2017 International Conference on Systems, Signals and Image Processing (IWSSIP)

Detecting Composite Image Manipulation based on Deep Neural Networks

Hak-Yeol Choi, Han-Ul Jang, Dongkyu Kim, Jeongho Son, Seung-Min Mun, Sunghee Choi, Heung-Kyu Lee*
* School of Computing, KAIST, Daejeon, South Korea
{*hychoi,hanulj,dkim,smmun*}@*mmc.kaist.ac.kr,sunghee*@*kaist.edu,*{*jhson,heunglee*}@*kaist.ac.kr*

***Abstract*—In this paper, we propose a composite manipulation detection method based on convolutional neural networks (CNNs). To our best knowledge, this is the first work applying deep learning for composite forgery detection. The proposed technique defines three types of attacks that occurred frequently during image forging and detects when they are concurrently applied to images. To do this, we learn the statistical change due to the manipulation through the proposed CNN architecture and classify the manipulated image. The proposed technique is effective since it learns integrated image of composite manipulation and extracts characteristic distinguished from original image. Since most attacks are applied in a composite way in real environment, the approach of the proposed technique has practical advantages compared to traditional forensics scheme. In addition, the experimental results demonstrate the reliability of the proposed method through results of high performance.**

***Index Terms*—Multimedia forensics, Composite manipulation detection, Deep learning, Convolutional neural networks**

I. Introduction

Due to the continuous development of computer graphics technology, manipulated images generated through computer graphics software such as Photoshop have become difficult to be distinguished from human eyes. The image forging can lead to a variety of social problems and losses. For example, if a manipulated image is used as evidence in a court of law, it can cause the misjudgment.

To solve this problem, multimedia forensic technology has been studied for a long time. Multimedia forensics is defined as a group of techniques that detect frequently occurring operations in image manipulation processes such as median filtering, resampling, and gamma correction. The multimedia forensics technology detects the manipulation of images by detecting fingerprints that are intrinsically present in the fake image.

In the past, forensic techniques were mostly focused on detecting specific single manipulations. It was a common approach to analyze the process of a specific operation and models statistical characteristics that can reveal the trace of forging. However, in the real application, multiple manipulation are applied in a composite way. Therefore, it is necessary to be able to detect the manipulation of images by analyzing them in an integrated way [1].

It is not easy to find a single hand-crafted feature in an image that contains various manipulations because it has different statistical properties for each manipulation type. In addition, the approach through the specific statistical analysis of each attack has its limitations because different kinds of image processing can change statistical fingerprints of different operations.

However, limitations of existing forensic methods are expected to be solved through a deep learning based approach. The deep learning framework is a technique that automatically learns the necessary features from a huge set of images and can distinguish different classes through it. Inspired by the action of the human brain, researchers have attempted to learn and classify various features in images through deep multi-layer neural networks such as Deep Boltzmann Machines [2], Deep Autoencoders [3], and Convolutional Neural Networks (CNNs) [4]. If such a deep learning approach is applied to forensics research, it is expected that it will be able to differentiate from existing techniques. Especially, the problem of composite manipulation detection can be solved very effectively.

It is possible to detect multiple manipulations without designing individual statistical characteristics for each attack since the deep neural network can automatically learn the features that are distinguished from the non-manipulated image sets. Also, it is possible to guide image features that are useful for detection of composite manipulation through back propagation algorithms.

In this paper, we perform composite manipulation detection method based on CNNs, the deep neural networks technique known to be the most efficient. To do this, we selected the three general image manipulation. Then, we designed and train a CNN architecture to distinguish compositely manipulated images from the original ones. The proposed network learns and detects statistical changes integrally and automatically. Through this approach, we achieve composite manipulation detection that has not been properly handled in existing techniques.

This paper is organized as follows. In the Section II we explore existing forensics detection methods related to the proposed method. In the following Section III the proposed CNN architecture is described. And Section IV verifies the performance of the proposed technique through various experimental results. The paper is concluded at Section V.

