IMAGE FORGERY DETECTION AND LOCALIZATION USING FASTER RCNN

P. Mounika

PG scholar, Department of ECE, Jawaharlal Nehru Technological University Anantapuramu, Andhra Pradesh.

Abstract:

Images serve as a powerful medium for conveying information and are widely utilized in official documents, medical records, financial reports, legal evidence, and multimedia presentations. Unfortunately, some individuals deliberately manipulate images with malicious intentions, compromising the genuineness and authenticity of the visual content. Detecting image forgery is paramount in safeguarding the genuineness of images, particularly in fields such as multimedia, forensics, and medicine. Image forgery encompasses various techniques, including copy-move and image splicing, where information within an image is deceitfully altered or misrepresented. Detecting these manipulations is essential to maintain the credibility and trustworthiness of visual data in diverse applications and contexts.

The current method employs a Convolutional Neural Network (CNN) to detect copy-move image forgery. However, This method can only identify a single kind of forgery. and does not specify the forgery's location within the image. In contrast, the implemented method introduces an efficient algorithm that can identify both image splicing and copy-move image forgery. Additionally, it precisely pinpoints the manipulated portion within the image, thereby improving the overall detection process. This algorithm employs Faster R-CNN (Faster regions with convolutional neural networks) and Error Level Analysis.

Keywords: Image Forgery detection, Error Level Analysis, Faster region with convolution neural network, Copy-move and image splicing forgeries.

1 INTRODUCTION

Detecting image forgery is a vital research field focused on identifying manipulated or tampered images to maintain their authenticity and credibility. In the digital age, where sophisticated editing tools make it easy to alter images, detecting these forgeries is essential in various fields such as forensics, journalism, and legal proceedings.

Forgery techniques, including copy-move (involving duplicating a portion of the image and relocating it) and image splicing (combining different images or segments of images), are methods used for digital image manipulation, that challenge the integrity of digital visual content. Detecting these manipulations involves employing advanced technologies, including deep learning algorithms and convolutional neural networks (CNNs). These techniques analyze intricate patterns, features, and inconsistencies within images to differentiate between genuine and manipulated content. Image forgery detection plays a vital role in upholding the trustworthiness of digital images, ensuring the reliability of visual information in diverse applications and contexts.

As image manipulation techniques evolve, continuous advancements in forgery detection methods are essential to maintain the integrity of digital media.

1.1 Image Forgery Detection Techniques:

Detecting image modification techniques can be broadly divided into two categories: active techniques and passive techniques.

1. Active techniques:

- In active techniques, supplementary information is incorporated into the image, either during the image capture process or subsequently by an authorized individual or entity. This added information can take the form of digital watermarks or cryptographic signatures.

- The added information is used to detect manipulation. Active approaches rely on this embedded data to determine whether an image has been altered. If the embedded information is altered, it indicates possible forgery.

2. Passive techniques:

- Passive techniques, frequently called "blind approaches," do not rely on any supplementary information within the image. They analyze the inherent features of the image itself without any embedded data.

- Passive methods focus on the analysis of image properties such as pixel values, noise patterns, and statistical features. Algorithms and techniques like digital forensics, pattern recognition, and machine learning are applied to these properties to identify inconsistencies that suggest manipulation.

In summary, active methods involve adding extra information to the image, while passive methods analyze the image itself without any additional data. Both types of methods play vital roles in the field of image forensics, ensuring the trustworthiness and credibility of digital imagery. [3].

In Fig.1, the categorization of forgery detection is depicted. Passive approaches relying on forgery-type detection can be classified into (i) copy-move and (ii) splicing.

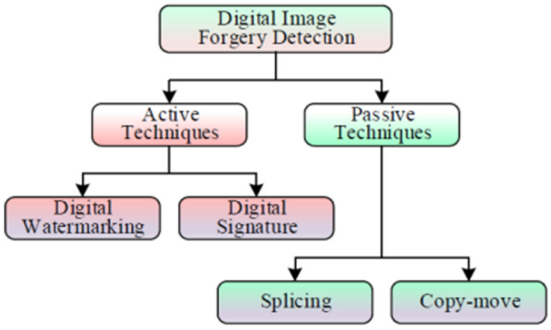

Fig.1: Approaches for digital image forgery detection

A. Copy-move forgery:

A type of digital image modification known as "copy-move forgery" involves copying and pasting a portion of an image to another location within a similar image. This technique is used to create the illusion of additional objects or elements within the photo. By transferring and duplicating a segment of the picture, the forger aims to deceive viewers into believing that the duplicated objects are authentic components of the original scene.

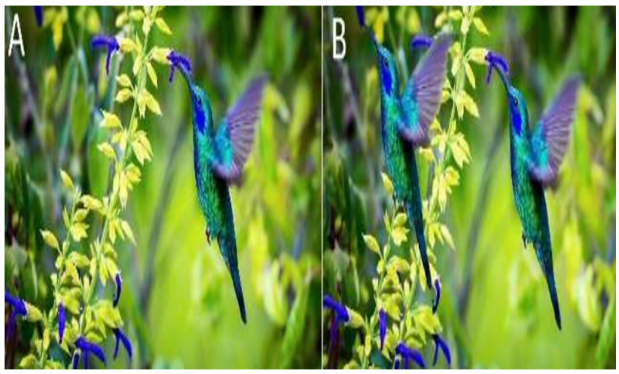

Fig 2. A: Authentic image B: Forged image

Techniques like rotation, scaling, and other operations can be performed on the cloned region. These methods are used to make it difficult for human sight to spot the fake. [3]. Fig.2 illustrates a comparison between an authentic image (labelled as A) and a forged image (labelled as B). In image B, the bird region has been clipped and attached within the same image, either to obscure specific details or for some other purpose. While this might seem like a straightforward case, its implications are substantial, especially when applied to photos relevant to fields such as medicine, forensics, and defense.

