International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received: 11 August 2023 Revised: 28 September 2023 Accepted: 18 October 2023

Deep Learning-based Copy-Move and Spliced Image Forgery Detection

[1]Divya Prathana Timothy, [2*]Ajit Kumar Santra
[1]School of Computer Science Engineering and Information Systems,
Vellore Institute of Technology,
Vellore, Tamil Nadu, India
e-mail: timothy.prathana@gmail.com
[2*]School of Computer Science Engineering and Information Systems,
Vellore Institute of Technology,
Vellore, Tamil Nadu, India
e-mail: ajitkumar@vit.ac.in
[* indicates Corresponding Author]
Corresponding Author: Ajit Kumar Santra(ajitkumar@vit.ac.in)

Abstract: This paper proposes a Deep Learning (DL) based pre-trained AlexNet model for detecting and localizing copy-move and spliced forgery in photos. To localise forgeries in a photo, a binary mask is constructed using sobel operators. Further feature vectors are extracted patch wise from the input pictures. The Spatial Rich Model (SRM) is employed to address the generalisation issues in the DL model. There are three datasets used: Columbia Uncompressed Image Splicing Detection Evaluation (CUISDE), CASIA 1, and CASIA 2. The overall performance of the model has a 98.59 percent total accuracy as against 98.176% reported in the existing literature.

Keywords: AlexNet Model, Copy-move, Deep learning, Detection, Image forgery, Splicing.

I. INTRODUCTION

In the era of digital technology, image fraud is pervasive, threatening the veracity and reliability of visual materials like photos. Digital photos are more evocative than any other type of written documentation. They are commonly utilized as evidence in a wide range of real-world scenarios, including social-media, experimental demonstrations, legal proceedings, and more [1]. They are manipulated extensively nowadays due to the easily accessible online manipulation tools like pixlr, canva, photo editor etc... The detection of photo forgery techniques is primarily necessary for the purposes of copyright protection and prevention of forgery [2]. In recent times, there has been significant progress in the field of digital forensic, particularly in the development of digital image forgery detection techniques. Digital image authentication is widely recognized as a crucial method in the field of digital image forensic [3]. Passive and active authentication are two distinct types of authentication methods that are commonly employed in various systems and applications. The achievement of active type is facilitated by the utilization of various techniques, such as the implementation of digital signatures and steganography. Passive authentication does not provide access to the original images. The utilization of this particular authentication method is commonly observed in the identification of forged images, where the original content of the image being scrutinized is not accessible. Forgery detection in digital images is a significant concern, as it involves the analysis of fragments of evidence. The process involves the identification and examination of distinctive attributes and traits. The creation of forgery images has been explored through the application of dissimilar methods, which can be considered into distinct groups: copy-move (CMF), and spliced images. Copy-move forgery is the practice of copying or duplicating one part of an image and then moving that copied or duplicated part to another part of the same picture. On the other hand, splicing forgery involves the manipulation of an image by combining distinct sections from another image to produce a composite image. Numerous studies have been conducted to identify these forgeries independently, but very few have attempted to identify both at the same time. This served as the basis for this paper.

Our work highlights the effect of subtle artifacts which are generated due to the operation of tampering, by using SRM. The proposed methodology utilizes a patch-based method for detecting copy-move forgeries. In this approach, image patches are extracted and subsequently compared for similarity. In the context of splicing forgery detection, an analysis is conducted on various image properties including lighting, texture, and noise. These properties are examined for inconsistencies, with the aim of identifying irregularities and drawing attention to regions that may have been spliced. This analysis is facilitated by an AlexNet model that has been trained specifically for this purpose in order to enhance the detection process. To measure the efficacy and resilience of the proposed methodology, a comprehensive set of experiments is carried out on widely recognized benchmark datasets. These datasets are CASIA 1, CASIA 2, and CUISDE, which are commonly used in the field for evaluating various approaches. The evaluation of our proposed model's performance is conducted by assessing its sensitivity and specificity, which serve as indicators of its ability to accurately detect patterns in the given datasets.

Deep learning has attracted more attention lately to identify the forgery presented in an image, and many notable outcomes are starting to show which are discussed in section 2. Methods and materials are presented in section 3. Experimental setup is presented in section 4, followed by the result and discussion in section 5, and the research conclusion is in section 6.

II. RELATED WORK

This part specifies the related work for passive image forgery identification methods which include copy-move, splicing, and combination of both forgery copy-move and splicing.

