Dual branch convolutional neural network for copy move forgery detection

Nidhi Goel1 | Samarjeet Kaur2 | Ruchika Bala3

1 Department of Electronics and Communication Engineering, Indira Gandhi Delhi Technical University for Women, Delhi, India
2 Department of Electronics and Communication Engineering, Bharati Vidyapeeth College of Engineering, Delhi, India
3 Department of Information Technology, Indira Gandhi Delhi Technical University for Women, Delhi, India

Correspondence
Nidhi Goel, Department of Electronics and Communication Engineering, Indira Gandhi Delhi Technical University for Women, Delhi, India.
Email: nidhi.iitr1@gmail.com

Abstract
The advent of digital era has seen a rise in the cases of illegal copying, distribution and forging of images. Even the most secure data channels sometimes suffer to validate the integrity of images. Forgery of multimedia data is devastating in various important applications like defence and satellite. Increased illegal tampering of images has paved way for research in the area of digital forensics. Copy move forgery is one of the various tampering techniques which is used for manipulating an image’s content. A deep learning–based passive Copy Move Forgery Detection algorithm is proposed that uses a novel dual branch convolutional neural network to classify images as original and forged. The dual branch convolutional neural network extracts multi-scale features by employing different kernel sizes in each branch. Fusion of extracted multi-scale features is then performed to achieve a good accuracy, precision and recall scores. Experiment analysis on MICC F-2000 dataset has been performed under two different kernel size combinations. Extensive result analysis and comparative analysis proves the efficacy of proposed architecture over existing architecture in terms of performance scores, computation time, and complexity.

1 | INTRODUCTION

Social networking has gained a lot of attention since the last one decade. The proliferation of social networking and its ubiquitous presence in both professional and personal life has led to increased transfer of multimedia data, namely, audio, video, images, and documents over insecure wired/wireless networks. With the increased data transfer in our day-to-day life, the illegal operations have also increased. Due to the technological advancement, potential attackers are now better equipped with various tools to illegally copy, copy-move, retouch, manipulate, or distribute digital data. To alleviate illegal operations, various techniques have been developed that encrypt or watermark the data as a line of defence to provide data confidentiality and copyright protection, respectively, [1–3].

The development of various image editing technologies has eased the image manipulating while making it difficult to distinguish between altered and natural images [4]. Image tampering methods primarily include image retouching, image morphing, resampling, splicing copy move forgery, image generation, and colourisation [4–6]. Individually or a combination of all these techniques are generally used to wrongfully alter the image contents and spread misinformation. Image splicing refers to using cut-paste operations to generate a new image by merging portions of two or more images [7], whereas copy move forgery is an image manipulation technique in which portions of a picture are duplicated, that is taken and repasted in some other location within the same image [8]. The region being duplicated may undergo some manipulations, for example, scaling and brightness change before being pasted somewhere else.

Image retouching involves small localised adjustments generally followed by global adjustments like contrast adjustment, brightness control and white balancing, while image inpainting conserves the image by substituting damaged or missing image content in accordance with the surrounding image content [4]. Similarly, colourisation, usually takes grayscale images and colourises them with visually realistic colours, causing discrepancy during specific objects/scenes identification/detection [4].

Such vast usage of image tampering methods has led to the emergence of digital image forensics that is essential to prevent or detect frauds and solve copyright disputes by establishing integrity and authenticity of digital images [9, 10]. The forgery
The next section discusses the existing work in this field. Section 3 discusses the details of dataset, whereas the proposed architecture including the pre-processing part and proposed dual branch CNN network is presented in Section 4. Sections 5 and 6 discuss the experimental and comparative analysis, respectively. The last section presents the conclusion of the proposed work.

2 | RELATED WORK

This section presents the state-of-the-art CMFD techniques. Forgery detection techniques must be highly accurate and reliable. In addition to that, the algorithms must be fast, efficient, robust to a variety of attacks like noise addition, rotation, and scaling and must have low computational complexity [20]. These properties are generally considered while evaluating the efficacy of a CMFD technique.