II. Related Work

There have rarely been conducted forensic studies considering composite manipulation. However, when various manipulations were applied independently, there were some detectable forensics techniques. Those techniques are capable of

978-1-5090-6344-4/17/$31.00 ©2017 IEEE

universal detection of various manipulations. Stamm and Liu proposed a technique for analyzing histograms of images to detect contrast enhancement, histogram equalization, gamma correction and JPEG compression [5]. Fan et al. constructed Gaussian mixture models (GMMs) of image patches to achieve universal forensics detection [6]. Li et al. detected various attacks by analyzing local pixels in the residual domain [7]. Also, Li et al. constructed a tampering possibility map by combining a statistical feature-based detector and a copy-move forgery detector for forgery detection [1].

In addition, CNN-based forensic techniques have been recently introduced. Because of the reliable performance of CNNs, deep learning techniques are being applied to forensics very quickly. The CNN-based approach has the advantage that the technique is more flexible than the approach using hand-crafted image features.

Chen et al. proposed a median filtering method based on CNNs [8]. In this technique, they expose median filtered images through learning with small image blocks. Bayar and Stamm detected a universal image manipulation using CNN architecture [9]. The method have a high detection rate when one of the attacks among median filtering, Gaussian blurring, additive white Gaussian noise, or resampling is applied.

Nogueira et al. introduced a fingerprint liveness detection scheme via CNNs [10]. In addition, research has been conducted to identify features that are impossible to identify with the human eye. In this regard, Tuama et al. and Bondi et al. developed a camera model identification technique using CNNs [11], [12].

III. Proposed Method

A. The goal of the proposed method

The proposed technique aims to detect common composite forgery. The proposed method is designed to detect the following three processes which occur frequently and complexly in the image manipulation process.

· Gaussian blurring
· Median filtering
· Gamma correction

In the real case, each operation does not necessarily occur in a composite manner. Therefore, the proposed scheme should be able to detect all combinations of each process. There are 7 combinations of manipulation cases for the above three attacks. The goal of proposed technique is to expose all 7 types of processes.

In addition, the proposed technique aims to detect small sub-image block units. This makes it possible to directly estimate the operation area in case of manipulation of image.

B. The General Architecture of CNN

As mentioned earlier, CNNs can automatically learn the characteristics of images and use them for image classification. CNNs have deep architecture based on multi-layer. Although the design of CNNs can vary, the general CNN architecture includes convolutional layers, pooling layers and fully-connected layers.

The convolutional layer consists of various convolutional filters and acts as a collection of feature extractors. The convolution result of multiple inputs is fed to element-wise non-linearity [13]. And an output feature map is generated as a result of element-wise non-linearity. In the pooling layer, the dimensionality of the feature map is reduced [14]. Such a dimensionality contraction leads the reduction of computational cost in the training and reduces the chance of over-fitting. Finally, in the fully connected layer, classification is performed through the features learned previously. Multiple fully connected layers can be stacked, and the final layer is learned to score each class. Training is the process of minimizing the loss between the network output and the actual label. During training, the coefficients of the convolutional filter are learned through iteration of the feed-forward and back-propagation process. Through such processes, it is possible to automatically learn feature or statistical properties of an image and classify them according to their purpose.

C. The Proposed CNN Architecture

Figure 1 describes the CNN architecture of the proposed scheme. The proposed architecture was inspired by VGGNet architecture [15]. The input of the proposed technique is a 64×64 sub-image block of RGB 3 channels. We designed the depth of the network layer not to be too deep because it fed images in small sub-block units. In the proposed architecture, two convolutional layers and one pooling layer are repeated three times. By stacking these two types of layers, the networks can learn from low-level features to high-level features. There are three fully-connected layers at the end of the proposed CNN architecture. Finally, the network determines whether the image has been manipulated or not by softmax activation function.

