Signal, Image and Video Processing
https://doi.org/10.1007/s11760-021-01895-5

ORIGINAL PAPER

# Automated image splicing detection using deep CNN-learned features and ANN-based classifier

Souradip Nath[1] · Ruchira Naskar[1]

Received: 15 August 2020 / Revised: 1 February 2021 / Accepted: 15 March 2021

**Abstract**
With the present-day rapid evolution of digital technology, images have become one of the most important means of communication and information carrier in our society. Since the last decade, with the emergence of social networking sites like Facebook, Instagram, Twitter, etc., there has been a huge increase in the amount of information exchanged in the form of digital images, on a regular basis. While traditionally we might have had faith in the credibility of these images, today's digital technology has begun to erode that faith. Before sharing an image over social networks, editing it with relevant software application has become one of the simplest things to do today. While not many people do this with any sinister intent behind, there has been a significant increase in cybercrimes related to malicious image manipulation and sharing. To this end, image splicing has emerged as one of the major forms of image manipulation attacks, among others today. This demands investigation of intrinsic differences between authentic and forged images and hence development of automated splicing detection tools. Here, we propose a blind image splicing detection technique that employs a deep convolutional residual network architecture as a backbone, followed by a fully connected classifier network, that classifies between authentic and spliced images. The classifier networks have been evaluated using the CASIA v2.0 dataset. Both are proven to yield accuracies more than 96% on an average, having surpassed the state-of-the-art results.

**Keywords** Convolutional neural networks · Deep learning · Digital forensics · Image forensics · Image splicing · ResNet-50

## 1 Introduction

With the advancement of digital technology in the current information age, digital image tampering has become an easy nut to crack for any common man. Such development in the image manipulation techniques has both constructive and destructive aspects, particularly concerning digital content security. On the one hand, it promotes showcasing ideas of visual art and beautifying existing images, whereas, on the other hand, it makes tampering image contents extremely easy for adversaries, in a visually imperceptible way. In recent years, the vast availability of low-cost sophisticated photo editing tools has largely triggered the production of natural-looking, aesthetic images, almost leaving no visible clue of tampering. As a result, anyone with a computer, mobile, tablet, or laptop can easily manipulate the contents of an image to spread fake information. This threatens the authenticity of digital images in broadcast, media, medicine, and legal industries. In the domain of digital forensics, the major types of image manipulation attacks that we deal with in the present day are: image splicing [1], copy-move forgery [2], retouching [3], and erase-fill [4]. If the attack involves replacing some content/object within a digital image with some other content/object, taken from a different location within the same image, then the attack is termed as *copy-move forgery*. On the other hand, attacks where some new content from an external source is intelligently added to an authentic image with the malicious intention of posing this composite image as a natural one, and with visually imperceptible traces of the manipulation, are termed as *splicing* or *compositing* attack. Figure 1, sampled from CASIA v2.0, illustrates one of these common types of image manipulation: the *copy-move forgery*, whereas Fig. 2 presents an example of *image splicing*.

✉ Souradip Nath
souradipnath.ug2018@it.iiests.ac.in

Ruchira Naskar
ruchira@it.iiests.ac.in

[1] Department of Information Technology, Indian Institute of Engineering Science and Technology, Shibpur, Shibpur 711103, India

Published online: 03 April 2021

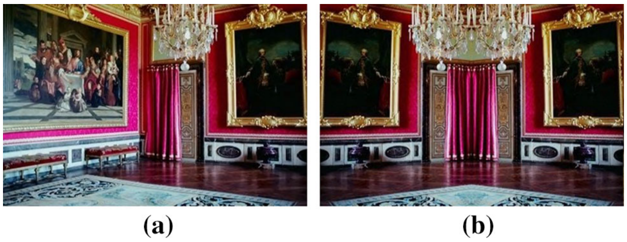

**Fig. 1** An example of *copy-move forgery*: **a** Authentic image; **b** Forged image generated by copying the right half of the authentic image, flipping it horizontally, and then pasting it to cover the left half of the same image

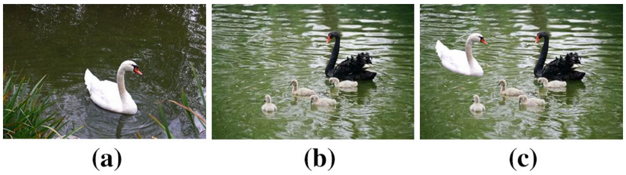

**Fig. 2** An example of *image splicing*: **a** Authentic image 1; **b** Authentic image 2; **c** Forged image generated by compositing images 1 and 2, after required re-scaling of the copied object