B. Image splicing:

A computer image editing technique called "image splicing" involves copying or cutting parts of an image, or several images, and then pasting them into another image. This process involves combining different visual elements to create a composite image. Splicing is often used for creative purposes in graphic design and digital art.

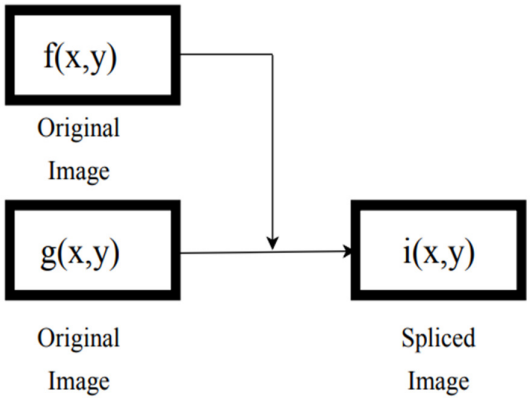

Fig.3: Methodology of Image splicing

Fig.3 illustrates the methodology of image splicing forgery. Image splicing is distinct from copy-move forgery as the copied portion cannot be located elsewhere in the same image. This fundamental difference makes recognizing image-splicing forgeries more demanding than identifying copy-move forgeries. Similar object contours within the same image are easier to identify due to shared qualities like texture, colour, size, and shape. However, in instances of image splicing, the newly inserted object contour possesses distinct image attributes. [3].

Fig.4: Image splicing Forgery

The photos that are altered using these two techniques are incredibly difficult for humans to recognize. Determining these two types of forgeries becomes crucial and will also help with digital image forensics. Passive procedures are more effective than active ones, like watermarking, but they are also more difficult. Passive techniques involve extracting features from an image, which are then utilized to identify potential forgeries. Based on coincidences (correlation) or discrepancies between the features derived from two different portions of the image, duplication is detected.

1.2 Deep Learning-Based Forgery Detection:

Forgery Detection through Deep Learning techniques involves the application of advanced neural networks to analyze and identify manipulated or altered images. This approach leverages deep learning algorithms to recognize patterns, inconsistencies, and subtle alterations within digital images, making it a powerful tool for forgery detection. Training deep learning models is indeed a complex task, demanding substantial processing power and large datasets. convolutional neural networks (CNN) are widely used. CNNs incorporate a convolution layer that serves as both a discriminator and a feature extractor. [2].

2. EXISTING SYSTEM:

The Convolution Neural Network (CNN) model serves as the foundation for this approach. The essential components of deep learning networks encompass the convolution layer, the pooling layer, and the fully connected layer. Convolutional neural networks (CNNs) are comprised of numerous convolutional layers, coupled with a pooling layer. Following this, one or more fully connected layers are integrated into the network structure.

A. Convolution layer: This layer, positioned at the top, is utilized to extract features from the source image. Convolution between the image and a MxM filter is done by this layer. The resulting feature map, generated through this convolution process, contains detailed details of the picture, such as its boundaries and corners.

B. Max Pooling layer: This layer's main objective is to reduce the convolved feature map's dimensions, thus conserving computational resources. This layer condenses the features extracted by convolution. Various types of pooling methods exist, each employing a different operation. In Max Pooling, the highest element in obtained features is chosen.

C. Fully connected layer: All neurons are coupled in this layer. This layer lessens CNN's reliance on human oversight. Prior to feeding the dense layer, this layer first flattened the image. In this layer, classification is also done.

In the CNN approach, the entire image is utilized, differing from the conventional method that employs a block-based algorithm. This methodology encompasses three key stages: preprocessing, feature extraction, and classification. The CNN model's basic function is to retrieve significant details from the image.

The framework of the existing algorithm is illustrated in Fig.5. The images are taken from different datasets. This approach was performed on three Standardized evaluation datasets namely MICC-F2000, MICC-F220, and MICC-F600.

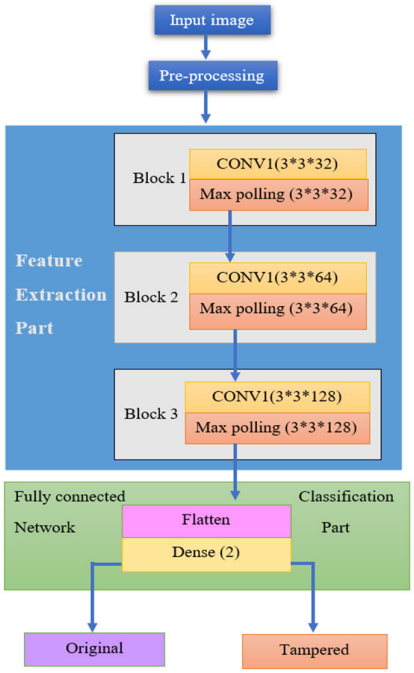

Fig.5: Framework of the existing algorithm

An input image is scaled for further processing during the preprocessing stage without any image elements being removed. Three convolution layers are used in the feature extraction step, and then a max-pooling layer. Features from the pre-processed image are extracted using three sets of convolution layers, and the image's dimensions are adjusted through max-pooling operations. The resulting feature maps, integrated with the final max-pooling, are vectorized and fed into fully connected layers.

