Multimed Tools Appl (2016) 75:16577–16595
DOI 10.1007/s11042-016-3575-z

Rotation and scale invariant upsampled log-polar fourier descriptor for copy-move forgery detection

Chun-Su Park[1] · Changjae Kim[2] · Jihoon Lee[3] · Goo-Rak Kwon[4]

Received: 19 October 2015 / Revised: 21 March 2016 / Accepted: 3 May 2016 /
Published online: 11 May 2016

Abstract Digital image forgery is becoming increasingly popular with the rapid progress of digital media editing tools. Copy-move forgery (CMF) is one of the most common methods of digital image forgery. For CMF detection (CMFD), we propose an upsampled log-polar Fourier (ULPF) descriptor that is robust to several geometric transformations including rotation, scaling, sheering, and reflection. We first describe the theoretical background of the ULPF representation. Then, we propose a feature extraction algorithm that can extract rotation and scale invariant features from the ULPF representation. In addition, we analyze the common CMFD processing pipeline and improve a part of processing pipeline to efficiently handle various types of tampering attacks. In our simulation, we present comparative results between the proposed feature descriptor and state-of-the-art ones with proven performance guarantees.

✉ Goo-Rak Kwon
grkwon@chosun.ac.kr

[1] Department of Software, Sejong University, 209, Seoul, South Korea

[2] Department of Civil and Environmental Engineering, Myongji University, 116, Yongin, South Korea

[3] Department of Information and Communication Engineering, Sangmyung University, 31, Cheonan, South Korea

[4] Department of Information and Communication Engineering, Chosun University, Gwangju, South Korea

Keywords Digital image forgery · Copy-move forgery · Upsampled log-polar Fourier features · Common processing pipeline

1 Introduction

With the rapid progress of digital-image-editing software, digital images can be easily manipulated, leaving no perceptible trace. The field of digital forensics has emerged to restore public trust in digital images. Image forensic techniques are generally categorized into two groups: active and passive forensic techniques [16, 32]. Active forensic techniques such as digital watermarking insert a watermark or signature into the image at the time of recording and detect tampered regions using the embedded information [38, 44]. Basically, the application of active approaches is limited because the majority of consumer images are created without containing any digital watermark or signature. In contrast to active approaches, passive image forensic techniques have been developed for image authentication without the need for any prior knowledge [7]. These techniques work on the assumption that, although digital forgeries may leave no visual clue, they alter the underlying statistics of an image [10]. In this work, we focus on passive image forensics.

Digital images can be tampered or manipulated in many different ways. Copy-move forgery (CMF), which copies a part of the image and pastes it into another region, is one of the most common methods for digital image tampering [26, 33]. In the CMF scenario, a tampered region might not be exactly the same as another region since it usually undergoes a sequence of post-processing operations such as rotation, scaling, blurring, and noising for a better visual appearance. Therefore, it becomes increasing difficult to manually identify tampered regions even for practiced users. Accordingly, the detection of the CMF has become one of the most actively researched topics in passive image forensics.

Exhaustive search where an image and its all cyclic-shifted versions are repeatedly compared for finding duplicated regions is a straightforward way to detect the CMF [8]. However, it has extremely high computational complexity and cannot detect the CMF if the copied region has been scaled. Therefore, many CMF detection (CMFD) algorithms with reduced computational complexity have been introduced to efficiently find the duplicated regions.

In this paper, we propose a new feature descriptor, upsampled log-polar Fourier (ULPF) features, which is robust to several geometric transformations including rotation, scaling, sheering, and reflection. We first describe the theoretical background of the ULPF representation. We then present how to extract rotation and scale invariant features from the ULPF representation by exploiting properties of the Fourier transform. In addition, we analyze the common CMFD processing pipeline. Based on the analysis, we improve a part of the CMFD processing pipeline to efficiently handle various types of tampering attacks. In our simulation, we present comparative results between the proposed feature descriptor and state-of-the-art ones with proven performance guarantees.

The rest of this paper is organized as follows. In Section 2, detailed overviews of conventional CMFD algorithms and the common CMFD processing pipeline are given. In Section 3, we introduce the ULPF representation and present how to extract features from the ULPF representation. The proposed CMFD processing pipeline is illustrated in Section 4. Comparative experimental results of the proposed and conventional algorithms are presented in Section 5. Finally, our conclusions are drawn in Section 6.

2 Related works

2.1 Conventional CMFD algorithms

Up until this point, researchers have developed various techniques to efficiently find tampered regions. The first CMFD method was proposed by Fridrich et al. in 2003 [14]. This method divides an image into 8×8 overlapping blocks and extracts discrete cosine transform (DCT) features from the blocks. Feature vectors are lexicographically sorted, and then similar feature vectors are identified to judge forgery. More efficient algorithms have been introduced in the literature including blur-invariant moments [31], principal component analysis (PCA) [36], Hu moments [41], discrete wavelet transform (DWT) features [17, 32], improved DCT features [15], and segmentation-based detections [23, 39].

Recently, there was an attempt to apply the Fourier-Mellin transform (FMT) to the CMFD application. The authors in [5] proposed to extract features from image blocks using the FMT. The FMT algorithm takes the Fourier transform of each block and the resulting magnitudes are mapped into log-polar coordinates. They claimed that the FMT-based features are robust to the geometric transformation including scaling and translation. In [22], an improved algorithm was proposed by Li and Yu. The algorithm in [22] modifies the projection operation of the algorithm in [5] to achieve better rotation invariance. A log-polar Fourier (LPF) transform-based algorithm was proposed in [42]. Unlike the FMT algorithms in [5] and [22], the LPF algorithm first performs a log-polar transform and then takes the Fourier transform of the result. The similar regions are identified by computing the cross power spectrum of the LPF results. Our work is mainly motivated by the FMT and LPF algorithms in [5] and [42].

Ryu et al. [37] proposed a feature extraction algorithm using Zernike moments (ZERNIKE). Since the magnitude of Zernike moments is algebraically invariant against rotation, Ryu's method can detect a forged region even though it is largely rotated. It was reported in [11] that the ZERNIKE algorithm shows relatively good performance for detecting rotated duplicated regions. Note that all the algorithms mentioned above divide the input image into overlapping blocks and apply a feature extraction process to each block.

There exists another type of CMFD algorithms, which does not utilize block-based feature representations. These algorithms identify high-entropy regions (keypoints) in the image and extract feature vectors only at the keypoints. Therefore, the number of feature vectors is reduced and the computational complexity of the keypoint-based algorithms is relatively lower than that of the block-based algorithms. On the other hand, duplicated regions are often sparsely covered by matched pairs in the keypoint-based algortihms. This may result in the duplicated regions being completely missed. A number of keypoint-based descriptors have been widely used for image retrieval and object recognition. Among them, scale invariant feature transform (SIFT) [28] and speed up robust feature (SURF) [4] have been applied to CMFD applications.