In this work, we deal with automated detection of image splicing attack, using deep neural network architecture for automated feature learning. Image splicing detection has major applications in the forensic industry. Most significant application domains include legal, media and broadcast industries. Given how low-cost and widely available image processing software and tools have become presently, it has become extremely challenging and crucial to authenticate a given image before it is considered for further processing in sensitive applications such as the above. More so, given the fact that digital images play the role of primary evidences in many legal cases, as well as media. Also, given the large boom of social networks in the recent days, and the large volumes of images and videos shared online on a day-to-day basis by every common man, forged images (including spliced images) can lead to deception and delusion of masses. Hence, it is the call of the hour to have automated tools in place, to detect such common forgeries with highest accuracy.

Till the dawn of the deep learning period, image classification tasks have mostly been performed using hand-crafted image features. Such state-of-the-art forensic imaging techniques primarily consist of *(a) pixel-based methods, (b) geometry-based methods, (c) camera-based methods, (d) format-based methods, and (e) physics-based methods.* [5]. *Pixel-based image forensic techniques* detect anomalies focusing solely on the spectral information in each pixel [6]. *Geometry-based methods* determine measurements of objects in the universe and their locations relative to the camera [7]. *Camera-based methods* are majorly based on color camera response, filter array, chromatic aberration, and sensor noise Camera-based techniques [8]. *Format-based techniques* exploit the statistical correlations formed by a particular lossy compression scheme [9]. *Physics-based methods* explicitly model anomalies by visualizing physical object interaction with the environment [10]. All such techniques, primarily exploit the visual information content of an image that *might not* be robust against subsequent processing, such as compression, deform, edge softening, blurring, and smoothing.

With the advent of convolutional neural networks (CNNs), automated feature engineering has been extensively explored in all spheres of machine learning. This has impacted the domain of image forensics as well. In recent years, in comparison with the aforementioned conventional approaches to image forgery detection, a number of researchers have successfully investigated the applicability and utility of CNNs in the field of image forensics. Some of the noteworthy works in the domain of image forgery detection using CNNs include [11,12].

In this paper, we address the issue of image splicing detection using *convolutional neural networks*, where we model the given challenge as a *two-way classification* problem. We break the given problem into smaller sub-problems, viz. (A) dataset preparation for feature extraction and classification using the extracted features and (B) finally detecting the spliced and the authentic images. Here we use two different networks for two major tasks above. Initially, state-of-the-art *ResNet-50* network [13] has been used as the backbone of our proposed model, which majorly allows feature extraction from the raw input images. The backbone is followed by a fully connected binary classifier network which is trained to detect whether the image is authentic or spliced. To summarize, our major contributions in this paper are as follows:

- We propose a deep learning CNN model for image splicing detection, which mainly consists of a backbone network for extraction of features, followed by a ANN-based binary classifier to detect spliced images.
- The proposed scheme saves us from the tedious job of handpicking image features and automatically generates a deep-learned representation of the features from unprocessed input images.
- For the classification problem, we propose the use of a fully connected artificial neural network in place of traditional classifiers such as the Support Vector Machine.

The proposed model has been thoroughly analyzed in terms of performance and compared with the current state of the art.

The rest of the paper is structured according to this. In Sect. 2, we outline the existing related literature. In Sect. 3, we present and describe in detail our proposed model along with dataset and parameter selection. Section 4 sums up our experimental analysis and results and discusses those in detail. Finally, the conclusions as well as the directions for future research are described in Sect. 5.

## 2 Related works

In this section, we present an overview of the significant research works related to image forensics and image splicing detection in particular. Most of the works listed here focus on image feature based machine learning approach for image forgery detection.

Zhang et al. [14] proposed a model which focuses on *moment features* extracted from *multi-size block discrete cosine transform* (MBDCT) along with *image quality metrics* (IQMs), which are purposed toward spliced image detection. This model analyses an original image and its tampered version to measure their statistical differences. The authors performed their experiments on the Columbia Dataset [15] and attained an accuracy of 89.16%.

Muhammad et al. [16] explored the *steerable pyramid transform* (SPT) and *local binary pattern* (LBP) to present their scheme for image forgery detection. They basically applied the steerable pyramid transform on the Cb and Cr channels of the YCbCr image color space, followed by a description of the texture in each SPT subband using LBP histograms, to produce a feature vector. Finally, they utilize support vector machine (SVM) as a classifier which detects image forgery based on the feature vector. The authors reported an accuracy of 97.33% on the CASIA v2.0 dataset [17].

