Multimed Tools Appl (2019) 78:26851–26865
DOI 10.1007/s11042-016-4251-z

Smooth filtering identification based on convolutional neural networks

Anan Liu[1] · Zhengyu Zhao[1] · Chengqian Zhang[2] · Yuting Su[1]

Received: 31 August 2016 / Revised: 26 November 2016 / Accepted: 7 December 2016 /
Published online: 30 December 2016

Abstract The increasing prevalence of digital technology brings great convenience to human life, while also shows us the problems and challenges. Relying on easy-to-use image editing tools, some malicious manipulations, such as image forgery, have already threatened the authenticity of information, especially the electronic evidence in the crimes. As a result, digital forensics attracts more and more attention of researchers. Since some general post-operations, like widely used smooth filtering, can affect the reliability of forensic methods in various ways, it is also significant to detect them. Furthermore, the determination of detailed filtering parameters assists to recover the tampering history of an image. To deal with this problem, we propose a new approach based on convolutional neural networks (CNNs). Through adding a transform layer, obtained distinguishable frequency-domain features are put into a conventional CNN model, to identify the template parameters of various types of spatial smooth filtering operations, such as average, Gaussian and median filtering.

This work was supported in part by the National Natural Science Foundation of China (61572356, 61472275, 61303208), the Tianjin Research Program of Application Foundation and Advanced Technology (15JCYBJC16200), a grant from the China Scholarship Council (201506255073), and a grant from the Elite Scholar Program of Tianjin University (2014XRG-0046).

✉ Yuting Su
ytsu@tju.edu.cn

Anan Liu
liuanan@tju.edu.cn

Zhengyu Zhao
zyzhao2014@tju.edu.cn

Chengqian Zhang
zhangcqj@tju.edu.cn

[1] School of Electronic Information Engineering, Tianjin University, Tianjin, China

[2] School of Electrical Engineering and Information, Southwest Petroleum University, Chengdu, 610500, China

Experimental results on a composite database show that putting the images directly into the conventional CNN model without transformation can not work well, and our method achieves better performance than some other applicable related methods, especially in the scenarios of small size and JPEG compression.

Keywords Digital forensics · Spatial smooth filtering · Convolutional neural network · Deep learning · Discrete Fourier transform · JPEG compression

1 Introduction

With the rapid development of digital technology and the popularity of digital devices, it becomes more convenient to transmit or store digital images. Moreover, efficient image editing softwares are not only available to professional researchers, but also common people. As a result, many computer vision-related studies on various general tasks such as image segmentation [37, 38, 41, 42], image cropping [23, 40], image categorization [21, 36, 43, 45, 47], and object recognition [39, 46], have been carried out. Besides, some other issues for specific applications have also been discussed, for example, rare category exploration in medical diagnoses and financial security [22, 23]. Instead of these positive uses, some people with ulterior motivations implement malicious manipulations, such as image forgery, to tamper the authentic digital information, especially the electronic evidence in the crimes. Therefore, the authenticity and integrity of images can not be taken for granted anymore and many forensics-related challenges have arisen accordingly.

Generally, lots of forensic methods concentrate on intentional content forgeries of the image, which mainly include copy-move forgery [7] and image splicing [1]. However, it is also beneficial to explore more about the manipulating history of an image, including plausible content-preserving operations, such as smooth filtering, compression [24], retargeting [44] and contrast enhancement [30]. Among them, smooth filtering is applied widely for blurring and denoising, as well as a post-processing technology used to decrease the reliability of forensic tools. For example, Most copy-move and splicing forgeries employ smooth filtering to reduce the discontinuity between the forged regions and the rest of the image, for the purpose of appearing more realistic. Also some researchers try to diminish subtle traces left by prior manipulations such as resampling [17] and JPEG compression [31], with the help of smooth filtering. As a result, implementing identification of smooth filtering, especially the detailed filtering parameters, can yield useful information for forensic analysis by exposing the history of manipulations.

In general, smooth filtering operations fall into two categories. One is linear, mainly including average and Gaussian filtering, and the other is nonlinear, i.e., median filtering. Considering that the spatial template filtering is the most representative method of smooth filtering and have been studied in almost all the related researches, identification of types and further detailed parameters (the window size to average and median filtering, and the window size and σ to Gaussian filtering) will be very significant for smooth filtering forensics. Notably, with the popularity of the internet and mobile terminals, many images are usually transmitted or stored in a low-quality with the form of small size or JPEG compression, therefore, such identification in this case is very practical and challenging.

Many related works confined to median filtering forensics [3–5, 11, 15, 18, 35, 49] have been carried out. Yuan [35] proposed a combined feature called median filtering forensics (MFF) for median filtering detection in the scenarios of JPEG compression and small size.

Kang et al. [15] utilized a feature from the autoregressive (AR) model of median filter residual (MFR) to improve detection performance for nearly saturated images. Chen et al. [5] made an attempt to adopt a deep learning method based on the same MFR and achieved significant improvement with a high-dimensional feature. We also proposed a novel low-dimensional feature vector coined (annular accumulated points) AAP to realize the detection of median filtering with a time-saving process in the previous work. All these methods achieved good performance on differentiating median filtering images from original images or images which have undergone other types of manipulations. However, these works did not involve the distinguishment between each pair of types from, such as original, average filtered, Gaussian filtered and median filtered images. Authors in [15] claimed that once a forensic investigator has identified that an image has been median filtered, they may wish to determine the window size used during median filtering. So they tried differentiating 3×3 median filtering from 5×5 and obtained good results. It also reflects the importance of identifying the template parameters of filters.

