A SIFT-based forensic method for copy-move attack detection and transformation recovery

Irene Amerini, Lamberto Ballan, *Student Member, IEEE*, Roberto Caldelli, *Member, IEEE*, Alberto Del Bimbo, *Member, IEEE*, and Giuseppe Serra

***Abstract*—One of the principal problems in image forensics is determining if a particular image is authentic or not. This can be a crucial task when images are used as basic evidence to influence judgment like, for example, in a court of law. To carry out such forensic analysis, various technological instruments have been developed in the literature. In this paper the problem of detecting if an image has been forged is investigated; in particular, attention has been paid to the case in which an area of an image is copied and then pasted onto another zone to create a duplication or to cancel something that was awkward. Generally, to adapt the image patch to the new context a geometric transformation is needed. To detect such modifications, a novel methodology based on Scale Invariant Features Transform (SIFT) is proposed. Such a method allows both to understand if a copy-move attack has occurred and, furthermore, to recover the geometric transformation used to perform cloning. Extensive experimental results are presented to confirm that the technique is able to precisely individuate the altered area and, in addition, to estimate the geometric transformation parameters with high reliability. The method also deals with multiple cloning.**

***Index Terms*—Digital image forensics, copy-move attack, authenticity verification, geometric transformation recovery.**

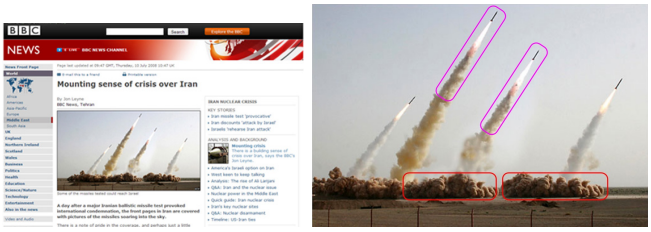

Fig. 1. An example of image tampering that appeared in press in July, 2008. The feigned image (on the right) shows four Iranian missiles but only three of them are real; two different sections (encircled in red and purple, respectively) replicate other image sections by applying a copy-move attack.

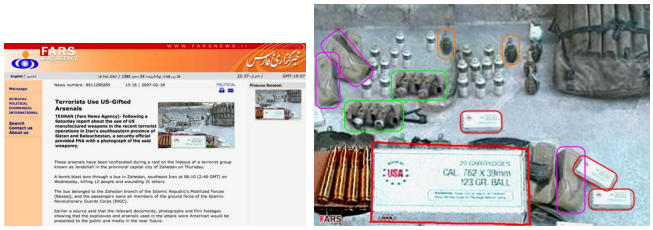

Fig. 2. A close look at this picture, appearing in press in 2007 (Fars News Agency, Tehran), shows that many elements are cloned over and over. Also in this case the cloned sections are encircled in different colors.

I. Introduction

DIGITAL crime, together with constantly emerging software technologies, is growing at a rate that far surpasses defensive measures. Sometimes a digital image or a video are incontrovertible evidence of a crime or the proof of a malevolent action. By looking at a digital content as a digital clue, *multimedia forensics* aims to introduce novel methodologies to support clue analysis and to provide an aid for making a decision about a crime. Multimedia forensics [1], [2], [3] deals with developing technological instruments operating in the absence of watermarks [4], [5] or signatures inserted in the image. In fact, different from digital watermarking, forensics means are defined as "passive" because they can formulate an assessment on a digital document by resorting only to the digital asset itself. These techniques basically allow the user to determine if particular content has been tampered with [6], [7] or which was the acquisition device used [8], [9]. In particular, by focusing on the task of acquisition device identification, two main aspects must be studied: the first is to understand which kind of device generated a digital image (e.g. a scanner, a digital camera or is a computer graphics product) [10], [11], while the second is to determine which specific camera or scanner (by recognizing model and brand) acquired that specific content [8], [9].

The other main multimedia forensics topic is image tampering detection [6], that is assessing the authenticity of a digital image. Information integrity is fundamental in a trial, but it is clear that the advent of digital pictures and relative ease of digital image processing today makes this authenticity uncertain. Two examples of this problem, that recently appeared in newspapers and TV news, are given in Fig. 1 and Fig. 2. Modifying a digital image to change the meaning of what is represented in it can be crucial when used in a court of law where images are presented as basic evidence to influence the judgement. Furthermore, it is interesting, once established that something has been manipulated, to understand exactly what happened: if an object or a person has been covered, if a part of the image has been cloned, if something has been copied from another image, or if a combination of these processes has been

Manuscript received September 14, 2010; revised March 09, 2011; accepted March 10, 2011. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Wenjun Zeng.

The authors are with Media Integration and Communication Center, University of Florence, 50134 Florence, Italy (e-mail: irene.amerini@unifi.it; ballan@dsi.unifi.it; roberto.caldelli@unifi.it; delbimbo@dsi.unifi.it; serra@dsi.unifi.it). This work was partially supported by the EU ICT 3D-COFORM Project (Contract FP7-231809).

Digital Object Identier 10.1109/TIFS.2011.2129512

carried out. In particular, when an attacker creates his feigned image by cloning an area of the image onto another zone (*copy-move* attack), he is often obliged to apply a geometric transformation to satisfactorily achieve his aim.

In this paper this issue is investigated, and the proposed method is able to individuate if copy-move tampering has taken place and also to estimate the parameters of the transformation used (i.e. horizontal and vertical translation, scaling factors, rotation angle). On the basis of our preliminary work [12], a new methodology which satisfies these requirements is presented hereafter. Such a technique is based on the Scale Invariant Features Transform (SIFT) [13], which are used to robustly detect and describe clusters of points belonging to cloned areas. After detection, these points are exploited to reconstruct the parameters of the geometric transformation. The proposed technique has also been tested against *splicing* attacks (i.e. when an image block is duplicated onto a different image). In fact, in a context where the source image is available (e.g. the forensic analyst has to check a suspect dataset which contains both the source and the destination image) this methodology can be still applied.

The rest of the paper is structured as follows: Section II presents related works regarding copy-move forgery detection and it reviews the SIFT technique. Moreover, the contribution and the novelty of our approach respect to the state-of-the-art is discussed. Section III presents the proposed method in its three main stages, while experimental results on forgery detection and on applied transformation parameters estimation are presented in Section IV. Conclusions are finally drawn in Section V.

II. SIFT Features for Image Forensics

One of the most common image manipulations is to clone (copy and paste) portions of the image, for instance, to conceal a person or an object in the pictured scene. When this is done with care, and retouching tools are used, it can be very difficult to detect cloning. Moreover, since the copied parts are from the same images, some components (e.g noise and color) will be compatible with the rest of the image and thus will not be detectable using methods that look for incompatibilities in statistical measures in different parts of the image [14], [15]. Furthermore, since the cloned regions can be of any shape and location, it is computationally infeasible to search all possible image locations and sizes with an exhaustive search as pointed out in [16].

The problem of copy-move forgery detection has been faced by proposing different approaches each of these based on the same concept: a copy-move forgery introduces a *correlation* between the original image area and the pasted one. Several methods search for this dependence by dividing the image into overlapping blocks and then applying a feature extraction process in order to represent the image blocks with a low dimensional representation. In [17] the averages of red, green and blue components are chosen together with four other features computed on overlapping blocks, and obtained by calculating the energy distribution of luminance along four different directions. A different approach is presented in [18] in which the features are represented by the Singular Value Decomposition (SVD) performed on low-frequency coefficients of the block-based Discrete Wavelet Transform (DWT). The authors in [19] proposed a block representation calculated using blur invariants. Their specific aim is to find features invariant to the presence of blur artifacts that a falsifier can apply to make detection of forgery more difficult. Then they used Principal Component Analysis (PCA) to reduce the number of features and a k-tree to identify the interesting regions. In [20] the authors present a technique to detect cloning when the copied part has been modified using two specific tools, the *Adobe Photoshop* healing brush and Poisson cloning. Another two algorithms [16] and [21] are based on low dimensional representations of blocks and fast sorting to improve efficiency have been developed to detect copy-move image regions. In particular, the authors in [16] apply a Discrete Cosine Transform (DCT) to each block. Duplicated regions are then detected by lexicographically sorting the DCT block coefficients and grouping similar blocks with the same spatial offset in the image. In [21] the authors apply PCA on image blocks to yield a reduced-dimension representation. Duplicated regions are again detected by lexicographically sorting and grouping all of the image blocks. A related approach is the method in [22] where a Fourier Mellin Transform is applied on each block. Forgery decision is performed when there are more than a given number of blocks that are connected to each other and the distance between block pairs is the same. To create a convincing forgery, it is often necessary to resize, rotate, or stretch portions of an image. For example, when creating a composition of two objects, one object may have to be resized to match the relative heights. This process requires re-sampling of the original image introducing specific periodic correlations between neighboring pixels. The presence of these correlations due to re-sampling can be used to detect that something happened to the image [23] but not to detect the specific manipulation.