Pham et al. [18] proposed an efficient algorithm for image splicing detection based on Markov features that prepares a feature vector combining two types of Markov features, coefficient-wise and block-wise in the discrete cosine transform (DCT) domain extracted by the model. This is followed by a support vector machine (SVM) which takes the feature vector and predicts a query image to be original or forged. They used CASIA v2.0 as their dataset and achieved an accuracy of 96.90%.

Rao and Ni [19] presented a deep learning-based novel image splicing detection scheme which employs a convolutional neural network (CNN). Here the CNN is utilized to extract *hierarchical representations* from the input RGB color images. The pre-trained CNN majorly extracts dense features from the input images as a patch descriptor. For the SVM classification, the final discriminative features is obtained by exploring a feature fusion technique. The proposed work achieves an accuracy of 97.83% on CASIA v2.0 Dataset.

Pomari et al. [20] proposed a novel approach for detecting splicing in digital photographs by putting together the *high representational capacity of illuminant maps* and convolutional neural networks as a way to learn the most important traces of a forgery, directly from the available training data. Their work proposes a method that eliminates the tedious process of feature engineering, allows to identify forged regions within an image, and is proven to yield a classification accuracy of more than 96%.

Tripathi et al. [21] use a methodology which extracts texture-based features from the forged and authentic images as *gray level run length matrix*, computed in four directions. Thereafter, a fuzzy support vector machine has been utilized as the classifier, with a high generalization capability. This method is trained and tested on CASIA v1.0 Dataset [17] and the final performance, in terms of F1 score, is reported to be 0.89.

Wu et al. [11] proposed an extension to the image splicing detection problem where they have worked with two different images and estimated the chance of one image being tampered using the other. They also introduced a deep neural network for the detection and localization of image splicing, called *deep matching and validation network* (DMVN). They used CASIA v2.0 and the Nimble 2017 [22] datasets in their experiments. This scheme produces precision and recall results of 94.15% and 79.08%, respectively.

In [12], Ahmed et al. proposed a new image forgery detection scheme which uses ResNet-Conv model [13] as the backbone of the proposed network architecture. This backbone majorly focuses on generating the initial feature map, which is then used to train a *Mask-RCNN* model to generate masks for spliced region in the forged images. They performed a comparative study by experimenting with different initialization techniques and different backbone architectures. Finally, training and evaluation of the proposed model is done on a huge dataset developed from COCO dataset. Their scheme yields an AUC value of 0.967.

Traditional feature-based image forgery detection schemes have used image features such as color and texture features like SPT and LBP and frequency domain features like DCT and DWT, with support vector machine and other classifiers. Apart from the traditional approaches, with the advent of deep learning, researches have successfully explored its applicability and usefulness in the domain. In our work, we propose a deep learning-based CNN model for feature extraction and image splicing detection, which has been described in detail next.

## 3 Proposed deep CNN-based image splicing detection model

In this work, we model image splicing detection as a binary classification problem, based on deciding whether or not a given image is spliced. In this paper, we propose an automated image splicing detection scheme using

**Table 1** Dimension of datasets used in our experiments

| Dataset | # Authentic samples | # Spliced samples | # Total samples |
|---|---|---|---|
| Original CASIA v2.0 | 7491 | 5123 | 12614 |
| Sampled CASIA v2.0 | 5123 | 5123 | 10246 |

deep CNN-learned features and ANN-based classifier. The proposed approach involves two main steps: (A) *feature engineering* and (B) *classification*. Initially, a pre-trained deep CNN model (referred here as the *backbone CNN*) is used to extract significant image features with respect to the given problem. Following this, the feature vectors produced by the backbone CNN are fed to a binary classification network, which is trained on labeled image samples to identify the query images as original or spliced.

The proposed method is elaborated in detail in this section, along with experiments conducted to assess the effectiveness of the proposed model in detecting spliced images. Our model is trained on CASIA v2.0 dataset following the current state of the art, as CASIA v2.0 is considered as a very standard and versatile dataset for image splicing detection, that consists of numerous examples of copy-move forgery and image splicing.

