Signal Processing: Image Communication 53 (2017) 65–72

Contents lists available at ScienceDirect

Signal Processing: Image Communication

journal homepage: www.elsevier.com/locate/image

Robust median filtering detection based on local difference descriptor

Yakun Niu[a,b], Yao Zhao[a,b,*], RongRong Ni[a,b]

[a] Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China
[b] Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing 100044, China

ARTICLE INFO

Keywords:
Image forensics
Median filtering
Local binary patterns
Pixel difference matrix
Local difference descriptor

ABSTRACT

As a content-preserved image manipulation, median filtering approach has received extensive attention from forensic analyzers. In this paper, we propose a local difference descriptor with two feature sets to reveal the traces of median filtering. The first set of features are fused rotation invariant uniform local binary patterns (LBP), which can quantify the occurrence statistics of micro-features in an image. The second features set is extracted from pixel difference matrix (PDM), which can better describe how pixel values change introduced by median filtering. To validate the effectiveness of the proposed approach, we compare it with the state-of-the-art median filtering detectors in the cases of JPEG compression and low resolution. Experimental results show that our approach outperforms existing detectors. Moreover, our approach is more reliable than prior methods to detect tampering involving local median filtering.

1. Introduction

Owing to the widespread applications of sophisticated digital image editing softwares, digital images can be edited easily and the detection of tampered images becomes more difficult. Meanwhile, digital images have wide applications in journalism, criminal investigation, and law enforcement, it is necessary to develop forensic methods for image authenticity detection. Up to now, a number of blind image forgery detection methods have been proposed. Such methods do not rely on any extra information besides the image itself with undetermined authenticity. By searching for imperceptible traces hidden in the image itself, known as fingerprints, a forensic investigator can determine whether and how an image is manipulated. Existing image forensic works involve the detection of various types of manipulations, such as median filtering [1], resampling [2,3], JPEG compression [4], image splicing [5], and contrast enhancement [6].

Median filtering is a nonlinear operation with the useful property of preserving edges as well as removing noise. Thus, forgers may use median filtering to remove fingerprints left by other manipulations and make their image forgeries more perceptually realistic [7]. As a result, the detection of median filtering is an important issue and the development of reliable methods for median filtering forensic is required.

In this paper, we provide a new robust median filtering forensic approach based on local difference information of an image. We exploit the fact that the occurrence statistics of certain micro-features of median filtered and non-median filtered images exhibit significant inconsistency. Rotation invariant uniform local binary patterns (LBP) can quantify such statistics of micro-features (e.g., edges, spots, flat areas) in an image [8]. Therefore, rotation invariant uniform LBP features are suitable for median filtering detection by tracing the inconsistency. To elaborate the behavior of local pixel values change caused by median filtering, the normalized cross correlation coefficients of pixel difference matrix (PDM) are utilized as the supplement to the rotation invariant uniform LBP. Experimental results demonstrate that the fused features yield better performance than the mere use of the either one. Compared with existing median filtering detection methods, the proposed approach achieves better or comparable performance. Moreover, our approach is more reliable than prior methods for detecting local median filtering in forgery.

The rest of this paper is organized as follows. In Section 2, we briefly review existing works on median filtering detection. Section 3 introduces the local difference descriptor (LDD), whose two feature sets are: the rotation invariant uniform LBP based features and the normalized cross correlation coefficients of pixel difference matrix. Section 4 provides the experimental results and theoretical analyses, and the last section concludes this paper.

2. Related works

Median filtering works by replacing the center pixel with the median of gray levels in a window which slides pixel by pixel over the entire image. For the rest of this work, we focus on the filter with $w \times w$ ($w = 3, 5, \ldots$) square windows of odd size as it is the most widely

* Corresponding author at: Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China.
E-mail address: yzhao@bjtu.edu.cn (Y. Zhao).

http://dx.doi.org/10.1016/j.image.2017.01.008
Received 19 November 2016; Received in revised form 22 January 2017; Accepted 23 January 2017
Available online 24 January 2017
0923-5965/

used form of median filter. Given an $H \times W$ image $I_{i,j}$ with $(i, j) \in \{1, 2, \ldots, H\} \times \{1, 2, \ldots, W\}$, a 2D median filter is defined as

$$\hat{I}_{i,j} = \text{median}\{I_{i+h,\, j+v}\}, \tag{1}$$

where $h, v \in (-(w-1)/2, \ldots, (w-1)/2)$.

Because of the nonlinearity of median filter, it is difficult to theoretically analyze the correlation between its input and output distributions. Median filtering tends to produce constant or nearly constant regions called streaks artifacts [9], which was exploited to detect median filtered images by Kirchner and Fridrich [1]. To identify the presence of streaking artifacts in an image $I(i, j)$, they presented statistical properties of the image's first order pixel difference. Denoting $d_{i,j}^{(k,l)}$ as the first-order difference image with lag (k, l), i.e.,

$$d_{i,j}^{(k,l)} = I_{i+j} - I_{i+k,\, j+l} \tag{2}$$

where $|k| + |l| = 1$ or $|k| + |l| = 2$, and $|\cdot|$ is the operation to calculate the absolute value. Defining $H^{(k,l)} = \{\ldots, h_{-1}^{(k,l)}, h_0^{(k,l)}, h_1^{(k,l)}, \cdots\}$ as the corresponding histogram of the first-order difference. They explored the ratio $\rho^{(k,l)} = h_0^{(k,l)}/h_1^{(k,l)}$ as a detection feature, where $\rho \gg 1$ indicates median filtering. Although this single feature works well in uncompressed images, it fails to detect median filtering for JPEG post-compressed ones. To address the problem, [1] further utilized the subtractive pixel adjacency matrix (SPAM) features [10], which are constructed by modeling the first-order difference as the nth order Markov chains. They reported that SPAM features can be used to detect median filtering in high to medium quality JPEG compressed images. Since typical filter characteristics are suppressed by JPEG artifacts, SPAM classifier might be not able to differentiate between the median filter and other smoothers.

In [11], Cao et al. also utilized the properties of first-order difference image to detect median filtering. They found the probability of zero values on the first-order difference image in textured regions of median filtered image usually was higher than that of original image. Moreover, they demonstrated that their approach can distinguish median filtering from other manipulations, such as rescaling, Gaussian filtering, and average filtering. Experimental results verified the effectiveness of this single feature for median filtering detection in uncompressed images. However, the performance of the approach degrades significantly after JPEG post-compression.