In this work, we compare the performance of the proposed algorithm with state-of-the-art algorithms. We first choose FMT, LPF, ZERNIKE, and SIFT as comparison targets. Recently, new keypoint-based descriptors such as binary robust invariant scalable keypoints (BRISK) [21] and fast retina keypoints (FREAK) [1] have received considerable attention due to their proven efficacy. In our simulations, we implemented FREAK and applied it to the CMFD application. To the best of our knowledge, this is the first study to evaluate the performance of the FREAK descriptor for the CMFD application.

2.2 Common processing pipeline

Most CMFD algorithms follow a common processing pipeline shown in Fig. 1 [11, 12]. First, the target image is preprocessed. For example, a color image is optionally converted to gray scale. Next, the block-based methods divide the image into overlapping blocks and compute a feature vector of each block. On the other hand, the keypoint-based methods find some keypoints and compute feature vectors at the keypoints.

In the matching step, highly similar feature vectors are matched using certain methods. Since computing the similarity between all possible feature-vector pairs introduces a huge computational load, several fast matching algorithms were introduced. Among the algorithms, most researchers propose the use of lexicographic matching [25] or kd-tree matching [6] in identifying similar feature vectors. It is worthwhile to note that a recent study [18] shows that, for the block-based methods, lexicographic matching might be a better trade-off in practice when taking both accuracy and runtime into account.

In the verification step, spurious matched pairs are removed and transformation parameters are estimated using reliable pairs. Many verification schemes have been proposed. For example, threshold-based schemes [31, 37], morphological operations [20, 35], and clustering-based schemes [2, 3] were introduced. In [2], Amerini et al. proposed a scheme that builds clusters from the locations of detected features and uses random sample consensus (RANSAC) to estimate the geometric transformation between the original area and its replica. Alternatively, the same affine transformation selection (SATS) scheme in [12] finds an initial transformation using matched pairs that are spatially close to each other. Then, the SATS scheme applies region growing on the pairs with the similar transformation parameters. In [11], the authors presented that SATS provides the most reliable results in their experiments.

The last step of the processing pipeline is post-processing. In this step, the input image is geometrically transformed using the transformation parameters estimated in the verification step. The original and transformed images are overlaid to find closely matched regions. The regions that do not exhibit common behavior are removed in the post-processing step.

The proposed algorithm mainly follows the common processing pipeline. Specifically, we adopt lexicographic sorting for feature matching. Further, we improve the verification step in the common processing pipeline to efficiently handle various types of tampering attacks.

3 Proposed feature descriptor

In this section, we first present the theoretical background of the proposed ULPF representation that can be successfully used for CMFD applications. Next, we describe how to extract feature vectors from the ULPF representation. The resultant feature vectors are used in the following matching and verification steps.

Fig. 1 Common CMFD processing pipeline

3.1 Upsampled Log-Polar Fourier (ULPF) representation

Consider an $H \times V$ input image which will be divided into T overlapping circular patches of radius R, where $T = (H - 2R + 1) \cdot (V - 2R + 1)$. Suppose that a circular patch $\mathbf{c}$ and its replica $\tilde{\mathbf{c}}$ are related by scale factor s and rotation angle λ. Let $\mathbf{m} = [m, n]^T$ be a column vector indicating the pixel position in the circular patch. Further, let us denote $\mathbf{A}^{(s,\lambda)}$ be a 2×2 linear matrix, which is represented by

$$\mathbf{A}^{(s,\lambda)} = \begin{bmatrix} s\cos\lambda & s\sin\lambda \\ -s\sin\lambda & s\cos\lambda \end{bmatrix}. \tag{1}$$

Then, the relationship between the two circular patches is expressed as

$$c(\mathbf{m}) = \tilde{c}(\mathbf{A}^{(s,\lambda)}\mathbf{m}) \tag{2}$$

where $s > 0$ and $0 \leq \lambda < 2\pi$.

We separate the effects of scaling and rotation by performing a log-polar transform on the circular patches. Let $\mathbf{p} = [r, \theta]^T$ be a column vector indicating the pixel position in the log-polar coordinates. Using this notation, (2) is rewritten as

$$c(\mathbf{p}) = \tilde{c}(\mathbf{p} + \Delta) \tag{3}$$

where $r = \log\sqrt{m^2 + n^2}$, $\theta = \arctan(n/m)$, and $\Delta = [\log s, \lambda]^T$. The above equation indicates that 2-dimensional (2-D) scaling and rotation in the Cartesian coordinates can be replaced with separate 1-dimensional (1-D) translations in the log-polar coordinates.

We propose a log-polar grid sampling strategy to precisely capture the spatial information. We use an upsampled log-polar grid with $N \cdot R$ bins in both the radial and angular directions to sample the circular patch of radius R. Let $\mathbf{b}$ and $\tilde{\mathbf{b}}$ be, respectively, $NR \times NR$ matrices containing sampled pixels of $\mathbf{c}$ and $\tilde{\mathbf{c}}$ on the upsampled log-polar grid (see Fig. 2). Further, let $\mathbf{B}$ and $\tilde{\mathbf{B}}$ be the Fourier transforms of $\mathbf{b}$ and $\tilde{\mathbf{b}}$, respectively. Then, according to the Fourier shift property, $\mathbf{B}$ and $\tilde{\mathbf{B}}$ are related by

$$B(\mathbf{u}) = e^{j2\pi(\mathbf{u}^T\Delta)}\tilde{B}(\mathbf{u}) \tag{4}$$

where $\mathbf{u} = [u, v]^T$. It is natural that the magnitudes of $\mathbf{B}$ and $\tilde{\mathbf{B}}$, have the following relationship:

$$\begin{aligned} |B(\mathbf{u})| &= |e^{j2\pi(\mathbf{u}^T\Delta)}\tilde{B}(\mathbf{u})| \\ &= |\tilde{B}(\mathbf{u})| \end{aligned} \tag{5}$$

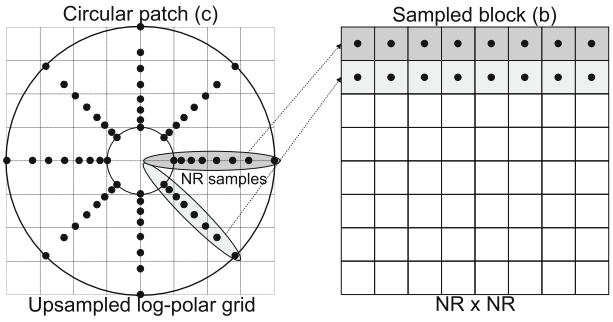

Fig. 2 Graphical explanation of $\mathbf{c}$ and $\mathbf{b}$ for the parameters $R = 4$ and $N = 2$

where $|e^{j2\pi(\mathbf{u}^T\Delta)}| = 1$. The above result forms the basis of a rotation and scaling invariant feature extraction scheme for the CMFD. Even when the copied region is rotated and scaled in the tampering attack, we clearly see that the magnitudes of $\mathbf{B}$ and $\tilde{\mathbf{B}}$ are always identical. Now, let us explain the proposed feature extraction algorithm using the ULPF representation.

