A DEEP LEARNING APPROACH FOR IMAGE SPLICING DETECTION USING ERROR LEVEL ANALYSIS

A Thesis submitted in partial fulfillment of the requirement for the Award of the Degree of

MASTER OF ENGINEERING

in

Electronics and Communication

Submitted by

Tarik

801761017

Under Supervision of

Dr. Kulbir Singh

Professor, ECED

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

ELECTRONICS AND COMMUNICATION ENGINEERING DEPARTMENT

THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY

(A DEEMED TO BE UNIVERSITY), PATIALA, PUNJAB

JULY, 2019

DECLARATION

I, **Tarik** hereby declare that the work presented in this thesis entitled "**A Deep Learning Approach For Image Splicing Detection Using Error Level Analysis**" in partial fulfillment for the award of degree of **Master of Engineering (ECE)** submitted at **Department of Electronics and Communication Engineering,** Thapar Institute of Engineering and Technology (Deemed to be University), Patiala is an authentic record of work carried out under supervision of **Dr. Kulbir Singh,** Professor, Department of Electronics and Communication Engineering, Thapar Institute of Engineering and Technology (Deemed to be University), Patiala from **August, 2017** to **July, 2019**. The matter presented in this has not been submitted either in part or full to any other university or institute for the award of any other degree.

Date: 21/08/2019

Tarik
Tarik

Roll No. 801761017

It is certified that the above statement made by the student is correct to the best of my knowledge and belief.

Date: 21/08/2019

Dr. Kulbir Singh

Professor, ECED

TIET, Patiala

ACKNOWLEDGMENT

Firstly, I want to thank god for providing me good health, well-being, and filling me with patience that was necessary to complete my thesis work.

I would like to express my gratitude to **Dr. Kulbir Singh**, Professor, Department of Electronics and Communication Engineering, Thapar Institute of Engineering and Technology, Patiala for his patient guidance and support throughout my work. I am truly very blessed to have the opportunity to work with him.

I am also thankful to our Head of the Department **Dr. Alpna Aggarwal**, PG Coordinator **Dr. Amit Mishra,** and Program Coordinator **Dr. Anil Arora**, for providing me the adequate environment to carry out my work. I would like to extend my thanks to **Dr. Neeru Jindal**, Assitant Professor, and **Dr. Sunil Kumar**, Assistant Professor, ABV-IITM, Gwalior, for their valuable inputs and concern.

I am profoundly grateful to my parents and my loving brother for their constant inspiration, attention, and care. I am highly indebted to **Mr. Gurinder Singh**, **Mr. Amit Kumar** and **Ms. Kanwarpreet Kaur** for their constant support and assistance in carrying out my work. I would like to acknowledge my friends for their enthusiastic support and time in any way possible to make my work successful.

Finally, I would like to express thanks to all those persons who directly or indirectly helped me and contributed towards this work

Tarik

Tarik

801761017

ABSTRACT

In this digital era, camera-equipped cell phones, scanners, and digital cameras are becoming progressively popular, thus making it simple to obtain digital images. These images are used as a medium for exchanging the information in social media, news, court, communication, etc.

Digital images are often seen as an evidence of reality or a fact. Images can be easily and cheaply manipulated due to the wide availability of photo-editing softwares with the intent and purpose to benefit one party. Therefore, the image that is already manipulated can be used for fake news, evidence, and any publication or to gain popularity to mislead others. As a result, the authenticity of the digital image can no longer be taken for granted. It becomes very challenging for the end users to distinguish whether the image is original or altered. Therefore, image verification has become a significant problem in ensuring the validity of digital images with applications in areas such as medical, government, finance, law enforcement, etc.

Earlier, machine learning algorithms were utilized for classification or regression problems. But, nowadays with the availability of huge data, classical machine learning methods are not working well. Therefore, to handle large and complex data, powerful methods and systems are required. The concept of deep learning is the right solution to this problem. Deep Learning models, with their multi-level structures, are very helpful in extracting complicated information from input images. This motivates to propose the technique that consists of deep learning models to detect image forgeries such as copy-move, splicing forgeries, etc. In copy-move forgery, a portion of an original image is copied and pasted within the same image while in image splicing, the copied portion is pasted on another image. It is difficult to identify the spliced images by the human visual system. Thus, there is a need to develop an efficient technique for the detection of spliced image. The proposed technique is designed for the detection of image splicing forgery by making

use of Convolutional Neural Network (CNN). It is a deep learning neural network which learns patterns from given data. Firstly, preprocessing is performed on images. For preprocessing of images, Error Level Analysis (ELA) technique is utilized. It generates comparable error levels by examining compression artifacts in the given images. After processing with ELA tool, the images are resized and normalized. Then these preprocessed images are fed to CNN for the classification. For extraction of features, CNN uses convolutional layers and for classification, it uses fully connected layers. It extracts features from the preprocessed images automatically and then classifies them between two classes, i.e., authentic and forged images. The experimental results show that the performance of the proposed method, when compared to some existing methods, is better in terms of accuracy, precision, recall, and F1 score.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS AND ACRONYMS

CNN	Convolutional Neural Network
ELA	Error Level Analysis
JPEG	Joint Photographic Expert Group
ANN	Artificial Neural Network
CFA	Color Filter Array
CMOS	Complementary Metal Oxide Semiconductor
CCD	Charge Coupled Device
RGB	Red, Green and Blue
DWT	Discrete Wavelet Transform
SIFT	Scale-Invariant Feature Transform
SURF	Speeded Up Robust Features
LBP	Local Binary Patterns
DQ	Double Quantization
PRNU	Photo Response Non-uniformity
CRF	Conditional Random Field
SAE	Stacked Autoencoder
MLP	Multi-layer Perceptron
SVM	Support Vector Machine
IDCT	Inverse Discrete Cosine Transform
PNG	Portable Network Graphics
FC	Fully Connected
DCT	Discrete Cosine Transform

CHAPTER 1

INTRODUCTION

1.1 Preamble

In today's world, lot of things are happening every minute. Several technological advancements, inventions, and discoveries are observed each moment. Therefore, there is a requirement to get the information out about these improvements over the world. This requirement prompts the establishment of a suitable and secure communication framework. Information, thoughts, and ideas can be shared through this communication system in the form of text, documents, images, and videos, etc. With the development of internet technology and smartphones, a large amount of information is being shared. Images are a skilled and inherent method of communication for individuals, unlike writing, due to their immaturity and the ease with which to understand the information it includes. There has dependably been trusting in the strength of the visual information as a picture. Generally speaking, the image published in a newspaper is considered to be real or genuine. This refers equally to recordings of video surveillance and can be utilized as proof in legitimate problems as probationary content [4].

The widespread of applications or tools that are easy-to-use and low-cost, making easy to alter the graphic material in an image. This prompts the sharing of misdirecting data through images. Therefore, these forgery assaults need to be curbed in order to ensure that correct or real data is communicated. Digital image forensics is the field of study the images that deal with the authentication and validation of the image by collecting image data and evaluating its origin. This situation focuses on the requirement for methods to restore a digital image's past in order to assess its validity and validate its trustworthiness [48].

In order to extract features for image forensics, previously traditional computer vision methodologies were used. These methodologies involve object detection, corner detection, edge detection, etc. The problem in those methodologies is that for a given image, features must be chosen manually. When the number of classes is increased then it is impossible to decide which feature to select. Now a day's deep learning is extensively used for image processing applications because it learns patterns from each class of object. Deep neural networks have recently gained more attention for the image classification tasks [35]. These networks automatically classify the images by extracting the complex features and learning from the images itself to get good performance. Mostly, CNN has performed well while playing with images. CNN has made use of convolution operation for the better extraction of information [65]. In this work, CNN is used for the classification of original and altered images. By the use of Error Level Analysis of the images, CNN is able to perform good results for the given task.

1.2 History of Image Tampering

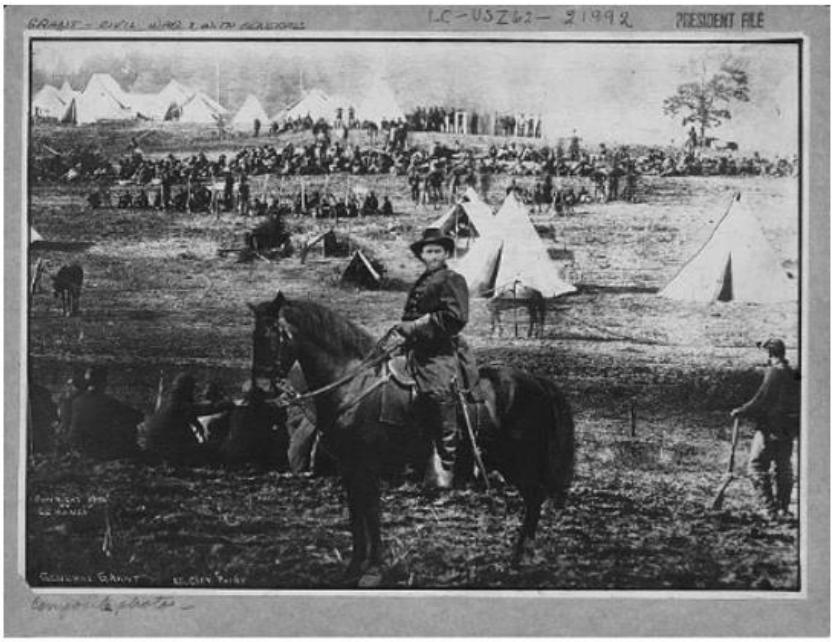
In spite of the fact that image altering has turned out to be increasingly common in the time of advanced cameras and image manipulation applications, it really goes back nearly to the extent the innovation of photography. Several years earlier, photography lost its purity. Just a few centuries after Niepce produced the first picture in 1814, photographs had already been altered. Due to the advancement of image editing tools, powerful computers, and high-resolution digital cameras, it is becoming more prevalent to manipulate images [47].

a)

b)

Figure 1.1 A forged image of airbrushed Stalin's enemies [47]

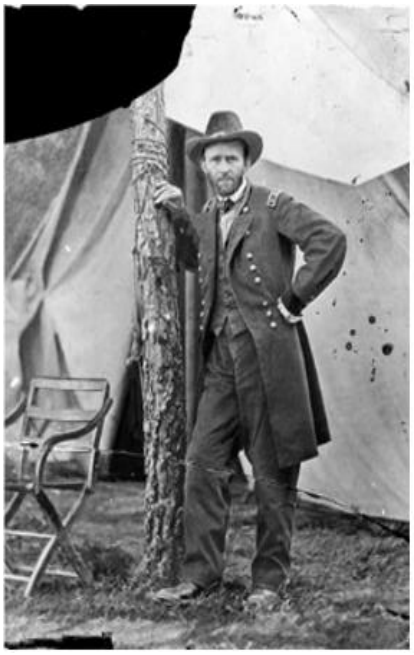
Stalin was known for many reasons during his rule as a pioneer of Soviet Union from the 1920s to his demise in 1953, including the infamous editing of pictures to prevent individuals who had dropped out of his favor. In Figure 1.1, after dropping out of favor with Stalin, Commissioner Nikolai Yezhov in part b) was withdrawn from the initial photograph shown in part a).



a)

b)

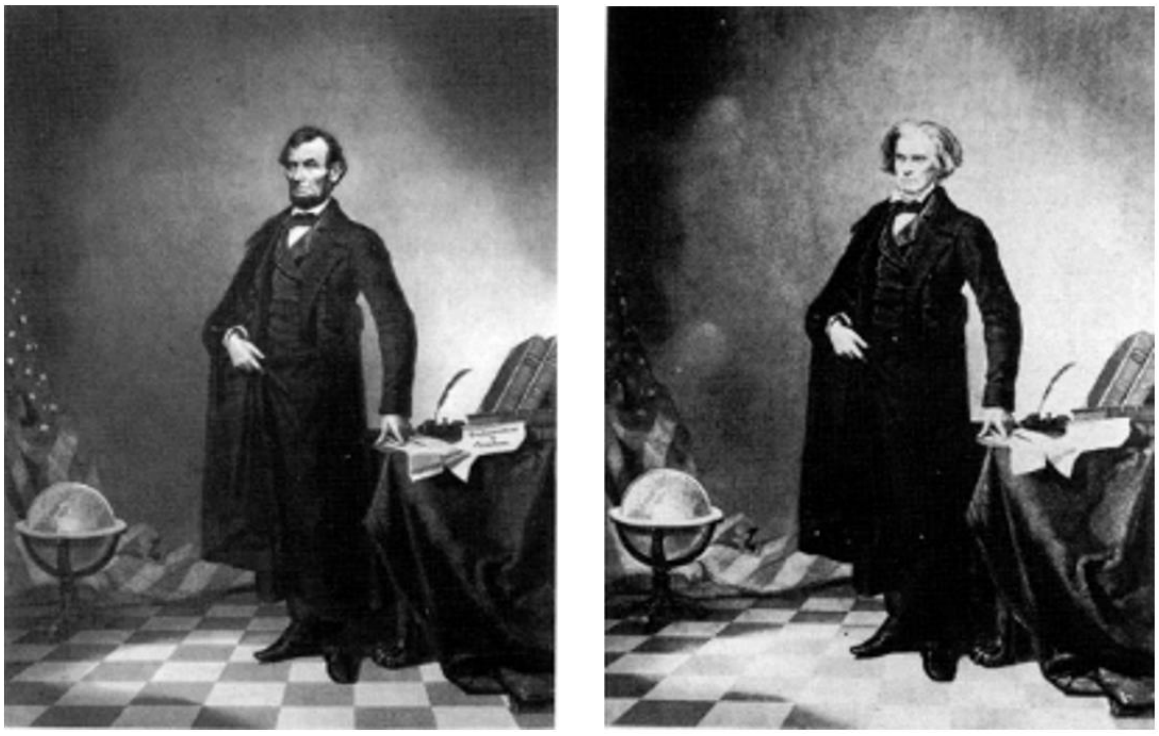
c)



d)

Figure 1.2 A forged image of Ulysses S. Grant during the American Civil War [47]

In 1864, during the American Civil War, the image at the top appears to be of US General S. Grant in front of troops shown in Figure 1.2. However, this print was interrogated by some researchers at Prints and Photographs Division, Library of Congress exposed that the image in a) is made up of three different images: b) the background is of associated inmates taken at the battle of Fisher's Hill, VA from the image at middle left; c) the body of Major General Alexander M. McCook and the horse from the image at bottom left; d) the face of the image is taken from Grant's portrait from the image at right bottom.



a) b)

Figure 1.3 A forged image of U.S. President Abraham Lincoln [47]

One of the oldest instances of manipulating an image is President Lincoln's iconic sketch which has been edited with his head emerging on John Calhoun's body shown in Figure 1.3 where a) is President Lincoln's image and b) is John Calhoun's image.

1.3 Digital Image Life Cycle

The digital image life cycle of an image is depicted as the construction of various phases, grouped into three primary stages: acquisition stage, coding stage, editing stage. In the acquisition stage, the digital image signal formed. When the light comes from the actual scene is collected by the digital camera and is then concentrated on sensors (CCD or CMOS) of the camera. Prior to achieving the sensor, however, the light is generally separated by the CFA (Color Filter Array), a thin film on the sensor that specifically allows a

specific portion of the light to go from it to the sensor. In practice, for each pixel, only one main color, i.e., red, green, or blue is accumulated. For creating a full-color image, the sensor output is then incorporated to obtain all the colors for each pixel. This procedure is called demosaicing. This signal passes through extra processing phases in the camera that is usually used to improve picture perception. These procedures include procedures such as color processing, white balance, gamma correction, image sharpening, improvement of contrast [48], etc.

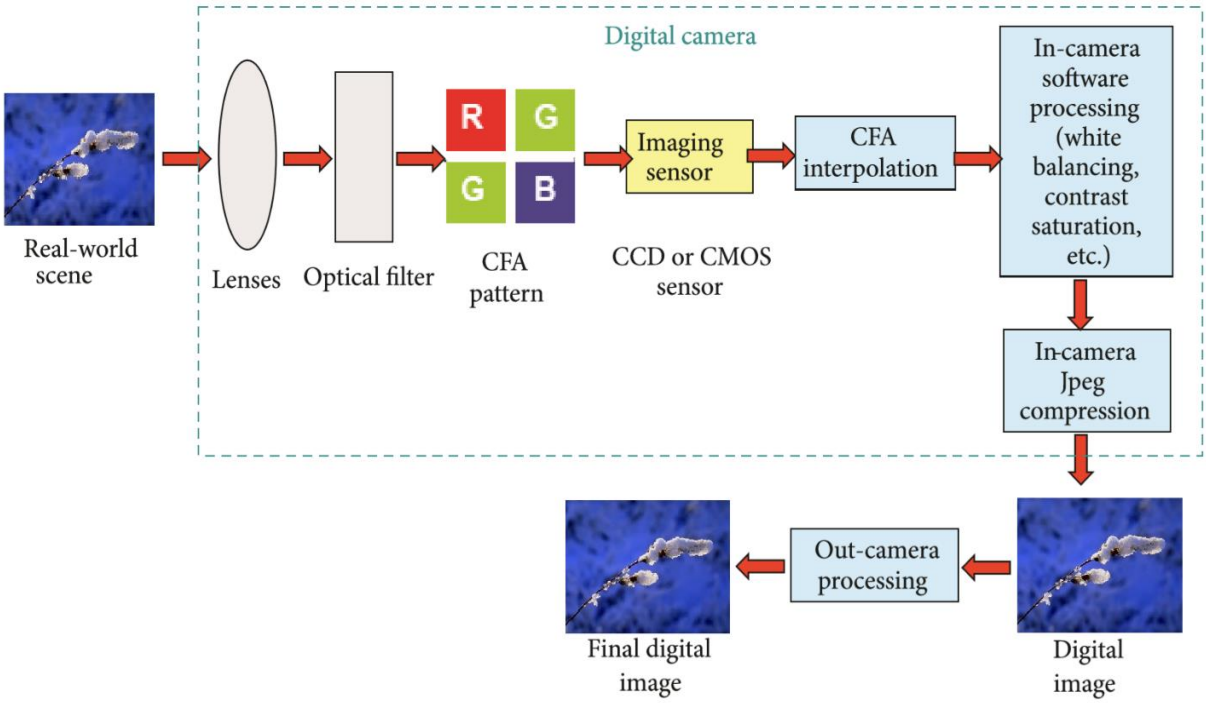


Figure 1.4 A scheme depicted the steps for digital image life cycle [48]

Using the coding method, the operated image is then placed in the camera storage. The lossy compression, JPEG being particular, is used in most globally used cameras because a load of memory is maintained during processing. At last, to enhance or adjust your data, the generated picture can be post-processed. Different procedures of image editing can be encountered by a picture during its life cycle: contrast enhancement, blurring, geometric conversion (scaling, rotation, etc.), sharpening and cloning (or copy-move) and image splicing. The image is finally compressed in JPEG format after processing, so it can be recompressed later [48].

The basic idea of digital image forensics is to analyze the distinctive marks (like digital fingerprints or traces) that are left behind in an image during the procurement phase as well as further successive procedures that occur during the life cycle. Hence, these digital traces can be confined for assessment of the image and catching the historical backdrop of digital information. These traces can be classified into

three groups of fingerprints, namely, fingerprint acquisition, fingerprint coding, and fingerprint editing, as the display of the digital image life cycle. Some common editing operations that are applicable to images are geometric modifications, content modification, and enhancement. These operations include rotation, filtering, zooming, cropping, shearing, cut and paste, seam carving, histogram equalization, copy-move, color modification, copy and paste, contrast adjustment [68], etc.

