Robust Image Alignment for Tampering Detection

Sebastiano Battiato, *Senior Member, IEEE*, Giovanni Maria Farinella, *Member, IEEE*, Enrico Messina, and Giovanni Puglisi, *Member, IEEE*

***Abstract*—The widespread use of classic and newest technologies available on Internet (e.g., emails, social networks, digital repositories) has induced a growing interest on systems able to protect the visual content against malicious manipulations that could be performed during their transmission. One of the main problems addressed in this context is the authentication of the image received in a communication. This task is usually performed by localizing the regions of the image which have been tampered. To this aim the aligned image should be first registered with the one at the sender by exploiting the information provided by a specific component of the forensic hash associated to the image. In this paper we propose a robust alignment method which makes use of an image hash component based on the Bag of Features paradigm. The proposed signature is attached to the image before transmission and then analyzed at destination to recover the geometric transformations which have been applied to the received image. The estimator is based on a voting procedure in the parameter space of the model used to recover the geometric transformation occurred into the manipulated image. The proposed image hash encodes the spatial distribution of the image features to deal with highly textured and contrasted tampering patterns. A block-wise tampering detection which exploits an histograms of oriented gradients representation is also proposed. A non-uniform quantization of the histogram of oriented gradient space is used to build the signature of each image block for tampering purposes. Experiments show that the proposed approach obtains good margin of performances with respect to state-of-the art methods.**

***Index Terms*—Bag of features (BOF), forensic hash, geometric transformations, image forensics, image registration, tampering.**

I. Introduction and Motivations

THE growing demand of techniques useful to protect digital visual data against malicious manipulations is induced by different episodes that make questionable the use of visual content as evidence material [1], [2]. Specifically, methods useful to establish the validity and authenticity of a received image are needed in the context of Internet communications. The problem of tampering detection can be addressed using a watermarking-based approach. The watermark is inserted into the image, and during tampering detection, it is extracted to verify if there was a malicious manipulation on the received image. A damage into the watermark indicates a tampering of the image under consideration. A clear disadvantage in using watermarking is the need for distorting the content. To overcome this problem signature-based approaches have been introduced. In this latter case the image hash is not embedded into the image; it is associated with the image as header information and must be small and robust against different operations. Different signature-based approaches have been recently proposed in literature [3]–[10]. Most of them share the same basic scheme: 1) a hash code based on the visual content is attached to the image to be sent; 2) the hash is analyzed at destination to verify the reliability of the received image.

An image hash is a distinctive signature which represents the visual content of the image in a compact way (usually just few bytes). The image hash should be robust against allowed operations and at the same time it should differ from the one computed on a different/tampered image. Image hashing techniques are considered extremely useful to validate the authenticity of an image received through a communication channel. Although the importance of the binary decision task related to the image authentication, this is not always sufficient. In the application context of Forensic Science is fundamental to provide scientific evidence through the history of the possible manipulations applied to the original image to obtain the one under analysis. In many cases, the source image is unknown, and, as in the application context of this paper, all the information about the manipulation of the image should be recovered from the short image hash signature, making more challenging the final task. The list of manipulations provides to the end user the information needed to decide whether the image can be trusted or not.

In order to perform tampering localization, the receiver should be able to filter out all the geometric transformations (e.g., rotation, scaling, translation, etc.) added to the tampered image by aligning the received image to the one at the sender [3]–[8]. The alignment should be done in a semi-blind way: at destination one can use only the received image and the image hash to deal with the alignment problem; the reference image is not available. The challenging task of recovering the geometric transformations occurred on a received image from its signature motivates this work. The main contribution of the paper is in the design of a robust forensic hash method to better perform both image alignment and tampering localization.

Despite the fact that different robust alignment techniques have been proposed by computer vision researchers [11]–[13], these techniques are unsuitable in the context of forensic hashing, since a fundamental requirement is that the image signature should be as "compact" as possible to reduce the overhead of the network communications. To fit the underlying requirements, authors of [6] have proposed to exploit information extracted through Radon transform and scale space theory in order to estimate the parameters of the geometric transformations (i.e., rotation and scale). To make more robust

Manuscript received August 09, 2011; revised March 15, 2012; accepted March 24, 2012. Date of publication April 09, 2012; date of current version July 09, 2012. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Alex ChiChung Kot.

The authors are with the Department of Mathematics and Computer Science, University of Catania, Catania 95125, Italy (e-mail: battiato@dmi.unict.it; gfarinella@dmi.unict.it; emessina@dmi.unict.it; puglisi@dmi.unict.it).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIFS.2012.2194285

the alignment phase with respect to manipulations such as cropping and tampering, an image hash based on robust invariant features has been proposed in [7]. The latter technique extended the idea previously proposed in [8] by employing the bag of features (BOF) model to represent the features to be used as image hash. The exploitation of the BOF representation is useful to reduce the space needed for the image signature, by maintaining the performances of the alignment component. In [4] a more robust approach based on a cascade of estimators has been introduced; it is able to better handle the replicated matchings in order to make a more robust estimation of the orientation parameter. Moreover, the cascade of estimators allows a higher precision in estimating the scale factor. A more effective way to deal with the problem of wrong matchings has been proposed in [3], where a filtering strategy based on the scale-invariant feature transform (SIFT) dominant directions combined in cascade with a robust estimator based on a voting strategy on the parameter space is presented.

Taking into account the technique in [3], we propose to extend the underlying approach by encoding the spatial distribution of the image features to deal with highly textured and contrasted tampering patterns. The estimator is based on a voting procedure in the parameter space of the model used to recover the geometric transformation occurred into the manipulated image. As pointed out by the experimental results, the proposed method obtains satisfactory results with a significant margin in terms of estimation accuracy with respect to [4] and [7]. Moreover, by encoding spatial distribution of features, the proposed strategy outperforms the original method proposed in [3] when strongly contrasted and/or texture regions are contained into the image. We also propose a block-wise tampering detection based on histograms of oriented gradients representation which makes use of a non-uniform quantization to build the signature of each image block for tampering purposes. Experimental results confirm the effectiveness of the non-uniform quantization in terms of both compactness of the final hash signature and tampering detection accuracy.

To sum up, the main contributions of the paper can be summarized as follows.

1) The exploitation of replicated matchings from the beginning of the estimation process. Feature encoding by visual words allows a considerable gain in terms of compression but introduces the problem related to the replicated matchings. Lu *et al.* [7] simply consider only the single matching in the first estimation and refine the results later considering the remaining ones. Although the refinement can be useful, the correctness of the final estimation heavily depends on the first estimation (only a refinement is performed later). Our approach does not discard replicated matchings retaining their useful information. The ambiguity of the matching is solved considering all the possible pairs with the same id. As discussed also in [4], this solution introduces additional noise (i.e., incorrect pairs) that has to be properly taken into account employing the voting procedure.
2) The robust estimator based on a voting strategy. It is worth noting that, although a voting strategy in a parameter space cannot be considered completely novel, the function that permits us to map the matchings from the image coordinate space to the parameters space is novel. Specifically, the equations related to the similarity model have been combined and the computational complexity of the approach has been considerably reduced with respect to the simple application of the voting procedure in the four-dimensional parameters space.
3) Feature selection based on their spatial distribution. In previous works (Lu *et al.* [7], Roy *et al.* [8], Battiato *et al.* [4]) the features were selected considering only their contrast properties. This selection strategy is not robust against some malicious attacks (see Section IV). The proposed approach introduces a novel selection strategy that considers both contrast properties and spatial distribution of the features.
4) Complex dataset of tampered images. We built a dataset to be used for tampering detection consisting of 23 591 images belonging to different classes employing lots of transformations. A realistic dataset of tampered images (DB-Forgery [14]) has been also considered for testing purposes.

The remainder of the paper is organized as follows: Section II presents the proposed signature for the alignment component and the overall registration framework. Section III introduces the tampering component used by the system, whereas Section IV discusses the importance of embedding the spatial distribution of the features into the image hash. Sections V and VI report experiments and discuss both the registration performances and the tampering localization results. Finally, Section VIII concludes the paper with avenues for further research.

II. Registration Component

As previously stated, one of the common steps of tampering detection systems is the alignment of the received image. Image registration is crucial since all the other tasks (e.g., tampering localization) usually assume that the received image is aligned with the original one, and hence could fail if the registration is not properly done. Classical registration approaches [11]–[13] cannot be directly employed in the considered context due the limited information that can be used (i.e., original image is not available at destination and the image hash should be as short as possible).

The schema of the proposed registration component is shown in Fig. 1. As in [3], [4], and [7], we adopt a BOF-based representation [15] to reduce the dimensionality of the descriptors to be used as hash component for the alignment. Differently than [4] and [7], we employ a transformation model and a voting strategy to retrieve the geometric manipulation [16].

