This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2656842

Image origin classification based on social network provenance

Roberto Caldelli, *Member, IEEE,* Rudy Becarelli, and Irene Amerini, *Member, IEEE.*

***Abstract*—Recognizing information about the origin of a digital image has been individuated as a crucial task to be tackled by the image forensic scientific community. Understanding something on the previous history of an image could be strategic to address any successive assessment to be made on it: knowing the kind of device used for acquisition or, better, the model of the camera could focus investigations in a specific direction. Sometimes just revealing that a determined post-processing such as an interpolation or a filtering has been performed on an image could be of fundamental importance to go back to its provenance. This paper locates in such a context and proposes an innovative method to inquire if an image derives from a social network and, in particular, try to distinguish from which one has been downloaded. The technique is based on the assumption that each social network applies a peculiar and mostly unknown manipulation that however leaves some distinctive traces on the image; such traces can be extracted to feature every platform. By resorting at trained classifiers, the presented methodology is satisfactorily able to discern different social network origin. Experimental results carried out on diverse image datasets and in various operative conditions witness that such a distinction is possible. In addition, the proposed method is also able to go back to the original JPEG quality factor the image had before being uploaded on a social network.**

***Index Terms*—Image classification, social networks, JPEG, quality factor, provenance identification.**

I. Introduction

NOWADAYS a huge amount of multimedia contents is generated in disparate manners with different devices and then uploaded on the Internet. During upload or once on-line, they are shared with other known users and, ultimately, played or downloaded. These digital assets, accessible on the Internet, mostly flow through social networks (SN) and constitute a real-time source of information. Simply searching throughout the Internet, it clearly transpires from the various, more or less reliable, statistics, that it exists an extraordinary interest on social networks. Just for instance, it is estimated that on average 350 million photos are uploaded daily on *Facebook* and around 60 millions monthly on *Flickr*; furthermore, month after month, the statistics reveal an exponential growth both of the active users and of the number of generated multimedia. The smartphone widespread usage has mainly determined such a phenomenon: a picture can be acquired and uploaded at the same time on one or more social networks. On the other side, illegal activities are proliferating by misusing such digital contents to achieve various, sometimes ignoble, objectives. In this context, both the identification of the origin of a digital content and the reconstruction of its history are crucial issues for disciplines such as multimedia forensics and security. In fact, recovering as much information as possible about the originating device or on the processing that has been applied could be fundamental to comprehend if, for instance, an image is authentic or has been manipulated to change its initial representation and meaning. In particular, it could be of basic importance to succeed in reconstructing the history of a specific digital document that might help in addressing an on-going investigation and/or excluding some suspected subjects. In the case of an image or a video, the aim of retracing its history can be achieved primarily by resorting at the metadata (e.g. EXIF) contained within the file itself but this grants only a limited degree of reliability being them easily modifiable or even erasable. On the contrary, looking for traces and inconsistences left over the image pixels, that can indicate a certain manipulation, provides a higher level of trustworthiness. In the scientific literature many methodologies have been designed to reveal such clues and, consequently, make an assessment on the image/video under analysis. The idea, behind this work, is to research if it is possible to individuate if an image has been downloaded from a specific social network (*provenance*) by analyzing some distinctive signs inevitably released on it by that platform. Doing that is important not only per se but could be propaedeutic to all those techniques dealing with the problem of forgery detection or source identification; knowing that an image has been processed by a certain social network could be useful in image phylogeny reconstruction or in fine-tuning operative and decisional thresholds of some forensic methods. In addition to this, such an instrument could be of support during investigations which, always more, do an extensive use of social networks to reconstruct facts on the basis of the information contained within personal profiles, the different established links and the carried out actions (e.g. comments, posts, etc.), but mainly by analyzing the digital contents associated with a specific account.

The paper is organized as follows: Section II presents some previous works inherent to the problem of recovering information about the origin and the history of a digital image while Section III introduces the proposed methodology; in Section IV some characteristics of the social networks taken into consideration within this work are reviewed and in Section V various experimental results are discussed to evaluate the performances of the presented technique. Section VI draws conclusions and Appendix A provides some implementation details to interact with APIs made available by the different

R. Caldelli is with National Interuniversity Consortium for Telecommunications (CNIT), Parma, Italy and with Media Integration and Communication Center (MICC), University of Florence, Florence, Italy.

R. Becarelli and I. Amerini are with Media Integration and Communication Center (MICC), University of Florence, Florence, Italy.

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2656842

social networks.

II. Related works

Recognizing information about the origin of a digital image has been individuated as a crucial task to be tackled by the image forensic scientific community [1], [2]. Understanding something on the previous history of an image could be strategic to address any successive assessment to be made on it: knowing the kind of device used for acquisition [3], [4], [5], [6], the specific model of the device [7], [8], [9], [10], [11] or the further processing applied to an image [12], [13], [14], [15], [16] could focus investigations in a specific direction. The main idea behind this kind of approaches is that each phase of the image acquisition process or each post-processing applied to an image leaves a sort of unique fingerprint on the digital content itself due to some intrinsic imperfections of the acquisition process or to some characterization of the applied operations. For example, the PRNU (Photo Response Non-Uniformity) noise [11] is used as fingerprint to identify a specific digital camera among a dataset of cameras. Furthermore, when the number of cameras and images scales up, methods which resort at the adoption of digest-based descriptors are taken into account [17], [18] to reduce computational burden but maintaining performances in terms of classification accuracy. Others methods consider the source camera attribution problem in a open set scenario i.e. images could have been generated by an unknown device not available in the set of cameras under investigation. The authors in [19], [20] proposed a classification system to distinguish among images taken by unknown digital cameras by resorting to the use of enhanced version of PRNU. Another interesting topic in the source identification task is about to distinguish among various classes of devices (e.g scanned images, photos, computer generated) extracting some robust and characterizing features. Such features are distinctive because they exploit some characteristic traces left over the digital content during the operation of image creation. Usually such features are extracted from a training set of images whose provenance is known and used to train a classifier (e.g. SVM); then the trained classifier is able to evaluate a digital asset and to establish which category it belongs to among scanned images, photos or computer generated. In [21] a method to identify photos created by different sources without any type of previous knowledge is proposed suggesting a blind clustering of the different source devices. Since one of the most common problems in the image forensics field is the reconstruction of the history of an image or a video, sometimes just revealing that a determined post-processing such as an interpolation, resampling, double JPEG compression or filtering operation has been performed on an image could be of fundamental importance to go back to its provenance [22], [23], [24], [25]. In particular, some approaches [26], [27], [28], [29], [30] take care of analyzing the statistical distribution of the values assumed by the DCT coefficients. In [31], [32], [33] methods for the detection of double JPEG compression using classifiers with feature vectors derived from histogram of DCT coefficients are proposed; in particular for applications in steganography and image forgery detection. Furthermore, the authors in [34], [35], [36] has proposed a set of possible solutions to perform phylogenetic analysis (reconstructing image history) based on image dissimilarity computation on near-duplicate images for image phylogeny tree reconstruction. The explosion in the usage of social network services enlarges the variability of image data and presents new scenarios and challenges in the source identification and classification task. In [37] the authors aims to exploit the algorithms and techniques behind the uploading process of a picture on *Facebook* in order to find out if any of the involved steps (resizing, compression, renaming, etc.) leaves a trace on the picture itself, so to infer the image authenticity. A study on social network services is done in [38], [39] where it tries to detect JPEG images on *Facebook*. In particular, in [38] the authors define a metric to measure the distance between two JPEG images in which one image is obtained by compressing the other and in [39] a technique to reveal tampering created using *Facebook* images is proposed. In [40] an analysis on how the social networks like *Facebook*, *Badoo* and *Google+* process the uploaded images and what changes are made to some of the characteristics, such as JPEG quantization table, pixel resolution and related metadata is performed.

