality, we only consider images of size $M \times M$, e.g., 256×256 and 128×128.

C. Performance Evaluation

For each training–testing pair, all detectors under investigation are implemented as binary classifiers using C-SVM as previous works [2], [4]. To be specific, we use LIBSVM package [18] with Gaussian and linear kernels. The inputs of the classifiers are the selected features under study, e.g., GLF, MFF, and SPAM, computed from all images in the respective training set. The best hyper-parameters, i.e., (C_0, γ_0) for Gaussian kernel and C_0 for linear kernel, are optimized over the parameter grid $(C, \gamma) \in \{(2^c, 2^g) \mid c, g \in \mathbb{Z}\}$ using five-fold cross-validation for each training set and for each selected features. All images in the corresponding testing set are then classified using the classifier trained on the training set after computing the selected features.

VI. Experimental Study

In this section, experimental study is carried out to demonstrate the efficacy of the proposed method from different aspects. Specifically, we first perform linear discriminant analysis (LDA) on the proposed features. After discussing about parameter settings and SVM kernel, we compare our method to prior arts using both uncompressed and JPEG post-compressed images. Furthermore, we study the sensitivity of existing MF detectors against noise. Finally, we apply the proposed scheme to detect image forgery involving median filtering.

A. Efficacy of the Proposed Features

Herein, some preliminary simulation results are presented to demonstrate that the proposed features are capable to capture traces caused by median filtering. To this end, we use each 200 512×512 images from D^{ORI}, D^{MF3}, and D^{AVE}, respectively. After feature extraction with $\{T, B, K\} = \{10, 3, 2\}$ (leading to 44 GPF features, 12 LCF features, and 56 GLF features) and linear discriminant analysis (LDA), high-dimensional features are projected to 2D space (Fig. 3), in which they are clearly clustered according to different image sources, thereby demonstrating the discrimination capability of our features.

We have to mention that during this simulation as well as the experiments hereafter, *non-overlapping* scan is used instead in the construction of LCF features for image size $M > 64$, since the performance gain for more samples due to *overlapping* scan is negligible while *non-overlapping* scan helps to calculate the features more efficiently.

B. Parameter Settings and SVM Kernel

Before comparing it to prior works, we first discuss the parameters of the proposed method. To simplify the analysis, we set $T = 10$ and only consider the combined GLF features with varying K and B. We also compare the accuracy of classifiers implemented by SVMs with Gaussian and linear kernels. We only consider the results for JPEG post-compressed MF35 vs. ALL training–testing pairs with a quality factor (QF) of 90, which we deem particularly insightful. Note that JPEG compressed images are chosen because the proposed features are very reliable to detect median filtering in uncompressed images (c.f. Table III) and the corresponding performance varies slightly for different parameter settings.

Table I summarizes our findings for varying image sizes and different $\{K, B\}$ in terms of the minimum average decision error under the assumption of equal priors and equal costs, i.e.,

$$P_e = \min\left(\frac{P_{\mathrm{FP}} + P_{\mathrm{FN}}}{2}\right) \tag{9}$$

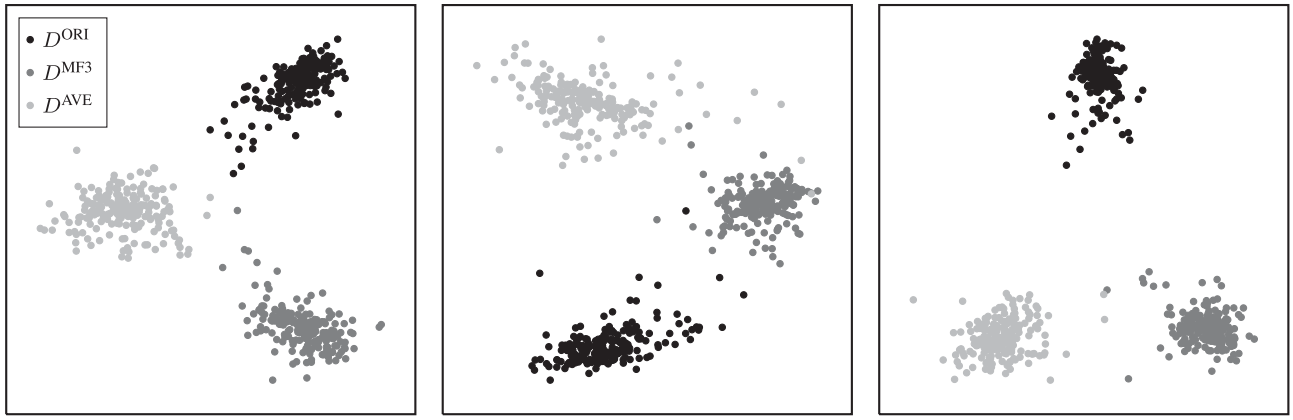


Fig. 3. 2D projection results from the proposed features: (a) GPF, (b) LCF, and (c) GLF by linear discriminant analysis (LDA). Markers with same color denote one specific image source, i.e., D^{ORI}, D^{MF3}, and D^{AVE}, respectively. Discrimination capability is shown via the clustering effects.

