Copy-move forgery detection using convolutional neural network and K-mean clustering

Ava Pourkashani[1], Asadollah Shahbahrami[2], Alireza Akoushideh[3]
[1]Department of Computer Engineering, Faculty of Engineering, Islamic Azad University, Rasht Branch, Iran
[2]Department of Computer Engineering, Faculty of Engineering, University of Guilan, Rasht, Iran
[3]Department of Electronic Engineering, Faculty of Shahid-Chamran, Technical and Vocational University (TVU),
Guilan Branch, Rasht, Iran

Article Info

Article history:

Received Dec 23, 2019
Revised Sep 30, 2020
Accepted Oct 2, 2020

Keywords:

Convolutional neural network
K-mean clustering
Copy-move
Corner detection
Image forgery

ABSTRACT

Copying and pasting a patch of an image to hide or exaggerate something in a digital image is known as a copy-move forgery. Copy-move forgery detection (CMFD) is hard to detect because the copied part image from a scene has similar properties with the other parts of the image in terms of texture, light illumination, and objective. The CMFD is still a challenging issue in some attacks such as rotation, scaling, blurring, and noise. In this paper, an approach using the convolutional neural network (CNN) and k-mean clustering is for CMFD. To identify cloned parts candidates, a patch of an image is extracted using corner detection. Next, similar patches are detected using a pre-trained network inspired by the Siamese network. If two similar patches are not evidence of the CMFD, the post-process is performed using k-means clustering. Experimental analyses are done on MICC-F2000, MICC-F600, and MICC-F8 databases. The results showed that using the proposed algorithm we can receive a 94.13% and 96.98% precision and F1 score, respectively, which are the highest among all state-of-the-art algorithms.

Corresponding Author:

Alireza Akoushideh
Department of Electronic Engineering, Faculty of Shahid-Chamran
Technical and Vocational University (TVU)
Guilan Branch, Rasht, Iran
Email: akushide@tvu.ac.ir

1. INTRODUCTION

Digital image processing has many advantages in many applications. Today's image processing tools without leaving obvious traces make editing or manipulating digital images easily and fast. The recent growth in image-manipulation software has led to challenges in prominent or evidence documents [1]. These tampered or manipulated digital images can be used for various targets such as to delude the public into thinking, change political views, and leave disturbing effects in public [2]. Therefore, image forgery detection algorithms have been proposed in this regard. In terms of previous knowledge of images, these algorithms can be divided into two categories: active and passive detection [3]. Active detection methods are based on digital watermarking or digital signature. In addition, passive detections include two types of approaches: forgery type-independent and forgery type-dependent. In the type-dependent detection, forgery is detected according to the type of forgery, while in the forgery types-independent, effects of image compressions or repetitive patterns are analyzed. There are some image forgery detection techniques such as copy-move, segmentation-based algorithms, passive detection, and splicing [4]. Splicing forgery is a method in which several copied regions of different images are pasted in an image [5], while copy-move forgery is a method that is done using pasting one or more copied parts of an image in the same image. Copy-move forgery is often used for hiding unwanted region(s) of an image. Copied contents are often selected from a textured region of the image to be invisible from naked eyes. This type of forgery is more popular among the mentioned forgeries because there is a more opportunity that copied regions of an image are similar in texture, content, and illumination features. Figure 1 presents an image taken from the MICC-Fx dataset series with a copy-move forgery attack.

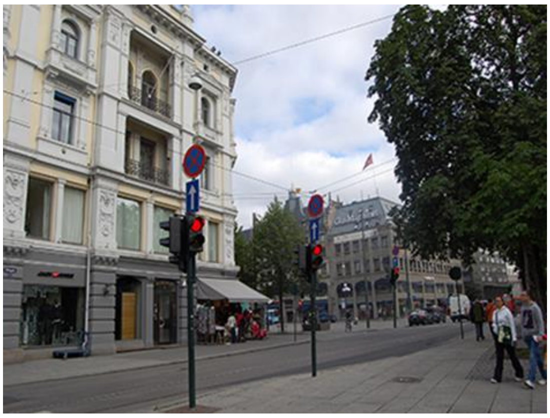

Figure 1. An example of copy-move forgery

As shown in Figure 1, the detection of copied regions of the image by naked eyes is hard. In addition, copied regions may be attacked. In addition, the copied parts can be noised, scaled, rotated, compressed, noised, or blurred, which state-of-the-art detection algorithms fall in the challenge to compare with images that have not any attacks. Several studies have been conducted on copy-move forgery detection (CPFD). In terms of their performance mechanism, the CMFD algorithms are classified into block detection and feature-based detection algorithms. In the block-based algorithm, an image is split into several non-overlapped blocks. After that, the similarity of the blocks is compared [6]. In the feature-based algorithm, the feature extractors such as Scale-invariant feature transform (SIFT) [7], speeded-up robust features (SURF), and local binary pattern (LBP) are applied to the image and are analyzed. One common feature extraction-based method is the Zernike moments or blur invariant [8]. It has provided good results. But still an effective algorithm for the CMFD to overcome the mentioned challenges especially compression algorithms (JPEG) is a research topic.