The order or the quantity of gray levels in the image region can be affected by the 2D median filter. Based on this fact, Yuan [12] developed a 44 dimensional features, known as the median filtering forensics (MFF), which extracted five feature subsets based on order statistics and gray levels. The experimental results indicated that the MFF approach can achieve comparable or better performance than SPAM when dealing with low resolution JPEG images. However, MFF approach also suffers performance loss as the JPEG quality factor decreases or as the size of the examined image shrinks.

The authors in [13] reported the first order difference contains a great deal of edge information and block artifacts, which may affect the conditional first order difference distributions. To suppress both image content and block artifacts, they proposed to detect features from the difference between a median filtered version of an image and the image itself. This difference is called an image's median filter residual (MFR), which is defined as

$$r(i, j) = \text{med}_w(y(i, j)) - y(i, j) = z(i, j) - y(i, j) \tag{3}$$

where $y(i, j)$ is original pixel value at point (i, j), and $z(i, j)$ is median filtered value of $y(i, j)$. MFR contains less edge information than the first order difference, thus the image content can be suppressed which may interfere with median filtering detection. Because an autoregressive (AR) model essentially performs linear prediction, the values of the AR coefficients depend heavily on how the MFR values of nearby pixels relate to one another. As a result, the AR coefficients of the MFR are suitable for median filtering detection. Experimental results have shown that AR features can detect median filtering in images that have been JPEG compressed using quality factors as low as 30, as well as in small image blocks.

Neighboring pixels in median filtered image are correlated to some extent because they originate from overlapping windows of the original image. [14] presented global probability feature set (GPF), which is constructed by computing the empirical cumulative distribution function (CDF) for the kth ($k = 1, 2, \ldots$) order difference of the median filtered image. [14] also proposed the local correlations between different adjacent image difference pairs to construct the local correlation feature (LCF) set. Finally, they fused GPF and LCF features to form the global and local feature set (GLF), which can be used to discover the image's history involved median filtering. Their method yielded significant performance improvement in the cases of low resolution and JPEG post-compression.

Motivated by the local ternary pattern (LTP) [15], a second-order LTP operator was proposed to effectively reveal the traces introduced by median filtering [16]. The 2nd LTP operator encoded LTP local derivative direction variations by using a 3-valued coding function instead of the binary one. However, the dimensionality of the 2nd order LTP histogram is too large. To overcome this problem, they split the 2nd order LTP in each direction into its positive and negative halves to reduce the features to 2048 dimensions. Similar with LTP, the second-order LTP also only use the sign component of the pixel difference, and the magnitude component is completely ignored.

3. Local difference descriptor feature set

In this section, we introduce the local difference descriptor which contains joint histogram of rotation invariant uniform LBP and normalized cross correlation coefficients of PDM. The two sets of features are jointly applied to reveal the traces of median filtering.

3.1. Joint histogram of rotation invariant uniform LBP

Local binary patterns (LBP) have been successfully applied to face recognition [17], texture classification [18], and background modeling [19]. Given a pixel in an image, the LBP code can be formally described as follows:

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c)2^p,\ s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{4}$$

where g_c is the gray value of the central pixel, g_p is the value of its neighboring pixel, P is the total number of involved neighbors, and R is the radius of the neighborhood. Fig. 1 illustrates an example of obtaining LBP code in case of $P = 8$ and $R = 1$.

To achieve rotation invariance, the binary pattern is circularly shifted and the minimum value is saved as the final binary pattern. The rotation invariant LBP is defined as

$$LBP_{P,R}^{ri} = \min\{ROR(x, i)\} \tag{5}$$

where $ROR(x,i)$ performs a circular bit-wise right shift on the binary number x i times. $LBP_{P,R}^{ri}$ quantifies the occurrence statistics of individual rotation invariant patterns corresponding to certain micro-features in the image. Using $P = 8$ and $R = 1$, there are 36 unique

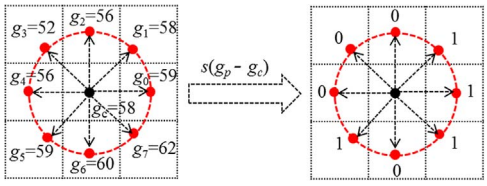

Fig. 1. Calculation of the original LBP code (10100011). g_p is the ith neighboring pixel and g_c is the center pixel.

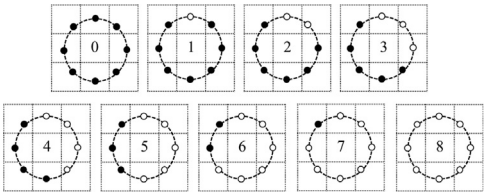

Fig. 2. The uniform rotationally invariant LBP. Black and white circles correspond to bit values of 0 and 1 in the 8-bit output of the operator.

rotation invariant LBP patterns which can detect spots, flat areas, and edges. If the binary pattern contains at most two bitwise transitions from 0 to 1 or vice versa, it is called the uniform LBP pattern. The rotation invariant LBP operator based on uniform patterns, denoted here as $LBP^{riu2}_{P,R}$, is achieved by circularly rotating each bit pattern to the minimum value.

$$LBP^{riu2}_{P,R} = \begin{cases} \sum_{p=0}^{P-1} s(g_p - g_c), & if\ U(LBP_{P,R}) \le 2 \\ P+1, & otherwise \end{cases} \quad (6)$$

where $U(LBP_{P,R})$ corresponds to the number of spatial 0/1 transitions in the LBP pattern. For example, the binary label sequences "01111100" and "00000010" are uniform LBP patterns due to the 2 transitions, while the sequences "00000101" and "00011010" are non-uniform LBP patterns due to 3 and 4 transitions, respectively. 9 ration invariant uniform patterns in neighborhood of 8 pixels are shown in Fig. 2.

Pattern #0 detects bright spots, #8 dark spots and flat areas, and #4 edges. Conventional $LBP_{P,R}$ could produce 2^P different patterns, and the histogram distribution of these patterns is uneven. Some codes appear less than others in many real-life images, which means that these bins in the LBP histogram are less informative. Therefore, improved rotation invariant uniform patterns not only remove some less informative bins but also reserve discriminative ones. Different from many popular smoothing filters, median filter not only performs well in smoothing noise and preserving edges, but also inclines to produce regions of constant or nearly constant pixel values. Therefore, the distributions of these patterns in median filtered images are inconsistent with that in non-median filtered images.