So a good copy-move forgery detection should be robust to some types of transformation, such as rotation and scaling, and also to some manipulations including JPEG compression, Gaussian noise addition and gamma correction. Most existing methods do not deal with all these manipulations and are often computationally prohibitive. In particular the method in [21] is not able to detect scaling or rotation transformation, whereas with the methods in [16] and [22] only small variations in rotation and scaling are identifiable as reported in [24]. The authors in [25] make an attempt to overcome the problem using Zernike moments to identify copy-move manipulation when only rotation of the copied area takes place. This issue is also discussed in [26] where rotation transformations, JPEG compression and Gaussian noise manipulations are analyzed to understand how they affect the copy-move detection. The authors in [27] instead propose a method to detect duplicated and transformed regions through the use of a block description invariant to reflection and rotation such as the log-polar block representation summed along its angle axis. Finally, a comparison among some of the copy-move methods described above has been reported in [28], evaluating the performance of each method with and without geometric transformation

applied to the copied patch.

Nowadays, local visual features (e.g SIFT, SURF, GLOH, etc.) have been widely used for image retrieval and object recognition, due to their robustness to several geometrical transformations (such as rotation and scaling), occlusions and clutter. More recently, attempts have been made to apply these kinds of features also in the digital forensics domain; in fact, SIFT features have been used for fingerprint detection [29], shoeprint image retrieval [30], and also for copy-move detection [31], [12], [32].

A. Review of the SIFT algorithm

Most of the algorithms proposed in the literature for detecting and describing local visual features usually requires two steps. The first is the detection step, in which interest points are localized, while in the second step robust local descriptors are built so as to be invariant with respect to orientation, scale and affine transformations. A comprehensive analysis of several local descriptors is provided in [33], while local affine region detectors are surveyed in [34]. These works confirm that SIFT feaures [13] are a good solution because of their robust performance and relatively low computational costs.

This method can be roughly summerized as the following four steps: *i)* scale-space extrema detection; *ii)* keypoint localization; *iii)* assignment of one (or more) canonical orientations; *iv)* generation of keypoint descriptors.

In other words, given an input image I, SIFT features are detected at different scales using a scale-space representation implemented as an image pyramid. The pyramid levels are obtained by Gaussian smoothing and sub-sampling of the image resolution while interest points are selected as local extrema (min/max) in the scale-space. These keypoints, referred to as $\mathbf{x}_i$ in the following, are extracted by applying a computable approximation of the Laplacian of Gaussian called Difference of Gaussians (DoG). Specifically, a DoG image D is given by: $D(x,y,\sigma) = (G(x,y,k\sigma) - G(x,y,\sigma)) * I(x,y) = L(x,y,k\sigma) - L(x,y,\sigma)$, where $L(x,y,k\sigma)$ is the convolution of the original image $I(x,y)$ with the Gaussian blur $G(x,y,k\sigma)$ at scale $k\sigma$.

In order to guarantee invariance to rotations, the algorithm assigns to each keypoint a canonical orientation o. To determine this orientation, a gradient orientation histogram is computed in the neighborhood of the keypoint. Specifically, for an image sample $L(x,y,\sigma)$ at scale σ (the scale in which that keypoint was detected), the gradient magnitude $m(x,y)$ and orientation $\theta(x,y)$ are precomputed using pixel differences:

$$m(x,y) = \Big(\big((L(x+1,y) - L(x-1,y)\big)^2 + \\ + \big((L(x,y+1) - L(x,y-1)\big)^2 \Big)^{1/2}, \quad (1)$$

$$\theta(x,y) = tan^{-1}\left(\frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)}\right). \quad (2)$$

An orientation histogram with 36 bins is formed, with each bin covering approximately 10 degrees. Each sample in the neighboring window added to a histogram bin is weighted by its gradient magnitude and by a Gaussian-weighted circular window with σ equal to 1.5 times respect to the scale of the keypoint. The peaks in this histogram correspond to dominant orientations. Once these keypoints are detected, and canonical orientations are assigned, SIFT descriptors are computed at their locations in both image plane and scale-space. Each feature descriptor consists of a histogram $\mathbf{f}$ of 128 elements, obtained from a 16×16 pixel area around the corresponding keypoint. This area is selected using the coordinates (x,y) of the keypoint as the center and its canonical orientation as the origin axis. The contribution of each pixel is obtained by accumulating image gradient magnitude, $m(x,y)$, and orientation, $\theta(x,y)$, in scale-space and the histogram is computed as the local statistics of gradient orientations (considering 8 bins) in 4×4 sub-patches.

Summarizing the above, given an image *I*, this procedure ends with a list of N keypoints each of which is completely described by the following information: $\mathbf{x}_i = \{x, y, \sigma, o, \mathbf{f}\}$, where (x,y) are the coordinates in the image plane, σ is the scale of the keypoint (related to the level of the image-pyramid used to compute the descriptor), o is the canonical orientation (used to achieve rotation invariance) and $\mathbf{f}$ is the final SIFT descriptor.

B. Our contribution

A very preliminary work on copy-move forgery detection based on SIFT features was proposed in [31], but in that paper no estimation of the parameters of the applied geometric transformation is performed and, furthermore, extended numerical results to evaluate real performances of the methodology (e.g. True/False Positive Rates) are not provided. Another very recent work has been presented in [32]. Although the technique is able to deal with region extraction by resorting to a correlation map, it can not manage affine transformation and, also in this case, quantitative results on the reliability of the estimate of geometric transformation parameters are not given; in addition, this approach adopts many different empirical thresholds whose setting seems to be not completely unsupervised. Moreover none of these contributions considers accurately the case of multiple copy-move forgeries. As we will show furthermore, this is a key point in a realistic forensic scenario since often a forged image contains several cloned areas (like in the case of Fig. 2).

In this scenario is placed our proposed method that is able to detect and then to estimate the geometrical transformation used in a copy-move forgery attack. Multiple copy-move forgeries are managed by performing a robust feature matching procedure and then a clustering on the keypoint coordinates in order to separate the different cloned areas. These two tasks are fundamental since otherwise, in case of multiple cloning, it is often impossible to detect and separate each forgery and also to estimate the geometric transformation. Estimating the geometric parameters with accuracy is deemed as a fundamental task not only to understand how the cloned patch has been processed [35] and possibly to infer which was the counterfeiter's motive, but also to compare the original source block of image and the forged one on a common ground. Furthermore, a reliable estimate of the transformation allows

us to register the two patches for a possible deeper forensics analysis [36]. The method proposed hereafter is able to deal with affine geometric transformations and, as demonstrated by experimental results, also gives reliable estimates of the transformation parameters. Our technique works by relying on a unique empirical threshold which regulates clustering operation, and that has been determined by a training procedure on a general dataset. This is a very important issue also in comparison with similar techniques like that in [32].

III. The proposed method

The proposed approach is based on the SIFT algorithm to extract robust features which can allows it to discover if a part of an image was copy-moved and furthermore which geometrical transformation was applied. In fact, the copied part has basically the same appearance of the original one, thus keypoints extracted in the forged region will be quite similar to the original ones. Therefore, matching among SIFT features can be adopted for the task of determining possible tampering. A simple schematization of the whole system is shown in Fig. 3: the first step consists of SIFT feature extraction and keypoint matching, the second step is devoted to keypoint clustering and forgery detection, while the third one estimates the occurred geometric transformation, if tampering has been detected.

A. SIFT features extraction and multiple keypoint matching

Given a test image, a set of keypoints $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ with their corresponding SIFT descriptors $\{\mathbf{f}_1, \ldots, \mathbf{f}_n\}$ is extracted. A matching operation is performed in the SIFT space among the $\mathbf{f}_i$ vectors of each keypoint to identify similar local patches in the test image. The best candidate match for each keypoint $\mathbf{x}_i$ is found by identifying its nearest neighbor from all the other $(n-1)$ keypoints of the image, which is the keypoint with the minimum Euclidean distance in the SIFT space. In order to decide that two keypoints match (i.e. "are these two descriptors the same or not?"), simply evaluating the distance between two descriptors with respect to a global threshold does not perform well. This is due to the high-dimensionality of the feature space (128) in which some descriptors are much more discriminative than others.