A. Copy-Move Forgery

Copy-move detection techniques can be divided into three groups, depending on feature extraction and matching schemes: block-based or patch methods, key-point methods, and irregular region-based approaches. [4] employed a fusion processing technique that combines an adversarial method and a deep convolution method for the purpose of analysing copy-move counterfeit detections. The study utilized a total of four distinct databases for data collection and analysis. [5] presented two distinct strategies in their research to solve the generalization problem. The first strategy involves utilizing a convention framework, while the second strategy involves employing the Transfer Learning (TL) based model. The proposed models do not provide an explanation of how they make their decisions, which limit their interpretability and trustworthiness. [6] proposed an accurate and computationally lightweight deep learning-based CNN architecture for copy-move forgery detection with rmsprop optimizer. However, the above methods are not robust against other types of forgery, such as splicing and retouching.

B. Splicing Forgery

Splicing image forgery detection is comparative more challenging than copy-move. [7] proposed a novel approach for an efficient Image Segmentation and Feature Detection (ISFD) system. They introduce a dual-channel U-Net architecture, referred to as DCU-Net, as a potential solution. The experimental findings demonstrate the resilience of the suggested methodology. The little dataset utilised in this study may have an impact on how well the deep learning model performs. The proposed model training heavily relies on photos that have completed several post-processing procedures since in real-world situations, photographs may not have gone through such processes. [8] put forth several ISFD techniques. In their study, the researchers employed the Mask R-CNN framework, utilizing MobileNet V1 as the underlying support construction. The findings indicated a high level of superiority. Nevertheless, the model proposed by the researcher lacks extensive testing on a larger dataset of attacks. Additionally, there is a notable absence of a comparative analysis between the evaluation results obtained with and without the inclusion of attacks.

C. Combination of both Copy-Move and Splicing Forgery

In [9] authors proposed handcrafted method based on Discrete Cosine Transformation (DCT) and Local Binary Pattern (LBP) to identify image forgery of type copy-move and splicing. They used mean operator and Support Vector Machine (SVM) for character extraction and classification respectively. The issue with the traditional methods for detecting picture forgeries is determining a certain form of manipulation. [10] introduced VGG-16 and semantic segmentation for identification and localization of copy-move and splicing image forgeries. Color map is applied after the step of pre-processing to the presented model using color illumination. However, the time complexity and computation complexity are very high for the proposed model. In the research of [11] focused on the copy-move and splicing image forgery detection with the help of mask R-CNN with MobileNet Version-1. The extraction of statistical features from Input data is performed using deep learning networks. The suggested model's evaluation on a small dataset, which could not be pretty standard of all scenarios, is one of the paper's limitations. Another drawback is that the suggested model would not be able to identify forgeries in pictures with intricate backgrounds or when they are inconspicuous. Comparing the approach described here with other newly published approaches, the method provided here shows better results in terms of rapidity as well as precision.

III. METHODS AND MATERIALS

This part will cover the proposed methodology, data collection that was analysed during this research, mask extraction and the AlexNet Model used to detect picture forgeries.

A. Proposed Methodology

The entire flow of the suggested method for applying deep learning to precisely notice and localize copy-move and spliced picture counterfeiting is depicted in Figure 1. The research commences by obtaining an input image as the initial reference. Prior to inputting the image into the deep learning model, a preprocessing step are performed to optimize the image quality and eliminate potential noise or artifacts that could potentially interfere with the detection process. The preprocessing stage typically encompasses a range of operations, including but not limited to resizing, denoising, and color correction. This paper refers resizing to the adjustment of the image dimensions, to meet specific requirements of DL model.

In the subsequent step, a thorough analysis of the image is conducted in order to extract a mask that effectively identifies and emphasizes regions within the image that may potentially indicate instances of forgery. The primary objective of the mask extraction step is to sense and locate regions within the image that are prone to containing instances of copy-move or spliced forgeries. Mask extraction is a crucial task in image processing and computer vision. It involves separating the foreground object from the contextual in an image. Numerous techniques have been developed and employed for this purpose, including edge detection, texture analysis, and statistical analysis of pixel intensities. One commonly used technique for mask extraction is edge detection. This method focuses on identifying the boundaries of objects by detecting abrupt changes in pixel intensity. In this research Sobel operators is used which involves applying gradient-based edge detection to the input image. The Sobel operators are used to compute the gradient magnitude and direction, which can be further processed to extract a binary mask highlighting potential forgery regions.