Most existing approaches [4, 11, 15, 29, 35, 49] manually extract reliable features, and then feed them into a classifier like the support vector machine (SVM), which has been trained with lots of labeled images, for detection. Such manual settings may result in imprecise parameters optimization, which could degrade the performance. Instead, we try to adopt deep learning thoughts to accomplish features learning and classification automatically with iterative parameters update. Deep learning networks, such as deep autoencoders [33], deep Boltzmann Machines [26] and Convolutional Neural Networks (CNNs) [20], have attracted increasing attention due to their excellent performances in artificial intelligence. Among them, CNNs-related methods show outstanding effectiveness in image classification and character recognition, and some typical models could be referenced.

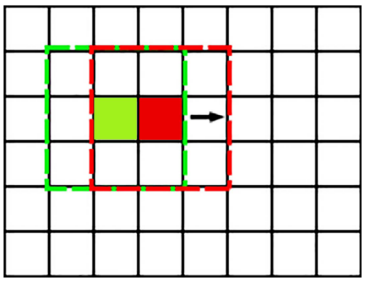
In this paper, we propose a modified framework based on a conventional CNN model [20]. By adding a transform layer in front of the conventional model, which is similar to the schemes in [5, 34], we reveal the distinctions between different types of smooth filtering operations based on the frequency-domain characteristics, which are usually more conspicuous than those in the spatial domain. Experimental results verify the effectiveness of our method and show it outperforms the state-of-the-art works, which have been proposed for median filtering detection and could be applicable in smooth filtering forensics. Besides, our method remains useful in the more practical and challenging cases of small size and JPEG compression.

2 Proposed method

2.1 Spatial smooth filtering

Spatial smooth filtering are widely applied in digital image processing. As shown in Fig. 1a, by moving a $h \times h$ (h is odd) square window throughout the given $M \times N$ image, each output in the position of the green pixel, is obtained based on its surrounding pixels in the dashed green window, and the red one denotes the repeated step. For linear smooth filtering, such as average and Gaussian filtering, it could be interpreted as a convolution-based operation, formulated as,

$$f(x, y) = \sum_{i,j} w_{i,j} \cdot g(x, y) \tag{1}$$



(a)

$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$

(b)

0.02	0.08	0.02
0.08	0.60	0.08
0.02	0.08	0.02

(c)

Fig. 1 **a** shows the smooth filtering operation in the spatial domain; the template of the **b** 3×3 average filter and **c** 3×3 Gaussian filter with $\sigma = 0.5$, respectively

where $w_{i,j}$ denotes the weights in the template, $(i, j) \in \{-\frac{h-1}{2}, ..., \frac{h-1}{2}\}$ and $(x, y) \in \{1, 2..., M\} \times \{1, 2..., N\}$. Figure 1b and c show the templates of 3×3 average filtering and 3×3 Gaussian filtering with $\sigma = 0.5$, respectively.

And for nonlinear median filtering, the expression is as follows,

$$f(x, y) = median\big(g(x + i, y + j)\big) \tag{2}$$

2.2 Frequency-domain response

Initially, we put the images themselves directly into the conventional CNN model, but it could not work well (detailed results will be presented in Section 3.2.2). Because there are few perceptible differences to be captured between the original and different filtered versions of the image, as shown in Fig. 2, the analysis had better resort to discernible patterns by implement some transformation operations.

We will explore the patterns introduced by various types of filtering operations in the frequency domain. Fig. 3 shows frequency-domain response amplitudes from original, average filtered, Gaussian filtered and median filtered version of the same image as Fig. 2, respectively. For easy-observing, the low-frequency region is shifted to the center part. It can be observed that the frequency-domain figures from average and Gaussian filtered version exhibit distinct patterns, compared with the original version. Moreover, for the same type of filtering, varied template parameters produce different forms, since the smoothing degree varies. However, as a nonlinear smoothing operation, median filtering presents fewer discernible patterns, and its frequency response depends on the properties of the input image. As a result, we introduce the empirical frequency response (EFR) [14] to further

(a) (b) (c) (d)

Fig. 2 An original image (**a**), and its (**b**) 5×5 average filtered version, 5×5 Gaussian filtered ($\sigma = 1$) version and (**c**) 5×5 median filtered version

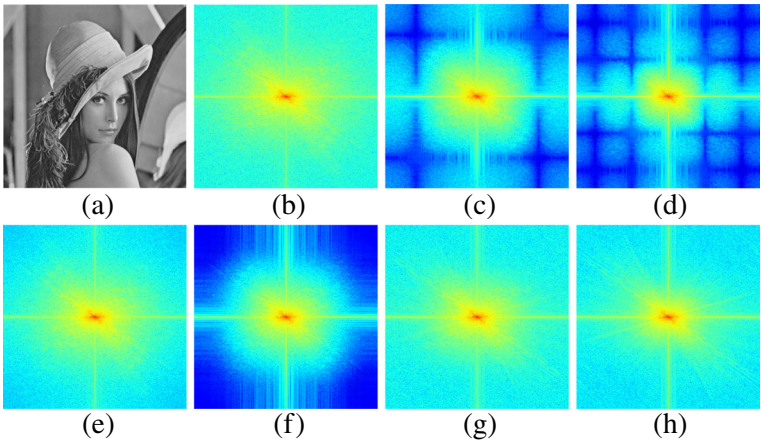


Fig. 3 An original image (**a**), and its frequency-domain response amplitudes of different versions: **b** original **c** 3×3 average filtered, **d** 5×5 average filtered, **e** 3×3 Gaussian filtered ($\sigma = 0.5$), **f** 5×5 Gaussian filtered ($\sigma = 1$), **g** 3×3 median filtered, and **h** 5×5 median filtered

reveal the characteristics in the frequency domain of different types of filtered images for identification.