### 3.1 Dataset description

As stated above, we have conducted our experiments on the benchmark image forgery detection dataset, viz., CASIA v2.0 [17]. This dataset is a collection of 12,614 color images, out of which there are 7,491 authentic and the rest 5,123 are forged. All of these forged images have gone through different post-processing techniques (such as, *resize*, *deform*, and *blurring*) which give them a more realistic visual appearance. It was originally published for use in research concerning both the problems of detecting image splicing and copy-move forgery. Images comprising the dataset depict sizes range from 240 × 160 to 900 × 600 pixels. There are compressed images in JPEG format along with uncompressed ones in TIFF format.

In order to keep the comparison fair with the other researchers, we initially take the entire dataset of 12,614 images and split it into three mutually exclusive subsets to create the training, validation, and test datasets in the ratio 80:10:10. However, it is worth mentioning that in the original dataset, the approximate ratio of authentic images to spliced images is about 3:2, i.e., during training, the model gets to come across an authentic image 1.5 times more than a spliced image. In order to avoid the unwanted classifier bias toward the authentic images, in our second experiment, we randomly select equal numbers of authentic and tampered images for our experiments and create a sampled dataset of 10,246 images and divide them into three non-overlapping subsets: Training, Validation, and Test. Table 1 sums up the dimensions of the datasets used in our experiments.

### 3.2 Proposed CNN architecture

It is evident that detecting a spliced image through bare eyes might be tough, but locating the forged areas of a given spliced image by the human visual cortex is not impossible. Convolutional neural nets are inspired by the principal of operation of the human visual cortex. CNN is therefore the ideal deep learning model for this work. The model proposed in this paper, uses a convolutional neural network as its backbone, followed by a *dense classifier network*. Figure 3 depicts the proposed model architecture.

The backbone of the proposed network is based on the existing implementation of ResNet-50 architecture [13]. ResNet-50 considers input images of both length and width as multiples of 32, and 3 as the width of the channel. The network uses 7 × 7 and 3 × 3 kernels to perform the initial convolution and max-pooling, respectively. ResNet-50 then contains five different stages [12]. The two foundational components of the ResNet-50 are *convolutional blocks* and *identity blocks*. The structure of these two components is similar, comprising of convolutional layers with batch normalization and activation functions. The only difference lies in the extra bridge in the convolutional blocks, which adds residuals to the output layer learned in the input layer. Stages 2–4 includes both of these blocks, while stage 1 has only identity blocks. Moving from one stage to another doubles the width of the channel and reduces the input size to half. The network finally has an average pooling layer at stage 5 followed by a fully connected layer having 1000 output neurons.

We excluded the top layer to customize the ResNet-50 network according to our specification by setting include_top as False and added our classifier network to it because our images can be classified only into two categories: *Authentic* and *Spliced*, while the ImageNet weights work for 1000 output categories.

Our classifier network contains two fully connected dense layers each having 1000 neurons with a 'ReLU' activation function followed by a 'sigmoid' output neuron. Our experiments are conducted both on the original CASIA v2.0 dataset and the sampled CASIA v2.0 dataset, described in Sect. 3.1.