Upon acquisition of the mask, the image undergoes a process of partitioning into smaller patches or sub-regions. Patch sampling is a standard method for studying images. The picture is broken up into lower, matching, and non-overlapping portions, which are then used as the input units for further analysis. This approach allows for a more detailed examination of the image, as each patch can be individually analyzed and processed. The patch sampling process can be represented using mathematical as follows:

Let's denote the input image as $X \in \mathbb{R}^{(H \times W \times C)}$. Define the patch size as $p \times p$, where p is the desired size of the patches. Typically, p is a small value such as 32 or 64 pixels. Introduce a stride parameter s, which determines the amount of overlap between adjacent patches. Iterate over the image with a sliding window approach to sample patches. The starting pixel coordinates of each patch are given by (i, j), where i ranges from 1 to $H - p + 1$ with a stride of s, and j ranges from 1 to $W - p + 1$ with a stride of s.

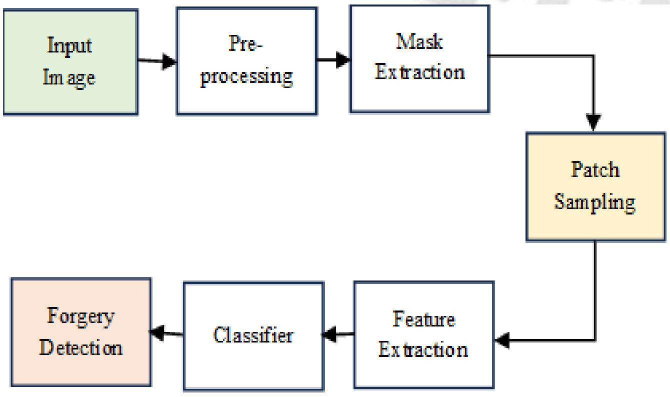


Figure 1. Overall Structure of Detect and Localize Copy-move and Spliced Picture Counterfeiting

Following the process of patch sampling, the subsequent step involves conducting feature extraction on every individual patch. The extraction of significant and distinguishing features that have the ability to accurately capture the distinctive attributes of copy-move and spliced forgery using SRM. The high-pass filter set which is also known as regulizer, is assigned in our network's initial layer to set the weight in contrast to random strategy which helped to increase the efficiency of our model. Deep learning architectures, such as CNNs, have become standard fare in the realm of machine vision for their effectiveness in extracting features from images. This is primarily attributed to their inherent capability to learn hierarchical representations. The process involves extracting feature patches from the input data, which are subsequently inputted into a deep learning classifier. This classifier is specifically trained to differentiate between regions that are authentic and those that have been manipulated. The deep learning classifier is tasked with acquiring knowledge about the intricate connections between the extracted features and the detection of forgery. The proposed methodology involves the utilization of a predictive model to estimate the probability of forgery for individual patches. This estimation serves as an indicator of the likelihood that a given patch belongs to a region that has been manipulated or tampered with.

B. Data Collection

The CASIA 1 dataset, comprises a total of 1721 images. Among these, 800 images are original, while the remaining 921 images are classified as forged. The dataset exhibits a resolution of either 384 × 256 or 256 × 384 pixels. The images have been saved in the widely used JPEG format, which is a lossy compression method for digital images.

The CASIA 2 dataset, comprises a total of 12,613 images as shown in Figure 2. Among these, 7,491 images are classified as original, while the remaining 5,122 images are categorized as forgery. The resolution of the device in question is reported to be 900 × 600 pixels.

The CUISDE dataset is comprised of a total of 363 pictures. Among these, 183 images are classified as original, while the remaining 180 images are categorized as forgery. The resolution of the subject in question ranges from 568 × 757 pixels to 768 × 1152 pixels. The extensions commonly associated with the file formats BMP and Tiff are BMP and Tiff, respectively.

