IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received January 25, 2019, accepted March 4, 2019, date of publication March 18, 2019, date of current version April 1, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2903876

Social Network Identification Through Image Classification With CNN

IRENE AMERINI[1,2], (Member, IEEE), CHANG-TSUN LI[2,3], (Senior Member, IEEE), AND ROBERTO CALDELLI[1,4], (Senior Member, IEEE)

[1]Media Integration and Communication Center, University of Florence, 50134 Florence, Italy
[2]School of Computing and Mathematics, Charles Sturt University, Wagga Wagga, NSW 2650, Australia
[3]Department of Computer Science, University of Warwick, Coventry CV4 7AL, U.K.
[4]National Inter-University Consortium for Telecommunications, 43124 Parma, Italy

Corresponding author: Irene Amerini (irene.amerini@unifi.it)

This work was supported in part by the Marie Skłodowska-Curie Action of the EU H2020 Programme through the project IDENTITY: Computer Vision Enabled Multimedia Forensics and People Identification under Grant 690907, in part by the Australia Awards-Endeavour Scholarship and Fellowship offered by the Australian Government Department of Education and Training.

ABSTRACT Identification of the source social network based on the downloaded images is an important multimedia forensic task with significant cybersecurity implications in light of the sheer volume of images and videos shared across various social media platforms. Such a task has been proved possible by exploiting distinctive traces embedded in image content by social networks (SNs). To further advance the development of this area, we propose a novel framework, called *FusionNET*, that integrates two established convolutional neural networks (CNNs), with the former (named *1D-CNN*) learning discriminative features from the histogram of discrete cosine transform coefficients and the latter (named *2D-CNN*) inferring unique attributes from the sensor-related noise residual of the images in question. The separately learned features are then fused by the *FusionNET* to inform the ensuing source identification or source-oriented image classification component. A series of experiments were conducted on a number of image datasets across various SNs and instant messaging apps to validate the feasibility of the *FusionNET* also in comparison with the performance of the *1D-CNN* and *2D-CNN*. The encouraging results were observed.

INDEX TERMS Source identification, multimedia forensics, image provenance, CNN, noise residual, cybersecurity.

I. INTRODUCTION

One of multimedia forensics main tasks is to infer the origin of a digital content. Surely linking a digital image to its source device [1] is one of the challenging problems faced by the research community together with that to recover the entire processing history an image under verification has been subjected to [2]. Applications such as identification of source camera device [3], common source inference [4], content integrity verification and source-oriented image clustering [5] have been devised to help with these tasks. With social networks (SNs, such as *Facebook*) or instant messaging apps (IMAs, such as *WhatsApp*) taking the central role of multimedia circulator, source SN and IMA identification becomes as important as the afore-mentioned source device identification. It has been proved that each social network or instant messaging app leaves specific traces in the content of images during the uploading and downloading processes [6]–[8]. Such traces, when properly extracted and exploited, can be used to serve as signatures of the platforms for identification purposes.

The goal of SN and IMA identification in multimedia forensics is to establish the provenance of an image based on the characteristics imprinted by those platforms within the image. This kind of forensic analysis may provide crucial information for combating cyber crimes, such as harassment, violence instigation, cyber bullying and cyber terrorism. In fact, in a forensic scenario, succeeding in identifying the social network of provenance of a certain image could be instrumental in addressing an investigation, analyzing in an automatic fashion large amounts of data extracted from personal devices of a suspect (e.g. smartphone, PC,

The associate editor coordinating the review of this manuscript and approving it for publication was Yan-Jun Liu.

SD card, hard disk) or from his *Facebook* profile. Being able to discern if those images are downloaded from a social network or directly captured by a digital camera, can be crucial in leading consecutive investigations, targeting the proper forensic method in order to decide on media integrity.

To tackle such an identification problem, previous approaches have based their analysis on metadata [6], [9], hand-crafted image features [10] or image features automatically learned by Convolutional Neural Networks (CNNs) [8], [11]. Although preliminary successes have been made, the problem to identify the source SN and/or IMA still remain largely unexplored; furthermore, some of the previous works present certain limitations such as the usage of metadata that can be easily manipulated. These have motivated us to propose in this work a novel deep learning framework that harnesses a diverse set of features derived directly from digital image content to better address the problem. In particular, the histogram of DCT coefficients proposed in [7] and the noise residuals introduced in [8] are taken into consideration. The rationale behind this choice is that the combination of diverse information derived from different modalities allows to obtain better generalization in the model to track specific traces left by the operations applied by each SN and IMA. In particular, DCT-based features give information mainly about the way re-compression is performed by the SNs and IMAs. On the other hand, noise residuals should convey knowledge about the resizing operation.

The main contributions of this work are as follows. First, a combination of different image features is employed to better detect the traces left by SN/IMA operations during uploading and downloading. Second, a novel CNN architecture, named *FusionNET*, is devised to take advantage of the combined learning power of the inter-layer activations of two single-feature-based CNNs. Third, a comprehensive series of comparative experiments are conducted on a number of publicly available image datasets, not only to validate our methodology, but also to shed light on the ways of harnessing multiple typologies of data.

The rest of the paper is organized as follows. We start with reviewing related works in Section II and then an overview of the single-feature-based CNNs and the way they are used as the building blocks of the proposed *FusionNET* are presented in Section III. Section IV reports the experiments and results on a number of image datasets. Finally, conclusions are drawn in Section V.

II. RELATED WORKS

The fact that the process of uploading images onto *Facebook* does leave unique and detectable traces in the content has been proved in a preliminary work introduced in [10]. Some of these traces are metadata alterations, image size and recompression [6], [9]. The authors of [10] refined the idea in [6] and used a K-NN classifier to separate different social networks based on the traces of resizing, recompression, renaming and metadata alterations left during the uploading and downloading processes. In [7], content-based information extracted from the histograms of the DCT (Discrete Cosine Transform) coefficients of JPEG images is used by a Bagged Decision Tree Classifier to differentiate among social networks (*Facebook*, *Flickr* and *Twitter* among others). In addition to content-based features as in [7], metadata (such as image dimension and quantization tables) are also included in [9] to inform the process of distinguishing among various IMAs (e.g. *WhatsApp*, *Telegram* and *Messenger*). Three classifiers, namely LR (Logistic Regression), SVM (linear Support Vector Machine) and RF (Random Forest) are used in [9] to perform the identification task.

