IMAGE FORGERY DETECTION USING CNN MODEL

A Dissertation submitted in partial fulfillment of the requirement for the

Award of degree of

MASTER OF TECHNOLOGY IN

INFORMATION SYSTEMS

Submitted By :

RASHI GUPTA

(2K20/ISY/17)

Under the esteemed guidance of

Dr. Ritu Agarwal

Assistant Professor

Department of Information Technology

Delhi Technological University

Bawana Road, Delhi-110042

2020-2022

CERTIFICATE

This is to certify that Ms. Rashi Gupta (2K20/ISY/17) has completed the major project titled "Image Forgery Detection Using CNN Model" as part of the Master of Technology degree in Information Systems specialization at Delhi Technological University.

During the academic session 2020-2022, the major project is a genuine piece of work that was carried out and finished under my supervision and guidance. This report's content has not been submitted anywhere else for the granting of any other degree.

(Project Guide)

Dr. Ritu Agarwal

Assistant Professor

Department of Information Technology

Delhi Technological University

Bawana Road, Delhi-110042

ACKNOWLEDGEMENT

I want to thank my major project guide Dr. Ritu Agarwal, Assistant Professor, lT Dept., Delhi Technological University, for her invaluable help and direction in completing this significant project. It gives me great pleasure to express my heartfelt gratitude to my esteemed mentor for his constructive critique and understanding, without which the project would not have taken the shape it has.

I respectfully express my thanks to the other faculty members in this department for their invaluable assistance and time whenever it was needed.

Rashi

Rashi Gupta

Roll No. 2K20/ISY/17

M.Tech (Information Systems)

E-mail: rashigupta28196@gmail.com

ABSTRACT

Image forgery detection has become more relevant in the real world in recent years since it is so easy to change a particular image and share it throughout social media, which may quickly lead to fake news and fake rumors all over the world. These editing softwares have posed a significant challenge to image forensics in terms of proposing and implementing various methods and strategies for detecting image counterfeiting. There have been a variety of traditional approaches for forgery detection, but they all focus on simple feature extraction and are more specialized to the type of forgery. However, as research advances, multiple deep learning approaches are being implemented to identify forgeries in images. Deep learning approaches have demonstrated exceptional outcomes in image forgery when compared to traditional methods.

The numerous sorts of image forgeries are discussed in this work. The work presents and compares different applied and proven image forgery detection approaches, as well as a comprehensive literature analysis of deep learning algorithms for detecting various types of image counterfeiting. Also CNN network is build based on a prior study and compare its performance on two different datasets to address this issue. Furthermore, the impact of a data augmentation approach is assessed as well as several hyperparameters on classification accuracy. Our findings imply that the dataset's difficulty has a significant influence on the outcomes. In this study, we have also aimed to determine detection of image forgery using deep learning approach. The CNN Model is used along with the ELA extraction model which is then used for detection of forgery in images. Later we also used two CNN Models, VGG16 Model and VGG19 Model for the better comparison and understanding.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

Is it possible to believe what we see? No. Despite the fact that a picture explain itself without words, image manipulation has never been easier, thanks to the ubiquitous provision and accessibility of image manipulation tools and software. Any software editing tools, such as Photoshop, PhotoScapeX, Adobe Lightroom, GIMP, and others, can readily manipulate images. The point is, you don't need any expert understanding to modify an image; image manipulation can be done by anyone, even if they aren't a professional.

As a result, it's easy to say, "Don't believe everything you see." Manipulation of images as well as the distribution of fake news on social media has never been easier. Keeping all of these considerations in mind, image forensics researchers are always developing new approaches and algorithms for detecting images and determining whether they are original or tempered.

Deep learning has recently gotten a lot of interest from forensic academics working under the subject of tampered image detection, with goal of detecting forgeries in photographs. Because traditional approaches for detecting picture counterfeiting have some drawbacks when compared to deep learning technologies. As a result, this work begins with a review of existing methodologies, followed by a discussion of deep learning algorithms that are more effective at detecting copy-move image forgeries.

1.2 IMAGE FORGERY DETECTION

Modification of a digital picture to disguise meaningful and useful information is defined as forgery detection., and detection of image forgery is to do identification of forgeries in the image [1].

Image forgery has become a critical topic that requires attention due to an increase in criminal activities. Picture editing software capabilities have enabled users to manipulate image content without being able to detect the difference between tampered and untampered photos with the naked eye, allowing them to spread false information. Furthermore, the primary goal of forgery detection in the digital era is to attain authenticity and integrity.

1.3 CLASSIFICATION OF IMAGE FORGERY DETECTION

Image forgery may be done in variety of ways, and the evolution of digital picture forgery has resulted in multiple varieties of forgeries done on images.

In digital picture forensics, there are essentially two approaches, first is active approaches and second is passive approaches. Both are made up of a number of approaches, as indicated in fig. 1.

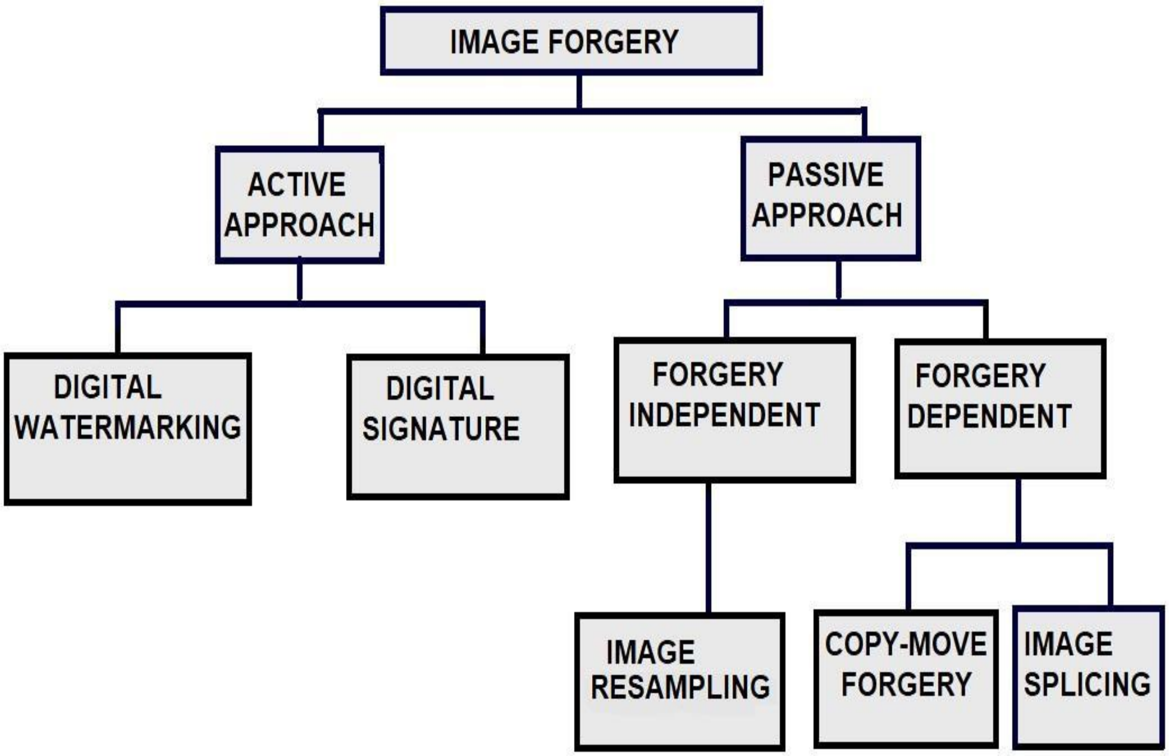


Figure 1: Classification of Image Forgery Techniques

1.3.1 ACTIVE APPROACH

In the case of Active Approaches, all the image knowledge is accessible ahead of time and is crucial for the authentication method. This approach is mostly used for data hiding, as some code is injected into the picture during the generation process [2]. By checking this code, the legitimacy of the image is validated. Digital watermarking is one of the two types of active authentication techniques., other is digital signatures. In the processing stage itself, digital watermarks are embedded inside the photos, while in case of the later, additional information is added by the digital signature during the capturing end which is generally obtained from the image [3].

1.3.2 PASSIVE APPROACH

Passive Approaches, often known as the blind approaches, is a method of authenticating photographs in which it requires only the image itself and without requiring any beforehand knowledge of the picture. The concept behind these methods is that even if there is no obvious evidence of manipulation, the underlying data will be altered.

The distinction between the two approaches, active and passive for is that passive forensics may evaluate a picture without knowing anything about it beforehand. As a result, it is better to use the passive forensic approach. Detecting digital forgeries in the absence of the original picture and without any pre-embedded watermark is the major aim of detecting these type of approaches, despite the fact that careful manipulation leaves no visual sign of change. The two categories of passive approaches are forgery dependent techniques and forgery independent procedures.

1.3.2.1 FORGERY DEPENDENT

Forgery-dependent detection techniques are intended for identify just certain forms of forgeries in picture, like the copy-move and image splicing, that are reliant on forgery type performed over the image.

1.3.2.1.1 COPY-MOVE FORGERY

Copy-Move forgery, other name is cloning, comprises of copying and pasting a portion in a picture's content to another area inside the same image. Doing so can be used to conceal critical information or to duplicate areas in a picture. Because the copied and pasted portion is from the same image, crucial attributes such as color, noise, and texture remain unchanged, making the detection procedure more challenging [4]. Because all of the distinct components, whether copied or pasted, belong to the same image, there is a significant correlation between them in copy-move forgery, which aids in the identification of forgery in the image.

Figure 2. Copy-Move Forgery Example

1.3.2.1.2 IMAGE SPLICING

Cut-and-paste technique is used in **Image Splicing** to cut and paste sections of the image content from one or more other photos to generate a new fake image [5]. As a result, a component of the image content in splicing forgery derives from other photographs. There can be two or more sources in the newly created image. If the splicing is done well, the boundaries between the spliced portions might be unnoticeable visually.