We can obtain a more effective procedure, as suggested in [13], by using the ratio between the distance of the closest neighbor to that of the second-closest one, and comparing it with a threshold T (often fixed to 0.6). For the sake of clarity, given a keypoint we define a similarity vector $\mathbf{D} = \{d_1, d_2, \ldots, d_{n-1}\}$ that represents the sorted euclidean distances with respect to the other descriptors. Following this idea, the keypoint is matched only if this constraint is satisfied:

$$d_1/d_2 < T \quad where \quad T \in (0,1). \tag{3}$$

We refer to this procedure as 2NN test. This matching procedure has one main drawback: it is unable to manage multiple keypoint matching. This is a key aspect in case of copy-move forgeries since it may happen that the same image area is cloned over and over (see for example Fig. 2). In other words, it only finds matches between keypoints whose SIFT descriptors are very different from those of the rest of the set (i.e. features that are *globally distinctive*). Therefore, the case of cloned patches is very critical since the keypoints detected in those regions are very similar to each other.

For this reason we propose a novel matching procedure, that is a generalization of (3), and is able to deal with multiple copies of the same features. Our generalized 2NN test (referred as *g*2NN) starts from the observation that in a high-dimensional feature space such as that of SIFT features, keypoints that are different from one considered share very high and very similar values (in terms of Euclidean distances) among them. Instead, similar features show low Euclidean distances respect to the others. The idea of the 2NN test is that the ratio between the distance of the candidate match and the distance of the 2*nd* nearest neighbor is low in the case of a match (e.g. lower than 0.6) and very high in case of two "random features" (e.g. greater than 0.6). Our generalization consists in iterating the 2NN test between d_i/d_{i+1} until this ratio is greater than T (in our experiments we set this value to 0.5). If k is the value in which the procedure stops, each keypoint in correspondence to a distance in $\{d_1, \ldots, d_k\}$ $(where\ 1 \leqslant k < n)$ is considered as a match for the inspected keypoint.

Finally, by iterating over keypoints in $\mathbf{X}$, we can obtain the set of matched points. All the matched keypoints are retained, but isolated ones are no longer considered in subsequent processing steps. Already at this stage a draft idea of the authenticity of the image is provided. But it can happen that images that legitimately contain areas with very similar texture yield matched keypoints that might induce false alarms. The following two steps of the proposed methodology reduce this possibility.

B. Clustering and forgery detection

To identify possible cloned areas, an *agglomerative hierarchical clustering* [37] is performed on spatial locations (i.e. x, y coordinates) of the matched points. Hierarchical clustering creates a hierarchy of clusters which may be represented by a tree structure. The algorithm starts by assigning each keypoint to a cluster; then it computes all the reciprocal spatial distances among clusters, finds the closest pair of clusters, and finally merges them into a single cluster. Such computation is iteratively repeated until a final merging situation is achieved. The way this final merging can be accomplished is basically conditioned both by the linkage method adopted and by the threshold used to stop cluster grouping.

Several linkage methods exist in the literature and our experiments evaluate their performance to estimate the best cut-off threshold T_h for forgery detection (see Subsection IV-A for a detailed description of such experiments). In particular, three different linkage methods have been evaluated: *Single*, *Centroid* and *Ward's* linkage. Given two clusters P and Q, respectively containing n_P and n_Q objects (where $\mathbf{x}_{Pi}$ and $\mathbf{x}_{Qj}$ indicate the i^{th} and the j^{th} object in the clusters P and Q), the linkage methods operate as follows:

- *Single* linkage uses the smallest euclidean distance be-

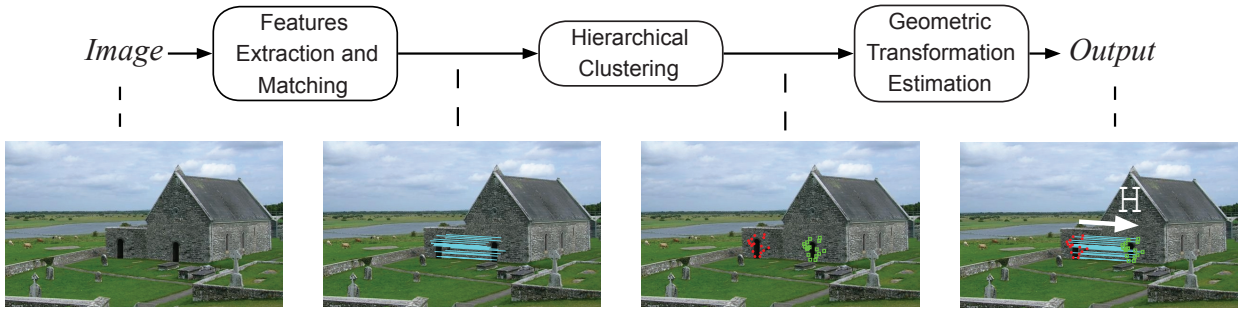

Fig. 3. Overview of the proposed system. SIFT matched pairs and clusters.

tween objects in the two clusters:

$$dist(P,Q) = \min(\|\mathbf{x}_{Pi}, \mathbf{x}_{Qj}\|_2) \quad with\ i = [1, n_P],\ j = [1, n_Q]. \tag{4}$$

- *Centroid* linkage uses the euclidean distance between the centroids of the two clusters:

$$dist(P,Q) = \|\overline{\mathbf{x}}_P - \overline{\mathbf{x}}_Q\|_2 \tag{5}$$

where

$$\overline{\mathbf{x}}_P = \frac{1}{n_P}\sum_{i=1}^{n_P}\mathbf{x}_{Pi} \quad \text{and} \quad \overline{\mathbf{x}}_Q = \frac{1}{n_Q}\sum_{i=1}^{n_Q}\mathbf{x}_{Qi}. \tag{6}$$

- *Ward's* linkage evaluates the increment/decrement in the *Error Sum of Squares* (ESS) after merging the two clusters into a single one with respect to the case of two separated clusters:

$$\Delta_{dist}(P,Q) = ESS(PQ) - [ESS(P) + ESS(Q)] \tag{7}$$

where

$$ESS(P) = \sum_{i=1}^{n_P}|\mathbf{x}_{Pi} - \overline{\mathbf{x}}_P|^2, \tag{8}$$

$\overline{\mathbf{x}}_P$ is the centroid (again) and PQ indicates the combined cluster.

According to the adopted linkage method, a specific tree structure is obtained. In addition to this, the proper choice of the threshold T_h, to determine where to cut the tree and consequently which is the final number of clusters, is crucial. The parameter which is compared with T_h is the *Inconsistency Coefficient* (IC) which characterizes each clustering operation; a higher value of this coefficient, the less similar the objects connected by the link, thus when it exceeds the threshold T_h clustering stops. IC takes basically into account the average distance among clusters and does not allow us to join clusters spatially too far at that level of hierarchy. It is easy to see that an appropriate choice of T_h directly influences tampering detection performance. At the end of the clustering procedure, clusters which do not contain a significant number (more than three) of matched keypoints are eliminated. On this basis, to optimize detection performance and consequently to the carried out experimental tests (see again Subsection IV-A), we consider that an image has been altered by a copy-move attack if the method detects two (or more) clusters with at least three pairs of matched points linking a cluster to another one. This aspect has been investigated and this assumption yields a good trade-off between the need to provide a low false alarm rate.

It should be noted that it can happen that no matched keypoints are obtained, mainly because salient features are not detected in the forged patch (e.g. when an object is hidden with a flat patch). Anyway, this is a very well-known open issue in SIFT-related scientific literature.

