

FAKE IMAGE DETECTION USING MACHINE LEARNING

A Thesis

Submitted

In Partial Fulfillment for the Degree

MASTER OF TECHNOLOGY

In

Computer Science & Engineering

Submitted by:

Vaibhav Kishore

Enroll No: 1900102764

Under the Supervision of:

Dr. Mohammad Suaib

(Assistant Professor)



Department of Computer Science and Engineering

INTEGRAL UNIVERSITY, LUCKNOW U.P, INDIA

June 2022

CERTIFICATE

This is to certify that **Mr. Vaibhav Kishore** (Enroll No. 1900102764) has carried out the research work presented in the dissertation titled " **FAKE IMAGE DETECTION USING MACHINE LEARNING** " submitted for partial fulfillment for the award of the **Master of Technology in Computer Science and Engineering from Integral University, Lucknow** under my supervision.

It is also certified that:

- i. This dissertation embodies the original work of the candidate and has not been earlier submitted elsewhere for the award of any degree/diploma/certificate.
- ii. The candidate has worked under my supervision for the prescribed period.
- iii. The dissertation fulfills the requirements of the norms and standards prescribed by the University Grants Commission and Integral University, Lucknow, India.
- iv. No published work (figure, data, table etc) has been reproduced in the dissertation without express permission of the copyright owner(s).

Therefore, I deem this work fit and recommend for submission for the award of the aforesaid degree.

Dr. Mohammad Suaib
Supervisor
(Assistant Professor)
Department of CSE,
Integral University, Lucknow

Date:26/05/2022
Place: Lucknow

DECLARATION BY STUDENT

I, **Vaibhav Kishore**, hereby declare that the work presented herein is original work done by me and has not been published or submitted elsewhere for the requirement of a degree programme. Any literature date or work done by other and cited within this thesis has given due acknowledgement and listed in the reference section.

Vaibhav Kishore

Place: Integral University

Date: 26/05/2022

RECOMMENDATION

On the basis of the declaration submitted by “**Vaibhav Kishore**”, a student of M.Tech CSE (Evening), successful completion of Pre presentation on 09-08-2021 and the certificate issued by the supervisor **Dr. Mohammad Suaib** (Assistant Professor) Computer Science and Engineering Department, Integral University, the work entitled “**FAKE IMAGE DETECTION USING MACHINE LEARNING**”, submitted to department of CSE, in partial fulfillment of the requirement for award of the degree of Master of Technology in Computer Science & Engineering, is recommended for examination.

Program Coordinator Signature

Dr. Faiyaz Ahmad

Dept. of Computer Science & Engineering

Date: _

HOD Signature

Mrs Kavita Agrawal

Dept. of Computer Science & Engineering

Date: _

ACKNOWLEDGEMENT

I am highly grateful to the Head of Department of Computer Science and Engineering for giving me proper guidance and advice and facility for the successful completion of my dissertation.

It gives me a great pleasure to express my deep sense of gratitude and indebtedness to my guide **Dr. Mohammad Suaib, Assistant Professor, Department of Computer Science and Engineering**, for his valuable support and encouraging mentality throughout the project. I am highly obliged to him for providing me this opportunity to carry out the ideas and work during my project period and helping me to gain the successful completion of my Project.

I am also highly obliged to the Head of Department, **Mrs Kavita Agrawal (Associate Professor, Head of Department or Computer Science and Engineering)** and PG Program Coordinator **Dr. Faiyaz Ahamad, Assistant Professor, Department of Computer Science and Engineering**, for providing me all the facilities in all activities and for his support and valuable encouragement throughout my project.

My special thanks are going to all of the faculties for encouraging me constantly to work hard in this project. I pay my respect and love to my parents and all other my friends and supporting member for their help and encouragement throughout this course of project work.

COPYRIGHT TRANSFER CERTIFICATE

Title of the Dissertation: **FAKE IMAGE DETECTION USING MACHINE
LEARNING**

Candidate Name: **Vaibhav Kishore**

The undersigned hereby assigns to Integral University all rights under copyright that may exist in and for the above dissertation, authored by the undersigned and submitted to the University for the Award at the M.Tech degree.

The Candidate may reproduce or authorize others to reproduce material extracted verbatim from the dissertation or derivative of the dissertation for personal and/or publication purpose(s) provided that the source and the University's copyright notices are indicated.

Vaibhav Kishore

Table of Contents

Table of Contents	Page No
Title Page.....	i
CERTIFICATE.....	ii
DECLARATION BY STUDENT.....	iii
RECOMMENDATION.....	iv
ACKNOWLEDGEMENT.....	v
COPYRIGHT TRANSFER CERTIFICATE.....	vi
Title of the Dissertation: FAKE IMAGE DETECTION USING MACHINE LEARNING.....	vi
Table of Contents.....	vii
List of Tables.....	ix
List of Figures.....	x
List of abbreviations.....	xi
ABSTRACT.....	xii
CHAPTER 1 INTRODUCTION.....	1
CHAPTER 2 LITERATURE REVIEW.....	4
2.1 Research Motivation.....	5
2.2 Research Objectives.....	5
2.3 CNN.....	6
CHAPTER 3 PROPOSED METHODOLOGY.....	8
3.1 Workflow.....	9
3.2 Dataset Collection.....	10
3.3 Data Pre-processing.....	11
CHAPTER 4 Implementation Approach.....	13
4. Architecture of Proposed Models.....	14
4.1 18-Layered CNN Model.....	14
4.2 Architecture of Pre-trained models.....	17
4.3 DenseNet121.....	23
4.4 VGG-19.....	25
4.5 ResNet50.....	27
4.6 Deployment of Parameter.....	30

4.6.1 COPY-MOVE:	31
4.6.2 IMAGE-SPLICING:.....	32
4.6.3 IMAGE- RETOUCHING.....	32
4.7 PROPOSED METHOD	32
4.7.1 Metadata Analysis	32
4.7.2 Error Level Analysis.....	32
4.7.3 Convolutional Neural Network.....	33
4.7.4 Transfer Learning	37
4.7.5 VGG 16 Model	37
CHAPTER 5 RESULT	38
CHAPTER 6 CONCLUSION	44
FUTURE WORKS	46
REFERENCES.....	47
ANNEXURE	
ANNEXURE 1: Published Paper : FAKE IMAGE DETECTION USING MACHINE LEARNING	
2: Communicated Paper	
3: Plagirism Report	

List of Tables

Table No.	Name of Table	Page No
Table 1	Attributes of Pre-trained Models	30
Table 2	Training and Testing Accuracy and Loss of Proposed CNN Model	40

List of Figures

Figure No	Title	Page No
Figure 1	Block Diagram of the Proposed Method	09
Figure 2	Block Diagram of the Proposed Method	10
Figure 3	Training and Testing Data Percentage	11
Figure 4	Architecture of the Proposed CNN Model	16
Figure 5	Summary Table of the Proposed CNN Model	17
Figure 6	Grid Size Reduction of Inception V3	19
Figure 7	Generic Architecture of InceptionV3	20
Figure 8	Summary Table of VGG-16 model	21
Figure 9	Architecture of VGG-16	23
Figure 10	Architecture of Dense Net 121	24
Figure 11	Summary Table of VGG-19 model	26
Figure 12	Architecture of VGG-19	27
Figure 13	Summary Table of ResNet50 model	28
Figure 14	Architecture of ResNet50	28
Figure 15	Training accuracy of the tested pre-trained models	41
Figure16	Testing accuracy of the tested pre-trained models	42

List of abbreviations

Abbreviation	Name
CNN	Convolutional Neural Network
VGG	Visual Geometry Group
AUC	Area under the Curve
GIMP	GNU Image Manipulation Program

ABSTRACT

Image editing is now so widespread because to the availability of image processing tools like Adobe Photoshop or GIMP. Detecting such phone photos is unavoidable if image-based cybercrime is to be exposed. Because of its ubiquity, a photograph captured with a digital camera or smartphone is frequently saved in the JPEG format. The JPEG method works with 8x8 pixel picture grids that are compressed individually. Unmodified photos have a comparable amount of inaccuracy. Due to a comparable quantity of faults over the whole picture, each block should deteriorate at about the same pace during the resaving procedure. Error Level Analysis detects that the compression ratio of this false picture differs from that of the genuine image. Our paper's goal is to create a picture forensics programme that can detect any type of photo modification. The vertical and horizontal histograms of the error level analysis image were then used to determine the site of the alteration. The suggested method was able to recognize the changed picture while also displaying the specific position of the adjustments, according to the results.

This allows users to take photos, add digital photographic filters and upload pictures. There are many unwanted contents in Instagram's posts such as threats and forged images, which may cause problems to society and national security. This research aims to build a model that can be used to classify Instagram content (images) to detect any threats and forged images. The model was built using deep algorithms learning which is Convolutional Neural Network (CNN), Alexnet network and transfer learning using Alexnet. The results showed that the proposed Alexnet network offers more accurate detection of fake images compared to the other techniques with 97%. The results of this research will be helpful in monitoring and tracking in the shared images in social media for unusual content and forged images detection and to protect social media from electronic attacks and threats.

CHAPTER 1

INTRODUCTION

The usage of technology in today's world has exploded, and one of the most prevalent forms of communication is the use of photographs. Images are now widely utilized in newspapers, magazines, websites, and ads, and they convey a wealth of information. Because of their widespread use, people's confidence in pictures is growing every day. Picture forging is the act of modifying or manipulating an image by changing some information inside it, and Image Forgery Detection is the process of determining whether the image is authentic or not.

In today's world, an enormous number of individuals have been victims of picture fraud. Many individuals modify photographs with image manipulation software and use them as evidence to deceive the court or numerous other people on social networking sites or applications. As a result, every image uploaded on social media should be assessed and classified as either authentic or fraudulent. Social media is one of the finest tools for socializing, sharing, and spreading knowledge, but it can also mislead individuals, causing mayhem due to unintended misleading propaganda.

This paper will then break down into three suggested methodologies for evaluating the original ideas of an image, with the first section focusing on metadata analysis, the second on image error level analysis, and the third section focusing on developing a machine learning model to evaluate the image.

It is a fact that social media have changed the way people interact and carry on with their everyday lives. Social networking sites are a prominent media phenomenon nowadays, and have attracted a large number of people. Worldwide, the number of users [1] now exceeds three billion. In the Gulf region, growth in the number of active users has exceeded 66% [2]. Saudi Arabia ranks seventh in the world in terms of social media use; more than 75% of its

estimated 25 million people [3] are active users of social media. Social media are based on specific foundations that bring people together and empower them to express themselves, share their interests and ideas, and forge new friendships with others who share their interests. Facebook, Twitter, and Instagram are among the most popular social networking sites of the day. It is a widespread practice to share images online through social networking services such as Instagram. At least 80 million images [4] are currently shared via Instagram every day. Instagram enables users to take photographs, apply digital photographic filters, and upload the pictures to website for social networking together with short captions. People upload and share billions of pictures [5] every day on social media.