Albeit its short adoption history, CNNs have emerged as an effective tools in many multimedia applications, including detection of image manipulations [12], [13], [14], [15], inference of image processing history [16], identification of source cameras [17], [18], [19] and analysis of anti-forensics techniques [20], [21]. Attempts of adopting CNNs for source SN identification have also been made in [8] and [11]. In particular in [11], a CNN is proposed to classify up to 8 different social networks based on the features extracted from the histograms of the DCT coefficients. Differently from the afore-described works, in [8] the idea of using Photo Response Non Uniformity Noise - PRNU [3] extracted from images as the carrier of SN traces is presented. In [8], a CNN based on noise residual is trained for SN identification in a similar manner as in [11] .

Driven by great interests in source SN and IMA identification, many benchmarking datasets have been proposed such as in [6], [9], [7], and [22], demonstrating the importance of the problem addressed in this paper. Most of these datasets will be used in this paper as well to validate the proposed method in a comparative fashion.

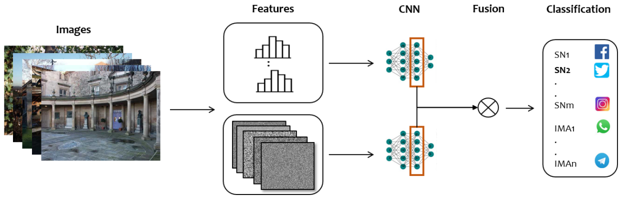


FIGURE 1. The proposed pipeline for social media identification.

III. THE PROPOSED METHOD

In this section, the proposed CNN-based framework, called *FusionNET*, for addressing the social network and instant messaging app identification problem is introduced. Figure 1 illustrates the conceptual framework of *FusionNet* with four main sequential components: i) dual-modal features for image representation, with one feature modality being the histogram of DTC coefficients and the other being the sensor-related noise residuals; ii) two different CNN branches fed with the respective feature modalities to pull out activation vectors; iii) fusion of activation vectors, and iv) classification of source SNs and IMAs of the images in question.

A. INPUT DATA

The different types of features used as input for the proposed CNN architecture are described as follows. In particular, we employed DCT-based features and image noise residuals to represent an image in two different domains in order to capture the signatures of the diverse unknown processing applied by each social network to every image during the process of uploading and downloading. In fact, the rationale behind such a choice is that this set of low-level features are able to reveal the signatures of various transformations such as JPEG recompressions, which can be detected from the DCT-based features, and resizing, which can be observed in the sensor-related noise residuals of an image.

1) DCT-BASED FEATURES

The use of the histogram of DCT coefficients as input data for social media identification has been shown to be feasible as presented in [7] and [11]. It is particularly effective in recognizing JPEG recompression at various quality factors. In this work, each image is subdivided into non-overlapping patches of $N \times N$ pixels. For each patch, DCT values are considered in all 8 × 8 blocks, and for each DCT block, the first 9 spatial frequencies in the zig-zag scan order are taken into account (i.e. the DC coefficient is excluded). For each spatial frequency (f_i, f_j) across all DCT blocks of each patch, the histogram h_{f_i,f_j} representing the occurrences of the quantized DCT coefficients at (f_i, f_j) is built. In particular, the occurrences of the coefficient with a value $v \in (-50; 0; +50)$ in the histogram h_{f_i,f_j} are taken. So a feature vector of 909 elements (i.e. 101 histogram bins × 9 DCT coefficients), associated with each $N \times N$-pixel patch (usually $N = 64$), is used as CNN input.

2) SENSOR-RELATED NOISE RESIDUALS

To better capture the distinctive features of the social networks, we decide to use the high-frequency sensor-related noise residuals of the images, rather than the original images themselves. This is because it has been shown that the noise residuals can be altered more significantly than the bulk of the original images by the uploading and downloading processes of social media platforms [8]. That is to say, social media platforms leave their fingerprints mainly in the noise residuals. Therefore, in social network identification, using the noise residuals should allow CNN to focus on the relevant information.

The noise residual is extracted from the image content through high-pass filtering. The term N_i stands for the noise residual, while I_i and I_i^{den} represent the i-th image and its denoised version, respectively.

$$N_i = I_i - I_i^{den} \tag{1}$$

Different kinds of denoising filters have been introduced to improve noise residual extraction [23]. In this work, we adopt the wavelet-based approach described in [24], commonly used for noise residual extraction and source device identification [3]. The noise residual N_i , as in Equation (1) is extracted at the full-frame size. The intensity of each noise residual pixel $N_i(x, y)$ is then scaled and normalized to the range [0 : 1] to obtain N_i^{norm}. Finally, N_i^{norm} is subdivided into non-overlapping squared patches of $N \times N$ size in order to consistently provide the CNN with the same number of to-be-learnt features.

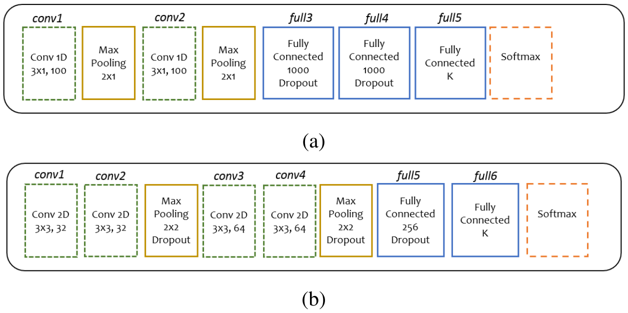


FIGURE 2. CNNs for social media identification. a) 1D-CNN [11] and b) 2D-CNN [8].

B. CNN FOR SOCIAL MEDIA IDENTIFICATION

In this section, we briefly introduce the two CNN-based methods, outlined in Section II, that constitute the basic structure of the two branches for the proposed framework, as shown in Figure 1. The first one [11] is named *1D-CNN* and the second [8] is called *2D-CNN*. Both of them will also be used for comparison in the experiments in Section IV. The input to the first CNN is a vector of 909 elements (101 histogram bins × 9 DCT frequencies) as already explained in Section III-A.1 while, for the second net, the input is a bi-dimensional matrix of size $N \times N$ with $N = 64$ (see Section III-A.2). The details of the two CNNs are described as follows.