***Article Received:** 11 August 2023 **Revised:** 28 September 2023 **Accepted:** 18 October 2023*

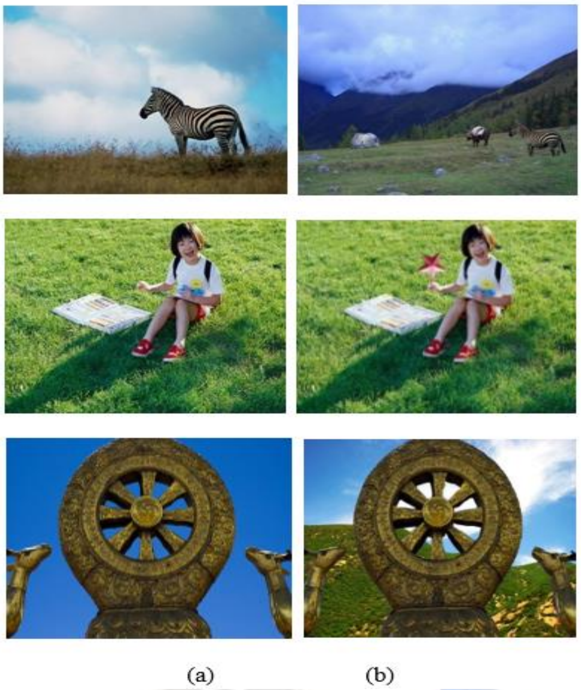


Figure 2. Sample (a) Original (b) Fake Image from the dataset CASIA

C. Mask Extraction

Mask extraction using Sobel operators involves applying gradient-based edge detection to the input image. The Sobel operators are used to compute the gradient magnitude and direction, which can be further processed to extract a binary mask highlighting potential forgery regions. Let's denote the input image as $X \in \mathbb{R}^{(H\times W\times C)}$, where H, W, and C represent the height, width, and number of channels of the image, respectively. If the picture being used is coloured, it must be transformed to grayscale before edge recognition can be done. The grayscale conversion can be represented as:

$$G = grayscale(X) \tag{1}$$

where $G \in \mathbb{R}^{(H\times W)}$ represents the grayscale image. The Sobel operators are applied to compute the gradients in the flat and vertical directions. Let's denote the horizontal gradient as G_x and the vertical gradient as G_y. These gradients can be computed as follows,

$$G_x(i,j) = G(i,j) * H_x \tag{2}$$

$$G_y(i,j) = G(i,j) * H_y \tag{3}$$

where H_x and H_y are the Sobel operator kernels in the horizontal and vertical directions, respectively. The Sobel operator kernels are defined as:

$$H_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \tag{4}$$

$$H_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \tag{5}$$

The gradient magnitude M can be obtained by combining the horizontal and vertical gradients:

$$M(i,j) = \sqrt{G_x(i,j)^2 + G_y(i,j)^2} \tag{6}$$

The gradient direction D can be calculated as the arctan of the ratio of the vertical gradient to the horizontal gradient:

$$D(i,j) = atan2(G_y(i,j), G_x(i,j)) \tag{7}$$

To extract the binary mask, a thresholding step is applied to the gradient magnitude M. Pixels with gradient magnitudes above a certain threshold T are considered as potential forgery regions and assigned a value of 1, while the rest are set to 0. The thresholding operation can be defined as:

$$Mask(i,j) = 1\ if\ M(i,j) > T, else\ 0 \tag{8}$$

The resulting $Mask \in \{0,1\}^{(H\times W)}$ represents the binary mask that highlights potential forgery regions in the input image. Regions with a value of 1 indicate areas where copy-move or spliced forgery may have occurred.

D. AlexNet Model

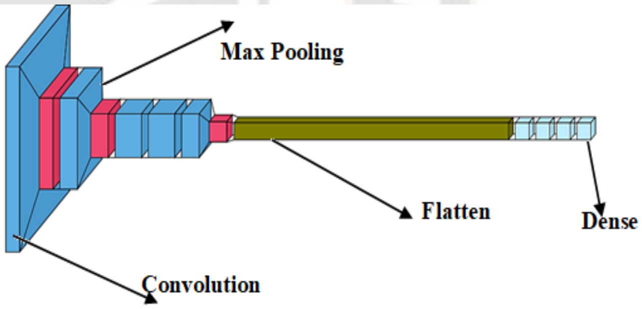


Figure 3. AlexNet's Framework for detecting Fake Photos

The deep CNN architecture known as AlexNet holds great significance in the promotion and widespread adoption of deep learning techniques. The architecture of AlexNet is composed of a total of eight layers as shown in Figure 3, which are divided into five convolutional layers and three fully connected layers.

The Input Layer is responsible for receiving the input image, which is commonly in the format of 224x224x3 (representing the dimensions of width, height, and RGB channels). The input picture is subjected to a convolution operation in the first convolutional layer, which employs 96 filters with dimensions of 11x11x3. In this procedure, a 4 stride is employed. In addition to its other functions, this layer incorporates the rectified linear unit (ReLU) activation function and carries out local response normalization.

$$Conv1 = ReLU(Convolve(X, W1) + B1) \tag{9}$$

$$MaxPool1 = MaxPool(Conv1) \tag{10}$$

The first convolutional layer's output is subjected to max pooling in Max Pooling Layer 1. This pooling operation utilizes a filter size of 3x3 and a stride of 2. In the research conducted, it was observed that in Convolutional Layer 2, the second convolution operation was functional to the output of the first pooling layer. This operation involved the utilization of 256 filters, each having a size of 5x5x48. It is worth noting that the value 48 corresponds to the number of output channels obtained from the preceding layer. In this study, the ReLU activation function and local response normalization technique was employed. The second convolutional layer's output is subjected to max pooling in max pooling layer 2, utilizing a 3x3 filter size and a stride of 2.