III. Proposed method

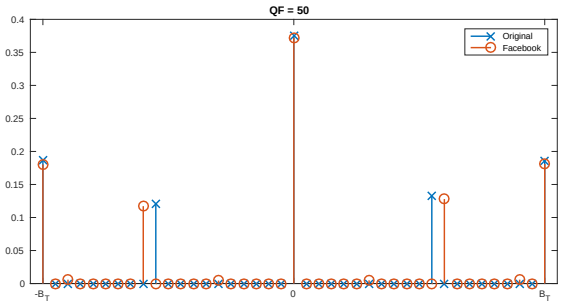
The proposed method is structured on two main phases: the extraction of the distinctive features and, consequently, the training of an ad-hoc classifier to be used during the testing step. The following two subsections address these phases respectively.

A. Features extraction

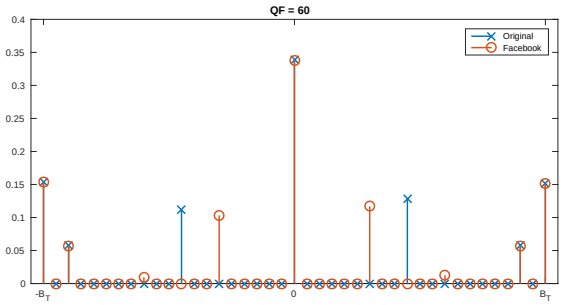
Before being uploaded on social networks digital images presumably are in JPEG format being usually created with a photo-camera or a smartphone and then they undergo a specific processing which is typical of each social network; though it is not known what actually happens it is expected that a JPEG compression is applied to reduce the size of the image and/or to adapt it to the needs of the social platform, for example in terms of visualization, sharing and small footprint. On this assumption, it has been decided to take into consideration the DCT (Discrete Cosine Transform) domain to look for distinctive traces of such a processing. In fact, it is well-known in forensic scientific literature that DCT coefficients are useful to track distortions introduced by JPEG compressions [31]. To do that, a certain number of DCT (8×8 block) dequantized coefficients $c_k(i,j)$ ($k = 1, 2, ..., N_c$) are taken for every 8×8 block and organized in a histogram separately for each k. The index k is associated to a specific spatial frequency (i,j) following a zig-zag scanning and N_c indicates the number of analyzed DCT coefficients. The DC coefficient ($k = 0$) is skipped. Following the objective to keep track of any possible distinctiveness, each histogram has a bin-step size of 1 and represents positive and negative values; anyway to avoid having a huge amount of accumulation classes, bin values have been limited between $\pm B_T$, so, for example, the B_T bin contains the occurrences of all the values $c_k \geq B_T$. In Figure 1, the sample histograms of the DCT-coefficients

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2656842

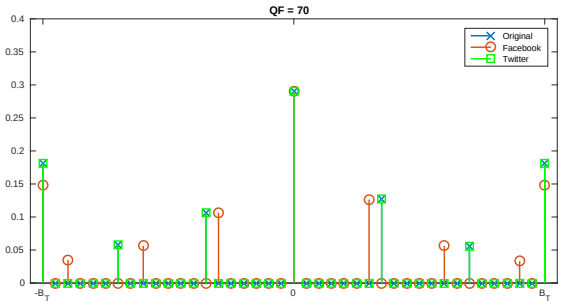


(a) Histograms before/after upload on *Facebook* ($QF = 50$)

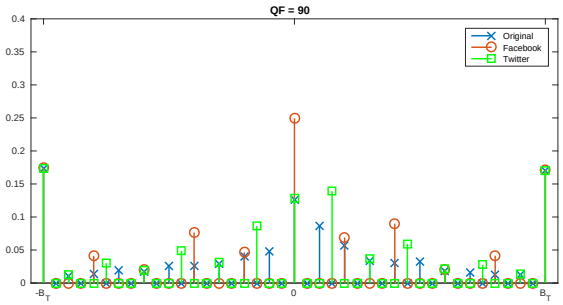


(b) Histograms before/after upload on *Facebook* ($QF = 60$)

Fig. 1: Histograms of DCT-coefficients for the mode $c_{k=1}$ before (blue) and after (red) uploading a sample image on *Facebook*: $QF = 50$ (a) and $QF = 60$ (b) are the quality factors of the images before uploading. Histograms are normalized with respect to the number of 8×8 blocks.



(a) Histograms before/after upload on *Facebook* or *Twitter* ($QF = 70$)



(b) Histograms before/after upload on *Facebook* or *Twitter* ($QF = 90$)

Fig. 2: Histograms of DCT-coefficients for the mode $c_{k=1}$ before (blue) and after uploading a sample image on *Facebook* (red) and on *Twitter* (green): $QF = 70$ (a) and $QF = 90$ (b) are the quality factors of the images before uploading. Histograms are normalized with respect to the number of 8×8 blocks.

belonging to the $c_{k=1}$ mode ($i = 0$, $j = 1$) of a JPEG image before uploading it on *Facebook* and then successively downloaded are pictured. In Figure 1a and Figure 1b, two different initial quality factors $QF = 50$ and $QF = 60$ have been represented for comparison respectively. Such histograms are normalized with respect to the number of their occurrences that is the number of 8×8 block DCT of the image in order to grant independence by the image size. It can be pointed out that the histograms show, in both cases of different initial quality factors, diverse values between the original (before upload) image (blue) and that one downloaded from the social network (red). Such different values are distinguishable both in magnitude and in position. In Figure 2 two other quality factors have been considered: $QF = 70$ and $QF = 90$. In this circumstance, the behavior of *Twitter* (green) is also presented in order to highlight the differences not only with the original image but between the social platforms. It is interesting to observe in Figure 2a that when the $QF = 70$ the features can not distinguish between images not uploaded on a social network and those coming from *Twitter*; on the contrary, this does not happen for $QF = 90$ (see Figure 2b) where the three categories appear well separated: this issue will be investigated in depth in Section V. Finally, in Figure 3, a comparison among the features behavior for three different social networks (*Facebook* (red), *Twitter* (green) and *Flickr* (black)) is pictured ($QF = 80$ and $QF = 85$ have been presented as sample quality factors and the category of original image has been omitted in this case): a sufficient and hopeful distinction is highlighted in such a circumstance too. The values of each histogram $\mathbf{H_k}$ (see Equation (1)) are taken sequentially as distinctive features for each of the $k = 1, 2, ..., N_c$ modes.