Fig. 3 shows the histogram of the nine uniform LBP patterns and non-uniform pattern appear in original and median filtered images, respectively, which are all from BOWS2 Database [20].

Bin 0 is the number of bright spots which can be considered as the noises. The frequency of this pattern in median filtered images is noticeably less than that in original images. This indicates that, median filter is very effective at removing noise. Bin 1 and 2 corresponds to #1 and #2 in Fig. 2, respectively, and those two patterns in original images are higher than that in median filtered images. Note that the quantity of bin 4 in median filtered images is equal to that in original images. This effect is related to another inherent nature of median filter known as good edge preservation. In median filtered images, bin 8 occurs more frequently than that in original images. This reveals that the median

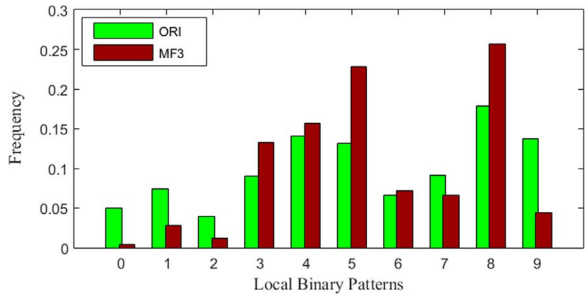

Fig. 3. The uniform rotationally invariant LBP histogram from 10000 3×3 median-filtered images (MF3) and 10000 original images (ORI). Bins 0–8 are the quantities of the uniform patterns; bin 9 is the quantity of the non-uniform patterns. All these images are from the BOWS2 database.

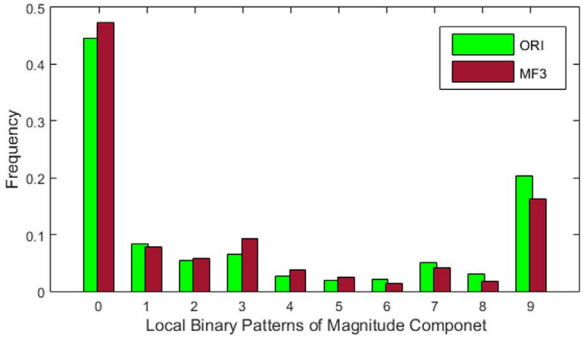

Fig. 4. The uniform rotationally invariant LBP histogram of magnitude component from 10000 3×3 median-filtered images (MF3) and 10000 original images (ORI). Bins 0–8 are the quantities of the uniform patterns; bin 9 is the quantity of the non-uniform patterns. All these images are from the BOWS2 database.

filter tends to produce regions of constant or nearly constant intensities, which is known as *streaks artifacts* [9]. Furthermore, the amount of non-uniform patterns in original images is more than that in median filtered images. Therefore, the histogram of rotation invariant uniform LBP can be used for median filtering forensics.

Because the conventional LBP descriptor just use the sign information of image local difference, and the magnitude information is discarded. [21] proposed a completed LBP descriptor by decomposing the image local difference into two complementary components, i.e., the sign (s_p) and the magnitude (m_p), respectively. $s_p = s(g_p - g_c)$ and $m_p = |g_p - g_c|$, where g_p, g_c and $s(x)$ are defined in Eq. (4). Then they defined the CLBP-Sign (CLBPS) (i.e., LBP) and CLBP-Magnitude (CLBPM) as:

$$CLBPM_{P,R} = \sum_{p=0}^{P-1} s(m_p - c)2^p. \quad (7)$$

where c denotes the mean value of m_p in the whole image.

The histogram of the nine uniform and the non-uniform CLBPM patterns are shown in Fig. 4. We can find that bin 0 and 9 occupy most of the patterns. The quantity ranging from bin 2–5 of original images is less than that of median filtered images. On the contrary, the quantity ranging from bin 6–9 of original images is more than that of median filtered images. The center gray level component of CLBP is discard, because it describes the global information of an image. The histogram of the CLBPC cannot distinguish the median filtered images from non-median filtered images. The mappings from $LBP_{P,R}$ to $CLBPS^{riu2}_{P,R}$ and $CLBPM^{riu2}_{P,R}$, which have $P+2$ distinct output values, can be implemented with a lookup table of 2^P elements. Finally, we use the joint 2D histogram of the $CLBPS^{riu2}_{P,R}$ and $CLBPM^{riu2}_{P,R}$ codes as our first features set (JHLBP) with $(P+2)^2$ dimensions for median filtering detection.

3.2. Correlation coefficients of PDM

In this section, features based on the correlation between local pixel difference pairs are utilized as the supplement to the JHLBP features. The local pixel differences can better describe how pixel values change and implicitly encode important micro structures. To show the behavior of local pixel difference pair, we present the joint probability distribution of local pixel difference pair which is denoted as

$$P^{p,q}_{i,j}(t_x, t_y) = \Pr(g^p_{i,j} - g_{i,j} = t_x,\ g^q_{i,j} - g_{i,j} = t_y). \quad (8)$$

$g^p_{i,j}$ is the neighbor pixel value of $g_{i,j}$ which is located at center (i,j) with $p \neq q$ and $p, q \in \{0, 1, \ldots, 7\}$. Fig. 5 shows the joint probability distributions of local pixel difference pairs, which are estimated using the images from BOWS2 database. The joint probability distributions of original images are shown on the left column, while the middle and right columns correspond to median filtered and average filtered images. In these figures, the horizontal and vertical axes represent t_x