In the proposed system, a codebook is generated by clustering the set of SIFT [17] extracted on training images. The clustering procedure points out a centroid for each cluster. The set of centroids represents the codebook to be used during the image hash generation. The computed codebook is shared between sender and receiver (Fig. 1). It should be noted that the codebook is built only once, and then used for all the communications between sender and receiver (i.e., no extra overhead for each communication). The sender extracts SIFT features and sorts them

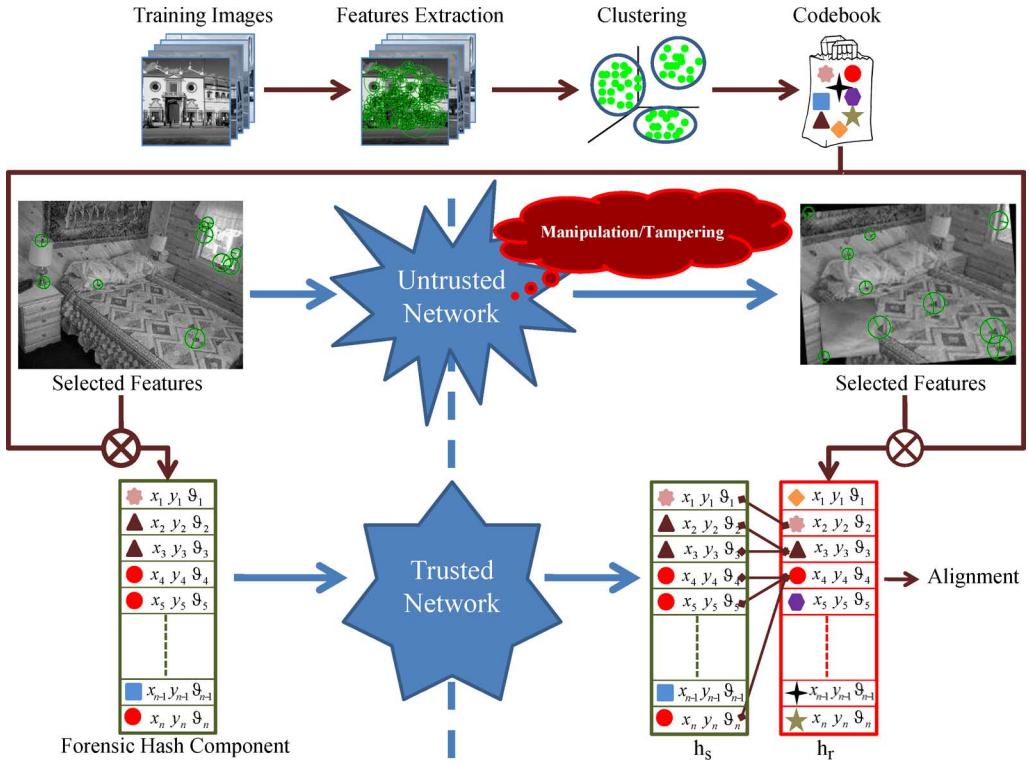


Fig. 1. Overall schema of proposed registration component.

in descending order with respect to their contrast values. Afterward, the top n SIFT are selected and associated to the id label corresponding to the closest centroid belonging to the shared codebook. Hence, the final signature for the alignment component is created by considering the id label, the dominant direction θ, and the keypoint coordinates (x, y) for each selected SIFT (Fig. 1). The source image and the corresponding hash component for the alignment (h_s) are sent to the destination. As in [5] the system assumes that the image is sent over a network consisting of possibly untrusted nodes, whereas the signature is sent upon request through a trusted authentication server which encrypts the hash in order to guarantee its integrity. During the untrusted communication the image could be manipulated for malicious purposes.

Once the image reaches the destination, the receiver generates the related hash signature for registration (h_r) by using the same procedure employed by the sender. Then, the entries of the hashes h_s and h_r are matched by considering the id values (see Fig. 1). Note that an entry of h_s may have more than one association with entries of h_r (and vice versa) due to possible replicated elements in the hash signatures. After matchings are obtained, the alignment is performed by employing a similarity transformation of keypoint pairs corresponding to matched hashes entries

$$x_r = x_s \sigma \cos\alpha - y_s \sigma \sin\alpha + T_x \tag{1}$$

$$y_r = x_s \sigma \sin\alpha + y_s \sigma \cos\alpha + T_y. \tag{2}$$

The previous transformation is used to model the geometrical manipulations which have been done on the source image during the untrusted communication. The model assumes that a point (x_s, y_s) in the source image I_s is transformed in a point (x_r, y_r) in the image I_r at destination with a combination of rotation (α), scaling (σ) and translation (T_x, T_y). The aim of the alignment phase is the estimation of the quadruple $(\widehat{\sigma}, \widehat{\alpha}, \widehat{T_x}, \widehat{T_y})$ by exploiting the correspondences $((x_s, y_s), (x_r, y_r))$ related to matchings between h_s and h_r. We propose to use a cascade approach; first, an initial estimation of the parameters $(\widetilde{\alpha}, \widetilde{T_x}, \widetilde{T_y})$ is accomplished through a voting procedure in the quantized parameter space $\overline{\alpha} \times \overline{T_x} \times \overline{T_y}$. Such a procedure is performed after filtering outlier matchings by taking into account the differences between dominant orientations of matched entries. The initial estimation is then refined considering only reliable matchings in order to obtain the final parameters $(\widehat{\alpha}, \widehat{T_x}, \widehat{T_y})$. Afterward, the scaling parameter $\widehat{\sigma}$ is estimated by means of the parameters $(\widehat{\alpha}, \widehat{T_x}, \widehat{T_y})$ which have been previously estimated on the reliable information obtained through the filtering described previously. The overall estimation procedure is detailed in the following.

Moving T_x and T_y on the left side and by considering the ratio of (1) and (2) the following equation holds:

$$\frac{x_r - T_x}{y_r - T_y} = \frac{x_s \cos\alpha - y_s \sin\alpha}{x_s \sin\alpha + y_s \cos\alpha}. \tag{3}$$

Solving (3) with respect to T_x and T_y we get the formulas to be used in the voting procedure

$$T_x = \left(\frac{x_s \cos\alpha - y_s \sin\alpha}{x_s \sin\alpha + y_s \cos\alpha}\right)(T_y - y_r) + x_r \tag{4}$$

$$T_y = \left(\frac{x_s \sin\alpha + y_s \cos\alpha}{x_s \cos\alpha - y_s \sin\alpha}\right)(T_x - x_r) + y_r. \tag{5}$$

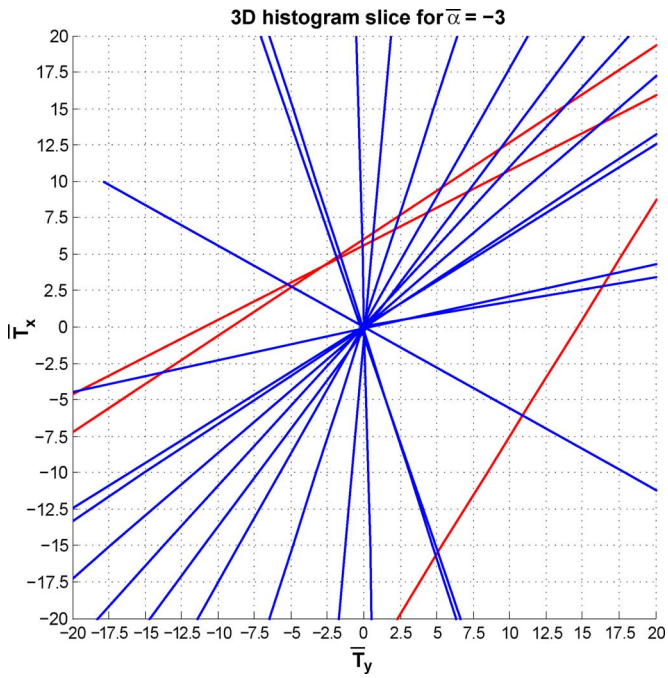


Fig. 2. Slice of 3-D histogram in correspondence of $\overline{\alpha} = -3$, obtained considering an image manipulated with parameters $(\sigma, \alpha, T_x, T_y) = (1, 3, 0, 0)$. For a fixed rotational angle $\overline{\alpha}$, each pair of coordinates (x_s, y_s) and (x_r, y_r) votes for a line in the quantized 2-D parameter space $(\overline{T_x}, \overline{T_y})$. Lines corresponding to inliers (blue) intersect in the bin $(\overline{T_x}, \overline{T_y}) = (0, 0)$, whereas remaining lines (red) are related to outliers.

For a given value of α, each pair of coordinates (x_s, y_s) and (x_r, y_r) can be used together with (4) and (5) to represent two lines in the parameter space $\alpha \times T_x \times T_y$.

The initial estimation of the parameters $(\widetilde{\alpha}, \widetilde{T_x}, \widetilde{T_y})$ is hence obtained by considering the densest bin of a 3-D histogram in the quantized parameter space $\overline{\alpha} \times \overline{T_x} \times \overline{T_y}$. This means that the initial estimation of $(\widetilde{\alpha}, \widetilde{T_x}, \widetilde{T_y})$ is accomplished in correspondence of the maximum number of intersections between lines generated by matched keypoints (Fig. 2). Actually (4) and (5) are two different ways of representing the same line ($y = mx + c$, and $x = (y - c)/m$) and are not two different lines. It may be possible that due to the quantization of the parameters, the "votes" based on these two equations fall into different bins, but essentially, every point is counted twice in the estimation method. Considering only one (4) or (5), due to the quantization of the parameters, some lines intersecting the correct bin could not be considered in the final count (it depends of the angular coefficient). Considering both lines, at least one count is recorded. As stated, to discard outliers (i.e., wrong matchings) the information coming from the dominant directions (θ) of SIFT is used during the voting procedure. In particular, $\Delta\theta = \theta_r - \theta_s$ is a rough estimation of the rotational angle α. Hence, for each fixed triplet $(\overline{\alpha}, \overline{T_x}, \overline{T_y})$ of the quantized parameter space, the voting procedure considers only the matchings between h_s and h_r such that $|\Delta\theta - \overline{\alpha}| < t_\alpha$. The threshold value t_α is chosen to consider only matchings with a rough estimation $\Delta\theta$ which is closer to the considered $\overline{\alpha}$ (e.g., consider just matchings with a small initial error of ± 3.5 degree). The proposed approach is summarized in Algorithm 1. The algorithm gives an initial estimation of rotation angle $\widetilde{\alpha}$ and translation vector $(\widetilde{T_x}, \widetilde{T_y})$ by taking into account the quantized values used to build the 3-D histogram into the parameter space.