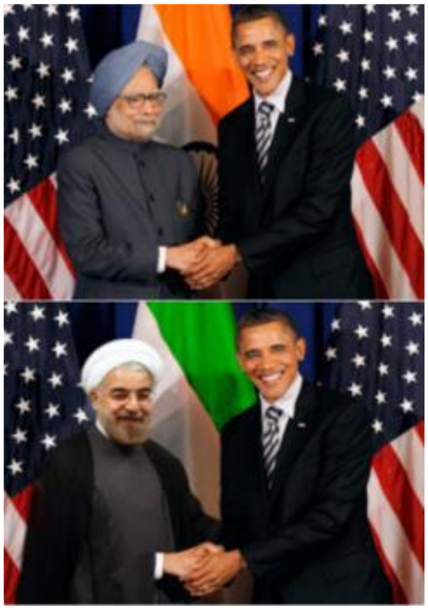


Figure. 3. Image Splicing Forgery Example

1.3.2.2 FORGERY INDEPENDENT

Forgery-independent approaches look for forgeries that aren't based on the sort of forgery, but rather on artefact traces left behind during the resampling process or owing to lightning discrepancies.

1.3.2.2.1 IMAGE RESAMPLING

Image Retouching is another name for **Image Resampling**. Particular processes are used in Image Resampling to enhance the image, decrease certain image features, or try to improve the picture's quality, which may be done to draw people's attention. Since image features are manipulated for tempering the image, it is the most extensively used and easiest to accomplish sort of image forgery. Rotation, resizing, sharpness, color contrast, modifying the brightness, and rotating, transformations in the geometry of images such as skewing, flipping rotation, can all be used to temper an image [5]. The interpolation stage is critical in the image resampling process since it results in significant statistical changes.

Figure. 4. Image Resampling Forgery Example

1.4 TRADITIONAL METHODS REVIEW

[6] developed a DCT-based approach that is more accurate. This is a more efficient method since every block of image is represented with fewer algorithmic characteristics. To get a lower-dimensional representation, [7] applied Principle Component Analysis (PCA) on the picture blocks.

[8], an automated method for identifying spliced forgeries, in which a severe examination of consistency notion of physical attributes is applied across different arbitrarily-shaped picture sections. Cross fitting and local picture features were generated using the CRF (Camera Response Function), after this SVM classification was done. With poor localization findings, the system accuracy was 70%.

[9] employs a block discrete cosine transform, in which DCT is applied to the picture as well as the duplicated portions are detected after applying the DCT. The DCT coefficients are then lexicographically ordered and sorted, and classified according to the similarity of blocks in the image having same spatial offset, allowing duplicated portions to be discovered.

[10] utilized DWT in conjunction with SVD to decrease the amount of data that was reviewed while also making the block representation more resilient. By using Zernike moments to tiny blocks of picture, a forgery detection approach was implanted to ensure that the copied regions in the image are localized. Later, [11] improved the work by employing the block-matching paradigm, which is a more dependable method since it is locality sensitive hashing based, which combines the phases of Zernike moments in a feature-based error reduction approach, resulting in higher robustness and performance.

[12] proposes a composite picture detection method based on resampling and JPEG compression traces. First, the picture is separated into overlapping halves. The following step is to create and evaluate a block measure factor. The block measure factor additionally takes into account the resampling and JPEG compression properties of each block. The approach suggested by [13] combines the undecimated wavelet transform (UWT) with the Zernike

moments. Scaling and other computations based on affine transforms are difficult for the algorithm to handle.

Using average pixel intensity of RGB channels, as well as some directional information, [14] proposed a method for extracting seven characteristics from each picture block, which is split across tiny overlapping blocks. [15] demonstrated how to employ the singular value decomposition (SVD) strategy for generating geometric invariant and and algebraic feature vectors and for doing the image authentication for copy-move picture fraud.

[16] developed a Discrete Wavelet Transform (DWT) for detection of tampering in images using pixel matching technique. The technique for evaluating the partitioning of recursively processed sub-images, is utilized for detecting the geographically localized regions of the copy-move forgeries in images.

1.5 DEEP LEARNING METHODS REVIEW

In recent years, the deep learning community's study has been seen as an ever-expanding field, with a vast network of scientists inspiring one another in various methods or methodologies. Many researchers from forensics have attempted in employing deep learning for image tampering identification, and the discipline of photo forensics has emerged in concert with this trend [17]. [18] presented a deep learning-based median filtering technique for detecting using CNN. The median filtering method learns and extracts features from the image automatically. The first researchers to use median filtering with CNNs for picture forensics were the author of this publication. This technique works effectively for median filtering detection in JPEG compression and tiny image blocks.

[19] proposed a novel concept that combines the work of convolutional and conventional layers. They primarily utilized prediction error filters to take out manipulation detection characteristics. In a separate paper, [20] present a data-driven manipulation parameter estimator that is independent of separately examining the estimate for each form of manipulation.

[21] developed a deep learning-based method for detecting image forgery. When RGB color pictures were utilized as input, the CNN was employed to learn the hierarchical structure. They utilized CNN to identify image counterfeit in the form of spicing and copy-move forgeries.

The CNN model presented by [22] achieves a very good performance in automatically detecting image counterfeiting from a computer using a basic image under different copy move tampering actions. [23] used CNN to identify picture forgeries resulting from splicing, retouching, and recompression. They came up with a suggested architecture that includes 5 convolutional layers, 2 fully connected layers, and a softmax classifier. A novel deep learning approach based on CNN has been proposed by [24]. CNN is employed in this case to accurately identify the traces left by the change. They added filter layer to the input image with the goal of suppressing the main content. [25] proposed the Convolutional Kernel Network (CKN) as a novel deep learning network for identifying image forgeries. After completing thorough testing, it was discovered that this suggested CKN outperformed previous features which are hand-crafted and that it also provide better results using GPU-based CKN.

A dual-domain-based Convolutional neural network (D-CNN) approach was proposed by [26]. It's essentially a CNN-based unified architecture. Sub-SCNN and Sub-FCNN are the two subnetworks they came up with. Sub-goal SCNN's is to detect and find image forgery, followed by Sub-goal FCNN's of presenting statistical characteristics. The suggested technique provides better accuracy while also eliminating significant processing times for the training process. For the detection of copy-move picture fraud, [27] presented an end-to-end DNN solution. An input picture is fed into a convolutional feature extractor, which extracts block-like features. The feature extractor VGG16 is utilized. At the end, bilinear up sampling is employed to improve the image's resolution. The suggested model had the disadvantage of not being accurate with pure texture pictures.

To recognize and locate image modification [28] uses two strategies. A deep neural network is utilized in the first approach to categorize modified photos. A long short-term memory (LSTM) network is used in the second strategy to learn the connection or frontier change between neighbouring blocks as well as the current resampling blocks of data.

1.6 Comparative analysis of several Deep Learning Techniques for detecting Image Forgery:

Paper	**Details**	**Dataset**	**Performance**
Liu, Guan, and Zhao 2017 [25]	Used CKN as deep learning network. Local descriptor driven by data, GPU-based adaptive over segmentation, resilient to change in brightness, Gaussian Blurring, post-processing, noise, and transformations.	CoMoFoD	$F1$ = 0.5997
Ouyang, Liu, and Liao 2018 [22]	ImageNet was used to complete the transfer learning. For copy-move forgery detection, used an existing model that was not resilient to real-world scenarios.	Oxford	Error = 2.32%
Bunk et al., n.d. [28]	CNN and LSTM are employed. JPEG quality, rescaling, rotation, and shearing are used to detect and locate tampering using resampling characteristics and deep learning.	NIST Nimble 2016	Accuracy = 94.86%
Chen et al. 2015 [18]	Model of a convolutional neural network with an extra filter layer. MFR-CNN is used to identify median filtering. Automatically, features are represented and learnt.	Composite using, UCIDBOSSBase, Dresden, NRCS, BOSS RAW	Accuracy = 96.84%
Bayar and Stamm 2016 [19]	The CNN model is employed. There is a multi-class classification. Multiple modifications, such as gaussian blurring, median filtering, noise, and resampling, have been found.	Synthesized dataset	Accuracy = 99.10%
Bayar and Stamm 2017 [20]	The CNN model is employed. Four distinct tampering procedures are identified using data-driven parameter estimation: Gaussian Blurring, JPEG compression, Resampling, gaussian blurring, and median filtering.	Dresden based synthesized	Accuracy = 95-99%
Kim and Lee 2017 [17]	The CNN model has been created. To obtain hidden characteristics in a picture, a high pass filter is employed. The following procedures are performed: Gaussian Blurring, median filtering, AWGN, and Re-Sampling.	BOSSBase 1.01	Accuracy = 95%
Wu et al. 2018 [27]	The CNN model is employed. The VGG16 model is used to extract features. Pure texture image is poor in the end-to-end DNN solution.	CASIA V2.0	$F1$ = 75.72
Rao and Ni 2016 [21]	SRM-CNN model was employed. In SRM, feature fusion for SVM classification, residual maps are employed.	CASIA V1.0	Accuracy = 98.04%

CHAPTER 2

PROBLEM STATEMENT

The purpose of this research is to see detection of image tampering using CNN's performance fluctuates depending on the sample complexity. To that purpose, a categorization pipeline is created based on [29]'s work. While their study has a high level of accuracy, the CASIA datasets are employed for the testing purpose of network and have been modified in a way that humans can detect. As a result, we'd like to see how such a CNN performs on a more difficult dataset. Its performance, in our opinion, will deteriorate dramatically. Two datasets are chosen to train the CNN on in order to validate this intuition. The CASIA v2 dataset is preferred over CASIA v1 dataset since the latter is considered easier because it contains more samples. This is crucial since previous research has shown that a relatively well-trained CNN requires a huge number of training data. CASIA v2 contains 12,622 photographs, with authenticated and changed shots divided 60-40. The same architecture is trained and assessed on the NC16 dataset, which comprises far more difficult-to-identify photos, based on the CASIA dataset's findings. The NC16 collection has 1,124 photographs with a 50-50 split.

Moving on to model evaluation, the major evaluation metric is the accuracy of the classifier during the test phase. The reason for this decision is that it is the primary parameter utilized in previous deep learning research to assess picture forgery detection ability. Furthermore, visual inspection of the misclassified pictures and analysis of the model's confusion matrix provide further insights on its behavior.