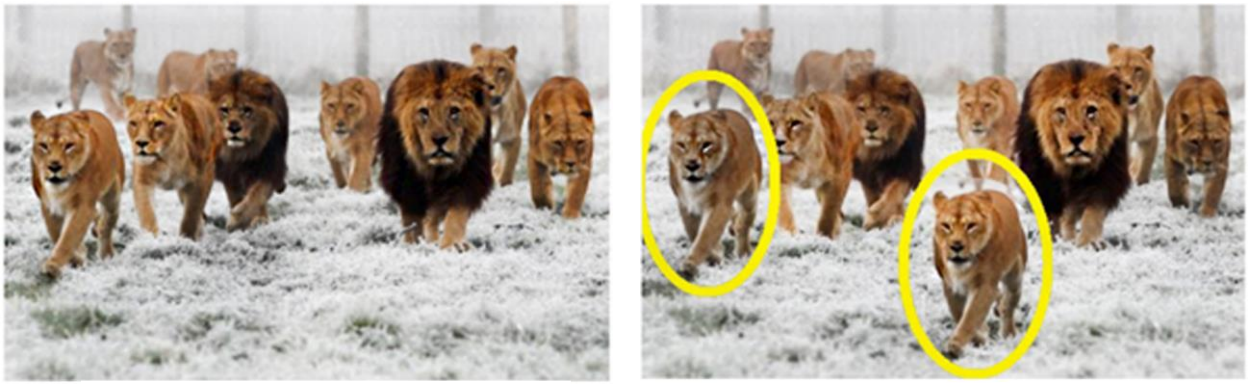
1.4 Digital Image Forensics

The image can be manipulated at any phase in the life cycle of a digital image. With an expansion in the availability of social media in a particular manner, information security has become a major problem. It has turned out to be essential to recognize the creativity of the information that is being shared as it influences many individuals. For images, it is necessary to ensure that the information given by image is unchanged and the modified image can easily be identified. This image processing domain is regarded as digital forensic image processing [4]. Digital image forensics is an area of research that approves and confirms the genuineness of an image by evaluating the digital fingerprints remaining in the picture during the life cycle of the digital image. Reconstruction of its digital image life cycle analyzes the picture being tested. Any anomalies are then inspected for the different traces remaining during the procedures. The image's inconsistencies would eventually show the data manipulation or forgery [48].

1.4.1 Types of Image Forgery

Three types of image forgeries are present that are image copy-move, image retouching, and image splicing. To get these manipulated images different operations are performed on the images. The image processing operations that are involved in creating these forgeries are discussed.

1.4.1.1 Image Copy-Move



a) b)

Figure 1.5 a) Original image b) Forged image – Copy Move Attack [18]

When a portion of the image is copied and placed on the same image then this type of manipulation of image is called as copy-move type forgery. With the use of various image processing applications, the copied region can be manipulated to mix the forged and original contents. Since the pasted part forms the same image, features like noise constituents, dynamic range, color palette, etc. would be in harmonization with some other part of the image. Figure 1.5 shows the example of copy-move type forgery. In part a) the original image is present and in part b) copy-move type forged image is present. Blurring is often implemented along the corners and boundaries to conceal the peculiarities among the actual and copied region. It is also hard for the state of the art technology to recognize the forged portion if the image is enhanced with advanced algorithms [18].

1.4.1.2 Image Retouching

Image retouching sometimes referred to as airbrushing. It is the method of manipulating images to alter a subject's appearance mildly. Retouching often called soft forging, because it applies some image processing tools to improve or degrade specific characteristics of an image. Retouching may be in various ways, such as removing the background, color change, fix imperfection of photos, making the shadow, liquify shapes, etc. This is most frequently used by photo magazine publishers to make the photograph more attractive [62].

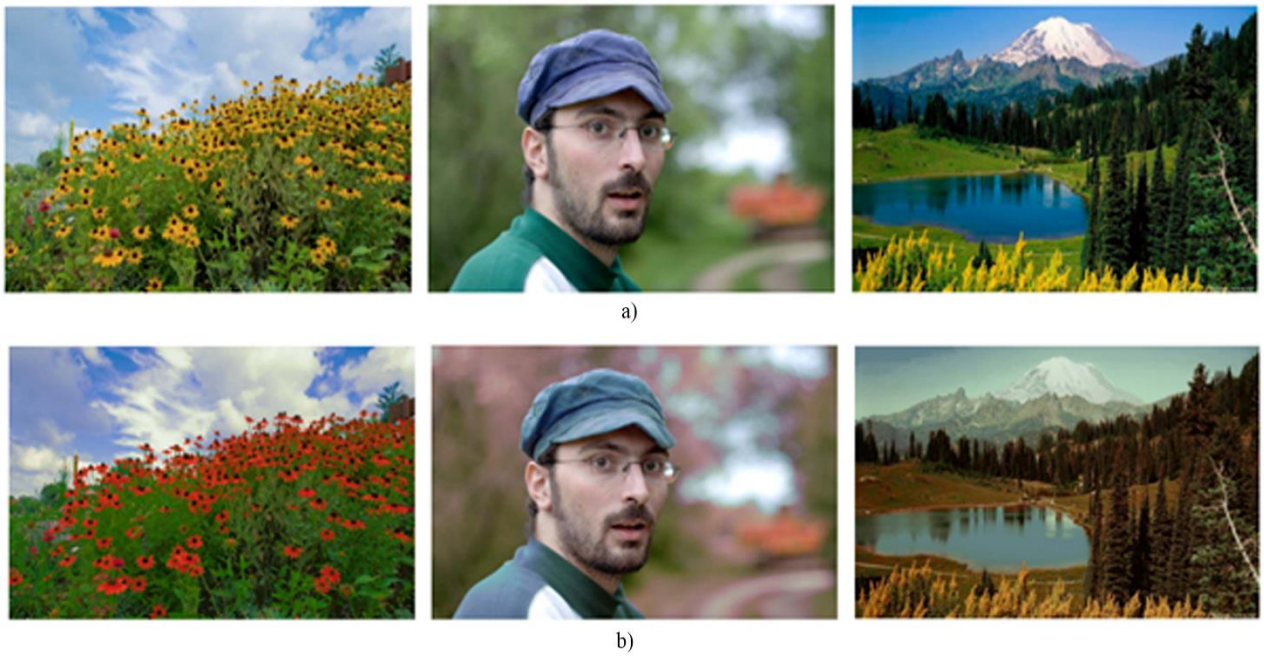


Figure 1.6 Image Retouching - a) Original images b) Recolored images [62]

Figure 1.5 shows image retouching. In a) these three images are original images and in b) these three images are recolored images or an image retouching of original images.

1.4.1.3 Image Splicing

It is almost like copy-move type forgery except that the portion of the image is copied from another image into the actual image in order to make a composite image that looks natural. However, such changes can be identified by researching irregularities in natural statistics of the image. Building an inaccurate image is the union of more than one image. The method is accomplished by removing a part of one picture and placing another picture on the same [39]. Figure 1.7 shows that by copying the portion of both images to create the new image, i.e., spliced image. The picture is taken directly from the camera is the authentic or reference image. It is more difficult to detect image splicing type forgery than copy-move type forgery because the spliced portion of the image is part of another image. Blurring and smoothening on the edges is often implemented after pasting the portion of the image which comes from another image.

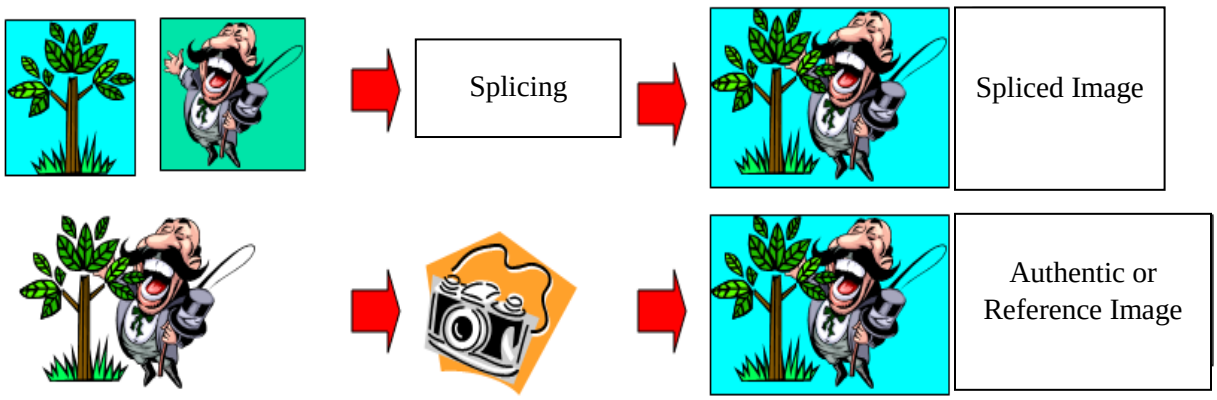


Figure 1.7 Image Splicing [39]

1.4.2 Image Forgery Detection Methods

The methods for determining an image's authenticity can be divided into two main types. In the first one, the data of the reference image is known. It becomes a very simple job to locate the modified region if the altered image is known and the reference image. Another category is where there is only a forged image. In this situation, the image's statistical characteristics are assessed to identify any irregularities that would disclose the forgery in a conclusive manner. This strategy has a wider implementation situation because the reference picture usually has very fewer data. This methodology has a more extensive application in light of the fact that for the most part, there is exceptionally fewer data accessible to the reference image.

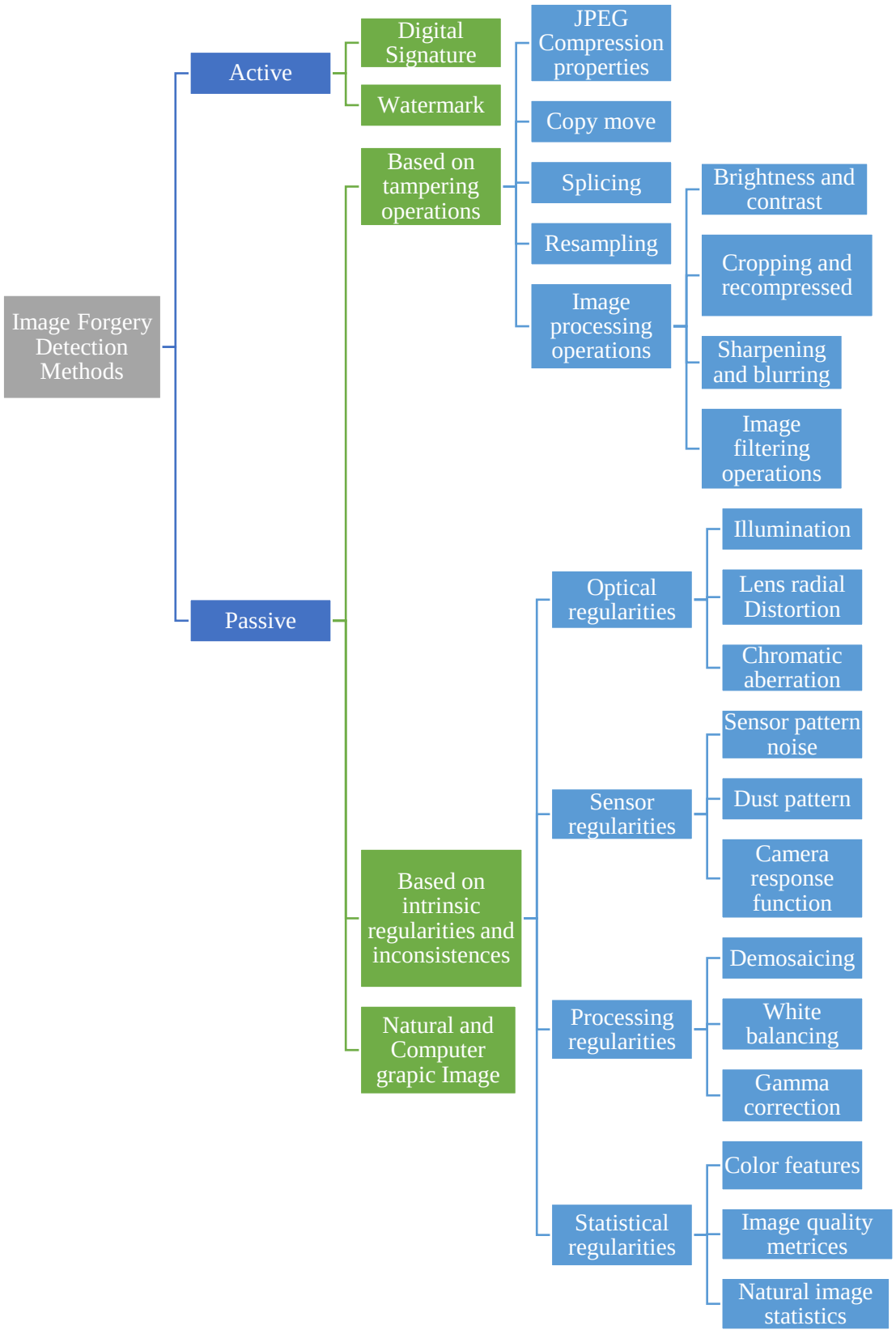


Figure 1.8 Different image forgery methods for the analysis of the image's history and reliability [4]

1.4.2.1 Active Methods

Two primary active protection privacy methods are digital watermarking and signature, as something is incorporated in pictures when pictures are acquired. If embedded information cannot be extracted from that image acquired, researchers can identify the image being tampered with. However, the image can be tampered by altering their properties before the digital signature and watermark creating a stage. To prevent image from tampering, the watermark takes time to install in the image, therefore it is the main problem in active methods. Additionally, the inserted watermark must be robust, i.e., the watermark can withstand attacks and can't be simply changed or removed and this robustness can't be demonstrated to be completely accomplished presently in principle [22].

1.4.2.2 Passive Methods

In passive methods, there is no such type embedding of information is there. Therefore, forgery detection using passive methods is a great challenge. There is no such technique that can handle all types of cases, however numerous strategies each can identify an uncommon forgery in its own particular manner. These methods detect the duplicated region form pixels of the image. In reality, this technique can determine whether the image is modified by any operation through two primary principles, first by attempting to reveal structural distortion (forgery) by learning the irregularities in natural picture statistics. The second set of methods responds to issues like which equipment has been used to record this picture [22]. In Figure 1.8, passive methods based on tampering operations, intrinsic regularities and inconsistencies and other types are shown.

The structure of blind image forgery detection methods involves image preprocessing, feature extraction, classifier selection, feature preprocessing, and classification. Image preprocessing helps to improve the performance by applying some operation. Some image processing operations are transforming RGB to gray, DCT or DWT transformation, cropping, resizing, etc. are utilized to get better feature extraction and to get better results. In feature extraction from each class, a set of features are obtained that helps to differentiate it from other class. The features are extracted by different existing methods like Haar wavelets, Speeded-up robust features (SURF), Color histograms, Histogram of oriented gradients (HOG), Local binary patterns (LBP), etc. Features like global transformation features, statistical features, series expansion features, geometrical features, and topological features are generally extracted by these methods. In classifier selection, a part appropriate classifier based on extracted features for better classification accuracy is selected. At last, the classifier will discriminate between the two sets of images, i.e., authentic images and forged images based on learned features [39]. The classification accuracy will tell how well the method has performed the task.

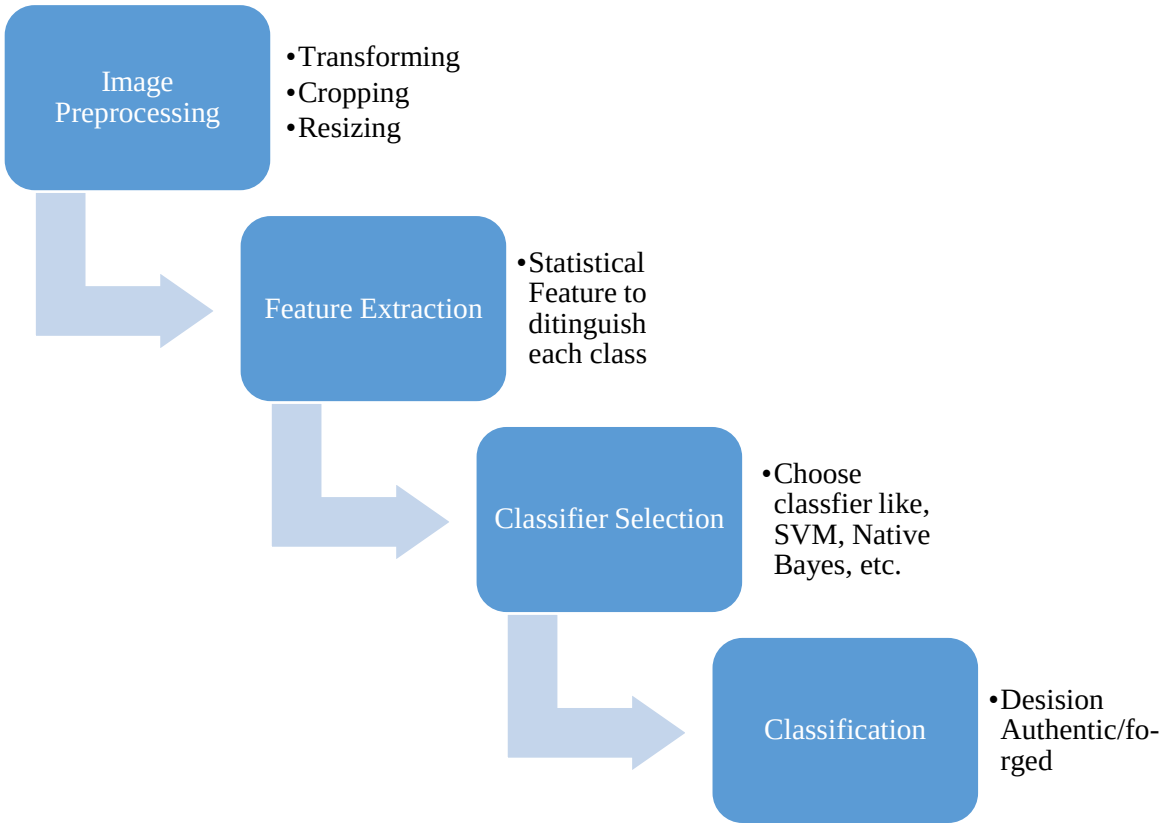


Figure 1.9 Image Forgery detection general framework [39]

Traditional methods used this general framework for forgery detection. In these methods, small and specific features are extracted for the problem analysis. When classes are increased, these methods need to choose significant features for each class. Therefore, feature extraction becomes more ponderous. If more features are selected these must be fine-tuned. Deep learning unveiled the idea of end-to-end learning where the system has just compiled images of which object categories appear. By the use of neural networks, descriptive features are extracted automatically for the given data. It is seen that deep neural networks have performed better than traditional methods when the data is large. CNN, i.e., deep learning model has recently gained the influence in image processing and computer vision applications like image recognition, text analysis, speech analysis, classification, etc. More information about CNN is in the next section.

1.5 Convolutional Neural Networks (CNN)

CNN's automatically learn features and perform the classification. The architecture of CNN consists of non-linear operations of multiple levels. CNN has a few sorts of layers, for example, convolutional layers, pooling layers, flattening layer and fully connected layers. The output function map generally mixes convolutions with various inputs at each convolution layer. Among neighboring components, it can capture

local dependencies [64]. When pixels of an image passes through the network, then network parameters, i.e., weights and bias are updated at each epoch. Each epoch has some iterations. After every epoch, the data shuffles and again passed through the network. Pooling can decrease each feature map's spatial resolution and transforms data into the form that is more global. Reported that, while examining the hypothetical part of feature pooling, max pooling or average pooling can contribute to quicker convergence and enhanced generalization. By means of substituting convolutional layer and pooling layer, the processed data is given to the classification layer in the form of the vector [7]. Finally, the classification layer will provide the likelihood of input images given that are classified into each class through the softmax connection. The calculated probabilities by softmax function will be in the range of zero to one and the sum will be equals to one [37].