Algorithm 1: Voting procedure

Input:
The set M of matching pairs $((x_s, y_s), (x_r, y_r))$
The filtering threshold t_α
Output:
The initial estimation $(\widetilde{\alpha}, \widetilde{T_x}, \widetilde{T_y})$
The selected bin $(i_{\max}, j_{\max}, k_{\max})$
begin
$Initialize\ Votes(i, j, k) \leftarrow 0\ \forall i, j, k$
for $\overline{\alpha} \leftarrow -180$ **to** 180 **do**
$V_\alpha \leftarrow \{((x_s, y_s), (x_r, y_r)) || (\theta_r - \theta_s) - \overline{\alpha}| < t_\alpha\}$
for each $((x_s, y_s), (x_r, y_r)) \in V_\alpha$ **do**
for $\overline{T_y} \leftarrow min_{T_y}$ **to** max_{T_y} **do**
$T_x \leftarrow ((x_s \cos\overline{\alpha} - y_s \sin\overline{\alpha})/(x_s \sin\overline{\alpha} + y_s \cos\overline{\alpha}))(\overline{T_y} - y_r) + x_r$
$\overline{T_x} \leftarrow Quantize(T_x)$
$(i, j, k) \leftarrow QuantizedVal2Bin(\overline{\alpha}, \overline{T_x}, \overline{T_y})$
$Votes(i, j, k) \leftarrow Votes(i, j, k) + 1$
for $\overline{T_x} \leftarrow min_{T_x}$ **to** max_{T_x} **do**
$T_y \leftarrow ((x_s \sin\overline{\alpha} + y_s \cos\overline{\alpha})/(x_s \cos\overline{\alpha} - y_s \sin\overline{\alpha}))(\overline{T_x} - x_r) + y_r$
$\overline{T_y} \leftarrow Quantize(T_y)$
$(i, j, k) \leftarrow QuantizedVal2Bin(\overline{\alpha}, \overline{T_x}, \overline{T_y})$
$Votes(i, j, k) \leftarrow Votes(i, j, k) + 1$
$(i_{\max}, j_{\max}, k_{\max}) \leftarrow SelectBin(Votes)$
$(\widetilde{\alpha}, \widetilde{T_x}, \widetilde{T_y}) \leftarrow Bin2QuantizedVal(i_{\max}, j_{\max}, k_{\max})$
end

To further refine the initial estimation we further exploit the m matchings which have generated the lines intersecting in the selected bin (see Algorithm 1). Specifically, for each pair $((x_{s,i}, y_{s,i}), (x_{r,i}, y_{r,i}))$ corresponding to the selected bin we consider the following translation vectors:

$$(\widehat{T_{x,i}}, \widehat{T_{y,i}}) = \left(\left(\frac{x_{s,i}\cos\widetilde{\alpha} - y_{s,i}\sin\widetilde{\alpha}}{x_{s,i}\sin\widetilde{\alpha} + y_{s,i}\cos\widetilde{\alpha}}\right)(\widetilde{T_y} - y_{r,i}) + x_{r,i}, \widetilde{T_y}\right) \quad (6)$$

$$(\widehat{T_{x,i}}, \widehat{T_{y,i}}) = \left(\widetilde{T_x}, \left(\frac{x_{s,i}\sin\widetilde{\alpha} + y_{s,i}\cos\widetilde{\alpha}}{x_{s,i}\cos\widetilde{\alpha} - y_{s,i}\sin\widetilde{\alpha}}\right)(\widetilde{T_x} - x_{r,i}) + y_{r,i}\right) \quad (7)$$

together with the subsequent equations

$$x_{r,i} = x_{s,i}\sigma_i \cos\alpha_i - y_{s,i}\sigma_i \sin\alpha_i + T_{x,i} \quad (8)$$

$$y_{r,i} = x_{s,i}\sigma_i \sin\alpha_i + y_{s,i}\sigma_i \cos\alpha_i + T_{y,i}. \quad (9)$$

The parameters values $(\widetilde{\alpha}, \widetilde{T_x}, \widetilde{T_y})$ in (6) and (7) are obtained through the voting procedure (see Algorithm 1).

Solving (8) and (9) with respect to $a_i = \sigma_i \cos\alpha_i$ and $b_i = \sigma_i \sin\alpha_i$ we obtain

$$a_i = \frac{y_{r,i}y_{s,i} + x_{r,i}x_{s,i} - x_{s,i}T_{x,i} - y_{s,i}T_{y,i}}{x_{s,i}^2 + y_{s,i}^2} \quad (10)$$

$$b_i = \frac{x_{s,i}y_{r,i} - x_{r,i}y_{s,i} + y_{s,i}T_{x,i} - x_{s,i}T_{y,i}}{x_{s,i}^2 + y_{s,i}^2}. \quad (11)$$

Since the ratio b_i/a_i is by definition equal to $\tan\alpha_i$, for each pair of matched keypoints we can estimate $\widehat{\alpha_i}$ by exploiting the following formula:

$$\widehat{\alpha_i} = \frac{1}{2}\arctan\left(\frac{x_{s,i}y_{r,i} - x_{r,i}y_{s,i} + y_{s,i}\widehat{T_{x,i}} - x_{s,i}\widetilde{T_{y,i}}}{y_{r,i}y_{s,i} + x_{r,i}x_{s,i} - x_{s,i}\widehat{T_{x,i}} - y_{s,i}\widetilde{T_{y,i}}}\right) + \frac{1}{2}\arctan\left(\frac{x_{s,i}y_{r,i} - x_{r,i}y_{s,i} + y_{s,i}\widetilde{T_{x,i}} - x_{s,i}\widehat{T_{y,i}}}{y_{r,i}y_{s,i} + x_{r,i}x_{s,i} - x_{s,i}\widetilde{T_{x,i}} - y_{s,i}\widehat{T_{y,i}}}\right). \quad (12)$$

Once $\widehat{\alpha_i}$ is obtained, (13) [derived from (8) and (9) by considering (6) and (7)] is used to estimate $\widehat{\sigma_i}$.

$$\widehat{\sigma_i} = \frac{1}{4}\frac{x_{r,i} - \widehat{T_{x,i}}}{x_{s,i}\cos\widehat{\alpha_i} - y_{s,i}\sin\widehat{\alpha_i}} + \frac{1}{4}\frac{y_{r,i} - \widetilde{T_{y,i}}}{x_{s,i}\sin\widehat{\alpha_i} + y_{s,i}\cos\widehat{\alpha_i}} + \frac{1}{4}\frac{x_{r,i} - \widetilde{T_{x,i}}}{x_{s,i}\cos\widehat{\alpha_i} - y_{s,i}\sin\widehat{\alpha_i}} + \frac{1}{4}\frac{y_{r,i} - \widehat{T_{y,i}}}{x_{s,i}\sin\widehat{\alpha_i} + y_{s,i}\cos\widehat{\alpha_i}}. \quad (13)$$

The previous method produces a quadruple $(\widehat{\sigma_i}, \widehat{\alpha_i}, \widehat{T_{x,i}}, \widehat{T_{y,i}})$ for each matching pair $((x_{s,i}, y_{s,i}), (x_{r,i}, y_{r,i}))$ corresponding to the bin selected with the Algorithm 1. The final transformation parameters $(\widehat{\sigma}, \widehat{\alpha}, \widehat{T_x}, \widehat{T_y})$ to be used for the registration are computed by averaging over all the m produced quadruples. Since the quadruples $(\widehat{\sigma_i}, \widehat{\alpha_i}, \widehat{T_{x,i}}, \widehat{T_{y,i}})$ are obtained in correspondence of matchings filtered through the voting procedure (i.e., outliers have been removed), the simple average on the quadruples is robust enough as confirmed by the experimental results. It is worth noting that the proposed registration method, by combining (1) and (2) (similarity transformation), obtains a considerable reduction of computational complexity and memory usage. The combination of (1) and (2) allows us to use a 3-D histogram to estimate four parameters instead of a 4-D histogram as in the case of a naive implementation based on classic Hough transform [18].

Although the proposed procedure was built considering a similarity model, it can be extended to deal with affine transformations

$$x_r = ax_s + by_s + T_x \quad (14)$$

$$y_r = cx_s + dy_s + T_y \quad (15)$$

where (x_s, y_s) and (x_r, y_r) are points in the source image I_s and transformed image I_r, respectively, and a, b, c, d, T_x, T_y are the six real parameters of the affine transformation. Specifically, the voting procedure based on the similarity model is used to select the matching pairs $((x_{s,i}, y_{s,i}), (x_{r,i}, y_{r,i}))$ corresponding to the bin selected by Algorithm 1. These pairs are then used to estimate the parameters of the affine transformation by using the least squares algorithm. The effectiveness of the proposed solution, together with its limits, is reported in Section V-A.