The CMFD techniques are usually divided into two categories—block-based CMFD techniques and key point–based CMFD techniques. In the block-based methods, the image is divided into overlapping or non-overlapping rectangular or circular patches, followed by extraction of certain features for each patch. Various pre-processing methods like image transforms, colour space transformation, and dimensionality reduction are utilised for feature extraction [21]. The literature review has also revealed several mathematical transforms that are used before the feature extraction step in CMFD technique. Characteristics like the image intensity and texture are also extracted and used to construct the final feature vector. The patches are then sorted using an appropriate algorithm followed by comparison to find out similarity of adjacent blocks [21]. This matching step is the most crucial as it determines the presence of a duplicated region [22].

Alkawaz et al. [23] has used Discrete Cosine Transform (DCT) separately for feature extraction from each block. The coefficients generated are used as the features, followed by lexicographic sorting of the feature vectors [23]. Similarly, Discrete Wavelet Transform (DWT) is another widely used operation since it allows analysis of both time and frequency signals. Jaiprakash et al. [24] used both DCT and DWT to propose a novel low-dimensional feature model in which statistical moments from inter-block differences, pixel correlation and histograms are used for the feature extraction. An ensemble classifier classifies the images as authentic and forged. An improved block-based CMFD method has also been proposed that detects geometric distortions in images by using Discrete Radial Harmonic Fourier Moments for feature extraction [25].

Despite providing good accuracy, block-based methods come at the cost of high computation complexity since each block is processed and depending on the image size, the overhead increases linearly. Hence, key point–based methods were also explored in many research works [26, 27]. Key point–based forgery detection techniques operates on the entire image at once. Mainly two key point descriptors, namely Scale Invariant Feature Transform (SIFT) and the Speeded Up Robust
Features (SURF), have been found to provide good results. SURF is used for key point detection and then GLCM is applied at the key points to obtain co-occurrence matrices [28]. Each matrix is summed up in a column-based manner to obtain the feature descriptors. Wavelet decomposition followed by SURF key point extraction and using an SVM to distinguish forged images from authentic ones is also proposed [29]. Similarly, SIFT-based key points are extracted and compared for scales followed by orientation adjustment for identification of possible forgery blocks in detection of rotational copy move forgery [30].

All the works discussed above rely on machine learning techniques and therefore manually engineered features in most of the cases. This, however, is not conducive in scenarios where the scope of application is very broad and variability in the input data cannot be predicted. Manually designed feature extraction methods suffer from a limitation on the kind and extent of information that can be extracted. For instance, a Local Binary Pattern (LBP) based feature extraction will fail to extract meaningful information from the colour of the image since it will focus only on the textural aspect. This led to the recent spike in the use of deep learning-based architectures for solving such problems.

CNNs have found a way in nearly every image processing-based application in today’s era. CNNs automate the feature extraction process by generating feature maps at every stage of the network. These feature maps extract features by performing convolution operation all over the image and learn weights while being trained on a set of images [31]. The kernels are capable of extracting features which may go amiss by statistical transforms and other mathematical feature extraction techniques. This gives CNN-based architectures an edge over the traditional methods especially in image processing problems [31].

The capability of CNN for CMFD is explored by Abdalla et al. [32], wherein the proposed model was tested on a combination of datasets for both forgery detection and localisation. Features are extracted using a CNN model and later classified using softmax decision function. Analysis indicated that the model was better able to detect active forgery compared to passive one [32]. A combination of classic and deep learning method was also proposed using a dense inception net architecture for learning feature correlations and thereby detecting the forgery [33]. The framework consists of (a) Pyramid Feature Extractor (PFE) to extract multi-scale and multidimensional features, (b) Feature Correlation Matching (FCM) looks for correlation within those dense features for forgery detection and (c) Hierarchical Post-Processing (HPP) modules. The FCM module helped the model to detect the forged regions in completely unseen snippets very efficiently [33]. Similarly, a CNN model is proposed, evaluated and tested for CMFD performance on multiple datasets [34]. The model extracts hierarchical features of an input image, learns those features and uses the information contained in the learned feature maps to classify the image as forged and pristine.