Convolutional Layer : The convolutional layer generates output feature map through convolution operation and element-wise non-linearity function. The output feature map of the convolutional layer can be viewed as a specific representation of the input image. Among the two components of the convolutional layer, the convolution operation can be expressed as follows:

$$x_j^l = \sum_{i=1}^{n} x_i^{l-1} * \omega_{ij}^{l-1} + b_j^l, \tag{1}$$

where $*$ is convolution operator and x_j^l is j-th output feature map of the hidden layer l. Also, ω_{ij}^{l-1} is convolution kernel which interlocks the i-th output map of layer $l-1$ and the j-th output map of layer l. And b_j^l is the bias parameter for the j-th output map of layer l. The convolution operation controls the number of pixels to be computed. It also increases the generalization performance by reducing the number of free variables. After convolution operation, non-linearity operation is obtained by applying element-wise non-linear activation function. The proposed scheme uses a rectified linear units (ReLUs) function as an activation function since it is known

978-1-5090-6344-4/17/$31.00 ©2017 IEEE

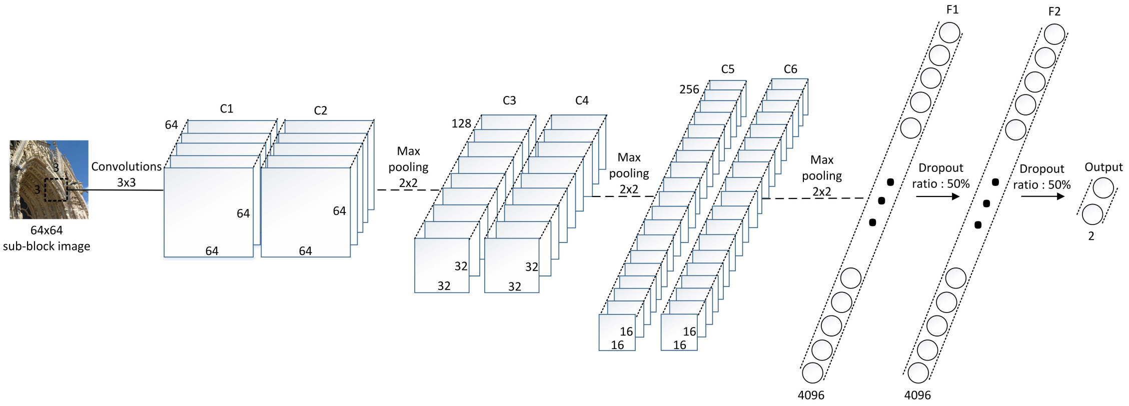

Fig. 1: The proposed CNN architecture

to help fast convergence on large datasets [16]. The ReLu function can be represented as follows:

$$f_{i,j} = \begin{cases} 0 & \text{for } {x^l}_{i,j} < 0 \\ {x^l}_{i,j} & \text{for } {x^l}_{i,j} \geq 0 \end{cases}, \qquad (2)$$

where (i, j) is the pixel index in the feature map and $x^l_{i,j}$ is the input patch located at (i, j) of layer l.

Pooling Layer : If all the feature maps passed from the convolutional layer are used for classification, there is a risk of overfitting as well as high complexity. Therefore, the pooling layer divides the feature map by a small overlapping window and reduces the dimension by leaving only a single value per window.

The proposed technique uses a max-pooling technique that leaves only the maximum value in the window. We used 2×2 for the window size and 2 as a striding factor. This dimensionality reduction made it possible to learn higher-level features [14].

Fully-Connected Layer : In this process, the learned feature is fed to the top layer of CNNs through two fully connected layers F1 and F2. The output layer has one neuron for each class. In the last layer, the probability of belonging to each class is quantified through softmax activation function. And the back-propagation algorithm adaptively changes the weight and bias. The back-propagation leads to the process of automatically extracting features as a classification result.

In the proposed scheme, two fully-connected layers with 4096 neurons were designed, and finally there is a final layer with two neurons. The dropout technique [17] was applied to proposed fully-connected layers F1 and F2. The dropout scheme helps performance improvement by reducing co-adaptations of neurons and forcing them to learn more intense features. We applied a dropout rate of 50% in the proposed architecture.

IV. Experimental Results

A. Experimental Setup

For objective testing, experimental images were obtained from the BOSS RAW database [18] and Dresden database [19]. The image used in the experiment is 7,429 raw image, and the size of images varies from 512×512 to 5,202×3,465. To minimize the dependency of the image own property during training, original and manipulated images were divided into 64×64 sub-image blocks and shuffled completely before learning. Training and test set were composed of 70% and 30%, respectively. The number of sub-image blocks are 700,000 and 300,000 for training and testing, respectively.