TABLE I
EFFECTS OF PARAMETER SETTINGS AND SVM KERNEL IN TERMS OF P_e (%) USING THE PROPOSED GLF FEATURES. LEFT: GAUSSIAN KERNEL. RIGHT: LINEAR KERNEL. "DIM." FOR FEATURE DIMENSIONALITY. THE BEST RESULT FOR EACH PARAMETER SETTING AND EACH IMAGE SIZE IS HIGHLIGHTED IN GRAY

K	B	Dim.	M				M			
			256	128	64	32	256	128	64	32
1	3	28	6.4	9.4	11.4	18.4	17.6	21.5	25.0	29.0
	5	42	5.5	7.7	9.2	16.0	17.5	20.9	24.0	28.1
	7	64	5.6	7.6	8.5	16.0	17.1	20.5	23.2	28.0
2	3	56	4.3	6.4	8.9	15.4	8.2	12.7	15.9	20.9
	5	84	4.1	5.6	7.9	14.5	7.9	10.7	14.9	19.9
	7	128	4.1	5.9	7.1	14.2	7.0	10.3	13.7	19.6
3	3	84	3.5	5.2	7.3	13.9	6.7	10.1	13.5	19.7
	5	126	3.6	5.0	7.0	13.1	6.2	8.6	12.6	18.3
	7	192	3.6	5.1	6.9	13.7	5.6	8.3	11.9	18.3

TABLE II
TIME IN HH:MM:SS TO PERFORM GRID-SEARCH FOR TRAINING OF SVM CLASSIFIERS WITH GAUSSIAN KERNEL USING THE PROPOSED GLF FEATURES. "DIM." FOR FEATURE DIMENSIONALITY. THE LEAST TIME FOR EACH IMAGE SIZE IS HIGHLIGHTED IN GRAY

K	B	Dim.	$M = 256$	$M = 128$	$M = 64$	$M = 32$	Average
1	3	28	0:46:06	0:59:28	1:14:19	1:54:15	1:13:32
	5	42	0:44:05	0:53:09	1:06:51	1:44:33	1:07:09
	7	64	0:46:22	0:47:38	1:06:27	1:05:47	0:56:27
2	3	56	0:39:18	0:49:37	1:01:10	0:57:04	0:51:47
	5	84	0:41:39	0:47:14	0:53:33	1:10:16	0:53:17
	7	128	0:57:38	1:03:59	1:12:36	1:35:53	1:12:32
3	3	84	0:41:57	0:49:57	1:02:38	1:06:36	0:55:17
	5	126	0:52:41	1:02:24	1:10:24	1:31:31	1:09:15
	7	192	1:08:02	1:17:30	1:32:43	2:13:42	1:32:59

where P_{FP} and P_{FN} denote the false positive and false negative rates, respectively. As shown in Table I, classifiers implemented as Gaussian SVMs are always better than their linear counterparts. This indicates that the decision boundaries between features of median-filtered images and those of non-median-filtered images are nonlinear. For this reason, we hereafter only consider Gaussian SVMs, which are also adopted in previous works [2], [4]. It is also observed that, for the same B, there is a considerable performance improvement from $K = 1$ to $K = 2$, and the performance with $K = 3$ is slightly better than the one with $K = 2$, and becomes saturated when $K \geq 3$. Note that large K results in high-dimensional features (c.f. "Dim." in Table I, feature dimensionality is in proportion to K as $K[2(T + 1) + (B^2 - B)]$).

We also investigate the effects of parameter settings on computational complexity. For practical implementation, the time required to train the SVM classifiers and in particular the time to perform grid-search for the best hyper-parameters, should be taken into account. Table II shows the actual times[2] of grid-search for Gaussian SVMs as well as the average time for each parameter setting (the last column). It is noticed that $\{T, B, K\} = \{10, 3, 2\}$ or $\{10, 5, 2\}$ is usually a reasonable choice for SVM training.

Taking both Table I and II into consideration, we suggest $\{T, B, K\}$ in practical implementation to be $\{10, 3, 2\}$ or $\{10, 5, 2\}$, for good detection capability, satisfied computational complexity, and moderate feature dimensionality. In the rest of this paper, if not specified, $\{T, B, K\} = \{10, 3, 2\}$ is adopted for the proposed features, i.e., GPF, LCF, and GLF.

C. Baseline Performance in Uncompressed Images

To evaluate the performance of detecting MF in uncompressed images, we take the MFF-based scheme as benchmark, which has low dimension (44-D) and gives good enough detection performance [4]. Table III summarizes the classification performance for varying image sizes in terms of P_e. It is observed that our GLF-based method performs comparable to the MFF-based method, and achieves nearly perfect classification performance even for image size as low as 16×16. In general, GPF features perform better than LCF features to a certain extent, which becomes more evident for small images. GLF features are generally superior to both GPF and LCF.

[2]The evaluation was carried out on a Dell computer (Intel(R) Core TM i3-2130 CPU @ 3.40GHz, 4GB memory, and the 64-bit version of Windows 7).