The deep learning methods for CMFD present in the literature either do not give good accuracy, precision, recall scores or have high computation complexity and time to achieve good scores. The existing techniques involve trade-off between time complexity with good parametric values by using too many parameters in the model. To overcome both of these issues, the present paper proposes a dual branch CNN architecture. In the proposed architecture, a deep learning backbone enables deep feature extraction and a dual branch design makes it possible to extract multi-scale features, helping to attain better scores. The proposed architecture is efficient, lightweight and gives good prediction performance. The proposed architecture and its performance analysis have been presented in the subsequent sections.

3 | DATASET

The deep learning frameworks require a large dataset from training and testing of the model. Many such datasets are publicly available for detection of copy move forgery attacks. The present work uses MICC-F2000 [35], which has a total of 2000 images (1300 tampered and 700 original images) from the Columbia photographic image repository [36]. The original image dimensions are 2048 × 1536, wherein the tampered region is constrained to occupy 1.12% pixels of an image size [34, 35].

The forged class is obtained by applying 14 different attacks on each authentic image to generate the tampered images. The dataset is deliberately given a class imbalance to reproduce a practical scenario, where only a fraction of images will be tampered. Therefore, only some of the images are tampered while the rest of them are just present in their original form in the dataset.

The forgeries were generated by selecting a rectangular patch from the image and copy-pasting it in the original image either in original form or after applying different image transformations like translation, scaling (both symmetric and asymmetric) and rotation. Combination of these attacks were also used to generate the forged images. This dataset encompasses a variety of attacks that are widely used to forge images and hence makes it suitable for evaluating the robustness of a CMFD algorithm. Few original images and their forged counterparts from the MICC-F2000 dataset are indicated in Figure 1. The first column in every row represents the original image and the subsequent columns in that row indicates the forged counterparts.

4 | PROPOSED FRAMEWORK

The main objective for the CMFD is to distinguish between an original and the tampered image. For achieving this, the proposed framework is divided into two parts: the first part performs minimal pre-processing and on-the-fly operations, whereas the second part is the modified CNN architecture that extracts the features from these pre-processed images and performs binary classification of images as original or tampered. The basic block diagram of the framework including the proposed architecture is indicated in Figure 2.
The MICC-F2000 comprises images of size $2048 \times 1536$. Images of this size increase the computational complexity and the model takes more time to converge. Through various transforms, feature extraction and dimensionality reduction methods for CMFD, it has been proved that image size is not the foremost factor affecting quality of predictions. The collective characteristics of a pixel group are seen to have more significance as compared to individual pixel characteristics. Thus, the images are reduced to a fixed size of $700 \times 700$ to make the computation feasible without affecting the image features or characteristics. The resized images are then standardised on-the-fly before giving it as an input to the proposed CNN-based architecture.

The proposed architecture is dual branch CNN-based architecture, where both the branches are connected to a common input. There are three convolution layers in each branch, with 16, 32 and 64 feature maps for the first, second and third layer, respectively. All the convolutional layers uses ReLU activation and each convolutional layer is followed by a $2 \times 2$ max-pooling layer. To extract multi-scale features from the images, CNN layers in these two branches have different kernel size.

Since experiments were conducted by varying the kernel sizes, hence in some cases the addition of one zero-padding layer has been done to ensure a symmetric output. The output of the third convolution layer from both the branches is passed through a concatenation layer. This generates a stack of
multi-scale feature maps extracted from a common input. The concatenated output of this layer is fed to a global max-pooling layer, which retains only maximum feature per feature map. This layer acts as a flattening layer in the architecture and converts the two-dimensional input to a one-dimensional output.

This 128 length one-dimensional vector is passed into the second last layer which is a dense layer with 32 units. This 32 length vector is fed to the last dense layer with a single unit only. Sigmoid activation has been used in both of the dense layers.

The last layer generates the class probability ‘p’ that denotes the image being authentic. Hence, ‘1-p’ will be the probability of the image being forged. A decision threshold of 0.5 is used to classify between an original and a forged image. An output probability of greater than 0.5 indicates an original image and otherwise a forged image. In binary labels, ‘1’ denotes an original image and ‘0’ denotes a forged image.