A huge number of people have become victims of photo forgery in this technological age. Some criminals use software to exploit and use pictures as evidence to confuse the courts of justice [17]. To put an end to this, all photographs exchanged via social media should be labeled as true or fake. Social media is a great platform for knowledge sharing and dissemination. Yet If there is no caution, people may be fooled and even induced by unintended false propaganda. Though most image editing using Photoshop is clearly evident, some of these images may indeed appear really due to pixelization and shoddy jobs by novices [16]. In particular, in the Policy arena, edited images can break the credibility of a politician. In this research using machine learning algorithms [6, 7], the researcher will attempt to propose a classifier model via a convolutional neural network (CNN) that is capable of take advantage of knowledge to take an image from social media and then classify and detect it.

CHAPTER 2

LITERATURE REVIEW

2.1 Research Motivation

Deepfake is a novel idea that has gained a lot of traction in recent years for creating ultrarealistic photographs with programmes like FaceApp [52], Photoshop, and others. Bogus news or fake information travels quickly on social media, posing a threat to an individual, society, political system, and, in some cases, a nation. Deepfake, on the other hand, offers a serious danger. Even more so than typical false news, because they are difficult to see, individuals are more inclined to believe the phoney to be true. To avoid such falsification, we developed a system that identifies deepfake photos using machine learning and convolutional neural networks networks, allowing us to distinguish between real and fake images, which would be difficult otherwise.

2.2 Research Objectives

To identify deep fake pictures, our study employs a Convolutional Neural Network (CNN) and the Transfer Learning approach. CNN is one of the Neural Network's core components, and it employs image detection and classification to detect objects and recognize real-life faces. Furthermore, we will be able to establish whether or not the photographs have been tampered with. Many individuals have already worked on this topic, and the false image processing approach based on CNN has been included into their work. As a result, determining whether the photographs in the data set are phoney or real is much easier. Making phoney photographs has been used to commit a variety of crimes in recent years.

The following are types of forensic facilities:

- 1. Image quality metrics:** They look at the difference between both the doctored image and the original image. If the actual picture is not accessible, a hazy rendition of the image is used to imitate the test. [2,5]

2. Higher - level wavelet statistics: These statistics are produced from the image's multi-level decomposition.

3. Binary similarities measurements: These measurements capture the texture and correlation within both the Bit planes of lesser relevance, which are more vulnerable to manipulation.

First, single tools are designed to determine the essential image-processing functions in order to affect the identification of doctorate effects. Then, these individual "weak" detectors assembled together to determine the presence of a doctorate in an expert fusion scheme.

4. Enhance the meet the individual needs contrast picture: Contrast enhancement can be used to disguise the visual proof of image manipulation. Evidence of cut-and-paste forgeries may be discovered if these operations are tracked down. Cut-and-paste forgeries can be detected with the use of forensic tools.

5. Detecting histogram equalization in images: The Histogram Equalization Operation, like other contrast enhancement operations, creates spontaneous peaks and gaps in the picture histogram. The methods for detecting picture histogram equalization have been improved.

2.3 CNN

A frequently used deep learning-based model for distinguishing between Deepfakes and real ones is the Convolutional Neural Network. Deep Fake ensemble learning technique built by comparing multiple state-of-the-art Deep Learning-based models, in their work. These level-0 models' predictions are incorporated into this model. The level-1 model is trained using out-of-sample data predictions supplied by base models. InceptionResNetV2,

InceptionV3, ResNet101, DenseNet121, MobileNet, and DenseNet169 are used as base-learners, while a newly developed CNN model, also known as Deepfake Classifier, is used as a classifier. In the experiment, the Deepfake Stack achieved an accuracy of 98.24% and an AUROC of 1.0 using these models.

CHAPTER 3

PROPOSED METHODOLOGY

3.1 Workflow

We utilized a proprietary 18-layer Deep Convolutional Neural Network Model and Transfer Learning technique to compare and acquire the best result for detecting whether a picture is real or false. The following models were utilized in the experiments: InceptionV3, VGG-19, VGG-16, DenseNet121, and Resnet50. Our project comprises of a total of two classes, which are referred to as Real and Fake, respectively. On the basis of Training Accuracy [2], Testing Accuracy [1], Recall [4], Precision [4], and AUC Performance [34], we tested our suggested model. Our project's plan is broken down into seven phases. The Workflow Graph, on the other hand, is shown in Figure 1.

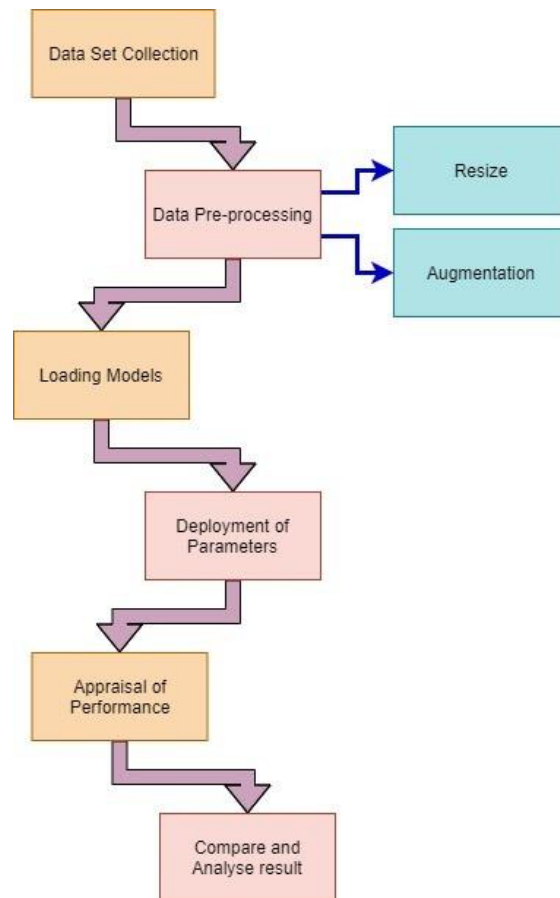


Figure 1: Block Diagram of the Proposed Method

3.2 Dataset Collection

We generated a dataset that includes both Real and DeepFake photos. The genuine photos were obtained from the kaggle dataset "140k Real and False Faces" [43], and the fake images were obtained from the kaggle dataset "DeepFakes" [48], where the Deep fake images were created using MTCNN [37]. In order to generate our dataset, we used a total of 13000 photos. These photographs are divided into two categories, Real and Fake, with 3000 images collected for testing and 10,000 images collected for training. Every picture was in RGB format [9], with a 92-pixel resolution for false photos and a 256-pixel resolution for genuine photographs that were processed afterwards.

Figure 2 shows a few photos from our dataset, while Figure 3 shows the dataset partition.

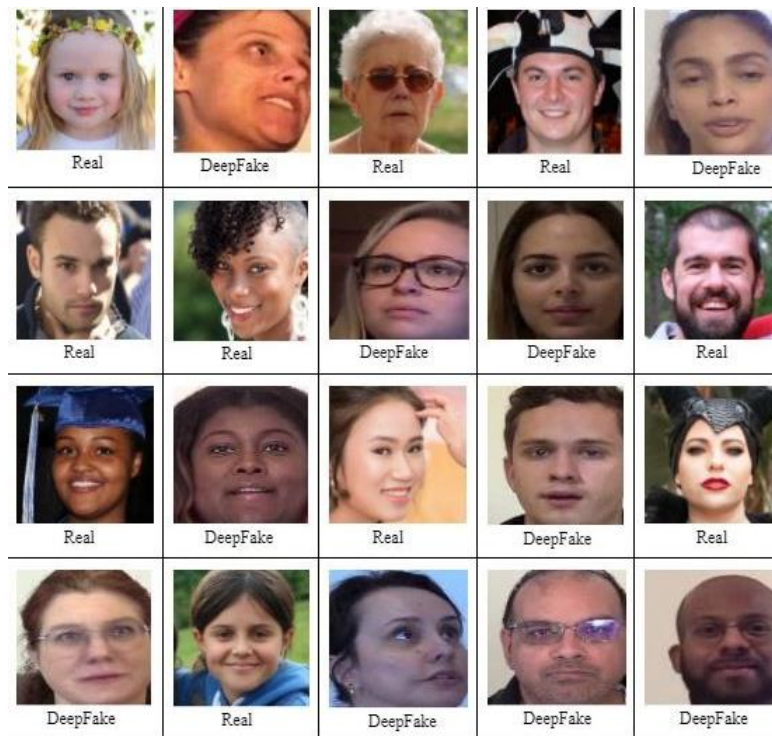


Figure 2: Deepfake and Real Images from the Used Dataset

The proportion of training and testing data in the dataset is shown in Figure 3.

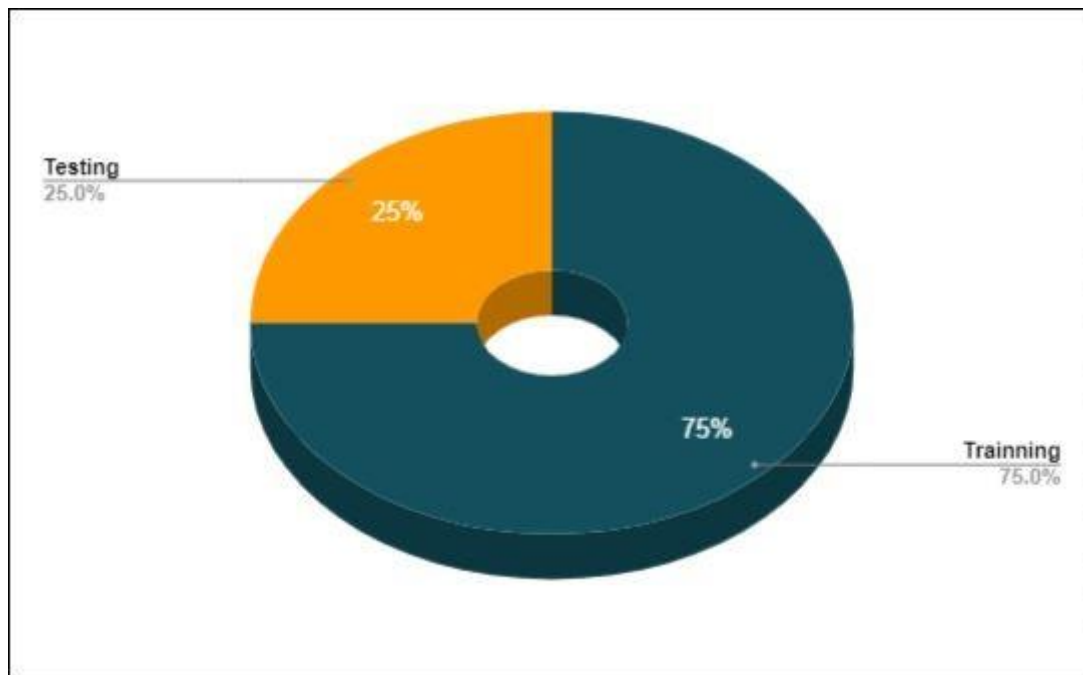


Figure 3: Training and Testing Data Percentage

Apart from the training and testing data, we generated classification reports and confusion matrices using 300 unseen photos divided into two classes: Real and Fake.