$$\mathbf{H_k} = [h_k(-B_T), ..., h_k(0), ...h_k(B_T)] \tag{1}$$

This determines that every image is represented by means of a features vector $\mathbf{V_{imm}}$ as reported in Equation (2):

$$\mathbf{V_{imm}} = [\mathbf{H_{k=1}}, \mathbf{H_{k=2}}, ..., \mathbf{H_{k=N_c}}] \tag{2}$$

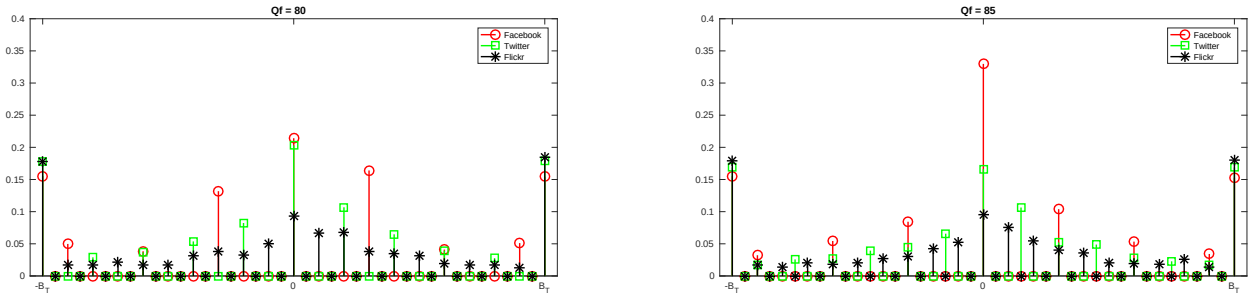
The features vector $\mathbf{V_{imm}}$ will be finally constituted by $N_v = [(2 \times B_T + 1) \times N_c]$ elements.

B. Training and classification

The process of image classification, according to the social network of provenance, has been performed by resorting at a trained classifier. The classifier is trained by using for each image, belonging to a specific class, the corresponding features

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2656842



(a) Histograms after upload on *Facebook*, *Twitter* and *Flickr* ($QF = 80$)

(b) Histograms after upload on *Facebook*, *Twitter* and *Flickr* ($QF = 85$)

Fig. 3: Histograms of DCT-coefficients for the mode $c_{k=1}$ after uploading a sample image on *Facebook* (red), *Twitter* (green) and *Flickr* (black): $QF = 80$ (a) and $QF = 85$ (b) are the quality factors of the images before uploading. Histograms are normalized with respect to the number of 8×8 blocks.

vector $\mathbf{v_{imm}}$ composed by $N_v = [(2 \times B_T + 1) \times N_c]$ elements computed as described in subsection III-A. The training phase is classically carried out by providing examples representing each category to be learnt. During experimental tests (see Section V), two basic situations have been taken into account: the first one envisaged to have a number of classes which depended on the JPEG quality factors considered for each social network and the second one which was determined only by the number of social networks (no categorization for QF was required). Of course, the trained classifier is then asked to decide on the features vector $\mathbf{v_{imm}}$ extracted from the test image and to associate it to one of the possible known classes. Some diverse classification procedures have been analyzed but performances were quite similar and, not being one of the main goals of the work, a possible evaluation of different classification approaches has been left to a future work. Within the Section V only experimental tests obtained with one kind of classifier though with various training manners are discussed. The adopted classifier is a *Bagged Tree Random Forest* which is based on a general technique of random decision forests [41] that are an ensemble learning method for classification and other tasks. It works by constructing a multitude of decision trees N_{trees} during the training phase and outputting the class that is the mode among all the classes.

IV. Social networks: upload/download phase

The basic idea behind this work was to understand if different social networks left some distinctive features on the images during the process of upload/download without analyzing which kind of specific and hidden transformation they applied. To do that three of the most common ones have been selected: *Facebook* and *Twitter* which are very well-known and *Flickr* which is much more image-oriented though it contains social features for comments and sharing. This choice has also been led by the basic need to deal with SNs which provided public APIs (Application Programming Interface) to automatically manage the operation of image uploading and downloading by resorting at a dedicated web service; in fact, it would not be feasible using the classical web interface and/or each proprietary application (see Appendix A for implementation details) having to deal with many images for experimental tests. For example, *Instagram*, though exposing a set of APIs basically to check the context of the social relationships, does not provide an official API for photo uploading but it can be done only through its application; similarly *Whatsapp* does not expose public APIs for developers (such a policy has been defined to avoid users to be inundated with unwanted messages). This has constituted the main requirement having to envisage that thousands of images were to be automatically posted on each platform and then to be downloaded to viably perform experimental tests for SN provenance identification. For what concerns download, both *Facebook* and *Twitter* present an API to be queried which permits to choose among various picture resolutions, during our experiments images have been always downloaded with the same resolution they had in upload. Similarly it happens for *Flickr* which has a wide set of parameterized methods that consent to download different kinds of images in terms of resolution. Such kinds start from the format "Original" going down, in terms of resolution, to the one named "Thumbnail"; in this case, the format "Original" means that the image is an exact copy of that one uploaded so being undistinguishable with respect to that (i.e. Peak Signal-to-Noise Ratio, $PSNR = \infty$). This situation did not represent a case of interest for our study about identification and, however, can be traced back to the case of discerning *only-Compressed* images (see Section V-B) with respect to those compressed but coming from a social network. According to that, during our experiments, images with a resolution format immediately under the "Original", for instance for an UCID image is "small-320" (aspect ratio is however preserved), have been required through the ad-hoc method and considered as downloaded from *Flickr*. Lower levels of resolution have not been taken into account in order to maintain as higher as possible the quality of the downloaded picture in comparison with the original one.

V. Experimental results

In this section some of the different experimental tests that have been carried out are presented. First of all, the whole experimental set-up is introduced (subsection V-A), while the

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2656842

case in which the problem to discern the social network provenance is analyzed in relation with the upload quality factor (QF) is debated within subsection V-B; then, the specific information on QF is not considered anymore and only the membership to a social network is investigated, results are discussed in subsection V-C and for multiple upload-download in V-D. Finally, in subsection V-E, the system has been tested in an open scenario and the achieved results are debated.