$$Conv2 = ReLU(Convolve(MaxPool1, W2) + B2) \tag{11}$$

$$MaxPool2 = MaxPool(Conv2) \tag{12}$$

In the research conducted, it was observed that Convolutional Layer 3 is responsible for performing the third convolution operation. This operation involves the utilization of 384 filters, each having a size of 3x3x256. The application of the ReLU activation function is limited to this scenario.

$$Conv3 = ReLU(Convolve(MaxPool2, W3) + B3) \tag{13}$$

In the fourth convolutional layer, a total of 384 filters with dimensions 3x3x192 are applied to perform the convolution operation. Several popular artificial intelligence and deep learning approaches make use of the ReLU activation function. The model is made non-linear by the use of an activation function that is non-linear.

$$Conv4 = ReLU(Convolve(Conv3, W4) + B4) \tag{14}$$

The fifth convolutional layer applies a convolution operation using 256 filters, each with a size of 3x3x192. The ReLU activation function is commonly applied in neural networks. The third Max Pooling Layer 3 applies the max pooling operation to the output of the fifth convolutional layer. This operation involves using a filter of size 3x3 and a stride of 2.

$$Conv5 = ReLU(Convolve(Conv4, W5) + B5) \tag{15}$$

$$MaxPool5 = MaxPool(Conv5) \tag{16}$$

The first fully connected layer in this research consists of 4096 neurons, which are connected in a fully connected manner to the output of the third pooling layer. Additionally, the model incorporates dropout regularization, a technique commonly employed to prevent overfitting by randomly disabling a proportion of the neurons during training.

$$Flatten5 = Flatten(MaxPool5) \tag{17}$$

$$FC6 = ReLU(FC(Flatten5, W6) + B6) \tag{18}$$

$$FC7 = ReLU(FC(FC6, W7) + B7) \tag{19}$$

$$FC8 = Softmax(FC(FC7, W8) + B8) \tag{20}$$

Convolve represents the convolution operation with weights (W) and biases (B), Maxpool represents the max pooling operation, ReLU, and FC denotes the fully connected operation. Softmax is used as the activation function for the final FC8 to obtain probability scores for different classes. The Fully Connected Layer 2 (FCL2) is a type of layer commonly used in neural networks for deep learning tasks. It is characterized by having every neuron in the layer connected to every neuron in a manner akin to the preceding layer, the current layer consists of 4096 neurons, employs the ReLU activation function, and incorporates dropout as a regularization technique. The final fully connected layer is composed of n neurons, which aligns with the n classes present in the dataset. The utilization of the softmax activation function is employed to generate the class probabilities.

IV. EXPERIMENTAL SETUP

In order to execute the proposed model, all research were performed on a computing system equipped with an Intel(R) Core i9-12900 CPU operating at a clock speed of 2.40 GHz. The system also featured NVIDA RTX A2000 graphics card with 12.0 GB of memory. The operating system utilized was a 64-bit version of Windows 10. Anaconda navigator python and Jupyter notebook were used to realize the suggested model.

A. K-Fold Cross-Validation

K-fold cross-validation checks the model's performance on the dataset after training. The dataset is divided into an arbitrary number of subsets of the same size using the k-fold cross-validation method. The optimal k was typically reported to be around 5 and 10, as statistical accuracy did not improve greatly for larger values of k, and as averaging of less than 10 divisions remained technically reasonable. The value of k was determined by balancing the model's efficiency with its accuracy. We note that among the k-fold cross-validation methods, 10-fold is the best option due to its fewer errors and less prejudiced while partitioning data into training and testing. The accuracy of the

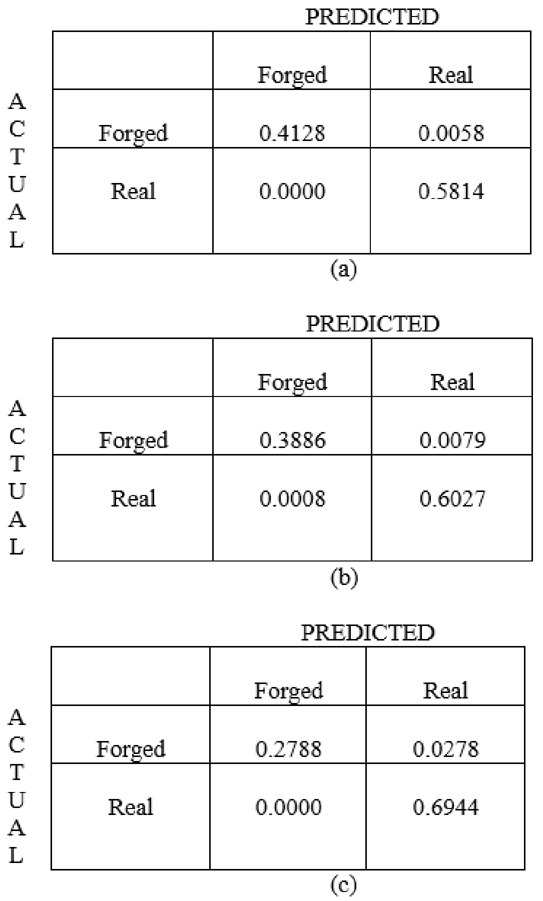
***Article Received:** 11 August 2023 **Revised:** 28 September 2023 **Accepted:** 18 October 2023*