- **1D-CNN**: As shown in Figure 2a), this CNN is composed of the following sequence of layers. 1) Two convolutional layers, *conv1* and *conv2*, each consisting of 100 filters of size 3 × 1 and followed by a 2 × 1 max pooling to reduce the size. 2) Two fully-connected layers (*full3*, *full4*) in cascade with 1000 dropout units. 3) A fully-connected layer (*full5*) with k units. 4) A softmax layer with as many units as the number of social networks to be identified. In particular, the softmax produces the probability of each sample being classified into each class. All convolutional layers use the rectified linear unit (ReLU) as activation function.
- **2D-CNN**: This CNN (see Figure 2b) is composed of the following sequence of layers. 1) Two convolutional layers, *conv1* and *conv2* consisting of 32 filters of size 3 × 3, with the last one followed by max pooling (2 × 2) and dropout. 2) *conv3* and *conv4* made up of 64 filters of size 3 × 3 followed again by max pooling of 2 × 2 and dropout. 3) A fully connected layer (*full5*) with 256 units plus another fully connected layer (*full6*) and a softmax layer with as many units as the number of classes k to be identified. Since we are dealing with a multi-class

problem, the output will be the probability that the image in question belongs to a specific class as in *1D-CNN*. All convolutional layers and the first fully-connected layer also use the ReLU as activation function.

During CNN training, the weights are learned using the mini-batch stochastic gradient descent algorithm with the AdaDelta optimizer [25] with dropout set to 0.5. The number of epochs is limited to 20 in *1D-CNN* while 50 epochs is the training stage limit for the *2D-CNN* net. At each epoch, we use a mini-batch of 32 samples for *1D-CNN* and a mini-batch of 256 samples for the *2D-CNN*. Each mini-batch is randomly selected from an unbalanced training set. In fact, in order to simulate real application scenarios, the number of images for each class is not the same and is determined by the image size. The training phase is stopped when the loss function on the validation set reaches its minimum and then the model associated with a certain epoch is selected. In our experiment that usually happens after ten/twenty epochs for *2D-CNN* net and before ten epochs for both *FusionNET* and *1D-CNN*. To run our experiments, we use the implementation of CNN provided by Keras[1] TensorFlow[2] that provides a high-level API to build and train deep learning models. A categorical cross-entropy function [26] is employed as loss function to guide the training process of the classification problem.

The full-frame images are split into patches of a fixed dimension and those patches are classified independently. Consequently the prediction is obtained at the patch level after processing each image patch with CNNs. Therefore, in order to derive a final identification at image level, a majority voting strategy on the labels assigned to each patch is applied.

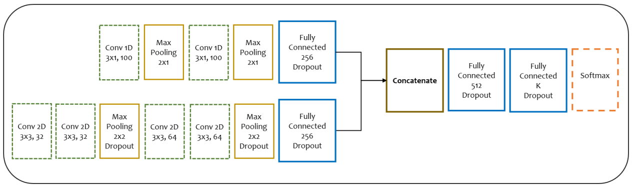


FIGURE 3. The architecture of the proposed *FusionNET*.

The way of fusing the two CNNs depicted in Figure 1to create our proposed framework, called *FusionNET*, is described as follows. The fusion could take place at any layers before the *softmax* to perform a combination learning to get information from the activations at that stage. In our case, we decide to locate the fusion before the first fully connected layer of each CNN. The activations of those layers are then concatenated and fed into a new set of additional layers to perform the actual fusion, as depicted in Figure 3. To ensure that both branches have balanced encoded vectors before they converge to the concatenation stage, the fully connected layer of both branches is made of 256-dimensional. After that, a fully connected layer composed of 512 neurons plus RELU and dropout, another fully connected layer and a softmax with as many units as the number of classes to be categorized are attached. With this framework, the training is performed simultaneously by both branches, with the same image patch as input. During the training process, the weights of *FusionNET* are shared by both branches in order to automatically learn their best combination.

IV. EXPERIMENTAL RESULTS

The experiments carried out to understand if the proposed *FusionNET* and specifically the fused features learned can be adopted to reliably track the source social network of a certain image are introduced in this section. The datasets used will be described in IV-A, while the subsection IV-B will cover a detailed analysis on noise residual features and in IV-C different kinds of experiments will be presented and discussed on each of the dataset taken into consideration.

A. DATASETS

We run our experiments on the following four datasets that were employed in related prior works [6], [7], and [22]. All these datasets present very different characteristics in terms of kind of images, number and types of social networks, amount of involved source cameras and also the number of sharings.

- **UCID social**[3]: This dataset consists of JPEG compressed images generated at different quality factors $QF = 50 : 95$ (step of 5) based on images from *UCID* (Uncompressed Colour Image Database) [27]. The *UCID* database is composed of 1338 images of 512×384 pixels acquired by a *Minolta Dimage 5* digital camera in raw format. Each JPEG compressed images has subsequently been uploaded and then downloaded from three selected social networks (*Flickr*, *Facebook* and *Twitter*), yielding this *UCID social* dataset of 40140 images (1338 images×10 QFs×3 social networks).
- **UCID social-DS (double sharing)**: Each image in the *UCID social* dataset was uploaded and downloaded twice from each of the three aforementioned social platforms *Facebook*, *Flickr* and *Twitter* (*Fb*, *Fl* and *Tw* in short) to create this *UCID social-DS* dataset with a total of 120420 images. So this dataset is an extension of the set mentioned above and contains the previous 40140 images of *UCID social* (single sharing) plus 80280 additional ones, that is 13380 images (1338 images×10 QFs) × 6 double sharing combinations (*FbTw, FbFl, TwFb, TwFl, FlTw, FlFb*).
- **IPLAB**[4]: Five social networks (*Facebook*, *Flickr*, *Google+*, *Instagram* and *Twitter*), two instant messaging apps (*WhatsApp* and *Telegram*) and one class of unshared images (just taken by the camera) are

[1] https://keras.io/

[2] https://www.tensorflow.org/guide/keras

[3] Dataset available at the following link: http://lci.micc.unifi.it/labd/2015/01/trustworthiness-and-social-forensic/

[4] Dataset available at the following link:http://iplab.dmi.unict.it/DigitalForensics/social_image_forensics/

chosen to compose the *IPLAB* dataset. So, in this circumstance, 8 classes are involved and the dataset consists of 1920 images (240×8 classes). The picture resolutions range from 640×480 to 5184×3456 pixels and the devices involved in the creation of the *IPLAB* dataset are *Canon 650D*, *QUMOX SJ4000*, *Samsung Note3 Neo* and *Sony Powershot A2300*.