The EFR of median filtering is defined as,

$$EFR(\omega) = O(\omega)/I(\omega) \tag{3}$$

where $I(\omega)$ denotes the spectrum of an original image and $O(\omega)$ is the spectrum of its median filtered version.

The EFR could show the changes before and after filtering. Figure 4 presents an example of respective EFRs of median, average and Gaussian filtering, where each curve is obtained by computing one-dimensional EFR of each row and then average the EFRs from all the rows in an image; The results of median filtering agree with the simulation experiments performed in [12]. It can be also observed that the EFR curve of average filtering appears oscillatory trailing, and the curve of Gaussian filtering decreases as the frequency increases in general. As is known, the spectral responses of moving median filtering and moving average filtering for frequencies of $\omega \leq 2\pi/h$ (where h denotes the size of the filter template) are highly similar. However, in the $\omega > 2\pi/h$ region, the curve is very irregular because of

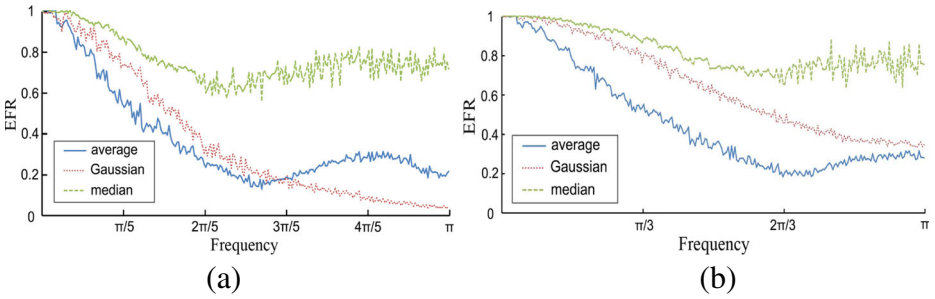


Fig. 4 EFR curves from three different types of **a** 5×5 and **b** 3×3 filtered versions

interferences from different frequencies [32]; Some components in this region are retained while others are weakened.

Above analyses indicate that median filtering could also bring detectable distinct characteristics into the frequency domain.

2.3 Modified CNN architecture

Hubel and Wiesel [13] proposed the receptive fields through the investigations on the cat's visual cortex in 1962. About twenty years later, the Japanese researcher Fukushima [8] introduced this concept to construct a neural network model called neocognitron, and it is generally regarded as the origin of CNNs. Different from a conventional fully connected network, CNN uses the concept of local receptive fields to achieve shift and deformation invariance, and reduce the number of parameters in the network for better generalization performance and less computational complexity with shared weights.

A typical CNN framework consists of three types of layers: convolutional layers, pooling layers and fully connected layers. Based on these typical types of layers, we propose a modified CNN model called T-CNN by adding a transform layer in advance.

2.3.1 Transformation layer

In order to capture discernible frequency-domain patterns shown above, we add this layer in front of a conventional CNN framework. It is composed of two units: DFT and log-scale transformation.

In case of the digital image, a 2-D DFT is performed to obtain the spectrum $F(u, v)$, defined as,

$$F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \cdot \exp(-\mathrm{j} \cdot 2\pi(ux/M + vy/N)) \tag{4}$$

where $f(x, y)$ denotes the pixel value at point (x, y) in the $M \times N$ image, $u \in \{1, 2, ..., M\}$ and $v \in \{1, 2, ..., N\}$.

For a real image, the spectral values in the low-frequency region are much bigger than other values in the medium- and high-frequency region, so the spectrum is firstly converted into a log-scale-form frequency-domain figure $F'(u, v)$, to ensure the values in the same order of magnitude for further analysis, as follows,

$$F'(u, v) = \log_{10}(|F(u, v)| + 1) \tag{5}$$

where $|F(u, v)|$ denotes the spectral magnitudes and "+1" ensures non-negative outputs.

Finally, the images are transformed into frequency-domain figures after above process.

2.3.2 Convolutional layer

At the convolutional layers, each neuron is connected to only a small region of the input to perceive local correlation. Every entry in the output can thus be interpreted as an output of a neuron that perceives a subregion in the input and shares weights with the same kernel (or filter). The convolution operation can be denoted as,

$$x_j^l = \sum_{i \in M_i} x_i^{l-1} * k_{i \to j}^l + b_j^l \tag{6}$$

where $*$ denotes the convolution operator, x_i^{l-1} is the i-th of all the output maps in layer $l-1$. The convolutional kernel $k_{i\to j}^{l}$, whose weights could be updated by training, is used to generate output maps in layer l from layer $l-1$, b_j^l is the trainable bias of the j-th output map in layer l.