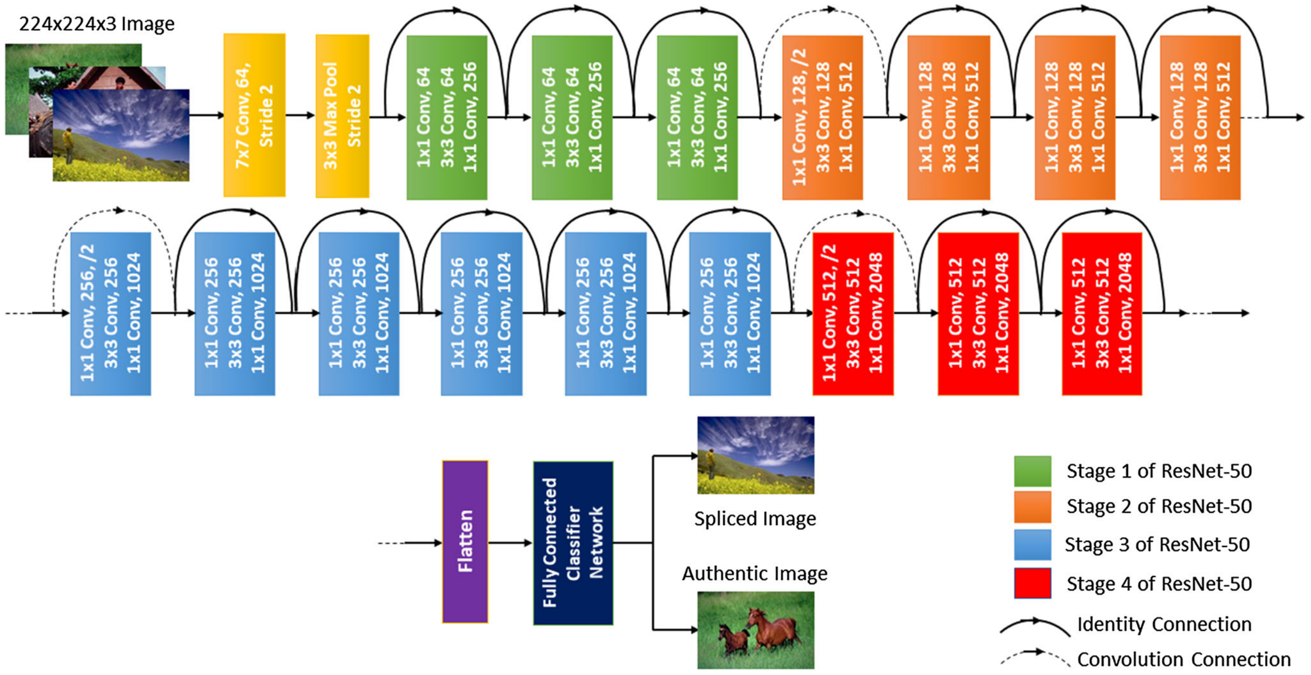

**Fig. 3** The proposed model

In this work, the proposed model is provided with input images of dimension $224 \times 224 \times 3$, which produces a binary output based on the two classes: *Authentic* and *Spliced*.

### 3.3 Implementation details and platform used

The implementation of the proposed method is constituted of three steps: the dataset formulation, training of the model, and finally the classification. Figure 4 presents the flow diagram of the proposed method. The proposed model has been implemented using Keras and evaluated on an Intel(R) Core(TM) i5-1035G4 CPU @1.10GHz 1.50GHz.

Initially, using the `ImageDataGenerator(·)` function from the Keras preprocessing library, the datasets are prepared as mentioned in Table 1. A batch size of 20 is used for mini-batch training.

Our model uses the ResNet-50 model to extract the initial image features meaningful to the classification. This is based on the existing implementation of ResNet-50. In order to build our desired classification network, we exclude the fully connected layers on top of the ResNet-50 architecture. The proposed model was initialized using ImageNet weights. We couldn't apply the same for the output layers because our images can be classified only into two categories: *Authentic* and *Spliced*, while the ImageNet weights work for 1000 output categories. For the optimization of the proposed model, we have used the Adam optimization with a learning rate of 0.001 for faster convergence. Since the training set is not large enough, there is always a chance of *overfitting*. To prevent this, cross-validation on the validation dataset is performed in conjunction with training, and we incorporate early stopping with a patience of 10, so that it monitors the validation loss and terminates the training if the validation loss is not improved within 10 consecutive epochs. (An epoch signifies one complete iteration over the entire training sample space.)

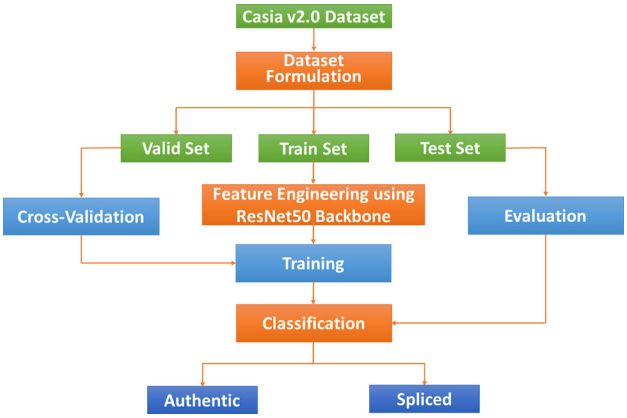

**Fig. 4** Flow diagram of the proposed method

In both of our experiments, we use the *transfer learning method* to use the pre-trained ResNet-50 backbone loaded with the ImageNet weights to extract the image features. These weights of the pre-trained backbone are frozen by setting trainable as 'False.' This stops any updates to the pretrained weights during the training as we do not want to train the ResNet layers. Our goal is to leverage the knowledge learned by the deep neural network trained on ImageNet.