model and the training experiment will determine whether or not k = 10 is selected. The value of k can be determined at will. If k is little, the model will be more biased towards the dataset. The bias was reduced with a larger estimate of k, however this estimate might be very variable. When k exceeds 10, the photographs in the dataset are split into ten groups. The first nine subsets make up the dataset used for training, while the tenth is used for testing. Using a new set of divisions for the test set and the existing nine as training data, we ran the training method ten times. The average value is used as a last measure of the model's performance.

V. RESULT AND DISCUSSION

This section presents an overview of the evaluation metrics employed for assessing performance. Accuracy, precision, recall, and F1 Score are used as measures for assessment in the present research. These metrics were selected based on their established relevance and widespread usage in the field of research. The accuracy of an algorithm is often evaluated by how well its forecasts actually turn out.

TABLE I. CONFUSION MATRIX ON (a) CASIA 1 (b) CASIA 2 (c) CUISDE



		PREDICTED	
		Forged	Real
ACTUAL	Forged	0.4128	0.0058
	Real	0.0000	0.5814

(a)

		PREDICTED	
		Forged	Real
ACTUAL	Forged	0.3886	0.0079
	Real	0.0008	0.6027

(b)

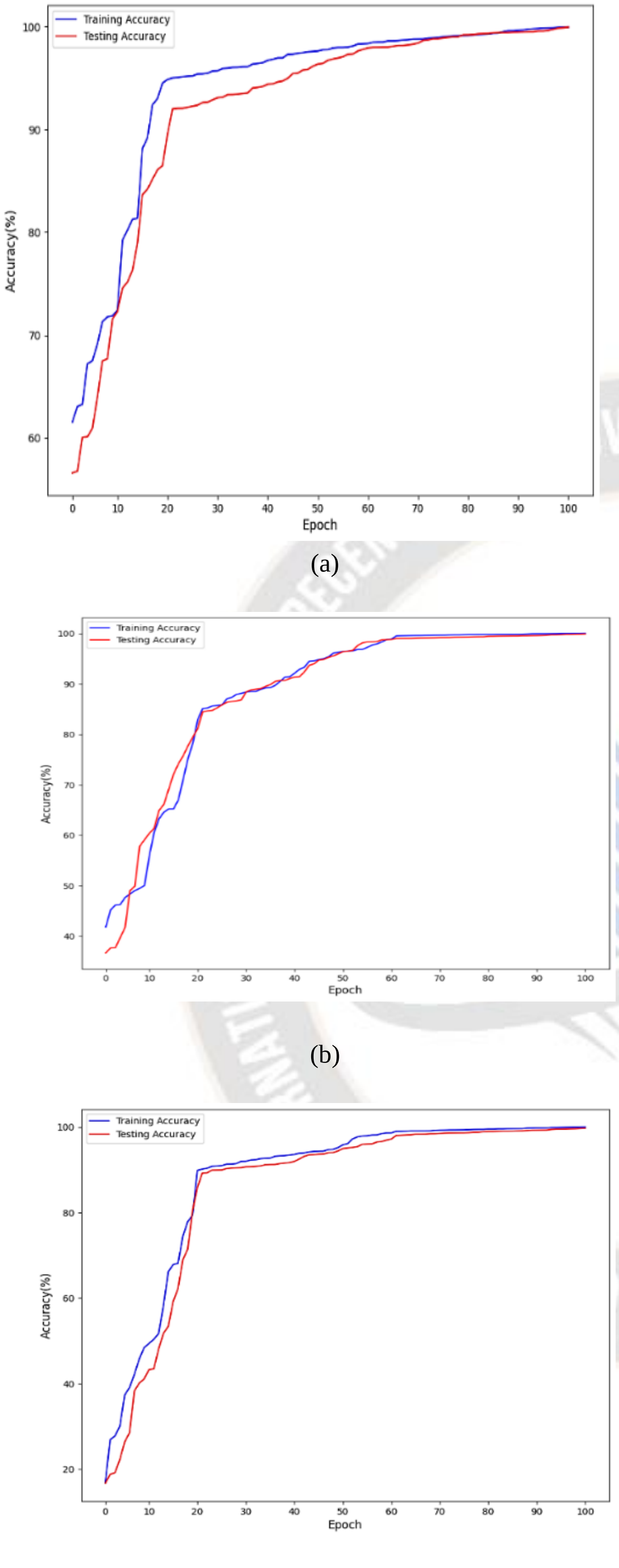
		PREDICTED	
		Forged	Real
ACTUAL	Forged	0.2788	0.0278
	Real	0.0000	0.6944