TABLE III

CLASSIFICATION RESULTS IN TERMS OF P_e (%) FOR UNCOMPRESSED IMAGES OF VARYING SIZES. THE PROPOSED FEATURES (GPF, LCF, AND GLF) AND MFF FEATURES FROM [4] ARE TESTED. THE BEST RESULT FOR EACH IMAGE SIZE AND EACH TRAINING–TESTING PAIR IS HIGHLIGHTED IN GRAY

M	Features	MF3				MF5				MF35
		ORI	AVE	GAU	RES	ORI	AVE	GAU	RES	ALL
512	GPF	0.0	0.2	0.1	0.1	0.0	0.3	0.1	0.1	0.3
	LCF	3.2	1.0	1.4	2.8	1.9	1.1	0.8	1.5	3.0
	GLF	0.0	0.2	0.1	0.1	0.0	0.2	0.1	0.1	0.2
	MFF	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.1
32	GPF	0.9	2.0	1.5	2.3	1.0	2.9	1.4	1.9	2.6
	LCF	9.3	6.6	7.5	12.0	4.9	7.0	3.8	6.4	11.1
	GLF	0.9	1.6	1.0	2.0	0.9	2.0	0.8	1.4	2.2
	MFF	0.3	1.0	0.5	0.2	0.4	2.2	0.9	0.5	1.3
16	GPF	1.9	4.1	3.6	5.4	1.7	4.7	2.8	3.3	5.1
	LCF	13.4	11.1	11.4	15.2	6.9	11.5	5.7	8.9	15.4
	GLF	1.7	2.9	2.0	3.5	1.2	3.3	1.2	2.3	3.8
	MFF	1.2	3.5	2.7	1.4	1.8	8.2	4.6	2.3	5.0

TABLE IV

CLASSIFICATION RESULTS IN TERMS OF P_e (%) FOR 512×512 JPEG POST-COMPRESSED IMAGES OF VARYING QFS BASED ON THE PROPOSED FEATURES, i.e., GPF, LCF, AND GLF. THE BEST RESULT FOR EACH QF AND EACH TRAINING–TESTING PAIR IS HIGHLIGHTED IN GRAY

QF	Features	MF3				MF5				MF35
		ORI	AVE	GAU	RES	ORI	AVE	GAU	RES	ALL
90	GPF	7.6	3.7	8.9	8.2	3.5	3.7	4.0	3.2	8.3
	LCF	4.2	1.6	3.5	4.0	1.7	1.2	1.2	1.8	4.5
	GLF	2.3	1.1	2.2	2.5	0.6	0.7	0.5	0.7	2.9
80	GPF	12.4	6.3	17.8	15.8	4.2	5.3	6.3	4.6	14.6
	LCF	5.3	2.5	6.7	5.4	2.2	1.6	1.5	1.8	6.4
	GLF	3.6	2.2	4.0	3.9	1.0	1.2	0.9	1.1	4.3
70	GPF	13.6	8.1	21.8	18.3	4.4	7.1	7.8	5.2	16.2
	LCF	7.1	3.3	9.9	6.1	2.5	2.2	2.3	1.9	7.5
	GLF	5.0	2.6	5.7	5.5	1.3	1.4	1.3	1.5	5.5

D. The Performance Under JPEG Post-Compression

Since JPEG is the widely used format for image storage and transmission, it is important to evaluate the performance of the proposed MF detection scheme in the case of JPEG post-compression. To this end, the JPEG compressed versions of the training–testing pairs described in Section V-B are adopted, with varying QFs, i.e., QF $\in \{90, 80, 70, 60, 50\}$. Both the MFF-based scheme [4] and SPAM-based [2] scheme using second-order SPAM features with a threshold $T_s = 3$ are taken as the benchmarks.

We first compare the performance obtained using GPF, LCF, and GLF features under JPEG post-compression. As reported in Table IV where the classification is carried out on 512×512 JPEG post-compressed images of varying QFs, the performance of these features degrades significantly with decreasing QF. Excluding untextured pixels might increase the performance of GPF features since the definite statistical fingerprint of median filtering is not necessarily assured in very smooth regions. However, this also poses a threat in the case of low image resolution, as there may not be enough pixels left to calculate GPF features robustly. This applies to LCF features as well, although LCF features seem generally more robust against JPEG post-compression than GPF features. However, GLF features give the best performance again. For this reason, we hereafter only report the results based on GLF features if not specified.

Now we compare our GLF-based scheme to the state-of-the-art methods, i.e., MFF-based and SPAM-based methods, in the case of JPEG post-compression. For the sake of brevity, we only report the results obtained on 512×512 JPEG post-compressed MF vs. ONE training–testing pairs with a typical QF of 70 (see Fig. 4), and report the results on JPEG post-compressed MF35 vs. ALL training–testing pairs with varying QFs and varying image sizes in detail (see Table V). As shown in Fig. 4, the classification performance for 5×5 median filtering is generally better compared with 3×3 median filtering for all the involved classifiers. While all three schemes show satisfactory detection performance (note the scaling of the axes), it is observed that GLF considerably outperforms MFF and SPAM for 3×3 filter windows among all the MF vs. ONE training–testing pairs. For windows of size 5×5, GLF and SPAM yield comparable results (and are superior to the MFF features), whereas SPAM works in a feature space of considerably higher dimension (686-D for $T_s = 3$, compared to 56-D for our scheme). Moreover, the ROC curves of our scheme for both 3×3 and 5×5 median filtering detection are relatively closer to each other, which indicates a more consistent classification performance of our scheme.

As for the classification on MF35 vs. ALL training–testing pairs (see Table V), the detection performance of the MFF-based and SPAM-based schemes degrades rapidly as the JPEG quality factor decreases or the size of the analyzed image shrinks. On the contrary, our scheme is more robust, indicating that the proposed scheme is more reliable for MF detection in low-quality images in terms of low resolution and strong JPEG compression. For practical forensic analysis, e.g., when the JPEG quality factor is unknown, instead of training a classifier for each specific QF, an alternative strategy is to train a single classifier for a variety of QFs. To this end, we randomly select $1,800$ images from each of the 5 JPEG compressed versions of D^{MF35} as positive samples, and similarly obtain negative samples from D^{ALL}. The results shown in Table V (denoted as #) indicate that our method also outperforms prior arts in this practical setup.