**Table 2** Performance of the proposed model

| | Accuracy | Precision | Recall | F1 score | AUC score |
|---|---|---|---|---|---|
| Experiment 1 | 0.9645 | 0.9669 | 0.9415 | 0.9540 | 0.9755 |
| Experiment 2 | 0.9208 | 0.9244 | 0.9167 | 0.9205 | 0.9525 |

Only classifier network weights are modified to minimize the loss. We perform the experiments twice with two different datasets as mentioned in Sect. 3.2. The details of our experiments are given following this section.

Finally, the trained model is used to classify a completely unknown set of images constituting the test set, into two classes, viz. *Authentic* and *Spliced*. The generalization capability of the proposed model is recorded in terms of the following evaluation metrics: *Binary Accuracy*, *Precision*, *Recall*, *F1 Score* and *AUC*. These metrics have been defined and described in more detail in Sect. 4.

*Experiment 1*. Here we take the entire CASIA v2.0 dataset of 12,614 images in order to train and evaluate our model.

*Experiment 2*. In this experiment, we repeat the entire procedure with the sampled dataset of 10,246 images.

# 4 Experimental results and discussion

## 4.1 Performance of the proposed model

The basic evaluation parameters to evaluate the proposed scheme may be defined as follows [23]:

*True positive (TP)*: An outcome correctly predicted as a Spliced Image.
*True negative (TN)*: An outcome correctly predicted as an Authentic Image.
*False positive (FP)*: An outcome incorrectly predicted as a Spliced Image.
*False negative (FN)*: An outcome incorrectly predicted as an Authentic Image.

The performance of the proposed model has been analyzed and evaluated in terms of the following evaluation parameters: *Accuracy*, *Precision*, *Recall*, *F1 Score*, and *AUC Score*.

– Accuracy is defined as the sum of the number of true positives and true negatives divided by the total number of samples. That is,

$$\text{Accuracy} = \frac{\#\text{TP} + \#\text{TN}}{\#\text{TP} + \#\text{TN} + \#\text{FP} + \#\text{FN}} \tag{1}$$

– Precision is the ratio of correct positive samples to the number of actual positive samples.

$$\text{Precision} = \frac{\#\text{TP}}{\#\text{TP} + \#\text{FP}} \tag{2}$$

– *Recall* is the ratio of the number of correct positive samples out of those that were classified as positive.

$$\text{Recall} = \frac{\#\text{TP}}{\#\text{TP} + \#\text{FN}} \tag{3}$$

– *F1 Score* is the harmonic mean of Precision and Recall.

$$F1\ \text{Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

– *AUC* suggests the area under the ROC Curve.

Table 2 summarizes the performance of the proposed model in Experiments 1 and 2 (on the two datasets described in Sect. 3.3). It is evident from Table 2 that the model while being trained on the entire CASIA v2.0 dataset performs better from all perspectives and yields an Accuracy of 0.9645, Precision of 0.9669, Recall of 0.9415, F1-Score of 0.9540, and an AUC score of 0.9755. In Fig. 5, we present a bar diagram depicting a comparative performance analysis of the two experiments conducted. It is needless to mention that our classifier model consists of a huge number of trainable parameters. In Experiment 1, the model has a larger dataset to get trained on, as compared to Experiment 2. This makes the model more capable of generalizing the act of splicing detection. Also, our experiments depict no major influence of the classifier bias as mentioned in Sect. 3.1.

## 4.2 Comparison with state-of-the-art

Here, we perform a comparative analysis of the performance of our model with state-of-the-art schemes. Specifically, the comparison is drawn with the following methods:

1. Image forgery detection using SPT and LBP by Muhammad et al. [16].
2. Markov features-based scheme in DCT domain by Pham et al. [18].
3. Detecting splicing in digital photographs using high representational capacity of Illuminant Maps and CNN by Pomari et al. [20].

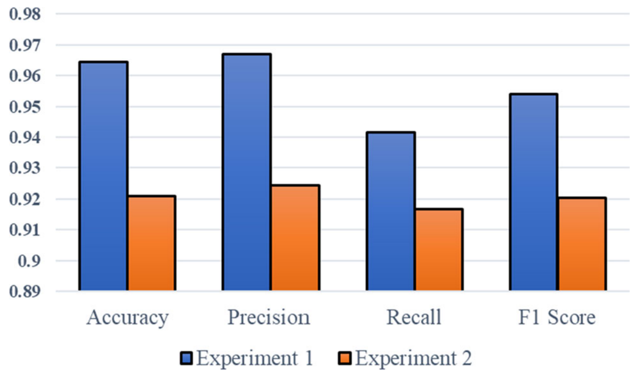

**Fig. 5** Comparison and performance analysis of Experiment 1 and Experiment 2, as discussed in Sect. 3.3

4. Texture feature criterion and Fuzzy SVM based splicing detection scheme by Tripathi et al. [21].
5. Using DMVN for the localization and detection of image splicing by Wu et al. [11].
6. Markov features-based scheme dealing with DCT and DWT domain by He et al. [24].
7. Markov features-based scheme in QDCT and QWT domain by Wang et al. [25].