During the classification step, the fully connected layer links all the extracted features to the dense layer, classifying the information as either authentic or falsified. The proposed model can be effectively trained due to its use of the "rmsprop" optimizer and a batch size of 32. [1].

This approach was only developed for copy-move forgery and it does not locate the forged part in an image.

3. PRELIMINARIES:

3.1. Object Detection:

Image classification and object localization are combined to form object detection. Images that are entered are categorized into two or more classes in image classification. Use bounding boxes to find the things in the image during object detection. The only purpose of the CNN algorithm is image classification. It is unable to identify the image's objects. Consequently, various object-detecting methods have been created.

Fig.6 illustrates the categorization of object detection algorithms. Single-shot detectors and two-stage detectors are the two main categories to which object detection algorithms belong. This classification is based on the amount of network iterations an input image has undergone.

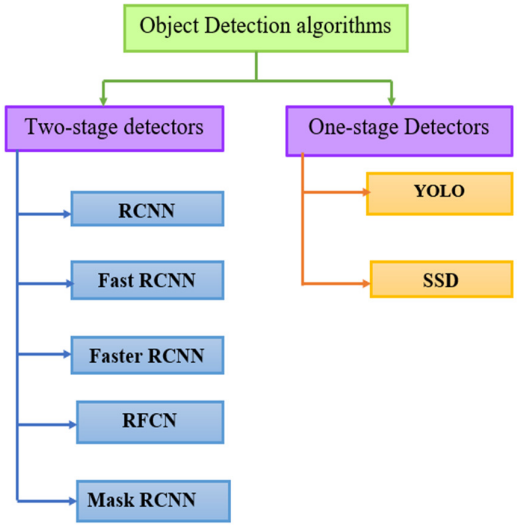

Fig.6: Classification of object detection algorithms

This object detection algorithm involves three key processes:

- Identifying regions in the image that potentially contain an object, known as region proposals.
- Extracting CNN features from these region proposals.
- Classifying the objects using the extracted features.

Different types of methods are used for extracting region proposals from the image. They are selective search, Region Proposal Network (RPN), and sliding window method. In this two-stage detector i.e., the Faster RCNN algorithm is used.

3.2. The description of FASTER RCNN:

Faster R-CNN belongs to the R-CNN family and serves as an object detection architecture. Its primary objective is to develop a powerful network capable of recognizing and locating objects within images effectively.

Convolution neural networks (CNNs) and Region Proposal Networks (RPNs) are combined into one network to increase the model's speed and accuracy. Fig.7 illustrates the architecture of the R-CNN model.

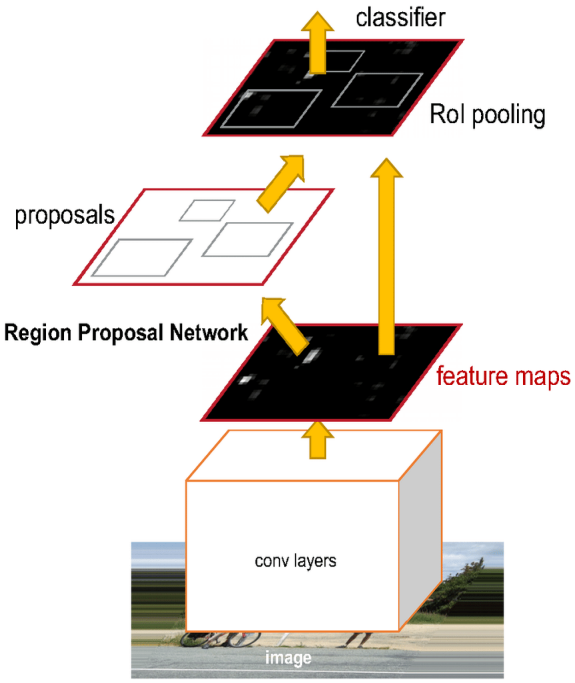

Fig.7: Faster R-CNN architecture

It is mostly made up of two sections:

A. Region Proposal Network (RPN)

B. Fast R-CNN detector

A. Region Proposal Network (RPN):

Regional proposals are produced by the RCNN and Fast RCNN models using a conventional selective search algorithm. In this computation process takes longer. The proposal time for each image is decreased in the Faster RCNN model from 2 seconds to 10 ms by the addition of a convolutional-based network i.e., RPN.

RPN is responsible for generating possible regions of interest (region proposals) in images that may contain objects. Fig.8 explains the operation of the Region Proposal Network.

The RPN employs the feature maps acquired by the backbone CNN. The RPN uses a sliding window method with anchor boxes of various sizes and shapes to indicate potential object positions on these feature maps. Throughout training, the network adjusts these anchor boxes to better match the sizes and positions of real objects.

The RPN indicates two parameters for each anchor:

- The probability of the anchor containing an object ("objectness Score")
- Adjustments to the anchor's coordinates to match the actual object's shape.

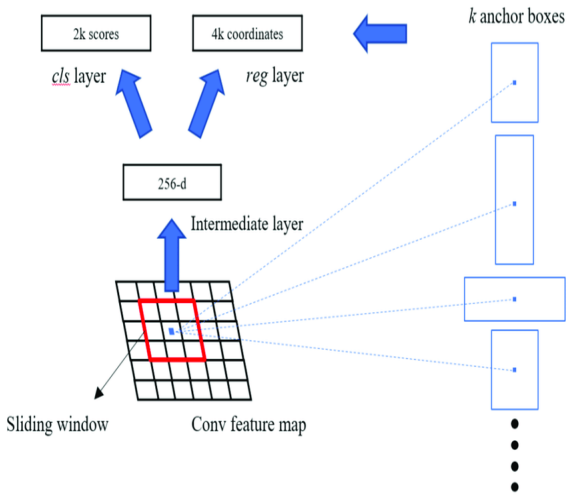

Fig.8: operation of RPN

When several area proposals are generated, many of them may overlap and correspond to the same object. Here, the anchor boxes are ranked according to their objectness probability using the Non-Maximum Suppression (NMS), and anchor boxes with the highest scores are selected in the top-N. NMS makes sure that the final, chosen proposals are correct and unique. These chosen anchor boxes are taken into consideration as possible region proposals.