3.2 Feature extraction from the ULPF representation

It is well known that high frequency components are not stable if the image suffered from signal processing operations such as JPEG compression, noise contamination, and so on [24]. Therefore, low-pass filtering can improve the detection performance in the case of a tampering attack performing signal processing operations. To exploit this feature, we apply a 3×3 Gaussian low-pass filter to the input image before the image is divided into overlapping blocks.

In addition, large flat areas (such as sky, cloud, and ocean) in the image often produce a number of false matches in the matching process [8, 37]. To deal with this issue, we calculate the standard deviation of each circular patch. Only the circular patches of which standard deviations are larger than a certain threshold α are considered in the following CMFD process. In this paper, the Greek symbols, α, β, and γ, are used to denote certain thresholds.

In the proposed algorithm, the feature vector of the circular patch consists of two parts, the feature header and the feature body. Let $\mathbf{f}$ be the feature vector of $\mathbf{c}$, which is represented by

$$\mathbf{f} = \{h, \mathbf{f}'\} \tag{6}$$

where h and $\mathbf{f}'$ are the feature header and the feature body of $\mathbf{c}$, respectively. The header h implicitly describes the entire characteristic of the current patch and the feature body $\mathbf{f}'$ is used for specifying the actual features. Note that, at the beginning of the feature extraction step, the proposed algorithm has already calculated the standard deviations of circular patches. Using the results, the feature header h is simply constructed as

$$h = \text{Round}(\sigma) \tag{7}$$

where σ is the pre-calculated standard deviation of $\mathbf{c}$ and Round$(\cdot)$ indicates a round operation.

Next, we extract features from the ULPF representation and construct $\mathbf{f}'$ using those features. It is well-known that the Fourier transform efficiently compacts the energy of an image into a few low-frequency coefficients. This implies that the similarity between two circular patches can be accurately estimated using only a few low-frequency coefficients [34, 43]. Based on this energy compaction property of the Fourier transform, the proposed algorithm rearranges the magnitudes of coefficients from low-frequency to high-frequency and uses only the first few ones. In addition, when the input samples are real numbers, the resulting Fourier transform is conjugate symmetric and the magnitude is symmetric (see Fig. 3a) [40]. This means that, for real input samples, the Fourier transform can be completely specified by about half of the coefficients. Therefore, not all of coefficients need to be considered in the feature extraction step.

Based on these observations, we propose an adaptive zigzag scanning (AZS) scheme for the Fourier transform of real input samples. Note that, in the Fourier transform, low frequency coefficients are centered at four corners. The proposed AZS starts the scanning process at the two upper corners, $(0, 0)$ and $(NR, 0)$, and the coefficients are ordered alternatively. Figure 3b shows an example of the AZS order for an 8×8 block where the

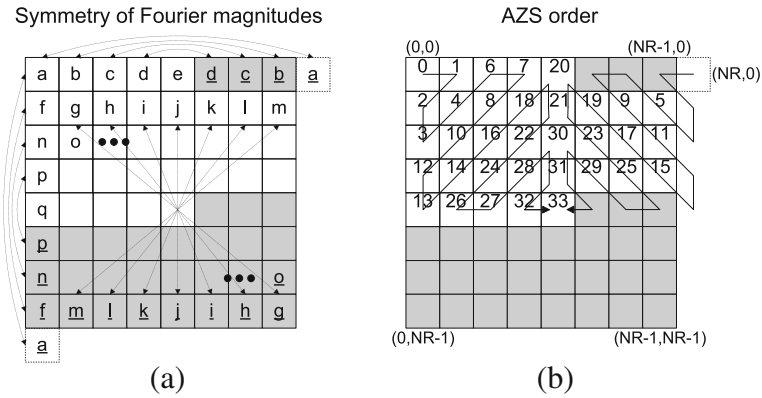

Fig. 3 Derivation of the AZS order based on the properties of the Fourier transform for an 8×8 block, where an under-bar indicates a duplicated magnitude. **a** symmetry of Fourier magnitudes. **b** AZS order

duplicated magnitudes are included only once in AZS. Let L be the total length of the proposed ULPF feature vector containing h and $\mathbf{f}'$. The proposed algorithm rearranges the magnitudes of $\mathbf{B}$ in the AZS order and forms a length-$(L-1)$ 1-D vector $\mathbf{f}'$, which is represented by

$$\mathbf{f}' = \{f'(0), f'(1), \ldots, f'(L-2)\} \tag{8}$$

where $f'(i)$, $i = 0, 1, \ldots, L-2$, is the i-th element in $\mathbf{f}'$. Here, $f'(i)$ is obtained by quantizing the magnitude of the i-th low-frequency coefficient as follows

$$f'(i) = \mathrm{Q}(|B(\mathbf{u}_i)|) \tag{9}$$

where $\mathrm{Q}(\cdot)$ is a quantization operation and $\mathbf{u}_i = [u_i, v_i]^T$ represents the i-th coefficient position in the AZS order.

It should be noted that the feature header h is used only in the lexicographic sorting process. The Euclidean distance between two feature vectors, $\mathbf{f}$ and $\tilde{\mathbf{f}}$, is calculated without using h's as follows

$$\begin{aligned} ||\mathbf{f} - \tilde{\mathbf{f}}||_2 &\triangleq ||\mathbf{f}' - \tilde{\mathbf{f}}'||_2 \\ &= \sqrt{\sum_{k=0}^{L-2} |f'(k) - \tilde{f}'(k)|^2}. \end{aligned} \tag{10}$$

Using the above measure, we determine whether the two patches are duplicated or not.

The novelty of the proposed ULPF descriptor is summarized as follows:

- The ULPF magnitudes of duplicated regions are mathematically equivalent to each other even when the copied region is rotated and scaled. On the other hand, the FMT magnitudes in [5] are varied if the copied region is scaled.
- The geometric transformation can be accurately estimated by using the upsampled log-polar grid. As mentioned, the LPF descriptor was proposed in the previous work [42]. However, the CMFD performance can be improved significantly by exploiting the upsampled grid.
- The proposed feature vector consisting of the header and the body is very useful for the CMFD.

- ✓ The feature header can improve the sorting performance by implicitly representing the entire characteristic of the circular patch.
- ✓ The AZS scheme sorting the Fourier magnitudes based on their frequencies can optimally constructs the feature body. On the contrary, the LPF algorithm in [42] uses all transform coefficients containing redundant information.