CHAPTER 3

TECHNICAL APPROACH

One of the key goals of our research is to develop a pipeline that can distinguish manipulated from genuine photos. As a result, this work is influenced by the architecture presented by Y. Rao et al [29]. They propose a CNN that may be used as a feature extractor, taking input, the patch of the image and producing a representation of $Y = f(X) \in R^K$, where K is the number of dimensions. The characteristics are then loaded into an SVM classifier, which predicts whether they relate to an original or modified picture. The next sections describe the network design as well as the training technique for the CNN and SVM.

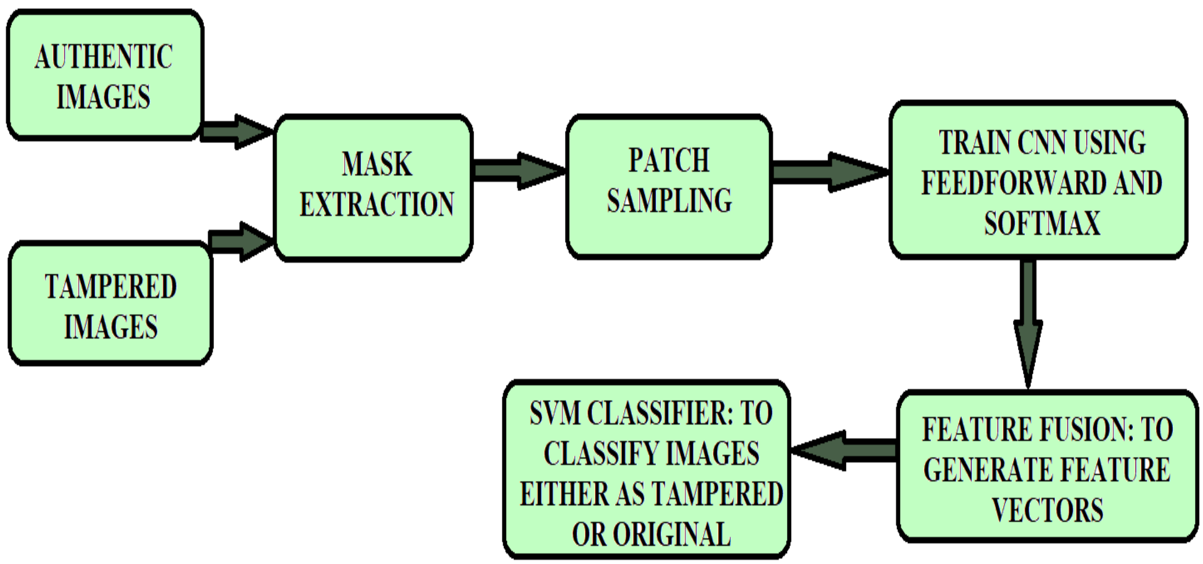


Figure 5: Flowchart for the Implementation

3.1 NETWORK ARCHITECTURE

CNNs (Convolutional Neural Networks) are a sort of deep neural network that is mostly utilised for image processing. In the basic structure of a CNN, several convolutional layers are followed by fully connected layer(s) and a softmax classifier. A convolution, a non-linear activation, and a pooling make up each convolutional layer. Feature maps are arrays that are used as the input and output of convolution layers.

As illustrated in Figure 3, the architecture used in this work is a CNN with nine convolutional and two max-pooling layers. The network's input size is a 128x128x3 patch, with 3 representing the RGB colour channels. The first two convolutions each produce 3 and 30 kernels and have a 5x5 kernel size. After these layers, a 2x2 filter is used to pool the data. The next eight layers each feature 16 kernels, with the convolutions using a 3x3 kernel size and the max pooling using a 2x2 filter.

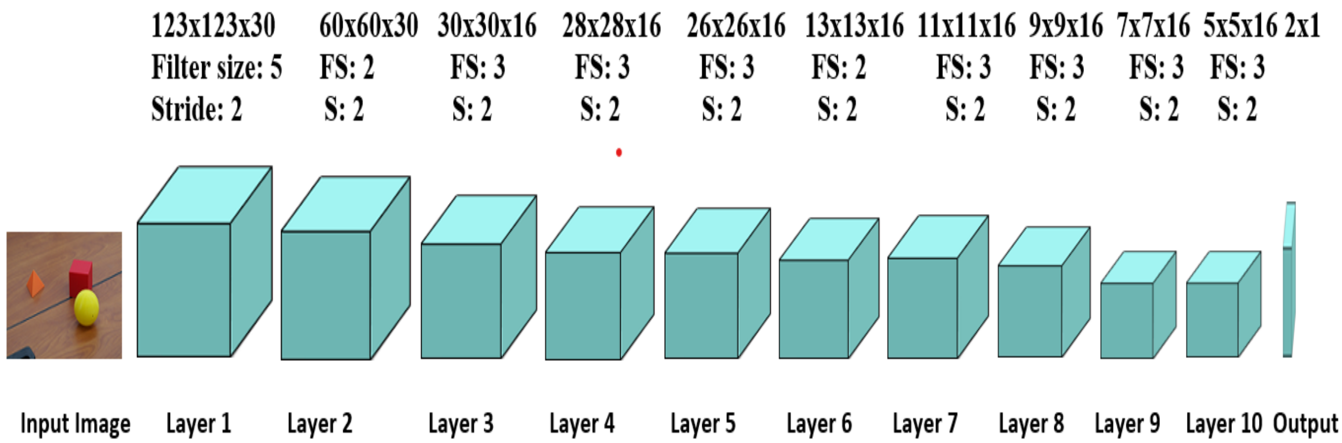


Figure 6: Architecture of the implemented 10-layer CNN.

3.2 NETWORK WEIGHT INITIALIZATION

Every other convolutional layer in our network is initialized via Xavier initialization, with the exception of the second convolution. The key idea is that high values or values that evaporate to zero are avoided. This is accomplished by maintaining the variance with each successive layer.

The kernels of the second convolutional layer are initialized, similarly to [29], using thirty SRM high-pass filters suggested in [30]. Eight first, four second, and eight third order SRM filters were employed. The filter becomes more sensitive to changes on the edges as the order increases. Apart from the aforementioned, two 3x3 and 5x5 square high pass filters are utilized to identify pixels with different values than their neighbors. Finally, eight edge detection filters are utilized that are the best at locating edges: 3x3 (4) and 5x5 (4). The primary reason for this is that altered photos may have irregular edges that do not mix in with their surroundings. As a consequence, the transition from one pixel region to the next would be dramatic, allowing our algorithm to recognize faked pictures more effectively.

In our work, we shuffle the filters on all channels, resulting in each RGB channel having a distinct filter in each dimension, as suggested by [29]. Due to its regularization impact, this has been empirically demonstrated to improve performance.

3.3 CNN TRAINING

Image patches must be collected from the dataset in order to train the aforementioned CNN architecture so that it can focus on the local areas of the artefacts and learn to recognize them. The extracted patches are 128x128x3, which means that each color channel has its own 128x128 patch. For the whole image, a patched-size sliding window with stride equal to eight was used to extract the data. After that, the altered patches are separated from the ones that haven't been tampered with. In terms of the tampered patches, each patch is compared to the mask of this picture's corresponding patch (from the same location of the image) and maintain the ones that include part of the tampered region, as shown in Figure 5. Furthermore, because training the CNN with a large number of extracted patches would be computationally costly, only two random tampered patches each picture is chosen. When it comes to the non-tampered patches, the same approach as before, but this time the comparable legitimate picture and two patches at random are chosen. Finally, the patches extracted are enhanced by rotating them four times by a step of 90 degrees to boost CNN's generalization capacity and minimize overfitting.

Figure 7: Mask generated using the tempered and original Images

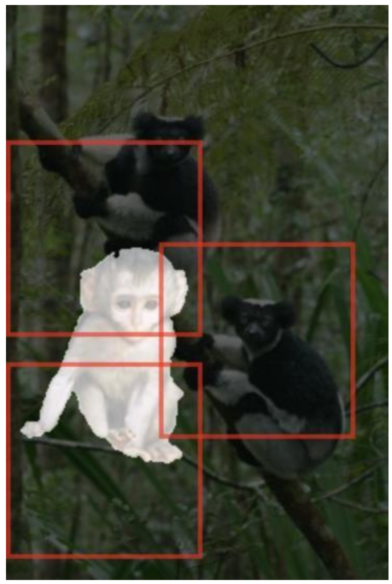


Figure 8: An example of patch extraction

The patches are input into the CNN after the aforementioned technique, which extracts a 400-D (5x5x16) feature representation of the patches. These characteristics are then sent to a fully-connected layer that employs a 2-way softmax classifier with dropout [26]. The neurons in the fully-connected layer, in particular, are set to zero with a chance of 50%. To limit the number of parameters, only one fully-connected layer is employed.

3.4 SVM TRAINING

The SVM classifier must be trained after the CNN network has been trained. To accomplish so, a sliding-window with stride s is used to scan the whole picture and extract every feasible ($p \times p$) patch from both the original and tampered images. This technique generates n new patches each picture, which are then processed by the CNN to generate n feature representations Y_i (400-D). However, before being supplied as an input to the SVM, these representations must be fused into a single Ŷ [k] representation for each picture.

Max or mean pooling is done to each dimension of Yi over all of the n patches taken from each picture. The SVM then uses the 400-D feature vector to identify photos as either original or altered.