B. Fast R-CNN detector:

The Fast R-CNN detector is a vital element of the Faster R-CNN architecture, tasked with identifying objects within the region proposals generated by the Region Proposal Network.

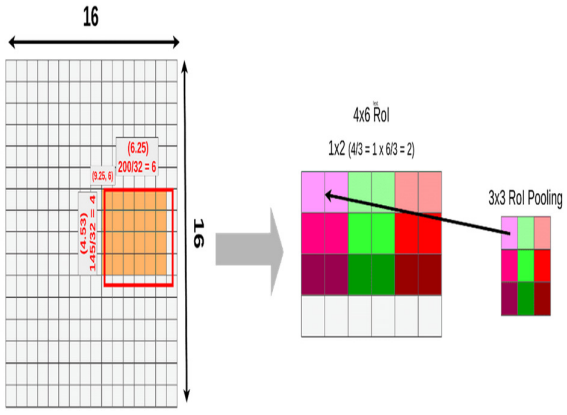

Fig.9: Mechanism of RoI Pooling

Fig.9 explains the Mechanism of RoI Pooling. The mechanism of Region of Interest (RoI) pooling involves dividing the region proposals into a fixed-size grid and then applying max-pooling or average-pooling to each

grid cell. This process ensures that the features extracted from various-sized RoIs are transformed into a uniform size, making them suitable for further processing in neural networks. RoI pooling is a crucial step in object detection architectures like Faster R-CNN, allowing the network to handle variable-sized regions efficiently.

The RoI-pooled feature maps are input into the CNN to extract meaningful features that capture object-specific information. The RoI-pooled and feature-extracted regions are then routed through a series of fully linked layers. The tasks of classifying objects and bounding box regression are carried out by these layers. After the network predicts class probabilities and bounding box changes, the final detection results are refined using a post-processing procedure. Non-maximum suppression (NMS) is employed in this stage to eliminate redundant detections while keeping the most certain and non-overlapping detections.

4. IMPLEMENTATION:

This approach is developed to identify both copy-move and image-splicing forgeries while also pinpointing the specific manipulated area within the forged image. A faster R-CNN model is used for this application. Fig.10 shows the Structure of the implemented model.

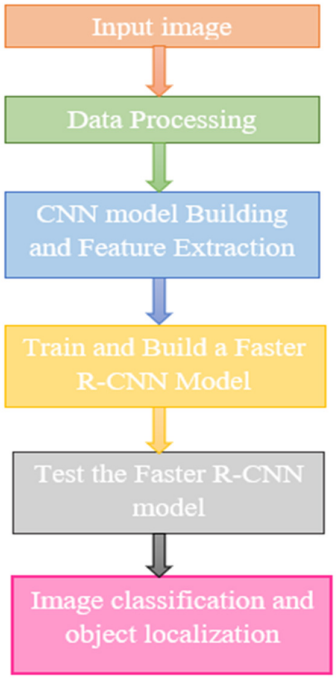

Fig.10: Structure of implemented model

Table 1: The details of the above-mentioned datasets

Dataset	Release year	Image size	Image Type	Authentic	Tampered
CASIA V2.0	2009	240x160 to 900x600	TIFF/JPEG	7491	5123
MICC-F2000	2011	2048 x 1536	JPEG	1300	700
MICC-F220	2011	722x480/ 800x600	JPEG	110	110
COVERAGE	2016	Various	TIFF	100	100
MISD	2021	384x256	JPEG	618	300

A. Input Data:

Standard and widely used datasets for evaluating image forgery detection techniques include CASIA V2.0, MICC-F2000, MICC-F220, COVERAGE, and MISD. Details regarding the contents of these datasets are provided in Table 1.

B. Data Processing:

In data preprocessing, it applies an ELA operation to the provided image. Error level analysis is a preprocessing method that compares the original image with a compressed version. Based on the JPEG compression ratio of each pixel, the image is coloured.

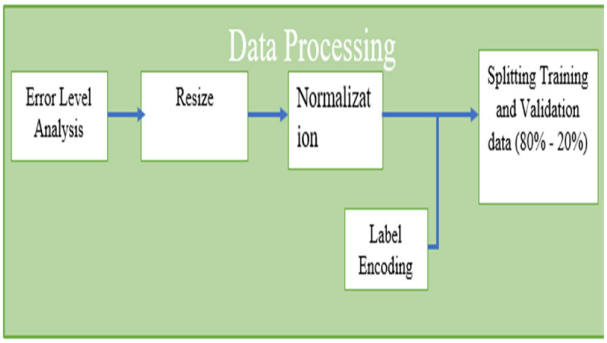

Fig.11: Data Processing block

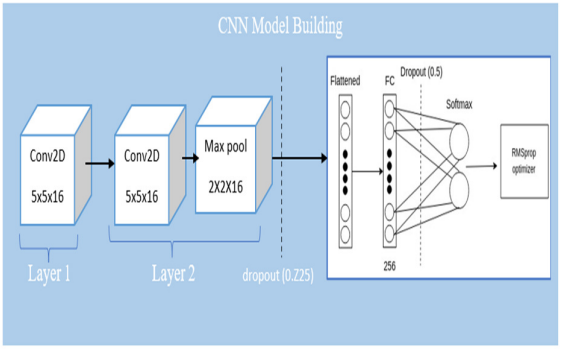

Fig.12: CNN Model for implemented model

A change in the level of compression artifacts in the image could suggest that it was manipulated. Resize and normalize the ELA pictures. The dataset is partitioned for training and validation purposes. Fig.11 shows the operations in the data processing step.