Once an image has been identified as "median-filtered," it is possible to further estimate the size of the applied median filter using the proposed features. We have found that GLF features are effective to distinguish among images median filtered by different windows, e.g., 3×3 and 5×5. To demonstrate this, we perform classification on MF3 vs. MF5 training–testing pairs with varying QFs and varying image sizes, which are constructed in a similar manner as in Section V-B. Results reported in Table VI indicate that GLF-based method generally gives higher accuracy than MFF-based and SPAM-based

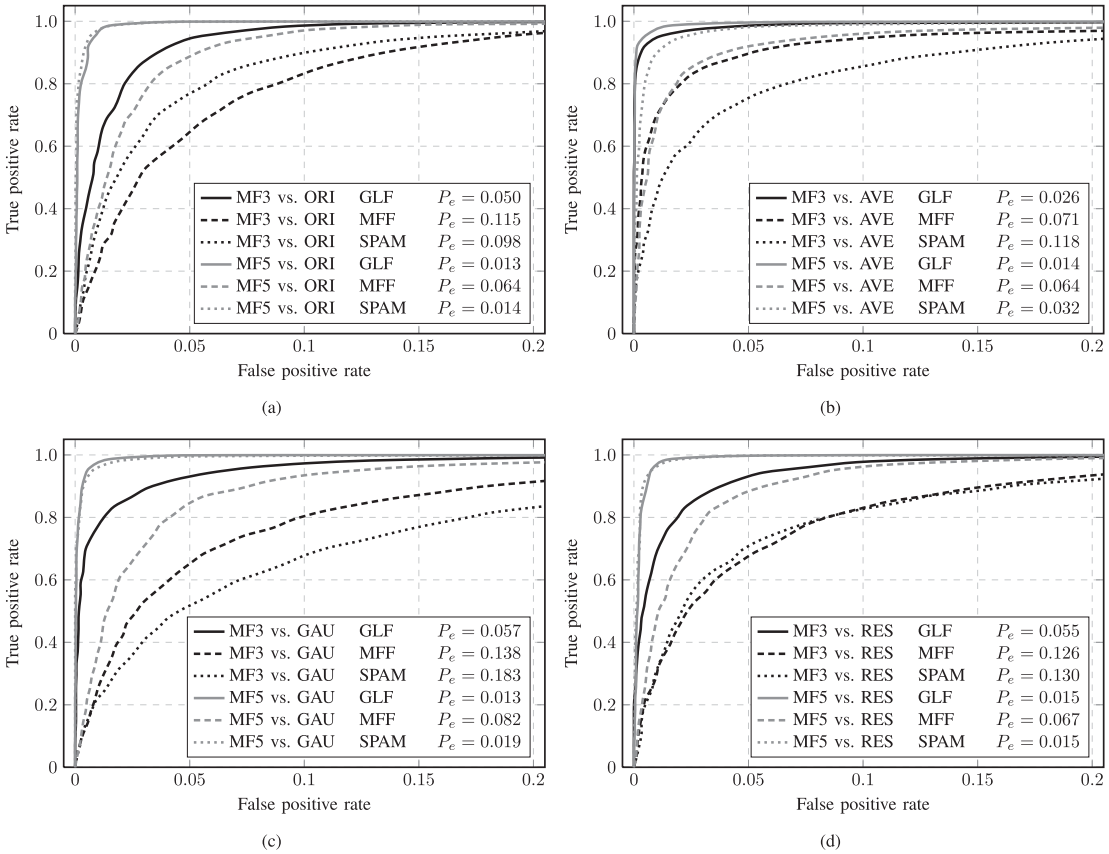


Fig. 4. (a)–(d) Classification results for each 512 × 512 JPEG post-compressed MF vs. ONE training–testing pair with a typical QF of 70. The proposed GLF features, MFF features from [4], and SPAM features from [2] are tested.

TABLE V

CLASSIFICATION RESULTS IN TERMS OF P_e (%) FOR JPEG POST-COMPRESSED MF35 VS. ALL TRAINING–TESTING PAIRS WITH VARYING QFS AND VARYING IMAGE SIZES. THE RESULTS FOR A SINGLE CLASSIFIER TRAINED FOR ALL QFS ARE DENOTED AS #. THE PROPOSED GLF FEATURES, MFF FEATURES FROM [4], AND SPAM FEATURES FROM [2] ARE TESTED. THE BEST RESULT FOR EACH QF AND EACH IMAGE SIZE IS HIGHLIGHTED IN GRAY

M	Features	QF = 90	QF = 80	QF = 70	QF = 60	QF = 50	#
512	GLF	2.9	4.3	5.5	6.8	7.9	6.0
	MFF	5.8	8.8	11.9	14.9	17.2	13.2
	SPAM	3.1	9.8	14.5	18.4	20.6	14.6
256	GLF	4.3	6.0	7.8	9.7	11.3	8.4
	MFF	7.5	13.3	17.2	21.4	24.4	17.8
	SPAM	7.2	16.3	20.7	24.8	26.2	20.4
128	GLF	6.4	9.9	11.9	15.2	16.9	12.8
	MFF	12.8	20.1	24.3	27.8	30.7	24.5
	SPAM	14.8	23.6	28.0	30.9	32.9	27.5
64	GLF	8.9	12.8	16.0	19.0	20.5	15.7
	MFF	20.5	27.4	31.9	33.5	34.4	30.3
	SPAM	23.4	30.3	33.8	36.1	37.2	33.1