3.3 Data Pre-processing

Since information retrieval during the training phase is more challenging, data pre-processing aims to minimize inconsistent data as well as distorted and inadequate input. The final training set is the outcome of data pre-processing. Filtering, normalization, modification, extraction of features [25], identification etc. are some of the techniques that are used generally [28]. Here for our work purpose we have chosen: i. Resizing and ii. Augmentation of the raw data inputs. An unprocessed dataset usage

might often result in disappointing outcomes. In this model we have processed our raw data in mainly two ways, which are: Resizing the and Augmentation of the raw images. Detailed descriptions are given below:

Resize the Data: For the transfer learning technique, the photographs were scaled down to a standard resolution of 224x224 pixels because our system requires a consistent dimension of the data. The quantity of shrinkage eventually decreases as a result of this. We scaled each RGB image into a consistent resolution of 256x256 pixels for the 18-layer CNN model since CNN models may use a fixed dimension as input.

Augmenting Images: By augmenting our data, we were able to increase the quantity of training data available. Production. Different augmentation tactics are more or less useful in different conditions. These have a significant impact on the training set and should not be included during the evaluation process. For our data, we employed a variety of picture augmentation techniques. For both methods, we employed the following techniques: rescale, shear, zoom, rotation, width shift, height shift, brightness, vertical flip, and horizontal flip.

CHAPTER 4

IMPLEMENTATION APPROACH

4. Architecture of Proposed Models

4.1 18-Layered CNN Model

To differentiate between actual and deep fake photos, we used the provided approach to build a multi-layered deep CNN model. Fully connected layer (FC) [16], Convolutional layer [23], and Pooling layer [28] are the three core layers of a CNN model. We employed two extra layers in addition to these three: the Activation layer and the Dropout layer. The next sections go through the specifics of each layer.

1. Convolutional Layer: The very first layer uses convolution to extract information from the input image. Through learning visual attributes with tiny portions of input data, the convolution keeps the connection among the pixels [47]. It is a mathematical system having 2 inputs: a filter or kernel and an image matrix. Convolution of an image with several filters may be used to accomplish tasks such as edge detection, blurring, and sharpening. For this CNN model we have used the Conv2D layer in the convolutional layer. In total we have used six Conv 2D layers to create the model.

2. Fully Connected Layer (FC): The Fully Connected layer or FC layer, one can flatten the matrix into a vector and send it to a Fully Connected layer, similar to a Neural Network. This layer accumulates the neurons, weights and biases.

3. Pooling Layer: Some of the images from the used datasets might be too extensive, the portion of pooling layers will degrade the count of parameters of such images. Subsampling or down sampling also known as, Spatial pooling, will decrease

the dimensionality of each map while preserving crucial data. For each of the convolutional layers we have used one MaxPooling2D layer. This means it adds six layers in total with the other layers.

- Flatten layer: One flatten layer is used after the sixth Max Pooling layer. This overall helps to flatten the entire network.
- Dense layer: A total number of three dense layers have been used in this model after the flatten layer. This layer basically feeds all the neurons with the outputs from previous layers.
- Batch Normalization layer: After every Conv2D layer a batch normalization layer is added. Apart from that two of the layers are added after the first two dense layers.
- Activation layer: In this model, Sigmoid function is used as the activation layer. The following equation shows the activation Sigmoid function that has been used in the Dense layer.
- Dropout layer: For every iteration during the training phase, this layer converts the inputs to zero with a frequency of rate, which helps to avoid over fitting.

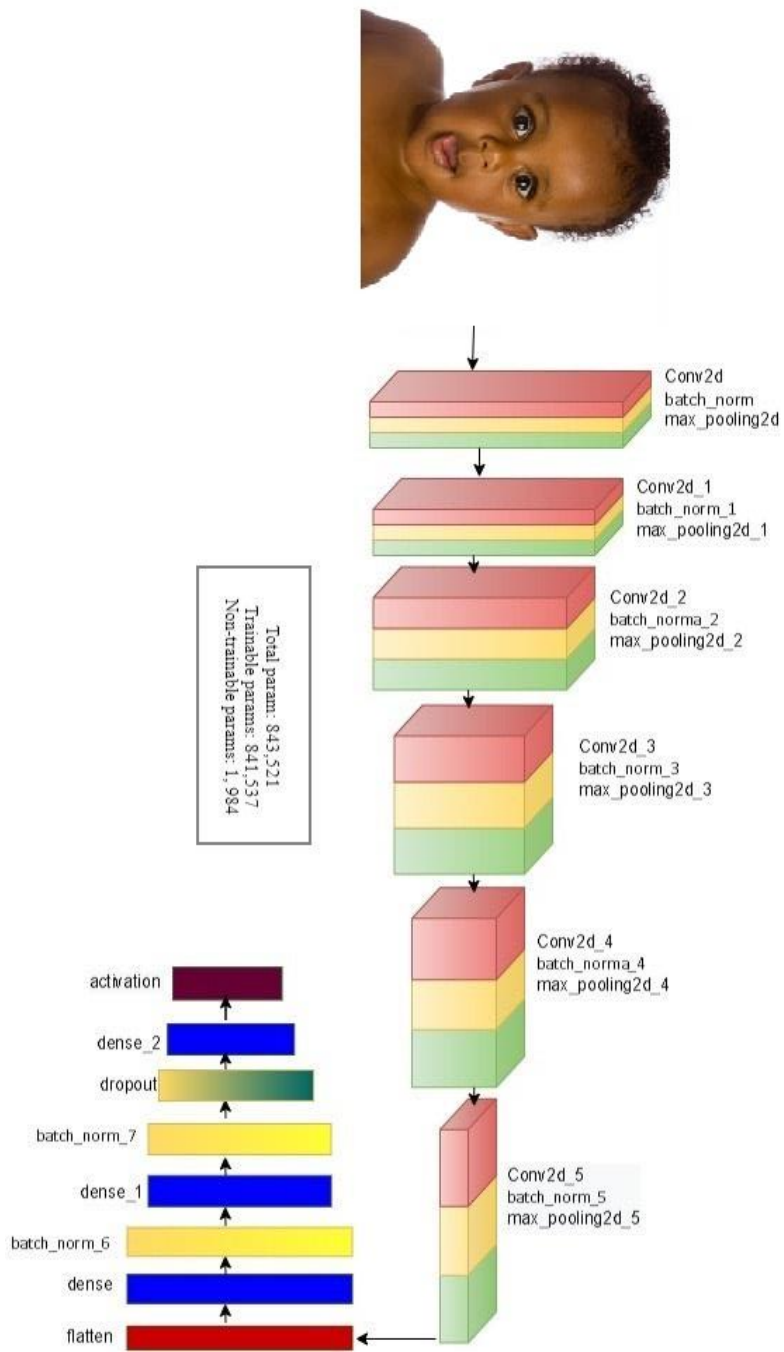


Figure 4: Architecture of the Proposed CNN Model

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 254, 254, 32)	896
batch_normalization (BatchNo	(None, 254, 254, 32)	128
max_pooling2d (MaxPooling2D)	(None, 127, 127, 32)	0
conv2d_1 (Conv2D)	(None, 125, 125, 64)	18496
batch_normalization_1 (Batch	(None, 125, 125, 64)	256
max_pooling2d_1 (MaxPooling2	(None, 62, 62, 64)	0
conv2d_2 (Conv2D)	(None, 60, 60, 128)	73856
batch_normalization_2 (Batch	(None, 60, 60, 128)	512
max_pooling2d_2 (MaxPooling2	(None, 30, 30, 128)	0
conv2d_3 (Conv2D)	(None, 28, 28, 256)	295168
batch_normalization_3 (Batch	(None, 28, 28, 256)	1024
max_pooling2d_3 (MaxPooling2	(None, 14, 14, 256)	0
conv2d_4 (Conv2D)	(None, 12, 12, 128)	295040
batch_normalization_4 (Batch	(None, 12, 12, 128)	512
max_pooling2d_4 (MaxPooling2	(None, 6, 6, 128)	0
conv2d_5 (Conv2D)	(None, 4, 4, 64)	73792
batch_normalization_5 (Batch	(None, 4, 4, 64)	256
max_pooling2d_5 (MaxPooling2	(None, 2, 2, 64)	0
flatten (Flatten)	(None, 256)	0
dense (Dense)	(None, 256)	65792
batch_normalization_6 (Batch	(None, 256)	1024
dense_1 (Dense)	(None, 64)	16448
batch_normalization_7 (Batch	(None, 64)	256
dropout (Dropout)	(None, 64)	0
dense_2 (Dense)	(None, 1)	65
activation (Activation)	(None, 1)	0
=====		
Total params: 843,521		
Trainable params: 841,537		
Non-trainable params: 1,984		

Figure 5: Summary Table of the Proposed CNN Model

4.2 Architecture of Pre-trained models

It is the neural network that has been used in previous scenarios and has gathered knowledge that can be applied to fresh targeted samples in pre-trained models. For the experiment, five pre-trained models were used: Inception V3, VGG-16, VGG19, DenseNet121, and ResNet50. The following are the specifications for each model:

- Inception V3: The architecture [27] of an InceptionV3 network is built one step at a time, as detailed below:

1. **Factorized Convolutions:** This reduces the amount of parameters in a network, making it more efficient to compute. It also keeps track of the network's efficiency.

2. **Smaller Convolutions:** Training is substantially faster when smaller convolution operations are used instead of larger convolution operations. A 5*5 filter, for example, has 25 characteristics; in its stead, two 3*3 filters have 25 properties.

There are only 18 properties in convolution (33% + 33% + 33% + 33% + 33% + 33% + 33% + 33% + 33% + 33%)

3. **Asymmetric Convolutions:** A 1*3 convolution preceded by a 3*1 convolution might be utilized instead of a 3*3 convolution. If a 3*3 convolution was replaced with a 2*2 convolution, the number of parameters would be somewhat higher than the asymmetric convolution stated.

4. **Auxiliary Classifier:** An auxiliary classifier is employed as a tiny CNN implanted between layers during training, and the loss is included into the actual network loss. In InceptionV3, the auxiliary classifier acts as a regularizer.