It should be noted that some id values may appear more than once in h_s and/or in h_r. Even if a small number of SIFT are selected during the image hash generation process, the conflict due to replicated id can arise. As experimentally demonstrated in the next section, by retaining the replicated id values the accuracy of the estimation increases, and the number of "unmatched" images decreases (i.e., image pairs that the algorithm is not able to process because there are no matchings between h_s and h_r). The described approach considers all the possible matchings in order to preserve the useful information. The correct matchings are hence retained but other wrong pairs could be generated. Since the noise introduced by considering correct and incorrect pairs can badly influence the final estimation results, the presence of possible wrong matchings should be considered during the estimation process. The approach described in this paper deals with the problem of wrong matchings combining in cascade a filtering strategy based on the SIFT dominant direction (θ) with a robust estimator based on a voting strategy on the parameter space of a geometric transformation model. In this way, the information of spatial position of keypoints and their dominant orientations are jointly considered, and the scale factor is estimated only at the end of the cascade on reliable information. Note that, as already shown in [4], the replication of the matchings makes unreliable the direct estimation of the scale factor; hence it is estimated at the end of the process on the filtered data. As demonstrated by the experiments, replicated matchings help to better estimate the rotational parameter, whereas the introduced cascade approach allows robustness in estimating the scale factor.

III. Tampering Localization Component

Once the alignment has been performed as described in Section II, the image is analyzed to detect tampered regions. Tampering localization is the process of localizing the regions of the image that have been manipulated for malicious purposes to change the semantic meaning of the visual message. The tampering manipulation typically changes the properties (e.g., edges distributions, colors, textures, etc.) of some image regions. To deal with this problem the image is usually divided into non-overlapping blocks which are represented through feature vectors computed taking into account their content. The feature vectors computed at the source are then sent to a destination where these are used as forensic hash for the tampering detection component of the system. The check to localize tampered blocks is performed by the receiver taking into account the received signature and the one computed (with the same procedure employed by the sender) on the received image. The comparison of the signatures is performed block-wise after the alignment (see Section II).

Among the different representations used to describe the content of a block, the histogram of gradients has been successfully applied in the context of image tampering localization [7], [8]. The image is convolved with simple derivative filters along the

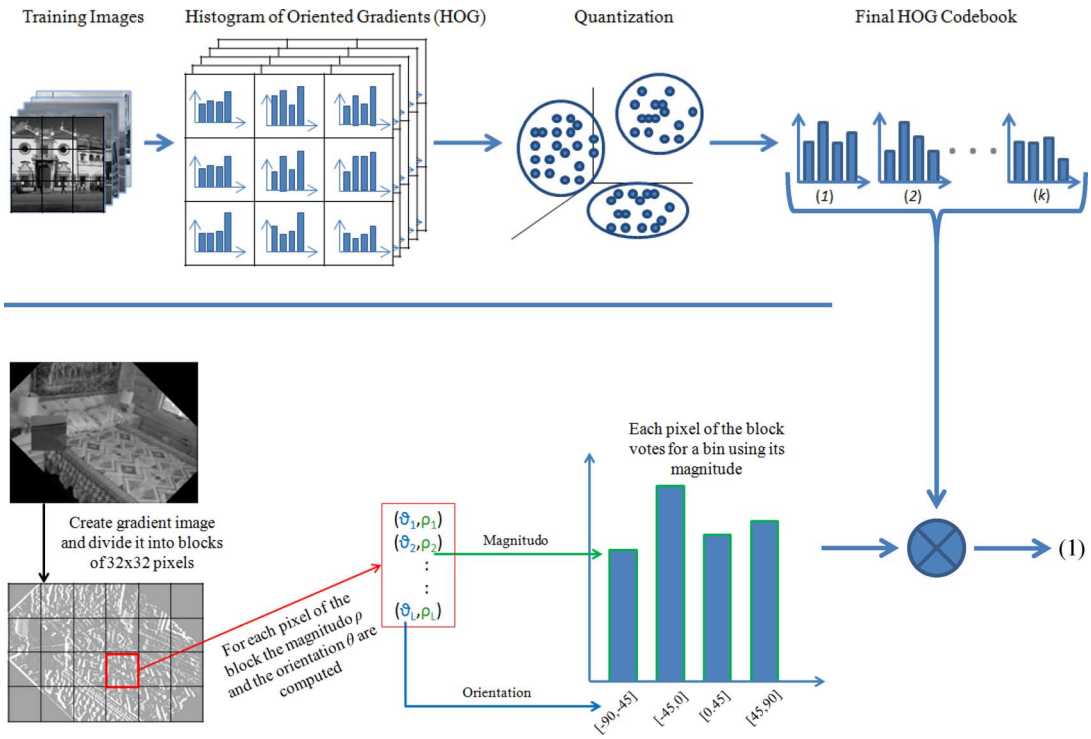


Fig. 3. Schema of proposed block description process. First each block is described by a histogram of gradient (HOG), then it is associated to a prototype belonging to a vocabulary previously generated on training samples.

horizontal and vertical direction obtaining magnitude and orientation for each pixel. For each block a histogram is built considering a set of quantized orientations. The orientation bins can be in the range $[0°, 180°]$ ("unsigned" gradient) or $[0°, 360°]$ ("signed" gradient). Each pixel of the considered block votes for a bin of the histogram through a function of the gradient magnitude of the pixel (e.g., root, square root), the magnitude itself or a clipped version representing the presence of an edge. Finally, the obtained histograms are normalized and then quantized in order to obtain a more compact hash for each block (e.g., 15 bit for each block [8]). The comparison of histograms of corresponding blocks is usually performed through a similarity measure (e.g., Euclidean distance, minimum intersection, etc.) and a thresholding procedure.

Although the histograms can be simply quantized in a uniform way [8], in this paper we propose to exploit a non-uniform quantization by computing a vocabulary of histograms of orientation. In the proposed approach, the orientation histograms related to blocks extracted on training images are clustered taking into account their similarity (Euclidean distance). The prototypes (i.e., centroids) of the produced clusters are retained to form the vocabulary. Images at sender and receiver are first split into blocks and then each block is associated to the closest histogram prototype belonging to the shared vocabulary. Comparison between signatures is made by simply comparing the IDs of corresponding blocks after the alignment. The overall scheme related to the generation of the block representation is reported in Fig. 3. Experimental results confirm the effectiveness of the proposed non-uniform quantization in terms of both compactness of the final hash signature and tampering detection accuracy.

IV. Deluding Registration in Tampering Detection Systems

As stated in Section I, the image signature to be used into the alignment component should be robust against malicious manipulations. Moreover, the image hash should be robust with respect to the different visual content to be encoded (textures, contrast variations, etc.). Tampering detection systems often employ robust invariant local features (e.g., SIFT) to deal with a large range of image manipulations during alignment phase. As described in this paper, in the context of communication a small subset of the robust features is retained to compose the image hash for the alignment component. These features are typically selected considering their contrast properties (higher stability) without taking into account their spatial distribution in the image (see Section II). In this section we show, through an example, that it is possible to design *ad hoc* attacks to SIFT-based hashing methods. Specifically, we show that a simple tampering, obtained by adding a patch containing a highly texturized and contrasted pattern to the original image, deludes the typical SIFT-based registration systems by concealing all the true local features useful to properly align the received image. Fig. 4 shows an example of malicious tampering which deludes the typical SIFT-based systems presented in [3], [4], [7], and [8]. In Fig. 4(a) the image at the source is shown, whereas the malicious pattern added during the transmission is reported in Fig. 4(b). Sixty SIFT selected by the approach discussed in Section II, at both source and destination, are shown in Fig. 4(c) and Fig. 4(d). As demonstrated by the figures, all the SIFT extracted by the sender which are used to build the alignment signature are concealed at destination, since all the 60 SIFT ex-

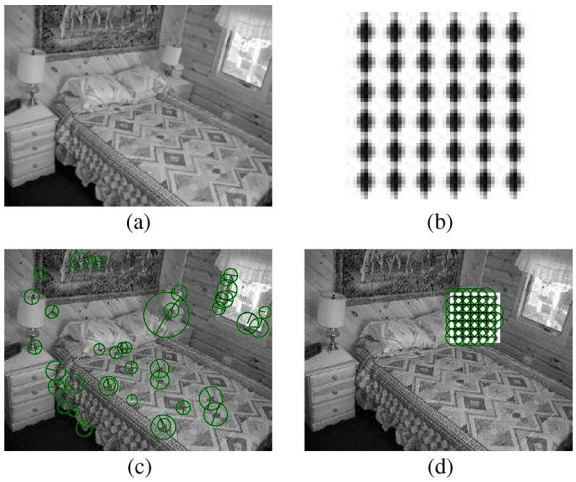


Fig. 4. Concealing true local features: (a) Original image, (b) tampering pattern, (c) 60 SIFT selected by ordering contrast values on the original image. (d) The 60 SIFT selected by ordering contrast values on tampered image.

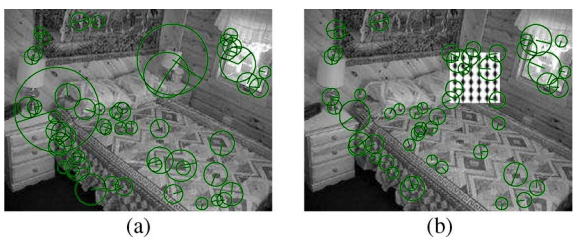


Fig. 5. Cluster-based feature selection. (a) The 60 SIFT selected considering spatial clustering and contrast values on original image. (b) The 60 SIFT selected with spatial clustering and contrast values on tampered image.

tracted by the receiver lie on the tampering pattern. The alignment procedure is hence invalidated, and all further processing to verify the authenticity of the image, to localize the tampered regions, and in general to tell the history of the manipulations of the image, will be unreliable. In order to improve the robustness of the registration phase we suggest to modify the initial step of feature selection by considering also the spatial distribution of the keypoints in the image. As already reported in [19]–[21] the spatial distribution of the features on the entire image is a property that registration algorithms have to take into account. The proposed spatial-based selection process works as follows: first the SIFT are extracted and then grouped taking into account the spatial coordinates of the obtained feature keypoints. The grouping can be done employing a classic clustering algorithm (k-means, hierarchical clustering, etc.). For each cluster, the best SIFT in terms of contrast value is selected. In this way, the proposed feature selection procedure allows us to extract k high contrasted features (corresponding to the k clusters) well distributed in the image in terms of spatial position. As shown in Fig. 5 and as pointed out by the experiments, this strategy allows coping with tampering obtained by adding a high textured and contrasted patterns.