TABLE VI

CLASSIFICATION RESULTS IN TERMS OF P_e (%) FOR JPEG POST-COMPRESSED MF3 VS. MF5 TRAINING–TESTING PAIRS WITH VARYING QFS AND VARYING IMAGE SIZES. THE RESULTS FOR A SINGLE CLASSIFIER TRAINED FOR ALL QFS ARE DENOTED AS #. THE PROPOSED GLF FEATURES, MFF FEATURES FROM [4], AND SPAM FEATURES FROM [2] ARE TESTED. THE BEST RESULT FOR EACH QF AND EACH IMAGE SIZE IS HIGHLIGHTED IN GRAY

M	Features	QF = 90	QF = 80	QF = 70	QF = 60	QF = 50	#
512	GLF	1.6	2.1	2.9	3.4	3.7	3.5
	MFF	18.2	20.3	21.3	21.6	21.8	22.2
	SPAM	1.0	2.6	4.2	5.5	7.1	5.2
256	GLF	3.1	3.7	4.3	5.6	6.2	5.3
	MFF	18.9	21.4	21.1	21.4	22.0	22.5
	SPAM	2.8	6.0	8.4	11.4	12.8	9.6
128	GLF	4.7	6.3	7.7	9.2	10.1	7.7
	MFF	22.1	23.1	23.2	23.9	23.8	25.0
	SPAM	7.3	11.9	15.4	19.1	20.4	16.5
64	GLF	7.1	9.3	11.2	12.9	14.6	11.8
	MFF	24.7	25.1	26.1	26.3	26.3	28.0
	SPAM	15.1	19.6	22.6	26.5	27.3	22.8

methods, especially in the case of low resolution and strong JPEG post-compression, when differentiating between 3 × 3 and 5 × 5 median filtering.

E. The Performance Under Additive Noise

Median filter is usually employed to remove noise (especially salt and pepper noise) in images. More recently, as we have mentioned in Section I, counter-forensic techniques show special interest in this operation [9], [10]. A typical example is the counter-forensic deblocking operation reported in [10], which attempts to hide JPEG compression history by removing the statistical traces of blocking artifacts with a median filter followed by additive noise. As mentioned in [19], it is uncertain if existing MF detectors still work in

TABLE VII

CLASSIFICATION RESULTS IN TERMS OF P_e (%) FOR IMAGES INTERFERED WITH GAUSSIAN NOISE. THE RESULTS FOR NOISE FREE IMAGES ARE DENOTED AS #. THE PROPOSED GPF AND LCF FEATURES, MFF FEATURES FROM [4], AND SPAM FEATURES FROM [2] ARE TESTED. THE BEST RESULT FOR EACH SNR AND EACH TRAINING–TESTING PAIR IS HIGHLIGHTED IN GRAY

SNR (dB)	Features	MF3				MF5				MF35
		ORI	AVE	GAU	RES	ORI	AVE	GAU	RES	ALL
#	GPF	0.0	0.2	0.1	0.1	0.0	0.3	0.1	0.1	0.3
	LCF	3.2	1.0	1.4	2.8	1.9	1.1	0.8	1.5	3.0
	MFF	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.1
	SPAM	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
50	GPF	15.6	21.4	22.0	20.0	21.4	25.9	10.7	22.3	25.3
	LCF	3.7	2.5	3.6	3.9	2.0	1.5	1.6	2.0	4.2
	MFF	5.9	18.2	10.4	5.0	6.7	8.3	5.2	4.9	8.6
	SPAM	12.2	12.7	14.1	10.2	12.4	6.2	12.8	12.6	13.5
40	GPF	49.0	49.3	49.6	40.3	49.8	41.3	37.7	43.9	49.8
	LCF	8.2	12.1	15.5	12.2	4.0	4.3	6.4	5.4	14.0
	MFF	23.8	43.5	36.1	28.9	17.2	31.5	25.6	19.7	28.0
	SPAM	49.4	48.6	49.6	48.4	49.4	47.1	49.5	49.4	46.2
30	GPF	50.0	45.0	50.0	44.8	50.0	49.4	49.9	44.4	49.9
	LCF	26.7	28.6	43.1	28.3	17.5	23.0	24.4	20.4	41.2
	MFF	36.7	49.9	43.5	39.9	26.2	42.3	33.5	30.2	38.3
	SPAM	50.0	47.9	50.0	50.0	50.0	49.8	50.0	50.0	49.9

the presence of additive noise and thus poses a serious threat. Therefore, it is interesting to study the performance of existing MF detectors under additive noise.

To demonstrate the capability of existing MF detectors against noise, the noisy versions of the 512×512 training–testing pairs described in Section V-B are adopted. Specifically, images are contaminated with zero mean Gaussian noise to generate noisy versions with varying signal-to-noise ratios (SNR), i.e., SNR $\in \{50\text{ dB}, 40\text{ dB}, 30\text{ dB}\}$. We note that noise is typically not visible for SNR > 45 dB, and image quality is usually considered acceptable for 36 dB $\leq$ SNR $\leq$ 45 dB, while it is considered poor for SNR < 36 dB.