In this research, we propose a CMFD algorithm based on the feature extraction technique. The proposed approach includes three steps. First, the Harris corner detection technique is applied to an image. In the second step, after extracting the patches, the matching process is done around each patch using convolutional neural networks (CNN) [5, 9]. We use a method inspired by Siamese networks [10]. As two matched patches are not good evidence for forgery, in the third step, the k-means clustering for matching several patches together is used. Our experimental results show that the proposed algorithm outperforms the state-of-the-art approaches, even in multiple forgeries.

The reminder of the paper is structured as: A review to related works and researches is mentioned in section 2. The proposed algorithm has been introduced and discussed in section 3. Experimental results are illustrated in section 4. In addition, the proposed method is compared to several state-of-the-art approaches [11] in terms of precision, recall, and F1-score criteria in this section. Finally, conclusions are in section 5.

2. RELATED WORK

There are many approaches for the CMFD based on blocking and feature extraction. Some of these algorithms will be introduced in the following. At first, we present some feature extractors such as local binary pattern (LBP) textural descriptor and Zernike moments which are used for CMFD. Some LBP feature properties such as being invariant against illumination, image transformations, and statistical information of the textural structure of an image are an efficient feature for defining the CMFD algorithms. In addition, multi-resolution LBP, one of the LBP extensions, was implemented for the CMFD [12]. The authors in [12], with adding a k-d tree algorithm to the LBP, were depicted that this approach could recognize copy-move forgery in various distortions challenges which have been mentioned before. The Zernike moments, the shape descriptors, are implemented because of their noises resistance properties [8, 11]. The Zernike moments and local sensitive hashing (LSH) have been used for copy-move forgery detecting [8]. Because of being local sensitive hashing, this feature achieved better performance against moderate scaling, additive white Gaussian noise, JPEG compression, and blurring [8]. Speeded-up robust features (SURF) and Scale-invariant features transform (SIFT) are two common and regular approaches for copy-move forgery detection. The researchers have been combined the SIFT method with other approaches to enrich the performance of the CMFD. The SURF and SIFT methods have conventionally implemented for detecting of similar regions in an image in typical challenges such as noise, scaling, and blurring. But matching procedure in these algorithms is not the evidence of forgery. To solve this challenge, the authors in [13] after running the SIFT algorithm, performed hierarchical clustering to detect matched points clusters regarding match single points. Random sample consensus (RANSAC) is another algorithm that estimates the homography matrix and matched the clusters. The authors in [14], depicted that the SIFT based algorithms are proper for the CMFD. A combination of the discrete-time wavelet transform (DWT) and SIFT made better results on the CMFD. Regarding to DWT theory, the LL sub-bands of DWT used in raw images instead of using SIFT [15].

The Dyadic wavelet transform (DyWT) approach was implemented for the CMFD. In other words, against traditional wavelet transform tools, coefficients in each decomposition are not reduced. Comparing of wavelet and scaling coefficients were run for each block to detect similar blocks. After dividing an image into some overlapped blocks, the LL1 and HH1 sub-bands were compared with each other. To make a decision in the last step, the Euclidean measure between matched blocks was calculated. The authors in [16] used the SURF algorithm for the CMFD. The experimental results depicted that the SURF can detect a forgery in point of view changed scenes and cases of textured that is still challenging in many algorithms. The authors in [17] implemented the singular value decomposition (SVD) on the regions of an image after quantization of discrete-time cosine transform (DCT). Regarding using this method, the CMFD had some advantages such as being resistant against Gaussian noise, blur attacks, and being able to detect multi copy-move forgery. Moreover, the CMFD was implemented in the spatial domain, while it is resistant to rotation attacks. First, the image was split into n×n overlapped blocks to extract the features from the blocks by four nested frames. The k-means clustering algorithm was used to group the overlapped blocks. Using radix sort, each block group was lexicographically ordered. After that, the distance between the nearby blocks was calculated to determine the overall similarity. Because of translation and scale-invariant properties, the Fourier Mellin transform (FMT) was selected for the CMFD. After splitting the image into several overlapped blocks, the FMT was applied for feature extraction. After that, counting Bloom filtering was applied with hashing. The low complexity of bloom filtering against other methods such as lexicographic sort was the main reason for using it. Regarding the fact that finding matched blocks is not an acceptable reason to detect forgery, the authors prove that the distance of matched blocks to an assumptive array was also considered to make a decision.