After convolution, the outputs can be represented as feature maps with specific feature detectors. Next, a nonlinear operation is performed to increase the nonlinear properties of the decision function and the overall network without affecting the receptive fields.

2.3.3 Pooling layer

Following pooling layer is applied for a downsampling operation. There are several nonlinear functions to implement pooling, among which max pooling is the most common. It divides each output map from the previous layer into a set of non-overlapping subregions and outputs their maximums. Through this operation, the feature maps with the smaller size reduce the amount of parameters and computation in the network, and hence avoid overfitting [28] to some extent.

2.3.4 Fully connected layer

Finally, after several repeated sections of alternating convolutional and max pooling layers, the higher level representation will be acquired. Generally, a certain number of consecutive fully connected layers, followed by a softmax loss layer, is used for classification. Their neurons, which is fully connected to all activations in the previous layer, will output the probability of a sample classified into a specific class through the softmax function. The trainable parameters in the network will be upgraded automatically by the backward error propagation procedure over and over [20]. This is definitely why the CNN-based methods mostly outperform other manually extracted features.

2.4 Parameter settings

The framework of the proposed T-CNN model is shown in Fig. 5. We will describe the detailed settings in this part. The architecture mainly contains one transform layer, two convolutional layers, two pooling layers and two fully connected layers. At first, the additional

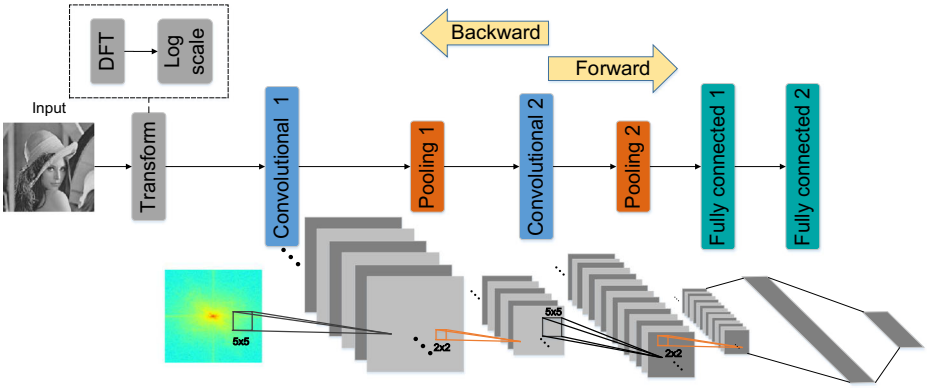


Fig. 5 The framework of the proposed T-CNN

transform layer before the convolution layers, performs DFT and log-scale transformation on the input images so as to generate frequency-domain figures. In this work, the data are firstly resized to 64 × 64 for saving learning time of the model. After transformation, the first convolutional layer convolves previous outputs with 20 kernels of size 5 × 5 (stride=1) and generate 20 corresponding feature maps with the size of 60 × 60. It is followed by an max pooling layer with filters of size 2 × 2 (stride=2), to decrease the size of feature maps to 30 × 30. Similarly, after following convolutional layer with 50 kernels of size 5 × 5 (stride=1), and the same pooling layer as before, 50 feature maps with the size of 13 × 13 are obtained. Finally, the first fully connected layer with 500 neurons converts previous outputs into a vector with the size of 1 × 500, and the neurons in the last fully connected layer must not be fewer than the classes. Its output is fed into a softmax loss layer for classification.

The Rectified Linear Units (ReLUs) function $f(x) = \max(x, 0)$ is used to activate the outputs of convolutional layers and the first fully connected layer. Compared with a common saturating activation function like sigmoid and tanh, the ReLUs has been argued to be more biologically plausible [10], and it can accelerate the convergence for training [25]. Meanwhile, a technique "dropout" [19] is used to reduce overfitting by randomly omitting half of the feature detectors on each training case.

3 Experiments

In this section, we will validate the effectiveness of our method and compare its performance with some other applicable outstanding methods, which have been proposed for median filtering forensics.

3.1 Experimental setup

We carried out the experiments on a composite database containing 15000 images from the following three image databases:

- 1338 images from the UCID [27]. This database contains 1338 uncompressed RGB images with a resolution of 512 × 384. Many of these images contain significant regions of mostly smooth patches and several images are defocused, which presents additional challenges.
- 3662 images from the Dresden Image Database (DID) [9]. It contains more than 14000 images of various indoor and outdoor scenes acquired from 73 different digital cameras.
- 10000 images from the BOSSbase [2]. It contains 10000 never-compressed grayscale images with the size of 512 × 512.

These popular databases have been used in many related works, such as [3, 5, 11, 15, 35, 49]. Before further processing, images were converted into 8-bit grayscale.

Theoretically, we could implement template filtering operations on images with any parameter. But in practice, only a limited number of typical parameters are applied. For example, the authors in [3–5, 11, 15, 18, 35, 49] carried out related researches with widely used 3 × 3, 5 × 5 and 7 × 7 median filters. Similarly, for the average filter, larger window sizes will cause excessively blurred appearance and smaller could not achieve the desired effectiveness for smoothing to interfere with the forensic methods. And for the Gaussian filter, much bigger or smaller σ will also cause unsatisfactory outcomes, and there is generally a positive correlation between the selected σ and corresponding window size. According to such actual conditions, in this paper, we discuss three different typical templates for each of

these three filters, to arrange the experiments and as a result, obtain ten datasets in total as follows.