C. CNN Modelling and Feature Extraction:

The CNN Model receives the Ela pictures in order to create feature vectors. Fig.12 contains the CNN model for the implemented model. It has two convolutional layers with a 5x5 kernel each and a 2x2 max pooling layer. A dropout layer is next applied, which reduces the features by a factor of 0.25. The Fully Connected layer, which connects every neuron, then uses the Softmax Classifier to determine whether the image is genuine or a forgery.

D. Training and Evaluation of Faster R-CNN Model:

Initially, construct the Faster R-CNN model and proceed with training it using the available datasets. Typically, 60% to 80% of images within the datasets are allocated for training, while the remaining 20% are set aside for testing the model. If the Convolutional Neural Network (CNN) classifies an image as forged, it is then passed to Faster R-CNN for the localization of the manipulated region. It gives whether it is a copy-move or image splicing with a forged part location.

5. RESULTS:

The hardware used for this is an HP Pavilion quad-core CPU with 8 GB RAM storage. The code is implemented in Python 3. The tests were executed on the Google Collaboratory server.

Table 2: Comparison of metrics over datasets

Dataset	Accuracy	Precision	Recall	F1-score
MICC-F2000	96.06	96.38	96.05	96.05
MICC-F220	95.00	95.35	95.00	95.05
Coverage	100	100	100	100
CASIA V2.0	87.52	87.68	87.52	87.58
MISD	85.56	85.72	85.55	85.20

A. Evaluation Metrics:

The parameters used to assess the capability of the implemented model are as follows:

- Accuracy = $\frac{(T_N + T_P)}{(T_P + F_P + T_N + F_N)} \times 100$
- Precision = $\frac{T_P}{T_P + F_P} \times 100$
- Recall = $\frac{T_P}{T_P + F_N} \times 100$

- $\text{F1} - \text{score} = \frac{2 \times (\text{precision} \times \text{recall})}{(\text{precision} + \text{Recall})}$

In the provided context, T_P signifies the count of tampered images accurately identified as tampered, whereas F_P stands for the count of original images incorrectly identified as tampered. F_N represents the count of tampered images mistakenly identified as original, and T_N represents the count of original images correctly identified as original.

B. The Results over Datasets:

In this method, five benchmark datasets are used. The performance of this method over different datasets is given below. Table 2 demonstrates that the implemented approach surpasses in terms of accuracy, precision, F1-score, and recall.

1.MICC-F200 Dataset:

At 20 epochs, this model attains a 96% accuracy rate. It gives 96.55% highest accuracy value at epoch 4 in validation accuracy and 97.78% in training.

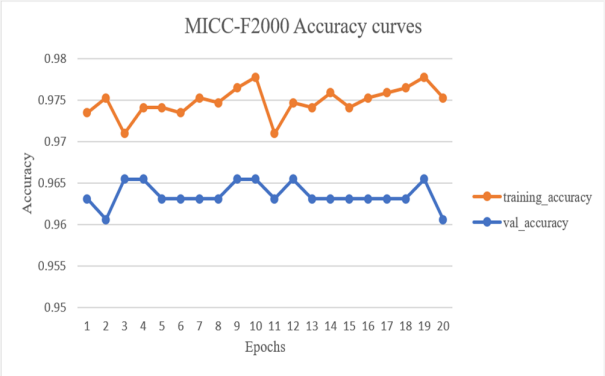

Fig.13: Accuracy values for MICC-F2000 dataset

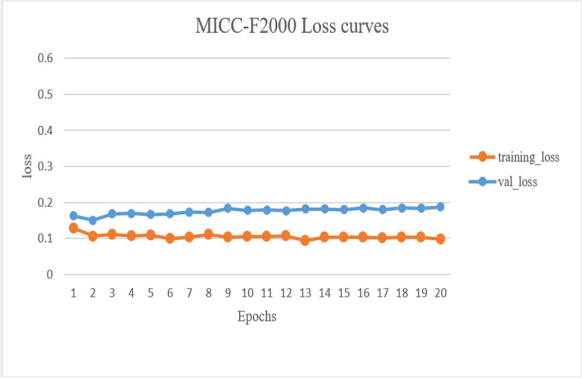

Fig.14: Loss values for the MICC-F2000 dataset

The outcomes are depicted through accuracy and loss curves illustrated in Fig.13 and fig.14. This model gives better accuracy for this dataset than others, but coverage gives 100%.

2. MICC-F220 Dataset:

At 20 epochs, this model achieves an accuracy of 95.35%. The results are displayed through accuracy and loss curves, as depicted in Fig.15 and fig.16.