3. PROPOSED ALGORITHM

The proposed algorithm for CMFD includes three main steps: corner detection, keypoint extraction, and matching, and making a decision.

3.1. Corner detection

The main part of an image may be attacked by scaling manipulation and there is no previous information about where the cloned region started or how it was scaled. To cope with these problems, the image pyramid presentation is proposed as illustrated in Figure 2. Based on Figure 2, level 1 is assumed as an input image that can be scaled to an image shown in levels 0, 2, and 3. Using a pyramid image makes the proposed approach robust against scale attacks. However, using scaled images for training a convolutional neural network (CNN) helps the proposed approach to be more resistant to scale attack. For each image in different pyramid presentation, corners are extracted. We split images in each pyramid level to m×n patches, where the centre of the block is a corner. Here, m and n are width and height of patch, respectively, which are adjusted according to the CNN input size. A modified version of the Harris corner detector is used for corner detection. Harris corner detector for a given image I is defined as (1).

$$M = \sum_{(x,y)\in\in w} \begin{bmatrix} I_x^2 & I_x I_y \\ I_y I_x & I_y^2 \end{bmatrix} \tag{1}$$

where x and y are the centres of the area over w, and Ix and Iy are derivatives of I in over x and y. Extracted corners using Harris detector are invariant against rotation, brightness variations, and scaling but not against blurring. Blurring causes the edges to be smoothed. Since corners can be defined by points of the image that have multi-directional edges, blurring reduces edge intensities and consequently reduces corner intensity or removes the corners. To cope with missed corners caused by blurring, we use sharpening techniques.

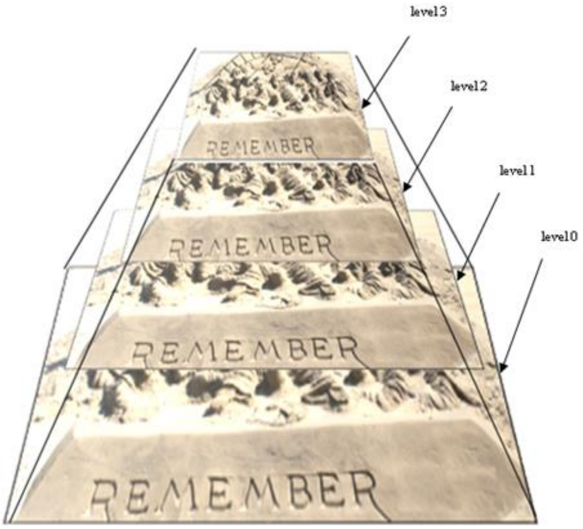

Figure 2. Mage pyramid presentation for the CMFD

There are several approaches for image sharpening (e.g., Laplacian [18]). However, we should consider that an input image may be not blurred. Therefore, using simple approaches may increase and bold the unwanted edges. These edges may increase the number of corners. Although the unwanted corners may reduce the overall performance, the next steps of the algorithm will reject them as much possible. To overcome these problems, iterative sharpening (IS) approach is used, which is defined as in (2) for input image I.

$$H = I - D * I \tag{2}$$

where D is an edge smoothing filter such as averaging or Gaussian filter and H is the result of the difference between input image I and blurred image (I * D). In (2), '*' denotes a convolution operator. Now, H is a high-frequency image that will be gamma corrected and then added to the blurred image as shown in (3) to (6).

$$I_1 = H^{\gamma} + I * D \tag{3}$$

$$I_{j+1} = H^{\gamma} + I * D_j \tag{4}$$

$$\mathrm{I_n(u,v)} = \left(\sum_{\mathrm{i=0}}^{\mathrm{n-1}} \mathrm{D(u,v)^i}\right) \mathrm{H^*(u,v)} + \mathrm{D(u,v)^n I}(u,v) \tag{5}$$

$$\mathrm{I_n(u,v)} = \frac{\mathrm{(1 - D(u,v)^n)H^*(u,v)}}{\mathrm{1 - D(u,v)}} + \mathrm{D(u,v)^n I(u,v)} \tag{6}$$

where $\mathrm{I_n}$(u, v) is the resulted image after n iteration in the frequency domain, and H*(u, v), D(u, v)^n, and I(u, v) are Fourier transformation of H*, D, and I, respectively. The difference between this algorithm and the Laplacian sharpening are depicted in Figure 3 as evidence of the effect of image sharpening for both normal image and sharpened image.