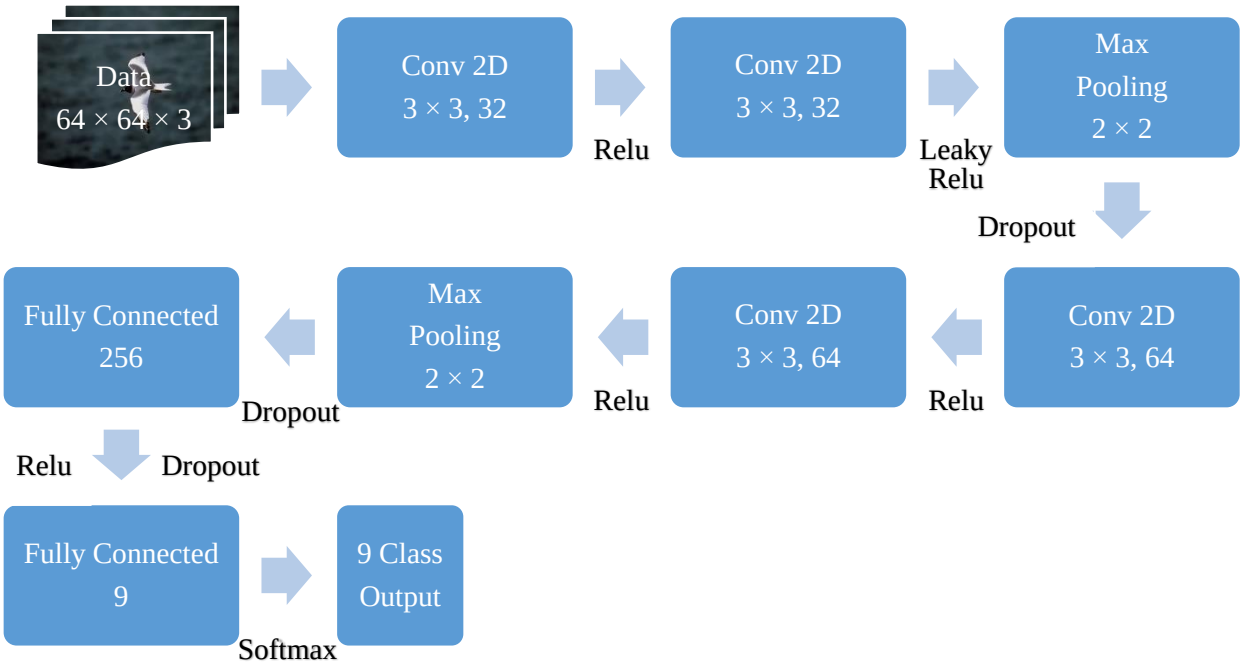


Figure 1.10 Convolutional neural network [1]

The layers of the network are explained:

1.5.1 Convolution layer

Convolution is the primary layer of CNN. In this layer, to extract features from an input image convolution operation is used. It is the key construction block of a CNN that performs most of the complex tasks. Convolution maintains the connection between pixels by studying picture characteristics using small input information boxes. It uses a sliding window to scan the full image with different filters structures. In Figure 1.11 convolution operation on 3 × 4 image by a 2 × 2 filter is shown. Convolution is a mathematical operation that needs input image matrix and a filter or kernel to extract the information from given image. Operations

like edge detection, blur and sharpen the image can be done by using different filters of Convolution layer. The mathematical formula of 2D convolution operation [7] is shown in Equation (1.1):

$$y[m,n] = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} h[i,j]\, x[m-i, n-j] \tag{1.1}$$

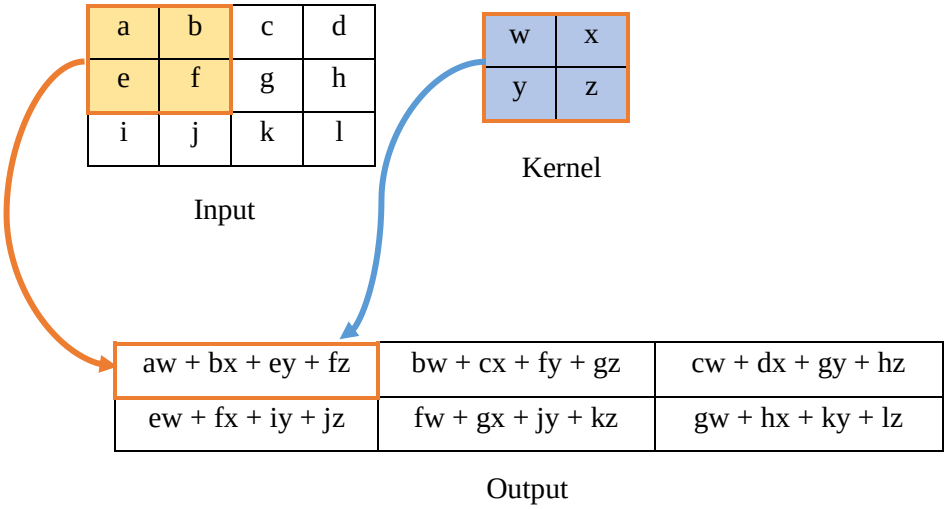


Figure 1.11 Convolution operation on image [35]

1.5.2 ReLU

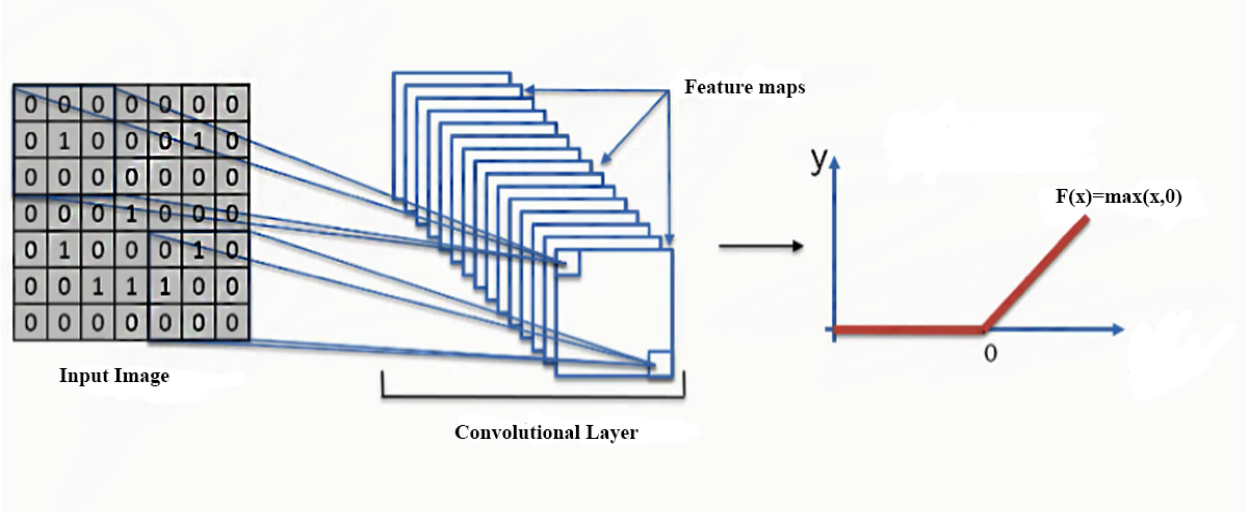


Figure 1.12 A Rectified Linear Unit [17]

Currently, Rectified Linear Unit (ReLU) has become very common. It was discovered to significantly boost the convergence of stochastic gradient descent. It is asserted, because of its linear, non-saturating shape. Compared to tanh / sigmoid activation functions [61] which involves higher computation cost processes

(exponentials, etc.), ReLU activation function can be achieved by merely thresholding the data of activations at zero.

It is computed by the function [17] given by the Equation (1.2):

$$f(z) = \max(0, z) \tag{1.2}$$

1.5.3 Leaky ReLU

When processing with ReLU in CNN, after several processes through various layers the values of the neurons go near to zero this is “Dying ReLU” problem. Leaky ReLUs is one endeavor to fix the "dying ReLU" issue. Leaky ReLU will have a small negative slope when z < 0, rather than the function is zero. The mathematical function that computes leaky ReLU activation function [17] is given in Equation (1.3):

$$f(z) = 1(z < 0)(\alpha z) + 1(z >= 0)(z) \tag{1.3}$$

where α is a small constant. In Figure 1.13, α is 0.01 considered.

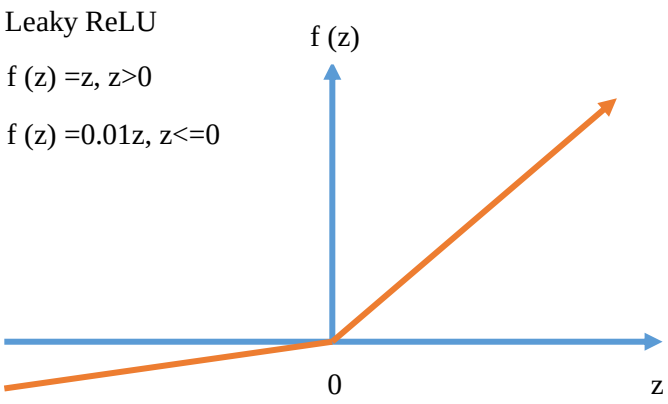


Figure 1.13 Leaky ReLU [17]

1.5.4 Pooling layer

However, this can be computationally difficult and prone to overfitting. Thus, only a certain feature, mean (or max) value over an image region is calculated. The procedure of aggregation is named pooling. Two typical pooling techniques are average-pooling and max-pooling, which propagate the peak and the mean estimation to the next layer respectively within the local region [10]. Spatial information loss is converted into a growing amount of depictions of higher-level features.

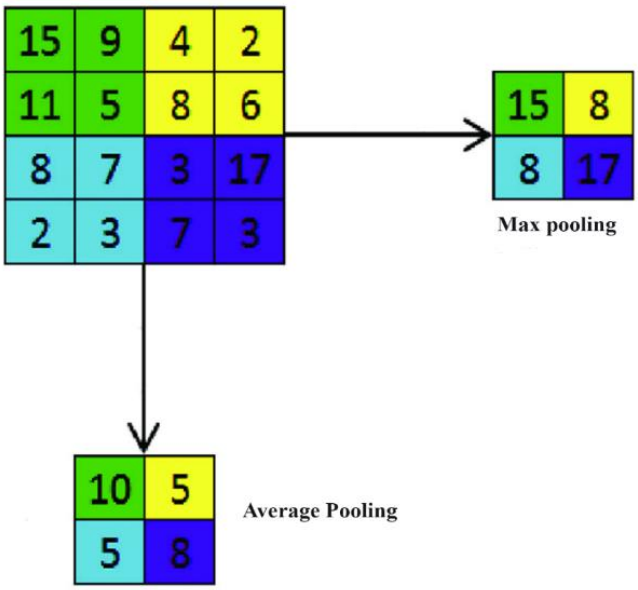


Figure 1.14 Max pooling and Average pooling [10]

After acquiring feature maps using convolution all extracted features can be utilized for classification.

1.5.5 Zero-Padding

To incorporate zeros symmetrically in the input matrix is called as zero paddings. It is a frequently used change that makes it possible to adjust the size of the input to our necessity. It is mostly used to design the CNN layers when the input unit sizes have to be maintained in the amount of output. It is utilized while processing small size images. To get features of small images zeros padded at the border so that after processing image do not shrink and extraction of features can be properly completed [21].

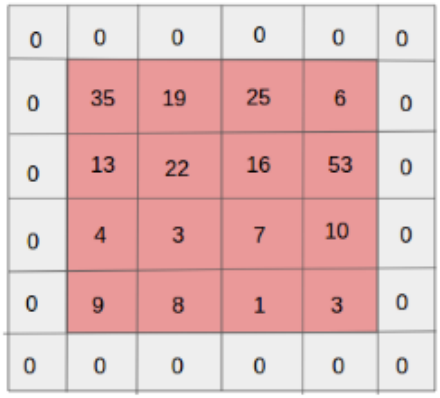
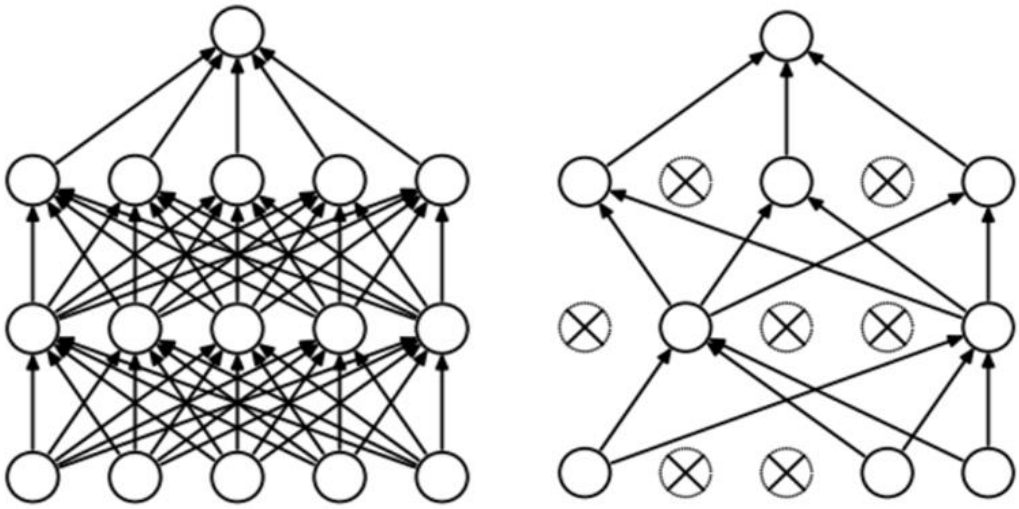


Figure 1.15 Zero-padding [21]

1.5.6 Dropout

To enhance efficiency, the dropout algorithm is implemented by randomly disabling neurons during training in each layer. At the beginning of each training iteration, a dropout map with the identical neuron size in each layer is arbitrarily initialized to check the on or off state of the corresponding neuron. During training, the neurons which are in the off state are separated, by disabling the activation signal forward propagation and backward propagation of error signal. For each learning iteration, it is proportional to switching between various models, so that many different models are trained simultaneously. All neurons are switched on during testing, but the activation signal attenuated during the learning stage to the probability of average turn-on rate [17]. Figure 1.16 shows the example of a simple neural network and neural network after applying dropout.

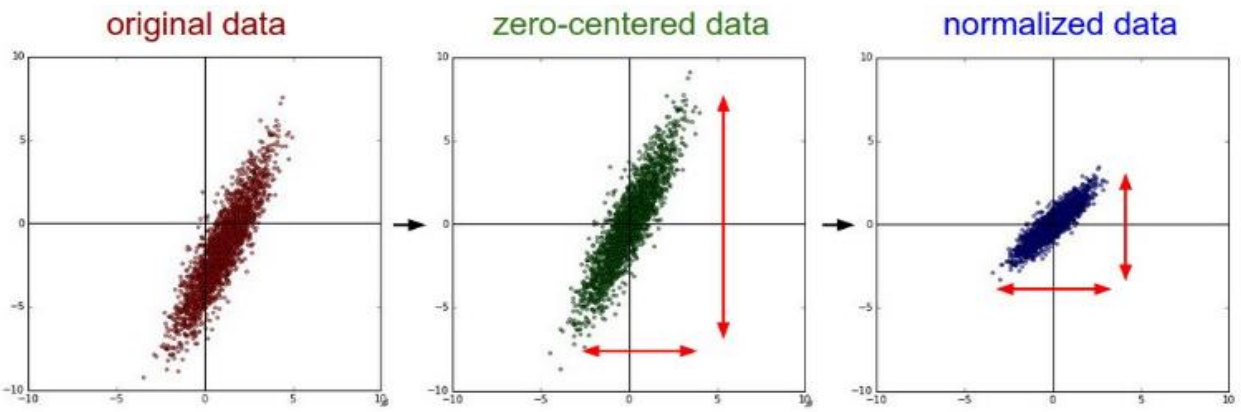


a) Standard neural net b) After applying dropout

Figure 1.16 Dropout applied to the neural network [17]

1.5.7 Batch Normalization

Deep neural networks learning is complicated by the reality that the allocation of the outputs of each layer varies as the parameters of the past layers shift during practice. This slow down the preparation by demanding lesser learning rates and cautious parameter introduction and makes it famously difficult to prepare models with saturating nonlinearities. Higher learning rates can be used while training by the use of batch normalization, and it also enables to be less cautious about initializations [28]. It also eliminates the need for dropout in some cases. For every batch of images, the zero mean and variance equal to one is achieved by batch normalization.



a) b) c)

Figure 1.17 Batch normalization [16]

Figure 1.17 shows how to batch normalization works on data. In a) original data points are shown, whereas in b) is mean-centered data or zero centered data is shown and in c) the normalized data is shown. For batch normalization [28] mathematical Equation (1.4) is used.

$$\hat{x}^{(k)} = \frac{x^{(k)} - E(x^{(k)})}{\sqrt{Var[x^{(k)}]}} \tag{1.4}$$

1.5.8 Flattening

The next stage is to flatten it once the features are acquired. Flattening includes transforming the full matrix of the 2D features map into a single column that is then supplied for application to the neural network. The reason for doing this is that subsequently, to enter this information into an artificial neural network (ANN) [21].

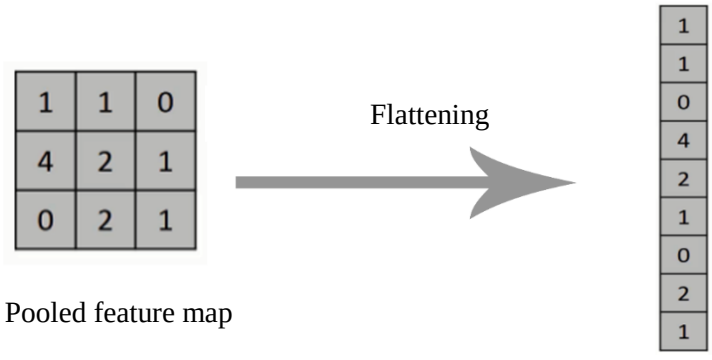


Figure 1.18 Flattening of image [21]

1.5.9 Classification Layer

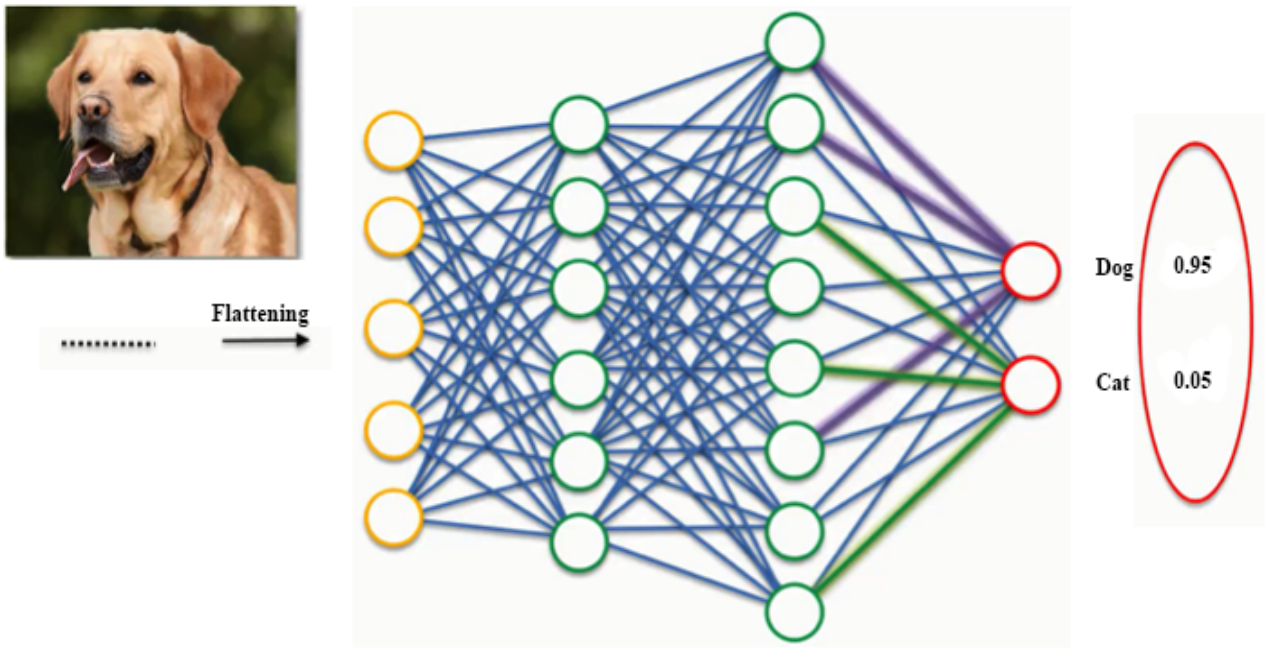


Figure 1.19 Classification probabilities using fully connected layers and softmax function [37]

Typically, few fully connected layers are used to make the classification layer. At the point when the learned features go through the first or two completely associated layers, then these features are given to the end layer of the CNN. A softmax function is utilized for obtaining classification probabilities. These fully connected layers actually form an ANN. The data given to the fully connected layers must be flattened first. Therefore, flattening is used for transforming data into vector form.

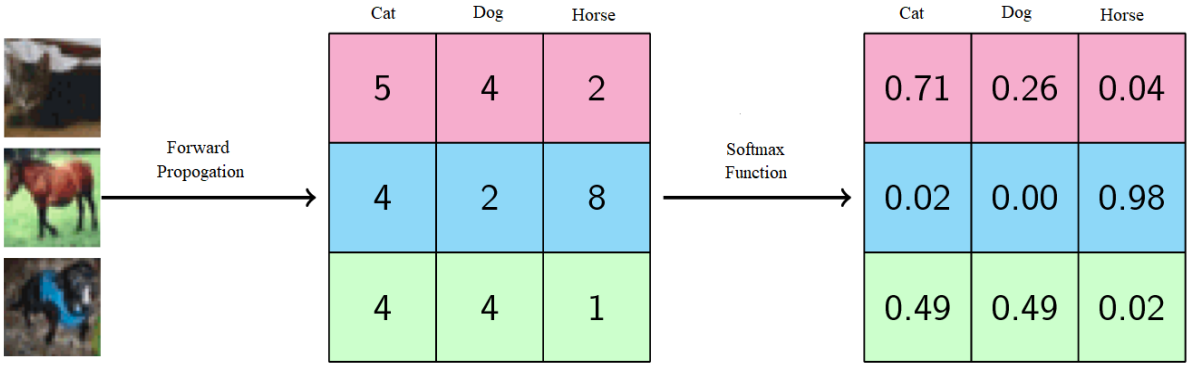


Figure 1.20 Softmax output probabilities [37]

By learning weights and bias from the input data it calculates the score for the classification of the given categories. This score is given to softmax function to get probabilities of each class. This is used in proposed CNN. It is trained by the use of a backpropagation algorithm. As explained above, in the convolutional and fully connected layers the weights and bias were adaptively restored after error propagation method. In this manner, the classification result can be given to control the feature extraction naturally and the learning mechanism can be build up. The softmax function [37] is given by Equation (1.5).

$$f(x_i) = \exp(x_i) / \sum_j \exp(x_i) \tag{1.5}$$

1.6 Organization of Thesis

The organization of the thesis is represented in the following chapters:

Chapter 2 Literature Review: An overview of the existing image forgery detection method is given in this chapter. The review is extended to image analysis using neural networks especially with CNN for the image splicing detection.