5. **Grid Size Reduction:** To lower grid size, pooling strategies are widely utilised. In Figure 5, a more effective way to overcoming computational cost constraints is presented:

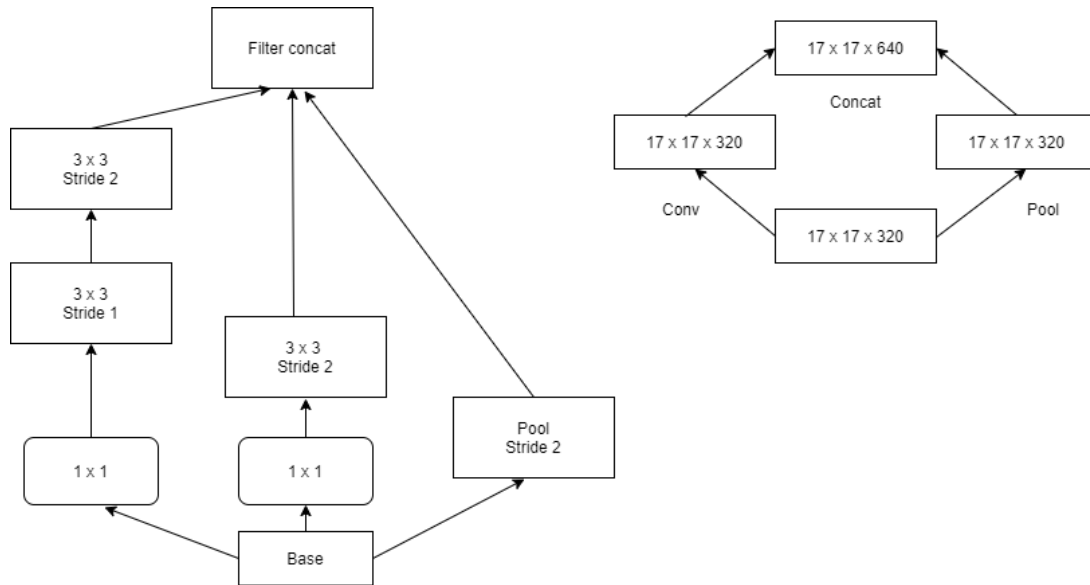


Figure 6: Grid Size Reduction of Inception V3

The generic architecture of InceptionV3 is given below in figure 6 :

VGG-16: A 224 by 224 RGB image with a predetermined size is used as an input the convolutional-1 layer. The picture is processed by a collection of convolutional layers with a very small field: 3*3 the size of the image in order to keep the sense of right, down, left, up, and center. It also utilizes 1*1 convolution filters in one of the

Settings, which may be thought of as a linear modification of the input channels. For

3 x 3 convolutional layers, the spatial padding of convolutional layer input is set to one pixel, and the stride of convolution is set to one pixel, so that the spatial resolution is preserved after convolution.

Spatial pooling is done via five max-pooling layers that come before the convolutional layers. Max-pooling is achieved by stride 2 throughout a 2*2 pixel frame. Three fully-connected layers are inserted after a sequence of convolutional layers of variable depth

in various designs, with the first two having 4096 channels each and the third doing classification and hence having 1,000 channels [31].

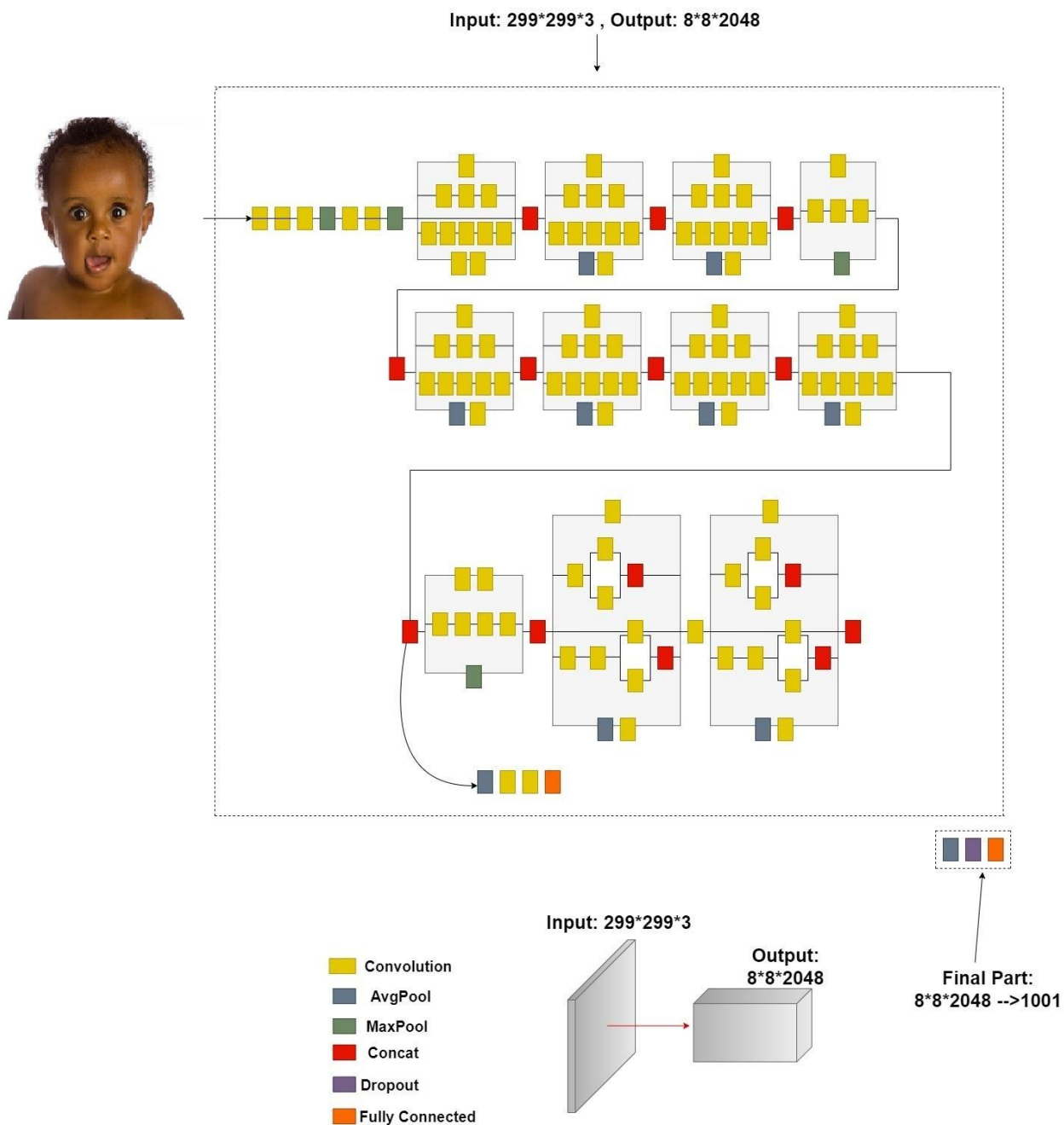


Figure 7: Generic Architecture of InceptionV3

The final layer is the activation layer. The entirely linked levels are configured in the same way in all networks. The VGG-16 model is summarized in Figure 8.

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
Total params: 14,714,688		
Trainable params: 0		
Non-trainable params: 14,714,688		

Figure 8: Summary Table of VGG-16 model

The architecture of VGG-16 is given below in figure 9:

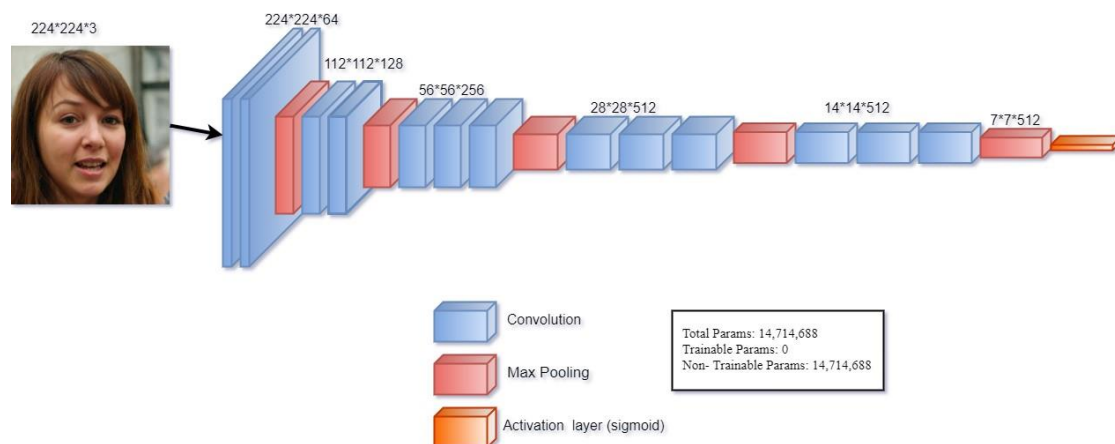


Figure 9: Architecture of VGG-16

4.3 DenseNet121

We've supplied this one (224 224) RGB pictures as input, and the network's needed parts are DenseBlocks [21], Growth Rate, Connectivity, and Bottleneck layers [36]. DenseNet-121 has 120 Convolutions and 4 AvgPool [19]. [59] A 1x1 kernel [10] serves as the bottleneck layer, while a 3x3 kernel handles the convolution process. Each transition layer also includes a 1*1 convolutional with 2 2 average pooling layer and a stride [13] of 2. As a consequence, a fundamental convolution layer with 64 (7X7) filters and a stride of 2 has been developed, as well as a maximum pooling of 3x3 with a stride of 2.

All of the network's feature mappings for classifications and distribution are included in the Global Average Pooling layer. All layers, even those in the same dense block and transition layers [3], disperse their parameters among many inputs, allowing hidden layers to exploit

data obtained earlier in the process [52]. Because transition layers create a lot of duplicated data, the output of the transition layers is given the least weight by the layers in the second and third dense blocks. DenseNet121's architecture is seen in Figure 10:

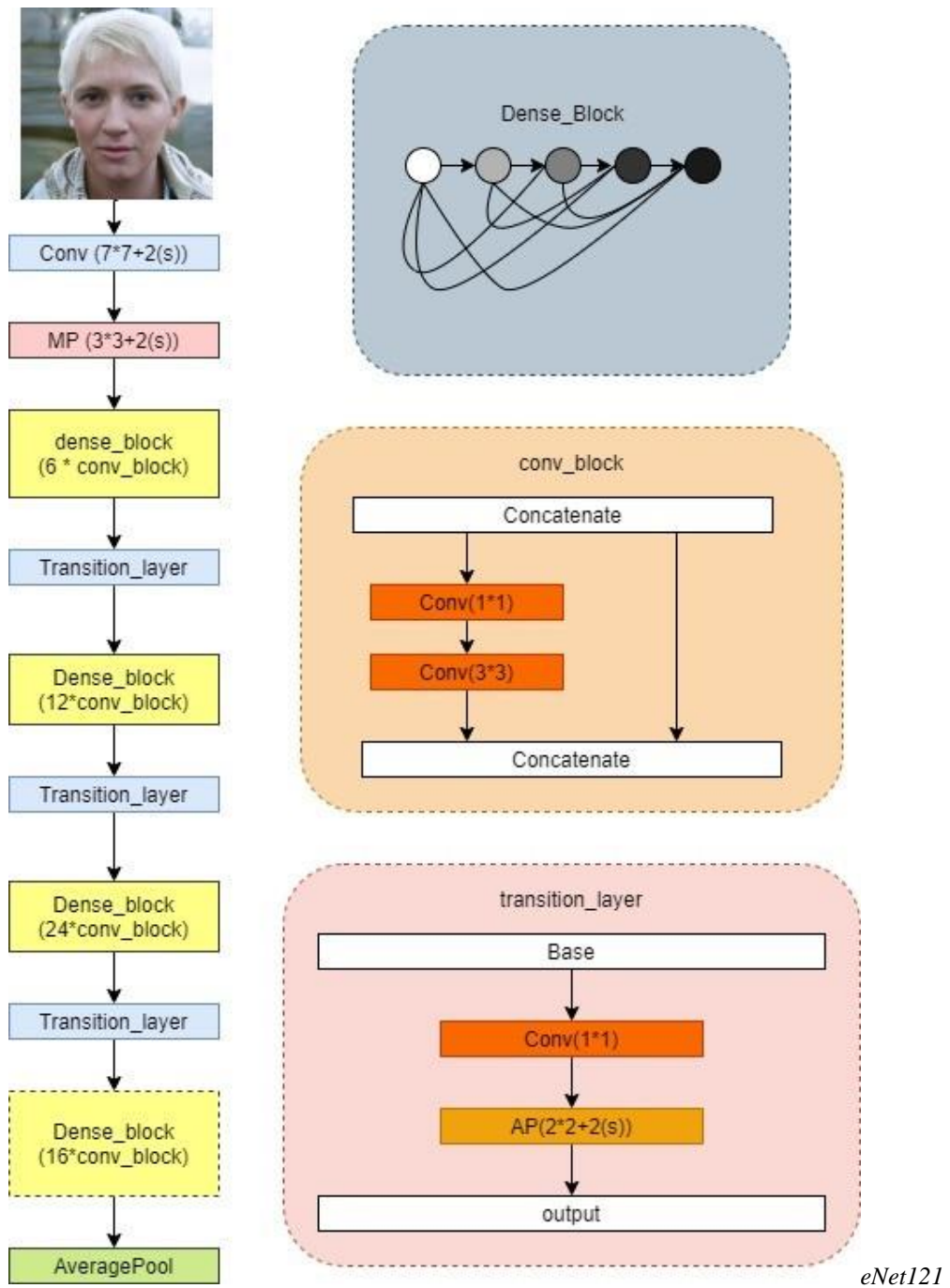


Figure 10: Architecture of Dens

4.4 VGG-19:

This network took an RGB image as input and gave it a specified size (224 224) [38], signifying a 224, 224, 3 matrix format. To eliminate the mean RGB value from every pixel, just one pre-processing was used, which was computed over the complete training dataset. Strides are made up of kernels with a size of (3 3) and a size of one pixel to cover the complete visual notion. Spatial padding was used to maintain the image's spatial resolution. Over a 2 2 pixel frame, max pool was attained with stride 2. Figure 11 shows the summary table for the aforementioned model:

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv4 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv4 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv4 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
=====		
Total params: 20,024,384		
Trainable params: 0		
Non-trainable params: 20,024,384		

Figure 11: Summary Table of VGG-19 model

The architecture of VGG-19 is given below in Figure 12 :

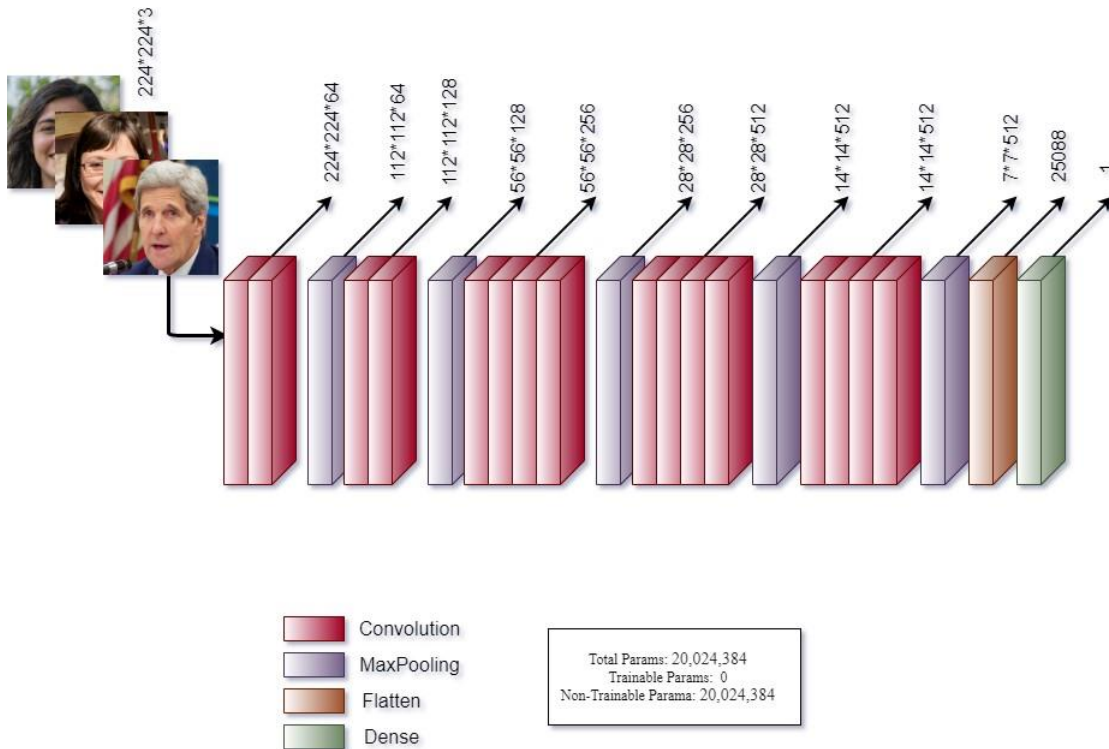


Figure 12: Architecture of VGG-19

4.5 ResNet50:

The following are the components of the ResNet 50 architecture: There is one convolutional layer with a kernel size of 7*7 and 64 distinct kernels, each with a stride size of 2. Then there's a maximum pool layer with a stride length of two. In the latter convolution, there is a 1164 sized kernel, followed by a 3364 sized kernel, and finally a 11256 large kernel. These three layers are done in triplicate, giving us a total of nine levels in this phase. It is followed by a kernel with the number 1128, then a kernel with the number 33128, and lastly a kernel with the number 1512. This method is repeated four times, giving us twelve layers in total. Then a 1×1512 kernel was added, followed by two additional 3×3512 and 1×12048 kernels, for a sum of 9 layers.

Layer (type)	Output Shape	Param #
resnet50 (Functional)	(None, 2048)	23587712
dense (Dense)	(None, 1)	2049

Total params: 23,589,761
Trainable params: 2,049
Non-trainable params: 23,587,712

Figure 13: Summary Table of ResNet50 model

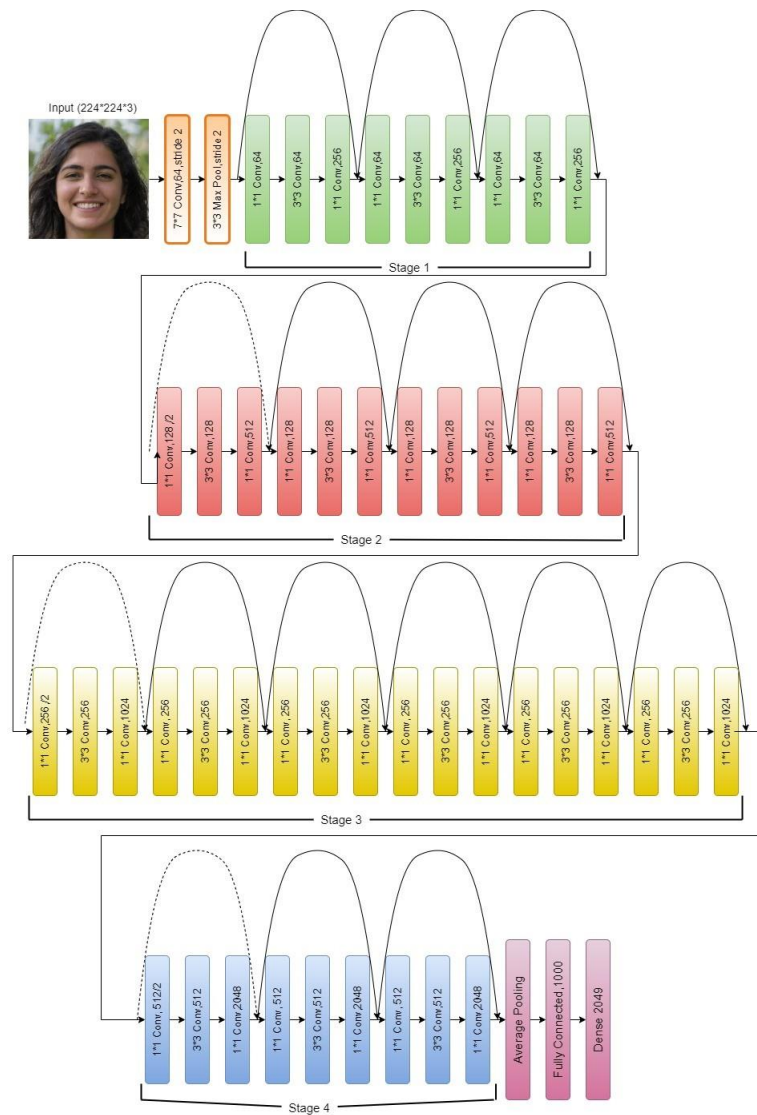


Figure 14: Architecture of ResNet50

Researchers are presently using pre-trained models, as shown in Table 4.1. The table shows how parameters and depth there are in the network and how much input picture size is required for training [51]. As a result of these models, DeepFake pictures are classified.

Table 1: Attributes of Pre-trained Models

Pre-Trained Model	Input Size	Depth	Network Size(MB)	Parameters
InceptionV3	299×299	159	89	23,851,784
VGG16	224×224	23	528	138,357,544
DenseNet121	224×224	121	33	8,062,504
VGG19	224×224	26	549	143,667,240
ResNet50	224×224	-	98	25,636,712

VGG16 and VGG19 are series-connected networks in this case, whereas the others are directed acyclic graphs [49]. Call the built-in Keras function and Transfer learning programmed in MATLAB, Google Colab, Jupyter Notebook, and other platforms that allow these technologies to access and utilize these models. For example, researchers utilized MATLAB to categories electronic devices in paper[20]. Each of the models has specified weights/features that make processing and recognizing layers simpler. Five pre-trained models are utilized to examine the data in this study. We want to test which model works best in terms of accuracy, loss, recall, precision, execution time, network size, and validity with Deep Fake photos to classify.