The classifiers in this case are trained with noise free training sets and then evaluated on noise free testing sets as well as noisy testing sets. Note that we separately train GPF-based and LCF-based detectors instead of the combined GLF-based detector since we want to investigate the individual ability of these two feature sets against noise. The classification results are summarized in Table VII in terms of P_e. For all the involved detectors, the classification performance for noise free images is nearly perfect and SPAM-based method gives the best results with $P_e = 0.0\%$ in all the cases. The performance, however, decreases after the noise is added. It is noted that the performance of GPF-based, MFF-based, and SPAM-based schemes decreases more rapidly, whereas LCF-based method is generally more robust.

Since GPF and SPAM are sort of probabilistic features, which are based on the empirical CDFs and the sample transition probability matrix, respectively, they are very sensitive to noise. Similarly, MFF is based on order statistics and the quantity of gray levels, which are also vulnerable to noise. On the other hand, LCF features capture the joint distribution "pattern," which still remains even in the additive noise environments as illustrated in Fig. 2 (last column). The effects of noise on different MF features and particularly LCF features are further illustrated with principal component analysis (PCA). To this end, we project all the involved features onto the top two principal components for 200 3×3 median-filtered images (512×512) from BOWS2 database and their corresponding noisy versions with SNR $= 50$ dB and 40 dB. As shown in Fig. 5, we can observe that for GPF, MFF, and SPAM features, the projection points of noisy images heavily deviate from those of noise free images, even for high quality images, say, SNR $= 50$ dB. As for LCF features, however, the points of noisy images only slightly shift from their corresponding points of noise free images, indicating LCF features are more robust against noise and therefore more effective to detect MF in the presence of noise.

F. Application to Image Authentication

As demonstrated in the previous subsections that the proposed scheme outperform existing MF detection schemes in the case of strong JPEG post-compression, low resolution, and noise contamination. Therefore, it is expected that our method is more reliable than state-of-the-art MFF-based method to detect tampering involving local median filtering in practical applications. Herein we give an example of a cut-and-paste image forgery in which the pasted region has undergone median filtering before it is inserted into an unaltered image in Fig. 6. To be specific, Fig. 6(a) and 6(b) show two original images of resolution 2100×1500 from NRCS database. The forged image shown in Fig. 6(c) was created by inserting part of the 3×3 median-filtered version of Fig. 6(a) into the unaltered image in Fig. 6(b). Fig. 6(d) shows a binary map that illustrates the part of pasted region with white area.

To detect forgery, we examine the image using a $W_s \times W_s$ sliding window with step size 8×8, and calculate the MF features, i.e., GLF and MFF, of each $W_s \times W_s$ image block. We then classify the image blocks using the corresponding MF detectors trained on the $W_s \times W_s$ MF35 vs. ALL training–testing pair. The detection results obtained using both our method and MFF-based method are illustrated in Fig. 7. Furthermore, we report the corresponding false positive rate P_{FP} and false negative rate P_{FN} in Table VIII. Herein, unaltered and median-filtered regions are taken as negative and positive samples, respectively. We note that when calculating P_{FP} and P_{FN}, image blocks which are around tampered regions and contain both unaltered and median-filtered regions, are not taken into account.

As shown in the first column of Fig. 7, both GLF-based and MFF-based methods can reliably locate the tempered regions in the uncompressed forgery with a small window, i.e., 32×32. The second and third columns of Fig. 7 correspond to the results of $W_s = 64$ for JPEG post-compressed forgery with QF $= 90$ and QF $= 80$, respectively. In this case, the MF detectors are trained with images of the corresponding QF. As shown, the detection performance for both detectors degrades under JPEG post-compression.

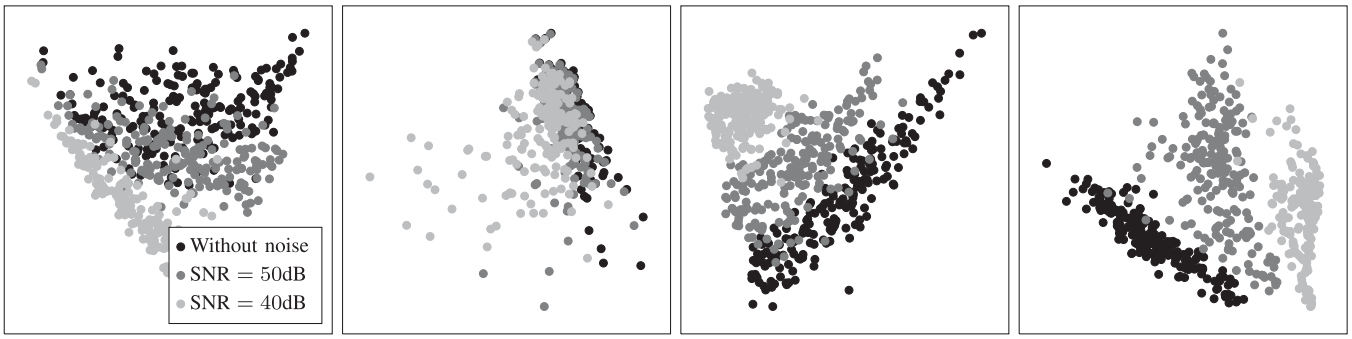


Fig. 5. Projections onto a 2D subspace spanned with top two PCA components for 200 3×3 median-filtered images (512×512) from BOWS2 database and their corresponding noisy versions with SNR = 50 dB and 40 dB. Different features are studied, i.e., (a) GPF, (b) LCF, (c) MFF [4], and (d) SPAM [2].