Chapter 3 Splicing Detection using CNN: The proposed method based on deep learning model, i.e., CNN that will use Error level analysis as preprocessing for better and automatic feature extraction is explained. The different layers and parameters are given in this chapter.

Chapter 4 Results and Discussion: In chapter 4, a comparison of the proposed technique with the existing techniques is explained. Finally, the performance of the proposed technique is elaborated.

Chapter 5 Conclusion and Future Scope: In this chapter, processing and experiential results of the thesis are concluded and the future scope of the work is also suggested.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The digital image forensics fundamental philosophy is to reconstruct the history of the digital image from its final level to its initial level. In doing so, different procedures are noted that an image has experienced. As mentioned above, the different procedures leave in the image a separate trace that can be obtained and used to evaluate the image's integrity. The thesis report is devoted to the study of error level analysis artifacts that are left in the image after resaving with JPEG compression and processing that error levels using CNN. Different methods related to interpolation are addressed in this chapter. There is also a short summary of other methods used to identify the forgery. However, forensics is never excellent. Inconsistencies may rely on incorrect equipment configurations, differs from expected configurations, etc. Barny *et al.* [2] observed that each forensic method usually deals with the identification of a particular footprint remaining under different configurations by a single image processing tool. Furthermore, manipulation is not the consequence of a single image processing tool, but the mixture of different tools.

2.2 Image forgery detection techniques

Most common techniques of tampering are copy-move, retouching and image splicing that can trigger important modifications in the graphic content of the image. For enhancement purpose typical altering or post-processing operations are brightness change, blurring, color reduction, adding noise, JPEG compression, etc. Classification can be performed by identifying the traces left by image manipulation to determine whether or not the image is being manipulated. Therefore, the task of detecting tampering is a binary, i.e., two class classification problem [68]. Tampering clues like camera-trace inconsistency, geometric inconsistency, lighting inconsistency, or edge discontinuity were explored to detect image splicing and retouching.

2.2.1 Copy-move detection methods

The major task in copy-move detection is to evaluate the presence of the copied area within a given image. To extract some key points form the image, firstly the square blocks are taken from the image, i.e., small units in order to look for similar regions. From each block, features are developed for pattern matching. By comparing the features of each block by using some algorithm, closely matched blocks can be found. If there is duplication in the image then there must be matched features pairs among two dissimilar portions

within the image. Christlein *et al.* [13] categorized copy-move detection methodologies into two different classes, i.e., techniques utilizing key-point based features and block-based features.

2.2.1.1 Key-point based approach

In Key-point based approach, the extracted and local features points are used for identification of the copied region. Key-point based algorithms are efficient and robust. The two common key-point based methods for image feature extraction are the scale-invariant feature transform (SIFT) and the speed-up robust features (SURF) [68].

In reality, it has been demonstrated that the features extracted by SIFT and its variations are efficient for face recognition, image matching, object recognition. In particular, the features extracted by SIFT is invariant to illumination changes, orientation, partly invariant to affinate distortion and uniform scaling, so it is not awful to be appropriate for identification of copy-move type forgery. Huang *et al.* [27] evaluated the extraction of descriptors using the SIFT algorithm. The SIFT descriptors are invariant to modifications in scaling, rotation, illumination, etc. These SIFT descriptors are utilized finding the similarity between the copied region and pasted region.

Pan and Lyu [45] additional formed procedures for localization of duplicated regions according to matched SIFT key-points. Researchers dedicated their attempts to enhancing localization methodologies depending on the SIFT features. For example, it has been proven that the J-Linkage method described by Toldo and Fusiello [55] is more efficient than RANSAC in finding many illustrations of seeing the identical setup in the image. Latterly, in order to improve the location of manipulated areas, Zhou *et al.* [69] suggested to use not just the SIFT features and also their attributes, along with the dominant orientation, coordinate and distinctive scale, to filter fake SIFT and guarantee right matches.

SURF is considered to be an accelerated variant of SIFT [31]. It was used for copy-move identification by Xu *et al.* [5] and was discovered to be reliable for modification functions like rotation, scaling or post-processing functions like blurring, adding noise and JPEG compression, etc. In flat areas as there will be no key-point in those areas, it has been observed that key-point-based methods might be ineffective to copy-move altering. To tackle this problem, Zandi *et al.* [63] enhanced the standard key-point allocation system which mostly removes key-points from the image's non-smooth areas. Their suggested technique is capable of detecting key-points throughout the entire image, even in areas of poor contrast.

2.2.1.2 Block-based approach

For copy-move identification, intuitive concept of separating an image into several blocks of the same size was first suggested by A.J. Fridrich *et al.* [23]. Sliding block window vertically and horizontally through

an image was regarded as overlapping between blocks. If there are copied portions in an image, copied blocks should also exist. Researchers have created an attempt to find an effective manner to match these blocks. The first concept is to match the pixel values of the blocks [23]. However, if the image had been post-processed by actions like adding noise, blurring, and JPEG compression, no precisely linked blocks can be discovered since owing to post-processing the duplicated areas would be changed differently [68].

Compressed features like statistical measurements that may be more useful in comparing blocks with tiny differences can be used rather than using pixel values in blocks. Lin *et al.* [38] suggested a technique for measuring the green channel of a color image, i.e., grayscale image. This presented technique splits it into four smaller-blocks for a specified block and computed the average pixel scores for the four. For determining and composing a 9-dimensional feature vector, the five mean values are utilized to signify an image block with the mean pixel value of the entire block in this technique. Another benefit of the statistical features relative to the naive pixel depiction is that their smaller sizes lead to quick search velocity for comparing blocks. The transformation of values of the pixel in the frequency domain may result in a more stable depiction of the function [23].

For evaluating image pixel intensity, image moment features are helpful. Since these image moment features have shown to be efficient in other image matching assignments, evaluating those image features in the copy-move forgery detection assignment is sensible. Wang *et al.* [57] analyzed an image that constitutes the Gaussian pyramid decomposition and removed Hu moments for the corresponding objective from each block. Finally, it was verified that the three-moment invariants are resistant to incorporating blurring, noise, and JPEG compression.

2.2.2 Image splicing detection methods

It is also known as cut-paste type forgery. The altering hints remaining by image splicing are usually much less visually noticeable than the powerful sign of copy-move region duplication. It may, therefore, be harder to detect image splicing instead of identifying moving copies. Another result is that the identification and localization of image splicing are two distinct functions owing to the lack of duplication portion. However, for splicing detection, earlier works did not aim to locate the spliced portion and numerous methods were built for the sole purpose of identification. In contrast to copy-move type forgery, definitely produces the indication of duplication of portion inside the manipulated image, image splicing does not essentially generate specific indication. Tampering hints are assumed to play a significant part in the image splicing detection [68].

Common tampering hints for image splicing include inconsistency in camera trace, edge discontinuity, inconsistency in geometry, and inconsistency in lighting. Similar to all image processing operations, to

uncover the presumed tampering hints, a decent representation of the image feature is required. Machine learning classifiers are thought to distinguish features between original images and manipulated images [68]. Concisely, techniques of detection-only treat a complete image from which an illustration of a function is obtained, but techniques of localization perform a fine-grained test on local units such as pixels or blocks.

2.2.2.1 Edge anomaly based methods

It is common to believe that fusing an item in an image will be left unusual artifacts on the corners of manipulated areas. When image splicing is achieved without blurring of the edge, manipulated images should have more edge discontinuity than original images since every camera has a smoothing operation to prevent aliasing impact. If blurring is performed after image splicing, the blurred edges should be distinct from the ordinary edges produced during the camera processing phase. The sharp edge is a noticeable proof of manipulation without any blurring [9].

For detection purpose, Ng *et al.* [44] further suggested removing the bicoherence features of the original part of the image and leaving only the splicing-induced features called residual estimation. The method found that the remaining estimate can show the discontinuity of the edge faster than the simple image. In addition, the edge proportion is calculated in an image as an extra feature is utilized to enhance the accuracy of the final detection in canny edge detection algorithm [8]. Subsequently, it is assumed that sharp corners in manipulated images will result in high phase congruence, Chen *et al.* [12] explored impact of characteristic functions and 2-d phase congruency for image splicing identification. Alternatively, the Sobel operator and the image run-length histogram were used by Dong *et al.* [19] to build edge-representing characteristics. Markov process and obtaining features with higher and higher dimensions are utilized to achieve better performance in image splicing detection [67]. For classification, discriminative features are cross-domain features obtained from block discrete meyer wavelet domain and discrete cosine transformation domain. These extracted features used by SVM classifier for the classification.

Other function depictions studied for edge discontinuity detection include local, steerable pyramid transform [43], LBP [66], and co-occurrence matrices [46]. From this point of perspective, not only these characteristics are susceptible to modifications induced by manipulation, but these characteristics are also helpful for identifying other image alterations. This also creates another issue, however, it is hard to decide that an image is made with manipulation. To tackle this issue, the manipulated area must be located to guarantee that tampering occurs.

For localization purpose, to locate edges produced by manipulating, it involves first detecting edge portions in an image and then distinguishing edges in actual areas from ordinary edges. Hsiao and Pei [25] presumed that attackers would normally use manual blurring on the edges of manipulated areas. Due to different

distribution coefficients normal image block and blurred image block, the blurred edges could not be exposed by verifying the coefficients of DCT of that image block. In both the prediction map and the plain image, Wang *et al.* [59] utilized additional features, such as non-sampled contourlet transformation attribute and the phase congruency [12] to better identify the blurred edges.

2.2.2.2 Region anomaly based methods

The most commonly used format is JPEG compression. The resulting picture will definitely have double quantization (DQ) impact owing to the two JPEG compression procedures when the intruder tampers the image which is of JPEG format and retains the modified form in JPEG as well. In order to recognize double compressed JPEG pictures, Popescu and Farid [50] created statistical methods. To enhance the efficiency of localization using the coefficients of DCT, Huang and Korus [34] have studied the local dependencies among findings of localization at distinct scales and evaluating three techniques of fusing multi-scale findings. Additionally, Wang *et al.* [59] presumed that because of the DQ phenomenon, the original areas contained fewer frequency data than the tampered areas. Therefore suggested building the noise map of JPEG compressed image, i.e., by subtracting the remaining map a specified image from its JPEG compressed image and revealing the components of high frequency in the feature map, which is obviously a manipulative mask.

CNN has recently been explored for the detection in image splicing of double compressed areas. Due to the observed effectiveness of DCT coefficients for double JPEG compression assessment. Zhang and Wang [58] introduced the DCT histogram features straight to 1D convolution models. Barni *et al.* [3] were encouraged to use automatic DCT histogram computing with 2D convolution layers. Amerini *et al.* [1] merged the two CNN models, i.e., the 1D CNN model for evaluating the extracted features of DCT, and a 2D CNN model for utilizing artifacts of JPEG in Red, Green, and Blue (RGB) channels. The mixture achieved a substantial increase in the location of the manipulated areas.

The inconsistency in original and manipulated areas of JPEG compression is one of the significant indications to hunt forensic picture experts in the previous century. Unlike edge anomalies that are restricted to strong corners or digitally blurred edges, this consistency is distributed throughout the whole manipulated area, making it comparatively helpful to locate the spliced portion in an image. In certain situations, like when the manipulated images are gone through image processing activities such as sampling or if modified images are not JPEG images these techniques will not work. Therefore, the main shortcoming of JPEG-based techniques is that these techniques will fail to insert between the two compressions.

With lightning inconsistencies, in other areas of the image, illumination within the manipulated area may not be consistent. Johnson and Farid [30] suggested estimating distinct object's illuminating patterns in an

image. The dispute between the projected directions shows that tampering exists. Farid and Kee [32] outlined in what way to predict the path of rays on face appearances in order to prevent altering in two or more face pictures. The concept has been demonstrated in a few instances but on any benchmarking datasets, the techniques have not been performed or assessed.

For image splicing identification, illuminant color characteristics were also explored in addition to the illuminant path. Wu and Fang [60] suggested measuring the error angle around distinct picture block features and discovered that the angle among two original rows is usually lower than the angle from a genuine block and a manipulated block. Their technique needs a manually chosen reference block to detect manipulated blocks rather than using a square block. Latterly, Pomari *et al.* [49] obtained illumination maps and implemented illumination transfer learning features representation for classification and subsequently manipulated localized region.

With inconsistencies in camera traces, different camera manufacturers often have distinct processing method schemes [68], leaving in the resulting pictures distinct trace marks. As cut-paste splices on two or more pictures, it is probable that distinct cameras will capture these pictures. Therefore, it is useful to examine the inconsistencies of camera traces to locate tampering. Three commonly used camera traces or fingerprints for image splicing identification are the picture reaction non-uniformity (PRNU), camera response function (CRF) and color filter array (CFA).

PRNU affects the processing method, i.e., distinct pixel values can be recorded by pixels which have same light intensity. And the PRNU noise models should be considered distinct with each other in two pictures made by two distinct cameras. Chen *et al.* [11] demonstrated that PRNU's inconsistency is helpful in locating manipulated areas. A PRNU correlation graph displaying the unusual blocks can be obtained by computing the correlations among the PRNU of each picture unit and a reference PRNU. In order to improve localization efficiency, Huang and Korus [33] performed multi scale analysis. The major disadvantage of PRNU-based approaches is that in order to assess the PRNU reference it require previous information of the camera or its pictures generated.

The CFA module includes the color values on the camera sensor and produces a full color [51] picture. The different CFA interpolation algorithms, i.e., error-making techniques, immediately influence the image's color correlations. The expectation-maximization algorithm was used by Popescu and Farid [51] to assess the interpolation kernel parameters. In genuine areas, the illustration of assessment in manipulated areas is visually distinct from that.

CRF is recognized as mapping cap distortions tested by a sensor in the camera to color values from pixel illumination values [68]. Lin *et al.* [40] chosen some patches manually along the noticeable edges and

measured the CRF in the individual image. If the key attributes of the predicted CRFs are uneven next to the edges, it is considered to be a spliced image. Hsu and Chang [26] used an image segmentation technique to identify object borders automatically and then created CRF estimates to investigate the inconsistency between the two boundary regions.

Another exciting job is to use CNN to identify camera traces. Bondi *et al.* [6] discovered that the fundamental distinction between two unidentified camera designs could also be identified by pre-trained CNN to recognize camera models. The pre-trained CNN is used to locate the tampered region by utilizing the output of CNN, i.e., the camera identity vector, as the image block feature representation. If a picture includes blocks obtained by different cameras, it is possible to distinguish the manipulated area by grouping the feature vector of blocks into two classes.

2.2.3 Feature learning methods

To identify particular manipulation, several genuine or real images can be supplied into large-capacity machine learning modules, e.g., deep learning models [35]. These modules are allowed to discover the inherent distinction among two categories automatically. Researchers in machine learning have created considerable attempts to design and manage these models among overfitting and underfitting. In the case of image processing, deep learning has recently gained more importance due to its automatic extraction and processing of features directly from the given images. Usually, in machine learning models the data gathered is split into a training dataset and a test dataset. The fundamental distribution of distinct categories should be captured by a nice model in the training dataset. A multi-layer (deep) system with a big amount of parameters can often achieve this. Models that fit the training set on the test set, indeed, do not always work well as the fundamental distributions provided by the training and test datasets might be incompatible.

The simplest but best efficient procedure is to obtain more training data, but this sometimes requires too much manual work. Another alternative is to create regularization methods to decrease the big distance among training datasets and test datasets in overfitting [35]. Research of developing and mastering deep learning models has progressed quickly over the previous century, and many models have been demonstrated to be overall solutions for learning and classification features. Some feature learning techniques are already evaluated to reveal a particular tampering hint in the past segments, like CNN for learning and identifying JPEG double compression, edge blurring, region duplication and camera traces [68]. These neural networks work well by learning from data given to them. In the following, some latest works that learn general or various tampering features are presented.

Zhang *et al.* [65] obtained fundamental features of DWT extracted from the chroma channel as it is recognized that this color space is more susceptible to artifacts being manipulated. An unsupervised deep

learning model, i.e., Stacked Autoencoder (SAE) is utilized to know higher-order features from the fundamental features of DWT. To fine-tune the parameters that the SAE has earlier calculated, another deep learning model multilayer perceptron (MLP) is used having identical layers as with the SAE. The learning and feature extraction processes of the function are conducted on blocks of an image so that each block can be determined as genuine or manipulated by the resulting model. The author regarded image segmentation to improve efficiency to locate the tampered part because the areas manipulated are generally semantically significant areas that correspond to distinct items or background images. Such that in [36], an image is separated automatically into semantically autonomous areas, and the location results are the areas that contain many manipulated images. Cozzolino and Verdoliva [14] also used SAE but used image residual features instead of DWT features to improve the performance.

Rao and Ni [52] suggested a particular structure for manipulating detection only using same size image blocks trained on CNN. The received manipulated blocks from actual images along the borders of manipulated areas randomly sampled genuine blocks. In addition, block transposition and rotation were preferred in order to produce more training data. The choice on the image level is achieved by combining the features on CNN's last layer and trained SVM classifier for the estimation of results. Most significantly, studies showed that it was useful for CNN to know the fundamental distinction between genuine and manipulated to analyze image residuals instead of the initial RGB inputs.

Cozzolino *et al.* [15] also demonstrated that the residual features of the hand-crafted image can be acquired by CNN. To accomplish the limitations, the constrained architecture is intended with a CNN and the parameters are finely tuned in CNN to gain more enhancement. The training data consists of with or without alterations like median blurring, resampling, adding noise, JPEG compression, and filtering on image blocks. Compared to other techniques, the suggested CNN demonstrated competitive efficiency in identifying such tampering maps. Liu *et al.* [41] expanded this concept by using various CNN's to evaluate a multi-scale image and fuse multi scale outcomes to strengthen localization, comparable to the concept of multi scale evaluation in [33] but removing hand-crafted features of DCT with CNN models.

In [52, 15], effectively implemented CNN to analyze the residual picture to detect image manipulation, but the distinction between the two concepts can also be noticed. The objective of Rao and Ni [52] is to detect edge anomalies and Cozzolino *et al.* [15] are more concerned with regional anomalies. Salloum *et al.* [53] mixed the two thoughts from both unusual edges and areas as a multi task strategy for features learning. The edge information is shown to be useful in improving the efficiency of tampering localization.

2.3 Gaps in Study

The recent techniques of detecting image forgery work well, on some databases but not generally. The following difficulties are experienced in present techniques for identifying image forgery.

- Robustness: To describe the results, the current procedures requires a human explanation. These techniques are not fully done by machines. There is still no unified methodology which is capable of detecting any type of forgery. Methods are not robust enough to identify all types of forgery using the same algorithm or technique.
- Performance evaluation: In terms of precision and false alarm levels, most of the suggested forgery detection methods do not provide feasible efficiency. Separate metrics need to be defined for algorithm assessment depending on chromatic aberration, lighting inconsistencies in the color filter array, and inter-pixel correlation. Due to prevalent types of post-processing, such as resizing recompression, edge blurring and adding noise, the image experiences a universal distortion. These fundamental hypotheses no longer work and there is a dramatic drop in efficiency.
- The primary task is to show the structural modifications that occur owing to forgery in the image.
- Methods often display restricted efficiency in complicated datasets of highly diverse origin images. Discriminative handcrafted features are tedious to develop, and the constructed features are not essentially the best fit for a particular forensic issue, particularly in complicated and difficult databases.

2.4 Thesis Objectives

The objectives of this research work are to detect the forged or manipulated part of an image. With the use of the proposed method above challenges will be attempted. The main objectives of this work are:

1. To design a forensic technique to detect image splicing by analyzing error levels using CNN.
2. To perform the comparative study of the proposed scheme with current methods.