After three and four iterations, the image produced using LoG sharpening has several noises while the IS sharpening this effect cannot be seen. White Gaussian noise was applied to the input image. The results are presented in the 4th iteration of LoG. Being robust against sharpening is important for the CMFD because the input image may be sharpened manually as an attack or may be naturally sharp. In this case, simple sharpening methods may add several noises, as shown in the 4th iteration of the LoG.

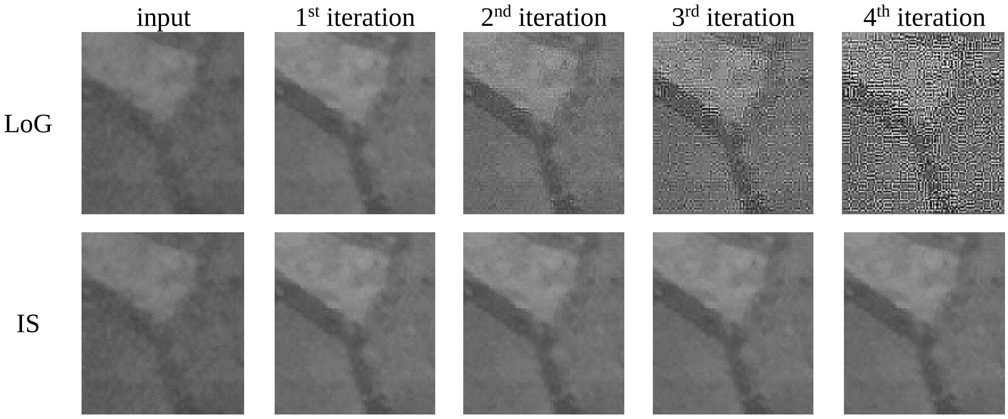

Figure 3. Difference of iterative sharpening and Laplacian sharpening

3.2. Matching

For matching two blocks, we use a non-conventional architecture for convolutional neural networks (CNN). Conventional image matching methods use features such as histogram of the oriented gradient, Zernike or hu moments, and local binary patterns. Instead of using the mentioned features, we leave them to do by the CNN. Figure 4 presents the architecture of CNN used for matching two patches of the image.

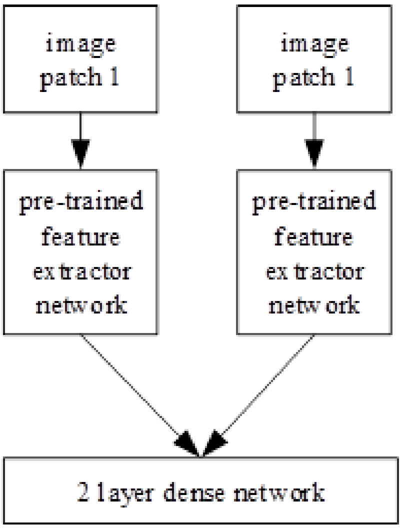
image patch 1
image patch 1
pre-trained feature extractor network
pre-trained feature extractor network
2 layer dense network

Figure 4. The non-conventional architecture of CNN used for matching two patches of the image

To achieve a pre-trained feature extractor network, a dense layer (as a fully connected network or support network) is removed. To train this network, we crop different patches from images in the ImageNet dataset (fall 2011 release). We randomly select 100 images from each category. Each image was segmented to m×n non-overlapped blocks, where m and n are the width and height of input of the network. In the training phase, we divide these blocks into two classes: similar and non-similar. Similar blocks are also augmented using conventional attacks and image manipulation including adding noise (such as salt and pepper and additive white Gaussian), rotation, scaling, brightness, and contrast. In addition, we augment image patches to avoid network sensitivity to shift translations. We choose the VGG16 network as a baseline for selecting the best pre-trained network. We also test VGG19, ResNet, and AlexNet. Among these networks, AlextNet was the best for finding image patch pairs. For learning the network, stochastic gradient descent with momentum was used. Drop-out strategy was used to avoid over fitting and make network connections as simple as possible. The learning rate was considered 0.001 and the number of mini-batch was selected to be 128 experimentally.