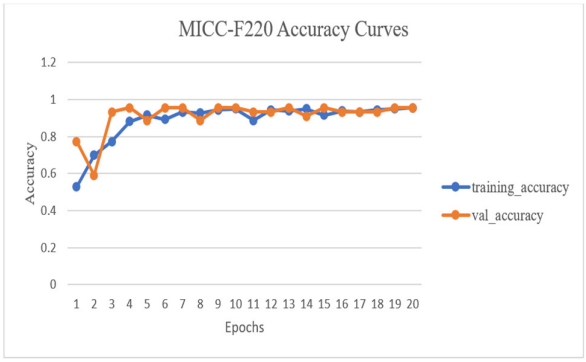

Fig 15: Accuracy values for MICC-F220 dataset

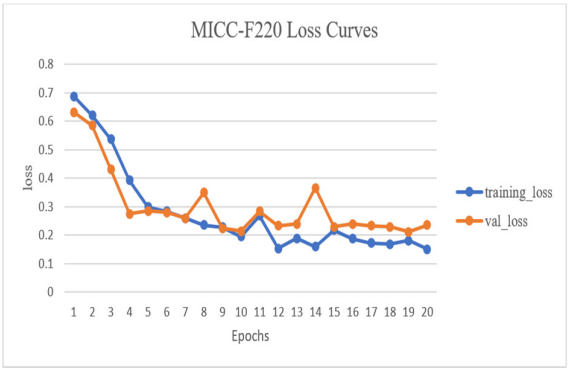

Fig 16: Loss values for the MICC-F2000 dataset

3. Coverage Dataset:

At 20 epochs, this model reaches a perfect accuracy of 100%. The results are visualized through accuracy and loss curves, as presented in fig.17 and fig.18.

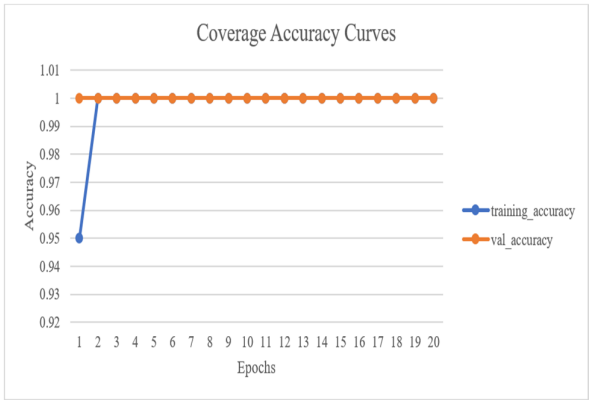

Fig.17: Accuracy values for the COVERAGE dataset

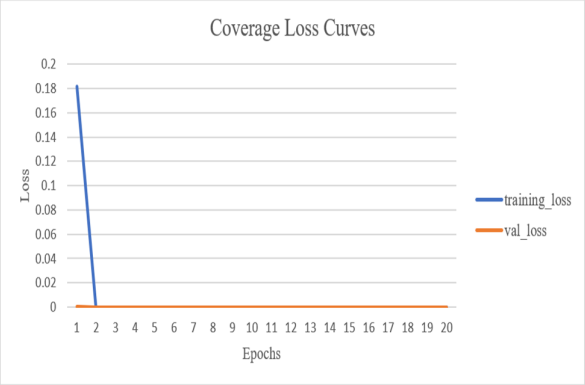

Fig.18: Loss values for the COVERAGE dataset

4. CASIA V2.0:

At 20 epochs, this model achieves an accuracy of 87.68%. The results are illustrated in terms of accuracy and loss curves, as displayed in fig.19 and fig.20.

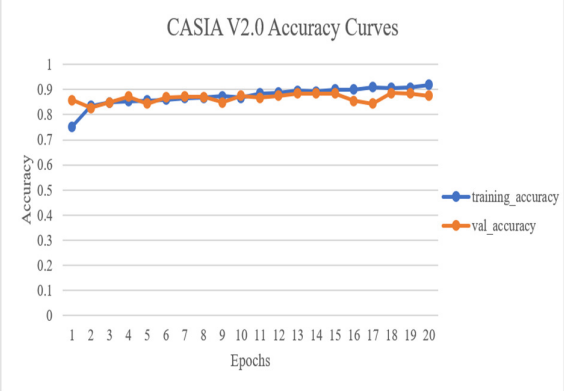

Fig.19: Accuracy values for CASIA V2.0 dataset

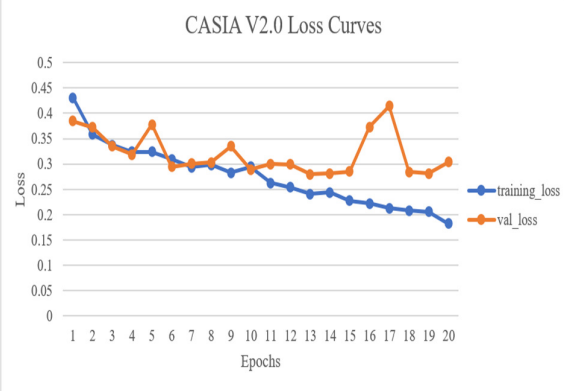

Fig.20: Loss values for CASIA V2.0 dataset

5. MISD:

At 20 epochs, this model reaches an accuracy of 85.72%. The results are visualized through accuracy and loss curves, as presented in fig.21 and fig.22.