CHAPTER 4

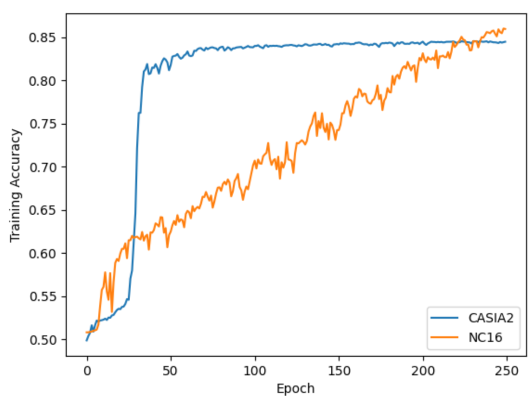
EXPERIMENT AND ANALYSIS

The following is the experimental workflow. To be more specific, we start by extracting the CNN training patches as described in Section 4.3 and using them to train the neural network. Then, using mean fusion as stated in Section 4.4, we extract new patches and their accompanying picture characteristics. The SVM is trained and evaluated using stratified 10-fold cross-validation once the features are collected. The patches necessary to extract the picture features are obtained using a stride s of 128 and 1024 for CASIA v2.0 and NC16, respectively, for the hyperparameters utilized in our studies. The picture size in NC16 is around 10 times greater than in CASIA v2.0, which accounts for the difference in stride. As a result, we chose a bigger stride to maintain the same number of patches extracted every image. All of the CNNs are trained for 250 epochs using cross-entropy loss and Stochastic Gradient Descent to optimize the network (SGD). The SGD implementation employs a momentum of 0.99, a weight decay of 5x104, and a decaying learning rate that drops by 10% every ten epochs. These parameters were chosen for each CNN trained because they were shown to increase the convergence of the network during early tests.

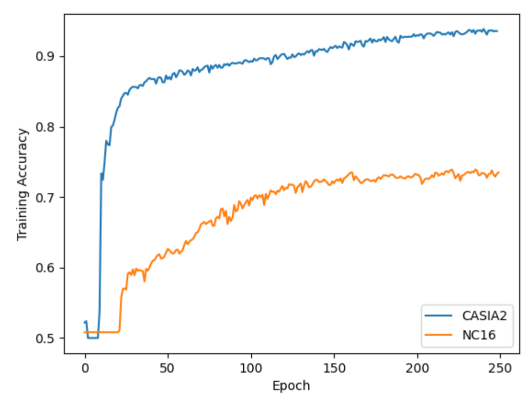
Finally, for each run of the SVM, we employed the RBF kernel and optimized the C and hyperparameters using exhaustive grid search. The following is how the remainder of the section is organized. To begin, we'll go through the various CNN networks that were developed. After that, we run a number of tests to see how the network performs on a variety of datasets and how generalizable it is. Then we investigate the impact of various hyperparameters on classification. Finally, we examine the misclassified samples visually and determine why the network failed in these instances.

4.1 NETWORK TRAINING

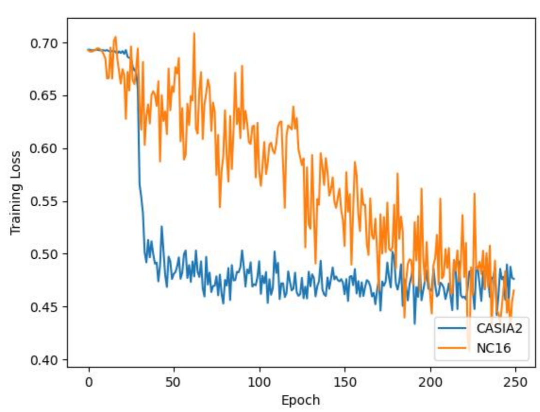
We trained four distinct networks and compared their classification performance using the network architecture specified in Section 4. More precisely, we used patches from each of the two datasets, CASIA v2.0 and NC16, to train a network with both augmented (four rotations) and non-augmented data. Figure 6 shows the training loss for each of the four setups stated above. In terms of the non-augmented data training loss, it is obvious that after 400 epochs, the loss for both datasets are nearly identical (≈0.45). The one for NC16, on the other hand, has progressively fallen, whilst the one for CASIA v2.0 has quickly decreased since the first epoch and has already achieved its minimal value after around 50 epochs. This pattern shows that for the latter, we might have used early stopping, but for the NC16 dataset, we could have trained the network for longer epochs and had better results.



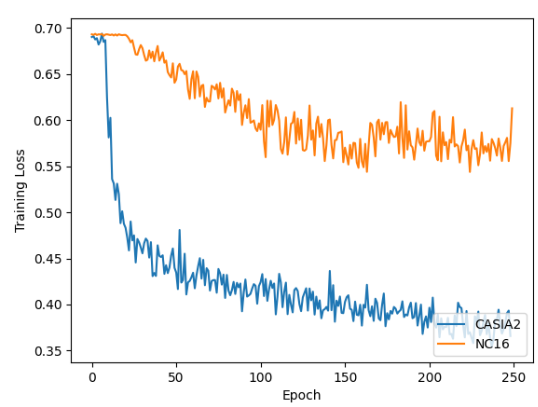
(a) Non-Augmented Data– CASIA v2.0 vs NC16



(b) Augmented – CASIA v2.0 vs NC16



(b) Non-Augmented Data– CASIA v2.0 vs NC16



(b) Augmented – CASIA v2.0 vs NC16

Figure 9: Training Accuracy and Loss comparison of CASIA v2.0 and NC16 with augmented vs non-augmented data

When it comes to networks with augmented data, CASIA v2.0 has a substantially smaller loss (≈0.4) after 400 epochs than the NC16 (≈0.6) counterpart. CASIA v2.0's loss, in particular, declines fast in the early epochs, then decreases at a slower rate until it reaches a plateau in the last epochs. For the first 100, the loss for NC16, on the other hand, gradually reduces until it reaches its lowest amount (≈0.6). As a result, an early termination approach might have greatly shortened the training time for the NC16 dataset while having no effect on its performance.

Table 1 lists the hyperparameters that were chosen for each dataset. To be more specific, the batch size and learning rate for each CNN model are adjusted so that it could correctly train. The settings chosen ensured that the SGD made the same number of gradient steps across all networks.

4.2 DATASET COMPARISION

A comparison of the classification performance of the two datasets utilized in this experiment is done in this section.

Table 1: CNN Hyperparameters

DATASET USED	BATCH SIZE	LEARNING RATE
CASIA V2.0	200	0.0005
NC16	32	0.001
CASIA V2.0 (AUGMENTED)	128	0.001
NC16 (AUGMENTED)	128	0.001

4.2.1 CASIA v2.0 vs NC16

First, the CNN is trained using the CASIA v2.0 dataset with a batch of size 200 pictures , having a LR(learning rate) of 0.0005. C = 1 and γ= 0.0001 were the optimum SVM hyperparameters for us to train on. With the preceding values, the categorization accuracy was 92.54 Table 2 also contains the equivalent confusion matrix, which was calculated using a random 80-20 split. In particular, the SVM successfully categorized 1,426 tampered and 1,008 non-tampered images, with just 17 tampered and 72 authentic images misclassified.

The second dataset is then used to train the feature extractor (CNN) using a learning rate of 0.001 and a batch of size of 32 pictures. Following the grid search, the best SVM parameters were C =100 and γ=0.001, resulting in an accuracy of 83.29%. It's worth noting that our system's classification accuracy on the more challenging NC16 is 10% lower than CASIA v2.0. The output of Table 3 was generated by constructing the confusion matrix in a similar manner as CASIA v2.0. In particular, 100 tampered and 94 original photos were accurately categorized, whereas 13 false negatives (FN) and 18 false positives (FP) were discovered (FP). Both of the preceding confusion matrices show that we get more FP than FN with this network topology. Both of the preceding confusion matrices show that we get more FP than FN with this network topology. Given the nature of the picture forgery detection problem, we can't say which of the two is more significant because it all relies on the case study. The results of the other tests all pointed to the same FP/FN behavior. As a result, in each of the following trials, the confusion matrix is not required.

Table 2: CASIA v2.0 – Confusion Matrix

CASIA v2.0	**Predicted Authentic**	**Predicted Tampered**
Actual Authentic	1,426	72
Actual Tampered	17	1008

Table 3: NC16 – Confusion Matrix

NC16	**Predicted Authentic**	**Predicted Tampered**
Actual Authentic	94	18
Actual Tampered	13	100

4.2.2 CASIA v2.0- Augmented vs NC16- Augmented

The same tests are then run on the supplemented datasets, with the photos rotated four times by a 90-degree step each time. Both datasets benefitted from the usage of the augmentation. CASIA v2.0 accuracy increased from 91.87% to 96.82%, but NC16 accuracy improved very slightly from 81.31 % to 84.89 %. Table 4 summarizes the outcomes of the non-enhanced and augmented tests.

Table 4: Augmentation Effect On CASIA v2.0 & NC16

Data Augmentation	CASIA v2.0 (%)	NC16 (%)
With	96.82	84.89
Without	91.87	83.31

To summarize, the picture forgery detection system we trained and tweaked for each dataset performed adequately, based on the prior findings. Furthermore, regardless of the use of augmentation, the NC16 dataset achieves worse accuracy than the CASIA v2.0 dataset, since it comprises samples with purposefully modified tampering sections.

Furthermore, when comparing classification with and without rotations, it is clear that supplementing the data improves the system's performance regardless of the dataset. However, using it improves CASIA v2.0 accuracy by 4.28%, compared to just 1.60% for the NC16.

4.2.3 GENERALIZATION PERFORMANCE

Another intriguing experiment was intended for training the CNN model using CASIA v2 enhanced dataset and then test it with the NC16 and vice versa. These two tests allow us to reason about the proposed system's generalization performance. To be more specific, we used the CASIA v2 pretrained CNN to extract the features for the NC16 dataset, and then used the SVM to classify the pictures.

The observed findings clearly illustrate that the suggested model does not generalize well to fresh data with a different underlying distribution. The accuracy of the NC16 network with the CNN network trained on the CASIA v2.0 was 67.54%, whereas the accuracy of the CASIA v2.0 with the NC16 network was 59.81%. However, within the 10 folds, the NC16 accuracy had a standard variation of 16.01%, ranging from 37.16% to 86.60%, indicating that its performance is highly dependent on the test fold chosen.

4.3 EFFECTS OF HYPERPARAMETER TUNING

Experiment is done on three distinct hyperparameters in this section: the CNN's initial learning rate, the stride used to extract the patches for feature extraction, and the feature fusion approach.

4.3.1 DIFFERENT LEARNING RATES

To begin, we train three networks with varying learning rates to see how the CNN learning rate affects system performance. It's worth noting that we used the patches derived from CASIA v2.0 without any data augmentation for these tests. The primary reasons for this decision are that the network performs better on CASIA v2.0 and because training it for this version is computationally cheaper than training it for the enhanced version since it takes about 75% less time. Table 5 shows the findings for learning rate values of 0.0001, 0.0005, and 0.001.