Enormous number of people has become victims of image forgery in our modern society. A lot of people use image manipulating software's to manipulate images and use it as evidence to mislead the court or several other people on social media sites or applications. This is why every image that is shared on the social media should be evaluated and generalized as either real or fake. Social media is one of the best platforms to socialize, share and spread knowledge but if no precautions are taken, it can mislead people resulting to cause havoc due to unintentional false propaganda. While it takes some practice to photoshop images and can clearly be observed due to pixelization and shady jobs by novices but some of them when manipulated by a professional can indeed appear genuine. Especially in the political aspect's images can be manipulated to make or break a politician's credibility.

4.6 Deployment of Parameter

Forensic techniques practiced these days to manipulate images require an expert to analyze the credibility of an image. This approach may be practical for a small number of images however it is not recommended to be used for evaluating a large number of images such as on a social media website. Therefore, we need to implement a system that can determine whether an image either real or fake with the help of current machine learning algorithms available to us and thereby make it available for use to the common public.

This paper will further unfold into three proposed methodologies that can be followed respectively to evaluate the originality of an image whereby first we will focus on the

metadata analysis, secondly, we will focus on the Error Level Analysis of the images and in the last part we will focus on generating a machine learning algorithm to evaluate the image.

Parameter	Pre- trained models (Value)	18-layered CNN Models (Value)
Training Data	75%	75%
Testing Data	25%	25%
Target Size	(224, 224)	(256,256)
Batch Size	32	32
Epoch	20	20
Step	313	313
Verbose	1	1
Initial Learning Rate	0.001	0.1
Minimum Learning Rate	-	0.000001
L1 L2 Regularization	0.001	0.001
Momentum	0.9	0.9
Execution Environment	GPU	GPU
Optimizer	Adam	Adam
Loss Function	Binary CrossEntropy	Binary CrossEntropy
Activation Function	Sigmoid	Sigmoid
Class Mode	Binary	Binary
Color Mode	RGB	RGB
Metrics	Accuracy, Loss, Precision, Recall	Accuracy, Loss, Precision, Recall
Callback	ReduceLROnPlateau	ReduceLROnPlateau

Table 5.1: Parameters Used for the Pre-trained Models and the 18-Layer CNN Model

4.6.1 COPY-MOVE:

This type of image tampering technique is used when a person needs to cover one part of the image in order to add or remove information i.e. done with the help of textured regions from the same image as they have similar color values, dynamic range and noise variation properties of that image.

4.6.2 IMAGE-SPLICING:

This type of image tampering technique is used when a person uses a part from one image and then pastes it on some another image without further postprocessing such as smoothing of boundaries between different fragments.

4.6.3 IMAGE- RETOUCHING

This type of image tampering technique is used when a person needs to alters an image to improve its appearance, retouching usually involves small localized adjustments to an image. Retouching an image can also be explained as polishing of an image by doing basic operations on it such as white balancing, cropping and adjusting other elements of an image.

4.7 PROPOSED METHOD

4.7.1 Metadata Analysis

Metadata analyser is basically a tag selecting and searching algorithm. If keywords like Photoshop, Gimp, Adobe etc. are found in the text and then the possibility of being tampered is increased. Two separate variables are maintained which are called fakeness and realness.

All of the variables represent the weight of being real or fake. Once a tag taken, is analyzed and corresponding variable is incremented by some predefined weight. These properties are already added in the photos by the cameras and the photo manipulation software's if they are used, but can easily be tampered or changed and cannot be trusted and should be only be used for preliminary analysis.

4.7.2 Error Level Analysis

Error Level Analysis resaves a particular image at a certain error rate, for example 96%,

then it checks for a virtual change, if any detected it means that the cells have reached its local minima for error at that quality level. Nevertheless, if many changes are discovered, then the pixels tend to be original. These analyses differentiate the real pixels from the fake one.

The system first saves an image at 100% quality and then it is resaved into the 90% quality image. The difference between these two is found out through difference method. The resultant image is the required error level analysis (ELA) image of the input image. Now, this image is saved as a buffered image and sent to the neural network for further processing.

4.7.3 Convolutional Neural Network

A multilayer perceptron neural network having an input layer and output layer both with a few hidden layers. When the image is opted for evaluation, the image is first converted for ELA representation from Compression and Error Level Analysis stage. The second step includes calculating the ELA since 90% of the images are used to construct ELA image. In the next step, the image is preprocessed and converted into 100x100px width and height. The image is serialized into an array containing approximate 30,000 integer values while representing 10,000 pixels. These pixels include red, green, and blue components; therefore, 10,000 pixels will have 30,000 values.

When the data is being trained, the array will be given as input to the neural network and output neurons also set. The 2 output neurons represent fake and real image. If the image is fake, then the neuron is set to one while if the neuron is real then it is set to zero. During testing, the image array will be fed into the input neurons and values of output neurons will be taken to display the result of the analysis.

- Due to the nonlinear nature of image data, the researcher will use a non-linear activation function called The Rectified Linear Unit (ReLU). The rectifier job is defined as the positive section of its argument.
- To reduce the size of the array in the precise step, we down sample it using an algorithm called max pooling to modify the output of the layer. Further pooling helps to make the representation almost invariant with respect to small translations of the input. Applying Anaconda software with the Machine Learning Toolbox helps you to train your own CNN from scratch or use a pretrained model to conduct transfer learning. The method you choose depends on the resource you have, the type of application you are creating and the purpose of the application. In order to train the network from scratch, the number of layers and filters must be determined and the other requirements adjusted. Training a specific model from the start also requires enormous amounts of data, based on millions of samples, which can take a long time. An appropriate alternative to CNN training from scratch is the use of a pre-training model to automatically extract properties from a new dataset. Known as transfer learning, this is an easy way to apply deep learning without a great data set and a long period spent on calculation and training.

A. Create Simple Machine Learning Networks for Classification

There are three networks: Alexnet, Classic CNN, and Alexnet using transfer learning. For each network, there are a training dataset and a test dataset. There are also cases in which the test data is from training data and vice versa. There are two datasets in our experiment. The first dataset contains 1400 images for training and 400 images for testing. The second dataset contains 400 images for training and 40 images for testing. In the second dataset, the fake images in the training data are extracted from the original images and, for each original image, three fake images were made. The researcher modified the original images via addition, deletion, changing colors. The dataset training steps are described as follow by

using CNN network.

1) **Load and analyze image data:** Load the data of the sample as a data store for the image. Image Datastore automatically labels images based on the name of the folder and stores the data as an object of the image datastore. An image datastore helps you to store large image information when training a convolution neural network and interpret image batches efficiently.

2) *Define the network architecture:* Determine the convolutional neural network architecture and create network layers.

3) **Define training options:** Defines the training options after defining the architecture of the network. Learning rate, number of epochs, momentum and batch size.

4) **Train the network:** Train the network using layer- defined architecture, training data & the training options.

5) **Predict the labels of new data and measure the classification accuracy**

Predict the labels of the data using the trained network, and measure the final accuracy.

Note that Alexnet network and Transfer Learning network follow the same training steps but with some additions where the load pretrained network is additional step in Alexnet network and the replace final layers is additional step in transfer learning.

Deployment of Parameter

VGG16 and VGG19 are series-connected networks in this case, whereas the others are directed acyclic graphs [49]. Call the built-in Keras function and Transfer learning programmer in MATLAB, Google Colab, Jupyter Notebook, and other platforms that allow these technologies to access and utilise these models. For example, researchers utilized MATLAB to categories electronic devices in paper[40]. Each of the models has specified

weights/features that make processing and recognizing layers simpler. Five pre-trained models are utilised to examine the data in this study. We want to test which model works best in terms of accuracy, loss, recall, precision, execution time, network size, and validity with DeepFake photos to classify.

- Equation for Accuracy:

$$\text{Accuracy} = \frac{\text{True Positive} + \frac{\text{True Negative}}{\text{Condition Positive}} + \text{Condition Positive} + \text{Condition Negative}}{\text{Condition Positive} + \text{Condition Negative}}$$

- Equation for Precision:

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \times \frac{\text{True Positive Rate}}{\text{True Positive}}$$

- Equation for recall:

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \times \frac{\text{True Positive Rate}}{\text{True Positive}}$$

- **AUC:** The Area under the Curve (AUC) is a statistic for how well a classifier can distinguish between classes [8]. It merely refers to the scale or degree of separability on which it is assessed. Because AUC is scale and classification threshold invariant, it provides an aggregated performance metric across all classification thresholds. The AUC rate shows how well the model can discriminate between positive and negative classes. A higher AUC rate suggests that a model is doing better.

Transfer Learning

Transfer learning transfers information from one domain to the required domain by improving the learner. It is a technique where a model is developed for one task but then reused as a starting point for another task. Let's take an example of two people and teach them drums. The

first person has never learned or played any kind of musical instrument before while the other has an extensive knowledge of playing xylophone. Hence, the person with the experience of xylophone will learn drums way more efficiently by transferring previously learnt knowledge and apply it in the current task. Transfer Learning is required when there is a narrow amount of target training data. The major reason for that maybe data being pricy and rare. But we need to use it because it decreases the training time for a model and gives a lower error level.

4.7.4 VGG 16 Model

VGG16 is a convolutional neural network architecture which focuses on having Convolution layer with stride 1 and same padding and maxpool layer of stride 2. VGG Network uses 3x3 convolutional layers putting right above each other. While the depth increases with each stack. It makes advanced changes over AlexNet since it replaces kernel sized filters to multiple 3x3 kernel sized filters.

CHAPTER 5

RESULT

The performance measures of the proposed methodology are discussed in detail. The major goal of this work is to detect both normal and fake images in an accurate manner. For this purpose, a convolution neural network is utilized in this work. This CNN comprises four layers: the convolution layer, pooling layer, activation layer, and Soft Max layer. Each layer performs a specific task individually. First, the input image is obtained from the image acquisition. Then the image is converted into non-overlapping patches, from these patches. Further, the values of the features are normalized and down sampled to obtain a reduced feature set. Finally, the probability of the output is determined to classify the given image as normal or fake. Here, the approach developed in this research is evaluated, based on performance metrics and relative with the current techniques.

From using transfer learning on the VGG16 model we were able to attain the following results. We can see from the graph below that in both Training and Validation the approach we used was significantly optimal as during both the training and validation in the graph it can be seen that our neural network was neither over-fitting or under-fitting, due to limited resources on our end we were still able to achieve a validation accuracy of 86.12%.

The quality of the technique proposed is tabulated and shown in the tables below. The accuracy of the results is shown to vary among networks. Alexnet is the most accurate, followed by Alexnet using TL and then Classic CNN. Table I illustrates comparison among three network types (Alexnet Network, Alexnet Using Transfer Learning, and Classic CNN) regarding the performance accuracy of results when the testing data is from outside the training data. The findings reveal differences among the three networks, in favor of Alexnet (93.4), followed by Alexnet using transfer learning (93.2) and, finally, classic CNN

(70.1). The findings also reveal differences among the three networks when testing data from the training data. These are again in favor of Alexnet (99.3), followed by Alexnet using transfer learning (94.0) and, finally, classic CNN (83.9). The results in the table below are specific to the first dataset.