TABLE I: The image processing combinations used to create training set of image (GB : Gaussian Blurring, MF : Median Filtering, GC : Gamma Correction)

Manipulation index	Combination of manipulation
Comb. #1	GB, MF, GC
Comb. #2	GB, MF
Comb. #3	GB, GC
Comb. #4	MF, GC
Comb. #5	GB
Comb. #6	MF
Comb. #7	GC

During training, the ratio between original and manipulated images was set to 1/1. The all of training, test and accuracy tests proceeded with independent image sets. The CNNs parameter was applied to batch size of 50, momentum of 0.9, and learning rate value of 0.01. Totally, 100,000 iterations were performed for training.

For the experiment, training and test images were generated through composite manipulation of Table I combination. For the Gaussian blurring simulation, 3×3 kernel size with a standard deviation σ=1.1 are set as a parameter. Also, the 3×3 kernel for median filtering and $1/2$ as a gamma value γ are applied. Each combination was included in the same number of times. Therefore, the manipulated image set contains images of $1/7$ rate for each forging combination.

For the accuracy test, 1,000 of 512×512 sized images were used and 7,000 manipulated images were generated by applying 7 combinations of Table I to 1,000 of orginal images. Figure 2 shows the sample original and manipulated images used in the accuracy test.

The images for the accuracy test were cropped and tested

978-1-5090-6344-4/17/$31.00 ©2017 IEEE

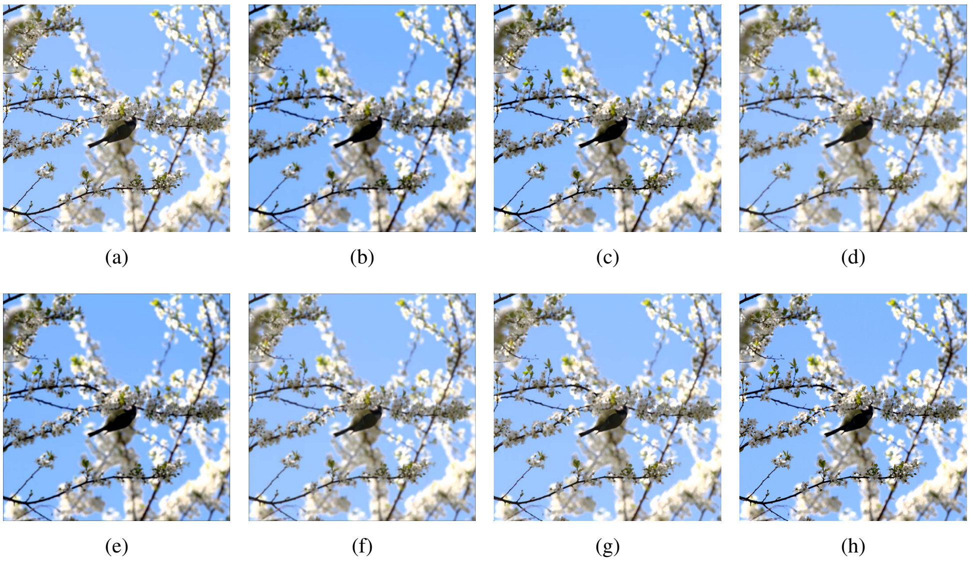

Fig. 2: The sample images used in accuracy test. (a) original, (b) Comb. #1, (c) Comb. #2, (d) Comb. #3, (e) Comb. #4, (f) Comb. #5, (g) Comb. #6, (h) Comb. #7

TABLE II: The accuracy of the proposed technique for 64×64 sub-image block units

Manipulation index	Accuracy
Original Image	81.93%
Comb. #1	96.50%
Comb. #2	92.72%
Comb. #3	96.44%
Comb. #4	92.85%
Comb. #5	92.40%
Comb. #6	89.73%
Comb. #7	55.91%

TABLE III: The accuracy of the proposed technique for image unit judged according to MVS

Manipulation index	Accuracy
Original Image	91.10%
Comb. #1	99.40%
Comb. #2	99.40%
Comb. #3	99.30%
Comb. #4	99.10%
Comb. #5	97.90%
Comb. #6	98.70%
Comb. #7	57.60%

at 64×64 as in the training process, and finally the majority voting system (MVS) was applied to make a judgement on one image. All learning and experiments were done through the GPU device. Image manipulation simulation was also performed in MATLAB R2014b. Finally, the proposed CNN architecture was implemented and tested through the Caffe deep learning framework [20].