5 | EXPERIMENTAL ANALYSIS

The proposed model architecture was implemented in Python using Keras as the backend library. All the stated experiments were performed on an Intel Core i5 8th Gen processor having 24GB system RAM and an NVIDIA GeForce GTX 1050Ti 4GB RAM graphics card.

The dataset was randomly split into train, test and validation sets before model training. The validation set was provided as an input to the model at every epoch. This made it possible to monitor the model’s performance on unseen data at every epoch. The final results were obtained on the test set. Total number of epochs was set to 100. The training process was monitored for improvements in validation loss for overlapping intervals of 20 epochs. In the absence of any improvement over this interval, training was automatically set to stop. This is called early stopping, that is, stop the training if model is not improving. The parameter ‘validation loss’ was monitored because that gives a better idea of the model’s prediction over unseen data.

Since same validation accuracy can lead to different validation losses, hence accuracy is not monitored. The objective was to look for the best version of the model. For training, the learning rate was set to 0.0001 and batch size to 5. During testing, the batch size was set to 1. Adam was used as the optimiser for the binary cross entropy loss function used in the last layer.

For a thorough analysis, data was divided into the ratio of 85:15 as train, test-validation split ratios. The 15% ratio is further divided equally as testing and validation set, having 7.5% of the total images in each set. The entire 2000 images present in dataset resulted into 1700 images in the training set, 150 images in validation set and 150 images in test set. Model performance was carefully monitored and evaluated for various parameters including prediction accuracy.

Size of the kernel directly determines the receptive field of the network. Large sized kernels can overlook finer details and skip essential information; on the contrary, very small sized kernels can provide too much information which can sometimes be misleading. In detection of copy move forgery attacks, lot of methods uses block-based approaches [21, 37] and look for similarities between the feature vectors generated by each block. The block size in that scenario is analogous to the receptive field of the CNN. It is observed that in most of the block-based approaches, the size of the block is never too small.

Two different combinations of kernel sizes were experimented with (a) $3 \times 3$ and $5 \times 5$, that is, (3,5) and (b) $5 \times 5$ and $8 \times 8$, that is, (5,8). The former combination depicts that the first branch uses kernel size $3 \times 3$ while the second branch uses $5 \times 5$. Similarly, the latter combination depicts that the first branch uses kernel size $5 \times 5$ while the second branch uses $8 \times 8$. The performance of the proposed architecture for these
combinations was carefully observed through metrics like precision, recall, F1 score, sensitivity, specificity, True Positive Rate (TPR) and False Positive Rate (FPR). A representation of the widely used parameters is as follows:

\[
\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \text{Recall},
\]
\[
\text{Specificity} = 1 - \frac{\text{FP}}{\text{TN} + \text{FP}},
\]
\[
\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}},
\]
\[
\text{F1 Score} = \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}},
\]

where TP is True Positives, FP is False Positives, TN is True Negatives, FN is False Negatives, TPR is True Positive Rate and FPR is False Positive Rate.

The values were calculated by treating forged as the positive class and original as the negative class. Therefore, true positives or the positive samples represent the forged class, whereas true negatives or the negative samples represent the original class. Table 1 summarises the obtained results. The obtained values indicate a good accuracy score of 0.96 for both the combinations, specificity of 0.93, precision of 0.89 with a perfect sensitivity and recall score of 1. The difference in the performance of these combinations can be seen in mean ROC-AUC score and mean precision-recall area under curve (AUC) score. The quantitative values obtained for both these scores indicate that the combination of (5,8) performs better than the combination of (3,5).

The performance of proposed architecture is also thoroughly analysed from training-validation accuracy and loss plots. These plots have been indicated in Figure 3 and these clearly depict the training quality of the model. For a model that is neither under-fitting nor over-fitting, training and validation curves closely follow each other. This signifies that with every progressing step the model is maintaining its generalisability on unseen data well.