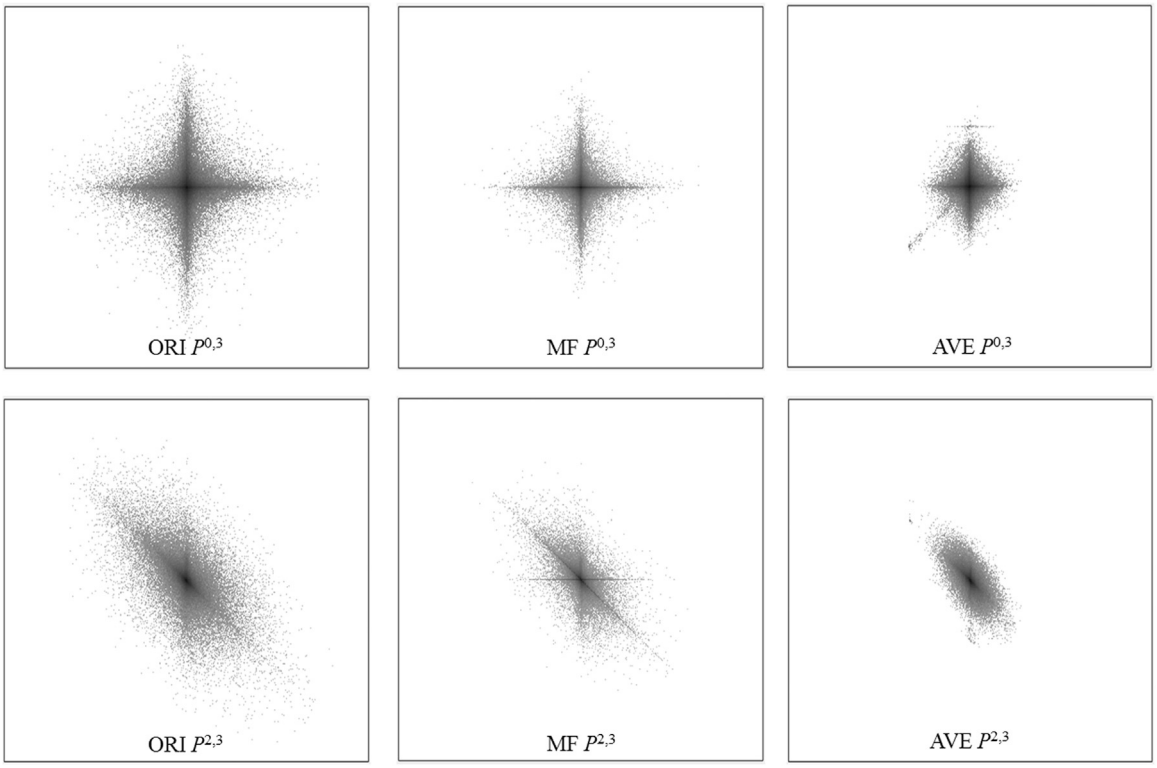

Fig. 5. Joint probability distributions of local pixel difference pairs for different image sources. All images are from the BOWS2 database. Top: $p=0$, $q=3$. bottom: $p=2$, $q=3$. From left to right: Original images (ORI), median filtered images (MF), and average filtered images (AVE). The median filtered images and average filtered images are obtained with the 3×3 square windows.

and t_y, respectively, and t_x, $t_y \in [-255, 255]$. The intensity values at (t_x, t_y) represent the probabilities, whereas greater points refer to larger probabilities.

From Fig. 5, we can observe that joint probability distribution of original images is similar to that of median filtered images. However, the latter more tends to cluster to center, i.e., $t_x = 0$ and $t_y = 0$. This refers to the inherent effect of the median filter that often produces regions of constant or nearly constant intensities [9]. In addition, like ORI $P^{0,3}$, MF $P^{0,3}$ has four peaks. This comes from the fact that median filter can preferably preserve the edge information. The right column shows the joint probability of local pixel difference pairs of average filtered images. Compared right column with middle column, the right column also tends to cluster to center, while has no peaks. This is mainly because of the fact that average filter does not possess the ability to preserve edge information.

In order to take advantage of the joint probability distribution of local pixel difference pairs, we obtain our second set features using a compact metric. Such metric can be computed based on the normalized cross correlation utilized in [14]. Firstly, we obtain the pixel difference vector (PDV), which consists of the local pixel difference pairs. For any center pixel in an image, the differences between the center pixel and P neighboring pixels are computed as the PDV, i.e., $\boldsymbol{d}_i = [d_{i,1}, d_{i,2}, \ldots, d_{i,P}]^T$. After eliminating the PDVs whose elements are all zero values, the remaining PDVs are composed the pixel difference matrix (PDM) which can be defined as

$$\mathbf{M} = \begin{bmatrix} d_{1,1} & d_{2,1} & \cdots & d_{N,1} \\ d_{1,2} & d_{2,2} & \cdots & d_{N,2} \\ \vdots & \vdots & \ddots & \vdots \\ d_{1,P} & d_{2,P} & \cdots & d_{N,P} \end{bmatrix}$$

where N is the number of PDVs.

The PDV measures the differences between the center point and neighboring pixels within a patch, thus it can better describe how pixel values change and can implicitly encode important visual patterns such as edges and lines in images [22,23]. Joint probability is suitable to elaborate the behavior of local difference pairs. Therefore, the Normalized Cross Correlation (NCC) coefficients can be used as features to capture the joint probability of local difference pairs. The NCC coefficient of PDM is defined in (9), where $i, j \in \{1, 2, \ldots, P\}$, and $i \neq j$. $[d_{1,i}, d_{2,i}, \ldots, d_{N,i}]$ and $[d_{1,j}, d_{2,j}, \ldots, d_{N,j}]$ are the ith and jth rows of PDM, respectively. $cov(,)$ is the covariance. We define $D([d_{1,i}, d_{2,i}, \ldots, d_{N,i}]) = cov([d_{1,i}, d_{2,i}, \ldots, d_{N,i}], [d_{1,i}, d_{2,i}, \ldots, d_{N,i}])$, and $D([d_{1,j}, d_{2,j}, \ldots, d_{N,j}])$ is defined similarly. Then, $\boldsymbol{\gamma}(i, j)$ is defined as

$$\boldsymbol{\gamma}(i,j) = \frac{cov([d_{1,i}, d_{2,i}, \ldots, d_{N,i}], [d_{1,j}, d_{2,j}, \ldots, d_{N,j}])}{\sqrt{D([d_{2,i}, \ldots, d_{N,i}])}\sqrt{D([d_{1,j}, d_{2,j}, \ldots, d_{N,j}])}} \tag{9}$$

Our second set of features can be summarized as follows

1. group the PDVs to form the PDM with P rows;
2. consider an arbitrary row of PDM as a random variable and obtain the NCC coefficients of any different variables;
3. concatenate all the NCC coefficients of PDM (CPDM) to yield a $P(P-1)/2$-D feature vector $\boldsymbol{C}$.