Table II compares among the three network types (Alexnet Network, Alexnet Using Transfer Learning, and Classic CNN) regarding the performance accuracy of results when testing data from outside the training data. The findings reveal no statistically significant differences in the performance accuracy of results among the three network types. The value of significance level amounted to 0.172; this means it is greater than 0.05, which is not statistically significant. There were, however, differences amongst the three networks when testing data from the training data, in favor of Alexnet (91.1), followed by Alexnet Using Transfer Learning (78.4) and, finally, Classic CNN (64.5). The results in the table below are specific to the second dataset. We selected a total of 3000 images for testing, separated into two portions for real and fake classes, each comprising 1500 images. Testing data occupies 25% of the entire data, whereas Training data accounted for 75%. Finally, our proposed model was shown to be accurate to the tune of 98.77 percent. The results for testing and training accuracy and loss are shown in Table 2.

Table 2: Training and Testing Accuracy and Loss of Proposed CNN Model

Training Accuracy	Testing accuracy	Training Loss	Testing Loss
99.82%	98.77%	0.63%	5.87%

The table shows the suggested model provides 98.77% testing accuracy. In addition to that, the visual of the training and validation or testing results through graphs are given below in Figure 2.1 and 2.2.

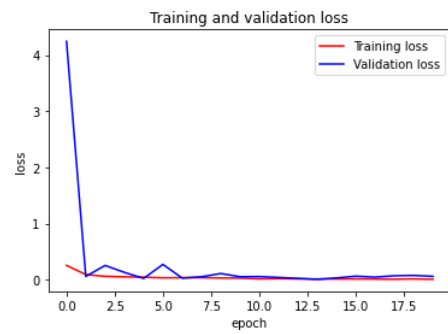
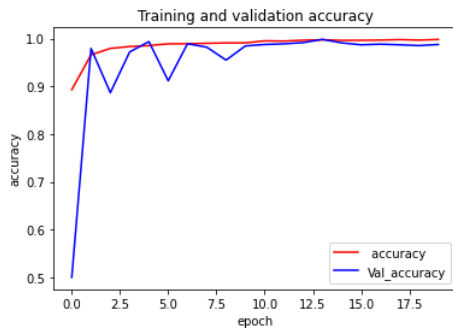


Figure 2.1: Training and testing accuracy Figure 2.2: Training and Validation loss

Moreover, the metrics results of accuracy, loss, auc, recall, summary for training and validation class are shown below in Figure 2.3 and 2.4:

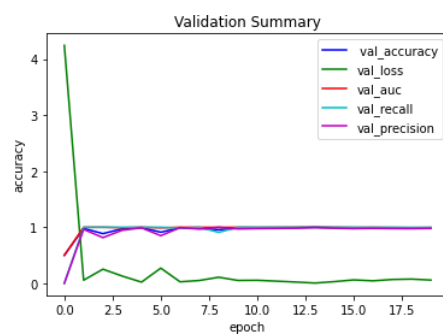
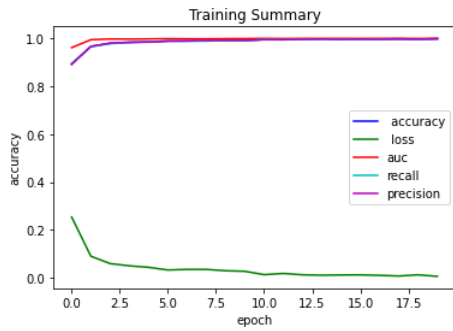


Figure 2.3: Training Summary Graph Figure 2.4: Testing Summary Graph

Further classification report is shown in Table 2.2.

Table 2.2: Classification Report for the Proposed CNN Model

Class	Precision	Recall	F1-Score	Support
Real	0.99	1.00	1.00	150
Fake	1.00	0.99	1.00	150
Accuracy	-	-	1.00	300
Macro avg	1.00	0.51	1.00	300
Weighted avg	1.00	1.00	1.00	300

The confusion metrics for the CNN model is given below in Table 2.3:

Table 2.3: Confusion Matrix

150	0
1	149

5.1 Performance of Pre-trained Models

A total of 3000 photographs have been separated into two portions, with 1500 images in each area. Fake and Real are the names of the two classes. These photos, on the other hand, have been put to the test against pre-trained models. The findings show that InceptionV3 has the best training and testing accuracy of 97.76 percent and 97.10 percent, respectively. According to the findings, the accuracy of VGG-16, DenseNet121, VGG-19, and ResNet50 was reached in declining order. Figures 6.5 and 6.6 depict the models' training and testing accuracy, respectively, while Figure 6.7 depicts the AUC rate comparison:

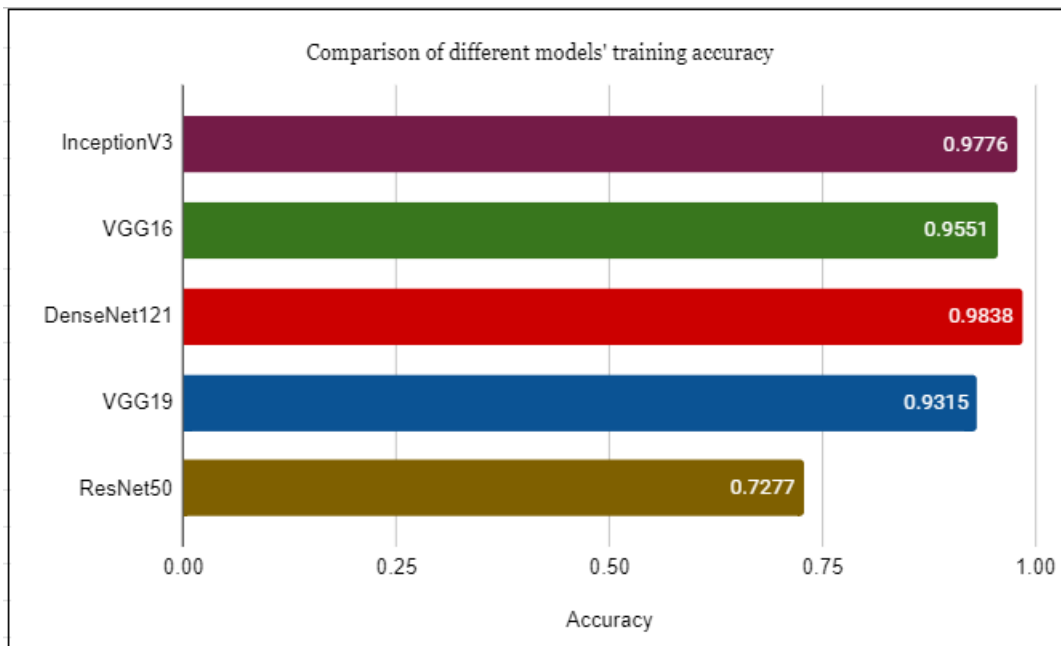


Figure 15 : Training accuracy of the tested pre-trained models

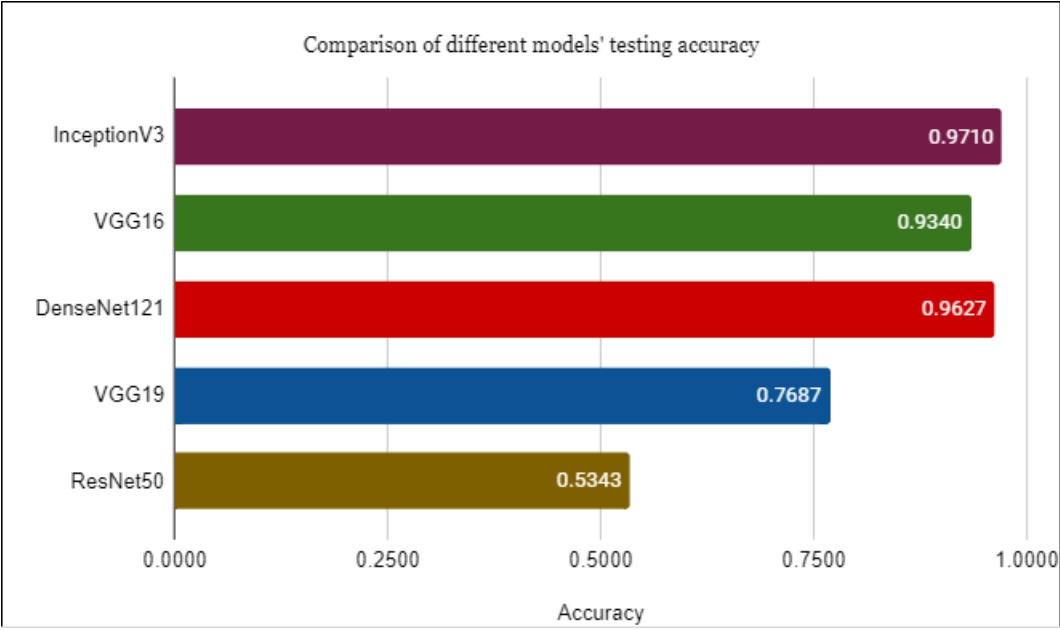


Figure16 Testing accuracy of the tested pre-trained models

CHAPTER 6

CONCLUSION

As the internet advances rapidly in modern society, there are many social network services such as Facebook, Instagram and so on which have been used not only for good reasons but also some take the misuse them for negative purposes. Under these circumstances, crimes against images are appearing for illegal purposes. Digital forensics need to detect such illegal purposes.

In this paper, we proposed image manipulation detection techniques using error level analysis. After we briefly observed the related works, the proposed model was explained in detail. Through intensive experiment the proposed model was analyzed and showed that at least 95 % accuracy was achieved.

The proposed model can be used to determine whether or not the image is manipulated, and can be applied for detection of more manipulation techniques if a better model is established in later studies. In addition, it will be possible to apply it to various multimedia as well as videos in the further research. Under these circumstances, crimes against images are appearing for illegal purposes. Hence, digital forensics need to detect these illegal purposes.

It is clear—from the results of the model used—that a large, deep convolutional neural network is capable of achieving record-breaking results on a highly challenging dataset using supervised learning where the results of this research achieved high accuracy of up to 97%. The results of this research will be helpful in monitoring and tracking social media content and in discovering fraud on social networking sites, especially in the field of images. To effectively identify objects, the convolution neural network architecture implicitly combines the benefits obtained from standard neural

network learning with the convolution process. Like a neural network, CNN and its variants can also be optimized to large datasets, which is often the case when classifying objects.

The recommendations for future work are for example using a more complex and deeper model for unpredictable problems. Integration of deep neural networks with the theory of enhanced learning, where the model is more effective. Neural network solutions rarely take into account non-linear feature interactions and non-monotonous short-term sequential patterns, which are necessary to model user behavior in sparse sequence data. A model may be integrated with neural networks to solve this problem. The dataset could be increased and another type of images could be used for training, for example gray-scale images.