Four plots were obtained over the training epochs for both the kernel combinations. Each graph has two curves corresponding to the two different kernel size combinations. The performance measures were obtained for both the training and validation sets in order to gauge the learning and generalisation ability of the model all at once. It can be seen that the accuracy curves rise and then saturate at a point of best performance. Similarly, the loss plots attain a minimum loss point and then saturate with a few spikes on and often. An important observation to be made here is the epoch at which the model converges for the two kernel combinations. For (3,8) combination, the model attain a point of best performance faster than the (3,5) combination. This clearly shows that a larger filter size is more suitable for the current image size of 700 × 700.

Two kinds of diagnostic curves were plotted for both the forged and authentic classes corresponding to both the train test split ratios—receiver operating characteristics (ROC) curves and precision-recall curves. ROC curve is a way to visualise the discrimination ability of a binary classifier as its decision threshold is varied. This curve plots the TPR versus FPR by varying the decision threshold.

ROC curves and the associated AUC depicts the diagnostic ability of the classifier at different thresholds. Each obtained ROC plots (Figure 4) has two curves corresponding to the two combinations of kernel sizes. The AUC is consistently higher for both the forged and original classes corresponding to the kernel sizes (5,8). This may be due to the larger receptive area of 8 × 8 kernel as compared to 3 × 3 kernel (5 × 5 kernel size being common in both the combinations).

Precision-recall curves are more suitable for representing the differences across classes when there is a class imbalance. In this case, the forged class is in minority while the original class is in majority. Therefore, the precision-recall curves are also plotted in a similar manner. This curve plots precision and recall values by varying the decision threshold. The obtained precision-recall curves for original and forged class are indicated in Figure 5. It can be observed that the kernel combination of (5,8) is evidently outperforming the (3,5) combination for the forged class. However, no such difference can be observed in the curve for the original class.

Performance of the proposed architecture is also analysed using the bar graph that compares minimum training and validation loss obtained for each kernel combination. It can be seen in Figure 6 that the minimum training loss for (3,5) combination is slightly less than that of (5,8). Despite better performance on the training set, this model was not able to generalise well on the unseen or validation set. The (5,8) combination obtains lower loss on the validation set. It is always desirable to have a model that performs well on both seen and unseen data. In this proposed architecture, it is observed that the (5,8) combination is superior as compared to (3,5) and this has been verified through various parametric values and graphs.

It can be inferred from here that the combination of (5,8) avoids noise by capturing relatively coarser details instead of finer details captured by the (3,5) counterpart. This makes the convolution operation a patch wise processing operation wherein the kernels are learning weights for every overlapping patch. Therefore, the kernels are able to capture all of the forged region at once. On the contrary, the 3 × 3 sized kernel being too

| Kernel | 3×3 and 5×5 | 5×5 and 8×8 |
|--------|-------------|-------------|
| Accuracy | 0.96 | 0.96 |
| Sensitivity | 1 | 1 |
| Specificity | 0.93 | 0.93 |
| Precision | 0.89 | 0.89 |
| Recall | 1 | 1 |
| F1 score | 0.94 | 0.94 |
| Mean ROC-AUC | 0.94 | 0.95 |
| Mean Precision-Recall-AUC | 0.87 | 0.895 |
FIGURE 3  Accuracy and loss plots: (a) training accuracy, (b) training loss, (c) validation accuracy, (d) validation loss

FIGURE 4  ROC curves
small, is probably failing to capture the region in its entirety and instead ends up capturing a lot of detailed information which is ultimately not helpful in the decision-making process.

6 COMPARATIVE ANALYSIS

This section presents a thorough comparative analysis of the objective and subjective aspects of some of the related work. Since one of the objectives of this analysis was to emphasise on the suitability of using deep learning architectures for CMFD, hence both kinds of works, that is, handcrafted feature based and deep learning feature based, have been analysed and critically evaluated. Some of the quoted works have an extended pipeline which also involves forged region localisation. However, the proposed work is only solving the detection problem and therefore the scope of comparison is limited to the stage of classification. For every literature deriving results on the MICC-F2000 dataset, only those results are compared, whereas for all the other cases, the best case result is quoted. Table 2 summarises the work and quantitative results of the literature being analysed.