V. Registration Results

This section reports a number of experiments on which the proposed approach (see Sections II and IV) has been tested and compared with respect to [3], [4], and [7]. Tests have been performed considering a subset of the following datasets: [22] and *DBForgery 1.0* [14]. The former dataset is made up of 4485 images (average size of 244 × 272 pixels) belonging to 15 different scene categories at basic level of description: bedroom, suburb, industrial, kitchen, living room, coast, forest, highway, inside city, mountain, open country, street, tall building, office, store. The composition of the considered dataset allows for coping with the high scene variability needed to properly test methods in the context of application of this paper. The training set used in the experiments is built through a random selection of 150 images from the aforementioned dataset. Specifically, ten images have been randomly sampled from each scene category. In order to consider also realistic tampering 29 images have been selected from *DBForgery 1.0* and added to the training dataset. The test set consists of 21 980 images generated through the application of different manipulations on the training images. Training and test sets are available for experimental purposes.[1] The following image transformations have been considered (Table I): cropping, rotation, scaling, translation, seam carving, tampering, linear photometric transformation and JPEG compression. The considered transformations are typically available on image manipulation software. Tampering on the [22] subset has been performed through the swapping of blocks (50 × 50) between two images randomly selected from the training set. A realistic tampering on *DBForgery 1.0* images has been obtained by simply using the script provided together with the dataset [14]. Images obtained through various combinations of the basic transformations, as well as the ones obtained adding the malicious tampering shown in Fig. 4(b), have been included into the test set to make more challenging the task to be addressed.

TABLE I
Image Transformations

Operations	Parameters
Rotation (α)	3, 5, 10, 30, 45 degrees
Scaling (σ)	factor = 0.5, 0.7, 0.9, 1.2, 1.5
Orizontal Traslation (T_x)	5, 10, 20 pixels
Vertical Traslation (T_y)	5, 10, 20 pixels
Cropping	19%, 28%, 36%, of entire image
Tampering	block size 50x50
Malicious Tampering	block size 50x50
Linear Photometric Transformation (a*I+b)	a = 0.90, 0.95, 1, 1.05, 1.10 b = -10, -5, 0, 5, 10
Compression	JPEG Q=10
Seam Carving	10%, 20%, 30%
Realistic Tampering [16]	
Various combinations of above operations	

The registration results obtained employing the proposed alignment approach (with and without spatial clustering) with hash component of different size (i.e., different number of SIFT) are reported in Table II. The results are related to the alignment of the test images on which both versions of the proposed method are able to find matching between h_s and h_r.

To demonstrate the effectiveness of the proposed approach, and to highlight the contribution of both the replicated matchings and cascade filtering during the estimation, we have performed comparative tests by considering our method (with and

[1]http://iplab.dmi.unict.it/download/Forensics/datasetTIFS.zip

TABLE II
REGISTRATION RESULTS OF PROPOSED APPROACH

	Proposed approach							
Number of SIFT	15		30		45		60	
Unmatched Images	5.00%		1.90%		1.04%		0.83%	
Spatial Clustering	without	with	without	with	without	with	without	with
Mean Error α	1.3826	1.9911	0.8986	0.8627	0.6661	0.6052	0.5658	0.4518
Mean Error σ	0.0462	0.0593	0.0306	0.0302	0.0241	0.0200	0.0208	0.0164
Mean Error T_x	2.7672	3.3191	1.8621	1.9504	1.5664	1.5626	1.4562	1.4227
Mean Error T_y	2.6650	3.2428	1.9409	2.0750	1.7009	1.7278	1.6008	1.5944

TABLE III
COMPARISON WITH RESPECT TO UNMATCHED IMAGES

	Unmatched Images			
Number of SIFT	15	30	45	60
Lu et al. [7]	7.87%	2.77%	1.52%	1.16%
Battiato et al. [4]	0.86%	0.48%	0.25%	0.08%
Proposed approach without spatial clustering	3.00%	1.35%	0.87%	0.73%
Proposed approach with spatial clustering	2.53%	0.64%	0.18%	0.10%

TABLE IV
AVERAGE ROTATIONAL ERROR

	Mean Error α			
Number of SIFT	15	30	45	60
Unmatched Images	10.99%	3.85%	2.02%	1.56%
Lu et al. [7]	7.3311	7.9970	7.8600	7.4125
Battiato et al. [4]	3.4372	2.4810	2.4718	1.9581
Proposed approach without spatial clustering	1.1591	0.8206	0.5485	0.4634
Proposed approach with spatial clustering	1.7933	0.8288	0.5735	0.4318

TABLE V
AVERAGE SCALING ERROR

	Mean Error σ			
Number of SIFT	15	30	45	60
Unmatched Images	10.99%	3.85%	2.02%	1.56%
Lu et al. [7]	0.0619	0.0680	0.0625	0.0592
Battiato et al. [4]	0.0281	0.0229	0.0197	0.0179
Proposed approach without spatial clustering	0.0388	0.0281	0.0214	0.0183
Proposed approach with spatial clustering	0.0541	0.0287	0.0195	0.0161

without spatial clustering), the approach proposed in [7] and the method proposed in [4] which exploits both replicated matchings and a cascade filtering approach. The results of Lu *et al.* [7] presented in this paper have been obtained considering the original code of the authors. The threshold t_α used in our approach to filter the correspondences (see Section II) has been set to 3.5°. The quantized values $\overline{T_x}$ and $\overline{T_y}$ needed to evaluate the right side of (4) and (5) in Algorithm 1 have been quantized considering a step of 2.5 pixels (see Fig. 2). In [4] a histogram with bin size of 7° ranging from $-180°$ to 180° has been used for the rotation estimation step, whereas a histogram with bin size equal to 0.05 ranging from 0 to $\max_\sigma = 10$ was employed to estimate the scale. Finally, a codebook with 1000 visual words has been employed to compare the different approaches. The codebook has been learned through k-means clustering on the overall SIFT descriptors extracted on training images.

First, let us examine the typically cases on which the considered registration approaches are not able to work. Two cases can be distinguished: 1) no matchings are found between the hash built at the sender (h_s) and the one computed by the receiver (h_r); 2) all the matchings are replicated. The first problem can be mitigated considering a higher number of features (i.e., SIFT). The second one is solved by allowing replicated matchings (see Section II).

As reported in Table III, by increasing the number of SIFT points the number of unmatched images decreases (i.e., image pairs that the algorithm is not able to process because there are no matchings between h_s and h_r) for all the approaches. In all cases the percentage of images on which our algorithm (with and without spatial clustering) is able to work is higher than the one obtained by the approach proposed in [7]. Despite the fact that percentages of unmatched images obtained by [4] is less than the one obtained by the proposed approach, the tests reported in the following reveal that our method strongly outperforms the other two in terms of parameter estimation error and robustness with respect to the different transformations.

Tables IV and V show the results obtained in terms of rotational and scale estimation through mean absolute error. In order to properly compare the methods, the results have been computed taking into account the images on which all approaches were able to work (the number of unmatched images is reported into the tables). The proposed approach (with and without spatial clustering) outperforms [4] and [7] obtaining a considerable gain both in terms of rotational and scaling accuracy. Moreover, the performance of our approach significantly improves with the increasing of the extracted feature points (SIFT). On the contrary, the technique in [7] is not able to takes advantage from the information coming from an increasing number of extracted SIFT points. The rotational estimation gain obtained by employing our approach instead of the one in [7] is about 6° exploiting the minimum number of SIFT, and reaches 7° with 60 SIFT. A good gain in terms of performance is also obtained with respect to the scale factor (Table V).

To better compare the different approaches, the Regression Error Characteristic Curves (REC) method has been employed [23]. The REC curve is the cumulative distribution function of the error. The area over the curve is a biased estimation of the expected error of an employed estimation model. In Fig. 6 the comparison through REC curves is shown for both rotation and scaling factor. The results have been obtained considering 60 SIFT to build the alignment signature of training and test images. REC curves confirm the effectiveness of the proposed method (with and without spatial clustering) which outperforms the other approaches.

Additional experiments have been performed in order to examine the dependence of the average rotational and scaling error with respect to the rotation and scale transformation parameters respectively. Looking at the results obtained with the proposed approach one can observe that the rotational estimation error slightly increases with the rotation angle [Fig. 7(a)]. For the scale transformation, the error has lower values in the proximity of one (no scale change) and increases considering scale factors higher or lower than one [Fig. 7(b)]. The experiments reported in Fig. 7(a) and (b) consider only single transformations. Specifically, Fig. 7(a) considers only the set of images that have been rotated whereas Fig. 7(b) the scaled ones. It should be noted that the proposed registration approach obtains the best performances in all cases for both rotational and scaling estimation.