C. Geometric transformation estimation

When an image has been classified as non-authentic, the proposed method can determine which geometrical transformation was used between the original area and its copy-moved version. Let the matched point coordinates be, for the two areas, $\tilde{\mathbf{x}}_i = (x, y, 1)^\intercal$ and $\tilde{\mathbf{x}}'_i = (x', y', 1)^\intercal$ respectively, their geometric relationships can be defined by an affine homography which is represented by a 3×3 matrix $\mathtt{H}$ as:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \mathtt{H} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{9}$$

This matrix can be computed by resorting to at least three matched points. In particular, we determine $\mathtt{H}$ using Maximum Likelihood estimation of the homography [38]. This method seeks homography $\mathtt{H}$ and pairs of perfectly matched points $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{x}}'_i$ that minimizes the total error function as in Equation 10:

$$\sum_i \left[d(\mathbf{x}_i, \hat{\mathbf{x}}_i)^2 + d(\mathbf{x}'_i, \hat{\mathbf{x}}'_i)^2\right] \text{ subject to } \hat{\mathbf{x}}'_i = \mathtt{H}\hat{\mathbf{x}}_i\ \forall i. \tag{10}$$

However, mismatched points (*outliers*) can severely disturb the estimated homography. For this reason we perform the previous estimation by applying the RANdom SAmple Consensus algorithm (RANSAC) [39]. This algorithm randomly selects a set (in our case three pairs of points) from the matched points and estimates the homography H, then all the remained points are transformed according to H and compared in terms of distance with respect to their corresponding matched ones. If this distance is under or above a certain threshold β, they are catalogued as *inliers* or *outliers* respectively. After a pre-defined number N_{iter} of iterations, the estimated transformation which is associated with the higher number of inliers is chosen. In our experimental tests, N_{iter} has been set to 1000 and the threshold β to 0.05; this is due to the

fact that we used a standard method of normalization of the data for homography estimation. The points are translated so that their centroid is at the origin and then they are scaled so that the average distance from the origin is equal to $\sqrt{2}$. This transformation is applied to both of the two areas $\mathbf{x}_i$ and $\mathbf{x}'_i$ independently.

Once the affine homography is found, rotation and scaling transformations can be computed from its decomposition, while translation can be determined by considering the centroids of the two matched clusters. In particular, $\mathtt{H}$ can be represented as:

$$\mathtt{H} = \begin{bmatrix} \mathtt{A} & \mathbf{t} \\ \mathbf{0}^{\mathtt{T}} & 1 \end{bmatrix} \quad \text{where} \quad \mathtt{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}. \tag{11}$$

The matrix $\mathtt{A}$ is the composition of rotation and non-isotropic scaling transformations. In fact, it can always be decomposed as

$$\mathtt{A} = \mathtt{R}(\theta)(\mathtt{R}(-\Phi)\mathtt{S}\mathtt{R}(\Phi)) \tag{12}$$

where $\mathtt{R}(\theta)$ and $\mathtt{R}(\Phi)$ are rotations by θ and Φ respectively, and $\mathtt{S} = \mathtt{diag}(s_1, s_2)$ is a diagonal matrix for the scaling transformation. Hence, the $\mathtt{A}$ defines the concatenation of a rotation by Φ, a scaling by s_1 and s_2 respectively in the rotated x and y directions; a rotation back by $-\Phi$; and finally another rotation by θ. This decomposition is computed directly by the SVD (Singular Value Decomposition). In fact, the matrix $\mathtt{A}$ can be also rewritten as: $\mathtt{A} = \mathtt{U}\mathtt{S}\mathtt{V}^{\mathtt{T}} = (\mathtt{U}\mathtt{V}^{\mathtt{T}})(\mathtt{V}\mathtt{S}\mathtt{V}^{\mathtt{T}}) = \mathtt{R}(\theta)(\mathtt{R}(-\Phi)\mathtt{S}\mathtt{R}(\Phi))$ since $\mathtt{U}$ and $\mathtt{V}$ are orthogonal matrices.

IV. Experimental results

In this section we evaluate the proposed methodology providing two main kinds of the tests: firstly, on a small dataset named MICC-F220, a benchmarking of the technique is done to properly set the threshold T_h and to compare it with other methods known in the literature; secondly, on a larger dataset named MICC-F2000, a complete evaluation is carried out by testing the system against different types of modifications. Both datasets are composed of images with different contents coming from the Columbia photographic image repository [40] and from a personal collection. The first dataset MICC-F220 consists of 220 images: 110 are tampered images and 110 are originals. The image resolution varies from 722×480 to 800×600 pixels and the size of the forged patch covers, on the average, 1.2% of the whole image. The second dataset MICC-F2000 is composed of 2000 photos of 2048×1536 pixels (3M pixels) and the forgery is, on average, 1.12% of the whole image: so it is again quite small and similar to the MICC-F220 dataset case. To reproduce as much as possible a practical situation, the number of original and altered images belonging to the MICC-F2000 dataset is not the same: 1300 original images and 700 tampered images have been taken. The forged images are obtained, in both the datasets, by randomly selecting (both as location and as dimension) an image area (square or rectangular) and copy-pasting it over the image after having applied a number of different attacks such as translation, rotation, scale (symmetric/asymmetric) or a combination of them.

Table I and Table II summarize the geometric transformations for the attack applied in the MICC-F220 dataset (10 attacks, from *A* to *J* in Table I) and in the MICC-F2000 (14 attacks, from *a* to *o* in Table II), respectively. In particular, for each attack we report the rotation θ in degrees and the scaling factors s_x, s_y applied to the x and y axis of the cloned image part (e.g. in the attack *G*, the x and y axes are scaled by 30%, and no rotation is performed).

TABLE I
THE 10 DIFFERENT COMBINATIONS OF GEOMETRIC TRANSFORMATIONS APPLIED TO THE ORIGINAL PATCH FOR THE MICC-F220 DATASET.

Attack	θ °	s_x	s_y	Attack	θ °	s_x	s_y
A	0	1	1	*F*	0	1.2	1.2
B	10	1	1	*G*	0	1.3	1.3
C	20	1	1	*H*	0	1.4	1.2
D	30	1	1	*I*	10	1.2	1.2
E	40	1	1	*J*	20	1.4	1.2

TABLE II
THE 14 DIFFERENT COMBINATIONS OF GEOMETRIC TRANSFORMATIONS APPLIED TO THE ORIGINAL PATCH FOR THE MICC-F2000 DATASET.

Attack	θ °	s_x	s_y	Attack	θ °	s_x	s_y
a	0	1	1	h	0	1.2	1.6
b	0	0.5	0.5	i	5	1	1
c	0	0.7	0.7	j	30	1	1
d	0	1.2	1.2	l	70	1	1
e	0	1.6	1.6	m	90	1	1
f	0	2	2	n	40	1.1	1.6
g	0	1.6	1.2	o	30	0.7	0.9

A. Settings for forgery detection

First of all, the proposed method is analyzed to determine the best settings for the cut-off threshold T_h introduced in Section III-B according to the chosen linkage method. These values are set for the all remaining experiments and comparisons. To address this problem, the following experiment has been set-up applying a *4*-fold cross-validation process: from the database of 220 images (MICC-F220), 165, that is $3/4$ of the image set, (82 tampered and 83 original) have been randomly chosen to perform a training to find the best threshold T_h for each of the three considered linkage methods (*Single*, *Centroid*, *Ward's*); the remaining 55 images ($1/4$ of the whole set) have been used in a successive testing phase to evaluate detection performances of the proposed technique. The experiment was repeated four times, by cyclically exchanging the four image sub-sets belonging to the training (3 sub-sets) and to the testing set (1 sub-set), and the results have been averaged. Detection performance was measured in terms of True Positive Rate (TPR) and False Positive Rate (FPR), where TPR is the fraction of tampered images correctly identified as such, while FPR is the fraction of original images that are not correctly identified:

$$\text{TPR} = \frac{\text{\# images detected as forged being forged}}{\text{\# forged images}},$$
$$\text{FPR} = \frac{\text{\# images detected as forged being original}}{\text{\# original images}}.$$

We repeat that an image is considered to have been altered by a copy-move attack if the method detects two (or more)

clusters with at least three pairs of matched points that link a cluster to another one (as detailed in Subsection III-B).

TABLE III
TRAINING PHASE: TPR AND FPR VALUES (IN PERCENTAGE) FOR EACH METRIC WITH RESPECT TO T_h.