For authentication of images, CMFD technique is proposed that extracts low-dimensional DWT and DCT based features [24]. The author proposed the use of ensemble classifier for classifying between the authentic and the forged images. A comparison with the present work is indicated in Table 2. It can be seen that the time taken in [24] is quite less which may be due to the fact that the analysis has been done on varying image sizes of $240 \times 60 - 900 \times 600$ while the proposed work is working on a constant image size of $700 \times 700$. Also, the dimension wise proportion has not been specified by Jaiprakash et al. [24]. It can be deduced that some portion of the test set images must have had dimensions significantly smaller than $700 \times 700$. For instance, a $240 \times 60$ image is only $0.03\%$ of the size of a $700 \times 700$ pixels. In addition to this, only two channels of the image, that is, $C_b + C_r$, of a YCbCr image were used as input. This further reduces the amount of data to be processed. The smaller image size handled and the lesser number of channels used have led to smaller time taken in [24].

A CNN-based architecture is proposed in [32] for deep feature-based CMFD wherein the very first layer takes an input of arbitrary size, resizes it to a $64 \times 64$ pixels and then feeds the input to the CNN. The dataset used in this work was curated by using a combination of publicly available datasets and images readily available online. A total of 1254 samples were used for training while 537 were used for testing. The CNN network uses a softmax layer to output the class probabilities. Accuracy of $90\%$, precision and recall scores of $69.63\%$ and $80.42\%$ were obtained, respectively. This method was more successful in detecting active forgery compared to passive one, as stated in the literature. Further, the minimum testing time for one image comes out to be $0.8$ seconds which is nearly $8$ times the prediction time of the proposed method, which is $0.1$ seconds. Compared to our work, their method differs in three major aspects—lack of multi-scale features, the use of average pooling layers in the CNN and smaller training set. Also, the images are being...
downsized significantly by the first resizing layer. This might be leading to a drastic loss of information thereby deteriorating the model’s performance.

The single branch CNN architecture designed by Elaskily et al. [34] is deeper and heavier than the proposed model. It uses convolution layers with 16, 32, 64, 128, 256 and 512 feature maps, whereas our model uses only 16, 32 and 64 feature maps in the convolution layers with maximum of 64 feature maps in the last convolution layer in both the branches. Using a very large number of feature maps to improve performance comes at the cost of a drastic increase in the model’s weights and parameters. This directly impacts the prediction time and makes it unsuitable for real time implementation. Even though the model in [34] achieves near perfect scores, it should be noted that it has an input image size of only (224 × 224), approximately three times smaller than the image size used in proposed architecture (700 × 700). Despite a bigger image size, the proposed model has a prediction time of 0.1 seconds as opposed to achieved 0.5 seconds by authors in [34].

A novel method namely Convolutional Kernel Network was proposed which is robust against a variety of attacks including geometric transformations [38]. The model was trained on the Rome Patches dataset. The entire training process was computationally very expensive. The predictions were obtained on various datasets. However, for comparison purposes, only the results obtained on MICC-F2000 dataset are included. A perfect sensitivity score of 100% was obtained but the specificity score in this method is only 91.2% which is lower than that of the proposed model (93%). This implies that even though the model is able to identify forged images correctly, it is not able to identify all the authentic images correctly and misclassifies an authentic image as a forged one roughly 8 out of 100 times. In comparison, our model has both high specificity and sensitivity scores. A clear comparison on various other metrics for the proposed architecture and state-of-the-art work is indicated in Table 2.

### 7 | CONCLUSION

The present work proposes a novel dual branch CNN architecture for detection of copy move forgery attacks in digital images. The proposed architecture has two branches that implements different sized kernels for feature extraction. With the well-established CNN architecture as backbone, the dual branches ensure extraction of multi-scale features, which are then fused together to take the dominant feature for binary classification of images as forged or original. Extensive experimental and performance analysis has been done for the two different kernel size combinations. Thorough comparative analysis with state-of-the-art work indicates that the proposed architecture is lightweight and can achieve good prediction accuracy. In the future, the architecture can be tested and evaluated on more datasets with varying image sizes, constituting different kinds of forgeries to test the robustness of the model.