(c)

In contrast, the goal of precision is to maximize the fraction of correct forecasts that are favourable among every favourable forecast generated by the four values shown in the confusion matrix in Table-I serve as the basis for the assessment criteria used in this research. The confusion matrix is a crucial tool that compares the predicted class with the actual class. In the context of image forgery detection, the term "True Positive" (TP) denotes to the number of forged images that are correctly identified as forged. On the other hand, "True Negative" (TN) states to the number of pristine images that are correctly identified as pristine. Conversely, "False Positive" (FP) refers to the number of pristine images that are erroneously identified as forged, while "False Negative" (FN) denotes to the number of forged images that are incorrectly identified as pristine.

The accuracy results for image forgery detection on three distinct datasets, namely CASIA 1, CASIA 2, and CUISDE, are depicted in Figure 4. According to the findings presented in Figure 4.a the CASIA 1, the accuracy achieved for identifying image forgeries on the CASIA 1 dataset is documented as 99.42%. The achieved accuracy of the detection model in identifying forged or manipulated images in the CASIA 1 dataset was found to be best as evidenced by an impressively low error rate of merely 0.58%. According to the findings presented in Figure 4.b of the research paper, the accuracy achieved for identifying image forgeries on the CASIA 2 dataset is documented to be 99.12%. The results of the study demonstrate that the detection model exhibited a remarkably high level of accuracy when it came to discerning forged or manipulated images within the CASIA 2 dataset. The model's error rate was found to be a mere 0.87%, indicating its proficiency in accurately identifying such images.

According to the findings presented in Figure 4.c of the research paper titled "CUISDE: A Comprehensive Dataset for Image Forgery Detection," the accuracy achieved for image forgery detection on the CUISDE dataset is reported to be 97.22%. The results indicate that the detection model exhibited less accuracy in discerning forged or manipulated images within the CUISDE dataset, with an error rate of 2.78%. Compared to other datasets CUISDE involves small size of data for training the model that catalyst the reduced accuracy.



(a)

(b)

(c)

Figure 4. Accuracy for image forgery detection on (a) CASIA 1 (b) CASIA 2 (c) CUISDE

The results presented in Table-II demonstrate the remarkable performance of the forgery detection model across the three datasets. The outcomes of this research suggest that the mathematical framework demonstrates effectiveness in detecting instances of image manipulation across diverse scenarios.

In this work, the performance metric for image forgery detection on three distinct datasets, namely CASIA 1, CASIA 2, and CUISDE, are presented in Table II. The performance metric includes the accuracy, precision, recall and F-1 score values for copy-move and splicing image forgery detection. According to the research on dataset CASIA 1, the precision value is recorded as 100%. The findings indicate that the forgery detection model, which underwent training using the CASIA 1 dataset, exhibits a comparatively minimal loss. The recall value for the same dataset is recorded as 98.30%. This suggests that the model demonstrates a commendable ability to differentiate between genuine and altered images within the dataset. The F-1 score is 99.3% and the accuracy is 99.42% achieved. In general, a lower loss value is often indicative of superior model performance. According to the findings of the CASIA 2 presented in Table II, it is observed that the precision value is recorded as 99.796%. The results suggest a marginally increased level of loss in comparison to the findings reported in CASIA 1.

Table-II Performance Metrics for image forgery detection on CASIA 1.0, CASIA 2.0 and CUISDE

Datasets	Precision (%)	Recall (%)	F-1 Score (%)	Accuracy (%)
CASIA 1	100	98.61	99.30	99.42
CASIA 2	99.8	98	98.89	99.12
CUISDE	100	90.9	95.23	97.22

Other values of the dataset CASIA 2 recorded as recall is 98%, F-1 score is 98.89% and accuracy is 99.13%. The findings indicate that the forgery detection model trained on CASIA 2 exhibits a marginally reduced performance or potentially encounters increased challenges in accurately identifying image forgeries when compared to CASIA 1. Nevertheless, the lack of additional context or specific details regarding the model and evaluation metrics makes it difficult to establish conclusive findings. According to the findings presented for CUISDE dataset in table 2, the precision value is reported to be 100%, recall is 90.90%, F-1 score is 95.23% and accuracy is 97.22%. The results demonstrate a greater level of loss when compared to both CASIA 1 and CASIA 2 datasets. This implies that the forgery detection model, which was trained on the CUISDE dataset, exhibits a relatively lower performance in accurately distinguishing between genuine and manipulated images within this specific dataset.