	Single		*Centroid*		*Ward's*	
T_h	FPR(%)	TPR(%)	FPR(%)	TPR(%)	FPR(%)	TPR(%)
0.8	2.729	41.827	1.822	23.626	0.911	10.906
1	5.455	70.001	4.547	56.373	3.636	32.739
1.2	8.180	89.994	7.273	90	7.273	82.714
1.4	8.180	95.456	8.180	95.456	7.273	90.905
1.6	**8.180**	**98.185**	7.273	97.274	8.180	97.274
1.8	7.269	96.360	**8.180**	**98.182**	9.088	99.089
2	6.362	91.820	7.269	95.456	9.088	100
2.2	5.451	82.721	5.451	92.723	**8.177**	**100**
2.4	4.544	63.639	4.544	84.536	7.269	96.364
2.6	2.726	48.185	2.729	70.897	7.273	89.998
2.8	0.911	22.726	1.822	46.360	3.640	78.170
3	0.911	15.461	0.911	18.179	3.640	61.813

In Table III, for each linkage method, the TPR and the FPR obtained during the training phase are reported with respect to the threshold T_h, which varies in the interval [0.8, 3] with steps of 0.2. The goal was to minimize the FPR while maintaining a very high TPR; we see that FPR is almost always very low, while on the contrary TPR is highly variable. The optimal threshold T_h has been chosen, as evidenced in Table III, for the maximum value of TPR that is 1.6 for the *Single* linkage method, 1.8 for the *Centroid* and 2.2 for the *Ward's* linkage. Finally, the test phase has been launched for the best metrics using the T_h previously obtained in the training phase. The final detection results are reported in Table IV. These values show that the proposed method performs satisfactorily, providing a low FPR while maintaining a high rate of correct tampering detection for all the used linkage methods, though *Ward's* metric seems to be slightly better. We conclude that the choice of linkage methods is not so fundamental, while the T_h setting is crucial.

TABLE IV
TEST PHASE ON MICC-F220 DATASET: DETECTION RESULTS IN TERMS OF FPR AND TPR.

	Single	*Centroid*	*Ward's*
FPR (%)	8.16	8.16	8
TPR (%)	98.21	98.17	100

Furthermore, for the cases of correctly detected forged images, the estimation of the geometric transformation parameters which bring the original patch onto the forged one has also been computed. The Mean Absolute Error (MAE) between each of the true values of the transformation parameters and the estimated ones are reported in Table V. As in the previous tables, s_x and s_y refers to the scaling factors occurred in the transformation; θ refers to the rotation (in degrees) while t_x and t_y are translation on x/y direction respectively.

Results show high degree of precision in the estimate of the various parameters of the affine transformation. In addition to this, Table VI reports for one of the test image belonging to the MICC-F220, named *Cars* (see Fig. 4, first column), each transformation parameter (the original value applied to the patch, the estimated one and the absolute error ($|e|$)). It can

TABLE V
TRANSFORMATION PARAMETER ESTIMATION ERRORS FOR THE MICC-F220 (*Single* LINKAGE METHOD WITH $T_h = 1.6$, AS PREVIOUSLY UNDERLINED OTHER METRICS GIVE SIMILAR PERFORMANCES). THE VALUES t_x AND t_y ARE EXPRESSED IN PIXELS WHILE θ IN DEGREES.

MAE (t_x)	MAE (t_y)	MAE (θ)	MAE (s_x)	MAE (s_y)
4.04	2.48	0.94	0.021	0.015

be observed how reliable the estimate is, specifically for the scale parameters and also for an asymmetric scaling combined with a rotation.

1) Qualitative evaluation: Here we report some experimental results on images where a copy-move attack has been performed by taking into account the context. In this case the patch is selected according to the specific goal to be achieved and, above all, transformed by paying attention to perfectly conceal the modification. Alterations are not recognizable at least at a first glance and a forensic tool could help in investigations. In Fig. 4, four such cases are pictured by presenting the tampered image and the corresponding one where matched keypoints and clusters, extracted by the proposed method, are highlighted. An interesting situation concerns the individuation of a cloned patch for the image named *Dune* (second column) where, though the duplicated area is quite flat, the method is able to detect a sufficient number of matched keypoints. On the contrary, the opposite occurs for the image named *Santorini* (last column), where a large number of matched keypoints is obtained; now the cloned block is very textured and though it has undergone a geometrical transformation to be properly adapted to the context, the SIFT algorithm is robust enough not to be disturbed.

2) Copy-move methods comparison: The proposed approach has been compared to the results obtained with our implementations of the methods presented in [16], based on Discrete Cosine Transform (DCT), and in [21], based on Principal Component Analysis (PCA) (both have been previously introduced in Section II). The input parameters required by the two methods are set as it follows: $b = 16$ (number of pixels per block), $N_n = 5$ (number of neighborhood rows to search in the lexicographically sorted matrix), $N_f = 1000$ (threshold for the minimum frequency) and $N_d = 22$ (threshold to determine a duplicated block). These parameters are used in both the algorithms, while $e = 0.01$ (fraction of the ignored variance along the principle axes after PCA is computed) and $Q = 256$ (number of the quantization bins) are only used for the method presented in [21]. In our method, *Ward's* linkage with a threshold $T_h = 2.2$ has been used.

The experiments have been launched on the whole MICC-F220 image database on a machine with an *Intel Q6600 quad core with 4-GB RAM* and the FPR, TPR and processing time have been evaluated. Table VII shows the detection performance and the processing time on average (in seconds) for an image.

The results indicate that the proposed method performs better with respect to the others methods; in fact the processing time (per image) is on average about 5 seconds, whereas the other two take more than 1 minute and almost 5 minutes

TABLE VI
TRANSFORMATION PARAMETER ESTIMATION ON IMAGE *Cars*. THE VALUES t_x AND t_y ARE EXPRESSED IN PIXELS WHILE θ IN DEGREES.

A	t_x	$\hat{t}_x$	$\|e\|$	t_y	$\hat{t}_y$	$\|e\|$	θ	$\hat{\theta}$	$\|e\|$	s_x	$\hat{s}_x$	$\|e\|$	s_y	$\hat{s}_y$	$\|e\|$
A	304	304.02	0.02	80.5	81.01	0.51	0	0.040	0.040	1	1.004	0.004	1	0.998	0.002
B	304	305.20	1.20	80.5	82.42	1.92	10	9.963	0.037	1	1.001	0.001	1	0.999	0.001
C	304	305.55	1.55	80.5	82.64	2.14	20	20.009	0.009	1	1.006	0.006	1	0.998	0.002
D	304	305.04	1.04	80.5	82.49	1.99	30	30.092	0.092	1	1.002	0.002	1	0.998	0.002
E	304	306.08	2,08	80.5	78.43	2.07	40	39.932	0.067	1	1.007	0.007	1	1.004	0.004
F	304	304.88	0.88	80.5	80.41	0.09	0	0.080	0.080	1.2	1.202	0.002	1.2	1.198	0.002
G	304	305.07	1.07	80.5	79.87	0.63	0	0.108	0.108	1.3	1.304	0.004	1.3	1.303	0.003
H	304	305.78	1.78	80.5	80.18	0.32	0	0.037	0.037	1.4	1.403	0.003	1.2	1.206	0.006
I	304	305.23	1.23	80.5	81.76	1.26	10	9.910	0.090	1.2	1.203	0.003	1.2	1.201	0.001
J	304	305.02	1.02	80.5	80.82	0.32	20	20.067	0.067	1.4	1.404	0.004	1.2	1.198	0.002

Fig. 4. Some examples of tampered images are pictured in the first row, the corresponding detection results are reported in the second row.

TABLE VII
TPR, FPR VALUES (%) AND PROCESSING TIME (AVERAGE, PER IMAGE) FOR EACH METHOD.

Method	FPR (%)	TPR (%)	Time (s)
Fridrich *et al.* [16]	84	89	294.69
Popescu and Farid [21]	86	87	70.97
Our method	8	100	4.94

respectively. Furthermore, the DCT and PCA methods, though obtaining an acceptable TPR, fail when a decision about original image is required (high FPR values in Table VII). This is basically due to the incapacity of such methods to properly deal with cases where a geometrical transformation which is not pure translation is applied to the copy-moved patch. For the specific case of pure patch translation FPR is 0% for all the three methods.

B. Test on multiple copied regions

In this experiment we analyze the performance of our method in presence of tampered images which have multiple copies of the same region. This test has been performed on ten photos of 2048×1536 pixels. In these pictures, one or more image areas are copied and pasted in several different positions over the image, taking into account the context in order to hide, at first glance, the forgery.

In this scenario, as we have previously highlighted in Subsection III-A, the standard 2NN matching procedure is a critical point for copy-move forgery detection methods based on SIFT features [31], [32]. In fact, comparing the standard SIFT matching technique with our g2NN strategy, we see that our method increases (on average) by 195% the number of the extracted matches. A high number of matches is fundamental in order to have sufficient information for a correct estimation of the geometric transformation, but it can introduce false alarms. To this end, we tested our matching strategy on the MICC-F220 dataset in order to evaluate how these new matches influence the results. We found that we lose on average 3% in terms of FPR with the same results in terms of TPR.