### CONFLICT OF INTEREST

There is no conflict of interest.

### DATA AVAILABILITY STATEMENT

Data openly available in a public repository that does not issue DOIs. The data that support the findings of this study are openly available at http://lci.micc.unifi.it/labdl/2015/01/copy-move-forgery-detection-and-localization/.

### ORCID

Nidhi Goel 🌐 https://orcid.org/0000-0001-7089-7077
REFERENCES

1. Molina, J., et al.: Watermarking framework for authentication and self-recovery of tampered colour images. IEEE Lat. Am. Trans. 18, 631–638 (2020)
2. Sarveshthari, S., Akhaee, M.: A source-channel coding approach to digital image protection and self-recovery. IEEE Trans. Image Process. 24, 2266–2277 (2015)
3. Goel, N., Raman, B., Gupta, I.: Chaos based joint compression and encryption framework for end-to-end communication systems. Adv. Multimedia 2014 (2014). https://doi.org/10.1155/2014/910106
4. Qazi, T., et al.: Survey on blind image forgery detection. IET Image Process. 7(7), 660–670 (2013)
5. Elaskily, M.A., et al.: Comparative study of copy-move forgery detection techniques. International Conference on Advanced Control Circuits Systems (ACCS) Systems and International Conference on New Paradigms in Electronics Information Technology (PEIT), November 2017, pp. 193–203. IEEE, Piscataway, NJ (2017)
6. Abdin, A.B.Z., et al.: Copy-move image forgery detection using deep learning methods: A review. 6th International Conference on Research and Innovation in Information Systems (ICRIIS), December 2019, pp. 1–6. IEEE, Piscataway, NJ (2019)
7. Xiao, B., et al.: Image splitting forgery detection combining coarse to refined convolutional neural network and adaptive clustering. Inf. Sci. 511, 172–191 (2020)
8. Christlein, V., et al.: An evaluation of popular copy-move forgery detection approaches. IEEE Trans. Inf. Forensics Secur. 7(6), 1841–1854 (2012)
9. Mahmood, T., et al.: Copy-move forgery detection technique for forensic analysis in digital images. Math. Prob. Eng. 6, 1–13 (2016)
10. Taneja, N., et al.: Selective image encryption in fractional wavelet domain. AEU Int. J. Electron. Commun. 65(4), 338–344 (2011)
11. Huynh, T.K., et al.: A survey on image forgery detection techniques. The 2015 IEEE RIVF International Conference on Computing Communication Technologies - Research, Innovation, and Vision for Future (RIVF), January 2015, pp. 71–76. IEEE, Piscataway, NJ (2015)
12. Panda, S., Mishra, M.: Passive techniques of digital image forgery detection: Developments and challenges. In: Kalam, A., Das, S., Sharma, K. (eds.) Advances in Electronics, Communication and Computing, pp. 281–290. Springer, Berlin (2018)
13. Huang, D.Y., et al.: Robustness of copy-move forgery detection under high jpeg compression artifacts. Multimedia Tools Appl. 76, 1509–1530 (2015)
14. Costanzeo, A., et al.: Forensic analysis of sift keypoint removal and injection. IEEE Trans. Inf. Forensics Secur. 9, 1450–1464 (2014)
15. Pandey, R., et al.: Passive copy move forgery detection using surf, hog and sift features. Adv. Intell. Syst. Comput. 327, 659–666 (2014)
16. Zhang, Y., et al.: Image region forgery detection: A deep learning approach. In: Mathur, A., Roychoudhury, A. (eds.) Proceedings of the Singapore Cyber-Security Conference, pp.1–11. IOS Press, Amsterdam (2016)
17. Ouyang, J., et al.: Copy-move forgery detection based on deep learning. 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), pp. 1–5. IEEE, Piscataway, NJ (2017)