The proposed ULPF descriptor can be efficiently implemented on the parallel processor. The proposed descriptor needs to computes the ULPF representations of the circular patches of which standard deviations are larger than a certain threshold. This process incurs a large amount of computational load. Therefore, if possible, we recommend to compute the feature vectors of circular patches in a parallel manner using multi-core CPU and GPU.

4 Improved verification process

In the verification step of the common CMFD processing pipeline, the geometric transformation is estimated using reliable matched pairs of which matching distortions are lower than a predefined threshold. We observed that a variety of tampering attacks such as rotation, scaling, blurring, and noising cannot be properly handled using a single fixed threshold. For example, in the case of plain CMF, a sufficient number of pairs satisfy the given constraint. However, if the copied regions are rotated and scaled, the number of pairs satisfying the constraint is too small to detect the geometric transformation. To address this problem, we improve the verification step of the common CMFD pipeline.

Suppose that all feature vectors have been lexicographically sorted in the matching step. Let $\mathbf{f}_k$ be the k-th feature vector in the sorted list and $\tilde{\mathbf{f}}_k$ be its matched pair. In the lexicographically sorted list, $\tilde{\mathbf{f}}_k$ indicates $\mathbf{f}_{k+1}$. Let $\mathbf{x} = [x, y]^T$ be a column vector indicating the pixel position in the image coordinates. And, let $\mathbf{x}_k(\tilde{\mathbf{x}}_k)$ be the center of the circular patch $\mathbf{c}_k(\tilde{\mathbf{c}}_k)$ from which $\mathbf{f}_k(\tilde{\mathbf{f}}_k)$ is extracted. The proposed verification algorithm first constructs the distortion list $\mathbf{D}$ containing the distortions of matched pairs. The proposed algorithm selects reliable matched pairs using $\mathbf{D}$ and estimates the geometric transformation using the reliable pairs. The proposed verification step proceeds as follows.

(1) The distortion list $\mathbf{D}$ is constructed by applying the following procedure to each feature vector pair, f_k and $\tilde{f}_k$.

- **(a)** Calculate the spatial distance $z_k = ||\mathbf{x}_k - \tilde{\mathbf{x}}_k||_2$ between $\mathbf{f}_k$ and $\tilde{\mathbf{f}}_k$. If $z_k \leq \beta$, this pair is verified as a false match (spatially too close regions) and its verification process is terminated. Otherwise, if $z_k > \beta$, the algorithm goes to the next step.
- **(b)** Compute the matching distortion $d_k = ||\mathbf{f}_k - \tilde{\mathbf{f}}_k||_2$ of the current pair. The resultant d_k is inserted into the distortion list D in ascending order.

(2) Set the initialize value of the parameter w to 1. Examine the number of elements in $\mathbf{D}$, which are less than w. If the number of elements is larger than γ, the algorithm goes to the next step. Otherwise, the algorithm repeatedly increases w by 1 until the number of elements is larger than γ. The updated w will be the input of the following step.

(3) Estimate the geometric transformation between duplicated regions using the pairs satisfying $d_k \leq w$. The proposed algorithm computes the affine transformation of the pairs using the SATS algorithm as recommended in [11]. The resultant affine parameters are used in the following post-processing step.

Through the above procedure, the proposed algorithm can efficiently handle different types of tampering attacks without introducing the computational overhead. In the post-processing step, the input image is geometrically transformed using the transformation parameters estimated in the verification step. The original and transformed images are overlaid to find closely matched regions. The regions that do not exhibit common behavior are removed in the post-processing step.

5 Experimental results

We evaluated the performance of the proposed feature descriptor by comparing it with existing ones including FMT, LPF, ZERNIKE, SIFT, and FREAK. All feature descriptors were implemented using a highly efficient ANSI-C code and the performance was evaluated on an Intel i7 3.4GHz CPU with 16 GB RAM. The source codes of FMT, ZERNIKE, and SIFT are available online [45] and the FREAK descriptor was implemented based on the code of SIFT. In our implementation, the overall CMFD process was accelerated with OpenMP [9].

Basically, we measured the forgery detection performance of all descriptors using the common CMFD processing pipeline introduced in [11]. We used the lexicographic sorting in identifying similar feature vectors in the matching step. For a fair comparison, the improved verification step in Section 4 was applied to the CMFD process of all feature descriptors. In the simulations, the parameters R, N, L, α, β, and γ were set to 16, 2, 65, 4, 100, and 200, respectively.

5.1 Datasets and evaluation criteria

There exist several benchmarking CMFD datasets for evaluating the performance of feature descriptors. In this paper, we use the realistic and challenging dataset introduced in [11]. The tampered images in the dataset were manually created by skilled artists. In addition, common noise sources, such as JPEG artifacts, noise, additional scaling or rotation, are automatically included using a software framework. The dataset also provides ground truth images which are very useful for the performance evaluation. The average size of the images is about 3000×2300 pixels. In our simulations, the image *scotland* that generates the tampered region using the saturated region is excluded.

To quantitatively evaluate the detection performance, we adopt two metrics, precision M_p and recall M_r, which are calculated as [15]

$$M_p = \frac{\#\text{correctly detected pixles}}{\#\text{all detected pixels}} \tag{11}$$

and

$$M_r = \frac{\#\text{correctly detected pixles}}{\#\text{all forged pixels}}. \tag{12}$$

Hence, precision is the fraction of pixels identified as tampered that are truly tampered and recall is the fraction of tampered pixels that are correctly classified as such. A trade-off exists between precision and recall. Larger precision might decrease recall and vice versa. To consider both precision and recall, we compute their harmonic mean M_F, called F_1-score, as follows

$$M_F = \frac{2M_pM_r}{M_p + M_r}. \tag{13}$$

Using these metrics, we show how precisely the tampered regions are identified.

5.2 Performance evaluation

We evaluate the performance of the feature descriptors for four CMF scenarios: rotation, scaling, JPEG compression, and additive white Gaussian noise (AWGN). Next, the measured CMFD processing times of six descriptors are presented.