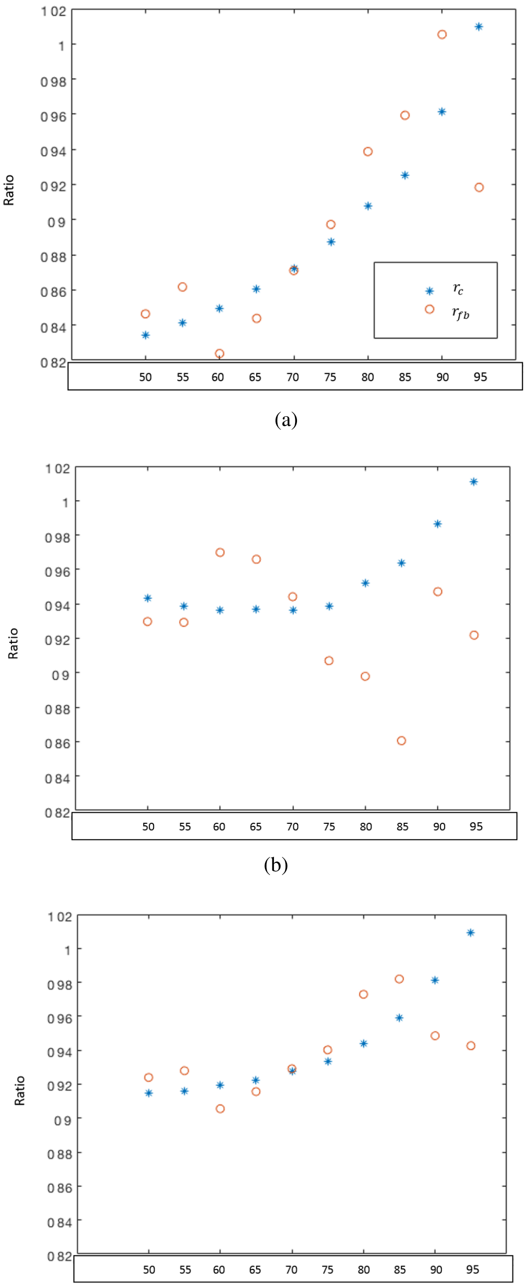
- **VISION subset**: A total of 2135 images are selected from *VISION* dataset [22][5] attribuited to 10 smartphones (*Samsung Galaxy S3 mini*, *Huawei P9*, *LG D290*, *Apple iPhone5c*, *Apple iPhone6*, *Lenovo P70A*, *Samsung GalaxyTab3*, *Apple Iphone4* and 2 models of *Apple iPhone4s*). All the images have been uploaded onto *Facebook* and then downloaded in high and low quality, and through *WhatsApp*, resulting in a total of 6405 (2135×3) images. Image dimensions vary according to the smartphone camera resolution and the number of samples per device is not uniform.

B. ANALYSIS ON NOISE RESIDUAL FEATURES

The rationale behind the use of DCT-based features for social network identification has already been discussed in [7]. In this subsection, we outline some interesting observations that motivate the use of noise residual features. In particular, we want to analyze how the up/downloading of an image on a social network affects the local variance of noise residual following the idea of the paper in [28], where the variance is measured on image intensity to deal with JPEG compression. To this end, we consider images of the *UCID social* dataset described in the previous subsection and for each observed image, we extract the noise residual according to Equation (1). The local variance for each 7×7 overlapping noise residual block (with a stride of 1 pixel) is computed. To obtain a single reference value for each image, we summed up the local variance of each block. We then evaluate, for each *QF* in *UCID social*, the ratio r_c between the local variance of the noise residuals of uncompressed images (N_{Orig}) with respect to the compressed images but not uploaded onto any social networks (N_{Comp}). We also calculate the ratio r_{SN} between, again, the local variance of uncompressed images and the compressed images uploaded onto/downloaded from a social network (N_{SN}), as summarized in Equation (2), where SN takes values from Fb, Fl, Tw depending on the social media in question.

$$r_C = \frac{Var(N_{Orig})}{Var(N_{Comp})}, \quad r_{SN} = \frac{Var(N_{Orig})}{Var(N_{SN})} \tag{2}$$

In Figure 4, the ratios r_{fb} for each $QF = 50:95$, for the case of *Facebook* ($r_{SN} = r_{fb}$), are illustrated. In particular, Figure 4(a) and 4(b) are related to two different images from the *UCID social* dataset (number #1 and number #5), while Figure 4(c) is related to the entire *UCID social* dataset, with r_c and r_{fb} averaged over all images. Figure 4(c) shows the global trend of the noise residual variance of the entire dataset. The difference between r_c and r_{fb} as demonstrated in Figure 4 is significant enough to differentiate the two classes (i.e. processed or not processed by *Facebook*).



(a)

(b)

(c)

FIGURE 4. **Ratio between local variances of uncompressed/compressed images r_C (blue star) and between uncompressed/*Facebook* images r_{Fb} (red circle). In (a) and (b), the results are related to image number #1 and number #5 of *UCID social* dataset, while in (c) results on the entire *UCID social* dataset.**

Similarly, we plotted in Figure 5 the values of r_{Fb} and r_{Tw} with respect to various *QF* to demonstrate the impact *Facebook* and *Twitter* have on the noise residuals of the images from *UCID social* dataset. It is evident that, in most cases, the ratios are different enough to distinguish the diverse behaviors of the two social networks.

[5] Dataset available here:ftp://lesc.dinfo.unifi.it/pub/Public/VISION/

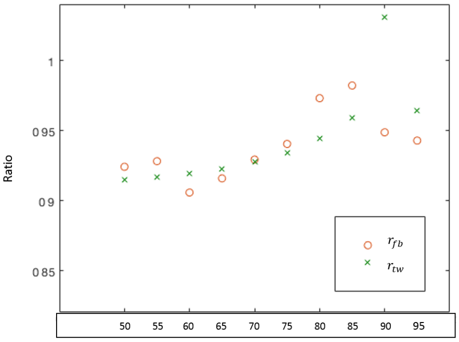


FIGURE 5. Ratios of noise residual variance r_{Fb} (red circles) and r_{Tw} (green crosses) with respect to *Facebook* and *Twitter*.

C. PERFORMANCE EVALUATION

A comprehensive set of experiments are presented in this subsection with the aim to show the performance of the proposed method in relation to different datasets composed of disparate image resolutions, social networks and messaging applications. Each of the datasets has been divided into training, validation and test sets with the proportion of 80%, 10% and 10%, respectively. Images of the three subsets are randomly selected. The input of the *FusionNET* is a 64×64 matrix and the outputs are k social network classes which vary according to the dataset under analysis.