Table 5: Comparison Based on Learning Rate

Learning Rate	CNN Accuracy (%)	Testing Accuracy (%)
0.0001	87.35	91.49
0.0005	83.54	91.56
0.001	84.43	91.78

Based on the preceding findings, we infer that, despite the disparities in CNN training accuracy for the three distinct learning rates, there are no significant changes in SVM test accuracy.

4.3.2 MAX vs MEAN FEATURE FUSION

The technique we use to merge the values of all the patches for each of the 400 characteristics is the next parameter we experimented with. The two strategies we tried, as mentioned in Section 3.4, were mean and max fusion. Because it is the arrangement that yields the best accuracy so far, we utilized CASIA v2.0 with augmentation as our dataset.

Table 6: Mean vs Max Fusion on CASIA v2.0 (Augmented)

Fusion Method	Testing Accuracy (%)
Mean	96.14
Max	96.79

Furthermore, the same hyperparameters are applied to get the best test accuracy. The results in Table 6 were obtained by comparing the performance of the mean vs max fusion. As can be seen, the variation in accuracy is insignificant (0.08 %). The standard deviation for the max fusion approach, on the other hand, decreases significantly.

4.3.3 64 vs 128 FEATURE FUSION STRIDE

Finally, when we extracted the patches for the testing step, we experimented with different stride sizes. We utilized the upgraded patches of CASIA v2.0 and the model hyperparameters that yielded the greatest results thus far, just as we did in the last experiment. As a consequence, we chose the version that used maximum fusion and examined two distinct stride values, 64 and 128. Table 7 depicts the categorization findings achieved.

Table 7: 64 vs 128 Stride Size in Patch Extraction During Testing

Stride Size	Testing Accuracy (%)
64	96.71
128	96.91

Based on the preceding findings, we can infer that these two stride values have no effect on the classifier's performance, since the accuracy and standard deviation are almost identical in both circumstances.

4.4 ERROR LEVEL ANALYSIS (ELA)

ELA is a technique that evaluates images using varied levels of compression. This approach helps in identifying digitally changed images. There are several ways for distinguishing between genuine and fake photographs. [31] proposed a method for distinguishing counterfeit area for getting the image information using the video as an input. [32] developed a system for detecting the camera filter mode for predicting the error between the current and the falsified image by giving the ELA to as input while preprocessing .The changed regions may be discovered by using the classification model. [33] developed a technique based on ELA for identifying picture to be forged or not by employing automated wavelet soft-thresholding to filter any components which are noisy. [34] presented a novel technique in which they used hybrid of ELA and CNN to address the problem of distinguishing genuine from false photographs.

To create a face-swap photo dataset, [35] proposed a way for reproducing a technique of manufacturing and getting the deepfakes images as faces, they also used a feature extraction technique for having to embed the deep learning and doing the preprocessing of the high enhanced images using ELA.

A course is utilised to enhance the CNN version's preparation talent by converting basic data to ELA outcomes. This productivity may be carried out in light of the fact that ELA images include records that aren't usually as necessary as their individual image.

4.4.1 EXPERIMENTAL WORK

Figure 1 depicts the total system architecture. First how the pre-processing of data is done then the ELA part to enhance the images more, and then further using CNN model for the training purpose. The next section delves into these modules in further depth.

a) Data Pre-processing: Images are normalised during pre-processing. The goal of normalisation is to ensure that the data distribution across all photos is consistent. The entire dataset is scaled to 128x128 pixels for standardisation.

b) Error Level Analysis (ELA): Before converting to ELA, on receiving the images after the pre-processing from the last step is to brighten the images by whitening or brightening the images for further processing the images in next steps. When resaving images, consider both real and counterfeit photographs which were received from the last step. Making the pixels to be of a certain level the images are then passed for the training purpose to the neural network. The pre-processed shots are compared to newly generated images with the intention of making comparison and understanding the differences they make. The newly generated images are slightly brighter than the previous ones. Because the images received from the ELA after processing contains only the information which is not similar with the prior image. On using the pre-processing step with the error level analysis helps the CC model to optimize very nicely and as a consequence, our CNN training optimises after having just 10-15 epochs along with LR(learning rate) of 0.0001.

Image size must then be adjusted. For normalising the CNN training we have divided each cell value by 255.0. Next, each image is classified, with 0 being a real image and 1 representing a manufactured one. The photos are then sorted into two categories: We used an 80-20 ratio for the training and validation sets, respectively and also to ensure that the CNN is working appropriately.

c) Network Training : The convolutional layer serves as a feature extractor and represents the features. And pooling layer reduces the output map of the convolution layer to avoid overfitting. Many convolutional are lined up and pooling layer precede the fully connected layer. The max pooling layer is of size 2x2. The convolutional layer is having a kernel size and no. of filters as 5x5 and 32 respectively. The dropout of .025 is used to prevent the issue of overfitting from the pictures. This dropout is inserted during the max pooling layer. Feature vector is transformed using the softmax activation function in certain manner.For the purpose of recognition of pattern in the layers of the training model, the fully connected layer is used.

   Training optimization tool is RMSPprop , using this we can automatically change the learning rate for each parameter without human intervention, allowing us to optimize number of hidden layers, and other features as well. The neural network was trained using two standard neural architectures, notably VGG16 and VGG 19.

Two different ELA (Error Level Analysis) algorithms are employed for pre-processing. Fine tuning is done using models, and incorrect results are displayed in a pie chart with the percentage of forged and unforged images. The NC16 and CASIA v2 datasets are combined in this suggested model.

To detect whether falsified photographs are false or not in the datasets, we utilise a model that is trained first and then tested. Despite the fact that the forged image seems to have no more information than the original image to the naked eye, our recommended approach enhances training efficiency.

Only the part of the whole images is brought into focus to fulfil the purpose of determining the forgery. Further, because the part of the image which is forgets, it’s pixels are highly different when compared with the other parts of the image, with a distinct difference, the image transformed by ELA improves the training model's performance. As previously stated, before detecting if the pixel data is faked, the CNN training is definitely require to train the network for all sorts of images. Because the forged photos formed would emphasise parts of the actual picture, only convolution layers are required in the framework we use.

4.5 VGG16

VGG16 is a form of neural network that includes 22 layers, the last layer that output is the the Softmax classifier which helps in discriminating the original and the fake image. The output of the enhanced images received from the ELA is passed to the VGG16 for training puspose. This model has 16 layers for making the datasets to be better trained as the training set is distributed into 13,000 images for the real and 6,500 for the false images.

The VGG16 architecture has 2 convolutional layers, then comes 1 maxpool layer with stride of 2 and size of 2, then 2 convolutional layers having 128 channels, and another maxpool layer with stride of 2 and size of 2, 3 convolutional layers having 256 channels, 1 maxpool layer of 2x2 , 3 convolutional layers having 512 channels, and finally maxpool layers. Lastly, for filtering all the values RELU layers are added to every phase having value less than zero, then we will pass the ouput to the dense layer with the purpose of flattening it. The Output from the softmax layer will result out to be either 0 or 1 based on how well the input fits into the model while the training phase.

4.6 VGG19

VGG 19 is a 24-layer neural network with a similar temporal distribution as VGG 16. After ELA, the result received from pre-processing stage is sent to the 19-layer VGG19 model. The photos are then categorised using SoftMax activation, with the training set id being used to compare the images to the testing set and then returning the probability distribution classes for the image sets. Having 2 convolutional, 1 pooling, followed by 2 convolutional and a pooling, and the stack repeats again. The softmax is used as the output layer . Here we are achieving a training accuracy of 95.12% with a learning rate of 0.0001 and 15 epochs.

4.7 RESULTS AND DISCUSSION

For training photos, each model is meant to train on a mix of CASIA v2 and NC16 datasets. The dataset is made up of different images from two datasets that are divided before being fed into our CNN models for training and testing purposes, which classify them as real or manufactured. Initially we are to divide the dataset into two sections: true and false photographs. As a consequence, for training 13568 photographs are there using a 50-50 mix of genuine and altered images using all the three models. Similarly, each model was put to the test with a total of 1000 images, 500 of which were genuine and 500 fake. Figure 10-12 shows the respective train and test result on passing the datasets.