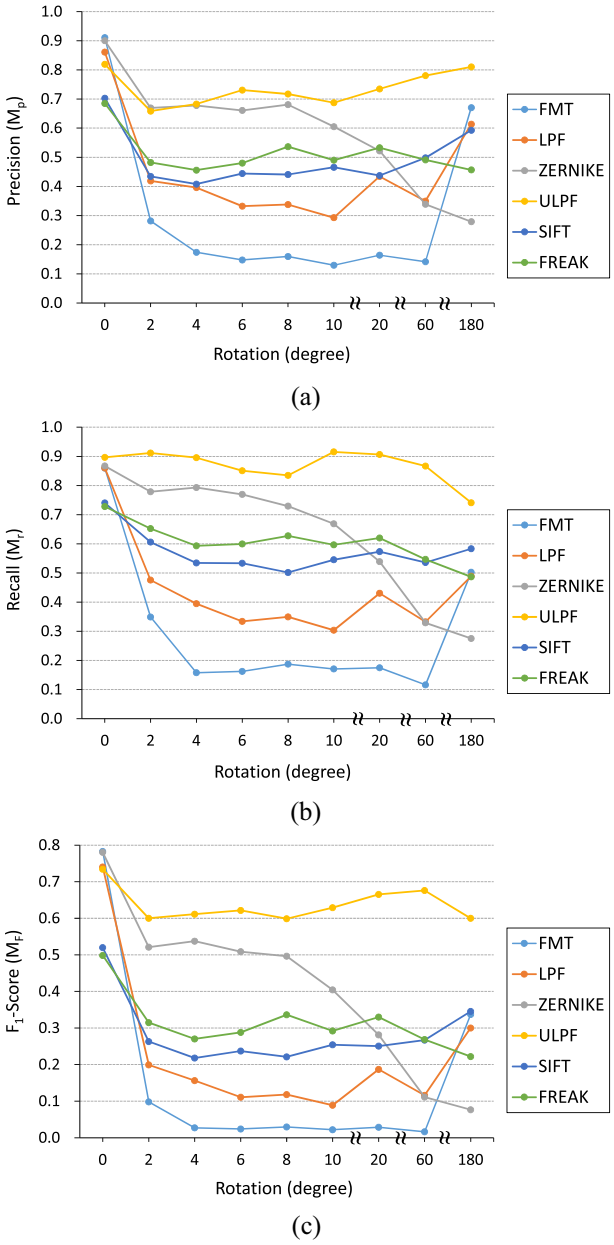

Fig. 4 Measured M_p, M_r, and M_F for the CMF with rotation

5.2.1 Rotation invariance

In this scenario, the copied regions are rotated in the range of 0° and 10° in steps of 2°. Further, we test three larger rotation angles of 20°, 60°, and 180°. Figure 4 shows the measured results for the CMF with rotation. As shown in Fig. 4, the proposed ULPF descriptor usually provides the best precision and recall over the entire range of rotation angles. Especially, ULPF achieves a significant performance improvement for large amounts of rotation as compared to the existing feature descriptors. For example, M_F of ULPF is almost double of those of FMT, LPF, and SIFT. Therefore, for the applications that need to detect the CMF with rotation, we strongly recommend the use of the ULPF descriptor. To show the result more clearly, we present the CMFD results of ULPF for the plane copy-move and the rotation of 20° in Figs. 5b and d, respectively.

In our simulation, for small amounts of rotation, ZERNIKE also shows relatively good results. Further, we observed that the keypoint-based descriptors, SIFT and FREAK, yield stable results for the CMF with rotation. For the rotation of 180°, the FMT and LPF descriptors also achieve relatively good detection performance.

5.2.2 Scale invariance

We investigate the case where the copied regions are scaled between 101 % and 109 % of its original size in increments of 2 % as well as 120 % and 200 %. Figure 6 presents the results

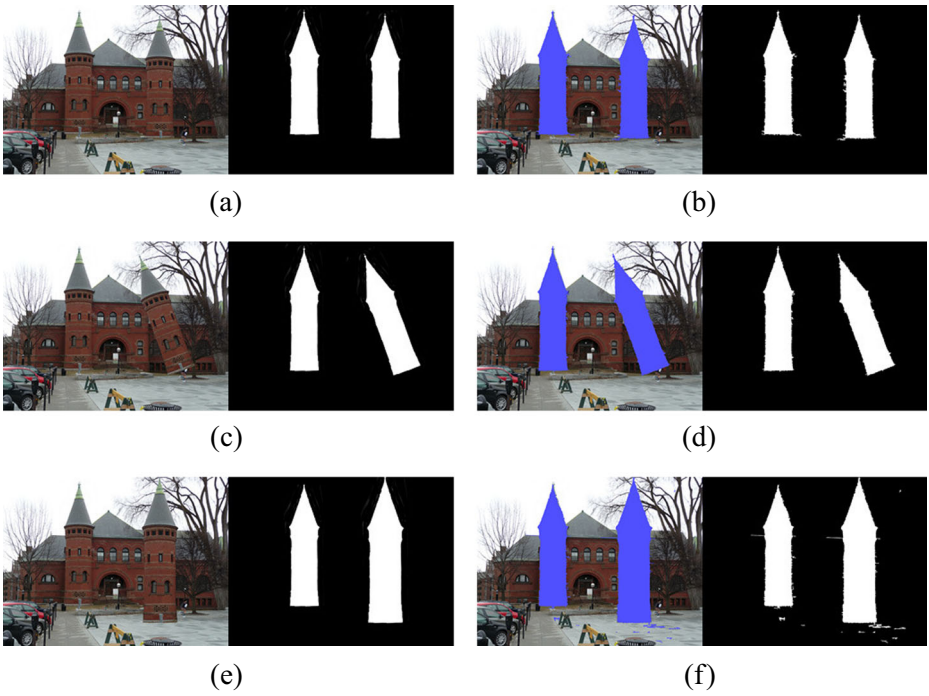

Fig. 5 Three test cases and their CMFD results using the ULPF descriptor. **a** plane CMF. **b** CMFD result of (**a**). **c** CMF with the rotation of 20°. **d** CMFD result of (**c**). **e** CMF with the scaling of 120 %. **f** CMFD result of (**f**)