1) UCID SOCIAL

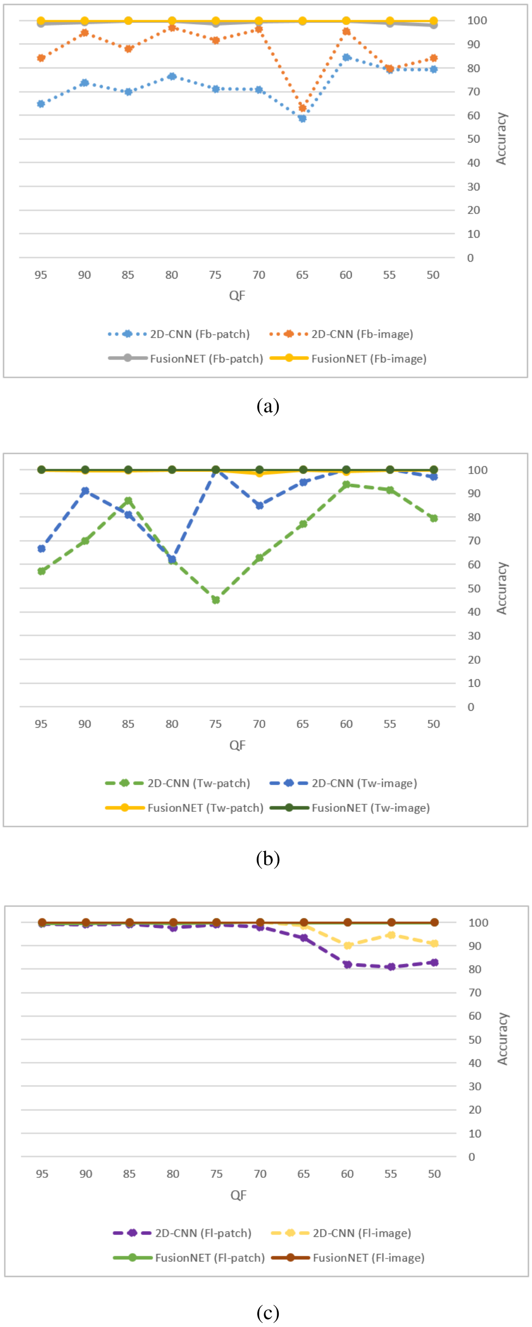
In this section, we examine the impact of combining different features in terms of SN classification accuracy on *UCID social* dataset with the proposed *FusionNET* approach.

TABLE 1. Performance in terms of classification accuracy (%) on *UCID social* dataset obtained by the three CNNs: *1D-CNN, 2D-CNN*, and *FusionNET*.

	Patch level			Image level		
	Fb	Fl	Tw	Fb	Fl	Tw
1D-CNN	96.15	99.79	99.30	97.37	100	100
2D-CNN	72.80	93.15	72.49	87.35	97.42	87.73
FusionNET	99.15	99.83	99.44	100	100	100

Table 1 shows the comparison among *FusionNET* and the state-of-the-art CNNs described in Section III: *1D-CNN* with DCT-based features and *2D-CNN* based on noise residuals. Results have been computed by averaging over the accuracy on all the considered JPEG QFs on the *UCID social* dataset. From Table 1, it appears that the proposed fusion method has the best average performance globally all over the three classes and a more stable behavior across the three different classes even though, at least in this circumstance, the *1D-CNN* with DCT-based features yields a very similar accuracy.

Specifically, when classifying *Facebook* images, the proposed *FusionNET* outperforms *1D-CNN* and *2D-CNN* by around 3% and more than 12%, respectively. In Figure 6, the comparison between the accuracy of the *FusionNET* and *2D-CNN* relatively to each of the 10 JPEG QFs is reported. The performance of *FusionNET* is almost always 100% both at the patch and image levels QFs; on the other hand, *2D-CNN*'s performance in both cases varies significantly and, in general, this performs better at image level. Table 2 presents a more in-depth view of the very good performances achieved by *FusionNET* for each of the JPEG QFs at both patch and image level.



(a)

(b)

(c)

FIGURE 6. Accuracy on *UCID social* dataset for *Facebook* (Fb) (a), *Twitter* (Tw) (b) and *Flickr* (Fl) (c). Comparison between *FusionNET* and *2D-CNN* at patch and image level.

The afore-mentioned experiment was conducted and evaluated with each of the 10 JPEG quality factors considered

TABLE 2. ***FusionNET*'s performance in terms of classification accuracy (%) on the *UCID social* dataset for *Facebook* (Fb), *Flickr* (Fl), and *Twitter* (Tw).**

QF	*Patch level*			*Image level*		
	Fb	**Fl**	**Tw**	**Fb**	**Fl**	**Tw**
50	98.02	99.93	99.84	100	100	100
55	98.80	99.93	99.82	100	100	100
60	99.82	99.93	99.11	100	100	100
65	99.67	99.93	99.55	100	100	100
70	99.31	100	98.32	100	100	100
75	98.56	99.89	99.84	100	100	100
80	99.74	99.68	99.84	100	100	100
85	99.78	99.74	99.57	100	100	100
90	99.14	99.65	99.68	100	100	100
95	98.68	99.57	99.83	100	100	100

TABLE 3. **Confusion matrix, in terms of classification accuracy, of the proposed *FusionNET* when applied to images processed by *Facebook*, *Flickr*, and *Twitter* at patch level with mixed *QF*s on *UCID* dataset.**

Classification (%) vs SNs	Fb	Fl	Tw
Facebook	**96.04**	0	3.96
Flickr	0.16	**98.50**	1.43
Twitter	1.12	0.02	**98.86**

independently at the training stage. As described below, we have also conducted experiments by training/testing with images of all *QF*s mingled to validate the feasibility of the proposed *FusionNET* in a more realistic scenario. In the experiment whose results are reported in the form of confusion matrix in Table 3, an equal number of images (135) for each *QF* have been randomly considered in order to get 1300 images as it previously happened for each QF independently taken. Again in this case, the *FusionNET* demonstrates good performances. It is also interesting to see that when the "*Facebook* images" are misclassified, they tend to be exchanged with "*Twitter* images" and vice versa. This, to some extent, suggests that the processing applied by these two networks on the images shares more commonality than the processing attributed to *Flickr*. On the other hand, when "*Flickr* images" are misclassified, the majority of them are confused as "*Twitter* images" rather than "*Facebook* images".