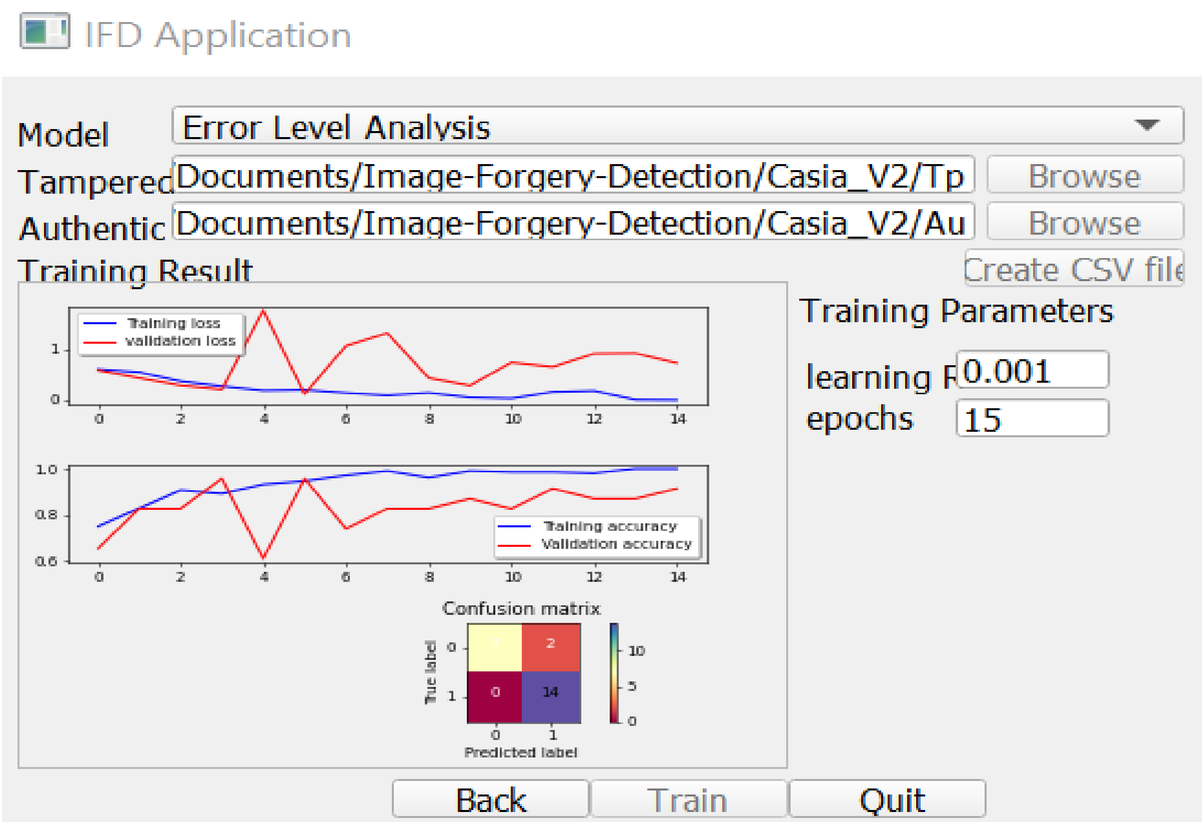


Figure 10 : ELA training dataset

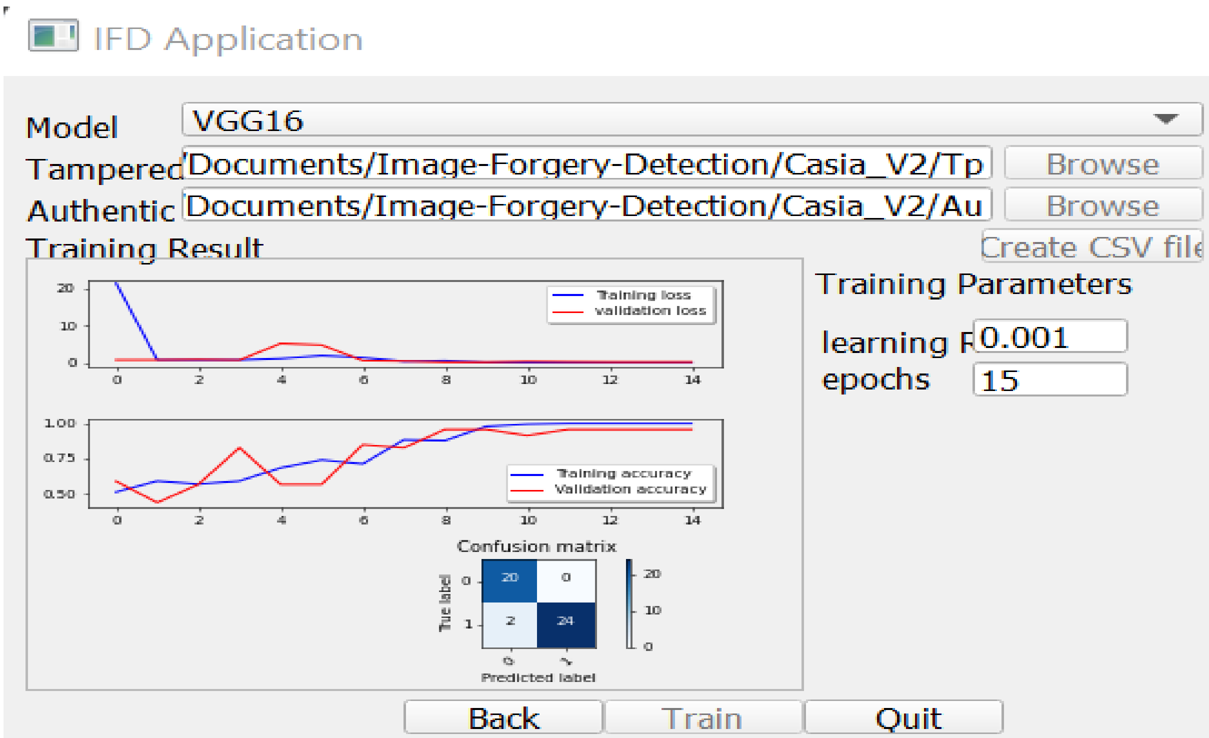


Figure 11 : VGG16 Training Dataset

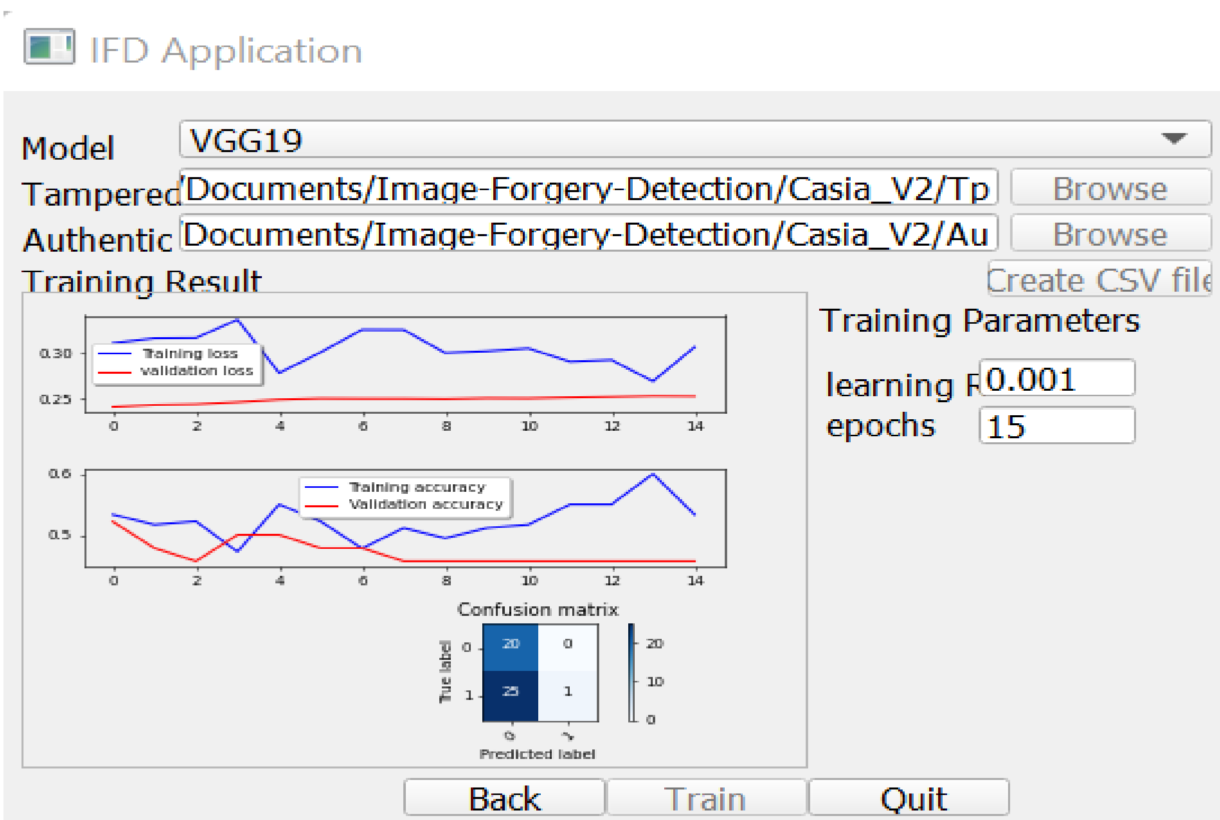


Figure 12 : VGG19 Training Dataset

According to the confusion matrix, VGG16's training accuracy is 93.21% with a learning rate of 0.0001 and 15 epochs.

With an LR rate of 0.0001 and 15 epochs, the confusion matrix has a training accuracy of 95.12% for VGG19. When training is done, all three neural models commence. The 50% criterion was also maintained for all three models, meaning that if the CNN predicts an authenticity accuracy of more than 50%, it will be considered as legitimate, else it will be generated. Figure 13-18 displays an example of test result obtained on testing images over different models.