Finally, we have performed comparative tests to highlight the contribution of adding spatial clustering during SIFT selection (see Section IV). To this aim the test images obtained by

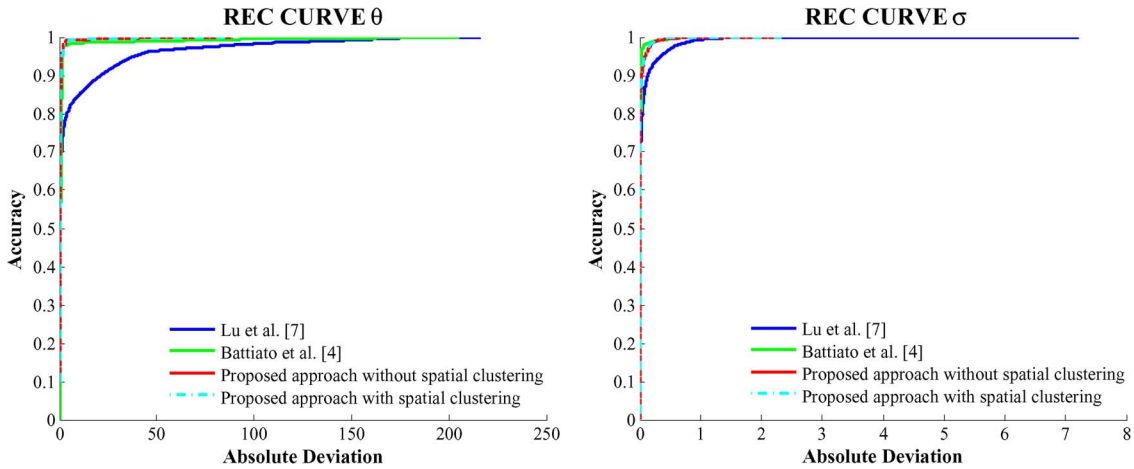


Fig. 6. REC curves comparison.

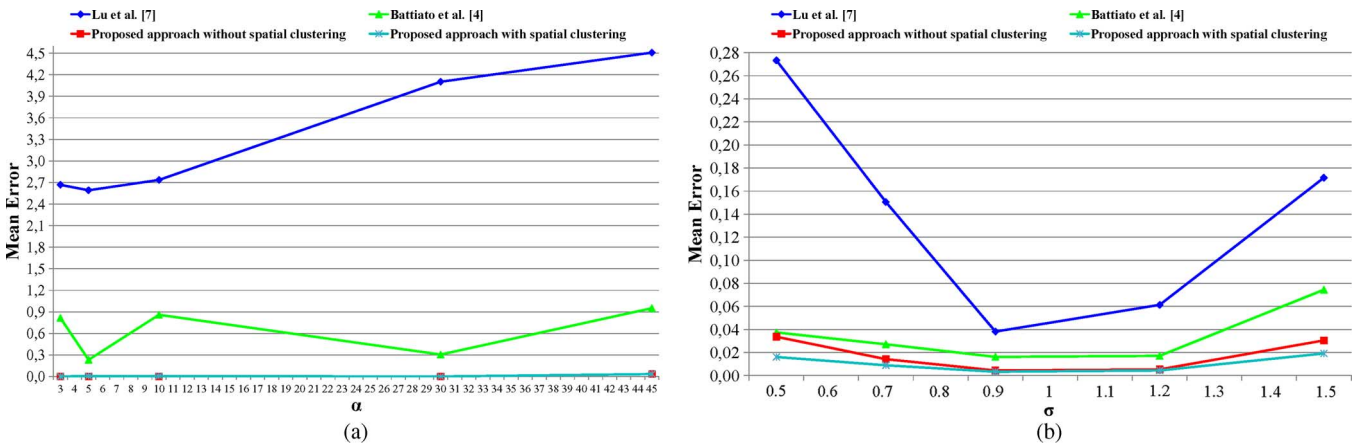


Fig. 7. Comparison on single transformation (60 SIFT). (a) Average rotation error at varying of the rotation angle. (b) Average scaling error at varying of the scale factor.

adding the malicious pattern [Fig. 4(b)] have been considered. Table VI shows the percentage of malicious manipulated images that cannot be considered by the different approaches (i.e., there are no matchings between h_s and h_r), whereas Tables VII and VIII report the results obtained by the different approaches on the malicious manipulated images on which matchings between h_s and h_r have been found. In Table IX the different approaches are compared taking into account only the images on which all the approaches are able to find matchings between h_s and h_r. The results demonstrate that robustness can be obtained embedding spatial information during the selection of the features to be used as a signature for the alignment component. The embedded spatial information helps to deal with tampered images obtained by adding patches containing a highly texturized and contrasted pattern. It is worth noting that the proposed *ad hoc* attack has been considered to underline the weakness of selecting the features considering only their contrast properties. Although the considered patch is pretty evident, similar results should be obtained by other smaller patches distributed in the images in non-uniform regions.

The compactness of the hash code is an important feature of the alignment method. In the following a brief analysis in terms of number of bits is reported. Considering the parameters used

TABLE VI
PERCENTAGE OF UNMATCHED IMAGES OBTAINED THROUGH MALICIOUS MANIPULATION

	Unmatched Images			
Number of SIFT	**15**	**30**	**45**	**60**
Lu et al. [7]	90.50%	87.71%	81.01%	73.74%
Battiato et al. [4]	68.72%	54.19%	29.61%	9.50%
Proposed approach without spatial clustering	87.15%	86.03%	74.86%	64.25%
Proposed approach with spatial clustering	0%	0%	0%	0%

TABLE VII
AVERAGE ROTATIONAL ERROR ON IMAGES OBTAINED THROUGH MALICIOUS MANIPULATION

	Mean Error α			
Number of SIFT	**15**	**30**	**45**	**60**
Lu et al. [7]	85.6844	79.9884	88.4555	97.4700
Battiato et al. [4]	86.9447	92.0451	92.5144	91.8478
Proposed approach without spatial clustering	35.6087	33.6800	42.5111	38.5156
Proposed approach with spatial clustering	1.2458	0.0000	0.0000	0.0000

in our tests, the final size of the hash code for the alignment component is 44 bits for each selected keypoint. Our vocabulary contains 1000 visual words, each id can be then represented by ten bits. Dominant orientation, ranging from 0° to 360°, has been represented by using ten bits. Finally, the coordinates of

TABLE VIII
AVERAGE SCALING ERROR MALICIOUS ON IMAGES OBTAINED THROUGH MALICIOUS MANIPULATION

	Mean Error σ			
Number of SIFT	**15**	**30**	**45**	**60**
Lu et al. [7]	0.2868	0.2934	0.2920	0.3482
Battiato et al. [4]	0.3141	0.3453	0.3505	0.3493
Proposed approach without spatial clustering	0.8249	0.7891	0.9284	0.7706
Proposed approach with spatial clustering	0.0193	0.0005	0.0002	0.0006

TABLE IX
COMPARISON OF DIFFERENT APPROACHES ON IMAGES OBTAINED THROUGH MALICIOUS MANIPULATION

Number of SIFT	**45**		**60**	
Unmatched Images	**92.74%**		**89.39%**	
Mean Error	**α**	**σ**	**α**	**σ**
Lu et al. [7]	81.1994	0.2750	88.2215	0.3126
Battiato et al. [4]	96.5480	0.4163	88.3088	0.3058
Proposed approach without spatial clustering	32.3846	0.7285	34.9474	0.6213
Proposed approach with spatial clustering	0.0000	0.0001	0.0000	0.0009

the detected interest point are described by using 12 bits for each component (x and y). Considering, as an example, 60 keypoints the final hash size is 330 bytes. Lu *et al.* consider a five parameters vector for each selected keypoint. As reported in [7], each five-parameter vector takes around 50 bits. Considering 60 SIFT 375 bytes must be used for the registration component. Battiato *et al.* technique [4], taking into account only id, keypoint scale and dominant orientation uses 225 byte to describe 60 SIFT.

A. Dealing With Affine Transformations

As already stated in Section II, we extended our approach considering a final estimation step based on the affine model [see (14) and (15)]. To assess the performances of this approach several tests have been conducted on the previously generated dataset (see Table I). In Table X the comparison with respect to the versions based on a similarity model [see (1) and (2)] is reported. The affine-based approach obtains, on average, results similar to the other considered approaches based on a similarity transformation.

A second experiment has been performed to test the accuracy of the proposed approach in presence of two typically affine transformations. The first one is the shearing transformation defined as follows:

$$x_r = x_s + k y_s \tag{16}$$

$$y_r = y_s \tag{17}$$

where (x_s, y_s) and (x_r, y_r) are points in the source image I_s and transformed image I_r, respectively, and k is the shear parameter.

The anisotropic scaling has been also considered

$$x_r = \sigma_x x_s \tag{18}$$

$$y_r = \sigma_y y_s \tag{19}$$

where σ_x and σ_y represent the scaling factor along the x and y axis, respectively.

A novel test dataset has been hence built by using (16) and (17) for shear and (18) and (19) for the anisotropic scale (see Table XI). As reported in Tables XII and XIII the accuracy of the proposed affine solution, although dependent on the degree of the affine warping, can be considered satisfactory. Finally, the results obtained with the affine model by considering the dataset containing all the transformation in Tables I and XI are reported in Table XIV. The obtained results confirm the effectiveness of the proposed approach.