The composite database without any manipulation is regarded as the original dataset D^{ORI}. Processing the D^{ORI} using 3×3, 5×5 and 7×7 median filters, 3×3, 5×5 and 7×7 average filters, and 3×3 ($\sigma = 0.5$), 5×5 ($\sigma = 1$) and 7×7 ($\sigma = 1.5$) Gaussian low-pass filters to obtain the datasets D^{MED3}, D^{MED5} ,D^{MED7}, D^{AVE3}, D^{AVE5}, D^{AVE7}, D^{GAU3}, D^{GAU5} and D^{GAU7}, respectively. All the images in these ten datasets will be fed in the proposed modified CNN model. Randomly selected 70 % of each dataset is designated as the training set, while the complement 30 % is designated as the testing set. Totally, the training set contains 105000 images from ten different classes, while the testing set contains the remaining 45000 images.

Similar to other related methods [4, 5, 15, 35], we also evaluate the performance in terms of accuracy,

$$\mathrm{Acc} = \frac{N_c}{N_t}, \tag{7}$$

where N_c and N_t denote the number of correctly classified and total testing samples, respectively.

3.2 Experimental results

To evaluate the performance of our method, we present the classification results of above ten classes with a confusion matrix, as shown in Table 1. The diagonal elements denote the accuracy of each class and the rest show the error rates. It can be observed that the proposed method effectively discriminate ten classes from one another. Fig. 6 shows the changes of the testing loss and accuracy in the iterative process. We can see the testing loss tends to be convergent after about 20000 iterations, and the accuracy is maintained a high level.

Several state-of-the-art methods, such as the MFF, MFR+AR, MFR+CNN and AAP, have been proposed for the most challenging median filtering detection. Although they were not introduced in general smoothing identification involved in this paper, they could also be migrated to this application, considering that they also explored the statistical differences between different operations, including median, average and Gaussian filtering. Table 2 presents the comparing results of these methods. We can see that our method T-CNN outperforms the manually-extracted-features-based methods MFF, MFR+AR and AAP, and

Table 1 The confusion matrix of T-CNN between ten classes on the composite database

%	ORI	MED3	MED5	MED7	AVE3	AVE5	AVE7	GAU3	GAU5	GAU7
ORI	96.77	0.38	0.20	0	0.61	0.69	0.65	0.48	0.22	0
MED3	0.40	97.35	0.64	0.48	0.18	0	0	0.14	0.44	0.36
MED5	0.34	0.60	97.55	0.48	0.38	0	0	0.08	0.22	0.36
MED7	0.23	0.24	0.34	97.83	0.28	0.32	0.34	0.10	0.18	0.16
AVE3	0.24	0.62	0.16	0	98.52	0	0.08	0.32	0	0.06
AVE5	0.14	0.57	0.18	0.02	0	98.96	0.02	0	0.08	0.04
AVE7	0.16	0.32	0	0.14	0.06	0.06	98.84	0.04	0.04	0.32
GAU3	0.59	0.20	0.04	0.24	0.20	0.55	0.18	97.58	0.20	0.22
GAU5	0.64	0.06	0.24	0.10	0.48	0.12	0.48	0	97.84	0.04
GAU7	0.32	0.18	0.26	0.06	0.44	0.24	0.16	0.02	0.14	98.17

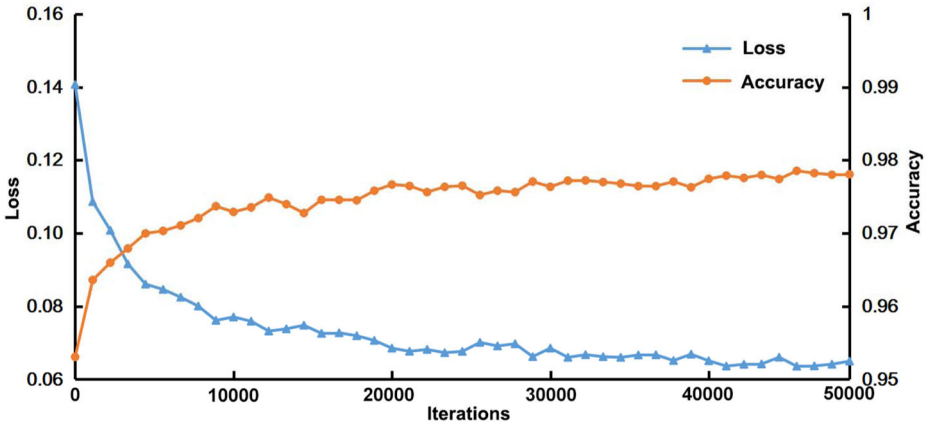


Fig. 6 Changing loss and accuracy in the iterative process, where the left vertical axis indicates the loss and the right indicates the accuracy

also the deep learning-based MFR+CNN, which failed perhaps because the MFR features extracted from different types of filtered images confuse the CNN model, since they present fewer perceptible differences than those in frequency domain. As expected, directly putting images into the conventional model achieve a very low overall accuracy 21.18 %.

3.2.1 Generic Features

Sometimes, effective features are learned not only for a specific task such as the template parameters identification in this paper. As a result, it is essential to separate the feature extraction and the classification to get a generic feature. Various pre-trained CNN models [6, 16, 48, 50] have been successfully used for extracting image features, which are normally the activations from some of the networks last few fully connected layers.