Fig. 5 shows a qualitative comparison between the two techniques. It is interesting to note that the number of the matched keypoints between the gun a and c, obtained by the standard 2NN technique, are very few (2 matches) with respect to g2NN (54 matches). In this case the technique, based on standard matching, fails to detect the relationship between the two guns. Finally, Fig. 6 shows some examples of multiple cloning obtained with the g2NN test. Detection results are reported by highlighting matched keypoints and clusters.

C. Test on a large dataset

In this section, experimental results obtained on a larger dataset, named MICC-F2000, to verify the behavior of the proposed technique are presented; detection performances and geometric transformation parameters estimation are investigated as well. Furthermore tests to check the robustness of the method against usual operations such as JPEG compression or

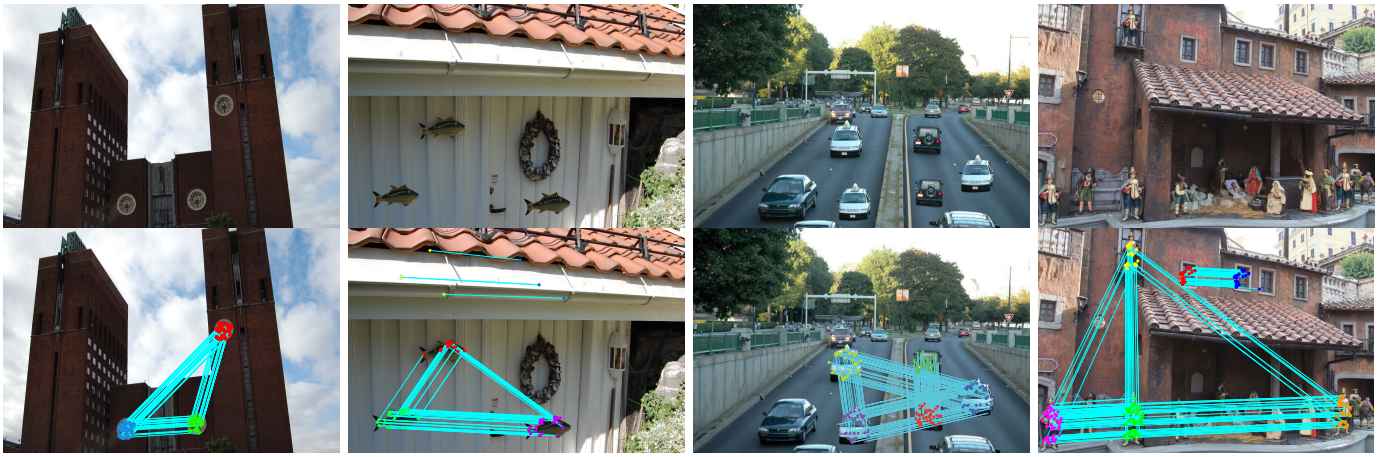

Fig. 6. Examples of tampered images with multiple cloning are shown in the first row, while the detection results are reported in the second row.

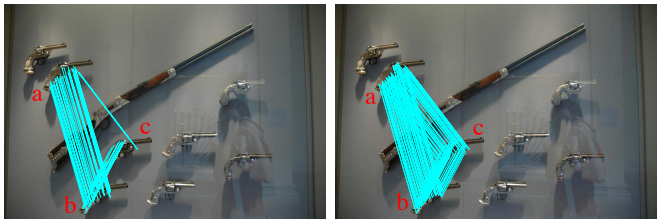

Fig. 5. Matched keypoints computed by the 2NN standard SIFT matching technique (left) and our g2NN strategy (right).

noise addition, an image can undergo, have been carried out; such kinds of processing have been considered as applied both to the whole forged image and only to the altered image patch.

TABLE VIII
TRAINING PHASE ON MICC-F2000 DATASET: TPR AND FPR VALUES FOR EACH METRIC WITH RESPECT TO T_h.

	Single		*Centroid*		*Ward's*	
T_h	FPR(%)	TPR(%)	FPR(%)	TPR(%)	FPR(%)	TPR(%)
0,8	3.41	51.86	1.69	32.29	0.54	11.43
1	5.56	70.19	4.92	62.43	3	51.29
1,2	10.28	89.95	10.31	87.43	9.54	83.86
1,4	10.95	91.24	12.15	90.14	11.62	88.43
1.6	**10.97**	**93**	13.23	93.57	13.15	93.14
1.8	9.46	91	**12.46**	**93.43**	14.54	93.86
2	7.46	84.43	11.23	92.29	13.85	93.86
2.2	4.79	72.38	9.00	89.43	**11.62**	**93.43**
2,4	2.72	54.43	6.46	78.43	9.85	91.29
2,6	1.00	29.14	3.23	62.86	8.46	87.71
2,8	0.21	19.86	1.23	40.86	5.62	79.43
3	0.08	12.86	0.38	23.29	3.38	67.43

First of all, we set up again an experiment to determine the best threshold T_h, according to the three linkage methods, as done in Subsection IV-A for the MICC-F220; this has been made to further check if the established thresholds were correct. To do this, a 4-fold cross-validation process has been carried out. Results are listed in Table VIII. It can be observed that a similar behavior to that obtained with MICC-F220 is obtained and, above all, that the values chosen in Subsection IV-A for T_h (1.6 for *Single*, 1.8 for *Centroid* and 2.2 for *Ward's*) still yield higher performance in terms of TPR and FPR. After this, the test phase is launched by setting such values for T_h. In Table IX the detection rates are reported demonstrating both the effectiveness of the proposed method which achieves a TPR of around 93% for all the three metrics, and its robustness obtaining again performances very consistent with those in Table VIII for these fixed thresholds.

TABLE IX
TEST PHASE ON MICC-F2000 DATASET: DETECTION RESULTS IN TERMS OF FPR AND TPR OBTAINED WITH $T_h = 1.6$, $T_h = 1.8$ AND $T_h = 2.2$ FOR THE THREE LINKAGE METHODS, RESPECTIVELY.

	Single	*Centroid*	*Ward's*
FPR (%)	10.99	12.45	11.61
TPR (%)	92.99	93.23	93.42

Going into detail, in Fig. 7 the number of errors for each attack is listed with regard to tampered images not detected as such. The most critical attacks seem to be the f (θ = 0°, $s_x = 2$ and $s_y = 2$) and the n (θ = 40°, $s_x = 1.1$ and $s_y = 1.6$) which increase twice the patch dimension and apply a 40 degrees rotation combined with a consistent variation in scale. The histogram in Fig. 7 shows that these two kinds of attacks generate around 30% of the total errors.

In Table X are reported the errors in estimated geometric transformation parameters averaged over all 500 test images. The Mean Absolute Error (MAE) still remains small although the transformations applied to the images for the MICC-F2000 dataset are more challenging with respect to those in the MICC-F220 dataset.

TABLE X
TRANSFORMATION PARAMETER ESTIMATION ERRORS. THE VALUES t_x AND t_y ARE EXPRESSED IN PIXELS, WHILE θ IS IN DEGREES.

MAE(t_x)	MAE(t_y)	MAE(θ)	MAE(s_x)	MAE(s_y)
22.49	8.49	1.55	0.27	0.2

1) JPEG compression and noise addition: The proposed methodology has also been tested in terms of detection performance from a robustness point of view; in particular, the impact of JPEG compression and noise addition on all the 2000 images of the MICC-F2000 dataset has been investigated.

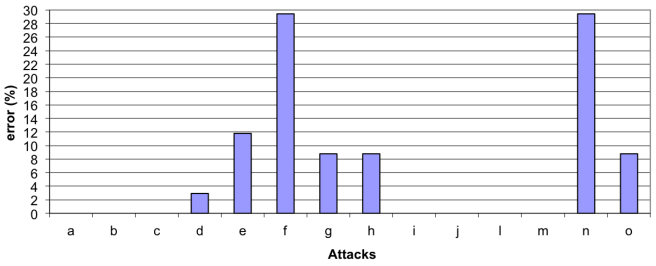

Fig. 7. Error analysis of tampered images misdetection for each different attack (in percentage).

In the first experiment all the images which were originally in the JPEG format (quality factor of 100), have been compressed in JPEG format with a decreasing quality factor of 75, 50, 40 and 20. Table XI (left) shows the FPR and TPR (*Ward's* linkage method with $T_h = 2.2$) for all the diverse JPEG quality factors; it can be seen that FPR is practically stable while the TPR tends to slightly diminish when image quality decreases. In the second experiment, in the same way as before, the images of MICC-F2000 dataset are distorted by adding a Gaussian noise to obtain a decreasing signal-noise-ratios (SNR) of 50, 40, 30 and 20 db. Noisy images are obtained by adding white Gaussian noise to the image with a JPEG quality factor of 100. In Table XI (right), obtained results are shown and it can be noticed that the TPR is over 90% till a SNR of 30 dB while FPR is again quite stable, though it seems to even improve.