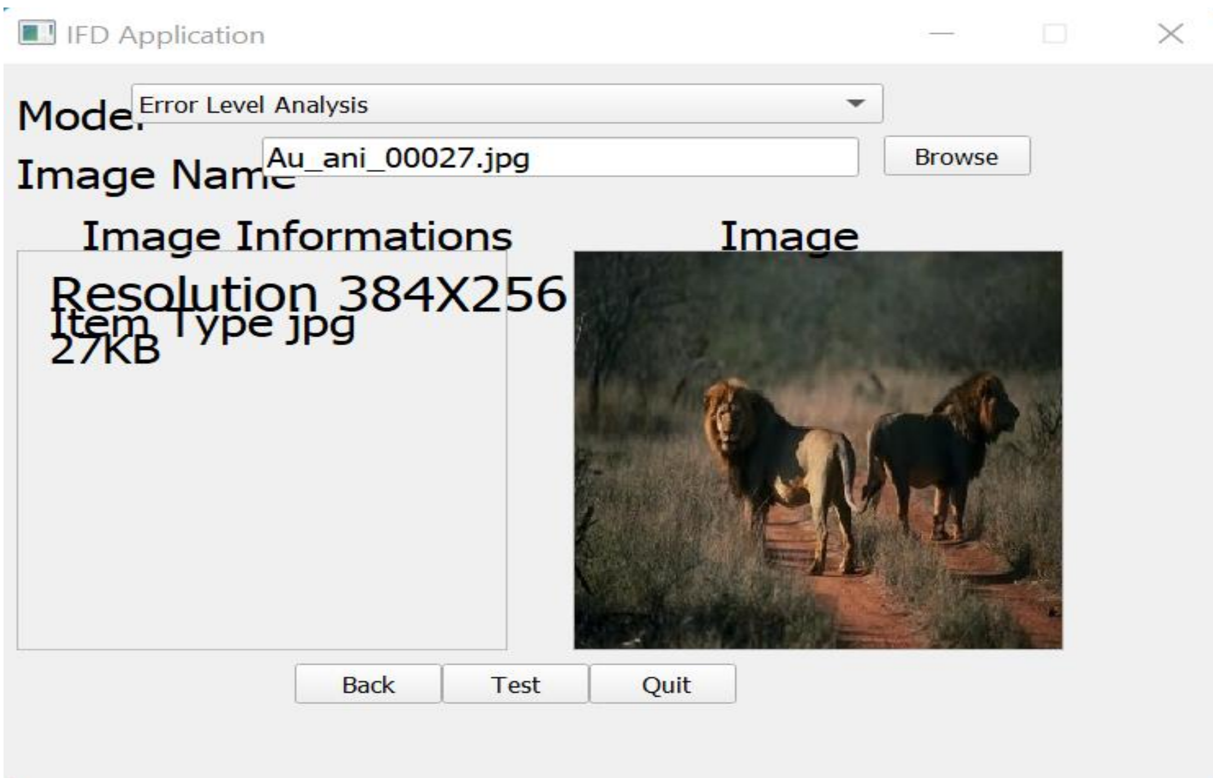


Figure 13 : ELA Testing Image

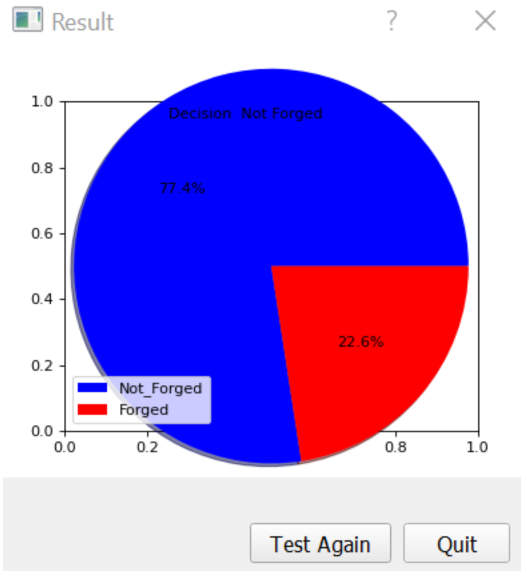


Figure 14 : ELA Test Result

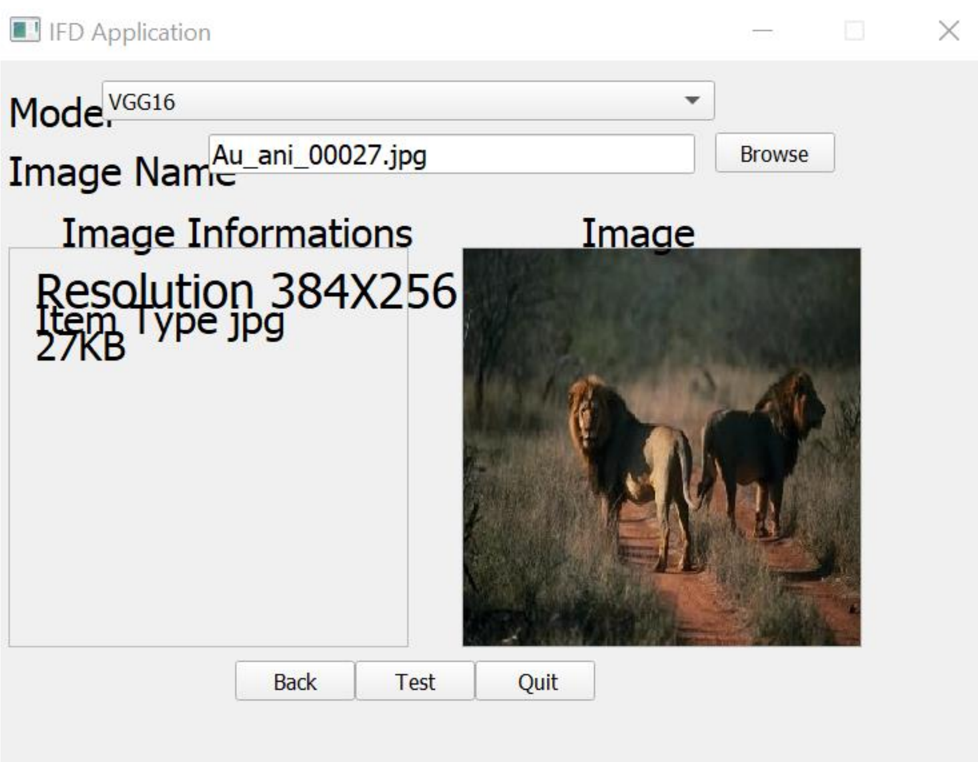


Figure 15 : VGG16 Testing Image

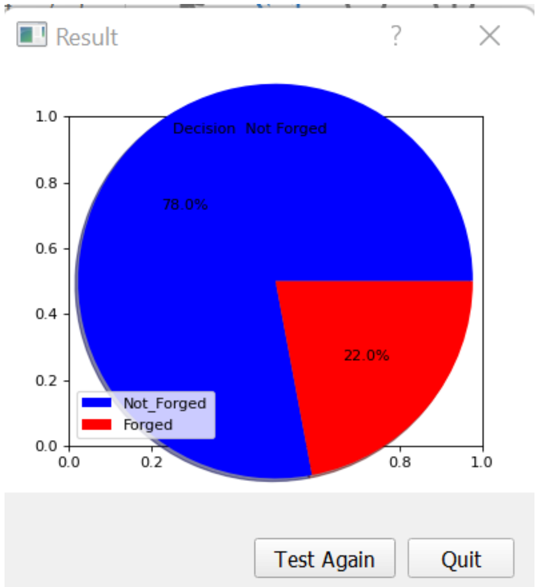


Figure 16 : VGG16 Test Result

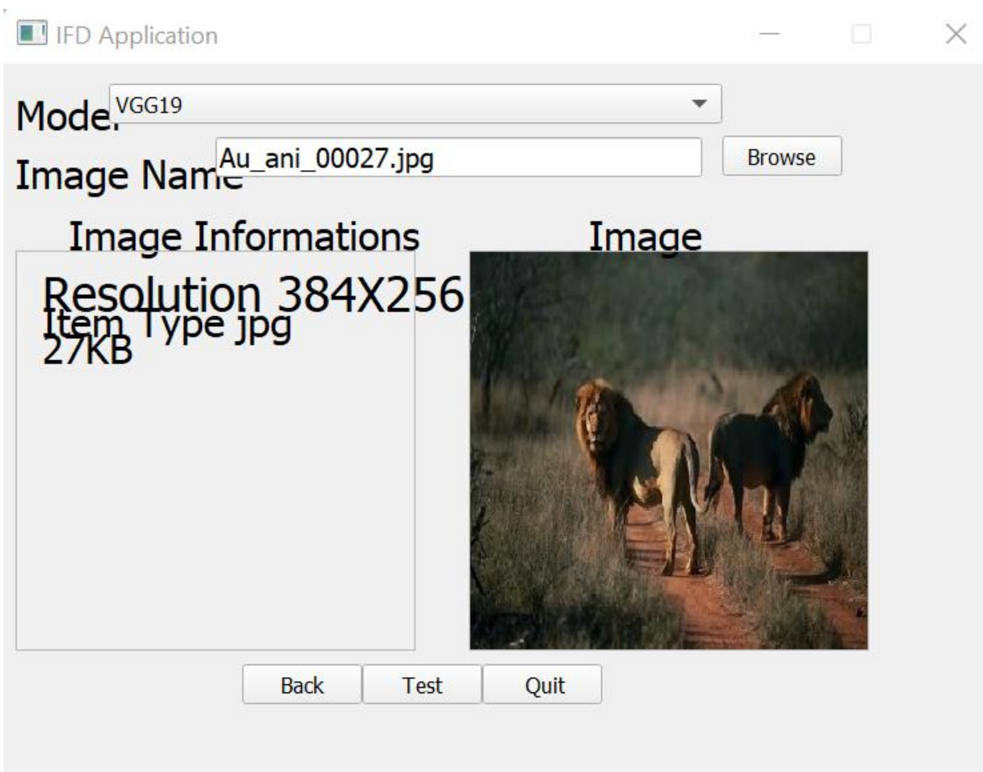


Figure 17 : VGG19 Testing Image

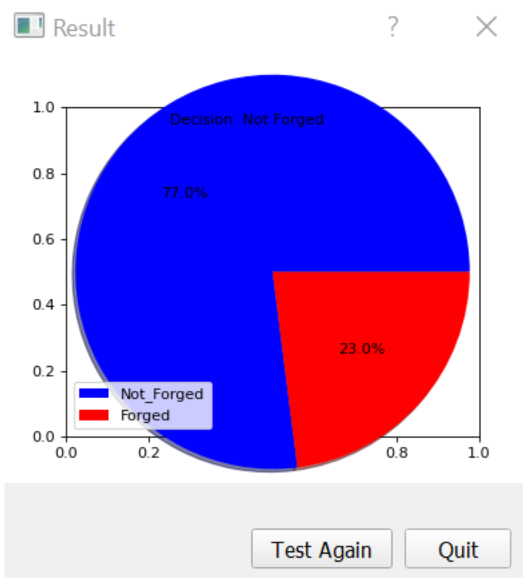


Figure 18 : VGG16 Test Result

Table 8 summarises the overall comparison of different models, as well as their accuracy.

Table 8: Comparison of Different Models

Model Name	**Training Accuracy (%)**	**Validation Accuracy**
CNN	91.87	83.31
ELA + CNN	96.25	84.19
ELA+VGG16	93.21	82.56
ELA + VGG19	95.12	83.27

As seen in the table above, CNN + ELA is the most accurate model for predicting forged photos as fabricated, whereas VGG19 is the most accurate model for legitimate photographs. CNN + ELA ‘s overall accuracy is slightly better, if not identical, to VGG19's.