VI. CONCLUSION

In conclusion, images show a substantial role in various arenas, particularly with the widespread use of the Internet. However, the ease of editing photographs using powerful online photo editors has made it challenging to detect fake or manipulated images. This poses a serious threat to the authenticity and reliability of visual materials in today's digital age. Traditional image processing techniques often rely on manual pattern recognition and are limited in their capacity to process massive volumes of information. While the accuracy of deep learning models has gotten better, they still struggle with generalize since they still rely on training datasets and require accurate hyperparameter optimization. This research proposes a new counterfeit detection approach using SRM that makes use of deep learning to solve this problem. In order to extract vectors of attributes from overlapped subblocks of pictures, this method makes use of a pre-trained model called AlexNet. Comparison between these feature vectors has been investigated intensively after the extraction procedure and is used for both identifying counterfeits and localization. The suggested model's performance is evaluated on three datasets: CASIA 1, CASIA 2, and CUISDE. The results demonstrate high sensitivity and specificity in detecting image forgeries. The proposed model achieved a 99.42% detection rate on the CASIA 1 dataset, 99.12% on the CASIA 2 dataset, and 97.22% on the CUISDE dataset. The overall performance of the model is found to be 98.59%, which is better than the state-art model [12]. Moreover, the model showcased accurate results while requiring only a few variables, indicating its efficiency.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Conceptualization, methodology, software, validation, formal analysis, investigation, resources, data curation, writing—original draft preparation, writing—review and editing, visualization, Divya Prathana Timothy; supervision, project administration, Ajit Kumar Santra;

References

[1] Parveen A., Khan Z.H. and Ahmad S.N., “Block-based copy–move image forgery detection using DCT”, Iran Journal of Computer Science, 2, pp.89-99, 2019.

[2] Alkawaz M.H., Sulong G., Saba T. and Rehman A., “Detection of copy-move image forgery based on discrete cosine transform”, Neural Computing and Applications, 30, pp.183-192, 2018.

[3] Sadeghi S., Dadkhah S., Jalab H.A., Mazzola G., and Uliyan D., “State of the art in passive digital image forgery detection: copy-move image forgery”, Pattern Analysis and Applications, 21, pp.291-306, 2018.

[4] Abdalla Y., Iqbal M.T. and Shehata M., “Copy-move forgery detection and localization using a generative adversarial network and convolutional neural-network”, Information, 10(9), p.286, 2019.

[5] Rodriguez-Ortega Y., Ballesteros D.M. and Renza D., “Copy-move forgery detection (CMFD) using deep learning for image and video forensics”, Journal of imaging, 7(3), p.59, 2021.

[6] Hosny K.M., Mortda A.M., Fouda M.M., and Lashin N.A., “An efficient CNN model to detect copy-move image forgery”, IEEE Access, 10, pp.48622-48632, 2022.

[7] Ding H., Chen L., Tao Q., Fu Z., Dong L., and Cui X., “DCU-Net: a dual-channel U-shaped network for image splicing forgery detection”, Neural Computing and Applications, 35(7), pp.5015-5031, 2023.

[8] Kadam K., Ahirrao S., Kotecha K., and Sahu S., “Detection and localization of multiple image splicing using MobileNet V1”, IEEE Access, 9, pp.162499-162519, 2021.

[9] Islam M.M., Karmakar G., Kamruzzaman J., and Murshed M., “A robust forgery detection method for copy–move and splicing attacks in images”, Electronics, 9(9), p.1500, 2020.

[10] Abhishek and Jindal N., “Copy move and splicing forgery detection using deep convolution neural network, and semantic segmentation”, Multimedia Tools and Applications, 80, pp.3571-3599, 2021.

[11] Kadam K.D., Ahirrao S, and Kotecha K., “Efficient approach towards detection and identification of copy move and image splicing forgeries using mask R-CNN with MobileNet V1” Computational Intelligence and Neuroscience, 2022, 2022.

[12] Samir S., Emary E., El-Sayed K., and Onsi H., “Optimization of a pre-trained AlexNet model for detecting and localizing image forgeries”, Information, 11(5), p.275, 2020.