Our comparison results are presented in Table 3. The results included in Table 3 are based on those reported by the authors in their original papers mentioned above.

The comparison results presented in Table 3 illustrate that the proposed scheme outperforms most of the other state-of-the-art techniques [11,20,21,24,25] in terms of splicing detection efficiency. The schemes presented in [16] and [18], slightly outperform our method (by 0.88% and 0.45% accuracy, respectively) in terms of forgery detection accuracy. However, it is worth mentioning here that such methods as the above two are based on texture/frequency domain feature extraction (like *steerable pyramid transform* (SPT), *local binary pattern* (LBP), discrete cosine transform (DCT), discrete wavelet transform (DWT), etc.), which is computationally much more intensive, given the tedious job of handpicking those features. On the contrary, the proposed method is completely based on automated feature learning, which, given the right implementation platform, is capable of optimizing manual intervention efficiently.

**Table 3** Comparison of performance with state-of-the-art

| Methods | Accuracy | Precision | Recall | F1 score |
|---|---|---|---|---|
| Muhammad et al. [16] | 0.9733 | – | – | – |
| Pham et al. [18] | 0.9690 | – | – | – |
| Pomari et al. [20] | 0.96 | – | – | – |
| Tripathi et al. [21] | 0.89 | 0.89 | 0.885 | 0.887 |
| Wu et al. [11] | 0.8708 | 0.9415 | 0.7909 | 0.8596 |
| He et al. [24] | 0.8976 | – | – | – |
| Wang et al. [25] | 0.92 | – | – | – |
| Proposed | 0.9645 | 0.9669 | 0.9415 | 0.9540 |

## 5 Conclusion and future work

With the advent of new digital technologies, the credibility of digital images is becoming more and more vulnerable with each passing day. Today, it is very easy to process a spliced image, so that it is incredibly difficult for a human being to detect the traces of forgery with naked eyes. In this paper, we discuss the issue of detecting intelligent image splicing attack. We address why it is important today to have an automated measure that can effectively detect whether an image is spliced, followed by proposing a novel approach to detect image splicing, based on deep learning. A convolution neural net-based model is introduced in this paper to perform automated feature engineering, saving the tedious task of handpicking image features. To determine whether an image is authentic or spliced, the feature vector is then fed to a dense classifier network. The proposed model is trained, validated, and finally, tested on CASIA v2.0, a standard dataset for image splicing detection and related researches [26–28].

Our experimental results demonstrate that the performance of the proposed model is superior to that of the state of the art. However, it has its own limitations. This model can only detect whether an image is spliced or not, but does not involve localization of sliced regions within such an image. In real-life cases, this is utmost important. This constitutes the primary motivation of our future research in this direction. Also, it would be interesting to explore whether the tampered regions of the image can be restored from the knowledge of location of forgery, in a computationally feasible way. This opens up a huge scope of research in the given domain.

**Acknowledgements** This work is partially funded by Department of Science and Technology (DST), Govt. of India, under Grant No.: DST/ICPS/Cluster/CS Research/2018 (General), dated: 13.03.2019.

## References

1. Shi, Y.Q., Chen, C., Chen, W.: A natural image model approach to splicing detection. Proceedings of the 9th Workshop on Multimedia and Security, pp. 51-62 (2007)
2. Ardizzone, E., Bruno, A., Mazzola, G.: Copy-move forgery detection by matching triangles of keypoints. IEEE Trans. Inf. Forensics Secur. **10**(10), 2084–2094 (2015)
3. Bharati, A., Singh, R., Vatsa, M., Bowyer, K.W.: Detecting facial retouching using supervised deep learning. IEEE Trans. Inf. Forensics Secur. **11**(9), 1903–1913 (2016)
4. Zheng, L., Zhang, Y., Thing, V.L.: A survey on image tampering and its detection in real-world photos. J. Vis. Commun. Image Represent. **58**, 380–399 (2019)