3.3. Decision making

Finding two patches that are similar together is not evident for the CMFD because sometimes images have their repetitive patches. To avoid this problem, we make a decision by matching several patches. To match several patches together, we applied k-means clustering. The main idea is that instead of matching separated patches of images, a cluster of patches should be matched. Each patch in a cluster should be close to other patches in the point view of pixel distance. Figure 5 illustrates the location of corners and corresponding matches.

This figure depicts the location of patches. We use k-mean clustering to classify them. One problem of the k-means clustering is estimating the number of clusters. To achieve an optimal number of clusters, the Davies-Bouldin criterion (DBC) was used. We test different values of clusters and selected an optimal (minimum) number of clustering. Figure 6 illustrates various numbers of clusters and the corresponding DBC.

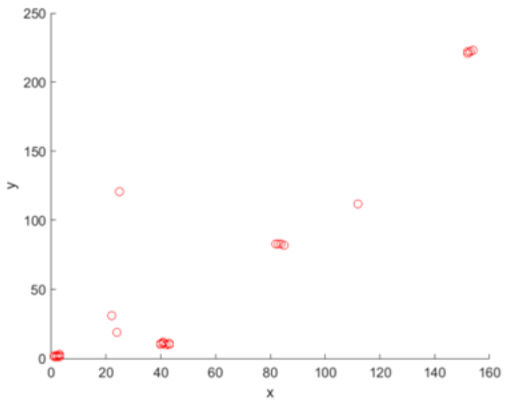

Figure 5. The location of the image of patches

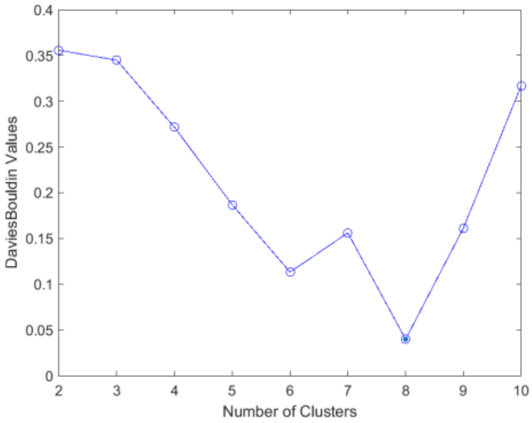

Figure 6. Different values for the number of clusters and Davies-Bouldin criterion

As shown in Figure 7, the minimum number of DBC is 8. Therefore, we consider 8 clusters to solve the problem. Figure 8 demonstrates the result of k-mean clustering where k=8. Next, we show the relation of clusters with each other as a weighted graph. Figure 8 illustrates a weighted graph inspired from the clustering result shown in Figure 7. Vertices and their names are the clusters and legends, respectively. Also, the edges are the number of matched images patches between each cluster. To simplify the graph, the nodes (so corresponding clusters) and the edges with a few numbers of patches and little weights, respectively, are truncated. The results show that two parts of the image are cloned.

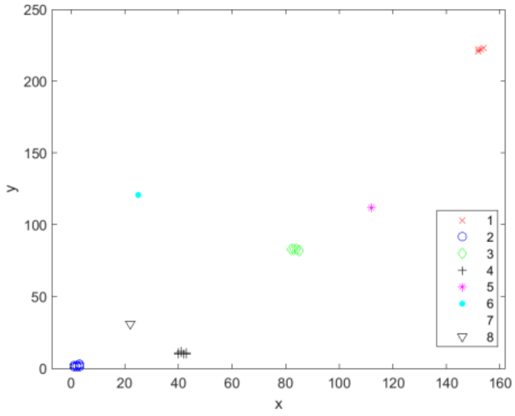

Figure 7. Clustered patches of images according to their location

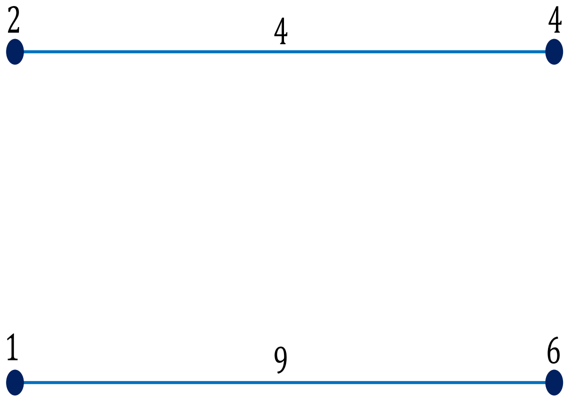

Figure 8. Weighted graph inspired from the clustering result shown in Figure 7

4. RESULTS AND DISCUSSION

In this section, experimental results are presented. First, we define how we evaluate basic measure criteria including true positive rate and false-positive rate. Then, environmental platforms and datasets are introduced and, finally, the implementation results are presented.