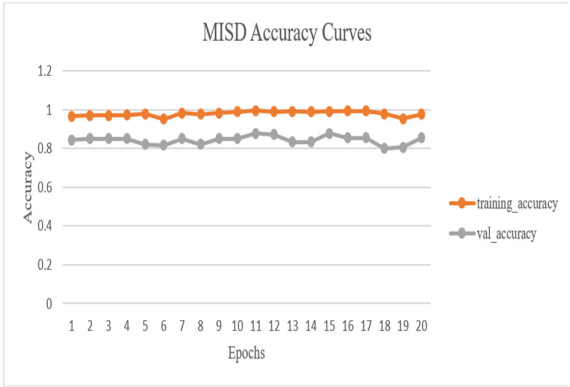

Fig.22: Accuracy values for MISD dataset

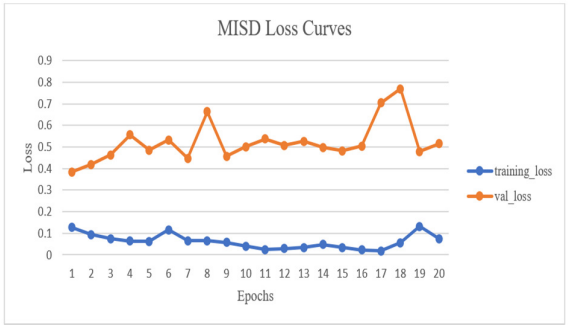

Fig.22: Loss values for the MISD dataset

6. CONCLUSION:

This study introduces Deep learning architectures designed for image forgery detection. The implemented model can efficiently differentiate between altered and original photos, categorizing candidate images into two groups: forged and original. Specifically, it identifies the manipulated portion within altered images. The approach involves extracting feature vectors from the image's features, and automatically detecting feature dependencies and correspondences through the utilization of the fully connected layer. To employ the proposed model for testing and identifying altered photos, it must undergo a training process. Additionally, utilizing one of the object detection techniques, Faster R-CNN, enables the identification of forged parts within images. The performance of the suggested model was evaluated using Standardized datasets including MICC-F2000, MICC-F220, COVERAGE, CASIA V2.0, and MISD. Notably, the proposed method achieved a remarkable 100% accuracy after 20 epochs when tested with the COVERAGE dataset.

REFERENCES

[1] K. M. Hosny, A. M. Mortda, M. M. Fouda and N. A. Lashin, "An Efficient CNN Model to Detect Copy-Move Image Forgery," in IEEE Access, vol. 10, pp.

48622-48632, 2022, doi: 10.1109/ACCESS.2022.3172273.

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

[3] M. T. H. Majumder and A. B. M. Alim Al Islam, "A Tale of a Deep Learning Approach to Image Forgery Detection," 2018 5th International Conference on Networking, Systems and Security (NSysS), Dhaka, Bangladesh, 2018, pp. 1-9, doi: 10.1109/NSysS.2018.8631389.

[4] A. Ross, S. Banerjee, and A. Chowdhury, ''Security in smart cities: A brief review of digital forensic schemes for biometric data,'' Pattern Recognit. Lett., vol. 138, pp. 346–354, Oct. 2020.

[5] M. M. Eltoukhy, M. Elhoseny, K. M. Hosny, and A. K. Singh, ''Computer-aided detection of mammographic mass using exact Gaussian–Hermite moments,'' J. Ambient Intell. Humanized Comput., pp. 1–9, Jun. 2018, doi: 10.1007/s12652-018-0905-1.

[6] F. Marcon, C. Pasquini, and G. Boato, ''Detection of manipulated face videos over social networks: A large-scale study,'' J. Imag., vol. 7, no. 10, p. 193, Sep. 2021, doi: 10.3390/jimaging7100193.

[7] K. Sunitha and A. N. Krishna, ''Efficient keypoint based copy move forgery detection method using hybrid feature extraction,'' in Proc. 2nd Int. Conf. Innov. Mech. Ind. Appl. (ICIMIA), Mar. 2020, pp. 670–675.

[8] S. Velmurugan, T. Subashini, and M. Prashanth, ''Dissecting the literature for studying various approaches to copy move forgery detection,'' Int. J. Adv. Sci. Technol., vol. 29, pp. 6416–6438, Jun. 2020.

[9] X. Tian, G. Zhou, and M. Xu, ''Image copy-move forgery detection algorithm based on ORB and novel similarity metric,'' IET Image Process., vol. 14, no. 10, pp. 2092–2100, 2020.

[10] J. Zhao and J. Guo, ''Passive forensics for copy-move image forgery using a method based on DCT and SVD,'' Forensic Sci. Int., vol. 233, nos. 1–3, pp. 158–166, 2013, doi: 10.1016/j.forsciint.2013.09.013.

[11] T. Chihaoui, S. Bourouis, and K. Hamrouni, ''Copy-move image forgery detection based on SIFT descriptors and SVD-matching,'' in Proc. 1st Int. Conf. Adv. Technol. Signal Image Process. (ATSIP), Mar. 2014, pp. 125–129.

[12] M. F. Hashmi, A. R. Hambarde, and A. G. Keskar, ''Copy move forgery detection using DWT and SIFT features,'' in Proc. 13th Int. Conf. Intelligent Syst. Design Appl., Dec. 2013, pp. 188–193.

[13] S. Dhivya, B. Sudhakar, and K. Devarajan, ''2-level DWT based copy move forgery detection with surf features,'' in Proc. 3rd Int. Conf. Commun. Electron. Syst. (ICCES), Oct. 2018, pp. 800–805.

[14] P. G. Singh and K. Singh, ''An improved block-based copy-move forgery detection technique,'' Multimedia Tools Appl., vol. 79, pp. 13011–13035, May 2020.

[15] J. Park, T. A. Kang, Y. H. Moon, and I. K. Eom, ''Copy-move forgery detection using scale invariant feature and reduced local binary pattern histogram,'' Symmetry, vol. 12, p. 492, Apr. 2020, doi: 10.3390/sym12040492.

[16] X. Tian, G. Zhou, and M. Xu, ''Image copy-move forgery detection algorithm based on ORB and novel similarity metric,'' IET Image Process., vol. 14, pp. 2092–2100, Oct. 2020.