B. Performance Evaluation

Table II shows the performance of the block unit of the proposed technique. Table III shows the accuracy rate of an image unit through MVS. General experimental results showed a very high level of results. However, we found that the results were significantly lower for Comb. #7. That means that gamma correction is not enough trained compared to other manipulations. We can observe that the learning process is more sensitive to the specific image processing when the evenly distributed manipulated image is learned in the deep neural network architecture.

Figure 3 shows false detection samples of the proposed method for each manipulation combinations. False detection was found to be predominant in highly textured regions, very dark regions, and defocused regions.

Especially, in the accuracy test for the original image, false detection occurred when blurred region due to defocusing exists in the image itself. It can be interpreted to be due to Gaussian blurring in trained manipulation. In the manipulated image, the area that is often misjudged was mostly the dark, flat area such as evening sky, and a highly textured region such as branches. The remaining false detection cases showed similar tendency.

V. Conclusion

In this paper, we introduced a deep neural network based forensic method which able to detect composite manipulation. Since the manipulation occurs mostly in the composite ways in the real environment, the proposed technique is more practical than the approach through analysis of single manipulation. This novel forensics framework was possible since CNNs can integrally recognize the statistical changes.

In the proposed method, we define three types of image manipulation process which are applied generally. And the system distinguished compositely manipulated image from original image based on learned CNNs. The CNN architecture was well suited to detecting composite manipulations because

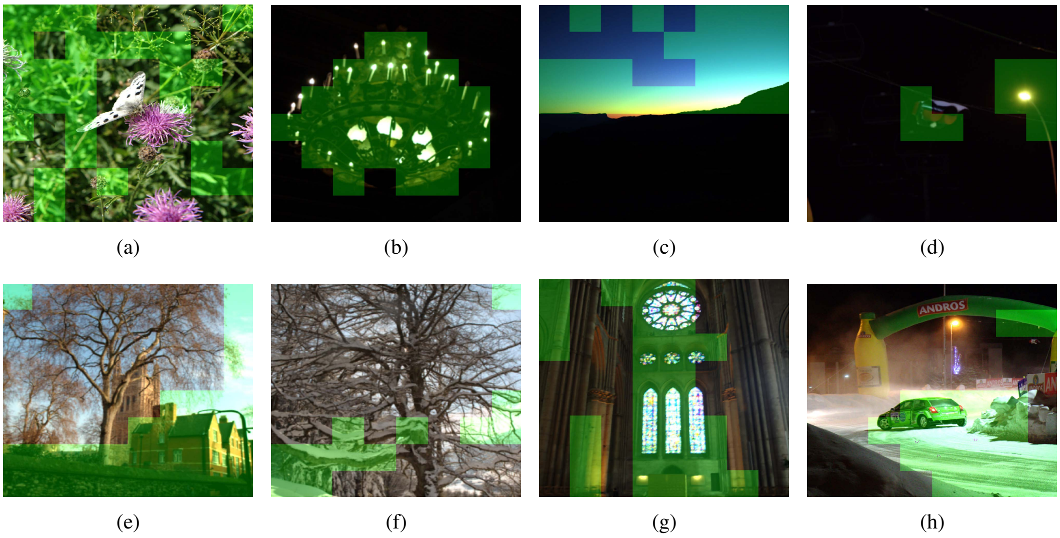

Fig. 3: The samples of false detection of the proposed method. The green colored blocks are correctly judged one. (a) original, (b) Comb. #1, (c) Comb. #2, (d) Comb. #3, (e) Comb. #4, (f) Comb. #5, (g) Comb. #6, (h) Comb. #7

they learned multiple features in a compounding way. The experimental results demonstrated the high performance of the proposed architecture.