4.1. Criteria

True positive rate (TPR) and false-positive rate (FPR) were defined according to Jaccard index (7).

$$J(A,B) = \frac{A \cap B}{A \cup B} \tag{7}$$

where J(A, B) is the Jaccard index between measurable A and B. Also, ∩ and ∪ are intersection and union operators, respectively. This index is also well known as the intersection of the union. Since we cannot deliver the output of our approach as ground truth, we measure them using a bounding box. To be more precise, the intersection of the delivered bounding box and the bounding of ground truth are used. When this index exceeds 0.5, it is assumed as true positive; otherwise, as a false positive.

4.2. Environmental platform and database

Experimental results platform in this research was a laptop with a Core i7 processor, 12 GB Memory, and GeForce graphic card, Ti980GTX series, with Windows 10 operating system. The proposed algorithm was implemented using MATLAB 2018b. Evaluation of the proposed approach are done on the MICC-F8multi, MICC-F600, and MICC-F2000 public databases which are include 2000, 600, and 8 images, respectively.

4.3. Results of implementation

At image level, the important measures are the number of correctly detected forged images, TP, the number of images that have been erroneously detected as forged, FP, and the falsely missed forged images FN. Using these parameters, we computed the measures Precision (p) and Recall (r) [4], which are defined as (8) and (9), respectively.

$$p = \frac{T_P}{T_P + F_P} \ and \ r = \frac{T_P}{T_P + F_N} \tag{8}$$

where p denotes the probability that a detected forgery is truly a forgery and r shows the probability that a forged image is detected. In Table 1, we also give the F1-score as a measure that combines recall and precision in a single value.

$$F1 = 2 \times \frac{p \times r}{p + r} \tag{9}$$

Recall, precision, and F1-score for different approaches (obtained from [11]) are illustrated in Table 1. Our proposed approach is the best in Recall, Precision, and F1-score criteria. The proposed approach can detect multiple cloned regions. We illustrate one of the experimental results on an image depicted in Figure 9. The red rectangles in Figure 9 denote the cloned parts. It can be realized that our proposed approach is highly robust against multiple forgeries attacks with scaling challenges.

Table 1. Comparison of the state of the arts in Recall, Precision, and F1-Score terms

Method	Precision (%)	Recall (%)	F1-score (%)
BLUR [5]	88.89	100	94.12
BRAVO [19]	87.27	100	93.2
CIRCLE [20]	92.31	100	96
DCT [21]	78.69	100	88.07
DWT [22]	84.21	100	91.43
FMT [23]	90.57	100	95.05
HU [23]	67.61	100	80.67
KPCA [22]	87.27	100	93.2
LIN [24]	94.12	100	96.97
LUO [25]	87.27	100	93.2
PCA [26]	84.21	100	91.43
SIFT [27]	88.37	79.17	83.52
SURF [28]	91.49	89.58	90.53
SVD [29]	68.57	100	81.36
ZERNIKE [30]	92.31	100	96
Proposed	94.13	100	96.98

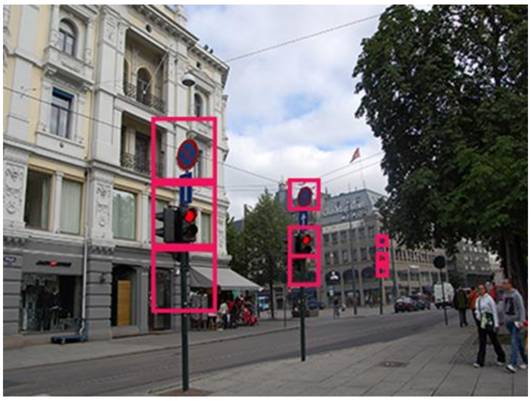

Figure 9. Forgery detection results of the proposed algorithm with multiple cloned regions

5. CONCLUSION

Regarding to importance of copy-move forgery, a common type of image tampering, we proposed an algorithm for copy-move forgery detection (CMFD) based on feature extraction. To find same patches or similar regions of an image, Harris corner detection is used. Convolution neural network (CNN) is also used for the matching process. To achieve the best result, we use a pre-trained network. We also use k-mean clustering to reduce the false-positive rate. The experimental results on considered datasets depicted that our algorithm outperforms others in terms of detection rate. In addition, experiments show the proposed can detect multiple forgeries. Ease of using CNN as a feature extractor makes it a good candidate solution for the CMFD. As a future work, the CNN architecture should be analyzed more, especially regarding the Siamese networks.