CHAPTER 6

CONCLUSION AND FUTURE WORK

In this study, we investigated the use of a CNN in the detection of picture counterfeiting. More precisely, we utilized a CNN network for extracting features from CASIA v2.0 and NC16, two datasets of varied complexity. The collected features were also utilized for training and testing purpose of SVM, which achieved accuracy of 96.82 % on CASIA v2.0 and 84.89 % on NC16 with data augmentation, respectively. These findings support our hypothesis that the more difficult the data are, the worse the classification performance becomes. Furthermore, our research found that even when done by specialists, picture manipulation can be identified with an accuracy of more than 84 percent.

The findings of the experiments show that when the samples are more difficult, classification performance suffers. Taking different datasets and on performing data augmentation helped us to do better comparison study between different models. Our study, which is using 3 different neural network models, CNN + ELA, ELA + VGG19 and ELA + VGG16, and we made good comparison by changing different hyperparameters and by manipulating the dataset images, like doing data augmentation helped us achieve good results with respective accuracies of 96.25 percent, 93.21 percent, and 95.12 percent.

Nonetheless, there is undoubtedly much more work to be done in the domain of picture fraud detection, the aim is always to have the neural network which best identifies the forgery based on how difficult the images are. The CNN+ELA training model still have huge area for improvement extending the dataset and employing high computing power devices.

REFERENCES

[1] L. Zheng and Y. Zhang, "A Survey on Image Tampering and Its Detection in Real-world Photos A Survey on Image Tampering and Its Detection in Real-world Photos," no. December, 2018, doi: 10.1016/j.jvcir.2018.12.022.

[2] K. Asghar, Z. Habib, and M. Hussain, "Copy-move and splicing image forgery detection and localization techniques : a review," *Aust. J. Forensic Sci.*, vol. 0618, pp. 1–27, 2017, doi: 10.1080/00450618.2016.1153711.

[3] S. K. Mankar and P. A. A. Gurjar, "Image Forgery Types and Their Detection : A Review," vol. 5, no. 4, pp. 174–178, 2015.

[4] N. K. Gill, "A Review Paper on Digital Image Forgery Detection Techniques," 2017.

[5] R. Dhanya and R. K. Selvi, "A State of the Art Review on Copy Move Forgery Detection Techniques," no. Iccs, pp. 58–65, 2017.

[6] Y. Huang, W. Lu, W. Sun, and D. Long, "Improved DCT-based detection of copy-move forgery in images," *Forensic Sci. Int.*, vol. 206, no. 1–3, pp. 178–184, 2011, doi: 10.1016/j.forsciint.2010.08.001.

[7] A. C. Popescu and H. Farid, "Exposing Digital Forgeries by Detecting Duplicated Image Regions," no. 2000, pp. 1–11.

[8] Y. Hsu and S. Chang, "Camera Response Functions for Image Forensics : An Automatic Algorithm for Splicing Detection," vol. 5, no. 4, pp. 816–825, 2010.

[9] B. Mahdian and S. Saic, "Detection of copy-move forgery using a method based on blur moment invariants," *Forensic Sci. Int.*, vol. 171, no. 2–3, pp. 180–189, 2007, doi: 10.1016/j.forsciint.2006.11.002.

[10] a S. Neighborhood, A. For, D. Duplicated, R. In, I. Forgeries, and B. On, "A SORTED NEIGHBORHOOD APPROACH FOR DETECTING DUPLICATED REGIONS IN Guohui LiI , Qiong WuI , Dan TuI , Shao / ie SunI," pp. 2007–2010, 2007.

[11] S. J. Ryu, M. Kirchner, M. J. Lee, and H. K. Lee, "Rotation invariant localization of duplicated image regions based on zernike moments," *IEEE Trans. Inf. Forensics Secur.*, vol. 8, no. 8, pp. 1355–1370, 2013, doi: 10.1109/TIFS.2013.2272377.

[12] J. Zuo, S. Pan, and B. Liu, "Tampering Detection for Composite Images Based on Re-sampling and JPEG Compression," no. 20095201110002, pp. 169–173, 2011.

[13] M. H. Hussain and M. H. Hussain, "Passive Detection of Copy-Move Image Forgery using Undecimated Wavelets and Zernike Moments Passive Detection of Copy-Move Image Forgery using Undecimated Wavelets and Zernike Moments."

[14] W. Luo and J. Huang, "Robust Detection of Region-Duplication Forgery in Digital Image," pp. 18–21, 2006.

[15] X. Kang and S. Wei, "Identifying Tampered Regions Using Singular Value Decomposition in Digital Image Forensics," pp. 926–930, 2008, doi: 10.1109/CSSE.2008.876.

[16] H. Shabanian and F. Mashhadi, “A new approach for detecting copy-move forgery in digital images,” *2017 IEEE West. New York Image Signal Process. Work. WNYISPW 2017*, pp. 1–6, 2018, doi: 10.1109/WNYIPW.2017.8356252.

[17] D. H. Kim and H. Y. Lee, “Image manipulation detection using convolutional neural network,” *Int. J. Appl. Eng. Res.*, vol. 12, no. 21, pp. 11640–11646, 2017.

[18] J. Chen, X. Kang, Y. Liu, and Z. J. Wang, “Median Filtering Forensics Based on Convolutional Neural Networks,” *IEEE Signal Process. Lett.*, vol. 22, no. 11, pp. 1849–1853, 2015, doi: 10.1109/LSP.2015.2438008.

[19] B. Bayar and M. C. Stamm, “A deep learning approach to universal image manipulation detection using a new convolutional layer,” *IH MMSec 2016 - Proc. 2016 ACM Inf. Hiding Multimed. Secur. Work.*, pp. 5–10, 2016, doi: 10.1145/2909827.2930786.

[20] B. Bayar and M. C. Stamm, “A Generic Approach Towards Image Manipulation Parameter Estimation Using Convolutional Neural Networks,” pp. 147–157, 2017.

[21] Y. Rao and J. Ni, “A Deep Learning Approach to Detection of Splicing and Copy-Move Forgeries in Images,” 2016.

[22] J. Ouyang, Y. Liu, and M. Liao, “Copy-move forgery detection based on deep learning,” *Proc. - 2017 10th Int. Congr. Image Signal Process. Biomed. Eng. Informatics, CISP-BMEI 2017*, vol. 2018-Janua, pp. 1–5, 2018, doi: 10.1109/CISP-BMEI.2017.8301940.

[23] N. Huang, J. He, and N. Zhu, “IEEE International Conference On Big Data Science And Engineering A Novel Method for Detecting Image Forgery Based on Convolutional Neural Network,” pp. 1702–1705, 2018, doi: 10.1109/TrustCom/BigDataSE.2018.00255.

[24] B. Yang, X. Sun, E. Cao, W. Hu, and X. Chen, “Convolutional neural network for smooth filtering detection,” *IET Image Process.*, vol. 12, no. 8, pp. 1432–1438, 2018, doi: 10.1049/iet-ipr.2017.0683.

[25] Y. Liu, Q. Guan, and X. Zhao, “Copy-move Forgery Detection based on Convolutional Kernel Network,” *arXiv*, pp. 18269–18293, 2017.

[26] Z. Shi, X. Shen, H. Kang, and Y. Lv, “Image Manipulation Detection and Localization Based on the Dual-Domain Convolutional Neural Networks,” *IEEE Access*, vol. 6, pp. 76437–76453, 2018, doi: 10.1109/ACCESS.2018.2883588.

[27] Y. Wu, W. Abd-almageed, P. Natarajan, A. Way, and M. Rey, “Image Copy-Move Forgery Detection via an End-to-End Deep Neural Network,” pp. 1907–1915, 2018, doi: 10.1109/WACV.2018.00211.

[28] J. Bunk *et al.*, “Detection and Localization of Image Forgeries using Resampling Features and Deep Learning.”

[29] Y. Rao and J. Ni, “A deep learning approach to detection of splicing and copy-move forgeries in images,” *8th IEEE Int. Work. Inf. Forensics Secur. WIFS 2016*, pp. 1–6, 2017, doi: 10.1109/WIFS.2016.7823911.

[30] J. Fridrich and J. Kodovsky, “Rich models for steganalysis of digital images,” *IEEE Trans. Inf. Forensics Secur.*, vol. 7, no. 3, pp. 868–882, 2012, doi:

10.1109/TIFS.2012.2190402.

[31] H. CPatel and M. M Patel, “An Improvement of Forgery Video Detection Technique using Error Level Analysis,” *Int. J. Comput. Appl.*, vol. 111, no. 15, pp. 26–28, 2015, doi: 10.5120/19615-1508.

[32] P. Ferrara, T. Bianchi, A. De Rosa, and A. Piva, “Image forgery localization via fine-grained analysis of CFA artifacts,” *IEEE Trans. Inf. Forensics Secur.*, vol. 7, no. 5, pp. 1566–1577, 2012, doi: 10.1109/TIFS.2012.2202227.

[33] D. C. J. Corresp and D. C. Jeronymo, “Semi-automatic wavelet soft-thresholding applied to digital image error level analysis Semi-Automatic wavelet soft-thresholding applied to digital image error level analysis,” pp. 0–12.

[34] I. B. K. Sudiatmika, F. Rahman, Trisno, and Suyoto, “Image forgery detection using error level analysis and deep learning,” *Telkomnika (Telecommunication Comput. Electron. Control.*, vol. 17, no. 2, pp. 653–659, 2019, doi: 10.12928/TELKOMNIKA.V17I2.8976.

[35] W. Zhang and C. Zhao, “Exposing Face-Swap Images Based on Deep Learning and ELA Detection,” vol. 5, no. November, p. 29, 2020, doi: 10.3390/ecea-5-06684.

PAPER NAME	AUTHOR
Rashi Gupta image forgery thesis.pdf	**Rashi Gupta**

WORD COUNT	CHARACTER COUNT
9336 Words	**53334 Characters**
PAGE COUNT	FILE SIZE
41 Pages	**1008.4KB**
SUBMISSION DATE	REPORT DATE
May 24, 2022 10:22 PM GMT+5:30	**May 24, 2022 10:23 PM GMT+5:30**

9% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 4% Internet database
- 4% Publications database
- Crossref database
- Crossref Posted Content database
- 6% Submitted Works database

Excluded from Similarity Report

- Bibliographic material