VI. TAMPERING RESULTS

Despite the main contribution of this paper being related to the design of a robust registration procedure, tampering detection has been included for completeness and to experimentally show that a good alignment step is fundamental to obtain satisfactory results. The final step of the proposed framework is the tampering detection, i.e., the localization of the image regions that have been modified for malicious purposes. As already stated in Section III we adopt an image representation based on histogram of gradients to properly describe image blocks (see Fig. 3). Our technique has been implemented making use of some useful routines already available for HOG-like feature computing [24]. More specifically, image gradients are computed by using the simple 1-D filters $[-1\ 0\ 1]$ and $[-1\ 0\ 1]^T$. These gradients are then used to obtain magnitude and orientation for each image pixel. The image is then divided into non-overlapped blocks 32×32 and an orientation histogram is built considering four orientations in the range $[-90, 90]$ degrees ("unsigned" gradient). Each pixel of the considered block votes for a bin of the histogram with its magnitude. Finally, the obtained histograms are normalized and quantized. Two kinds of quantization have been considered to test the proposed framework: uniform and non-uniform. The uniform quantization [8] simple uses a fixed number of bits (e.g., three in our tests) to describe a single bin of the histogram. Considering four orientation, 12 bits are used to describe a single block. The proposed non-uniform quantization makes use of a precomputed vocabulary of histograms of orientations containing k prototypes ($\log_2 k$ bit for each block). This vocabulary has been obtained making use of a simple k-means clustering. All the aforementioned parameters have been derived through experimental analysis taking into account the specificity of the considered problem (e.g., the hash has to be as compact as possible).

In order to validate the tampering localization step several experiments have been performed considering a subset of the dataset previously used for alignment purposed. This subset consists of 2714 images containing a tampering patch. First we tested the tampering detection performances considering a uniform quantization of the orientation histograms. Euclidean distance between represented blocks is used to discriminate between tampered and not tampered image regions. All the approaches already considered in the alignment tests have been compared through ROC curves (Fig. 8). The area under the curves indicates a biased estimation of the expected tampering detection accuracy of the different methods. These curves have been obtained at varying of the threshold (Th_{un}) used to localize local tampering. It is worth noting that all the image blocks are divided in three groups: tampered blocks, not tampered blocks and border blocks (i.e., blocks that after alignment contain black pixels of the border). In our tests the blocks of

TABLE X
COMPARISON AMONG AFFINE AND SIMILARITY BASED MODEL APPROACHES. SIXTY SIFT HAVE BEEN CONSIDERED IN IMAGE HASH GENERATION PROCESS

	Unmatched	Mean Error α	Mean Error σ	Mean Error T_x	Mean Error T_y
Proposed approach without spatial clustering	0.0675	0.4331	0.0155	1.2072	1.3249
Proposed approach with spatial clustering		0.2287	0.0097	1.0831	1.2095
Proposed approach with spatial clustering and affine estimation		0.2076	0.0088	1.2077	1.3144

TABLE XI
IMAGE TRANSFORMATIONS

Operations	Parameters
Anisotropic Scaling (σ_x or σ_y)	0.7, 0.9, 1.2
Shear (k)	0.05, 0.1, 0.15

TABLE XII
SHEAR ERROR AT VARYING OF k PARAMETER. SIXTY SIFT HAVE BEEN CONSIDERED IN IMAGE HASH GENERATION PROCESS

Shear (k)	0.05	0.1	0.15
Mean Error	0.0092	0.0076	0.0315

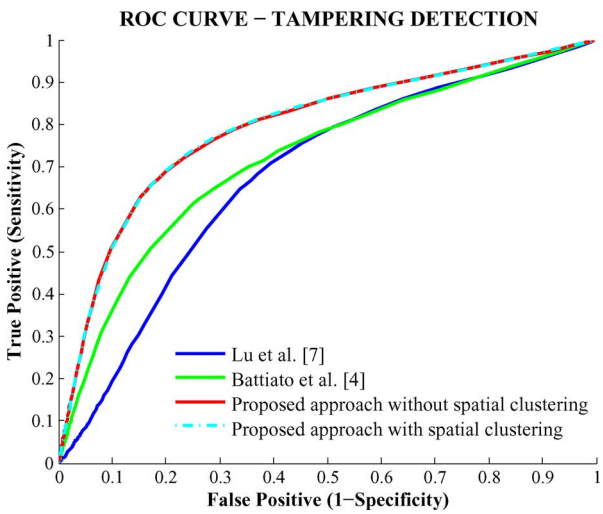


Fig. 8. Tampering detection comparison through ROC curves. Results have been obtained by using uniform quantization of histograms of gradients (12 bits per block).

TABLE XIII
SCALING ERROR AT VARYING OF σ_x AND σ_y PARAMETERS. SIXTY SIFT HAVE BEEN CONSIDERED IN IMAGE HASH GENERATION PROCESS

Anisotropic Scaling	0.7	0.9	1.2
Mean Error σ_x	0.0665	0.0044	0.0096
Mean Error σ_y	0.0717	0.0061	0.0171

the border have been discarded. The overall workflow is shown in Fig. 9. As shown by Fig. 8, our approach (with and without spatial clustering) outperforms all the other techniques. Better alignment corresponds to better localization of manipulations.

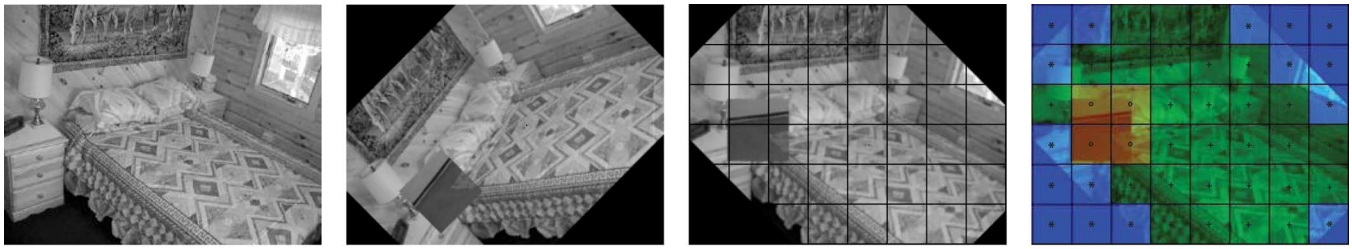
Further experiments have been conducted by considering the proposed non-uniform quantization. Several vocabularies of histograms have been generated through k-means considering k (i.e., the number of prototypes) ranging from 2 to 2^{12}. This clustering has been performed on the histogram of gradients extracted from the whole scene category dataset [22]. Since the performance of the uniform quantization depends on the selected threshold Th_{un}, to properly compare uniform and non-uniform quantization the threshold has been fixed to obtain similar true positive values [Fig. 10(a)]. As reported in Fig. 10(b) the non-uniform approach obtains better results considering a number of bits greater than 4 (i.e., 16 prototype). It is worth noting that the non-uniform quantization describes a single block making use of only $\log_2 k$ bits instead of 12 bits used by uniform quantization as in [8].

VII. ALIGNMENT COMPUTATIONAL COMPLEXITY

The complexity of the proposed voting approach is proportional to the range of the translations (R_T) to be considered during the estimation, to the considered rotations (N_R) and to the number of involved matchings (N_{mv}). To sum up, the complexity is $O(R_T N_R N_{mv})$. In order to have a quantitative measure of the complexity of the different algorithms, we performed a comparative test taking into account their execution time (Table XV). The tests have been performed on a Quad Core i7 with 8 Gb of RAM. All the considered techniques are implemented in Matlab (R2011b). Although the computational complexity of the proposed approach is higher than the other considered techniques, its results are considerably better.

VIII. CONCLUSION AND FUTURE WORKS

The main contribution of this paper is related to the alignment of images in the context of distributed forensic systems. A robust image registration component which exploits an image signature based on the BOF paradigm has been introduced. The proposed hash encodes the spatial distribution of features to better deal with highly texturized and contrasted tampering patches. Moreover, a non-uniform quantization of histograms of oriented gradients is exploited to perform tampering localization. The proposed framework has been experimentally tested on a representative dataset of scenes. Comparative tests show that the proposed approach outperforms recently appeared techniques by obtaining a significant margin in terms of registration accuracy, discriminative performances and tampering detection. Future works should concern a more in-depth analysis to establish the minimal number of SIFT needed to guarantee an accurate estimation of the geometric transformations and a study in terms of bits needed to represent the overall image signature.



(a) Original image. (b) Tampered image. (c) Image registration. (d) Tampering localization

Fig. 9. Example of proposed tampering detection workflow. In (d) orange (∘) indicates recognized tampered blocks, whereas green (+) indicates blocks detected as not tampered. Blue (∗) indicates image blocks falling on border of images after registration. The 32 × 32 grid in (c) and (d) has been overimposed just for visual assessment. This result has been obtained employing alignment with spatial clustering and non-uniform quantization for tampering detection. (a) Original image. (b) Tampered image. (c) Image registration. (d) Tampering localization.

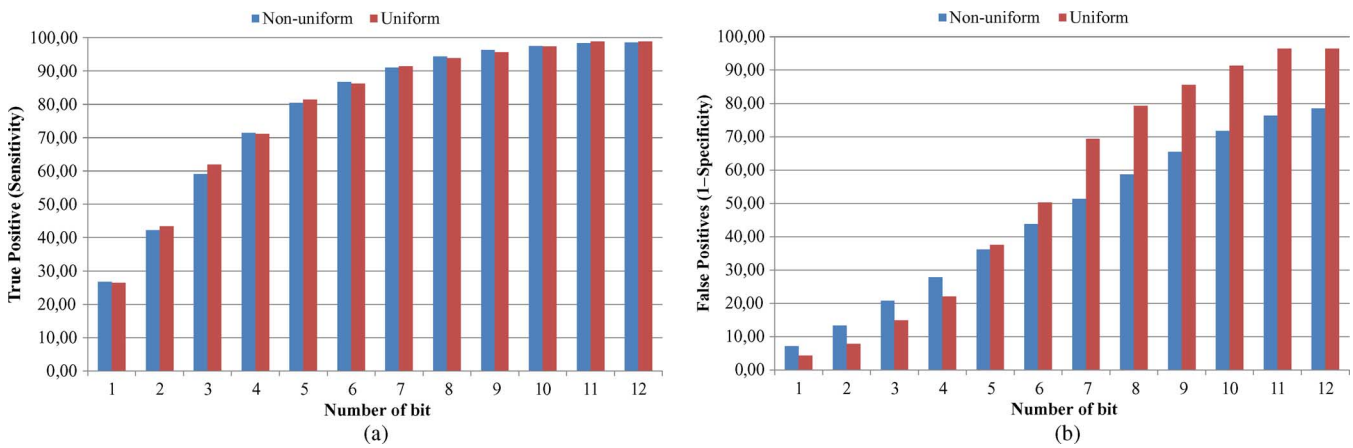


Fig. 10. Comparison of tampering detection results by considering uniform (12 bits per block) and non-uniform (from 1 to 12 bits per block) quantization of histograms of gradient space.