Following this, we use activations of the first fully connected layer as features called fc1. We qualitatively evaluate our learned fc1 to verify if it is a good generic feature by visualizing the features on the composite database. For each of the ten classes, we compute the average of 500-dimensional features from 4500 testing images to form a 1×500 vector representing this class. The comparison results in Fig. 7 indicate that our learned features show obvious differences between these ten different types. And also we can observe there are fewer differences between three median filtering classes with different parameters than those presented in average and Gaussian filtering. That is why the accuracies in Table 1 of the median filtering are slightly lower than those of average and Gaussian filtering. Moreover, it also reflects that median filtering identification is the most challenging task as shown in Fig. 3 above. Like some of baselines with manually extracted features, we also proceed to feed our fc1 features into the simple multi-class SVM and achieve a good overall accuracy 97.49 %.

Table 2 Overall accuracy of different methods on the composite database

Method	MFF	MFR + AR	MFR + CNN	AAP	T-CNN
Acc (%)	80.53	93.34	52.47	92.75	97.86

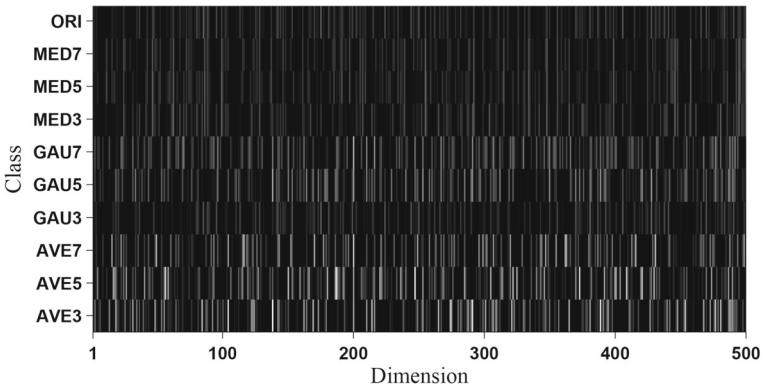


Fig. 7 Visualization of features from 10 different types of images. Each row denotes a 500-dimensional feature vector from one type, and for easy-observing, each row is duplicated 25 times continuously (totally 250 rows)

3.2.2 For low-resolution images

Most copy-move and splicing forgeries use smooth filtering to reduce the discontinuity between the forged regions and the rest of the image, for the purpose of appearing more realistic. In this case, local filtering operations in small sized regions will be employed. And in most practical situations of storage and transmission, images are saved in JPEG compressed format. As a result, it is necessary to further identify smooth filtering operations in the scenarios of small size and JPEG compression.

We firstly crop 64×64 center portion from each image in the composite database, and then compress cropped images with JPEG (quality factor(QF) $= 70$). Next, corresponding datasets and training-testing pair are generated following the steps in Section 3.1.

In this part, we compare our methods (including T-CNN and fc1+SVM) with above well performed MFF, MFR+AR and AAP. As known to all, the JPEG compression will affect the reliability of forensic methods because filter characteristics are suppressed by JPEG artifacts, and in small sized images, the number of captured features is reduced. Therefore, compared with the previous experimental results, the performances of all the methods degrade as shown in Table 3. However, our method is still effective with an accuracy over 80 %, and the method fc1+SVM could achieve better performance with 78.82 % than other baselines, which also put manually extracted features into a simple SVM for classification. That is because both our two methods exploit characteristics in the frequency domain, where the patterns might be more robust to resist the JPEG effects than those in the spatial domain, where the blocking effect caused by JPEG compression would affect the correlation between the pixels and some of their neighbors. Fig. 8 reports the detailed comparison results in the form of gray-scale confusion matrix. It can be observed that both the fc1+SVM and T-CNN present the most concentrated distribution in the diagonal of the matrix, while

Table 3 Accuracies of different methods for smooth filtering identification in small sized and JPEG compressed images

Method	MFF	MFR + AR	AAP	fc1 + SVM	T-CNN
Acc (%)	49.96	53.40	53.04	78.82	80.12

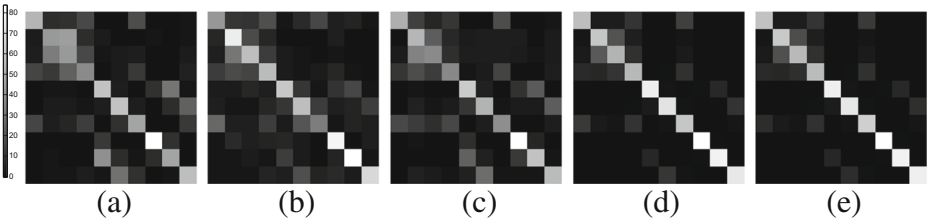


Fig. 8 The confusion matrixes of **a** MFF, **b** MFR+AR, **c** AAP, **d** fc1+SVM, and **e** T-CNN, respectively, for 64×64 JPEG compressed images; The ten filtering classes, in order form left to right and from top to bottom, are ORI, MED7, MED5, MED3, AVE7, AVE5, AVE3, GAU7, GAU5 and GAU3

other methods show higher values in some other components. This suggests our method could obtain the best performance in the practical scenarios.