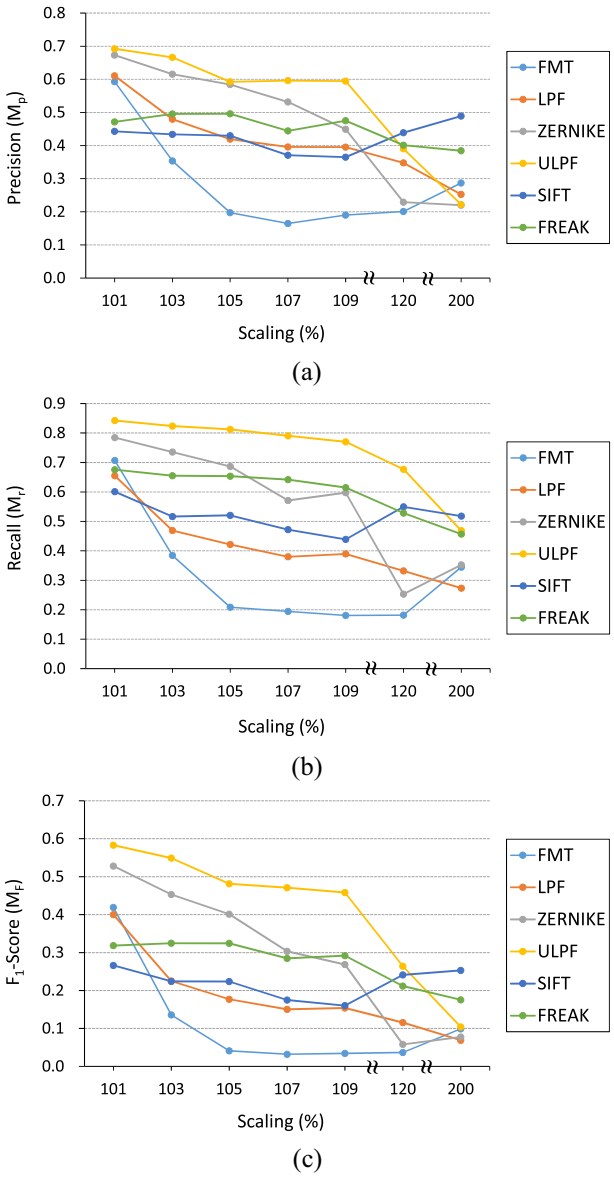

Fig. 6 Measured M_p, M_r, and M_F for the CMF with scaling

for the CMF with scaling. As compared to the CMFD of rotation, most features show a relatively weak invariance. The proposed ULPF tends to exhibit the best scale invariance in the experiments. However, for the scaling of 200 %, SIFT achieves the highest M_F among the descriptors. In general, if the scale factor is very large, the keypoint-based descriptors perform better than the block-based ones. In summary, the proposed ULPF can be used to handle a moderate amount of scaling which is often the case in real-world CMF manipulations. However, the detection performance of ULPF decreases sharply as the scale factor

increases. Therefore, the keypoint-based descriptors are the better choice for relatively large scale factors. Figure 5f shows the CMFD result of ULPF for the scaling of 120 %.

5.2.3 *Robustness to JPEG compression artifacts*

Robustness to JPEG compression artifacts is investigated. The quality factor of JPEG is varied between 100 and 20 in steps of 10. In general, ULPF and ZERNIKE outperform the other methods. As shown in Fig. 7, M_F's of the two methods moderately decrease as the

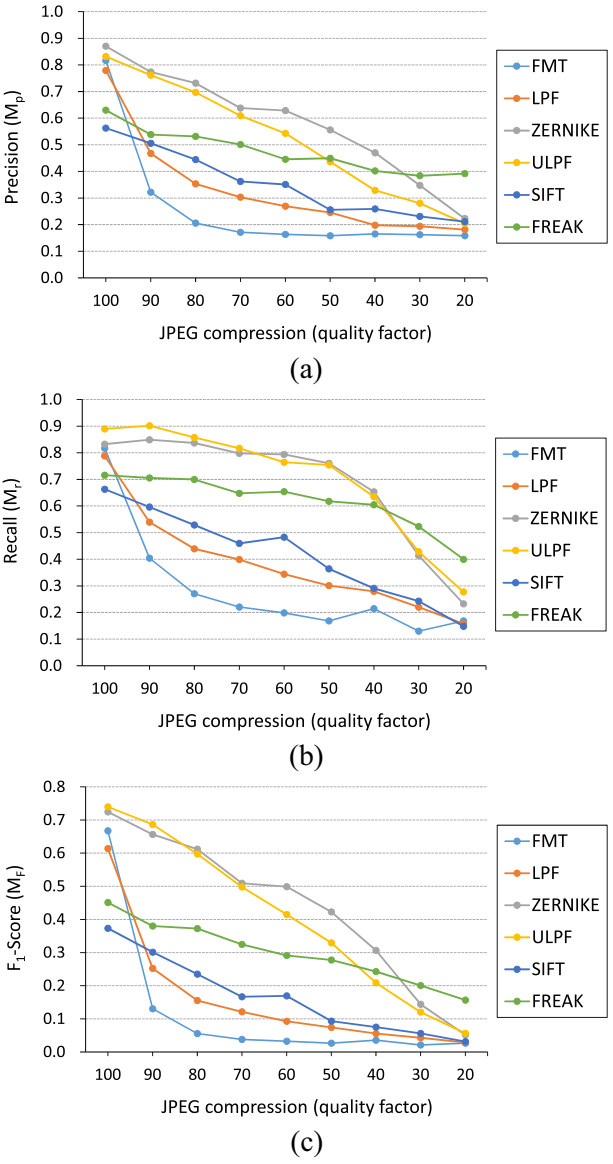

Fig. 7 Robustness to JPEG compression artifacts

quality factor decreases. For the high quality factors 100 and 90, M_F of ULPF is slightly higher than that of ZERNIKE. Further, for the quality factors 80 and 70, M_F of ULPF is almost the same as that of ZERNIKE. FREAK is the best feature for very low quality factors 30 and 20. It is worthwhile to note that the quality factor is usually equal to or larger than 70

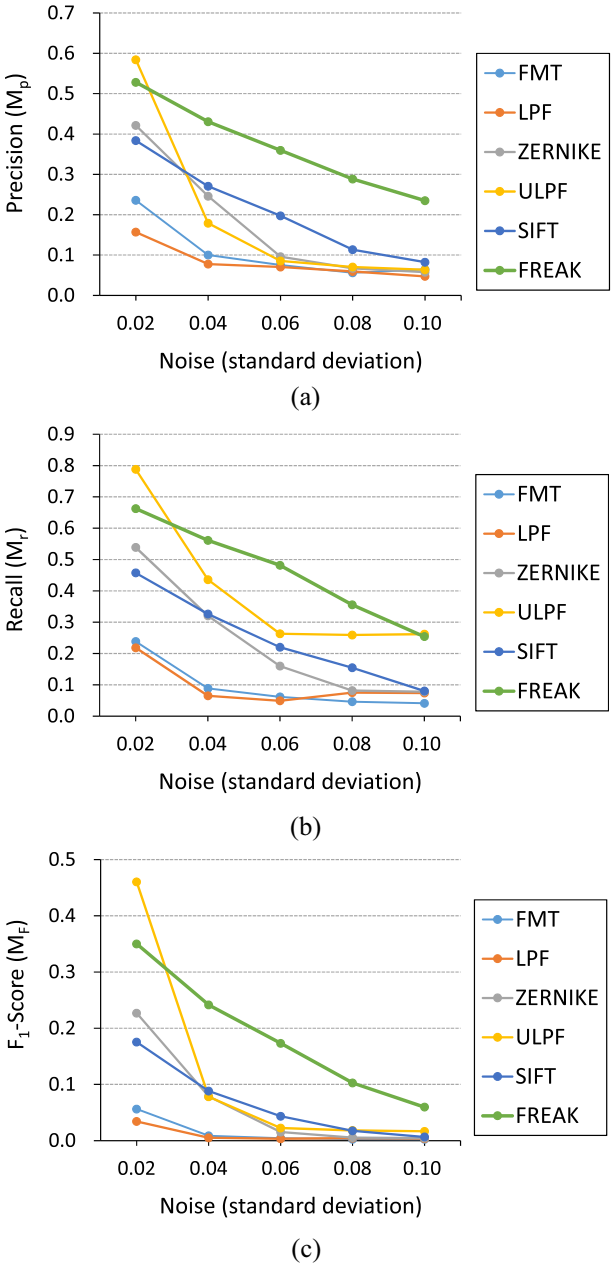

Fig. 8 Robustness to Gaussian noise

for real-world forgeries. In our setup, the FMT and LPF descriptors yield a weak robustness to JPEG compression artifacts.