The PDM has P rows in total. Considering an arbitrary row of PDM as a random variable, we can obtain $\binom{P}{2}$ different cross correlation coefficients. Concatenating all the coefficients together leads to CPDM features of $P(P-1)/2$ elements. If we set $P=8$, the dimensions of CPDM features are 28. Indeed, the NCC coefficient of PDM is similar to LCF feature, which captures the correlation of adjacent difference pairs [14]. The difference between LCF and CPDM is that CPDM not only includes adjacent difference pairs, but also includes the non-adjacent difference pairs. Fig. 6 shows the two classes of local difference pairs: adjacent and non-adjacent local difference pairs.

Combining JHLBP features and CPDM features, we obtain the final LDD features with $(P+2)^2 + P(P-1)/2$ elements for median filtering detection. Although the theory of LBP is simple, it is computationally expensive because the LBP dimension can be very high. For example,

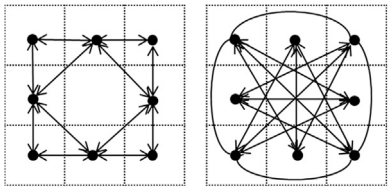

Fig. 6. The adjacent local difference pairs (left) and the non-adjacent local difference pairs (right). There are 12 kinds of ways to combine the adjacent difference pairs (left), and each line presents a adjacent local difference pair. There are 16 kinds of ways to combine the non-adjacent difference pairs (right), and each line presents a non-adjacent local difference pair.

when $P=16$, $R=2$, $LBP_{16,2}$ has 2^{16} distinct output values. In order to reduce the computational complexity, we set $R=1$ and $P=8$ in this paper.

In order to demonstrate the discriminate ability of above extracted JHLBP, CPMD and combined LDD features, we project them to 2D space by using linear discriminant analysis (LDA). Fig. 7 shows the 2D projection results from the three features of unaltered, median filtered and average filtered images in the Uncompressed Color Image Database (UCID) [24]. We use randomly 200 512×384 images from the three different sources. From these figures, we can clearly see that unaltered, median filtered and average filtered images can be separated via the three features. This further prove the discrimination capability of our features.

4. Experiment and discussion

We evaluate our proposed median filtering detector on the widely used UCID [24] which consists of 1338 uncompressed RGB images with the size of 512×384. Most of the images are characterized by saturated pixels or uniform intensity regions. Such images are very challenging for discriminating the median filtered images from non-median filtered images. All color images are first converted to gray scale images before further processing. Median filtered images are generated by performing 3×3 and 5×5 median filtering on the unaltered gray-scale images.

To test the effectiveness of the proposed approach, we compare it with other widely used methods, which include GLF method [14], AR method [13] and the 2nd LTP method [16]. We also evaluate the ability our proposed approach by differentiating between median filtering and other popular tools, including 3×3 Gaussian low-pass filter with the standard deviation $\sigma=0.5$ (GAU), 3×3 average filter (AVE) and the rescale operation with the scaling factor of 1.1 (RES). The C-SVM [25] with Gaussian kernel is used as the classifier in our experiments. Using the five-fold cross-validation in conjunction with a grid search, we obtain the best kernel parameters for the SVM. The grid search for the optimal parameters are performed on the multiplicative grid $(C,\gamma)\in\{(2^i,2^j)|i,j\in Z\}$. We use those optimal parameters to get the classifier model on the entire training set, and the trained classifier model is used to perform a classification on the testing set. Specifically, the images in the UCID database are randomly divided into four folds of nearly equal size. The training set is composed of three folds, while the remaining fold is used for evaluation. The performance of each detection method is summarized by the minimal average decision error of each technique under the assumption of equal priors and equal costs,

$$P_e=\min\left(\frac{P_{fp}+1-P_{tp}}{2}\right) \tag{10}$$

where P_{fp} and P_{tp} denote the false positive and true positive rates, respectively.

4.1. Efficacy of the proposed local difference descriptor

To demonstrate the efficacy of the proposed approach, we evaluate its ability to detect median filtering in images that are JPEG compressed with QF=70. As shown in Fig. 8, the three features can achieve satisfactory detection performance. It is observed that LDD features are superior to both JHLBP and CPDM. To demonstrate the discrimination capability of non-adjacent difference pairs, we compare our proposed features with the 1st-order adjacent difference pairs component of LCF (FLCF) for median filtering detection. The detailed experimental results on classification accuracy are presented in Table 1. MF3 and MF5 denote 3×3 and 5×5 median filtering operation, respectively. ORI denotes an image without any operations. We can observe that CPDM achieves better performance than FLCF. Such result demonstrates non-adjacent difference pairs also have discrimination capability to detect median filtering. It is noted that the LDD is superior to both JHLBP and CPDM. At $P_{fp}=1\%$, the JHLBP, CPDM, and LDD features achieve $P_{tp}=78.7\%$, $P_{tp}=77.5\%$ and $P_{tp}=81.1\%$ for 3×3 median filtering.

4.2. The performance of detecting median filtering in uncompressed images

To evaluate the performance of our approach to detect median filtering in uncompressed images, we compare our approach with the state-of-the-art methods on the UCID database. The experimental results are presented in Table 2. From these results, we can see that those detectors are very reliable to reveal the traces caused by median filtering and other operations. Although LDD can not achieve optimal performance in the cases of MF3 VS AVE and MF5 VS ORI, it outperforms other detectors in general.