2.5 Database used

For forensic analysis and expansion, investigation of image data is essential. Institute of Automation under the Chinese Academy of Sciences actively researches on image forensics. To examine various algorithms CASIA has created image splicing database. This database is commonly used by the researchers for image splicing detection. In this work, CASIA image tampering detection evaluation database version 2.0 is used for our research purpose [20]. The dataset contains 12,560 color images with different image sizes from

240 × 160 to 900 × 600 pixels. There are 7437 authentic and 5123 forged images in this dataset. The database consists of BMP, JPEG, TIFF, format images.

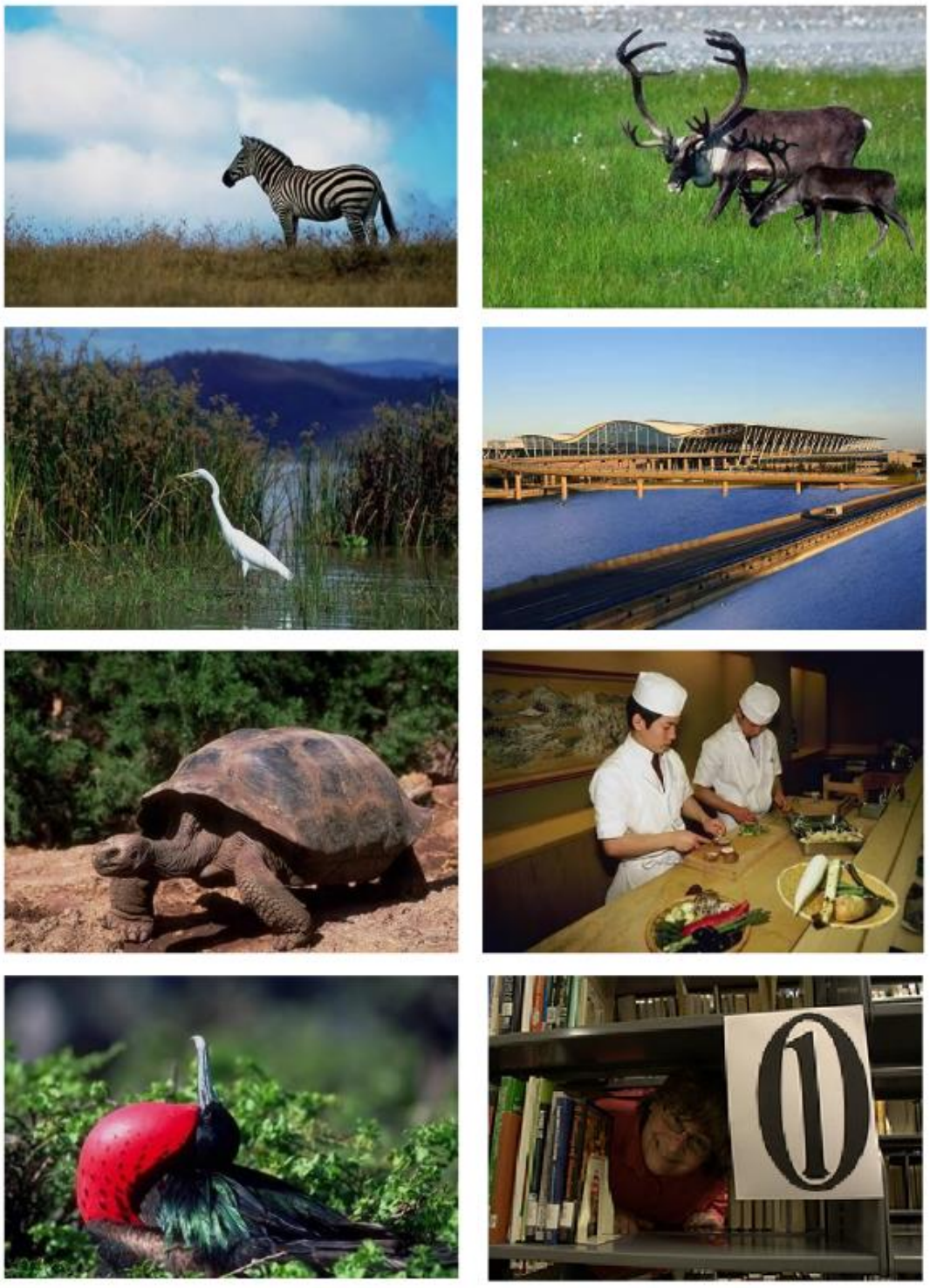


Figure 2.1 Authentic (left-sided) and forged (right-sided) images [20]

2.6 Chapter Summary

It has been noted in this chapter that digital image forensic discovers its meaning in today's situations where a large amount of data is exchanged in the form of images. This chapter deals with the basics of image forensics specifically image splicing are discussed along with different methods that can be used to identify the manipulation in an image. It also consists of details of the database CASIA v2.0 is given which is used for training and testing purpose. As an example, some of the authentic and forged images are also presented for a better understanding of the database. The thesis objectives are fulfilled by the detailed discussion in this chapter taken as a base for the proposed method. The steps involved in the proposed method is represented in the next chapter.

Chapter 3

Image Splicing Detection using CNN

3.1 Introduction

Existing methods of forgery detection embrace certain techniques of depiction to merge the data obtained by estimators of proof. However, each method of depiction has its own constraints and disadvantages. CNN's have increasingly proven enormous success in the classification of images and other activities related to computer vision. Traditional neural networks use the actual image as the entry in RGB channels as it includes picture data like color and structural properties. In this section, multiple steps engaged in the identification of image splicing are briefly described. The fundamental philosophy behind this work is to investigate the JPEG artifacts remaining in an image while resaving the image. Different error levels produced using ELA of authentic and forged images are investigated for this work. CNN is used for the classification purpose. The ability of CNN for automatic feature extraction is utilized for the better extraction of features form the error level images is studied. The architecture with parameters of the used CNN is explained in this section.

3.2 Overview of the proposed method

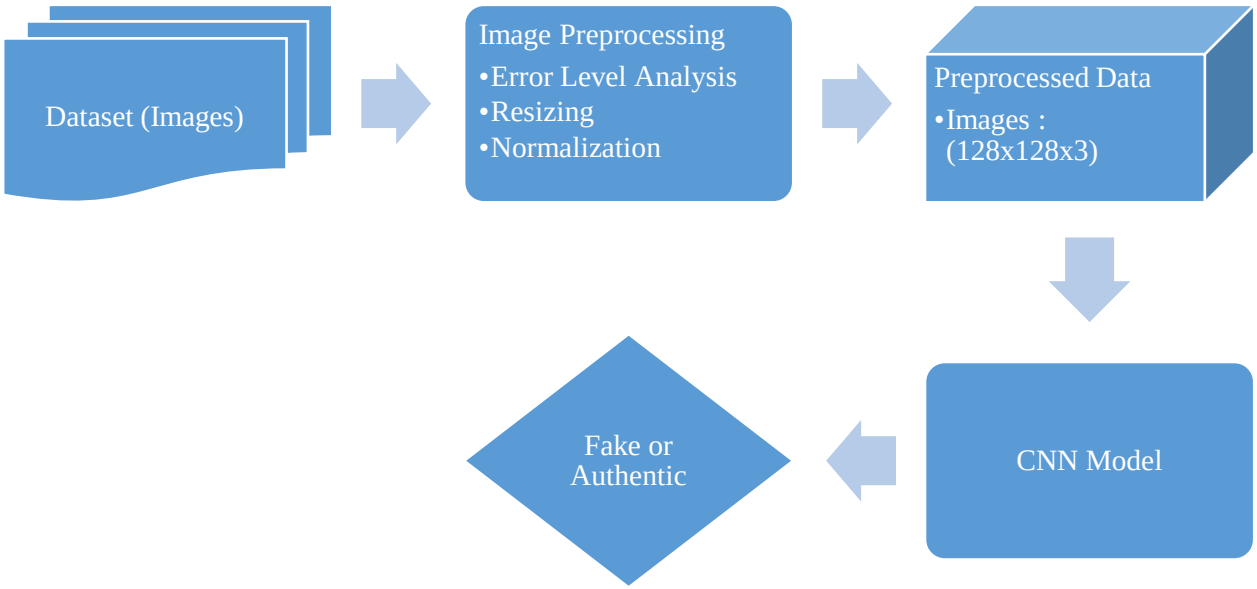


Figure 3.1 Proposed method using CNN

To classify tampered and authentic images, a deep learning architecture is used to extract features automatically. Specifically, error level analysis (ELA) is utilized to detect areas of the image that may have been tampered. These error level images are then resized to 128 × 128 × 3 sized images. Then the data is normalized to get values between zero and one. By making it so the data is scaled down so that the model can easily estimate the values. By processing the computation of the low values required is less. This whole process is preprocessing. After image preprocessing, the images are given to CNN. It is used to train a supervised model for classifying whether an image is authentic or tampered. CNN's generally use several convolutional layers where primary layers decrease input pictures to more particular features while closer layers discover more complicated, descriptive features. CNN can discover comparable properties to several handmade features used in past computer vision research studies. Figure 3.1 presents the flow chart of the model used.

3.3 Image Preprocessing

Image Pre-processing is aimed at improving picture information, suppressing reluctant distortions or enhancing certain image attributes that are essential for any further processing. Although, geometric image changes, e.g. translation, scaling, and rotation are categorized as pre-processing operations here since comparable algorithms are used [54]. For this work, ELA is used as the image pre-processing tool which gives comparable results between original and manipulated images. After that, the data is resized to 128 × 128 × 3 size. Then the data is normalized. The aim of normalization is to alter numeric column data in the dataset to a general scale without altering value range distinctions. Normalization is done to improve data integrity. Then this preprocessed data is fed to the CNN model.

3.3.1 Error Level Analysis

ELA is the evaluation of compression artifacts in the digital image with lossy compression, for example, JPEG. It is one of the methodologies used for detecting image tampering by way of storing pictures on a certain quality level and calculating the comparison between its levels. In general, this technique is performed on an image that has a lossy format (lossy compression). Picture type used in mining this data is JPEG. On JPEG images, compression is done independently for each 8 × 8 pixels in the image. If an image is not manipulated, each 8 × 8 pixels of an image must have had the same error level [54].

A block-based compression approach is utilized for lossy JPEG compression. Every 8 × 8 block that does not overlap within an image is treated separately. The JPEG compression and recompression procedures on an 8 × 8 uncompressed block will be considered.

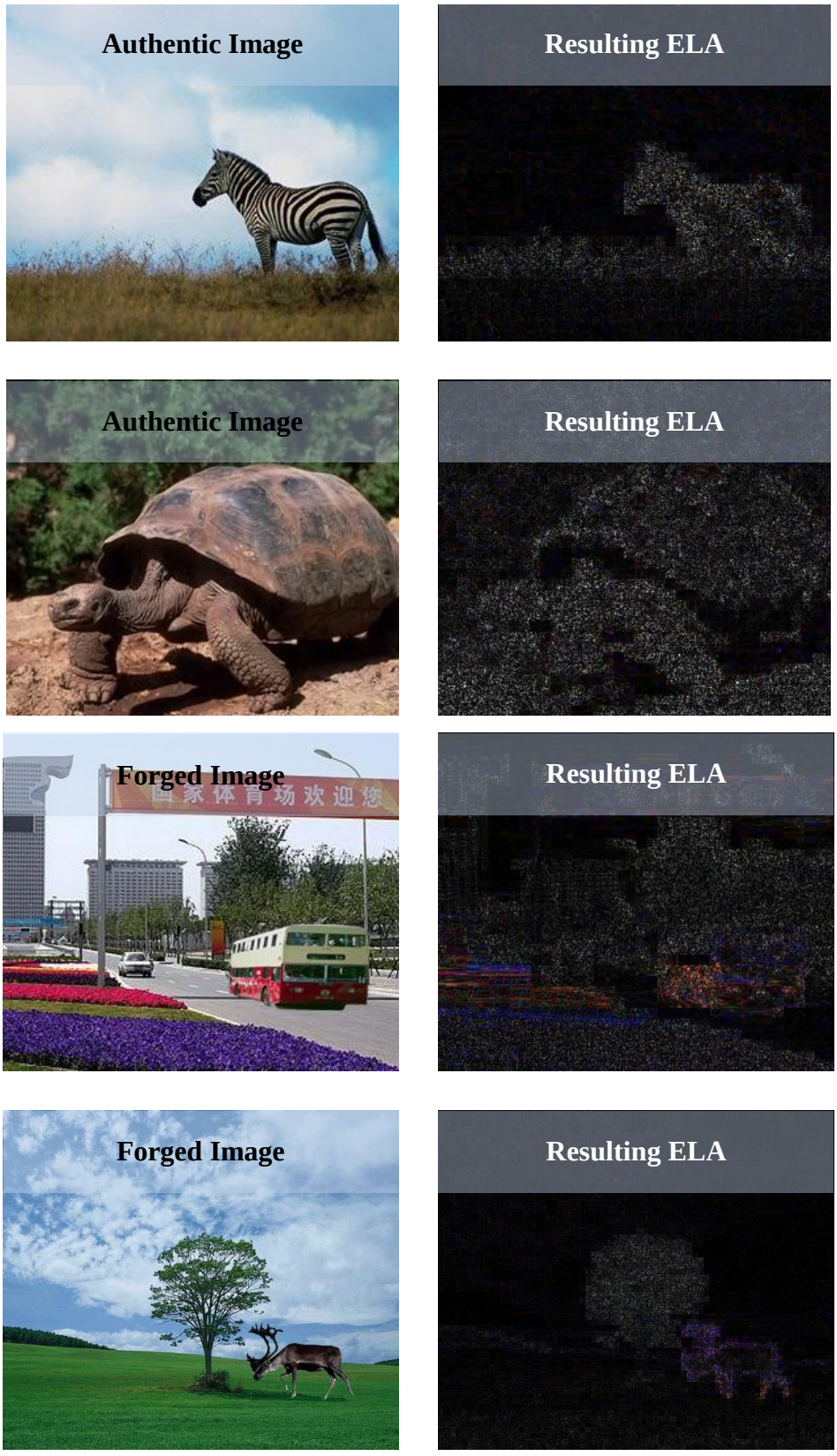


Figure 3.2 Examples showing ELA error with quality level 90

Firstly, Discrete Cosine Transform (DCT) is performed on the image data to transform from the spatial domain into frequency domain and then a quantization table quantizes the frequency coefficient. The error originated at this point is called quantization error, which was not previously there in compressed JPEG. The error generated because of data failure for the images. The lossless entropy encoding further compresses the quantization coefficient. To accurately recover the quantized coefficient, firstly, compressed file is decoded in JPEG decompression which is then multiplied by the table to get the de-quantized coefficient. For obtaining the pixel values in real valued representation inverse discrete cosine transform (IDCT) is accomplished. Rounding and truncated errors are generated in this stage. The rounding error typically happens in each block, whereas there is no truncation error [54]. Usually, lower the quality factor (bigger steps of quantization), the greater probability of the error of truncation. In this work, quality factors of 80, 85, 90 and 95 are considered.

A small amount of color is changed after saving the image in JPEG. After resaving the image, ELA highlights the pixels which are most vulnerable to color degradation. Typically, edits stand out as a region with a greater capacity for degradation relative to the remaining picture. ELA resaves the picture at a quality level of the given JPEG. This resaved image presents a known measure of blunder over the whole picture. The resaved picture compared against the original picture. If the picture is totally unmodified, all 8 × 8 squares should have the comparable potential for errors. Each square should degrade at approximately the equivalent rate if a picture is not altered and resaved [54].

If an image is altered, then each 8 × 8 square affected by the change should have a greater capacity for error than the remaining part of the image. Changed regions will show up with a higher potential error level. Therefore, high potential error levels show the modified part of the forged image. By analyzing these error features image can be distinguished between original and manipulated images. By its tendency, ELA does not work on lossless compressions [54]. For PNG format images, ELA does not work because conversion is lossless and it gives a very high error in textures and edges. Some examples are shown in Figure 3.2 information about ELA patterns of authentic and forged images.

3.4 CNN Architecture

The architecture of proposed CNN is shown in Figure 3.3 with the architecture's comprehensive configurations and an additional explanation for the method. After preprocessing of the image, i.e., ELA, resizing and normalizing the input images data is given to the CNN. The CNN architecture consists of 4 convolution layers, with different configurations of kernel size, no. of filters, strides, and padding. ReLU and Leaky ReLU are used to introduce non-linear complicated functional mappings among inputs and response variable [61].

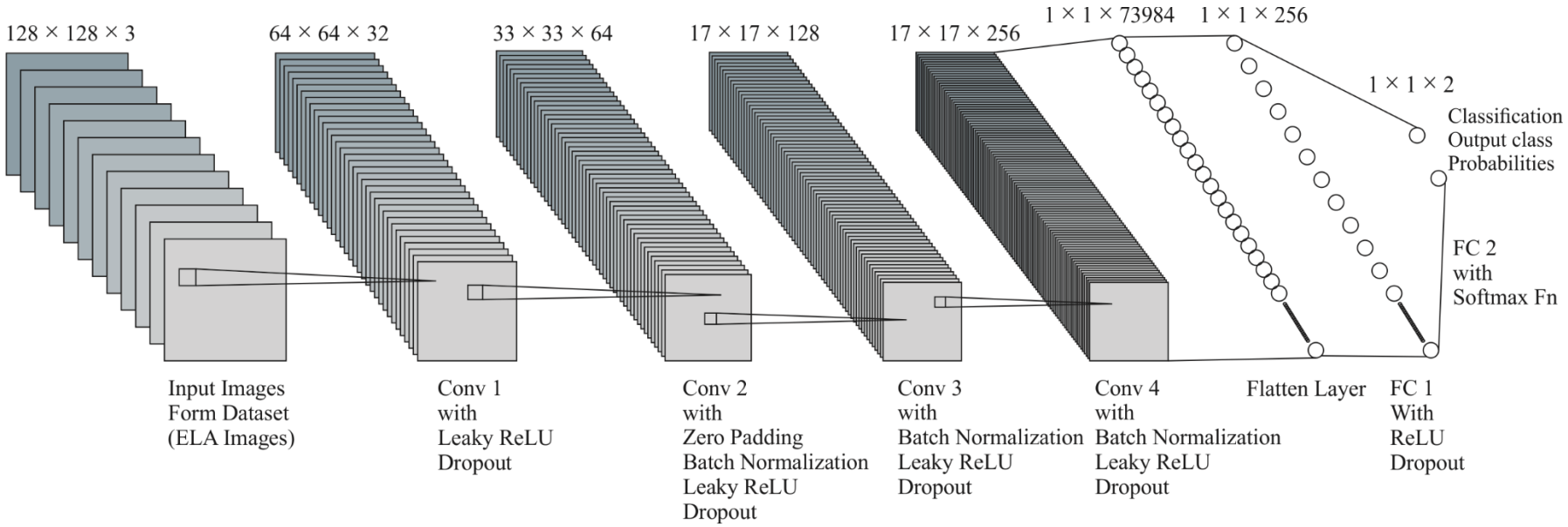


Figure 3.3 Architecture of proposed CNN

To boost the performance of these layers operations like batch normalization, and dropout are utilized. There two fully connected layers are utilized for classification. In the end, softmax function is used to get the probabilities for each class. As the data passed through the network, the weights are updated itself by making use of the parameters and input images given to CNN. Starting from first the convolution layer, it will extract the low-level features as the data moved to next convolution layer this layer will make use of the extracted low level features to create more of high level features than the first one [37]. As the data moving towards the next layers, high-level features are acquired by utilizing the previously extracted features.

In this work, the proposed method is used to solve the problem of detecting the splicing forgery in images. Images of size 128 × 128 are taken to give to model. Error level RGB images are given to the network as shown in architecture in Figure 3.3. Firstly, the convolution layer named Conv 1 convolves the Error level images with 32 kernels of size 3 × 3 with a stride of 2 × 2. After convolving from Conv 1, size of output feature map comes of size 64 × 64 × 32. Therefore, the resolution of a feature map that is extracted by the first layer is "64 × 64" and 32 number these feature maps are obtained. Now, these feature maps are fed to next convolution layer named Conv 2 as input, it convolves it with 64 kernels of size 3 × 3. With this layer zero paddings, batch normalization, Leaky ReLU, and dropout are performed. Next convolutional layers named Conv 3 and Conv 4, apply 128 kernels of size 3 × 3 and 256 kernels of size 3 × 3 respectively. With them, batch normalization, Leaky ReLU, and dropout are achieved. After that, the feature maps are flattened in terms of single neurons and given to fully connected layers. The flattening layer has 73984 neurons. The first fully connected layer named FC 1 having 256 neurons will learn from the flatten layer. The ReLU and dropout are applied to FC 1 layer. Because the forgery detection is a two-class problem, therefore, last fully connected layer has two neurons. The output is fed to softmax function to get predicted probabilities of each class.