The proposed technique solved the composite manipulation problem for the first time, but many issues can still be studied further. First, it is possible to improve the phenomenon of being more sensitive to specific attacks. It is also meaningful to analyze how the CNN architecture shows some detection capability for new, untrained manipulations.

ACKNOWLEDGMENT

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP) (No. 2016R1A2B2009595) and the Institute for Information & communications Technology Promotion(IITP) grant funded by the Korean government(MSIP) (No.R0126-16-1024, Managerial Technology Development and Digital Contents Security of 3D Printing based on Micro Licensing Technology)

REFERENCES

[1] H. Li, W. Luo, X. Qiu, and J. Huang, "Image forgery localization via integrating tampering possibility maps," *IEEE Trans. on Information Forensics and Security*, vol. 12, pp. 1240–1252, 2017.

[2] R. Salakhutdinov and G. Hinton, "Deep Boltzmann Machines," *International Conf. Artificial Intelligence and Statistics (AISTATS)*, vol. 1, pp. 448–455, 2009.

[3] H. Larochelle, Y. Bengio, J. Louradour, and P. Lamblin, "Exploring Strategies for Training Deep Neural Networks," *Journal of Machine Learning Research*, vol. 1, pp. 1–40, 2009.

[4] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, pp. 2278–2323, 1998.

[5] M. C. Stamm and K. J. R. Liu, "Forensic detection of image manipulation using statistical intrinsic fingerprints," *IEEE Trans. on Information Forensics and Security*, vol. 5, pp. 492–506, 2010.

[6] W. Fan, K. Wang, and F. Cayre, "General-purpose image forensics using patch likelihood under image statistical models," in *IEEE Int. Workshop on Information Forensics and Security*, 2014, pp. 165–170.

[7] H. Li, W. Luo, and X. Qiu, "Identification of various image operation using residual-based features," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. PP, pp. 1–15, 2016.

[8] J. Chen, X. Kang, Y. Liu, and Z. J. Wang, "Median Filtering Forensics Based on Convolutional Neural Networks," *IEEE Signal Processing Letters*, vol. 22, pp. 1849–1853, 2015.

[9] B. Bayar and M. C. Stamm, "A Deep Learning Approach To Universal Image Manipulation Detection Using A New Convolutional Layer," *2016 Information Hiding and Multimedia Security*, pp. 5–10, 2016.

[10] R. F. Nogueira, R. d. A. Lotufo, and R. C. Machado, "Fingerprint liveness detection using convolutional neural networks," *IEEE Trans. on Information Forensics and Security*, vol. 11, pp. 1206–1213, 2016.

[11] A. Tuama, F. Comby, and M. Chaumont, "Camera model identification with the use of deep convolutional neural networks," in *IEEE Int. Workshop on Information Forensics and Security*, 2016.

[12] L. Bondi, L. Baroffio, D. Guera, and P. Bestagini, "First steps toward camera model identification with convolutional neural networks," *IEEE Signal Processing Letters*, vol. 24, pp. 259–263, 2017.

[13] Y. Liu and X. Yao, "Evolutionary design of artificial neural networks with different nodes," *Proceedings of IEEE International Conference on Evolutionary Computation*, pp. 913–917, 1996.

[14] D. Scherer, M. Andreas, and S. Behnke, "Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition," *International Conference on Artificial Neural Networks (ICANN)*, pp. 92–101, 2010.

[15] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *International Conference on Learning Representations*, pp. 1–14, 2015.

[16] A. Krizhevsky, I. Sutskever, and H. Geoffrey E., "Imagenet Classification with Deep Convolutional Neural Networks," *Advances in Neural Information Processing Systems (NIPS)*, pp. 1–9, 2012.

[17] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *In:arXiv:1207.0580*, pp. 1–18, 2012.

[18] Boss raw database. [Online]. Available: http://exile.felk.cvut.cz/boss/BOSSFinal/index.php?mode=VIEW

[19] T. Gloe and R. Bohme, "Dresden image database for benchmarking digital image forensics," *Journal of Digital Forensic Practice*, vol. 3, pp. 150–159, 2010.

[20] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional Architecture for Fast Feature Embedding," *Proceedings of the ACM International Conference on Multimedia (ACMMM)*, pp. 675–678, 2014.