5.2.4 Robustness to Gaussian noise

We also evaluate the robustness of all feature descriptors to AWGN. We normalize the image intensities between 0 and 1, and added zero-mean Gaussian noise with standard deviations of 0.02, 0.04, 0.06, 0.08, and 0.10 to tampered regions. In Fig. 8, we clearly see that the detection performance of all descriptors sharply decreases as the standard deviation increases. When the standard deviation is equal to 0.02, the detection performance of ULPF is the highest. However, if the standard deviation is equal to or larger than 0.04, FREAK achieves higher M_F than the other algorithms. Therefore, it can be seen that the keypoint-based methods tend to show a more stable performance than the block-based ones in terms of robustness to Gaussian noise.

5.2.5 Computational complexity

The CMFD processing time of a descriptor varies depending on the complexity of its feature vector and the number of feature vectors. The measured processing times are listed in Table 1. As shown in Table 1, our implementation is highly optimized in terms of the processing time. In our simulations, FMT requires the highest processing time for extracting feature vectors. Accordingly, the total processing time of FMT is the highest among all the methods. We see that ZERNIKE also yields a relatively high processing time for feature extraction. However, the matching and verification time of ZERNIKE is relatively lower than the other block-based methods. This is because the feature length of ZERNIKE is shorter than those of the other block-based methods. The total processing time of ULPF is lower than those of FMT and ZERNIKE but it is higher than that of LPF. In our simulation, the total processing time of LPF is the lowest among the block-based algorithms.

As we expected, the processing times of the keypoint-based methods are much lower than those of the block-based methods. This is because the number of feature vectors of the keypoint-based methods are much smaller than those of the block-based methods. In our simulations, FREAK achieves the lowest processing time.

5.2.6 Detailed performance analysis

In order to provide more insight into the simulation results, we measured the performances of the upsampled log-polar grid, feature header, AZS, and improved verification schemes

Table 1 Average CMFD processing time (s)

Descriptors	Extraction	Matching & Verification	Post-Processing	Total
FMT	47.79	8.89	10.09	66.77
LPF	10.61	7.89	9.94	28.43
ZERNIKE	29.45	6.71	11.12	47.28
ULPF	19.60	8.67	12.91	41.18
SIFT	5.26	0.35	9.82	15.42
FREAK	2.29	0.19	9.30	11.78

Table 2 Detailed Performance Analysis (Variation of M_F)

Schemes	CMF scenario		
	Rotation	Scaling	Average
Feature header	3.79	2.23	3.01
AZS	0.85	4.83	2.84
Upsampled log-polar grid	9.85	0.01	4.38
Improved verification	12.19	11.86	12.02

separately. We evaluated the effectiveness of each scheme by excluding the scheme from the CMFD processing pipeline. In detail, we measured the average change of M_F by excluding each scheme. The measured results are listed in Table 2.

As shown in Table 2, all schemes can improve the detection performance. Especially, the improved verification scheme introduced in Section 4 yields a significant performance enhancement for both the CMF with rotation and scaling. Further, we can see that AZS achieves a relatively high performance improvement for the scaling and the upsampled log-polar grid shows a high performance improvement for the rotation. The scheme that inserts the feature header into the feature vector consistently enhances the detection performance for both the CMF with rotation and scaling.

6 Conclusions

A new feature descriptor was presented for the efficient detection of CMF. The proposed ULPF descriptor has a solid theoretical background and its actual performance is superior than existing descriptors. Especially, the proposed descriptor achieves a very stable detection performance over the entire range of rotation angles. In addition, the proposed feature vector structure and AZS order can be utilized in a wide range of applications dealing with images in the Fourier domain.

References

1. Alahi A, Ortiz R, Vandergheynst P (2012) FREAK: Fast Retina keypoint, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 16–21
2. Amerini I, Ballan L, Caldelli R, Bimbo AD, Serra G (2011) A SIFT-based forensic method for copy-move attack detection and transformation recovery. IEEE Trans Inf Forensics Secur 6(3):1099–1110
3. Amerini I, Ballan L, Caldelli R, Bimbo AD, Tongo LD, Serra G (2013) Copy-move forgery detection and localization by means of robust clustering with J-Linkage. Signal Process Image Commun 28(6):659–669
4. Bay H, Ess A, Tuytelaars T, Gool LV (2008) SURF: Speeded Up robust features. Comput Vis Image Understand 110(3):346–359
5. Bayram S, Sencar HT, Memon N (2009) An efficient and robust method for detecting copy-move forgery, IEEE International Conference on Acoustics, Speech and Signal Processing, 1053–1056
6. Beis JS, Lowe DG (1997) Shape indexing using approximate Nearest-Neighbour search in High-Dimensional spaces, IEEE Conference on Computer Vision and Pattern Recognition, 1000–1006
7. Birajdar GK, Mankar VH (2013) Digital image forgery detection using passive techniques: a survey. Digit Investig 10(3):226–245
8. Cao Y, Gao T, Fan L, Yang Q (2012) A robust detection algorithm for copy-move forgery in digital images. Forensic Sci Int 214(1):33–43
9. Chapman B, Jost G, Pas RVD (2008) Using openMP: portable shared memory parallel programming MIT press