TABLE XI
DETECTION PERFORMANCES AGAINST JPEG COMPRESSION (LEFT) AND NOISE ADDITION (RIGHT).

JPEG quality	FPR	TPR
100	11.61	93.42
75	12.07	93.42
50	11.15	93.16
40	11.38	92.14
20	10.46	87.15

SNR (dB)	FPR	TPR
50	11.46	93.71
40	11.69	94.14
30	11.46	92.00
20	8.15	82.42

2) JPEG compression, noise addition, and gamma correction on copied patch: Duplicated patches are often modified by applying further processing such as brightness/contrast adjustment, gamma correction, noise addition and so on, in order to adjust the patch with respect to the image area where it has to be located. To explore this scenario the following experiment has been performed. Starting from 10 original images, a block is randomly (as explained before) selected for each of them and 4 geometric transformations (*a*, *d*, *j* and *o* from Table II) are applied to every one of these patches. Furthermore, before pasting them, 4 different gamma corrections with values [2.2, 1.4, 0.7, 0.45] are applied to each block. Finally, 160 tampered images are obtained. In the same way, the final stage of gamma correction is firstly substituted by JPEG compression with different quality factors [75, 50, 40, 20] and secondly by Gaussian noise addition with SNR (dB) equal to 50, 40, 30, 20. For every case, 160 fake images have been created. So for each of the three situations (gamma correction, JPEG compression and noise addition), a dataset composed by 160 fake images and by 350 original ones randomly taken from the MICC-F2000 database is built. In Table XII, performance in terms of TPR and FPR are reported.

Fig. 9. Example of wrong detection of a splicing attack.

TABLE XII
DETECTION PERFORMANCE AGAINST GAMMA CORRECTION, JPEG COMPRESSION AND NOISE ADDITION APPLIED TO THE DUPLICATED AND GEOMETRICALLY TRANSFORMED PATCH.

Kind of processing	FPR	TPR
Gamma correction	9.23	99.37
JPEG	11.38	100.00
SNR (dB)	12.00	100.00

These experiments show that the proposed method maintains its level of accuracy though diverse kinds of post-processing are applied to the duplicated patch in addition to a geometric transformation, to adapt it to the image context where it is pasted.

D. Image splicing

Though the proposed technique has been presented to operate in a copy-move attack scenario, it can also be utilized in a context where a splicing operation has occurred. With the term splicing attack we mean that a part of an image is grabbed and, possibly after having been adapted (geometrically transformed and/or enhanced), and pasted onto another one to build a new, fake, image. In most cases only the final fake photo is available to the forensic analyst for inspection, the source image is often undeterminable; because of this, the SIFT matching procedure, which is the core of the proposed method can not take place and would seem that there is no room for it in such circumstance. This is not always true in practice. In fact, often, the analyst is required to give an assessment of a dataset of images, for example, belonging to a specific person under judgement, or that have been found on a hard disk or a pen drive, and so on. In this operative scenario, it can happen that the source image used to create fraudulent content belongs to the image collection at disposal. It is easy to understand that the proposed method can be adopted again to determine both if within the to-be-checked collection there is a false image containing an "external" patch and, above all, where it comes from. It is interesting to highlight that succeeding in detecting such links could help investigation activities. To show that the proposed technique can be used in such a scenario the following experimental test has been set up.

A subset of 100 images (96 original and 4 tampered with) taken from a private collection with size of 800×600 pixels has been selected. In particular, the 4 fake images have been

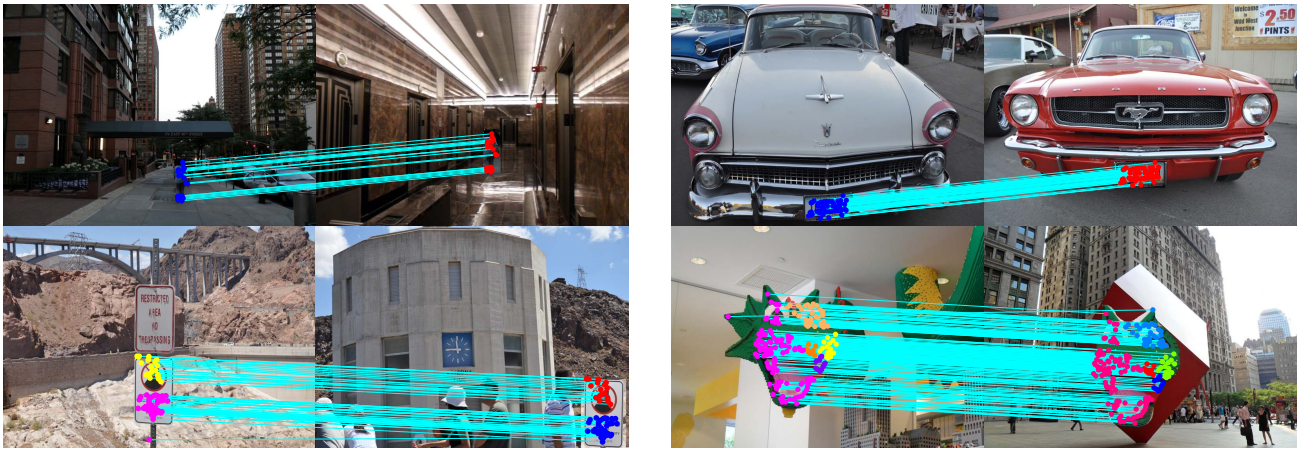

Fig. 8. Examples of correct detection of a splicing attack.

created by pasting a patch that was cut from another image belonging to the other original 96. The proposed technique has been launched to analyze all the possible pairs of photos $\left(\frac{n\cdot(n-1)}{2} = \frac{100\cdot 99}{2} = 4950\right)$ in the dataset looking for duplicated areas. To allow to the presented algorithm to perform as is, the pair of images to be checked are considered as a single image with a double number of columns (size equal to $N \times 2M$); due to this, the detection threshold T_h has been increased up to 3.4 (it was 2.2 in the previous experiments of this section) for *Ward's* linkage method which was chosen for this experiment. In Table XIII performances in terms of FPR and TPR are reported.

TABLE XIII
DETECTION PERFORMANCES AGAINST SPLICING ATTACK (IN PERCENTAGE).

Splicing attack	FPR (%)	TPR (%)
	0.04	100.00

The method is able to correctly reveal all four fake pairs, determining a link between the possible original image and the forged one, though it can not distinguish the source from the destination if other tools are not adopted. The procedure also detects two other innocent pairs of images causing false alarms. In Fig. 8 the four cases of splicing attack detection are pictured, while in Fig. 9 one of the false alarms is illustrated. In this last case, we see immediately that the error is caused by the presence of the same objects (the posters over the wall of the wooden box) in both the photos taken in the same real context. However this could be an actual situation that happens in practical scenarios (e.g. establishing possible relations among photos acquired in similar environments).

V. CONCLUSION

A novel methodology to support image forensics investigation based on SIFT features has been proposed. Given a suspected photo, it can reliably detect if a certain region has been duplicated and, furthermore, determine the geometric transformation applied to perform such tampering. The presented technique shows effectiveness with respect to diverse operative scenarios such as composite processing and multiple cloning. Future work will be mainly dedicated to investigating how to improve the detection phase with respect to the cloned image patch with highly uniform texture where salient keypoints are not recovered by SIFT-like techniques. In particular, integration with other forensics techniques applied locally onto flat zones is envisaged. Furthermore, the clustering phase will be extended by means of an image segmentation procedure.

ACKNOWLEDGMENTS

The authors would like to thank Luca Del Tongo for his support in the preparation of the experiments.