3.5 Parameters of CNN

The used CNN is having learning rate of 0.005, optimizer used is RMSProp, the loss is categorical cross-entropy, batch size of 32, and epoch are 100. For this work, early-stopping is utilized so that the network should overcome the problem of overfitting the data. It should stop training when it is not improving performance. Therefore, early-stopping is set to 10. By monitoring the validation loss, if the loss is not improving then after 10 epochs the network should stop training. Table 3.1 shows the different parameters of the network. In the Table 3.1 kernel size is the filter size, total learnable parameters are obtained by calculating the sum of a total number of weights and bias for the convolutional layer. For batch normalization, total learnable paramenters are calculated using the sum of a number of offset and scale.

The stochastic gradient descent approach trains deep learning models. It is an optimization algorithm that uses examples from the training dataset to estimate the error gradient for the present state of the network, then updates the model's weights using backpropagation. During training, the amount of changed weights is referred to as the "learning rate". In particular, the learning rate is a small positive value lies between zero and one. Learning rate is a hyperparameter that can be configured during training.

Table 3.1 CNN Parameters

#	Layer type	Parameter	Value	Learnables	Total Learnables
1	Input Images	Image Size	128 × 128 × 3	-	-
2	Convolution	Kernel size \# of filters Stride Pad	3 × 3 × 3 32 2 Same	Weights 3 × 3 × 3 × 32 Bias 1 × 1 × 32	896
3	Leaky ReLU	Scale	0.1	-	-
4	Dropout	Dropout Rate	0.26	-	-
5	Convolution	Kernel size \# of filters Stride Pad	3 × 3 × 32 64 2 Same	Weights 3 × 3 × 32 × 64 Bias 1 × 1 × 64	18496
6	Zero Padding	Pad	((0,1),(0,1))	-	-
7	Batch Normalization	-	-	Offset 1 × 1 × 64 Scale 1 × 1 × 64	128

8	Leaky ReLU	Scale	0.1	-	-
9	Dropout	Dropout Rate	0.26	-	-
10	Convolution	Kernel size \# of filters Stride Pad	3 × 3 × 64 128 2 Same	Weights 3 × 3 × 64 × 128 Bias 1 × 1 × 128	73856
11	Batch Normalization	-	-	Offset 1 × 1 × 128 Scale 1 × 1 × 128	256
12	Leaky ReLU	Scale	0.1	-	-
13	Dropout	Dropout Rate	0.26	-	-
14	Convolution	Kernel size \# of filters Stride Pad	3 × 3 × 128 256 1 Same	Weights 3 × 3 × 128 × 256 Bias 1 × 1 × 256	295168
15	Batch Normalization	-	-	Offset 1 × 1 × 256 Scale 1 × 1 × 256	512
16	Leaky ReLU	Scale	0.1	-	-
17	Dropout	Dropout Rate	0.26	-	-
18	Flatten	Kernel size	1 × 1 × 73984	-	-

19	Fully connected	Kernel size \# of Combinations	1 × 1 × 73984 256	Weights 256 × 73984 Bias 256 × 1	18940160
20	ReLU	-	-	-	-
21	Dropout	Dropout Rate	0.5	-	-
22	Fully connected	Kernel size \# of Combinations	1 × 1 × 256 2	Weights 2 × 256 Bias 2 × 1	514
23	Softmax	-	-	-	-

The total number of parameters is 19,330,882 from which trainable parameters are 19,329,986 and non-trainable parameters are 896. Due to dropout set to convolution layers of 0.26, it means 26% of neurons at those layers will not be considered. It is used to improve the efficiency of the network so that the network should not face the issue of overfitting the data while training. In the last fully connected layer, dropout is set to 0.5, i.e., 50% of neurons are disabled. Batch normalization is performed to speed up the training. After batch normalization moderates the amount by which hidden unit value swing around. Because no activation goes really high therefore higher learning rates can be used [28].

It makes the value of the features between zero and one. It will change weights because it will subtract mean and divide by "standard deviation". By use of batch normalization, two parameters are added to each layer, i.e., "standard deviation" and "mean". "Standard deviation" will be multiplied with each activation and means will be added to each activation to denormalization of the data.

The error propagates back using backpropagation algorithm and weights are scaled by the learning rate. Learning rate regulates how a neural network learns an issue quickly or slowly. To further enhance efficiency, learning rate schedules, monument, and adaptive learning rates are used [35]. For this work, the learning rate is 0.005 means the weights are updated each time with the factor of 0.005 or 0.05% weight error changes each time.

For training of the deep learning neural networks, the database must be passed from the entire network multiple times, therefore epochs are used. Epochs will tell how many times an entire database is passed forward and backward from the network. One epoch means that dataset is passed once from the whole network. The data is divided into several smaller batches since one epoch is too big. 100 epochs are considered for this work. It means 100 times entire data of images passed through the network. Batch size gives information about the amount of training samples exists in a single batch of images. For this work, the batch size is chosen as 32. Therefore, 32 images are processed during a single batch while training. Iterations are the number of batches needed to complete one epoch.

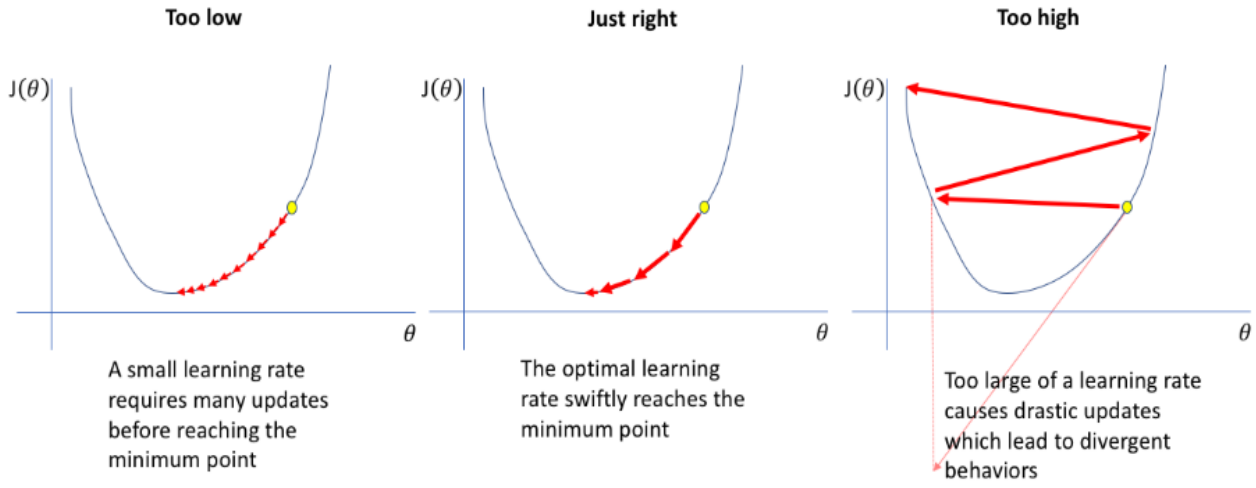


Figure 3.4 Different learning rates with their behavior [35]

Optimizers used in the CNN is RMSProp optimizer. The modification of AdaGrad by changing the aggregation of gradients into an exponentially weighted moving average to perform much better in the non-convex configuration is RMSProp Optimizer's algorithm. When applied to a convex function, AdaGrad is intended to converge quickly. When the training trajectory is added to a non-convex feature to train a neural network, it can move through several different constructions and eventually arrive at a region that is locally convex. AdaGrad reduces the learning rate based on the squared gradient's full history and it may have rendered the learning rate too low before reaching such a convex framework [35].

RMSProp utilizes an exponentially declining average to remove background from its extreme past so that it can progress quickly after finding a convex bowl as if it was an example of the AdaGrad algorithm in that bowl. Compared to AdaGrad, using the moving average presents a fresh hyperparameter, p, which regulates the moving average length scale [35]. Observationally, it has been demonstrated that RMSProp

is an effective and practical algorithm for optimizing deep neural networks. It is presently one of the go-to optimization techniques that deep learning regularly use.

Adam is yet another algorithm for optimizing adaptive learning rates. "Adaptive moments" is the phrase from which the name "Adam" derived. This might be best as a modified version of the combo of RMSProp and momentum with some fundamental differences in the sense of the previous algorithms. First, momentum is immediately integrated into Adam as an estimation of the gradient's first order time (with exponential weighting). Applying momentum to the rescaled gradients is the simplest way to contribute momentum to RMSProp. There is no apparent theoretical motive for using momentum in conjunction with rescaling. Secondly, Adam involves bias adjustments to the calculations of both the first order moments and the second order moments (uncentered) to account for their original initialization. RMSProp also includes a calculation of the second-order (uncentered) moment, but the adjustment factor is missing [35]. Adam is usually considered to be relatively solid in choosing hyperparameters, although sometimes it is necessary to change the learning rate from the recommended rule.

Thus, CNN is able to extract the required features for the classification task. At last, the softmax function will calculate the probabilities of each class by using the extracted features by the upper layers of the network.

3.6 Chapter Summary

In this chapter, the steps followed by the proposed method is briefly explained. How does the method work is elaborated as ELA works for image forensics and classification of images through CNN. The architecture and the parameters used for the method and their role are analyzed. Extraction of features by the network at every layer, use of activation function, normalization, etc. The results that are obtained are described in the next chapter along with the related discussions.

Chapter 4

Results and Discussions

4.1 Introduction

The automatic classification of image forgery using ELA and CNN verified to be effective when investigated over CASIA v2.0 image tampering dataset. The dataset of images is processed by ELA, resizing and normalizing operations. These processed images show significant error differences between actual images and modified images as described in the previous section. By learning for these processed images, CNN is able to classify them by automatically extracting vital features. In this chapter, various steps involved in the whole process with their results and performance analysis is done after classification. The method is tested with different quality levels of JPEG compression in order to find ELA of images. This technique is also been compared with existing established state of art techniques, which validates its prestige in modern eras.

The dataset is consists of 7437 are authentic images and 5123 are forged images are shown by Figure 4.1.

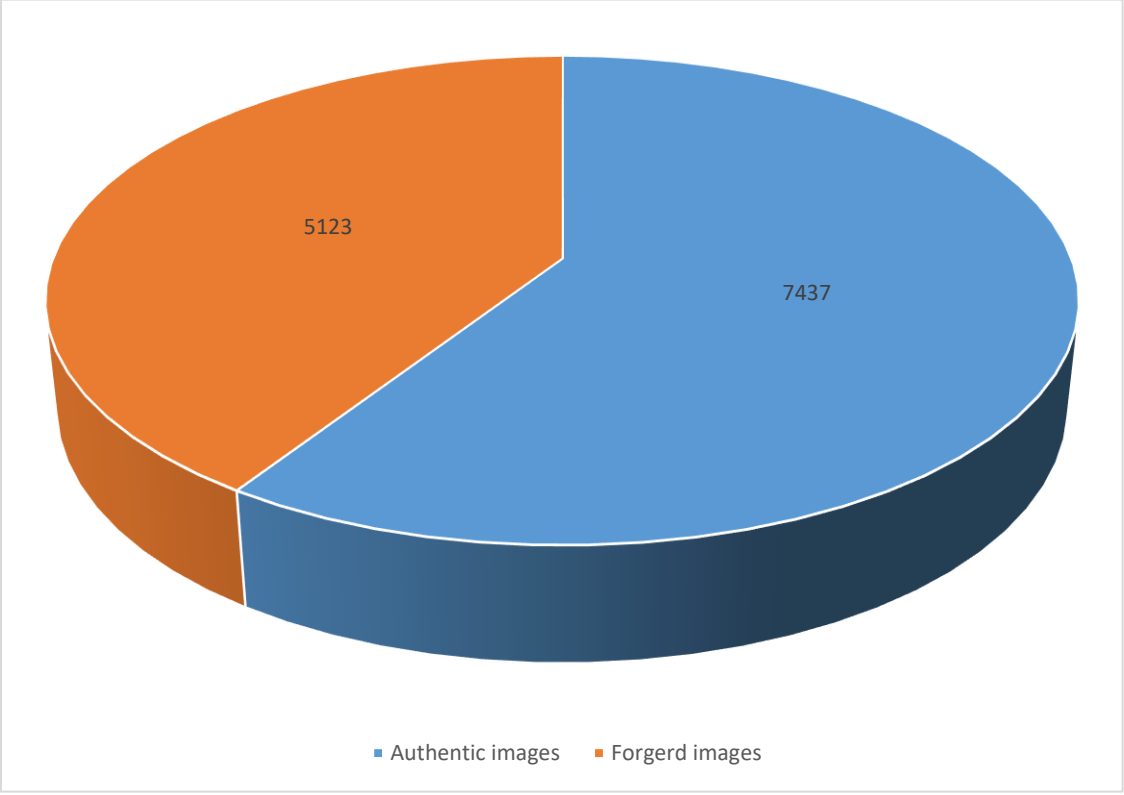


Figure 4.1 Distribution of images in the dataset

4.2 Experimental Results

The proposed method is evaluated with different settings of training data and testing data. The training dataset and the testing dataset are in the ratio of 60%, 70%, 80%, and 40%, 30%, 20%, respectively are considered. The training data and the testing data is also shuffled randomly while training and testing procedure. It is very important that the dataset is well shuffled before training the machine learning model to avoid any bias/pattern element in the split datasets. If data is not shuffled, then the information can be arranged or comparable information spots are next to each other, resulting in slow convergence. Shuffling your information after each epoch guarantees you are not "caught" with too many poor batches. The aim of shuffling information is to reduce variability and ensure that designs stay general and less overfit [35]. Particularly, data with 60%, 70% and 80% as training and 40%, 30% and 20% as testing with different JPEG compression quality levels are studied in this chapter. Results are shown, in the form of accuracy and loss of training and testing are shown.

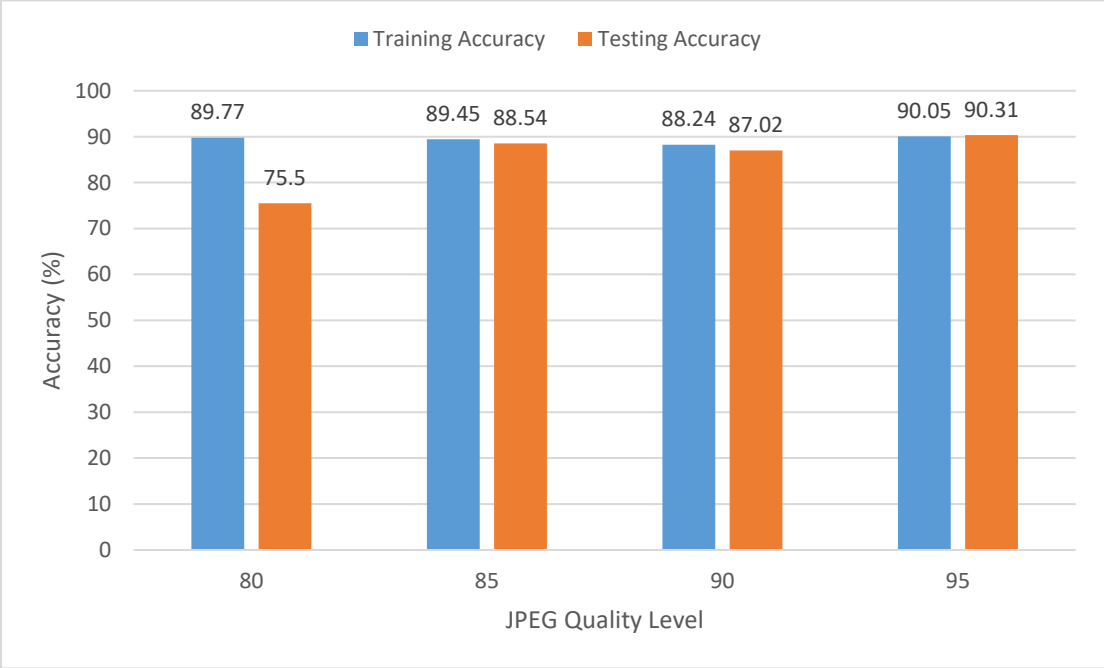


Figure 4.2 Training and Testing accuracy at different quality levels with 60% training data

Figures 4.2 to 4.7 show accuracy of the training and testing of the presented method with different quality levels of JPEG compression, i.e., 80, 85, 90, and 95 by using images in the database while training data having 60%, 70%, and 80% of data and testing data having 40%, 30% and 20% of images. It can be observed from Figure 4.2 that training accuracy of 90.05% and testing accuracy of 90.31% is achieved when JPEG compression quality level is 95. The training loss 0.2282 is less with JPEG compression quality

level is 80 but the testing loss is lower in both when JPEG compression quality level is 85 and 95 are analyzed from Figure 4.3. In terms of accuracy and loss at JPEG compression quality level, 95 is considered best with training data 60%.

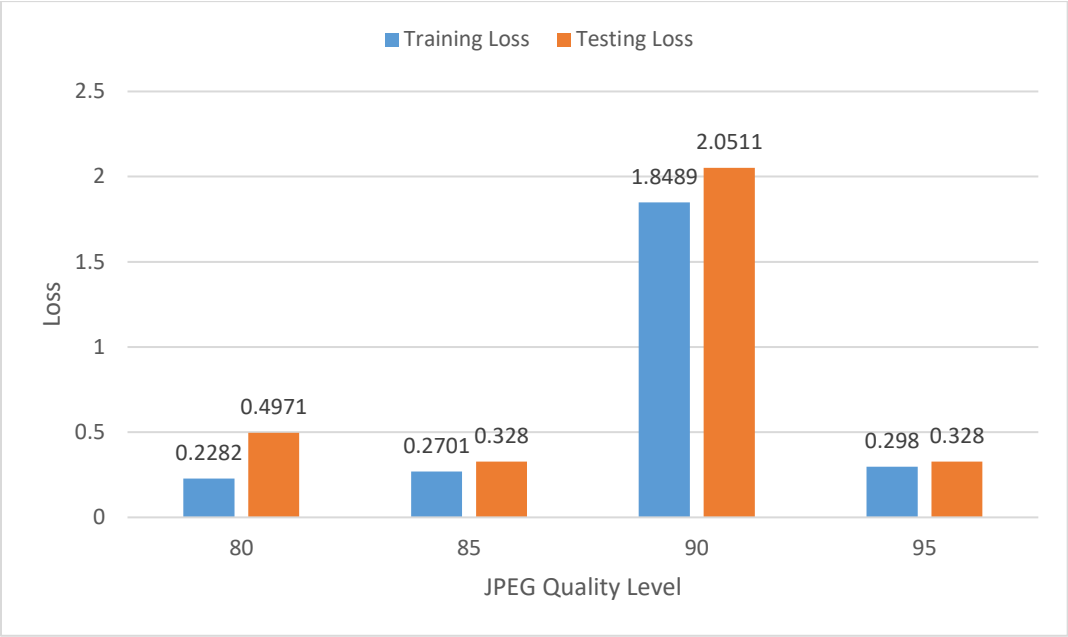


Figure 4.3 Training and Testing loss at different quality levels with 60% training data

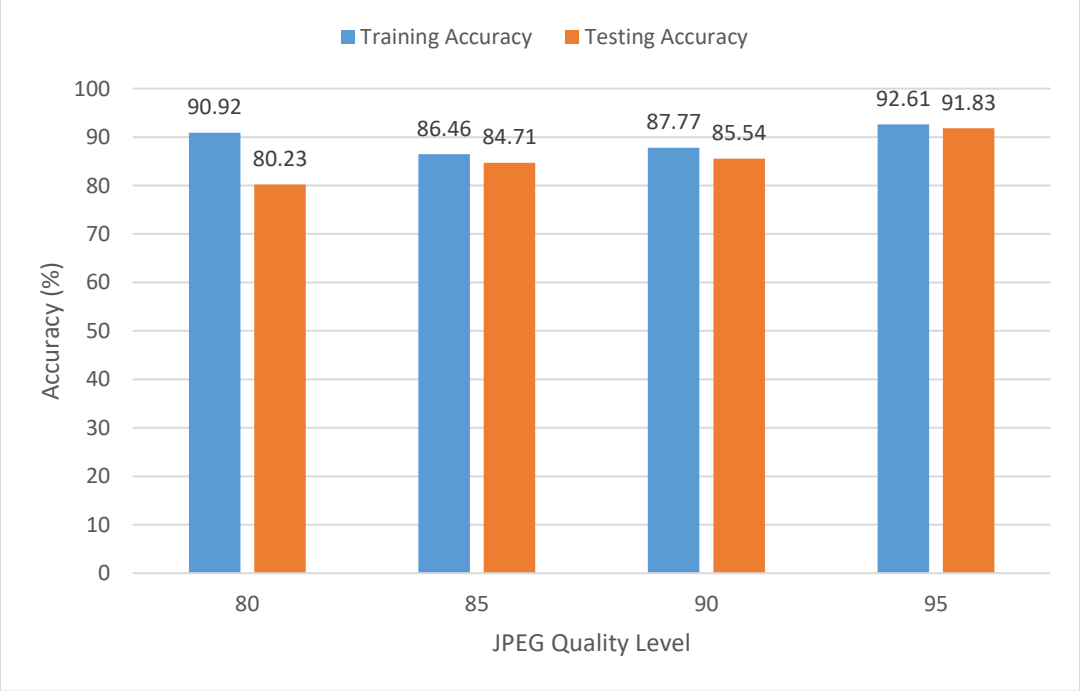


Figure 4.4 Training and Testing accuracy at different quality levels with 70% training data

Figure 4.4 shows the training accuracy of 92.61% and testing accuracy of 91.83% is with quality level is 95 when tested on 30% data. In Figure 4.5 training loss of 0.205 at quality level 80 and testing loss of 0.256 is best considered. Therefore, again JPEG compression quality level 95 is best for training data 70%.