REFERENCES

[1] M. F. Hashmi, V. Anand, and A. G. Keskar, "Copy-move Image Forgery Detection Using an Efficient and Robust Method Combining Un-decimated Wavelet Transform and Scale Invariant Feature Transform," *AASRI Procedia*, vol. 9, pp. 84-91, 2014.

[2] Y. P. B. Liu, "Accurate estimation of primary quantisation table with applications to tampering detection," *Electronics Letters*, vol. 49, no. 23, pp. 1452-1454, 2013.

[3] J. Li, X. Li, B. Yang, and X. Sun, "Segmentation-Based Image Copy-Move Forgery Detection Scheme," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 3, pp. 507-518, 2015.

[4] T. M. Shashidhar and D. K. Ramesh, "Reviewing the Effectivity Factor in Existing Techniques of Image Forensics," *International Journal of Electrical and Computer Engineering (IJECE),* vol. 7, no. 6, pp. 3558-3569, 2017.

[5] B. Xiao, Y. Wei, X. Bi, W. Li, and J. Ma, "Image splicing forgery detection combining coarse to refined convolutional neural network and adaptive clustering," *Information Sciences,* vol. 511, pp. 172-191, 2020.

[6] Shashidhar T. M. and K. B. Ramesh, "Novel framework for optimized digital forensic for mitigating complex image attacks," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 5, pp. 5198-5207, 2020.

[7] S. Teerakanok and T. Uehara, "Copy-Move Forgery Detection: A State-of-the-Art Technical Review and Analysis," *IEEE Access*, vol. 7, pp. 40550-40568, 2019.

[8] S. J. Ryu, M. Kirchner, M. J. Lee and H. K. Lee, "Rotation Invariant Localization of Duplicated Image Regions Based on Zernike Moments," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 8, pp. 1355-1370, 2013.

[9] Z. J. Barad and M. M. Goswami, "Image Forgery Detection using Deep Learning: A Survey," *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS),* Coimbatore, India, 2020, pp. 571-576.

[10] G. Koch, R. Zemel and R. Salakhutdinov, "Siamese Neural Networks for One-shot Image Recognition," *Proceedings of the 32 nd International Conference on Machine Learning*, Lille, France, vol. 37, 2015.

[11] V. Christlein, C. Riess, J. Jordan, C. Riess, and E. Angelopoulou, "An Evaluation of Popular Copy-Move Forgery Detection Approaches," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 6, pp. 1841-1854, 2012.

[12] R. Davarzani, K. Yaghmaie, S. Mozaffari, and M. Tapak, "Copy-move forgery detection using multiresolution local binary patterns," *Forensic Science International*, vol. 231, no. 1, pp. 61-72, 2013.

[13] I. Amerini, L. Ballan, R. Caldelli, A. D. Bimbo, and G. Serra, "A SIFT-Based Forensic Method for Copy-Move Attack Detection and Transformation Recovery," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 1099-1110, 2011.

[14] K. B. Meena and V. Tyagi, "A hybrid copy-move image forgery detection technique based on Fourier-Mellin and scale invariant feature transforms," *Multimedia Tools and Applications*, vol. 79, no. 11, pp. 8197-8212, 2020.

[15] M. F. Hashmi, A. R. Hambarde, and A. G. Keskar, "Copy move forgery detection using DWT and SIFT features," *2013 13th International Conference on Intellient Systems Design and Applications*, Bangi, 2013, pp. 188-193.

[16] X. Bo, W. Junwen, L. Guangjie, and D. Yuewei, "Image Copy-Move Forgery Detection Based on SURF," *2010 International Conference on Multimedia Information Networking and Security*, Nanjing, Jiangsu, 2010, pp. 889-892.

[17] A. Parveen, Z. H. Khan, and S. N. Ahmad, "Block-based copy–move image forgery detection using DCT," *Iran Journal of Computer Science*, vol. 2, no. 2, pp. 89-99, 2019.

[18] A. Alsam, I. Farup, and H. J. Rivertz, "Iterative sharpening for image contrast enhancement," *2015 Colour and Visual Computing Symposium (CVCS)*, pp. 1-4, 2015.