TABLE XIV
AVERAGE ERRORS OBTAINED BY PROPOSED SOLUTION BASED ON AFFINE MODEL. SIXTY SIFT HAVE BEEN CONSIDERED IN IMAGE HASH GENERATION PROCESS

Proposed approach with spatial clustering and affine estimation	
Unmatched	0.0824
Mean Error α	0.2093
Mean Error σ_x	0.0109
Mean Error σ_y	0.0069
Mean Error k	0.0274
Mean Error T_x	1.2210
Mean Error T_y	1.2742

TABLE XV
EXECUTION TIME COMPARISON BETWEEN PROPOSED APPROACH AND OTHER CONSIDERED TECHNIQUES

Execution time (sec) with 60 SIFT	
Lu et al. [7]	0.01
Battiato et al. [4]	0.02
Proposed approach without spatial clustering	1.56
Proposed approach with spatial clustering	2.73

ACKNOWLEDGMENT

The authors would like to thank the authors of [7] for providing their original implementations.

REFERENCES

[1] Photo tampering throughout history [Online]. Available: www.cs.dartmouth.edu/farid/research/digitaltampering/
[2] H. Farid, "Digital doctoring: How to tell the real from the fake," *Significance*, vol. 3, no. 4, pp. 162–166, 2006.
[3] S. Battiato, G. M. Farinella, E. Messina, and G. Puglisi, "Robust image registration and tampering localization exploiting bag of features based forensic signature," in *Proc. ACM Multimedia (MM'11)*, 2011.
[4] S. Battiato, G. M. Farinella, E. Messina, and G. Puglisi, "Understanding geometric manipulations of images through BOVW-based hashing," in *Proc. Int. Workshop Content Protection Forensics (CPAF 2011)*, 2011.
[5] Y.-C. Lin, D. Varodayan, and B. Girod, "Image authentication based on distributed source coding," in *Proc. IEEE Computer Soc. Int. Conf. Image Processing*, 2007, pp. 3–8.
[6] W. Lu, A. L. Varna, and M. Wu, "Forensic hash for multimedia information," in *Proc. SPIE Electronic Imaging Symp.—Media Forensics Security*, 2010.
[7] W. Lu and M. Wu, "Multimedia forensic hash based on visual words," in *Proc. IEEE Computer Soc. Int. Conf. Image Processing*, 2010, pp. 989–992.
[8] S. Roy and Q. Sun, "Robust hash for detecting and localizing image tampering," in *Proc. IEEE Computer Soc. Int. Conf. Image Processing*, 2007, pp. 117–120.
[9] N. Khanna, A. Roca, G. T.-C. Chiu, J. P. Allebach, and E. J. Delp, "Improvements on image authentication and recovery using distributed source coding," in *Proc. SPIE Conf. Media Forensics Security*, 2009, vol. 7254, p. 725415.
[10] Y.-C. Lin, D. P. Varodayan, and B. Girod, "Didstributed source coding authentication of images with affine warping," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP 2009)*, 2009, pp. 1481–1484.
[11] M. Irani and P. Anandan, "About direct methods," in *Proc. Int. Workshop Vision Algorithms, held during ICCV*, Corfu, Greece, 1999, pp. 267–277.

[12] P. H. S. Torr and A. Zisserman, "Feature based methods for structure and motion estimation," in *Proc. Int. Workshop Vision Algorithms, held during ICCV*, Corfu, Greece, 1999, pp. 278–294.
[13] R. Szeliski, "Image alignment and stitching: A tutorial," *Foundations Trends in Computer Graphics Computer Vision*, vol. 2, no. 1, pp. 1–104, 2006.
[14] S. Battiato and G. Messina, "Digital forgery estimation into DCT domain—A critical analysis," in *Proc. ACM Conf. Multimedia 2009, Multimedia in Forensics (MiFor'09)*, 2009.
[15] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Proc. ECCV Int. Workshop Statistical Learning Computer Vision*, 2004.
[16] G. Puglisi and S. Battiato, "A robust image alignment algorithm for video stabilization purposes," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 10, pp. 1390–1400, 2011.
[17] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
[18] L. Shapiro and G. Stockman, *Computer Vision*. Upper Saddle River, NJ: Prentice-Hall, 2001.
[19] M. Brown, R. Szeliski, and S. Winder, "Multi-image matching using multi-scale oriented patches," in *Proc. IEEE Conf. Computer Vision Pattern Recognition*, 2005, vol. 1, pp. 510–517.
[20] L. Gruber, S. Zollmann, D. Wagner, D. Schmalstieg, and T. Hollerer, "Optimization of target objects for natural feature tracking," in *Proc. 20th Int. Conf. Pattern Recognition (ICPR 2010)*, Washington, DC, 2010, pp. 3607–3610.
[21] S. Gauglitz, L. Foschini, M. Turk, and T. Hllerer, "Efficiently selecting spatially distributed keypoints for visual tracking," in *Proc. IEEE Int. Conf. Image Processing (ICIP 2011)*, 2011.
[22] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Computer Soc. Conf. Computer Vision Pattern Recognition*, 2006, pp. 2169–2178.
[23] J. Bi and K. P. Bennett, "Regression error characteristic curves," in *Proc. Int. Conf. Machine Learning*, 2003, pp. 43–50.
[24] S. Maji, A. Berg, and J. Malik, "Classification using intersection kernel support vector machines is efficient," in *Proc. IEEE Int. Conf. Computer Vision Pattern Recognition*, 2008, pp. 1–8.

Sebastiano Battiato (M'04–SM'06) was born in Catania, Italy, in 1972. He received the M.S. degree in computer science (*summa cum laude*), in 1995, and the Ph.D. degree in computer science and applied mathematics in 1999.

From 1999 to 2003, he was the Leader of the "Imaging" team at STMicroelectronics, Catania. He joined the Department of Mathematics and Computer Science, University of Catania, as an Assistant Professor, in 2004, and became an Associate Professor in the same department in 2011. His research interests include image enhancement and processing, image coding, camera imaging technology and multimedia forensics. He has edited four books and co-authored more than 120 papers in international journals, conference proceedings and book chapters. He is a co-inventor of about 15 international patents, reviewer for several international journals, and he has been regularly a member of numerous international conference committees. He is Director (and Co-founder) of the International Computer Vision Summer School (ICVSS), Sicily, Italy.

Prof. Battiato has participated in many international and national research projects. He has been Chair of several international events (ECCV2012, VISAPP 2012, IWCV 2012, CGIV 2012, ICIAP 2011, ACM MiFor 2010–2011, SPIE EI Digital Photography 2011–2012). He is an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEM FOR VIDEO TECHNOLOGY and of the SPIE *Journal of Electronic Imaging*. He has been a Guest Editor of the following special issues: "Emerging Methods for Color Image and Video Quality Enhancement" published in EURASIP *Journal on Image and Video Processing* (2010) and "Multimedia in Forensics, Security and Intelligence" published in IEEE MULTIMEDIA MAGAZINE (2012).

Giovanni Maria Farinella (M'11) received the M.S. degree in computer science (*egregia cum laude*) from the University of Catania, Catania, Italy, in 2004, and the Ph.D. degree in computer science in 2008.

He joined the Image Processing Laboratory (IPLAB) at the Department of Mathematics and Computer Science, University of Catania, in 2008, as a Contract Researcher. He became an Associate Member of the Computer Vision and Robotics Research Group, University of Cambridge, in 2006. He is a Contract Professor of Computer Vision at the Academy of Arts of Catania (since 2004) and Adjunct Professor of Computer Science at the School of Medicine of the University of Catania (since 2011). His research interests lie in the fields of computer vision, pattern recognition and machine learning. He has edited two volumes and coauthored more than 50 papers in international journals, conference proceedings and book chapters. He is a co-inventor of two international patents. He serves as a Reviewer and on the programme committee for major international journals and international conferences. He founded (in 2006) and currently directs the International Computer Vision Summer School.

Enrico Messina was born in San Giovanni La Punta, Italy, in 1982. He received the M.S. degree in computer science (*summa cum laude*), in 2008. He is currently working toward the Ph.D. degree in the Department of Mathematics and Computer Science, University of Catania, Italy.

His interests lie in the fields of computer vision and image analysis.

Giovanni Puglisi (M'11) was born in Acireale, Italy, in 1980. He received the M.S. degree in computer science engineering (*summa cum laude*) from Catania University, Catania, Italy, in 2005, and the Ph.D. degree in computer science in 2009.

He is currently a contract Researcher at the Department of Mathematics and Computer Science, Catania University. His research interests include video stabilization and raster-to-vector conversion techniques. He is the author of several papers on these activities.