4 Conclusion

In this paper, we proposed a novel method of smooth filtering forensics based on deep learning algorithms. By adding a transform layer in front of a conventional CNN model, we capture discernible patterns to identify the types and template parameters of the spatial smooth filtering operations. Experimental results supported by theoretical analysis have showed that our approach achieved outstanding performance, compared with several state-of-the-art methods, especially in the challenging cases of small size and JPEG compression. Different from most methods based on the spatial-domain analysis, we explore the characteristics of different filtering operations in the frequency domain, which makes our T-CNN method and the fc1+SVM method with extracted features from the first fully connected layer achieve better anti-interference performance to JPEG compression. Besides, the learned feature fc1 has also shown better performance than other manually extracted features of the baselines, which also apply simple SVMs to fulfil the classification task. As a result, it effectively separates the feature extraction and classification to get a generic feature, which will be employed in more general forensic tasks.

In the future, the proposed thoughts based on discernible frequency-domain patterns could be extended by involving more types of filtering operations and further other various manipulations in general forensic situations. Moreover, we will explore the distribution characteristics in the frequency domain more deeply, and try some statistical analysis to capture more respective low-dimensional features for efficient computation in the classification phase.

References

1. Bahrami K, Kot AC (2015) Image splicing localization based on blur type inconsistency. IEEE Trans Inf Forensics Secur 10(5):999–1009
2. Bas P, Filler T, Pevný T (2011) Break our steganographic system: The ins and outs of organizing BOSS. In: International Conference on Information Hiding, pp 59–70
3. Cao G, Zhao Y, Ni R, Yu L, Tian H (2010) Forensic detection of median filtering in digital images. In: 2010 IEEE International Conference on Multimedia and expo (ICME), IEEE pp 89–94
4. Chen C, Ni J, Huang J (2013) Blind detection of median filtering in digital images: A difference domain based approach . IEEE Trans Image Process 22:4699–4710

5. Chen J, Kang X, Liu Y, Wang ZJ (2015) Median filtering forensics based on convolutional neural networks. IEEE Signal Process Lett 22:1849–1853
6. Chen J, Song X, Nie L, Wang X, Zhang H, Chua TS (2016) Micro tells macro: Predicting the popularity of micro-videos via a transductive model. In: ACM Multimedia Conference, pp 898–907
7. Christlein V, Riess C, Jordan J, Riess C, Angelopoulou E (2012) An evaluation of popular copy-move forgery detection approaches. IEEE Trans Inf Forensics Secur 7(6):1841–1854
8. Fukushima K (1980) Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biol Cybern 36(4):193–202
9. Gloe T, Bohme R (2010) Dresden image database for benchmarking digital image forensics. In: Proceedings 2010 ACM symp. on appl. computing, Sierre, Switzerland, Mar. 22-26, pp 1584 –1590
10. Glorot X, Bordes A, Bengio Y (2010) Deep sparse rectifier neural networks. J Mach Learn Res 15
11. Gui X, Li X, Qi W, Yang B (2014) Blind median filtering detection based on histogram features. In: Asia-pacific signal and information processing Association, 2014 Annual Summit and Conference (APSIPA), IEEE, pp 1–4
12. Heygster G (1982) Rank filters in digital image processing. Comput Graphics Image Process 19:148–164
13. Hubel DH, Wiesel TN (1962) Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. J Physiol 160(1):106–154
14. Justusson B (1981) Median filtering: Statistical properties. Springer
15. Kang X, Stamm MC, Peng A, Liu KJR (2013) Robust median filtering forensics using an autoregressive model. IEEE Trans Inf Forensics Secur 8:1456–1468
16. Karayev S, Trentacoste M, Han H, Agarwala A, Darrell T, Hertzmann A, Winnemoeller H (2013) Recognizing image style. Computer Science
17. Kirchner M, Bohme R (2008) Hiding traces of resampling in digital images. IEEE Trans Inf Forensics Secur 3:582–592
18. Kirchner M, Fridrich J (2010) On detection of median filtering in digital images. In: Proceedings of SPIE - The International Society for Optical Engineering 7541:1–12
19. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. Adv Neural Inf Proces Syst 25(2):2012
20. Lecun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proc IEEE 86(11):2278–2324
21. Liu X, Song M, Tao D, Liu Z, Zhang L, Chen C, Bu J (2013) Semi-supervised node splitting for random forest construction. In: IEEE Conference on Computer Vision and Pattern Recognition, pp 492–499
22. Liu Z, Chiew K, He Q, Huang H, Huang B (2014) Prior-free rare category detection: More effective and efficient solutions. Expert Syst Appl Inter J 41(17):7691–7706
23. Liu Z, Chiew K, Zhang L, Zhang B, He Q, Zimmermann R (2016) Rare category exploration via wavelet analysis: Theory and applications. Expert Syst Appl 63:173–186
24. Luo W, Huang J, Qiu G (2010) JPEG error analysis and its applications to digital image forensics . IEEE Trans Inf Forensics Secur 5:480–491
25. Nair V, Hinton GE (2010) Rectified linear units improve restricted boltzmann machines vinod nair. In: International Conference on Machine Learning, pp 807–814
26. Salakhutdinov R, Hinton G (2009) Deep boltzmann machines. J Mach Learn Res 5(2):1967–2006
27. Schaefer G, Stich M (2003) UCID: an uncompressed color image database. In: Electronic Imaging 2004, International Society for Optics and Photonics, pp 472–480
28. Scherer D, Müller A, Behnke S (2010) Evaluation of pooling operations in convolutional architectures for object recognition. In: Artificial Neural Networks - ICANN 2010 - International Conference, Thessaloniki, Greece, September 15-18, 2010, Proceedings, pp 92–101
29. Song X, Ming ZY, Nie L, Zhao YL, Chua TS (2016) Volunteerism tendency prediction via harvesting multiple social networks. ACM Trans Inf Syst 34(2)
30. Stamm MC, Liu K (2010) Forensic detection of image manipulation using statistical intrinsic fingerprints. IEEE Trans Inf Forensics Secur 5:492–506
31. Stamm MC, Liu KJR (2011) Anti-forensics of digital image compression. IEEE Trans Inf Forensics Secur 6:1050–1065
32. Velleman PF (1980) Definition and comparison of robust nonlinear data smoothing algorithms. J Am Stat Assoc 75:609–615
33. Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol PA (2010) Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. J Mach Learn Res 11(6):3371–3408
34. Wang W, Yan Y, Zhang L, Hong R, Sebe N (2016) Collaborative sparse coding for multiview action recognition. IEEE Multimedia Magazine 23(4):80–87