Fig. 6. (a) and (b) The original images, (c) A cut-and-paste forgery sample produced by cutting part of the 3×3 median-filtered version of (a) and pasting into (b), (d) Binary map for the forgery with white area indicating the part of pasted region.

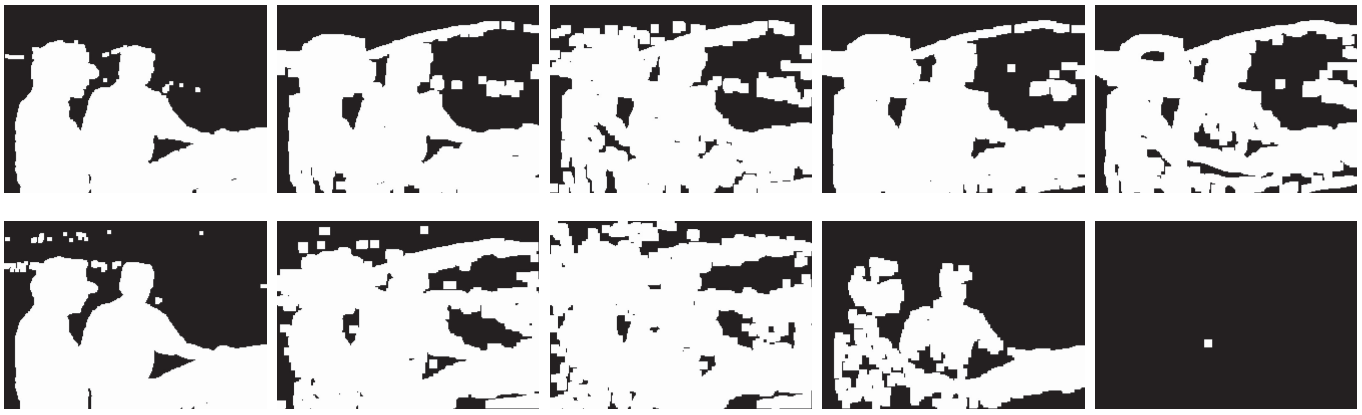


Fig. 7. The detection results using the proposed MF detector (top) and MFF-based detector [4] (bottom). The first column corresponds to the results of $W_s = 32$ for the original forgery. The second and third columns correspond to the results of $W_s = 64$ for JPEG post-compressed forgery with QF = 90 and QF = 80, respectively. The last two columns correspond to the results of $W_s = 64$ for noisy forgery with SNR = 50 dB and SNR = 40 dB, respectively.

TABLE VIII

FALSE POSITIVE RATE P_{FP} (%) AND FALSE NEGATIVE RATE P_{FN} (%) FOR DETECTION RESULTS IN FIG. 7. LEFT: THE PROPOSED MF DETECTOR. RIGHT: MFF-BASED DETECTOR [4]

Forgery detection case	P_{FP}	P_{FN}	P_{FP}	P_{FN}
Original forgery	0.2	0.0	0.8	0.0
JPEG post-compression, QF = 90	4.3	0.0	23.1	1.2
JPEG post-compression, QF = 80	11.8	0.1	38.0	1.3
Noise contamination, SNR = 50dB	4.6	0.0	0.0	7.3
Noise contamination, SNR = 40dB	11.2	0.4	0.0	98.9

Moreover, as reported in Table VIII, GLF-based method yields less P_{FP} as well as less P_{FN} than MFF-based scheme. We also test the performance of involved detectors for forgery detection in the noise environment. To be specific, we herein use LCF-based detector since it is less sensitive against noise. As shown in the last two columns of Fig. 7, LCF-based method gives satisfied detection results for the given noisy images with $W_s = 64$, whereas MFF-based method fails to expose the tampered regions and yields a false negative rate of $P_{\text{FN}} = 98.9\%$ when SNR = 40 dB. We note that better detection performance can be achieved for both detectors if larger window size is adopted, e.g., 128×128, but at the expense of degraded resolution of the tampered regions.

VII. CONCLUSION

In this paper, we investigate the blind detection of median filtering (MF) in digital images. Our work is motivated and

supported with analyses of statistical behaviors in difference domain for median-filtered images and other different image sources. Specifically, we analyze and highlight the statistical artifacts brought into the difference domain of median-filtered images, due to the inherent nature of median filtering. An effective and reliable scheme based on two new feature sets, i.e., GPF and LCF, for MF detection in both uncompressed and JPEG post-compressed images, is then presented. Compared with prior arts, the proposed MF detection method achieves significant performance improvement in the case of low resolution and strong JPEG post-compression. The performance of existing MF detectors against noise is also studied. It is demonstrated that our LCF-based MF detectors yield the most reliable performance.

The proposed method presented in this paper can also be used to discover the image's processing history involving median filtering, providing a useful clue in passive image authentication. For example, it can be employed to detect tampering when part of a median-filtered image is inserted into a non-median-filtered image or vice versa. In this case, our method yields considerable improvement compared with previous approaches.