[17] A. Diwan, R. Sharma, A. Roy, and S. Mitra, ''Keypoint based comprehensive copy-move forgery detection,'' IET Image Processing, vol. 15, pp. 1298–1309, May 2021.

[18] I. Amerini, L. Ballan, R. Cardelli, A. Del Bimbo, and G. Serra, ''A sift-based forensic method for copy–move attack detection and transformation recovery,'' IEEE Trans. Inf. Forensics Security, vol. 6, no. 3, pp. 1099–1110, Mar. 2011.

[19] I. Amerini, L. Ballan, R. Caldelli, A. Del Bimbo, L. Del Tongo, and G. Serra, ''Copy-move forgery detection and localization using robust clustering with J-Linkage,'' Signal Process., Image Commun., vol. 28, no. 6, pp. 659–669, Jul. 2013. M. Elaskily,

[20] H. Elnemr, M. Dessouky, and O. Faragallah, ''Two stages object recognition based copy-move forgery detection algorithm,'' Multimedia Tools Appl., vol. 78, no. 11, pp. 15353–15373, 2019.

[21] N. Mishra, S. Sharma, and R. Patel, ''Region duplication forgery detection technique based on SURF and HAC,'' Sci. World J., vol. 7, pp. 1–8, Nov. 2013.

[22] H. Kaur and J. Saxena, ''Simulative comparison of copy-move forgery detection methods for digital images,'' Int. J. Electron., Elect. Comput. Syst., vol. 4, pp. 62–66, Sep. 2015.

[23] K. M. Hosny, H. M. Hamza, and N. A. Lashin, ''Copy-move forgery detection of duplicated objects using accurate PCET moments and morphological operators,'' Image. Sci. J., vol. 66, no. 6, pp. 330–345, Aug. 2018.

[24] K. Hosny, H. Hamza, and N. Lashin, ''Copy-for-duplication forgery detection in colour images using QPCETMs and sub-image approach,'' IET Image Process., vol. 13, no. 9, pp. 1437–1446, 2019.

[25] K. Meena and V. Tyagi, ''A copy-move image forgery detection technique based on Gaussian-Hermite moments,'' Multimedia Tools Appl., vol. 78, pp. 33505–33526, Dec. 2019.

[26] C. Wang, Z. Zhang, Q. Li, and X. Zhou, ‘‘An image copy-move forgery detection method based on SURF and PCET,’’ IEEE Access, vol. 7, pp. 170032–170047, 2019.

[27] Y. Wang, X. Kang, and Y. Chen, ‘‘Robust and accurate detection of image copy-move forgery using PCET-SVD and histogram of block similarity measures,’’ J. Inf. Secure. Appl., vol. 54, Oct. 2020, Art. no. 102536.

[28] M. Elaskily, H. Elnemr, A. Sedik, M. Dessouky, G. El Banby, O. Elshakankiry, A. Khalaf, H. Aslan, O. Faragallah, and F. A. El-Samie, ‘‘A novel deep learning framework for copy-move forgery detection in images,’’ Multimedia Tools Appl., vol. 79, pp. 19167–19192, Jul. 2020.

[29] Abhishek and N. Jindal, ‘‘Copy move and splicing forgery detection using deep convolution neural network, and semantic segmentation,’’ Multimedia Tools Appl., vol. 80, pp. 3571–3599, Jan. 2021.

[30] Y. Rodriguez-Ortega, D. M. Ballesteros, and D. Renza, ‘‘Copy-move forgery detection (CMFD) using deep learning for image and video forensics,’’ J. Imag., vol. 7, no. 3, p. 59, Mar. 2021, doi: 10.3390/ jimaging7030059.

[31] N. Goel, S. Kaur, and R. Bala, ‘‘Dual branch convolutional neural network for copy move forgery detection,’’ IET Image Process., vol. 15, p. 656, Feb. 2021.

[32] A. Jaiswal and R. Srivastava, ‘‘Detection of copy-move forgery in digital image using multi-scale, multi-stage deep learning model,’’ Neural Process. Lett., vol. 54, pp. 75–100, Aug. 2021, doi: 10.1007/s11063-021- 10620-9.

[33] K. Dhananjay, S. Ahirrao, and K. Kotecha, ‘‘Efficient approach towards detection and identification of copy-move and image splicing forgeries using mask R-CNN with MobileNet V1,’’ Comput. Intell. Neurosci., vol. 2022, pp. 1–21, Jan. 2022, doi: 10.1155/2022/6845326.

[34] J.-L. Zhong and C.-M. Pun, ‘‘An end-to-end dense-inceptionNet for image copy-move forgery detection,’’ IEEE Trans. Inf. Forensics Security, vol. 15, pp. 2134–2146, 2020, doi: 10.1109/TIFS.2019.2957693.

[35] Y. Wu, W. Abd-Almageed, and P. Natarajan, ‘‘BusterNet: Detecting copy-move image forgery with source/target localization,’’ in Proc. Conf. Comput. Vis. Pattern Recognit., 2018, pp. 168–184. [Online]. Available:https://link.springer.com/conference/eccv

[36] A. Islam, C. Long, A. Basharat, and A. Hoogs, ‘‘DOA-GAN: Dual-order attentive generative adversarial network for image copy-move forgery detection and localization,’’ in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2020, pp. 4676–4685.

[37] Y. Zhu, Ch. Chen, G. Yan, Y. Guo, and Y. Dong, ‘‘AR-Net: Adaptive attention and residual refinement network for copy-move forgery detection,’’ IEEE Trans. Ind. Info