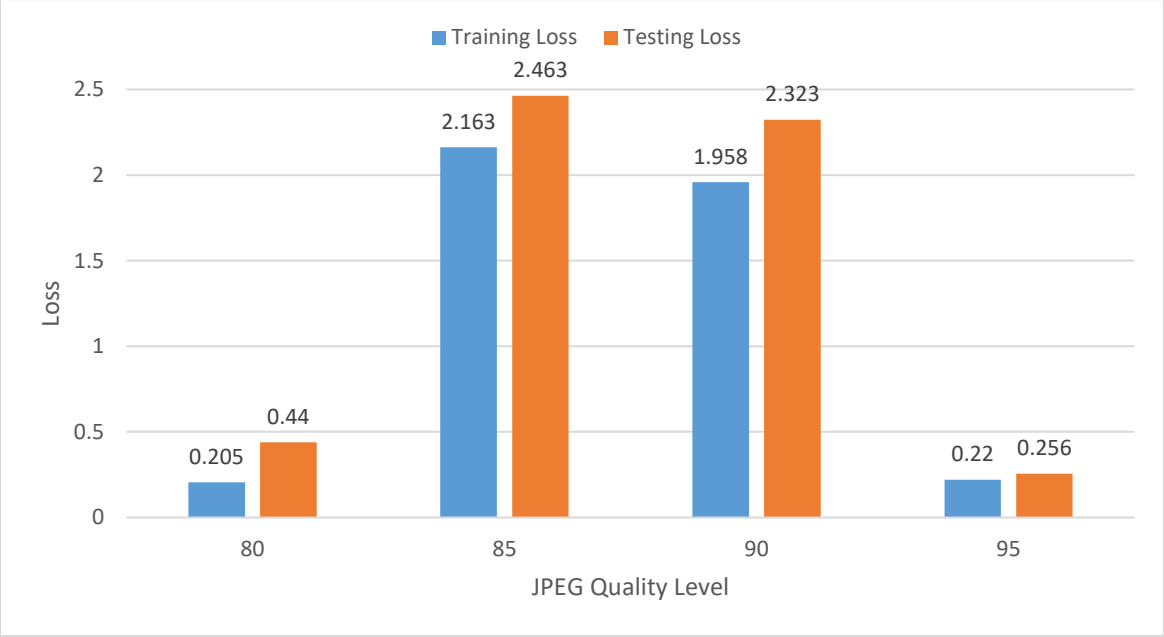


Figure 4.5 Training and Testing loss at different quality levels with 70% training data

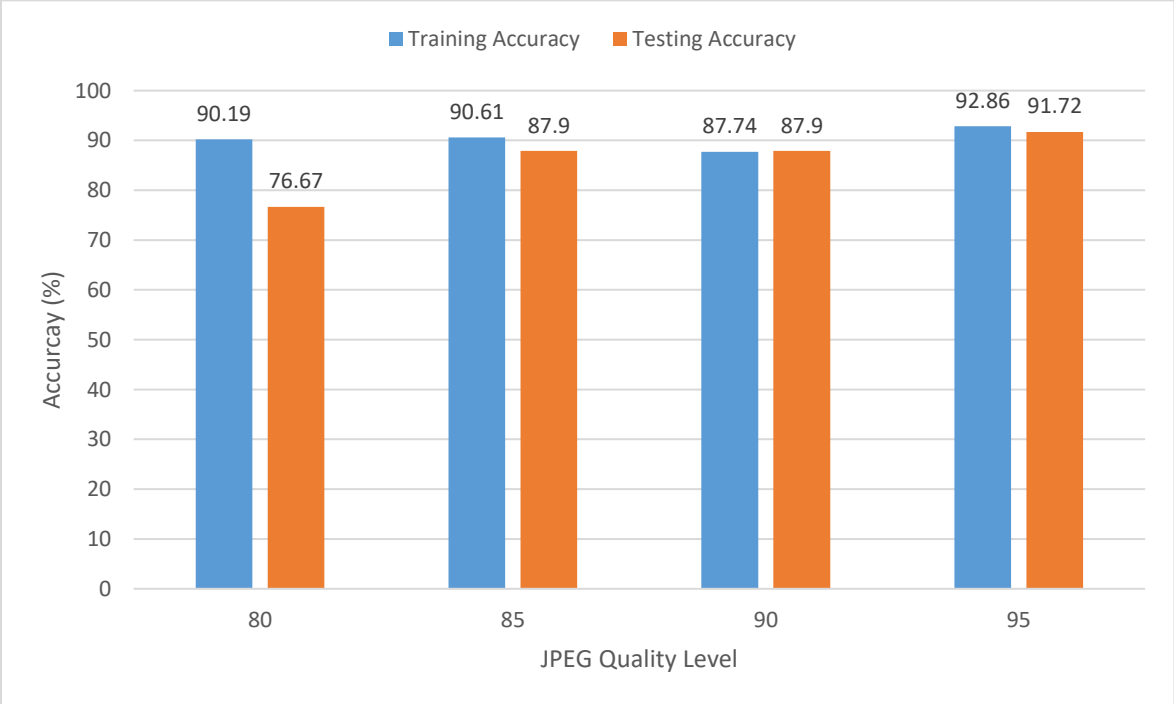


Figure 4.6 Training and Testing accuracy at different quality levels with 80% training data

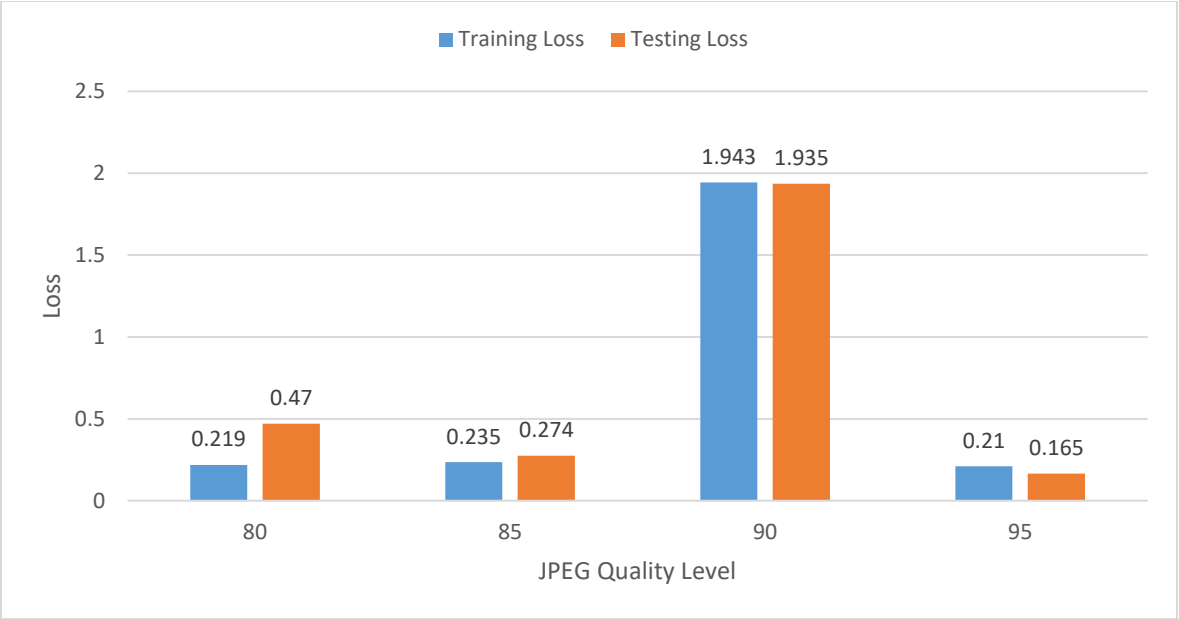


Figure 4.7 Training and Testing loss at different quality levels with 80% training data

From Figure 4.6, it can be realized that training accuracy 92.86 is found at 95 quality level of JPEG compression and testing accuracy of 91.72% is obtained. Training and testing loss is presented in Figure 4.7, the best training loss is 0.21 and testing loss of 0.165 is out. The best accuracy and loss of training data 80% again 95 quality level of JPEG compression gives the best results. Overall, the best performance is considered from these results is carried out by using training data of 70% and testing data of 30% with JPEG compression quality level 95. For best performance, the training accuracy of 91.62% and testing accuracy of 91.83% with training loss of 0.22 and testing loss of 0.256 is measured.

4.3 Performance analysis

The performance analysis based on parameters, i.e., accuracy, precision, recall, and F1 score is evaluated in this section. The accuracy percentage is accomplished by taking the ratio of number of correctly classified images to the total number of images. Equation 4.1 gives the formula of accuracy.

$$Accuracy\ Percentage = \frac{Number\ of\ correctly\ classified\ images}{Total\ number\ of\ images}\ X\ 100 \tag{4.1}$$

For classification tasks, precision is given by the ratio of true positive to the sum of true positive and false positive and recall (also called sensitivity) is given by the ratio of true positive to the sum of true positive and false negative. The formula of precision and recall are given by Equation 4.2 and 4.3.

$$Precision = \frac{True\ positive}{True\ positive + False\ positive} \tag{4.2}$$

$$Recall = \frac{True\ positive}{True\ positive + False\ negative} \tag{4.3}$$

F1 score measures the test's accuracy by investigating the harmonic mean of precision and recall as evaluated by Equation 4.4.

$$F1\ score = 2 * \frac{precision * recall}{precision + recall} \tag{4.4}$$

Specificity estimates the proportion of actual negatives that are correctly identified. Its mathematical operation is given in Equation 4.5.

$$Specificity = \frac{True\ negative}{True\ negative + False\ positive} \tag{4.5}$$

The training loss, testing loss are shown in a) part and training accuracy, and testing accuracy is shown in b) part of Figure 4.9.

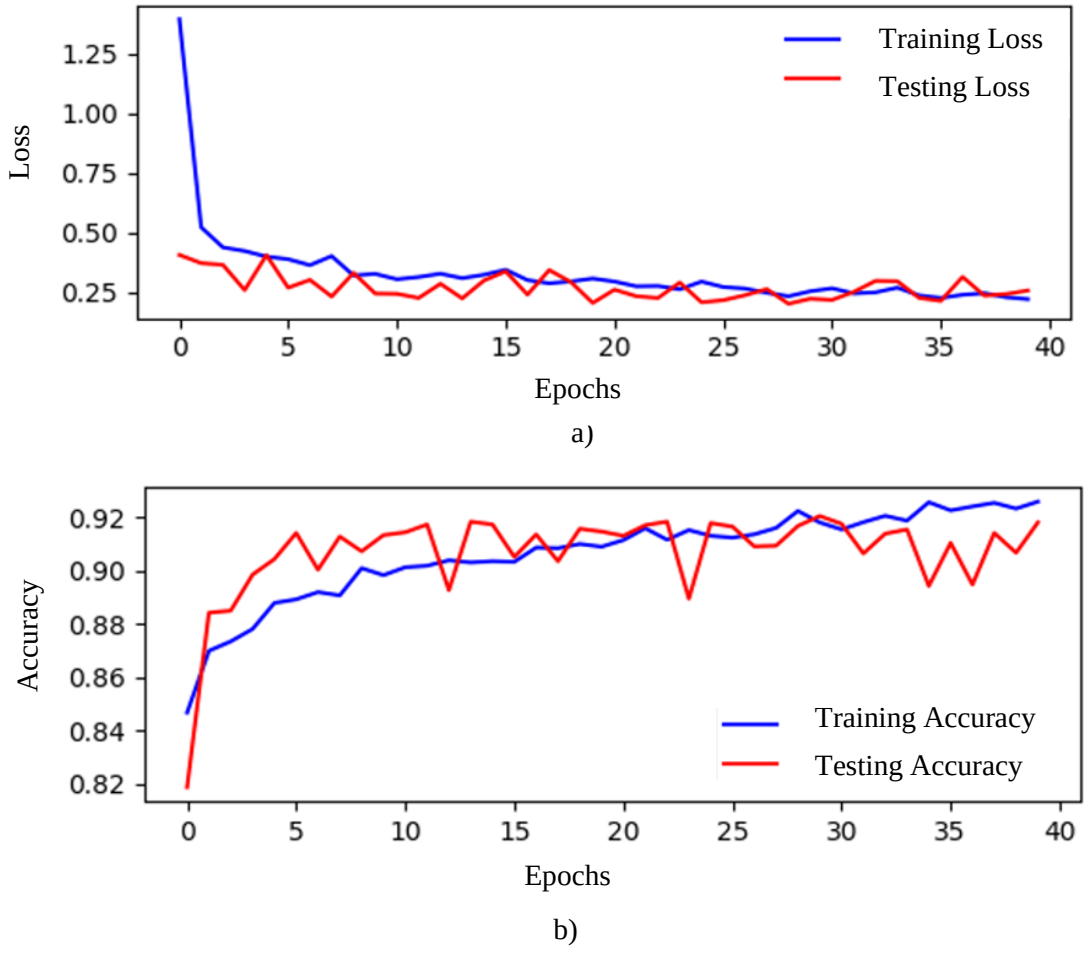


Figure 4.8 Training and testing curves w.r.t. epochs a) Loss b) Accuracy

The network runs for 40 epochs means 40 times the whole dataset is shuffled and passed through the network. Figure 4.6 shows that as the epochs are increasing, the accuracy curve is also increasing and loss curve is decreasing. The X-axis shows in the above Figure show epochs and the Y-axis shows the accuracy of the network. After 40 epochs the network stops training because of early stopping patience set to 10. Early stopping is used to avoid overfitting of the data. Too little training epochs means the model goes for under fitting and too much training epochs means a model with overfitting. If there is overfitting of the model then the network performs better in terms of training accuracy and loss but might not be working better for testing accuracy. Therefore, to overcome the issue of overfitting, the model not trained on full 100 epochs. This means after 30 epochs, the loss of testing is not improving, therefore, it stops after 40 epochs.

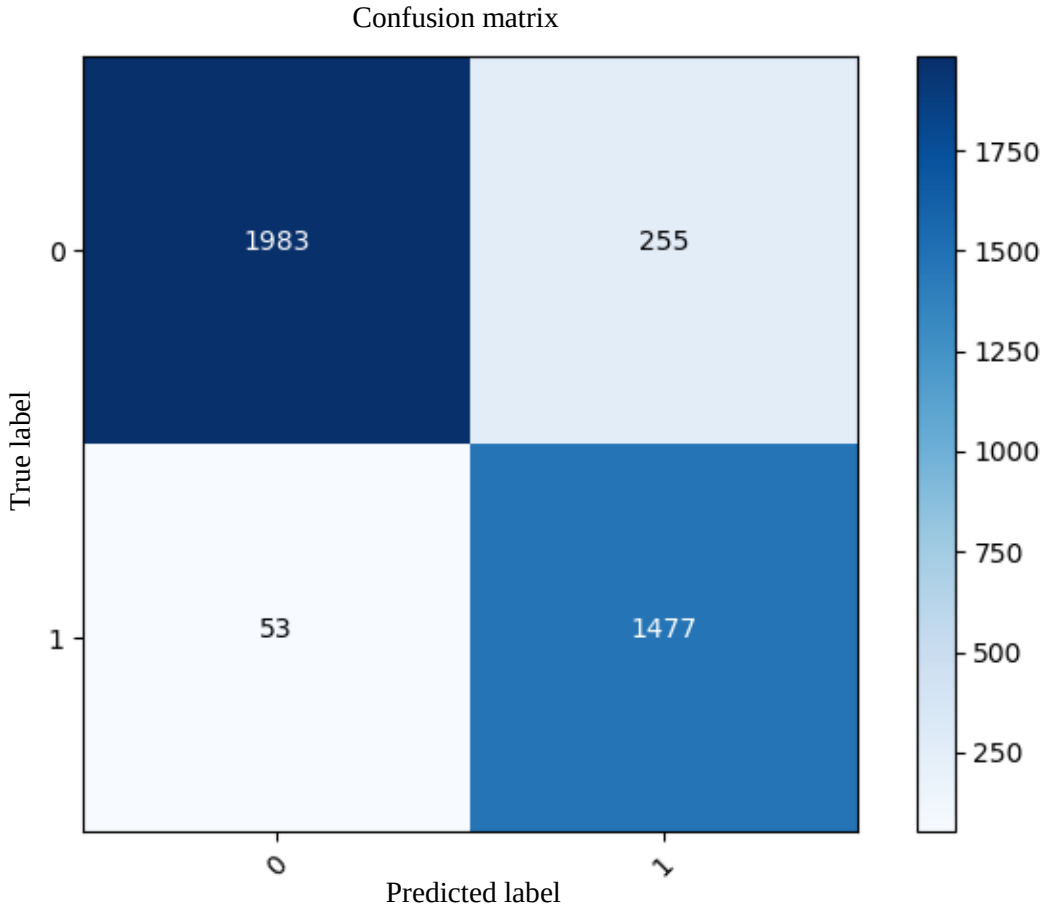


Figure 4.9 Confusion matrix

The confusion matrix of results in 3768 tested images is shown in Figure 4.7. The confusion matrix has 1983 true positives, 1477 true negatives, 255 false negatives, and 53 false positives. The method gives the performance parameters are given in Table 4.1. The proposed method is tested on CASIA v2.0 Database.

There are 12560 images in the database out of which 7437 are authentic images and 5123 are manipulated images. There are different size images in the database, which varies from 240 × 160 to 900 × 600 pixels. The dataset consists of .jpg and .tiff format images. The database is divided into a training set and testing set from which 70% is training set and 30 % is testing set. The parameters that are calculated after obtaining the true positives, true negatives, false positive and false negatives from the confusion matrix are shown in Table 4.1.

Table 4.1 Calculated performance parameters

Parameters	***Value***
Accuracy percentage	91.83%
Precision	0.9739
Recall	0.8860
F1 score	0.9279
Specificity	0.9653

4.4 Comparative analysis

The comparison between different existing methods is shown in Table 4.2. The proposed method is compared with the approach used by Jaiswal and Srivastava [29], Majumder, *et al.* [42], Walia, *et al.* [56], Zhang, *et al.* [65] and Rao, *et al.* [52]. In [29] deep learning model, i.e. CNN is utilized to classify two-class images. Resnet-50 a pre-trained deep learning model used to extract features for the classification task. By directly applying input images the CNN give the accuracy of 70.26% only. Further, Majumder, *et al.* [42] used shallow CNN's suffice to extract necessary features. To keep a number of parameters small without using max-pooling, the method utilized high sized filters. With the use of the proposed technique, the author got 79% accuracy for the binary classification of authentic and forged images. This may happen due to the direct extraction of features using CNN without any preprocessing. Another existing method used for the detection of image splicing is given by Walia, *et al.* [56]. The author used spatial-structure based features and frequency transform based features. It has been concluded that form LBP, DCT, and DWT features, DCT based features giving the best performance. LBP features do not work well in the existence of geometrical transformations. If post-processing procedures are applied then frequency domain features do not work well. This approach performs better the previous method with 87.60% accuracy.

Zhang, *et al.* [65] presented two-phase deep learning scheme, i.e. stacked autoencoder integrated with contextual information of every mask or patch. In the proposed scheme, stacked autoencoder is used to extract composite features of patches for the detection of tampered regions. By incorporating contextual information, improvement of detection of tampering is attained with the accuracy of 87.51 %. Rao, *et al.* [52] used CNN to learn hierarchical representations. For the calculation of spatial rich model residual maps (SRM), the weights of the first layer are made as high pass filter. By using pre-trained CNN as patch descriptor the author got the accuracy of 97.83%. Figure 4.6 indicated that our method works better than some of these methods by extracting the necessary features by CNN with an accuracy of 91.83%. It can be seen from Table 4.2 that our method is more precise than other existing methods.