5. Farid, H.: Image forgery detection. IEEE Signal Process. Mag. **26**(2), 16–25 (2009)
6. Ansari, M.D., Satya, P.G., Vipin, T.: Pixel-based image forgery detection: A review. IETE J. Educ. **55**(1), 40–46 (2014)
7. Ng, T.-T., Chang, S.-F., Hsu, J., Xie, L., Tsui, M.-P.: Physics-motivated features for distinguishing photographic images and computer graphics. In: Proceedings of the 13th Annual ACM International Conference on Multimedia pp. 239-248 (2005)
8. Cozzolino, D., Gragnaniello, D., Verdoliva, L.: Image forgery localization through the fusion of camera-based, feature-based and pixel-based techniques. In: IEEE International Conference on Image Processing (ICIP) pp. 5302–5306 (2014)
9. Warbhe, A.D., Dharaskar, R.V.: Survey on pixel and format based image forgery detection techniques. In: Recent Trends in Computing (NCRTC), MPGI National Multi Conference pp. 7-8 (2012)
10. Kumar, M., Srivastava, S.: Image forgery detection based on physics and pixels: a study. Aust. J. Forensic Sci. **51**(2), 119–134 (2019)
11. Wu, Y., Abd-Almageed, W., Natarajan, P.: Deep matching and validation network: An end-to-end solution to constrained image splicing localization and detection. In: Proceedings of the 25th ACM International Conference on Multimedia, pp. 1480-1502 (2017)
12. Ahmed, B., Aaron Gulliver, T.: Image splicing detection using mask-RCNN. Signal, Image and Video Process, pp. 1-8 (2020)
13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770-778 (2016)
14. Zhang, Z., Kang, J., Ren, Y.: An effective algorithm of image splicing detection. IEEE Int. Conf. Comput. Sci. Softw. Eng. **1**, 1035–1039 (2008)
15. Hsu, Y.-F., Chang, S.-F.: Detecting image splicing using geometry invariants and camera characteristics consistency. In: IEEE International Conference on Multimedia and Expo, pp. 549-552 (2006)
16. Muhammad, G., Al-Hammadi, M.H., Hussain, M., Bebis, G.: Image forgery detection using steerable pyramid transform and local binary pattern. Mach. Vis. Appl. **25**(4), 985–995 (2014)
17. Dong, J., Wang, W., Tan, T.: Casia image tampering detection evaluation database. In: IEEE China Summit and International Conference on Signal and Information Processing, pp. 422-426 (2013)
18. Pham, N.T., Jong-Weon, L., Goo-Rak, K., Chun-Su, P.: Efficient image splicing detection algorithm based on markov features. Multimedia Tools Appl. **78**(9), 12405–12419 (2019)
19. Rao, Y., Ni, J.: A deep learning approach to detection of splicing and copy-move forgeries in images. In: IEEE International Workshop on Information Forensics and Security (WIFS), pp. 1-6 (2016)
20. Pomari, T., Ruppert, G., Rezende, E., Rocha, A., Carvalho, T.: Image splicing detection through illumination inconsistencies and deep learning. In: 25th IEEE International Conference on Image Processing (ICIP), pp. 3788-3792 (2018)
21. Tripathi, E., Kumar, U., Tripathi, S.P., Yadav, S.: Automated Image Splicing Detection using Texture based Feature Criterion and Fuzzy Support Vector Machine based Classifier. In: International Conference on Cutting-edge Technologies in Engineering (ICon-CuTE), pp. 81-86 (2019)
22. Nimble Challenge (2017) https://www.nist.gov/itl/iad/mig/nimble-challenge-2017-evaluation
23. Marsland, S.: Machine Learning: An Algorithmic Perspective. CRC Press, Boca Raton (2015)
24. He, Z., Wei, L., Sun, W., Huang, J.: Digital image splicing detection based on Markov features in DCT and DWT domain. Pattern Recognit. **45**(12), 4292–4299 (2012)
25. Wang, R., Lu, W., Li, J., Xiang, S., Zhao, X., Wang, J.: Digital image splicing detection based on Markov features in QDCT and QWT domain. In: Digital Forensics and Forensic Investigations: Breakthroughs in Research and Practice, pp. 61–79, IGI Global (2020)
26. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press, Cambridge (2016)
27. Ng, A.: Deep learning specialization. Coursera (2018). https://www.coursera.org/pecializations/deep-learning
28. Jaiswal, A.K., Srivastava, R.: A technique for image splicing detection using hybrid feature set. *Multimedia Tools Appl.* pp. 1–24 (2020)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

# Terms and Conditions