4.3. Detecting median filtering in low-resolution and jpeg post-compressed images

In reality, only a portion of a median filtered image is inserted into a non-median filtered image, then the forged image is saved as a JPEG format. It is necessary to detect median filtering in low-resolution and JPEG compressed images. To test the performance of the proposed

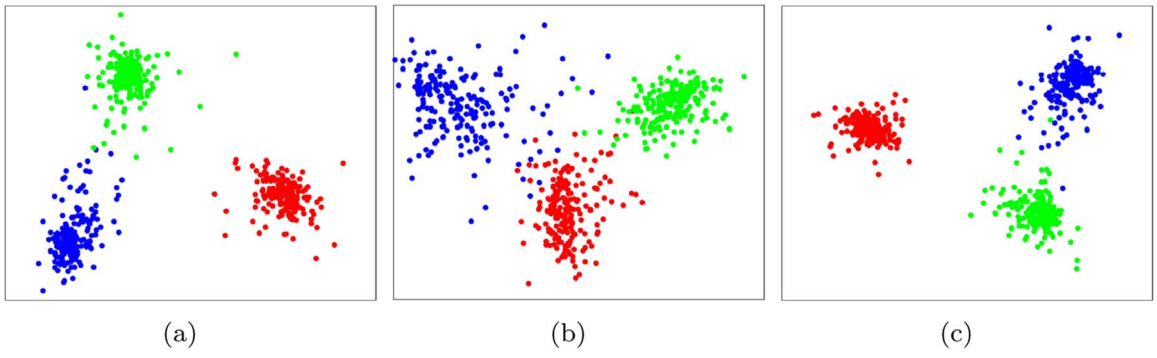

Fig. 7. Distribution of 2D projection results from the proposed features by using linear discriminant analysis (LDA). (a), (b), and (c) correspond to the JHLBP, CPDM and LDD features. Markers with the blue, red, green color denote the unaltered, median filtered and average filtered images source, respectively. Discrimination capability is shown via the clustering effects.

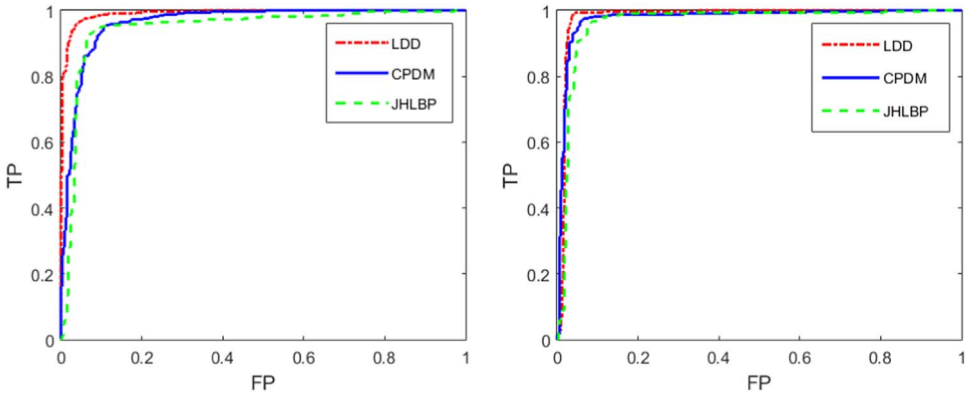

Fig. 8. Roc cures show 3×3 (left) and 5×5 (right) median filtering detection performance on JPEG compressed images. The proposed JHLBP, CPMD and LDD features are tested.

Table 1
Classification results in terms of P_e(%) for 256×256 JPEG post-compressed images with QF=70. The JHLBP, CPDM, LDD and FLCF features are tested. (The best results are displayed with bold texts.).

		MF3				MF5			
		ORI	GAU	RES	AVE	ORI	GAU	RES	AVE
FLCF	P_e(%)	11.5	16.4	15.7	5.8	4.0	4.3	3.1	3.9
JHLBP	P_e(%)	7.3	10.0	11.9	6.3	6.6	8.5	10.3	8.0
CPDM	P_e(%)	9.1	9.2	8.9	2.1	4.3	3.5	3.0	1.5
LDD	P_e(%)	**2.9**	**2.7**	**3.5**	**1.1**	**2.4**	**2.0**	**2.3**	**0.6**

Table 2
Classification results in terms of P_e(%) for 512×384 uncompressed images. Our proposed method, GLF method, AR method, and 2nd order LTP method are tested. (The best results are displayed with bold texts.).

	MF3				MF5			
	ORI	GAU	RES	AVE	ORI	GAU	RES	AVE
GLF	0.20	0.18	0.12	0.09	0.05	0.21	0.33	0.13
AR	0.59	0.68	1.17	1.05	0.32	0.34	0.20	0.18
2OLTP	**0.07**	0.10	0.12	0.60	**0.04**	0.05	0.10	0.46
LDD	0.10	**0.05**	**0.10**	**0.05**	0.09	**0.03**	**0.02**	**0.08**

approach on small image windows, we create a database to test image blocks by cropping a block of size 256×256, 128×128, 64×64 and 32×32 from the center of an image. For the sake of brevity, we only compare our approach with other methods on JPEG 70 compressed images. The experimental results are presented in Fig. 9 and Table 3.

From Fig. 9, we can observe that the ROC curves of four schemes for both 3×3 and 5×5 median filtering detections are similar to each other. It illustrates the four schemes are all robust against JPEG compression. The magnifications of the ROC curves show that our scheme is more robust against JPEG post-compression. The P_e values of LDD, AR, GLF, and LTP are 2.9%, 3.4%, 3.6%, and 3.3%, respectively. As shown in Table 3, although our scheme can not obtain best classification performance in cases of MF5 VS ORI and MF5 VS RES, it achieves nearly classification performance. For example, if median filtering is distinguished from upscaling operation for 128×128 pixel blocks, the P_e values achieved by our scheme and 2OLTP are 4.7% and 4.6%, respectively.

Table 3
Classification results in terms of P_e(%) for JPEG 70 compressed images with varying sizes. Our proposed method, GLF method, AR method and 2nd order LTP (2OLTP) method are tested. (The best result for each training-testing pair is displayed with bold texts).

M	Features	MF3				MF5			
		ORI	GAU	RES	AVE	ORI	GAU	RES	AVE
128	GLF	10.2	11.1	13.2	4.3	5.2	3.9	5.5	2.9
	AR	**4.8**	9.1	13.4	4.9	7.1	7.2	8.0	2.2
	2OLTP	7.7	6.5	9.2	2.0	**4.1**	5.8	**4.6**	0.9
	LDD	6.4	**3.3**	**7.2**	**1.2**	4.3	**3.4**	4.7	**0.9**
64	GLF	12.2	14.9	19.1	6.1	**6.9**	5.4	**8.0**	3.9
	AR	11.7	10.1	16.3	9.2	8.3	9.7	11.8	3.9
	2OLTP	12.9	9.7	17.5	2.3	7.9	4.7	9.5	1.9
	LDD	**10.9**	**5.8**	**13.5**	**1.6**	7.6	**4.0**	9.6	**1.1**
32	GLF	20.2	18.2	28.1	8.9	**10.3**	7.7	13.8	5.3
	AR	18.9	16.1	27.5	7.7	16.7	13.0	16.6	5.1
	2OLTP	19.7	14.3	**23.1**	3.4	11.6	7.2	**13.5**	3.0
	LDD	**17.8**	**8.1**	23.6	**3.0**	13.4	**6.3**	15.8	**2.3**

In order to demonstrate our method is reliable to identify the median filtering involved spliced image forgeries. An example of a cut-and-paste image forgery is shown in Fig. 10. The pasted region is from an median filtered image before it is insert into an unaltered image.