A. Set-up description

In this subsection the experiment set-up is described. First of all, the two parameters B_T and N_c involved within the proposed methodology (see subsection III-A) have been fixed at $B_T = 20$ and $N_c = 9$ respectively; the first choice has been determined on the basis of an empirical analysis trying to achieve a trade-off between distinctiveness and length of the features vector, while the second one was indicated by a previous work [29] in which such a value has been adopted resulting in a effective representation of the most significant DCT coefficients. According to this setting, the features vector $\mathbf{V_{imm}}$ characterizing each image has a dimension of 369 elements ($N_v = [(2 \times (B_T = 20) + 1) \times (N_c = 9)]$). The digital images used for the experiments are taken from UCID (Uncompressed Colour Image Database) database [42] which is composed by 1338 images (512×384 pixels) in TIFF format. These have been used as a basis to generate JPEG compressed images at different wanted quality factors. JPEG compression has been performed by resorting at MATLAB R2015b (*jpegtooolbox 1.4* library). JPEG images, created according to this process, are then uploaded via each corresponding API onto a test profile for each selected social network (e.g. *Flickr*, *Facebook* and *Twitter*). Again via the API, images are downloaded so having undergone the specific transformation every platform applies (dataset downloadable here[1]). The quality factors have been considered within the range $[QF = 50 \div 95]$ with a step of 5, so this leads to 10 different values. For each of these quality factors 1000 images of the UCID database have been uploaded/downloaded on each of the three social networks, so this means $10 \times 1000 \times 3 = 30000$ pictures in total. In the following subsections with the terms N_{TR} and N_{TS} the number of images used for training and testing phases will be indicated respectively. The adopted classifier is a *Bagged Tree Random Forest* with a number of trees $N_{trees} = 10$.

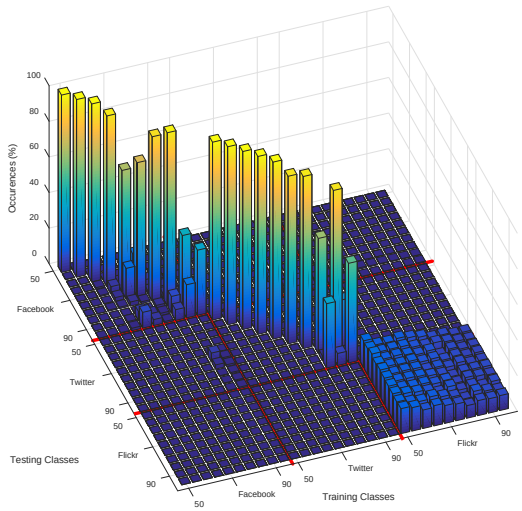
B. Classification tests: social network provenance with QF detection

In this subsection experimental results obtained for each of the three social networks in relation with the QF used during uploading are presented. Training has been done in this manner: $N_{TR} = 500$ images, that is 50 images for each quality factor, have been selected and used to train the classifier on each of the ten classes of each social network (i.e. 1500 images globally for training). On the contrary, 100 images for each quality factor ($N_{TS} = 1000$) have been used for testing for every social network (i.e. 3000 images globally for testing). To improve image independency the whole set-up has been repeated and tested according to a cross-validation approach based on a circular shift s of 5 images each time over the dataset of 150 images (50 training and 100 testing) so determining 30 different training/testing conditions that means that evaluation results, presented in Figure 4a, have been finally achieved averaging on 90000 (3000×30) test images. Figure 4a shows the confusion matrix obtained for the classification test for each of the 30 classes: 10 for each social network *Facebook*, *Twitter* and *Flickr* respectively. On the left axis of the confusion matrix the test classes are located while on the right one there are the classes that compose the trained classifier; therefore the height of each column represents the number (in percentage) of images which are assigned to a specific training class actually belonging to a certain testing class (ground-truth). It is worthy highlighting that most of the images are correctly classified for *Facebook* and *Twitter* as evidence by the columns on the diagonal of the confusion matrix. It is interesting to underline that without any information about the QF of the image before uploading, the system is almost always able to re-identify it. The same thing is not true for *Flickr* where images are globally spread over the diverse quality factors; however it is important to point out that, though it is not possible to recognize the upload quality factor, the method does not misclassify images coming from *Flickr* with those of the other social networks. The same experiment has been repeated but, in this circumstance, 10 more classes have been added to the classifier (40 classes in total). Such classes represent the *only-Compressed* (but not uploaded on any social network) images with quality factor again within the range of $[QF = 50 \div QF = 95]$ with a step of 5: achieved results are presented in Figure 4b (tested images are now $4000 \times 30 = 120000$). It is interesting to notice that performances for *Facebook* and *Flickr* are globally maintained but there is a strong symmetric misclassification among *only-Compressed* images and *Twitter* ones. This is particularly true for $QF \leq 85$ while the error is drastically reduced for higher quality factors. This phenomenon is basically due to the fact that images with an original $QF \leq 85$ when uploaded on *Twitter* are usually not processed so, when downloaded, they are equal to their original version and are averagely confused with the *only-Compressed* classes. Such an issue has already been evidenced in subsection III-A and, particularly, in Figure 2a and 2b which present an indistinguishability between *only-Compressed* and *Twitter* images for $QF = 70$ but that is not maintained for $QF = 90$ anymore. Another interesting aspect concerns *Flickr*; in this case, for all the quality factors, it happens that around 20% of the images are wrongly classified out of *Flickr* and as belonging to the class *only-Compressed-90*. This is not unexpected because generally pictures downloaded from *Flickr* have a quality factor of 90; this issue is evidenced also by the dual situation when *only-Compressed* images with $QF = 90$ are labeled as coming from *Flickr* with different quality factors (see Figure 4b in the top corner). Another similar experiment (named second experiment) has been carried out in order to double-check the behavior of the method when training and testing sets are constituted by a higher number of images and, above all, the

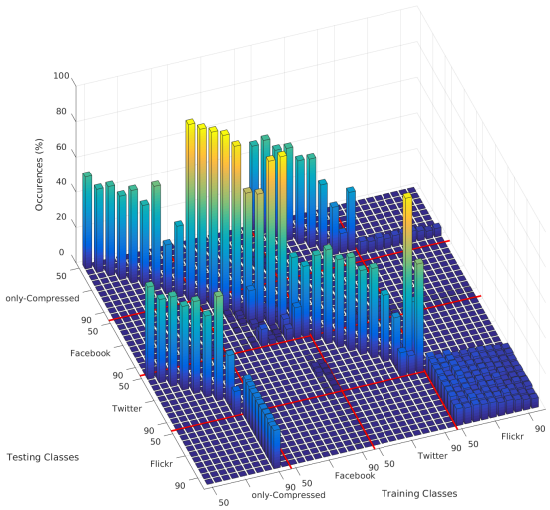
[1]http://lci.micc.unifi.it/labd/2015/01/trustworthiness-and-social-forensic/

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2656842

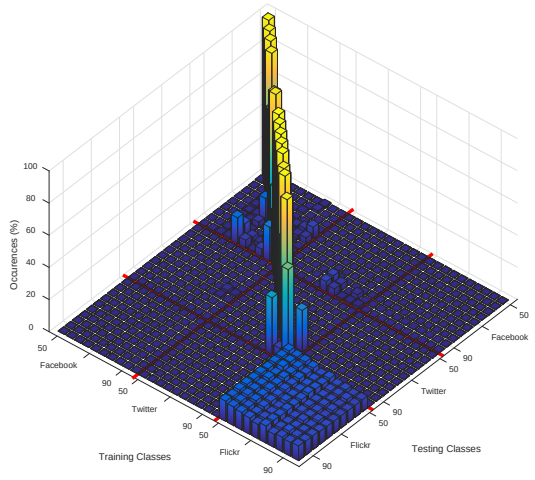


(a) Classification among *Facebook*, *Twitter* and *Flickr*: 30 classes, 10 for each category (social network).