35. Yuan HD (2011) Blind forensics of median filtering in digital images. IEEE Trans Inf Forensics Secur 6:1335–1345
36. Zhang L, Han Y, Yang Y, Song M, Yan S, Tian Q (2013a) Discovering discriminative graphlets for aerial image categories recognition. IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society 22(12):5071–5084
37. Zhang L, Song M, Liu Z, Liu X, Bu J, Chen C (2013b) Probabilistic graphlet cut: Exploiting spatial structure cue for weakly supervised image segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp 1908–1915
38. Zhang L, Song M, Zhao Q, Liu X, Bu J, Chen C (2013c) Probabilistic graphlet transfer for photo cropping. IEEE Trans Image Process 22(2):802–15
39. Zhang L, Gao Y, Hong C, Feng Y, Zhu J, Cai D (2014a) Feature correlation hypergraph: exploiting high-order potentials for multimodal recognition. IEEE Trans Cybernetics 44(8):1408–1419
40. Zhang L, Gao Y, Ji R, Xia Y, Dai Q, Li X (2014b) Actively learning human gaze shifting paths for semantics-aware photo cropping. IEEE Trans Image Process 23(5):2235–45
41. Zhang L, Gao Y, Xia Y, Lu K, Shen J, Ji R (2014c) Representative discovery of structure cues for weakly-supervised image segmentation. IEEE Trans Multimedia 16(2):470–479
42. Zhang L, Yang Y, Gao Y, Yu Y, Wang C, Li X (2014d) A probabilistic associative model for segmenting weakly-supervised images. IEEE Trans Image Process 23(9):4150–4159
43. Zhang L, Hong R, Gao Y, Ji R, Dai Q, Li X (2015a) Image categorization by learning a propagated graphlet path. IEEE Transactions on Neural Networks and Learning Systems 27(3):674–685
44. Zhang L, Li X, Nie L, Yan Y, Zimmermann R (2015b) Semantic photo retargeting under noisy image labels. ACM Trans Multimed Comput Commun Appl 12(3)
45. Zhang L, Wang M, Hong R, Yin BC (2015c) Large-scale aerial image categorization using a multitask topological codebook. IEEE Trans Cybernetics 46(1)
46. Zhang L, Li X, Nie L, Yang Y, Xia Y (2016a) Weakly supervised human fixations prediction. IEEE Trans Cybernetics 46(1):258–269
47. Zhang L, Yang Y, Wang M, Hong R (2016b) Detecting densely distributed graph patterns for fine-grained image categorization. IEEE Trans Image Process 25(2):553–565
48. Zhang N, Paluri M, Ranzato M, Darrell T, Bourdev L (2014e) PANDA: Pose aligned networks for deep attribute modeling. In: Computer Vision and Pattern Recognition, pp 1637–1644
49. Zhang Y, Li S, Wang S, Shi YQ (2014f) Revealing the traces of median filtering using high-order local ternary patterns. IEEE Signal Process Lett 21:275–279
50. Zhou B, Garcia AL, Xiao J, Torralba A, Oliva A (2014) Learning deep features for scene recognition using places database. Adv Neural Inf Proces Syst 1:487–495

Anan Liu received the B.S., M.S. and Ph.D. degrees in electronic engineering from Tianjin University of China. He is currently an associate professor at the School of Electronic Information Engineering in Tianjin University. His research interests include computer vision, gesture recognition and medical image processing.

Zhengyu Zhao is currently a master candidate at the School of Electronic Information Engineering, Tianjin University. His current research interests are in computer vision, image forensics and multimedia retrieval.

Chengqian Zhang received the Ph.D. degree in electronic engineering from Tianjin University of China. She is currently an associate professor at the School of Electrical Engineering and Information in Southwest Petroleum University. Her research interests include digital watermarking and data hiding.

Yuting Su received the B.S., M.S. and Ph.D. degrees in electronic engineering from Tianjin University of China, in 1995, 1998 and 2001, respectively. He is currently a professor at the School of Electronic Information Engineering in Tianjin University. His research interests include digital video coding, digital watermarking, multimedia forensics, and multimedia retrieval.