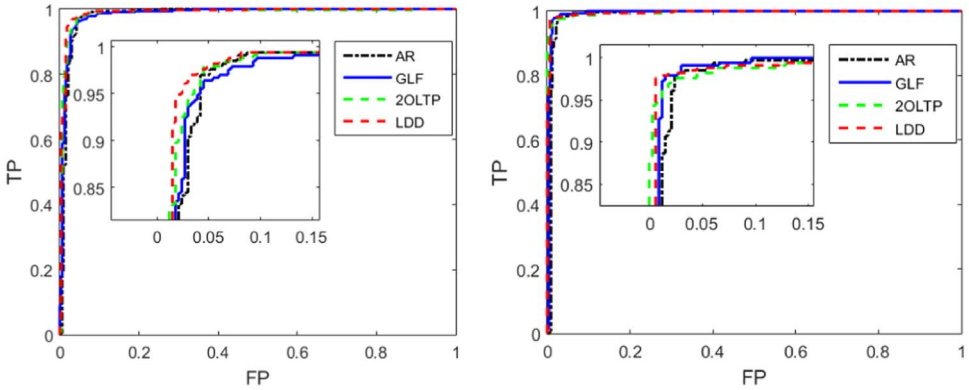

Fig. 9. Roc cures show 3×3 (left) and 5×5 (right) median filtering detection performance on JPEG compressed images of size 256×256.

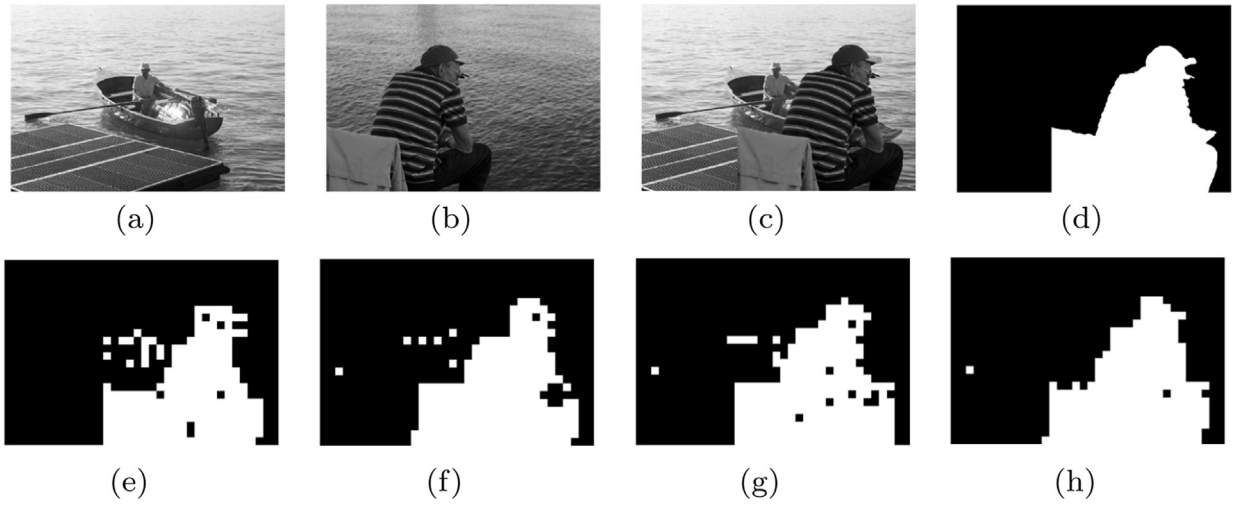

Fig. 10. (a) The unaltered image into which the cut object is pasted, (b) the 3×3 median filtered image from which an object was cut, (c) the composite image which is JPEG compressed using a quality factor of 70, (d) binary map for the forgery with white area denoting the part of pasted region. (e) Block-wise detections using AR method, (f) block-wise detections using GLF method, (g) block-wise detections using 2OLTP method, and (h) block-wise detections using LDD method.

Table 4
Classification results in terms of P_e(%) for JPEG compressed images with varying QFs. Our proposed method, GLF method, AR method and 2nd order LTP (2OLTP) method are tested. (The best result for each QF and each training-testing pair is displayed with bold texts.).

Features	MF3			MF5		
	QF = 50	QF = 40	QF = 30	QF = 50	QF = 40	QF = 30
GLF	12.1	12.4	13.8	5.1	6.0	6.8
AR	9.1	**9.4**	10.6	6.9	7.1	7.8
2OLTP	10.7	11.5	13.7	5.4	5.6	7.4
LDD	**8.9**	9.8	**10.1**	**4.7**	**5.6**	**6.4**

Figs. 10(a) and 10(b) show the unaltered image into which the cut object is pasted and the 3×3 median filtered image from which an object is cut, respectively. Fig. 10(c) shows the composite image, which has been undergone JPEG compressed using a quality factor of 70. Fig. 10(d) shows a binary map that illustrates the part of pasted region with white area. To identify the forgery, the spliced image is segmented into 128×128 pixel blocks, and the AR, GLF, 2OLTP and LDD features of each block are calculated. Then we classify the image blocks using the corresponding detection methods trained on 128×128 MF3 VS NMF (non-median filtered) blocks. Figs. 10(e), 10(f), 10(g) and 10(h) show the results block-wise detections on the composite image using the AR, GLF, 2OLTP and LDD methods. As we can see, multiple false alarms occur in Figs. 10(e), 10(f), and 10(g), and the inauthentic object can not be located with the three methods correctly. We can observe that our proposed method achieves the best performance in the cut-and-paste forgery detection.