[19] S. Bravo-Solorio and A. K. Nandi, "Exposing duplicated regions affected by reflection, rotation and scaling," *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Prague, 2011, pp. 1880-1883.

[20] J. Wang, G. Liu, H. Li, Y. Dai, and Z. Wang, "Detection of Image Region Duplication Forgery Using Model with Circle Block," *2009 International Conference on Multimedia Information Networking and Security*, Hubei, 2009, pp. 25-29.

[21] S. Battiato and G. Messina, "Digital forgery estimation into DCT domain: a critical analysis," *MiFor '09: Proceedings of the First ACM workshop on Multimedia in forensics*, pp. 37-42, 2009.

[22] M. Bashar, K. Noda, N. Ohnishi, and K. Mori, "Exploring Duplicated Regions in Natural Images," *IEEE Transactions on Image Processing*, pp. 1-1, 2018.

[23] L. G.-J. Wang Jun-Wen, Zhang Zhan, Dai Yue-Wei, Wang Zhi-Quan, "Fast and Robust Forensics for Image Region-duplication Forgery," *Acta Automatica Sinica*, vol. 35, no. 12, pp. 1488-1495, 2009.

[24] H.-J. Lin, C.-W. Wang, and Y.-T. Kao, "Fast copy-move forgery detection," *WSEAS Transactions on Signal Processing*, vol. 5, no. 5, pp. 188-197, 2009.

[25] W. Luo, J. Huang, and G. Qiu, "Robust Detection of Region-Duplication Forgery in Digital Images," *18th International Conference on Pattern Recognition (ICPR'06)*, Hong Kong, 2006, pp. 746-749.

[26] A. Popescu and H. Farid, "Exposing digital forgeries by detecting duplicated image regions," Department of Computer Science, Dartmouth College, 2004. [Online]. Available: www.cs.dartmouth.edu/farid/publications/tr04.html

[27] X. Pan and S. Lyu, "Detecting image region duplication using SIFT features," *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, Dallas, TX, 2010, pp. 1706-1709.

[28] B. Shivakumar and L. S. S. Baboo, "Detection of Region Duplication Forgery in Digital Images Using SURF," *International Journal of Computer Science*, vol. 8, no. 4, pp. 199-205, 2011.

[29] X. Kang and S. Wei, "Identifying Tampered Regions Using Singular Value Decomposition in Digital Image Forensics," *2008 International Conference on Computer Science and Software Engineering*, Hubei, 2008, pp. 926-930.

[30] S.-J. Ryu, M.-J. Lee, and H.-K. Lee, "Detection of copy-rotate-move forgery using Zernike moments," *International Workshop on Information Hiding*, vol. 6387, pp. 51-65, 2010.

BIOGRAPHIES OF AUTHORS

Ava Pourkashani Received the B.Sc. Degree in computer engineering (software orientation) from Islamic Azad University, Lahijan Branch, in 2007. She got her M.Sc. degree in E-Commerce from Azerbaijan University of Architecture and Construction. She is PhD candidate in Computer Engineering, Software Systems, in Islamic Azad University, Rasht Branch. Mrs. Pourkashani research interest is image forgery.

Asadollah Shahbahrami Received the B.Sc. and M.Sc. degrees in computer engineering (hardware and machine intelligence) from Iran University of science and technology and Shiraz University in 1993 and 1996, respectively. He got his PhD degree from Delft University of Technology, The Netherlands in 2008. He has been working at University of Guilan since August 1996. He is an associated professor position in Department of Computer Engineering at the University of Guilan. Dr. Shahbahrami research interests include advanced computer architecture, image and video processing, multimedia instructions set design, reconfigurable computing, parallel processing, and SIMD programming.

Alireza Akoushideh Received the B.Sc. and M.Sc. degree in Electrical engineering from University of Guilan and Amirkabir University of Technology (Tehran Polytechnic) in 1997 and 2000, respectively. From 2001 until now, he is a faculty member of Technical and Vocational University, Shahid-Chamran community college, Rasht, Iran. He got his Ph.D. degree from Shahid-Beheshti University, Tehran, Iran in 2016. As a visiting researcher, he worked with the SCS group in the Twente University, the Netherlands from January to September 2015. He has taught courses in FPGA, microprocessor and microcontrollers, computer architecture, and digital circuits. He research interests include machine vision, texture analysis, and Intelligence Transformation System (ITS), and FPGA implementation.