REFERENCES

[1] S. Lyu and H. Farid, "How realistic is photorealistic?" *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 845–850, 2005.
[2] H. Farid, "Photo fakery and forensics," *Advances in Computers*, vol. 77, pp. 1–55, 2009.
[3] J. A. Redi, W. Taktak, and J.-L. Dugelay, "Digital image forensics: a booklet for beginners," *Multimedia Tools and Applications*, vol. 51, no. 1, pp. 133–162, 2011.
[4] I. J. Cox, M. L. Miller, and J. A. Bloom, *Digital watermarking*. San Francisco, CA: Morgan Kaufmann, 2002.
[5] M. Barni and F. Bartolini, *Watermarking Systems Engineering: Enabling Digital Assets Security and Other Applications*. Marcel Dekker, 2004.
[6] H. Farid, "A survey of image forgery detection," *IEEE Signal Processing Magazine*, vol. 2, no. 26, pp. 16–25, 2009.
[7] A. Popescu and H. Farid, "Statistical tools for digital forensics," in *Proc. of Int.'l Workshop on Information Hiding*, Toronto, Canada, 2005.
[8] A. Swaminathan, M. Wu, and K. Liu, "Digital image forensics via intrinsic fingerprints," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 1, pp. 101–117, 2008.
[9] M. Chen, J. Fridrich, M. Goljan, and J. Lukas, "Determining image origin and integrity using sensor noise," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 1, pp. 74–90, 2008.
[10] N. Khanna, G. T.-C. Chiu, J. P. Allebach, and E. J. Delp, "Forensic techniques for classifying scanner, computer generated and digital camera images," in *Proc. of IEEE ICASSP*, Las Vegas, USA, 2008.
[11] R. Caldelli, I. Amerini, and F. Picchioni, "A DFT-based analysis to discern between camera and scanned images," *International Journal of Digital Crime and Forensics*, vol. 2, no. 1, pp. 21–29, 2010.
[12] I. Amerini, L. Ballan, R. Caldelli, A. Del Bimbo, and G. Serra, "Geometric tampering estimation by means of a SIFT-based forensic analysis," in *Proc. of IEEE ICASSP*, Dallas, USA, 2010.
[13] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int.'l Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[14] S. Bayram, I. Avcibas, B. Sankur, and N. Memon, "Image manipulation detection with binary similarity measures," in *Proc. of EUSIPCO*, Antalya, Turkey, 2005.
[15] H. Farid and S. Lyu, "Higher-order wavelet statistics and their application to digital forensics," in *Proc. of IEEE CVPR Workshop on Statistical Analysis in Computer Vision*, Madison, WI, USA, 2003.
[16] J. Fridrich, D. Soukal, and J. Lukás, "Detection of copy-move forgery in digital images," in *Proc. of DFRWS*, Cleveland, OH, 2003.
[17] W. Luo, J. Huang, and G. Qiu, "Robust detection of region-duplication forgery in digital image," in *Proc. of ICPR*, Washington, D.C., USA, 2006.
[18] G. Li, Q. Wu, D. Tu, and S. J. Sun, "A sorted neighborhood approach for detecting duplicated regions in image forgeries based on DWT and SVD," in *Proc. of IEEE ICME*, Beijing, China, 2007.
[19] B. Mahdian and S. Saic, "Detection of copy-move forgery using a method based on blur moment invariants," *Forensic Science International*, vol. 171, no. 2-3, pp. 180–189, 2007.
[20] B. Dybala, B. Jennings, and D. Letscher, "Detecting filtered cloning in digital images," in *Proc. of ACM Int.'l Workshop on Multimedia & Security (MM&Sec)*, New York, NY, USA, 2007.
[21] A. Popescu and H. Farid, "Exposing digital forgeries by detecting duplicated image regions," Dartmouth College, Computer Science, Tech. Rep. TR2004-515, 2004.
[22] S. Bayram, H. Taha Sencar, and N. Memon, "An efficient and robust method for detecting copy-move forgery," in *Proc. of IEEE ICASSP*, Washington, DC, USA, 2009.
[23] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting traces of resampling," *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 758–767, 2005.
[24] S. Bayram, H. T. Sencar, and N. Memon, "A survey of copy-move forgery detection techniques," in *Proc. of IEEE Western New York Image Processing Workshop*, Rochester, NY, 2008.
[25] S.-J. Ryu, M.-J. Lee, and H.-K. Lee, "Detection of copy-rotate-move forgery using zernike moments," in *Proc. of International Workshop on Information Hiding*, Calgary, Canada, 2010.
[26] H.-J. Lin, C.-W. Wang, and Y.-T. Kao, "Fast copy-move forgery detection," *WSEAS Transactions on Signal Processing*, vol. 5, no. 5, pp. 188–197, 2009.
[27] S. Bravo-Solorio and A. K. Nandi, "Passive method for detecting duplicated regions affected by reflection, rotation and scaling," in *Proc. of EUSIPCO*, Glasgow, Scotland, 2009.
[28] V. Christlein, C. Riess, and E. Angelopoulou, "A study on features for the detection of copy-move forgeries," in *Proc. of Information Security Solutions Europe*, Berlin, Germany, 2010.
[29] X. Shuai, C. Zhang, and P. Hao, "Fingerprint indexing based on composite set of reduced SIFT features," in *Proc. of ICPR*, Tampa, Florida, USA, 2008.
[30] H. Su, A. Bouridane, and M. Gueham, "Local image features for shoeprint image retrieval," in *Proc. of BMVC*, Warwick, UK, 2007.
[31] H. Huang, W. Guo, and Y. Zhang, "Detection of copy-move forgery in digital images using SIFT algorithm," in *Proc. of IEEE Pacific-Asia Workshop on Computational Intell. and Industrial Application*, Wuhan, China, 2008.
[32] X. Pan and S. Lyu, "Detecting image region duplication using SIFT features," in *Proc. of IEEE ICASSP*, Dallas, USA, 2010.
[33] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.
[34] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, "A comparison of affine region detectors," *International Journal of Computer Vision*, vol. 65, no. 1/2, pp. 43–72, 2005.
[35] W. Wei, S. Wang, X. Zhang, and Z. Tang, "Estimation of image rotation angle using interpolation-related spectral signatures with application to blind detection of image forgery," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 3, pp. 507–517, 2010.
[36] W. Lu, A. L. Varna, and M. Wu, "Forensic hash for multimedia information," in *Proc. of SPIE Media Forensics and Security*, San Jose, CA, 2010.
[37] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning*. Springer, 2003.
[38] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.
[39] M. Fischler and R. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
[40] T.-T. Ng, S.-F. Chang, J. Hsu, and M. Pepeljugoski, "Columbia photographic images and photorealistic computer graphics dataset," ADVENT, Columbia University, Tech. Rep., 2004.

Irene Amerini received the Laurea degree in computer engineering in 2006 and the Ph.D. degree in computer engineering, multimedia and telecommunication in 2011, both from the University of Florence, Italy. She was a visiting scholar at Binghamton University, Binghamton, NY, in 2010. Currently she is a postdoctoral researcher at the Image and Communication Lab at the Media Integration and Communication Center, University of Florence. Her main research interests focus on multimedia forensics and image processing.

Lamberto Ballan received the Laurea degree in computer engineering in 2006 and the Ph.D. degree in computer engineering, multimedia and telecommunication in 2011, both from the University of Florence, Italy. He was a visiting scholar at Télécom ParisTech/ENST, Paris, in 2010. Currently he is a postdoctoral researcher at the Media Integration and Communication Center, University of Florence. His main research interests focus on multimedia information retrieval, image and video analysis, pattern recognition, and computer vision.

Roberto Caldelli received the Laurea degree in electronic engineering in 1997 and the Ph.D. degree in computer science and telecommunication in 2001, both from the University of Florence, Italy. Currently he is an assistant professor at the Media Integration and Communication Center of the University of Florence. He is also a member of CNIT. His main research activities, witnessed by several publications, include digital image processing, image and video digital watermarking, multimedia forensics.

Alberto Del Bimbo is a full professor of computer engineering at the University of Florence, Italy, where he is also the director of the Master in Multimedia, and the director of the Media Integration and Communication Center. His research interests include pattern recognition, multimedia information retrieval, and human-computer interaction. He has published more than 250 publications in some of the most distinguished scientific journals and international conferences, and is the author of the monograph Visual Information Retrieval. He is an IAPR fellow and associate editor of Multimedia Tools and Applications, Pattern Analysis and Applications, Journal of Visual Languages and Computing, and International Journal of Image and Video Processing, and was an associate editor of Pattern Recognition, IEEE Transactions on Multimedia, and IEEE Transactions on Pattern Analysis and Machine Intelligence.

Giuseppe Serra is a postdoctoral researcher at the Media Integration and Communication Center, University of Florence, Italy. He was a visiting scholar at Carnegie Mellon University, Pittsburgh, PA, and at Télécom ParisTech/ENST, Paris, in 2006 and 2010 respectively. His research interests include image and video analysis, multimedia ontologies, image forensics, and multiple-view geometry. He received the Laurea degree in computer engineering in 2006 and the Ph.D. degree in computer engineering, multimedia and telecommunication in 2010, both from the University of Florence.