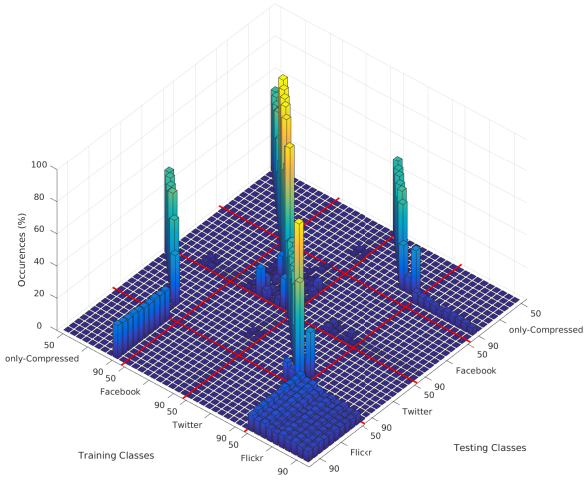


(b) Classification among *only-Compressed*, *Facebook*, *Twitter* and *Flickr*: 40 classes, 10 for each category.

Fig. 4: Confusion matrix (first experiment): social network provenance classification according to JPEG quality factors (QF).



(a) Classification among *Facebook*, *Twitter* and *Flickr*: 30 classes, 10 for each category (social network).



(b) Classification among *only-Compressed*, *Facebook*, *Twitter* and *Flickr*: 40 classes, 10 for each category.

Fig. 5: Confusion matrix (second experiment): social network provenance classification according to JPEG quality factors (QF).

procedure of cross-validation is implemented in a different manner (see Figure 5 in which confusion matrix are also presented from a diverse point of view). In this case, 1000 images have been used for each QF: 100 for training and 900 for testing. Then a circular shift s of 100 is applied (training sets are not overlapping this time) thus determining that 10 different tests are made. Globally, $900 \times 10 \times 10 \times 3 = 270000$ images have been classified in this experiment again (360000 for results in Figure 5b). It can be pointed out that the overall behavior is maintained with respect to the previous experiment, though a reduced misclassification is even obtained.

In Table I and II a quantitative evaluation of the previous two tests depicted in Figure 5 is presented respectively (results for Figure 4 are similar so they have been omitted); in particular, for each social network the values for correct classification (values on the diagonal of the confusion matrix) according

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2656842

TABLE I: Classification among *Facebook*, *Twitter* and *Flickr*

Classification (%) vs QF	50	55	60	65	70	75	80	85	90	95
Facebook Diagonal	98.03	97.24	97.04	95.72	66.93	71.58	89.08	94.33	42.28	33.52
Facebook Other	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.41	16.08	16.32
Twitter Diagonal	99.89	99.80	99.77	99.81	99.78	96.67	98.04	66.24	95.61	58.20
Twitter Other	0.10	0.20	0.20	0.18	0.20	3.32	1.92	0.06	0.09	0.09
Flickr Diagonal	13.63	12.73	11.66	10.52	9.63	9.92	8.43	8.44	8.68	9.09
Flickr Other	0.08	0.07	0.07	0.09	0.08	0.09	0.08	0.08	0.08	0.06

TABLE II: Classification among *only-Compressed*, *Facebook*, *Twitter* and *Flickr*

Classification (%) vs QF	50	55	60	65	70	75	80	85	90	95
only-Compressed Diagonal	51.12	52.24	51.50	52.13	56.62	43.86	49.03	31.61	43.40	99.99
only-Compressed Other	48.88	47.76	48.50	47.86	43.38	56.14	50.97	68.39	56.60	0.00
Facebook Diagonal	98.26	97.31	97.07	93.73	65.21	73.41	89.26	94.39	39.37	33.60
Facebook Other	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.40	15.37	15.58
Twitter Diagonal	48.78	47.61	48.26	47.50	43.26	53.01	49.36	28.36	97.24	65.29
Twitter Other	51.21	52.39	51.73	52.50	56.72	46.99	50.61	31.83	0.61	14.03
Flickr Diagonal	11.87	9.59	9.79	8.32	7.07	7.89	7.63	7.50	7.26	6.40
Flickr Other	19.14	19.00	19.60	18.93	19.10	19.48	20.32	19.70	20.01	20.76

to each quality factor are presented (indicated with the term *Diagonal*). Furthermore, the results, indicated with the term *Other*, show how many images are averagely catalogued out of the correct social network of provenance (i.e. out of the confusion matrix main diagonal indicated by the red squares). It can be seen that in Table I images coming from *Facebook* are rightly classified till $QF \leq 85$ and if there are some lower percentages for $QF = 70$ and $QF = 75$, they do not determine a migration towards other social networks (see second row of Table I). On the contrary, for $QF = 90$ and $QF = 95$ image classification is not satisfactory and even social network mis-assignment is quite relevant (16.08% and 16.32%). A similar situation is noticed for the *Twitter* case, though the wrong classification out of the social network is more limited (see forth row of Table I). Finally, it can be appreciated that the method is not able to well distinguish the original quality factor of the images downloaded from *Flickr* while there is a misclassification out of the social network approximately null (see sixth row of Table I). Looking at Table II, it can be observed that the method behaves as before for images coming from *Facebook* but there is a misclassification mainly between the categories *only-Compressed* and *Twitter*, only for high quality factors identification seems to be satisfactory for both the classes (*only-Compressed* with $QF = 95$ gets 99.99% and *Twitter* with $QF = 90$ and $QF = 95$ gets 97.24% and 65.29% respectively). *Flickr* shows results similar to the previous case of 30 classes with a further error (around 19%) induced by the presence of the class *only-Compressed-90* as already discussed.

C. Classification tests: provenance detection from Facebook, Twitter and Flickr

In this section experiments dedicated to investigate only social network provenance without the requirement to detect the original quality factor are presented. In this case the classifier is trained to recognize only three classes: *Facebook*, *Twitter* and *Flickr*. As before, 1000 images from UCID dataset have been used for each QF: 100 for training and 900 for testing with a circular shift *s* of 100 thus determining that 10 different tests are made. Globally, $900 \times 10 \times 10 \times 3 = 270000$ images have been classified in this experiment again but with respect to a three-classes classifier (different QF pictures are considered only to generate an uniform dataset). In Table III the confusion matrix is presented; it is evident that the system provides a satisfactory performance.