2) UCID SOCIAL-DS

Regarding the *UCID social-DS* dataset, a new series of experiments have been set up to evaluate the classification performances for the proposed *FusionNET*. In particular, we intend to verify the capacity of the proposed method to deal with double sharing and to understand how *FusionNET* behaves when it is asked to classify an image downloaded from social network A having previously been downloaded from social network B (see Section IV-A for details related to the dataset). The objective was to comprehend if the method is still able to recognize the last SN despite the passage onto another social platform. In Table 4, the average of the classification accuracy, in the case of training over all JPEG *QF*s as done just before, is reported over the three classes for *FusionNET* and *1D-CNN* (upper and bottom part of the Table respectively). The results for the double sharing configuration are obtained keeping the training phase fixed as in the single sharing configuration. The case of single upload/download (*single sharing*) is shown in the left part of Table 4 and this result is compared with the accuracy obtained in case of double shared images (*double sharing*). It is important to underline that accuracy values are now given at the image level. In particular, an average accuracy of 98.94% for single sharing configuration (similarly to what happened in Table 3) and 86.49% for the double sharing configuration are achieved in the case of *FusionNET*. This demonstrates the possibility to obtain a good identification of the social network provenance even in this more challenging circumstance. It is worth pointing out that *Twitter* achieves the lower value of accuracy (see Table 4), demonstrating that some transformations applied to an image by other SNs (*Facebook* and *Flickr*) during the first pass embed within the image much stronger information compared to the one inserted by *Twitter* in the last stage. Furthermore, the proposed method has obtained a gain in classification of about 16% in the case of double sharing scenario respect to the *1D-CNN* net for the *Twitter* class proving that, especially for more challenging dataset, in which a double sharing is involved, the use of more features in combination is a good strategy to achieve better results in classification.

TABLE 4. ***FusionNET* (top) and *1D-CNN* (bottom) performance on *UCID social-DS* in terms of classification accuracy (%) among *Facebook*, *Flickr*, and *Twitter* at image level.**

FusionNET	*Single sharing*			*Double sharing*		
	Fb	**Fl**	**Tw**	**Fb**	**Fl**	**Tw**
Facebook	**97.26**	0	2.74	**94.92**	0.04	5.04
Flickr	0.07	**99.56**	0.37	0.32	**96.17**	3.51
Twitter	0	0	**100**	31.54	0.07	**68.40**
1D-CNN	*Single sharing*			*Double sharing*		
	Fb	**Fl**	**Tw**	**Fb**	**Fl**	**Tw**
Facebook	96.47	0	3.53	94.88	0.48	4.65
Flickr	0.10	98.83	1.06	3.34	95.74	0.92
Twitter	0.86	0.01	99.14	47.17	0.10	52.72

3) IPLAB

In this subsection, the proposed method has been tested over the *IPLAB* dataset, defined in IV-A. Differently from the previous cases, now a higher number of social networks and two instant messaging applications are taken into account together with images directly from digital cameras. Results over this dataset are reported in Table 5, showing a patch-level average accuracy of 94.77% against an average of 93.17% obtained with *1D-CNN* and 72.92% with *2D-CNN* (detailed confusion matrices have not been reported for the sake of readability). In particular, by using the *FusionNET* technique, more stable results, especially for the *Facebook* class are achieved. In fact it is possible to obtain a gain in the accuracy of more than 4% with respect to the *1D-CNN* (i.e. 91.31% vs 87.12%; the reader is referred to [11] for detailed results of *1D-CNN*).

TABLE 5. Confusion matrix for the 8 classes of the *IPLAB* dataset: accuracy percentages (patch level) are reported obtained with *FusionNET*.

Classification (%) vs SNs	Facebook	Flickr	Google+	Instagram	Original	Telegram	Twitter	WhatsApp
Facebook	**91.31**	6.21	0.00	0.08	2.40	0.00	0.00	0.00
Flickr	0.90	**86.77**	0.03	0.18	3.26	0.70	8.14	0.02
Google+	0.01	0.03	**88.01**	0.48	11.44	0.02	0.00	0.02
Instagram	0.40	0.00	0.00	**98.80**	0.80	0.00	0.00	0.00
Original	0.00	0.00	0.00	0.00	**99.01**	0.99	0.00	0.00
Telegram	0.01	0.00	0.00	0.00	1.12	**98.87**	0.00	0.00
Twitter	0.11	2.00	0.00	0.00	1.51	0.11	**96.27**	0.00
WhatsApp	0.00	0.12	0.00	0.03	0.72	0.00	0.00	**99.13**

This behaviour is due to the jointly interaction of features when *FusionNET* is employed. Similar improvement in classification is also obtained on the *UCID social* dataset, as already observed in subsection IV-C.1. It is the noise residual feature that helps to capture the distinctive resizing properties of this particular social network when images of other SNs are recompressed with the same *QF* used by *Facebook*.

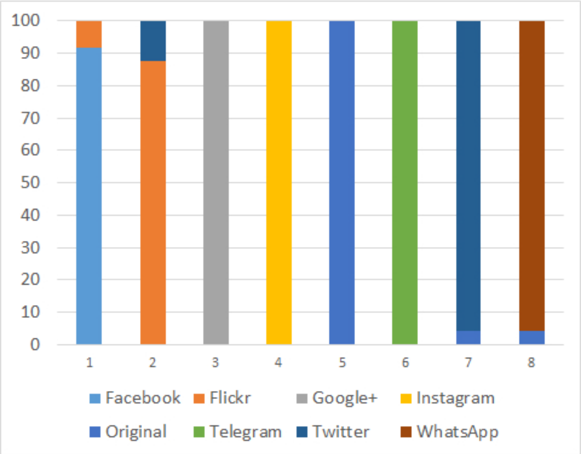


FIGURE 7. **Image level accuracy on *IPLAB* dataset with *FusionNET*.**

At the image level, *FusionNET* yields an average accuracy of 96.35% on the correct classes, evidenced in Figure 7. Such a value is obtained by averaging all the values related to the exact class on each column of Figure 7. It is worth noting that most of the images are well classified except for *Facebook* and *Flickr,* exchanged with *Flickr* and *Twitter* respectively with an error around 10% (see bars 1 and 2). *Twitter* and *WhatsApp* images are erroneously misplaced (about 5%) with *Original* images (see bars 7 and 8). Apart of these, all the other cases present a correct classification rate of 100% (see bars 3, 4, 5 and 6).

4) VISION

This subsection is concerned with the validation of the proposed *FusionNET* on the *VISION* subset. First of all, since *VISION* subset is composed of 10 different cameras or tablets, we propose a different view of the classification results in order to visually understand how social networks can be correctly separated given that the images of the same SN/IMA classes are taken with different cameras.