4.4. Detecting median filtering under jpeg post-compression with varying quality factors

The above discussions for median filter detection are on JPEG compressed images only with a QF of 70. To estimate the performance of the proposed method for median filtering detection in low quality images with strong JPEG compression, the JPEG QF are set to 50, 40 and 30, respectively. For the sake of brevity, We only report the results obtained on 128×128 JPEG post-compressed images. As shown in Table 4, we observe that the performances of all methods degrade as the JPEG QF decreases. However, our method also is more reliable than other methods for median filtering detection against strong JPEG compression.

5. Conclusion

In this paper, we have proposed a novel approach to detect median filtering in digital images. Median filter not only suppresses noise and preserves edges, but also tends to produce regions of constant or nearly constant pixel values. Therefore, the occurrence statistics of certain micro-features of median filtered and non-median filtered images exhibit significant inconsistency. Rotation invariant uniform LBP can quantify the statistics of micro-features. By tracing the inconsistency, the rotation invariant uniform LBP can be used as effective fingerprint for median filtering detection. The supplementary features extracted by using the NCC coefficients of PDM which captures the statistical artifacts introduced by median filter. We jointly apply the JHLBP and CPMD features to reveal the traces of median filtering. Experiments have demonstrated that our proposed median filtering forensic approach outperforms existing detectors. Additionally, our approach is more reliable than prior methods for detecting local median filtering in forgery.

Acknowledgment

The authors would like to thank X. Kang and A. Peng for providing the code of AR scheme in [13]. They also thank the anonymous reviewers and associate editor for their helpful comments that greatly improve the paper. This work is sponsored by the National Key Research and Development of China (NO. 2016YFB0800404), the National Natural Science Foundation of China (No. 61332012).

References

[1] M. Kirchner, J. Fridrich, On detection of median filtering in digital images, in: Proceedings SPIE, Electronic Imaging, Media Forensics and Security II. (2010) 1–12.
[2] A.C. Popescu, H. Farid, Exposing digital forgeries by detecting traces of resampling, IEEE Trans. Signal Process. 53 (2) (2005) 758–767.
[3] G. Cao, Y. Zhao, R. Ni, Forensic identification of resampling operators: a semi non-intrusive approach, Forensic Sci. Int. 216 (1) (2012) 29–36.
[4] R. Neelamani, R. de Queiroz, S.D.Z. Fan, R. Baraniuk, Jpeg compression history estimation for color images, IEEE Trans. Image Process 15 (6) (2006) 1365–1378.
[5] P. Ferrara, T. Bianchi, A.D. Rosa, A. Piva, Image forgery localization via fine-grained analysis of cfa artifacts, IEEE Trans. Inf. Forensics Secur. 7 (5) (2012) 1566–1577.
[6] G. Cao, Y. Zhao, R. Ni, X. Li, Contrast enhancement-based forensics in digital images, IEEE Trans. Inf. Forensics Secur. 9 (3) (2014) 515–525.
[7] M. Kirchner, R. Bo, Hiding traces of resampling in digital images, IEEE Trans. Inf. Forensics Secur. 3 (4) (2008) 582–592.
[8] T. Ojala, M. Pietikinen, T. Menp, Multiresolution gray scale and rotation invariant texture analysis with local binary patterns, IEEE Trans. Pattern Anal. Mach. Intell.

24 (7) (2002) 971–987.
[9] A.C. Bovik, Streaking in median filtered images, IEEE Trans. Acoust., Speech Signal Process. 35 (4) (1987) 493–503.
[10] T. Pevny, P. Bas, J. Fridrich, Steganalysis by subtractive pixel' adjacency matrix, IEEE Trans. Inf. Forensics Secur. 5 (2) (2010) 215–224.
[11] G. Cao, Y. Zhao, L.Y.R. Ni, H. Tian, Forensic detection of median filtering in digital images, in: IEEE International Conference Multimedia Expo. (2010) 89–94.
[12] H. Yuan, Blind forensics of median filtering in digital images, IEEE Trans. Inf. Forensics Secur. 6 (4) (2011) 1335–1345.
[13] K. X, S.M.C, P. A, et al., Robust median filtering forensics using an autoregressive model, IEEE Trans. Inf. Forensics Secur. 8 (9) (2013) 1456–1468.
[14] C. C, N. J, H. J, Blind detection of median filtering in digital images: A difference domain based approach, IEEE Trans. Image Process. 22 (12) (2013) 4699–4710.
[15] T. X, T. B., Enhanced local texture feature sets for face recognition under difficult lighting conditions, IEEE Trans. Image Process. 19 (6) (2010) 1635–1650.
[16] Z. Y, L. S, W. S, et al., Revealing the traces of median filtering using high-order local ternary patterns, IEEE Signal Process. Lett. 21 (3) (2014) 275–279.
[17] A. T, H. A, P. M, Face description with local binary patterns: Application to face recognition, IEEE trans. Pattern Anal. Mach. Intell. 28 (12) (2006) 2037–2041.
[18] L. S, L.M.W. K, C.A.C. S, Dominant local binary patterns for texture classification, IEEE trans. Image process. 18 (5) (2009) 1107–1118.
[19] M. Heikkil, M. Pietikinen, A texture-based method for modeling the background and detecting moving objects, IEEE Trans. Pattern Anal. Mach. Intell. 28 (4) (2006) 657–662.
[20] P. Bas, T. Furon, Break our watermarking system. http://dx.doi.org/bows2.ec-lille.fr/2nd.
[21] Z.H. Guo, L. Zhang, D. Zhang, A completed modeling of local binary pattern operator for texture classification, IEEE Trans. Image Process 19 (6) (2012) 3844–3852.
[22] L. J, L. V. E, Z. X, et al., Learning compact binary face descriptor for face recognition, IEEE Trans. Pattern Anal. Mach. Intell. 37 (10) (2015) 2041–2056.
[23] L. Z, M. Pietikinen, L. S. Z, Learning discriminant face descriptor, IEEE Trans. Pattern Anal. Mach. Intell. 36 (2) (2014) 289–302.
[24] G. Schaefer, M. Stich, Ucid-an uncompressed color image database, in Proceedings SPIE, Storage and Retrieval Methods and Applicat. for Multimedia. 36 (2004) 472–480.
[25] C.-C. Chang, C.-J. Lin, Libsvm: a library for support vector machines, ACM Trans. Intell. Syst. Technol. 36 (2011) (2:27:1–27:27).