TABLE III: Classification among *Facebook*, *Twitter* and *Flickr* (UCID dataset): training set 100 images and testing set 900 images (*s*=100).

Classification (%) vs SNs	Facebook	Twitter	Flickr
Facebook	96.85	2.77	0.37
Twitter	0.35	99.65	0.00
Flickr	0.38	0.00	99.72

TABLE IV: Classification among *Facebook*, *Twitter* and *Flickr* (UCID dataset): training set 500 images and testing set 100 images (*s*=50).

Classification (%) vs SNs	Facebook	Twitter	Flickr
Facebook	97.42	2.58	0.00
Twitter	0.33	99.67	0.00
Flickr	0.00	0.00	100.00

Hereafter, in Table IV, results obtained in another experiment are reported. In this case, 500 images are used as training set while the test set is composed by 100 pictures so globally 600 pictures for each QF; a cross-validation procedure with a shift *s* equal to 50 is implemented, so totally $100 \times 10 \times 12 \times 3 = 36000$ images have been evaluated. This demonstrates that performances still hold though training and testing set sizes have been changed.

D. Classification tests: provenance detection from Facebook, Twitter and Flickr after cross upload-download

In this section, results obtained in experimental tests when a cross upload-download has been performed are presented. With this term is to be intended JPEG images that firstly have been uploaded-downloaded on a social network (e.g.

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2656842

TABLE V: Cross upload-download experiment: classification among *Facebook*, *Twitter* and *Flickr* (UCID dataset). Training set 500 images and testing set 100 images (s=50).

	FB2FL	TW2FL	TW2FB	FL2FB	FB2TW	FL2TW
FB	0.02	0.01	99.50	88.39	96.90	0.17
TW	0.00	0.00	0.50	7.22	3.10	99.83
FL	99.98	99.99	0.00	4.39	0.00	0.00

Facebook), then uploaded-downloaded on another one (e.g. *Flickr*) and finally evaluated with a three-classes classifier used in section V-C. In this case, it has not been considered the option to train a multiclass classifier that was able to discern among different cases of cross upload-download: this has been left to future investigations. The same classification tests as in Tables III and IV have been carried out (only the second one is reported in Table V for sake of conciseness because results were very similar in both cases). All the possible combinations among the three considered social networks have been analyzed. It comes out that the method is still able to reliably identify the last social network of the chain in the circumstances when *Facebook* and *Flickr* are the final step: correct classification is around 99%, except for the case *Flickr2Facebook* (FL2FB) where performances are instead around 88%. This can be explained because *Flickr* introduces important distortions that remain partially detectable by the classifier notwithstanding the successive transition on *Facebook*. Diverse is what happens when *Twitter* is the last step. In the case *Facebook2Twitter* (FB2TW), the system does not basically recovers *Twitter* but it detects the social network of the previous step, that is *Facebook*. This is coherent with the behavior discussed in section V-B concerning the fact that *Twitter* does not seem to process images with $QF \leq 85$; in fact, in this situation, though images originally contained all the QFs (within the range of $[QF = 50 \div QF = 95]$) when they are downloaded from *Facebook* their QFs are all limited under $QF = 85$. On the contrary, in the case *Flickr2Twitter* (FL2TW), the system almost perfectly identifies *Twitter* as the final step and this is again in line with the previous explanation having all the images, downloaded from *Flickr*, a $QF = 90$ and, being higher than 85, are therefore successively processed by *Twitter* whose traces are revealed by the trained classifier.

E. Classification tests: provenance detection from Facebook, Twitter and Flickr in an open scenario

In order to further verify the proposed method in an open social network scenario, a new uncontrolled set of images has been downloaded by the three platforms without previously uploading them (*PUBLIC dataset*). For *Facebook* the plug-in $DownloadFB - AlbumMod$ has been used which permits to download images present in a friend's photo album; for *Twitter* the public API has been utilized by asking for some generic tags such as *whales*, *dinosaurs*, *steve jobs*, *star wars*, *trump*, *samsung* and so on. For *Flickr*, the public API has been used to request images asking for some keywords such as *renzi*, *obama*, *merkel*, *tsipras*, *putin* within the last (in a certain period of time) uploaded images. Doing so 1000 uncontrolled images (different sizes, JPEG quality factors, contents and so on) have been gathered for each social network with no information about their previous history. Similarly to what has been done in the last experiment, a subgroup of 600 has been selected (randomly chosen) for each social network: 500 have been used for training while the remaining 100 have been left for testing. This has been repeated with a shift s equal to 50 so generating 12 evaluation tests which means that $100 \times 12 \times 3 = 3600$ images have been classified in total. Results presented in Table VI witness that the classification capacity of the method is still good with respect to the controlled scenario: correct classification is around 88.75%. Results obtained by substituting to the proposed method the features described in [31] are also shown: it is clear that, though performances are a bit lower in this case, a certain distinctiveness is still granted.

TABLE VI: Classification among *Facebook*, *Twitter* and *Flickr* (PUBLIC dataset): training set 500 images and testing set 100 images (s=50).

Classific. (%) vs SNs	**Facebook**		**Twitter**		**Flickr**	
Features	Ours	[31]	Ours	[31]	Ours	[31]
Facebook	86.34	80.67	10.58	14.58	3.08	4.75
Twitter	10.00	18.92	89.33	74.92	0.67	6.17
Flickr	5.58	11.17	3.83	9.75	90.59	79.08

TABLE VII: Classification among *Facebook*, *Twitter* and *Flickr* (PUBLIC dataset): training set 900 images and testing set 100 images (s=100).

Classific. (%) vs SNs	**Facebook**		**Twitter**		**Flickr**	
Features	Ours	[31]	Ours	[31]	Ours	[31]
Facebook	77.40	74.20	19.20	18.10	2.90	7.70
Twitter	14.30	24.10	84.70	70.10	1.00	5.80
Flickr	5.50	13.70	4.80	9.80	89.70	76.50

In Table VII, the results achieved in another training/testing configuration are presented. In this circumstance, 900 images have been used for training and 100 for testing for each social network with a repetition shift s equal to 100 over a group of 1000 images; so yielding to $100 \times 10 \times 3 = 3000$ images classified in total. Correct classification percentages are averagely around 83.93% while using the features in [31] they are reduced at about 73.60%.

VI. Conclusion

In this paper, we have proposed a novel methodology to distinguish images coming from different social networks. The main contributions of the actual work are the following:

- the introduction of the usage of feature-based descriptors able to allow a distinction among the processing suffered by the images when uploaded on a specific social network.
- the definition of a technique based on such features which by resorting at trained classifiers is able to identify the social platform of provenance and also to detect the quality factor before uploading.
- the achievement of satisfactory performances in terms of SN source identification.