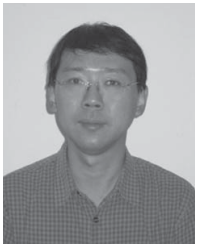
ACKNOWLEDGMENT

The authors would like to thank Dr. H. Yuan at Zhengzhou Information Science and Technology Institute for providing the code of MFF-based scheme in [4]. They also thank the anonymous reviewers and associate editor for their comments that greatly improved the manuscript.

REFERENCES

[1] H. Farid, "A survey of image forgery detection," *IEEE Signal Process. Mag.*, vol. 2, no. 26, pp. 16–25, Mar. 2009.
[2] M. Kirchner and J. Fridrich, "On detection of median filtering in digital images," *Proc. SPIE, Electronic Imaging, Media Forensics and Security II*, vol. 7541, pp. 1–12, Jan. 2010.
[3] G. Cao, Y. Zhao, R. Ni, L. Yu, and H. Tian, "Forensic detection of median filtering in digital images," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2010, pp. 89–94.
[4] H. Yuan, "Blind forensics of median filtering in digital images," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 4, pp. 1335–1345, Dec. 2011.
[5] C. Chen, J. Ni, R. Huang, and J. Huang, "Blind median filtering detection using statistics in difference domain," in *Proc. 14th Int. Conf. Inf. Hiding*, May 2012, pp. 1–15.
[6] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting traces of resampling," *IEEE Trans. Signal Process.*, vol. 53, no. 2, pp. 758–767, Feb. 2005.
[7] Y. T. Kao, H. J. Lin, C. W. Wang, and Y. C. Pai, "Effective detection for linear up-sampling by a factor of fraction," *IEEE Trans. Image Process.*, vol. 21, no. 8, pp. 3443–3453, Aug. 2012.
[8] R. Neelamani, R. de Queiroz, Z. Fan, S. Dash, and R. Baraniuk, "JPEG compression history estimation for color images," *IEEE Trans. Image Process.*, vol. 15, no. 6, pp. 1365–1378, Jun. 2006.
[9] M. Kirchner and R. Böhme, "Hiding traces of resampling in digital images," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 4, pp. 582–592, Dec. 2008.
[10] M. Stamm, S. Tjoa, W. S. Lin, and K. J. R. Liu, "Undetectable image tampering through JPEG compression anti-forensics," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 2109–2112.
[11] A. C. Bovik, "Streaking in median filtered images," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 35, no. 4, pp. 493–503, Apr. 1987.
[12] T. Pevný, P. Bas, and J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 2, pp. 215–224, Jun. 2010.
[13] P. Bas and T. Furon. (2007, Jul.) *Break Our Watermarking System* [Online]. Available: http://bows2.gipsa-lab.inpg.fr
[14] (2008). *Natural Resources Conservation Service Photo Gallery* [Online]. Available: http://photogallery.nrcs.usda.gov
[15] T. Gloe and R. Böhme, "The 'Dresden image database' for benchmarking digital image forensics," in *Proc. ACM Symp. Appl. Comput.*, vol. 2. Mar. 2010, pp. 1584–1590.
[16] I. Pitas and A. Venetsanopoulos, "Order statistics in digital image processing," *Proc. IEEE*, vol. 80, no. 12, pp. 1893–1921, Dec. 1992.
[17] A. C. Bovik, T. S. Huang, and D. C. Munson, "The effect of median filtering on edge estimation and detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 9, no. 2, pp. 181–194, Mar. 1987.
[18] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27:1–27:27, May 2011.
[19] R. Böhme and M. Kirchner, *Counter-Forensics: Attacking Image Forensics*, H. T. Sencar and N. Memon, Eds. New York, NY, USA: Springer-Verlag, 2013.

Chenglong Chen (S'12) received the B.S. degree in physics from Sun Yat-Sen University, Guangzhou, China, in 2010, where he is currently pursuing the Ph.D. degree with the School of Information Science and Technology. His current research interests include image processing, multimedia security, pattern recognition, and in particular digital image forensics.



Jiangqun Ni (M'12) received the Ph.D. degree in electronic engineering from the University of Hong Kong, Hong Kong, in 1998. He was a Post-Doctoral Fellow for a joint program between Sun Yat-Sen University, Guangzhou, China, and the Guangdong Institute of Telecommunication Research, Guangdong, China, from 1998 to 2000. Since 2001, he has been with the School of Information Science and Technology, Sun Yat-Sen University, where he is currently a Professor. His current research interests include data hiding and digital image forensics, image-based modeling and rendering, and image/video processing. He has published more than 50 papers.

Jiwu Huang (M'98–SM'00) received the B.S. degree from Xidian University, Xi'an, China, the M.S. degree from Tsinghua University, Beijing, China, and the Ph.D. degree from the Institute of Automation, Chinese Academy of Science, Beijing, in 1982, 1987, and 1998, respectively. He is currently a Professor with the School of Information Science and Technology, Sun Yat-Sen University, Guangzhou, China. His current research interests include multimedia forensics and security. He serves as a member of the IEEE CASS Multimedia Systems and Applications Technical Committee and the IEEE SPS Information Forensics and Security Technical Committee. He is an Associate Editor for the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, the *LNCS Transactions on Data Hiding and Multimedia Security* (Springer), and the *EURASIP Journal on Information Security* (Hindawi).