10. Chen C, Ni J, Huang J (2013) Blind detection of median filtering in digital images: a difference domain based approach. IEEE Trans Image Process 22(2):4699–4710
11. Christlein V, Riess C, Jordan J, Riess C, Angelopoulou E (2012) An evaluation of popular copy-move forgery detection approaches. IEEE Trans Inf Forensics Secur 7(6):1841–1854
12. Christlein V, Riess C, Angelopoulou E (2010) On rotation invariance in copy-move forgery detection, IEEE International Workshop on Information Forensics and Security, 1–6
13. Farid H (2009) Image forgery detection. IEEE Signal Proc Mag 26(2):16–25
14. Fridrich AJ, Soukal BD, Lukas A. J (2003) Detection of copy-move forgery in digital images, Digital Forensic Research Workshop
15. Huang Y, Lu W, Sun W, Long D (2011) Improved DCT-based detection of copy-move forgery in images. Forensic Sci Int 206(1):178–184
16. Kang X, Wei S (2008) Identifying tampered regions using singular value decomposition in digital image forensics. Int Conf Comput Sci Soft Eng 3:926–930
17. Khan Er. S, Kulkarni Er. A (2010) An efficient method for detection of copy-move forgery using discrete wavelet transform. Int J Comput Sci Eng 2(5):1801–1806
18. Kirchner M, Schttle P, Riess C (2015) Thinking beyond the block: block matching for copy-move forgery detection revisited, Media Watermarking, Security, and Forensics
19. Kwon GR, Lama RK, Pyun JY, Park CS (2015) Multimedia digital rights management based on selective encryption for flexible business model, Multimedia Tools and Applications. doi:10.1007/s11042-015-2563-z
20. Langille A, Gong M (2006) An Efficient Match-based Duplication Detection Algorithm, Canadian Conference on Computer and Robot Vision, 64–71
21. Leutenegger S, Stefan M (2011) BRISK: Binary Robust invariant scalable keypoints, IEEE International Conference on Computer Vision (ICCV), 2548–2555
22. Li W, Yu N (2010) Rotation robust detection of copy-move forgery, IEEE International Conference on Image Processing, 2113–2116
23. Li J, Li X, Yang B, Sun X (2015) Segmentation-based image copy-move forgery detection scheme. IEEE Trans Inf Forensics Secur 10(3):507–518
24. Li L, Li S, Zhu H, Chu SC, Roddick JF, Pan JS (2013) An efficient scheme for detecting copy-move forged images by local binary patterns. J Inf Hiding and Multimedia Signal Process 4(1):46–56
25. Lin H, Wang C, Kao Y (2009) Fast copy-move forgery detection. WSEAS Trans Signal Process 5(5):188–197
26. Lin HJ, Wang CW, Kao YT (2009) Fast copy-move forgery detection. WSEAS Trans Signal Process 5(5):188–197
27. Liu W, Zhang H, Tao D, Wang Y, Lu K (2016) Large-scale paralleled sparse principal component analysis. Multimedia Tools Appl 75(3):1481–1493
28. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60(2):91–110
29. Luo Y, Tao D, Geng B, Xu C, Maybank S (2013) Manifold regularized multitask learning for semi-supervised multilabel image classification. IEEE Trans Image Process 22(2):523–536
30. Luo Y, Wen Y, Tao D, Gui J, Xu C (2016) Large margin Multi-Modal Multi-Task feature extraction for image classification. IEEE Trans Image Process 25(1):414–427
31. Mahdian B, Saic S (2007) Detection of copy-move forgery using a method based on blur moment invariants. Forensic Sci Int 171(2):180–189
32. Muhammad G, Hussain M, Bebis G (2012) Passive copy move image forgery detection using undecimated dyadic wavelet transform. Digit Investig 9(1):49–57
33. Murali S, Chittapur GB, S 1 PH, Anami BS (2012) Comparison and analysis of photo image forgery detection techniques. Int J Comput Sci & Appl 2(6):45–56
34. Oppenheim AV, Schafer RW, Buck JR (1999) Discrete-time signal processing, 2nd. Prentice-Hall, Englewood-Cliffs, NJ
35. Pan X, Lyu S (2010) Region duplication detection using image feature matching. IEEE Trans Inf Forensics Secur 5(4):857–867
36. Popescu A, Farid H (2004) Exposing digital forgeries by detecting duplicated image regions, Department of Computer Science, Dartmouth College, Tech. Rep. TR2004-515
37. Ryu S, Lee M, Lee H (2010) Detection of copy-rotate-move forgery using Zernike moments. Lect Notes Comput Sci 6387:51–65
38. Schneider M, Chang S (1996) A robust content based digital signature for image authentication. Int Conf Image Process (ICIP) 3:227–230
39. Sekhar R, Shaji RS (2016) A study on segmentation-based copy-move forgery detection using DAISY descriptor, Proceedings of the International Conference on Soft Computing Systems, 223–233

40. Sorensen HV, etal (1987) Real-valued fast fourier transform algorithms. IEEE Trans Acoust Speech Signal Process 35(6):849–863
41. Wang J, Liu G, Zhang Z, Dai Y, Wang Z (2009) Fast and robust forensics for image Region-Duplication forgery. Acta Automatica Sinica 35(12):1488–1495
42. Wu Q, Wang S, Zhang X (2010) Detection of image region-duplication with rotation and scaling tolerance. Lect Notes Comput Sci 6421:100–108
43. Yaroslavsky L, Ye Wang DFT (2000) DCT, MDCT, DST and signal fourier spectrum analysis, Signal Processing Conference, 1–4
44. Yeung M, Mintzer F (1997) An Invisible Watermarking Technique for Image Verification, International Conference on Image Processing (ICIP), vol. 2, pp 690–683
45. http://www5.cs.fau.de/

Chun-Su Park received the B.S. and Ph.D. degrees in electrical engineering from Korea University, Seoul, in 2003 and 2009, respectively. From 2009 to 2010, he was a visiting scholar with the Signal and Image Processing Institute, University of Southern California, Los Angeles. He was a senior research engineer at Samsung Electronics from 2010 to 2012. From 2012 to 2014, he was an assistant professor with Dep. of Info. and Telecom. Eng. at Sangmyung University. In 2014, he joined Dep. of Software at Sejong University where he is currently an assistant professor. His research interests are in the areas of video signal processing, parallel computing, and multimedia communications.

Changjae Kim received the BS and MS degrees in civil engineering (urban engineering major) from Seoul National University, Seoul,Korea, in 1998 and 2000, respectively. He received a PhD degree at the Department of Geomatics Engineering, University of Calgary, Canada. He currently works for Myongji University, Korea as an assistant professor. His present research interests are image processing and the sensor fusion of LIDAR and photogrammetry.

Jihoon Lee received his B.S., M.S, and Ph.D. degrees in Electronics engineering from Korea University, Seoul, Korea in 1996, 1998, and 2001, respectively. From 2002 to 2011, he had worked at Samsung Electronics as a senior research member. He is currently an assistant professor at the Department of Information and Telecommunication engineering, Sangmyung University. His research interests include information centric networking, context-aware networking, secure vehicle communication, and network security.

Goo-Rak Kwon received the Ph.D. degree at the Department of Mechatronic Engineering of Korea University in 2007 and the M.S. degree in the School of Electrical and Computer Engineering at the SungKyunKwan University in 1999. He has also served as Chief Executive Officer and Director of Dalitech Co. Ltd. from May 2004 to Feb. 2007. He joined the Department of Electronic Engineering at Korea University where he was a Postdoc supporting the BK21 Information Technique Business from Mar. 2007 to Feb. 2008. At present, he is working an associate professor at Chosun University. He has contributed 73 articles to journals and conference proceedings. He also holds 12 patents on security of multimedia contents for digital rights management. He was a member in the IEEE, IEICE, and IS&T in the international institute. In the domestic institute, he had a member of signal processing society in the IEEK, KMMS, KIPS, and KICS. His interest research fields are A/V signal processing, video communication, and applications.