Future works will be dedicated to investigate the adoption of a diverse set of features (e.g. changing the parameter B_T)

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2656842

and to test different kinds of classifiers. Furthermore, other social networks, such as *Google+, Pinterest, Tumblr*, will be taken into account in the future to increase the number of the considered SNs. Another interesting issue will be to perform a comprehensive study regarding the behavior of the proposed method in the case of multiple upload/download already outlined in the experimental section.

APPENDIX A
API DESCRIPTION

The experimental results presented above have been generated uploading and subsequently downloading from social networks a certain number of standard images. Both the number of images to be dealt with and the complex user interactions necessary to download an image from a SN motivated the implementation of a multiclient able to automatically interact with each SN's public API. This API requires an OAuth (Open Authentication) access in order to ensure a "secure and delegate authentication" without exposing the user password during the interaction. In all these cases, in fact, the user cannot directly access to the social network API, he is instead requested to create an "Application", that is a controlled context which is authorized to access to some extent to the user profile. Once the Application has been created, the user can typically refer to the Application itself accessing to a public endpoint, being it a web page or a web service, and providing an "Application Unique ID" and an "Application Secret". The user is then asked to authorize the Application to act on behalf of him by providing his own credentials. The Application then returns an "Access Token" that can be later used by the multiclient to access, via the Media API Interface and on behalf of the user, to the user profile. The APIs, along with some libraries and programs, chose to ease the integration with the Java-based multiclient are listed hereafter:

- *Facebook* exposes all its functionalities, including the visual media management, by means of the Graph API[2]. The Graph API reflects the various and composite Facebook architecture where the user can interact with other users and SN entities in disparate ways. The integration with the multiclient application has been then facilitated by the adoption of the *RestFB* library[3] that is able to clearly separate the context of Media Management from the more generic Status Management.
- *Twitter* instead exposes a Developer API[4] that ensures a limited but effective interaction with the user profile. The developer has to just deal with five entities that constitute the basis of the Twitter platform. Because of its simplicity there was no reason to adopt a dedicated library to operate upload/download, but, in order to facilitate the integration with the multiclient application, the *Twitter4J* library[5] has been used.
- *Flickr* has a more straightforward and context specialized API[6] that makes the developer free to interact with the basic functions of the SN platform. Also in this case, even if it was not necessary, the *Flickr4Java* library[7] has been adopted.

For the public images downloading instead (section V-E), the public SN's APIs have been used where possible. Only for Facebook, a dedicated Chrome plug-in (FB Album Mod[8]) has been used in order to simplify the authorization process since the *Chrome mod* enables the user to download massively the pictures contained in the album of one of his own friends. The API usage, especially regarding the media uploading, is subjected to some limitations, mostly related to the maximal file size and the number of files uploadable during a certain extent of time. Consequently, the Java client has been developed with the possibility to catch the API exceptions, switch to an idle state for a defined time and then resume the upload/downloading. The aforementioned limitations mostly affect the capability of posting a large number of images on the SN's platform, since the size of a single image is widely compliant with any file size limitation. Facebook has a "Rate Limit" of 200 calls/hour; Twitter limits the "App Auth Calls" to 180 requests/15 minutes; Flickr allows a maximal limit of 3600 queries/hour.

[2]Facebook Graph API, https://developers.facebook.com/docs/graph-api/
[3]RestFB Library, http://restfb.com/
[4]Twitter Developer API, https://dev.twitter.com/
[5]Twitter4J Library, http://twitter4j.org/en/index.html
[6]Flickr Developer API, https://www.flickr.com/services/api/

REFERENCES

[1] M. Stamm, M. W., and K. Liu, "Information forensics: An overview of the first decade," *Access, IEEE*, vol. 1, pp. 167–200, 2013.
[2] M. Barni, "A game theoretic approach to source identification with known statistics," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2012, pp. 1745–1748.
[3] 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.
[4] C. McKay, A. Swaminathan, G. Hongmei, and M. Wu, "Image acquisition forensics: Forensic analysis to identify imaging source," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, 2008, pp. 1657–1660.
[5] S. Lyu and H. Farid, "How realistic is photorealistic?" *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 845–850, 2005.
[6] 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.
[7] S. Bayram, H. Sencar, N. Memon, and I. Avcibas, "Source camera identification based on cfa interpolation," in *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, vol. 3, Sept 2005, pp. III–69–72.
[8] N. Khanna, A. K. Mikkilineni, G. T.-C. Chiu, J. P. Allebach, and E. J. Delp, "Scanner identification using sensor pattern noise," in *Proc. of SPIE*, 2007.
[9] H. Gou, A. Swaminathan, and M. Wu, "Robust scanner identification based on noise features," in *Proc. of SPIE*, vol. 6505, 65050S, 2007.
[10] J. Fridrich, "Digital image forensic using sensor noise," *IEEE Signal Processing Magazine*, vol. 26, no. 2, pp. 26–37, 2009.
[11] J. Lukás, J. Fridrich, and M. Goljan, "Digital camera identification from sensor pattern noise," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 2, pp. 205–214, 2006.
[12] I. Amerini, R. Caldelli, A. D. Bimbo, A. D. Fuccia, A. P. Rizzo, and L. Saravo, "Detection of manipulations on printed images to address crime scene analysis: A case study," *Forensic Science International*, no. 0, pp. –, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0379073815001267
[13] I. Amerini, R. Caldelli, A. Del Bimbo, A. Di Fuccia, L. Saravo, and A. P. Rizzo, "Copy-move forgery detection from printed images," pp. 90 280Y–90 280Y–10, 2014. [Online]. Available: http://dx.doi.org/10.1117/12.2039509

[7]Flickr4Java Library, https://github.com/callmeal/Flickr4Java
[8]FB Album Mod for Chrome, https://chrome.google.com/webstore/detail/download-fb-album-mod/cgjnhhjpfcdhbhlcmmjppicjmgfkppok?hl=en