18. Wu, Y., et al.: Deep matching and validation network: An end-to-end solution to constrained image splicing localization and detection. Proceedings of the 25th ACM International Conference on Multimedia. MM’17, pp. 1480–1502. Association for Computing Machinery, New York, NY (2017)
19. Zhou, P., et al.: Learning rich features for image manipulation detection. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1053–1061. IEEE, Piscataway, NJ (2018)
20. Teerakanok, S., Uehara, T.: Copy-move forgery detection: A state-of-the-art technical review and analysis. IEEE Access 7, 40550–40658 (2019)
21. Soni, B., et al.: CMFD: A detailed review of block based and key feature based techniques in image copy-move forgery detection. IET Image Process. 12(2), 167–178 (2018)
22. Diane, W.N.N., et al.: A survey of partition-based techniques for copy-move forgery detection. Sci. World J. 2014, 1–13 (2014)
23. Alkawaz, M., et al.: Detection of copy-move image forgery based on discrete cosine transform. Neural Comput. Appl. 30, 183–192 (2016)
24. Jaiprakash, S.P., et al.: Low dimensional DCT and DWT feature based model for detection of image splicing and copy-move forgery. Multimedia Tools Appl. 79, 29977–30005 (2020)
25. Zhong, J., et al.: A new block-based method for copy move forgery detection under image geometric transforms. Multimedia Tools Appl. 76, 14887–14903 (2016)
26. Li, Y., Zhou, J.: Fast and effective image copy-move forgery detection via hierarchical feature point matching. IEEE Trans. Inf. Forensics Secur. 14(5), 1307–1322 (2019)
27. Wurth, A.D., et al.: A survey on keypoint based copy-paste forgery detection techniques. Procedia Comput. Sci. 78(C), 61–67 (2016)
28. Teerakanok, S., Uehara, T.: Copy-move forgery detection using glem-based rotation-invariant feature: A preliminary research. 42nd Annual Computer Software and Applications Conference (COMPSAC), July 2018, pp. 365–369. IEEE, Piscataway, NJ (2018)
29. Dhiyva, S., et al.: Copy-move forgery detection using surf feature extraction and SVM supervised learning technique. Soft Comput. 24, 14429–14440 (2020)
30. Chen, C., et al.: Rotational copy-move forgery detection using sift and region growing strategies. Multimedia Tools Appl. 78, 18293–18308 (2019)
31. LeCun, Y.H.P., et al.: Object recognition with gradient-based learning. In: Forsyth, D.A., Mundy, J.L., Gesu, V.D., Cipolla, R. (eds.) Shape, Contour and Grouping in Computer Vision, pp. 319–345. Springer, Berlin (1999)
32. Abdalla, Y., et al.: Convolutional neural network for copy-move forgery detection. Symmetry 11(10), 1280 (2019)
33. Zhong, J., Pan, C.: An end-to-end dense-inceptionnet for image copy-move forgery detection. IEEE Trans. Inf. Forensics Secur. 15, 2134–2146 (2020)
34. Elaskily, M.A., et al.: A novel deep learning framework for copy-move forgery detection in image. Multimedia Tools Appl. 79, 19167–19192 (2020)
35. Amerini, I., et al.: A sift-based forensic method for copy-move attack detection and transformation recovery. IEEE Trans. Inf. Forensics Secur. 6(3), 1099–1110 (2011)
36. Ng, T.T., et al.: Columbia photographic images and photorealistic computer graphics dataset. ADVENT Technical Report #205-2004-5, Columbia University (2004)
37. Yu, S.R.N., Zhao, Y.: Nonoverlapping blocks based copy move forgery detection. IEEE Trans. Inf. Forensics Secur. 14(5), 1307–1322 (2019)
38. Liu, Y., et al.: Copy-move forgery detection based on convolutional kernel network. Multimedia Tools Appl. 77, 18269–18293 (2017)

How to cite this article: Goel N, Kaur S, Bala R. Dual branch convolutional neural network for copy move forgery detection. IET Image Process. 2021;15:656–665. https://doi.org/10.1049/ipr2.12051