Table 4.2: Proposed method's comparison with existing methods

Methods	***Accuracy (%)***	***Precision (%)***	***F1 Score***
Jaiswal and Srivastava [29]	70.26%	63.39%	-
Majumder, et al. [42]	79%	-	-
Walia, et al. [56]	87.11%	81.53%	0.85
Zhang, et al. [65]	87.51%	80.65%	-
Rao, et al. [52]	97.83%	-	-
Proposed method	91.83%	97.39%	0.9279

The proposed CNN is implemented using Tensorflow and conducted on Nvidia GeForce GTX1050 Ti GPU of 4 GB RAM. The proposed method takes approximately half an hour for preprocessing and training of the network on our laptop with Intel i7 7th Generation Processor and 16 GB of RAM. While the method having better accuracy needs more epochs. 250 epochs are used to learn features from the network in [52]. There less number of features are acquired at each layer and eight convolutional layers, two pooling layers, and one fully connected layer is used but in our method more number of features are obtained at each layer. In our method CNN uses four convolutional layers and two fully connected layers. Presented network is making use of batch normalization, dropout and activation functions. These operations makes the data in appropriate form so that the network can quickly learn the features and perform better. Therefore, it has learned from only 40 epochs. The network learned from only by 40 times data passed through the network.

Consequently, the network is capable of learning distinguishable features for each class with very less time and computation.

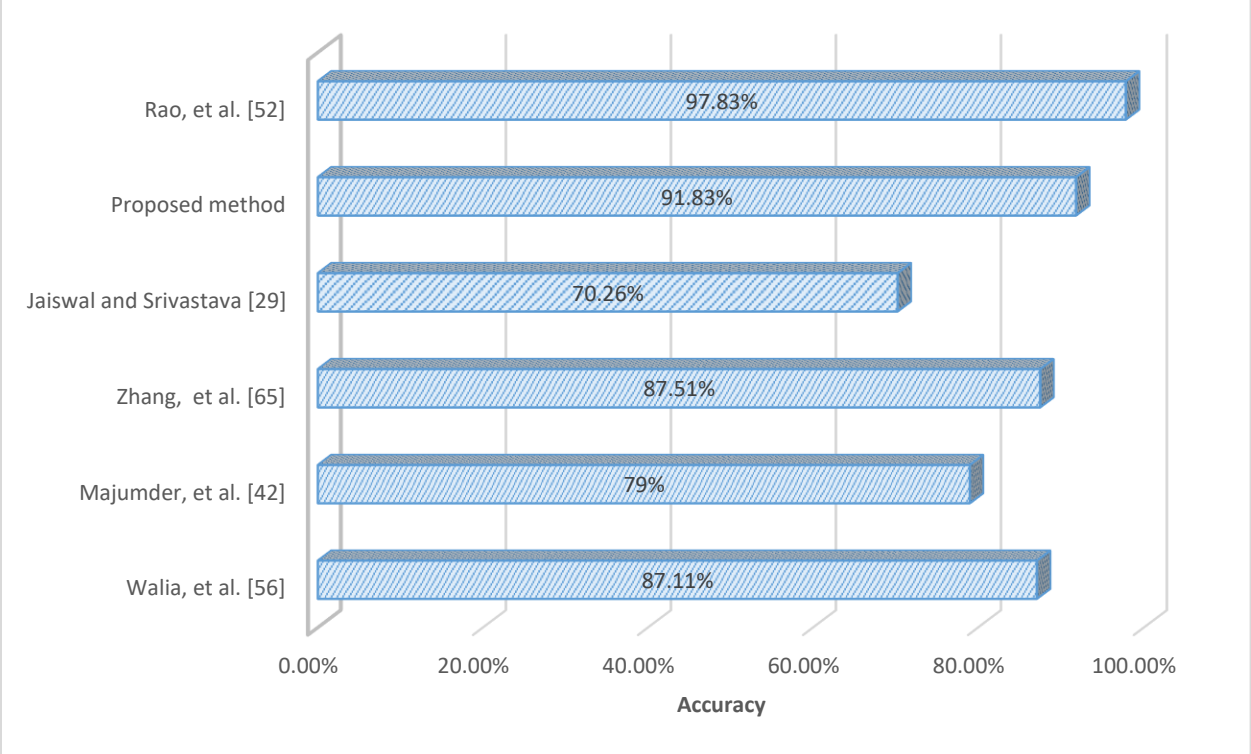


Figure 4.10 Comparison of present work with some recent works

4.5 Chapter Summary

In this chapter, inspected results acquired by using the proposed technique such as image preprocessing like ELA, resizing, normalizing and the learning of weights of CNN network with the help of final results in the form of table and graph representing the performance of splicing detection of images is discussed. The proposed technique is verified with different quality levels of JPEG compression in terms of accuracy and loss are shown by graphs in this chapter. The performance is measured by the use of the confusion matrix with the accuracy of 91.83% for the identification of forged images. Comparison results between the proposed method and the existing relevant approaches are also indicated in this chapter.

Chapter 5

Conclusion and Future work

5.1 Conclusion

The proposed digital image forgery detection method based on error feature is investigated in this work. This work presents a method to detect image forgeries, i.e., image splicing. Firstly, the images are preprocessed to improve the features of an image for further processing. Preprocessing includes operations like ELA, resizing and normalizing the data. In this, ELA with JPEG compression is used to show significant error differences between authentic and forged images. Then this preprocessed data, i.e., error level images are given to the CNN architecture for training and testing. For this work, the dataset is divided into three different settings such as training data 60%, 70%, 80%, and testing data 40%, 30%, 20%, respectively. Furthermore, testing of ELA with different quality levels of JPEG compression, i.e., 80, 85, 90, and 95 is done with different data settings. CNN learns important features by evaluating these error level images and updating its weights and bias for the classification. Data processed by ELA, batch normalization and dropout helped in better feature extraction and fast learning of features by the network. Therefore, network is capable of learning required features in just 40 epochs. This method takes very less time to train the CNN. By analyzing the results, JPEG compression with quality level 95 shows significant differences in error levels between original and manipulated images that are used by network for better performance. With further investigation, it is observed that ELA does not work well for lossless formats like PNG or GIF images. Another disadvantage of ELA is that after several resaves the grid square reaches its minimum amount of error. Therefore, less significant error levels will be found in those ELA images. The proposed method classifies between two classes, i.e., authentic and forged images with an accuracy of 91.83% when the training and testing data is considered to be 70% and 30% respectively. The proposed method provides better performance in less execution time to the existing techniques but still the accuracy can be improved further.

5.2 Future Scope

Nowadays, the primary issue in image forensics seems to be that falsification is introduced to an image with a mixture of different processes. Thus, while evaluating the traces, only one sort of attack would be revealed that would be inaccurate as image would not be tested further. Together with this technique, it is necessary to integrate different processes to uncover forgery. In this work, the scope of conceivable outcomes that CNN possibly empowers are not fully investigated. ELA reveals some of the hidden features but is certainly not a medium that can stand for itself and does not generate comprehensive forensic

outcomes for an unexperienced user. Although some recent methods have gained more accuracy but our method is more reliable in terms of training with fewer epochs and learnable parameters. This work has various opportunities for development in the future. The method can be tested with various attacks such as blurring, noise, etc. Moreover, the performance can also be improved by adding any constraint or by modifying the kernels or filters used in CNN. For further investigation, small neural networks with less number of features can be used to make systems faster and reliable. The use of different preprocessing techniques with CNN or other neural networks to improve the performance of image forgery detection can be used in the future.

REFERENCES

[1] Amerini I *et al.* (2017). Localization of JPEG double compression through multi-domain convolutional neural networks, In *Conference on computer vision and pattern recognition workshops*, IEEE, [Honolulu, Hawaii, United States: 2017], pp. 1865-1871.

[2] Barni M and Costanzo A (2012). A fuzzy approach to deal with uncertainty in image forensics, *Signal Processing: Image Communication*, *27*(9), 998-1010.

[3] Barni M *et al.* (2017). Aligned and non-aligned double JPEG detection using convolutional neural networks, *Journal of Visual Communication and Image Representation*, *49*, 153-163.

[4] Birajdar GK and Mankar VH (2013). Digital image forgery detection using passive techniques: A survey, *Digital investigation*, *10*(3), 226-245.

[5] Bo X *et al.* (2010). Image copy-move forgery detection based on SURF, In *International Conference on Multimedia Information Networking and Security*, IEEE, [2nd: United States: 2010], pp. 889-892.

[6] Bondi L *et al.* (2016). First steps toward camera model identification with convolutional neural networks, *IEEE Signal Processing Letters*, *24*(3), 259-263.

[7] Browne M and Ghidary SS (2003). Convolutional neural networks for image processing: an application in robot vision, In *Australasian Joint Conference on Artificial Intelligence*, Springer, [16th: Perth, Australia: 2003], pp. 641-652.

[8] Canny J (1987). A computational approach to edge detection, *Readings in computer vision*. Morgan Kaufmann, 184-203.

[9] Chen C, McCloskey S and Yu J (2017). Image splicing detection via camera response function analysis, In *Proceedings of the conference on computer vision and pattern recognition*, IEEE, [Hawaii, United States: 2017], pp. 5087-5096.

[10] Chen J *et al.* (2015). Median filtering forensics based on convolutional neural networks, *IEEE Signal Processing Letters*, *22*(11), 1849-1853.

[11] Chen M *et al.* (2008). Determining image origin and integrity using sensor noise, *IEEE Transactions on information forensics and security*, *3*(1), 74-90.

[12] Chen W, Shi YQ and Su W (2007). Image splicing detection using 2D phase congruency and statistical moments of characteristic function, In *Security, Steganography, and Watermarking of Multimedia Contents IX*, International Society for Optics and Photonics. [Vol. 6505: CA, United States: 2007], pp. 1-8.

[13] Christlein V *et al.* (2012). An evaluation of popular copy-move forgery detection approaches, *IEEE Transactions on information forensics and security*, *7*(6), 1841-1854.

[14] Cozzolino D and Verdoliva L (2016). Single-image splicing localization through autoencoder-based anomaly detection, In *International Workshop on Information Forensics and Security*, IEEE, [8th; Abu Dhabi, UAE: 2016], pp. 1-6.

[15] Cozzolino D, Poggi G and Verdoliva L (2017). Recasting residual-based local descriptors as convolutional neural networks: an application to image forgery detection, In *Proceedings of the ACM Workshop on Information Hiding and Multimedia Security*, ACM, [5th; Philadelphia, Pennsylvania, USA: 2017], pp. 159-164.

[16] CS231n Convolutional Neural network for Visual Recognition. Available at http://cs231n.github.io/neural-networks-2/ (Accessed on 19th March 2019).

[17] Dahl GE, Sainath TN and Hinton GE (2013). Improving deep neural networks for LVCSR using rectified linear units and dropout, In *international conference on acoustics, speech and signal processing*, IEEE, [Vancouver, British Columbia: 2013], pp. 8609-8613

[18] Dixit R and Naskar R (2017). Review, analysis and parameterisation of techniques for copy–move forgery detection in digital images, *IET Image Processing*, *11*(9), 746-759.

[19] Dong J *et al.* (2008). Run-length and edge statistics based approach for image splicing detection, *International workshop on digital watermarking*, Springer, [7th: Busan, Korea: 2008], pp. 76-87.

[20] Dong J, Wang W and Tan T (2013). Casia image tampering detection evaluation database, In *China Summit and International Conference on Signal and Information Processing*, IEEE, [1st; Beijing, China: 2013], pp. 422-426.

[21] Dumoulin V and Visin F (2016). A guide to convolution arithmetic for deep learning, *arXiv preprint arXiv:1603.07285*.

[22] Farid H (2009). Image forgery detection, *IEEE Signal processing magazine*, 26.2: 16-25.

[23] Fridrich AJ, Soukal BD and Lukáš AJ (2003). Detection of copy-move forgery in digital images, In *in Proceedings of Digital Forensic Research Workshop*, [Version 1: Cleveland, Ohio, 2003].

[24] Griebenow R (2017). Image Splicing Detection, *Machine Learning in Computer Vision and Natural Language Processing*, 24-30.

[25] Hsiao DY and Pei SC (2005). Detecting digital tampering by blur estimation, In *International Workshop on Systematic Approaches to Digital Forensic Engineering*, IEEE, [1st: Washington, DC, USA: 2005], pp. 264-278.

[26] Hsu YF and Chang SF (2010). Camera response functions for image forensics: an automatic algorithm for splicing detection, *IEEE Transactions on Information Forensics and Security*, *5*(4), 816-825.

[27] Huang H, Guo W and Zhang Y (2008). Detection of copy-move forgery in digital images using SIFT algorithm, In *Pacific-Asia Workshop on Computational Intelligence and Industrial Application*, IEEE, [Vol. 2: Wuhan, China: 2008], pp. 272-276.

[28] Ioffe S and Szegedy C (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift, *arXiv preprint arXiv:1502.03167*.

[29] Jaiswal AK and Srivastava R (2019). Image Splicing Detection using Deep Residual Network. *Available at SSRN 3351072*.

[30] Johnson MK and Farid H (2007). Exposing digital forgeries in complex lighting environments, *IEEE Transactions on Information Forensics and Security*, *2*(3), 450-461.

[31] Juan L and Gwon L (2007). A comparison of sift, pca-sift and surf, *International Journal of Signal Processing, Image Processing and Pattern Recognition*, *8*(3), 169-176.

[32] Kee E and Farid H (2010). Exposing digital forgeries from 3-D lighting environments, In *International Workshop on Information Forensics and Security*, IEEE, [2nd: Seattle, WA, USA: 2010], pp. 1-6.

[33] Korus P and Huang J (2016). Multi-scale analysis strategies in PRNU-based tampering localization, *IEEE Transactions on Information Forensics and Security*, *12*(4), 809-824.

[34] Korus P and Huang J (2016). Multi-scale fusion for improved localization of malicious tampering in digital images, *IEEE Transactions on Image Processing*, *25*(3), 1312-1326.

[35] LeCun Y, Bengio Y and Hinton G (2015). Deep learning, *nature*, *521*(7553), 436-444.

[36] Li J *et al.* (2014). Segmentation-based image copy-move forgery detection scheme, *IEEE Transactions on Information Forensics and Security*, *10*(3), 507-518.

[37] Li Q *et al.* (2014). Medical image classification with convolutional neural network, *International Conference on Control Automation Robotics and Vision (ICARCV)*, IEEE, [13th: Marina Bay Sands, Singapore: 2014], pp. 844-848.

[38] Lin HJ, Wang CW and Kao YT (2009). Fast copy-move forgery detection, *WSEAS Transactions on Signal Processing*, *5*(5), 188-197.

[39] Lin X (2018). Image Forgery Detection, In *Introductory Computer Forensics*, Springer Nature Switzerland AG, Springer, Cham, pp. 507-555.

[40] Lin Z *et al.* (2005). Detecting doctored images using camera response normality and consistency, In *Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, [San Diego, CA, USA: 2005] pp. 1087-1092.

[41] Liu Y *et al.* (2018). Image forgery localization based on multi-scale convolutional neural networks, *Proceedings of the ACM Workshop on Information Hiding and Multimedia Security*. ACM, [6th; Innsbruck, Austria: 2018], pp. 85-90.

[42] Majumder MTH and Al Islam AA (2018). A Tale of a Deep Learning Approach to Image Forgery Detection, In *International Conference on Networking, Systems and Security*, IEEE, [5th; Dhaka, Bangladesh: 2018], pp. 1-9.

[43] Muhammad G *et al.* (2014). Image forgery detection using steerable pyramid transform and local binary pattern, *Machine Vision and Applications*, *25*(4), 985-995.

[44] Ng TT, Chang SF and Sun Q (2004). Blind detection of photomontage using higher order statistics, In *International Symposium on Circuits and Systems*, IEEE, [Vol. 5: Vancouver, British Columbia: 2004], pp. 688-691.

[45] Pan X and Lyu S (2010). Region duplication detection using image feature matching, *IEEE Transactions on Information Forensics and Security*, *5*(4), 857-867.

[46] Park TH *et al*. (2016). Image splicing detection based on inter-scale 2D joint characteristic function moments in wavelet domain, *EURASIP Journal on Image and Video Processing, 2016*(1), 30.

[47] Photo tampering through history. Available at http://pth.izitru.com/1860_13_00.html/ (Accessed on 10th June 2019).

[48] Piva A (2013). An overview on image forensics, *ISRN Signal Processing*, 2013 (496701), 1-22.

[49] Pomari T *et al*. (2018). Image splicing detection through illumination inconsistencies and deep learning, In *International Conference on Image Processing*, IEEE, [25th: Athens, Greece: 2018], pp. 3788-3792.

[50] Popescu AC and Farid H (2004). Statistical tools for digital forensics, In *international workshop on information hiding*, Springer, [6th: Toronto, ON, Canada: 2004], pp. 128-147.

[51] Popescu AC and Farid H (2005). Exposing digital forgeries in color filter array interpolated images, *IEEE Transactions on Signal Processing*, *53*(10), 3948-3959.

[52] Rao Y and Ni J (2016). A deep learning approach to detection of splicing and copy-move forgeries in images, In *International Workshop on Information Forensics and Security*, IEEE, [8th; Abu Dhabi, UAE: 2016], pp. 1-6.

[53] Salloum R, Ren Y and Kuo CCJ (2018). Image splicing localization using a multi-task fully convolutional network (MFCN), *Journal of Visual Communication and Image Representation*, *51*, 201-209.

[54] Sonka M, Vaclav H and Roger B (1993). Image pre-processing, *Image Processing, Analysis and Machine Vision*. Springer, Boston, MA, 56-111.

[55] Toldo R and Fusiello A (2008). Robust multiple structures estimation with j-linkage, In *European conference on computer vision*, Springer, [10th: Marseille, France: 2008], pp. 537-547.

[56] Walia S and Kumar K (2018). Pragmatical investigation of frequency-domain and spatial-structure based image forgery detection methods, *International Journal of Computational Intelligence and IoT*, *1*(2).

[57] Wang J *et al.* (2009). Fast and robust forensics for image region-duplication forgery, *Acta Automatica Sinica*, *35*(12), 1488-1495.

[58] Wang Q and Zhang R (2016). Double JPEG compression forensics based on a convolutional neural network, *EURASIP Journal on Information Security*, *2016*(1), 23.

[59] Wang W, Dong J and Tan T (2010). Tampered region localization of digital color images based on JPEG compression noise, In *International Workshop on Digital Watermarking*, Springer,[9th: Seoul, Korea: 2010], pp. 120-133.

[60] Wu X and Fang Z (2011). Image splicing detection using illuminant color inconsistency, In *International Conference on Multimedia Information Networking and Security*, IEEE, [3rd: Shanghai, China: 2017], pp. 600-603.

[61] Xu B *et al.* (2015). Empirical evaluation of rectified activations in convolutional network, *arXiv preprint arXiv:1505.00853*.

[62] Yan Y, Ren W and Cao X (2018). Recolored image detection via a deep discriminative model, *IEEE Transactions on Information Forensics and Security*, *14*(1), 5-17.

[63] Zandi M, Mahmoudi-Aznaveh A and Talebpour A (2016). Iterative copy-move forgery detection based on a new interest point detector, *IEEE Transactions on Information Forensics and Security*, *11*(11), 2499-2512.

[64] Zeiler MD and Fergus R (2014). Visualizing and understanding convolutional networks, In *European conference on computer vision*, Springer, Cham, [13th: Zurich, Switzerland: 2014], pp. 818-833.

[65] Zhang Y *et al.* (2015). Image-splicing forgery detection based on local binary patterns of DCT coefficients, *Security and Communication Networks*, *8*(14), 2386-2395.

[66] Zhang Y *et al.* (2016). Image Region Forgery Detection: A Deep Learning Approach, *Proceedings of the Singapore Cyber-Security Conference*, [14th: Singapore: 2016], pp. 1-11.

[67] Zhao X *et al.* (2014). Passive image-splicing detection by a 2-D noncausal Markov model, *IEEE Transactions on Circuits and Systems for Video Technology*, *25*(2), 185-199.

[68] Zheng L, Zhang Y and Thing VL (2019). A survey on image tampering and its detection in real-world photos, *Journal of Visual Communication and Image Representation*, *58*, 380-399.

[69] Zhou Z *et al.* (2016). Effective and efficient global context verification for image copy detection, *IEEE Transactions on Information Forensics and Security*, *12*(1), 48-63.