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2656842

[14] M. Goljan, J. Fridrich, and J. Lukáš, "Camera identification from printed images," in *Electronic Imaging 2008*. International Society for Optics and Photonics, 2008, pp. 68 190I–68 190I.
[15] A. E. Dirik and N. Memon, "Image tamper detection based on demosaicing artifacts," in *2009 16th IEEE International Conference on Image Processing (ICIP)*, Nov 2009, pp. 1497–1500.
[16] E. Kee, J. F. O'brien, and H. Farid, "Exposing photo manipulation from shading and shadows," *ACM Trans. Graph.*, vol. 33, no. 5, pp. 165:1–165:21, Sep. 2014. [Online]. Available: http://doi.acm.org/10.1145/2629646
[17] M. Goljan, J. Fridrich, and T. Filler, "Managing a large database of camera fingerprints," in *Media Forensics and Security*, ser. SPIE Proceedings, N. D. Memon, J. Dittmann, A. M. Alattar, and E. J. Delp, Eds., vol. 7541. SPIE, 2010, p. 754108.
[18] D. Valsesia, G. Coluccia, T. Bianchi, and E. Magli, "Compressed fingerprint matching and camera identification via random projections," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 7, pp. 1472–1485, July 2015.
[19] C.-T. Li, "Unsupervised classification of digital images using enhanced sensor pattern noise," *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS'10)*, pp. 3429–3432, 2010.
[20] I. Amerini, R. Caldelli, P. Crescenzi, A. Del Mastio, and A. Marino, "Blind image clustering based on the normalized cuts criterion for camera identification," *Signal Processing: Image Communication*, vol. 29, no. 8, pp. 831 – 843, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S092359651400109X
[21] I. Amerini, R. Becarelli, B. Bertini, and R. Caldelli, "Acquisition source identification through a blind image classification," *IET Image Processing*, vol. 9, pp. 329–337(8), April 2015. [Online]. Available: http://digital-library.theiet.org/content/journals/10.1049/iet-ipr.2014.0316
[22] H. Farid, "Exposing digital forgeries from jpeg ghosts," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 1, pp. 154–160, March 2009.
[23] A. Swaminathan, M. Wu, and K. J. R. Liu, "Digital image forensics via intrinsic fingerprints," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 1, pp. 101–117, March 2008.
[24] B. Mahdian and S. Saic, "Blind authentication using periodic properties of interpolation," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 3, pp. 529–538, Sept 2008.
[25] M. Kirchner, "On the detectability of local resampling in digital images," pp. 68 190F–68 190F–11, 2008. [Online]. Available: http://dx.doi.org/10.1117/12.766902
[26] L. Zhouchen, H. Junfeng, T. Xiaoou, and T. Chi-Keung, "Fast, automatic and fine-grained tampered JPEG image detection via DCT coefficient analysis," *Pattern Recognition*, vol. 42, no. 11, pp. 2492 – 2501, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0031320309001198
[27] T. Bianchi and A. Piva, "Image forgery localization via block-grained analysis of JPEG artifacts," *Information Forensics and Security, IEEE Transactions on*, vol. 7, no. 3, pp. 1003–1017, 2012.
[28] S. Milani, M. Tagliasacchi, and S. Tubaro, "Discriminating multiple JPEG compression using first digit features," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2012, pp. 2253–2256.
[29] I. Amerini, R. Becarelli, R. Caldelli, and A. Del Mastio, "Splicing forgeries localization through the use of first digit features," in *Information Forensics and Security (WIFS), 2014 IEEE International Workshop on*, Dec 2014, pp. 143–148.
[30] T. Bianchi and A. Piva, "Detection of nonaligned double jpeg compression based on integer periodicity maps," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 842–848, April 2012.
[31] T. Pevny and J. Fridrich, "Detection of double-compression in jpeg images for applications in steganography," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 2, pp. 247–258, June 2008.
[32] J. He, Z. Lin, L. Wang, and X. Tang, *Detecting Doctored JPEG Images Via DCT Coefficient Analysis*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 423–435.
[33] Q. Liu, A. H. Sung, and M. Qiao, *A Method to Detect JPEG-Based Double Compression*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 466–476.
[34] A. A. de Oliveira, P. Ferrara, A. De Rosa, A. Piva, M. Barni, S. Goldenstein, Z. Dias, and A. Rocha, "Multiple parenting phylogeny relationships in digital images," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 2, pp. 328–343, Feb 2016.
[35] A. Melloni, P. Bestagini, S. Milani, M. Tagliasacchi, A. Rocha, and S. Tubaro, "Image phylogeny through dissimilarity metrics fusion," in *Visual Information Processing (EUVIP), 2014 5th European Workshop on*, Dec 2014, pp. 1–6.
[36] I. Amerini, R. Becarelli, R. Caldelli, and M. Casini, "A feature-based forensic procedure for splicing forgeries detection," *Mathematical Problems in Engineering*, p. 9, 2015.
[37] M. Moltisanti, A. Paratore, S. Battiato, and L. Saravo, *Image Analysis and Processing — ICIAP 2015: 18th International Conference, Genoa, Italy, September 7-11, 2015, Proceedings, Part II*. Cham: Springer International Publishing, 2015, ch. Image Manipulation on Facebook for Forensics Evidence, pp. 506–517.
[38] A. NG, L. Pan, and Y. Xiang, *Applications and Techniques in Information Security: 5th International Conference, ATIS 2014, Melbourne, VIC, Australia, November 26-28, 2014. Proceedings*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, ch. A Novel Method for Detecting Double Compressed Facebook JPEG Images, pp. 191–198.
[39] A. V. Mire, S. B. Dhok, N. J. Mistry, and P. D. Porey, "Localization of tampering created with facebook images by analyzing block factor histogram voting," *Int. J. Digit. Crime For.*, vol. 7, no. 4, pp. 33–54, Oct. 2015.
[40] A. Castiglione, G. Cattaneo, and A. De Santis, "A forensic analysis of images on online social networks," in *Intelligent Networking and Collaborative Systems (INCoS), 2011 Third International Conference on*, Nov 2011, pp. 679–684.
[41] T. K. Ho, "Random decision forests," in *Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, QC*, August 1995, pp. 278–282.
[42] G. Schaefer and M. Stich, "UCID - an uncompressed colour image database," in *Storage and Retrieval Methods and Applications for Multimedia 2004*, ser. Proceedings of SPIE, vol. 5307, 2004, pp. 472–480.

Roberto Caldelli (M11) received the degree in electronic engineering and the Ph.D. degree in computer science and telecommunication from the University of Florence, Florence, Italy, in 1997 and 2001, respectively. From 2005 to 2013, he was an Assistant Professor with the Media Integration and Communication Center, University of Florence. In 2014, he joined the National Inter-University Consortium for Telecommunications (CNIT) as a Permanent Researcher. His main research activities, witnessed by several publications, include digital image processing, interactive television, image and video digital watermarking, and multimedia forensics. He holds two patents in the field of content security and multimedia interaction. From 2016 he is member of the IEEE Information Forensics and Security Technical Committee of the Signal Processing Society.

Rudy Becarelli graduated in Electronic Engineering in February 2004 at University of Florence with a thesis concerning motion estimation algorithms and their applications. He has been involved in research and development activities at University of Florence since 2004, he has obtained PhD in Computer Science, Systems and Telecommunications at the University of Florence on April 2016. Research activities mostly concern with digital watermarking, interactive TV and image forensics. He is expert in design and development of J2EE applications for data exchange and marshalling with rich client platforms.

Irene Amerini (M17) 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. She is currently a post-doc researcher at the Media Integration and Communication Center, University of Florence, Italy. She was a visiting scholar at Binghamton University, NY, in 2010. Her main research interests focus on multimedia content security technologies, secure media, multimedia forensics.

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.