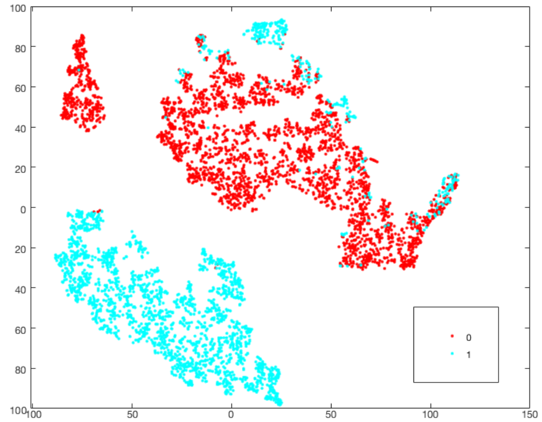


FIGURE 8. **Results obtained with *t-SNE* on *VISION* dataset demonstrating the separation between *Facebook* (class 0, red) and *WhatsApp* (class 1, cyan).**

TABLE 6. **Classification accuracy at the patch level on the *VISION* dataset with images of the *Facebook*, *WhatApp*, and *Original* classes involved.**

CNN	Fb	Wa	Orig
1D-CNN	97.76	98.61	99.99
2D-CNN	97.86	97.97	99.79
FusionNET	99.97	98.65	99.81

In Figure 8, the separation between *WhatsApp* and *Facebook* classes is depicted. This plot is obtained by using *t-Distributed Stochastic Neighbor Embedding (t-SNE)* [29]. The split of the *VISION* dataset into classes is based on the property extracted with *FusionNET* at the fully connected layer with 512 units (see Figure 3). The separation is very well evidenced, demonstrating *FusionNET*'s capability of distinguishing the two platforms. This suggests that the information *FusionNET* works on is SN-specific and is and independent of other factors, therefore it is well suited to solve the social network identification problem.

Finally, the *VISION* dataset is used to measure the performance of the proposed method in comparison to *1D-CNN* and *2D-CNN*. The images not shared on any social networks have been added in this analysis. So images of three classes, namely *Facebook*, *WhatsApp* and *Original* of the *VISION* dataset, are used in the experiments.

It can be observed that the results are very good for all three techniques although the proposed *FusionNET* appears to perform slightly better on average. The performance gain of *FusionNET* over the other two techniques is more prominent (over 2%) for *Facebook* images.

V. CONCLUSION

In this paper, the idea of integrating two different CNNs into a framework based on fused features is proposed. Such a *FusionNET* is used for classifying the source social network of downloaded images. The information fed to the proposed *FusionNET* for learning discriminative features are the histogram of the DCT coefficients and the noise residual of the images in question. Performances have been compared with established methods, namely *1D-CNN* and *2D-CNN*, against different image datasets across different types of social platforms. The observed results have validated the feasibility of fusing multiple features learned by *FusionNET* in differentiating source social platforms including not only social networks but also instant messaging apps. Based on the encouraging outcomes, a natural extension of this work is to investigate into the source SN identification in light of multiple sharing (i.e. when images have been uploaded onto and downloaded from more than two different SNs) and the exploitation of other distinctive traces superimposed on images by SNs and IMAs.

ACKNOWLEDGMENT

The authors would like to acknowledge NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

REFERENCES

[1] F. D. O. Costa, M. A. Oikawa, Z. Dias, S. Goldenstein, and A. R. de Rocha, "Image phylogeny forests reconstruction," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 10, pp. 1533–1546, Oct. 2014.

[2] A. Bharati *et al.*, "U-phylogeny: Undirected provenance graph construction in the wild," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 1517–1521.

[3] M. Chen, J. Fridrich, M. Goljan, and J. Lukás, "Determining image origin and integrity using sensor noise," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 1, pp. 74–90, Mar. 2008.

[4] X. Lin and C.-T. Li, "Image provenance inference through content-based device fingerprint analysis, *Information Security: Foundations, Technologies and Applications* (Security). Edison, NJ, USA: IET, 2018, pp. 279–310. [Online]. Available: https://digital-library.theiet.org/content/books/10.1049/pbse001e_ch12. doi: 10.1049/PBSE001E_ch12.

[5] X. Lin and C.-T. Li, "Large-scale image clustering based on camera fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 4, pp. 793–808, Apr. 2017.

[6] O. Giudice, A. Paratore, M. Moltisanti, and S. Battiato, "A classification engine for image ballistics of social data," in *Image Analysis and Processing—ICIAP*, S. Battiato, G. Gallo, R. Schettini, and F. Stanco, Eds. Cham, Switzerland: Springer, 2017, pp. 625–636.

[7] R. Caldelli, R. Becarelli, and I. Amerini, "Image origin classification based on social network provenance," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 6, pp. 1299–1308, Jun. 2017.

[8] R. Caldelli, I. Amerini, and C. T. Li, "PRNU-based image classification of origin social network with CNN," in *Proc. EUSIPCO*, Sep. 2018, pp. 1357–1361.

[9] Q.-T. Phan, C. Pasquini, G. Boato, and F. G. B. De Natale, "Identifying image provenance: An analysis of mobile instant messaging apps," in *Proc. MMSP*, Aug. 2018, pp. 1–6.

[10] M. Moltisanti, A. Paratore, S. Battiato, and L. Saravo, "Image manipulation on Facebook for forensics evidence," in *Image Analysis and Processing—ICIAP*, V. Murino and E. Puppo, Eds. Cham, Switzerland: Springer, 2015, pp. 506–517.

[11] I. Amerini, T. Uricchio, and R. Caldelli, "Tracing images back to their social network of origin: A CNN-based approach," in *Proc. IEEE Workshop Inf. Forensics Secur. (WIFS)*, Dec. 2017, pp. 1–6.

[12] Q. Wang and R. Zhang, "Double JPEG compression forensics based on a convolutional neural network," *EURASIP J. Inf. Secur.*, vol. 2016, no. 1, p. 23, 2016. doi: 10.1186/s13635-016-0047-y.

[13] M. Barni *et al.*, "Aligned and non-aligned double JPEG detection using convolutional neural networks," *J. Vis. Commun. Image Represent.*, vol. 49, pp. 153–163, Nov. 2017. doi: 10.1016/j.jvcir.2017.09.003.

[14] I. Amerini, T. Uricchio, L. Ballan, and R. Caldelli, "Localization of JPEG double compression through multi-domain convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 1865–1871.

[15] D. Cozzolino, G. Poggi, and L. Verdoliva, "Recasting residual-based local descriptors as convolutional neural networks: An application to image forgery detection," in *Proc. 5th ACM Workshop Inf. Hiding Multimedia Secur. (MMSec)*, New York, NY, USA, 2017, pp. 159–164. doi: 10.1145/3082031.3083247.

[16] M. Boroumand and J. Fridrich, "Deep Learning for Detecting Processing History of Images," *Electronic Imaging*, vol. 2018, no. 7, pp. 213-1–213-9, 2018. [Online]. Available: https://www.ingentaconnect.com/content/ist/ei/2018/00002018/00000007/art00011. doi: 10.2352/ISSN.2470-1173.2018.07.MWSF-213.

[17] A. Tuama, F. Comby, and M. Chaumont, "Camera model identification with the use of deep convolutional neural networks," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Dec. 2016, pp. 1–6.

[18] L. Bondi, L. Baroffio, D. Güera, P. Bestagini, E. J. Delp, and S. Tubaro, "First steps toward camera model identification with convolutional neural networks," *IEEE Signal Process. Lett.*, vol. 24, no. 3, pp. 259–263, Mar. 2017.

[19] D. Freire-Obregón, F. Narducci, S. Barra, and M. Castrillón-Santana, "Deep learning for source camera identification on mobile devices," *Pattern Recognition Letters*, to be published. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167865518300059. doi: 10.1016/j.patrec.2018.01.005.

[20] D. Güera, Y. Wang, L. Bondi, P. Bestagini, S. Tubaro, and E. J. Delp, "A counter-forensic method for cnn-based camera model identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 1840–1847.

[21] B. Tondi, "Pixel-domain adversarial examples against CNN-based manipulation detectors," *Electron. Lett.*, vol. 54, no. 21, pp. 1220–1222, Oct. 2018. [Online]. Available: http://digital-library.theiet.org/content/journals/10.1049/el.2018.6469

[22] D. Shullani, M. Fontani, M. Iuliani, O. A. Shaya, and A. Piva, "VISION: A video and image dataset for source identification," *EURASIP J. Inf. Secur.*, vol. 2017, no. 1, p. 15, Oct. 2017. doi: 10.1186/s13635-017-0067-2.

[23] I. Amerini, R. Caldelli, V. Cappellini, F. Picchioni, and A. Piva, "Analysis of denoising filters for photo response non uniformity noise extraction in source camera identification," in *Proc. 16th Int. Conf. Digit. Signal Process.*, Jul. 2009, pp. 1–7.

[24] M. K. Mihcak, I. Kozintsev, and K. Ramchandran, "Spatially adaptive statistical modeling of wavelet image coefficients and its application to denoising," in *Proc. IEEE ICASSP*, Phoenix, AZ, USA, Mar. 1999, pp. 3253–3256.

[25] M. D. Zeiler, "ADADELTA: An adaptive learning rate method," *CoRR*, vol. abs/1212.5701, 2012. [Online]. Available: http://arxiv.org/abs/1212.5701

[26] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, no. 12, pp. 3371–3408, Dec. 2010.

[27] G. Schaefer and M. Stich, "UCID: An uncompressed color image database," vol. 5307, 2003. doi: 10.1117/12.525375.

[28] J. Yang, G. Zhu, and Y. Shi, "Analyzing the effect of JPEG compression on local variance of image intensity," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2647–2656, Jun. 2016. doi: 10.1109/TIP.2016.2553521.

[29] L. van der Maaten, "Accelerating t-SNE using tree-based algorithms," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 3221–3245, Oct. 2014. [Online]. Available: http://dl.acm.org/citation.cfm?id=2627435.2697068

IRENE AMERINI (M'17) received the Laurea degree in computer engineering and the Ph.D. degree in computer engineering, multimedia, and telecommunication from the University of Florence, Italy, in 2006 and 2010, respectively, where she is currently a Postdoctoral Researcher with the Media Integration and Communication Center. She has received the Italian Habilitation for Associate Professor in telecommunications and computer science. She was a Visiting Scholar with Binghamton University, NY, USA, in 2010. She has been a Visiting Research Fellow of the School of Computing and Mathematics, Charles Sturt University, Australia, since 2018, offered by the Australian Government-Department of Education and Training, through the Endeavour Scholarship & Fellowship Program. Her main research activities include digital image processing, multimedia content security technologies, secure media, and multimedia forensics. She is a member of the IEEE Information Forensics and Security Technical Committee and EURASIP SAT Biometrics, Data Forensics, and Security. She is an Associate Editor of the IEEE Access and a Guest Editor of several international journals.

CHANG-TSUN LI (SM'12) received the B.Eng. degree in electrical engineering from National Defence University, Taiwan, in 1987, the M.Sc. degree in computer science from the U.S. Naval Postgraduate School, USA, in 1992, and the Ph.D. degree in computer science from the University of Warwick, U.K., in 1998. He was an Associate Professor with the Department of Electrical Engineering, NDU, from 1998 to 2002, and a Visiting Professor with the Department of Computer Science, U.S. Naval Postgraduate School, in 2001. He was a Professor with the Department of Computer Science, University of Warwick, U.K., until 2016. He is currently a Professor with the School of Computing and Mathematics, Charles Sturt University, Australia, where he is currently leading the Data Science Research Unit. His research interests include multimedia forensics and security, biometrics, data mining, machine learning, data analytics, computer vision, image processing, pattern recognition, bioinformatics, and content-based image retrieval. The outcomes of his multimedia forensics and machine learning research have been translated into award-winning commercial products protected by a series of international patents and used by a number of police forces and courts of law around the world. He has involved in the organisation of many international conferences and workshops, and he has also served as a member of the international program committees for several international conferences. He is a member of the IEEE Information Forensics and Security Technical Committee. He is also actively contributing keynote speeches and talks at various international events. He is currently the Vice Chair of Technical Committee 6 and Computational Forensics of International Association of Pattern Recognition. He is currently an Associate Editor of the IEEE Access, the *EURASIP Journal of Image and Video Processing*, and *IET Biometrics*.

ROBERTO CALDELLI (M'11–SM'17) 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. Since 2018, he has an Associate Professor of cybersecurity with Mercatorum University, Rome. In 2014, he joined the Media Integration and Communication Center, National Inter-University Consortium for Telecommunications (CNIT), as a permanent Researcher. He holds two patents in the fields of content security and multimedia interaction. His main research activities, witnessed by several publications, include digital image processing, interactive television, image and video digital watermarking, and multimedia forensics. Since 2016, he has been a member of the IEEE Information Forensics and Security Technical Committee, Signal Processing Society. He has received the Italian Habilitation for Associate Professor in telecommunications and computer science. He is an Associate Editor of main international journals, such as the IEEE Signal Processing Letters and *Signal Processing: Image Communication*.