I. FUTURE WORKS

Here are a few more approaches we are trying to get more accuracy:

1. more aggressive data augmentation
2. more aggressive dropout
3. use of L1 and L2 regularization (also known as "weight decay")
4. Fine-tuning one more convolutional block (alongside greater regularization)

REFERENCES

1. Luo, Weiqi, Jiwu Huang, and Guoping Qiu. "Robust detection of region-duplication forgery in digital image." Pattern Recognition, 2006. ICPR 2006. 18th International Conference on. Vol. 4. IEEE, 2006.
2. S. Gholap and P. K. Bora, Illuminant colour based image forensics, in Proc. IEEE Region 10 Conf. 2008
3. Leida Li, Shushang Li, Hancheng Z -Journal of Information Hiding and Multimedia Signal Processing, Vol. 4, No. 1, pp. 46-56, January 2013.
4. Tiago and Christian et al Exposing Digital Image Forgeries by Illumination Color Classification. IEEE Transactions on Information Forensics and Security (Page: 1182 1194) Year of Publication: 2013.
5. Reshma P.D and Arunvinodh C IMAGE FORGERY DETECTION USING SVM CLASSIFIER Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), 2015.
6. S. Shaid. "Types of Image Forgery." Internet: <http://csc.fsksm.utm.my/syed/research/image-forensics/11-types-of-image-forgery.html>, Feb. 08, 2010 12:17 [Dec. 4, 2012].
7. Z. He, W. Sun, W. Lu, and H. Lu. "Digital image splicing detection based on approximate run length," Pattern Recogn. Lett., vol. 32, pp. 1591-1597, 2011.
8. A picture's worth, Digital Image Analysis and Forensics, N Krawetz - 2007 Ph D, Hacker Factor Solutions.

9. <http://imagej.net/> Welcome ImageJ is an open source image processing program designed for scientific multidimensional images. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
10. <http://forensics.idealtest.org/> CASIA v2.0 CASIA V2.0 is with larger size and with more realistic and challenged fake images by using post-processing of tampered regions. It contains 7491 authentic and 5123 tampered color images.
11. <http://neuroph.sourceforge.net/> Neuroph Framework Neuroph is lightweight Java neural network framework to develop common neural network architectures. It contains well designed, open source Java library with small number of basic classes which correspond to basic NN concepts.
12. <https://github.com/drewnoakes/metadata-extractor> Metadata-extractor is a straightforward Java library for reading metadata from image files.
13. . <https://www.github.com/afsalashyana/FakeImageDetection> GitHub repositor for fake image detector desktop application written in javafx. Bellemare, M. G.; Danihelka, I.; Dabney, W.; Mohamed, S.; Lakshminarayanan, B.; Hoyer, S.; and Munos, R. 2017. The cramer distance as a solution to biased wasserstein gradients. arXiv preprint arXiv:1705.10743. Binkowski, M.; Sutherland, D. J.; Arbel, M.; and Gretton, A. 2018. ‘Demystifying MMD GANs. In ICLR. Chai, L.; Bau, D.; Lim, S.-N.; and Isola, P. 2020. What makes fake images detectable? Understanding properties that generalize. In ECCV, 103–120. Springer.
14. Chandrasegaran, K.; Tran, N.-T.; and Cheung, N.-M. 2021. A Closer Look at Fourier Spectrum Discrepancies for CNNgenerated Images Detection. In CVPR, 7200–7209. Chen, T.; Zhai, X.; Ritter, M.; Lucic, M.; and Houlsby, N. 2019. Self-supervised gans via auxiliary rotation loss. In CVPR, 12154– 12163.

15. Durall, R.; Keuper, M.; and Keuper, J. 2020. Watch your upconvolution: Cnn based generative deep neural networks are failing to reproduce spectral distributions. In CVPR, 7890–7899.
16. Frank, J.; Eisenhofer, T.; Schonherr, L.; Fischer, A.; Kolossa, D.; and Holz, T. 2020. Leveraging frequency analysis for deep fake image recognition. In ICML, 3247–3258. PMLR.
17. Haliassos, A.; Vougioukas, K.; Petridis, S.; and Pantic, M. 2021. Lips Don't Lie: A Generalisable and Robust Approach To Face Forgery Detection. In CVPR, 5039–5049.
18. Huh, M.; Liu, A.; Owens, A.; and Efros, A. A. 2018. Fighting fake news: Image splice detection via learned self-consistency. In ECCV, 101–117.
19. Jeon, H.; Bang, Y. O.; Kim, J.; and Woo, S. 2020. T-GD: Transferable GAN-generated Images Detection Framework. In ICML, 4746–4761. PMLR.
20. Joslin, M.; and Hao, S. 2020. Attributing and Detecting Fake Images Generated by Known GANs. In 2020 IEEE Security and Privacy Workshops (SPW), 8–14. IEEE.
21. Karras, T.; Aila, T.; Laine, S.; and Lehtinen, J. 2017. Progressive growing of gans for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196.
22. Karras, T.; Laine, S.; and Aila, T. 2019. A style-based generator architecture for generative adversarial networks. In CVPR, 4401–4410.
23. Karras, T.; Laine, S.; Aittala, M.; Hellsten, J.; Lehtinen, J.; and Aila, T. 2020. Analyzing and improving the image quality of stylegan. In CVPR, 8110–8119.
24. Kendall, A.; Gal, Y.; and Cipolla, R. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In CVPR, 7482–7491.
25. Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; and Krishnan, D. 2020. Supervised contrastive learning. arXiv preprint arXiv:2004.11362.

26. Kim, C.; Ren, Y.; and Yang, Y. 2020. Decentralized Attribution of Generative Models. arXiv preprint arXiv:2010.13974.
27. Lee, K. S.; Tran, N.-T.; and Cheung, N.-M. 2021. InfoMax-GAN: Improved adversarial image generation via information maximization and contrastive learning. In WACV, 3942–3952.
28. Li, H.; Li, B.; Tan, S.; and Huang, J. 2020. Identification of deep network generated images using disparities in color components. *Signal Processing*, 174: 107616.
29. Liu, H.; Li, X.; Zhou, W.; Chen, Y.; He, Y.; Xue, H.; Zhang, W.; and Yu, N. 2021. Spatial-phase shallow learning: rethinking face forgery detection in frequency domain. In CVPR, 772–781.
30. Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2015. Deep learning face attributes in the wild. In ICCV, 3730–3738.
31. Liu, Z.; Qi, X.; and Torr, P. H. 2020. Global texture enhancement for fake face detection in the wild. In CVPR, 8060–8069.

ANNEXURE
ANNEXURE 1: PUBLISHED PAPER

Link

<http://ijariie.com/FormDetails.aspx?MenuScriptId=215761>

FAKE IMAGE DETECTION USING MACHINE LEARNING

Vaibhav Kishore

Department of Computer Science and Engineering,
Integral University, Lucknow.

Mohammad Suaib

Assistant Professor Department of Computer Science and Engineering
Integral University, Lucknow.

Abstract

Image editing is now so widespread because to the availability of image processing tools like Adobe Photoshop or GIMP. Detecting such phone photos is unavoidable if image-based cybercrime is to be exposed. Because of its ubiquity, a photograph captured with a digital camera or smartphone is frequently saved in the JPEG format. The JPEG method works with 8x8 pixel picture grids that are compressed individually. Unmodified photos have a comparable amount of inaccuracy. Due to a comparable quantity of faults over the whole picture, each block should deteriorate at about the same pace during the resaving procedure. Error Level Analysis detects that the compression ratio of this false picture differs from that of the genuine image.

Our paper's goal is to create a picture forensics programme that can detect any type of photo modification. The vertical and horizontal histograms of the error level analysis image were then used to determine the site of the alteration. The suggested method was able to recognize the changed picture while also displaying the specific position of the adjustments, according to the results.

Keywords— *Fake Image detector, Photoshop Edited, Fake Image Processing, Detection, Meta Data Analysis.*

I. INTRODUCTION

The usage of technology in today's world has exploded, and one of the most prevalent forms of communication is the use of photographs. Images are now widely utilized in newspapers, magazines, websites, and ads, and they convey a wealth of information. Because of their widespread use, people's confidence in pictures is growing every day. Picture forging is the act of modifying or manipulating an image by changing some information inside it, and Image Forgery Detection is the process of determining whether the image is authentic or not.

In today's world, an enormous number of individuals have been victims of picture fraud. Many individuals modify photographs with image manipulation software and use them as evidence to deceive the court or numerous other people on social networking sites or applications. As a result, every image uploaded on social media should be assessed and classified as either authentic or fraudulent. Social media is one of the finest tools for socializing, sharing, and spreading knowledge, but it can also mislead individuals, causing mayhem due to unintended misleading propaganda.

This paper will then break down into three suggested methodology for evaluating the original ideas of an image, with the first section focusing on metadata analysis, the second on image error level analysis, and the third section focusing on developing a machine learning model to evaluate the image.

II. LITERATURE REVIEW

Today's forensic techniques for manipulating photographs necessitate the use of an expert to assess the

image's trustworthiness. This method may work for a limited number of photographs, but it is not suggested for a big number of images, such as those found on a social networking platform. As a result, we need to develop a system that can employ existing machine learning techniques to assess whether a picture is real or false, and then make it available to the general public for usage.

- S. Beram and colleagues devised a method for detecting doctoring in digital photographs [4]. Doctoring normally entails a number of processes, which are usually performed in the order of initial picture operations like scaling, rotation, brightness shift, smoothing, and so on. Binary similarity, picture quality, and wavelet statistics are among the statistical aspects that these approaches are dedicated to. The following are the three types of forensic facilities:

1. Image quality metrics: They look at the difference between both the doctored image and the original image. If the actual picture is not accessible, a hazy rendition of the image is used to imitate the test. [2,5]

2. Higher - level wavelet statistics: These statistics are produced from the image's multi-level decomposition.

3. Binary similarities measurements: These measurements capture the texture and correlation within both the Bit planes of lesser relevance, which are more vulnerable to manipulation.

First, single tools are designed to determine the essential image-processing functions in order to affect the identification of doctorate effects. Then, these individual "weak" detectors assembled together to determine the presence of a doctorate in an expert fusion scheme.

4. Enhance the meet the individual needs contrast picture: Contrast enhancement can be used to disguise the visual proof of image manipulation. Evidence of cut-and-paste forgeries may be discovered if these operations are tracked down. Cut-and-paste forgeries can be detected with the use of forensic tools.

5. Detecting histogram equalization in images: The Histogram Equalization Operation, like other contrast enhancement operations, creates spontaneous peaks and gaps in the picture histogram. The methods for detecting picture histogram equalization have been improved.

III. OBJECTIVE OF THE PROJECT

In essence, a metadata analyzer is a tag selection and search algorithm. If keywords like Photoshop, Gimp, Adobe, and other similar terms appear in the text, the likelihood of it being tampered with increases. Fakeness and realness are two distinct characteristics that are kept separate.

These qualities are already added to photographs by cameras and photo modification software if they are utilized, but they may be readily tampered with or modified, therefore they should only be used as a rough guide.

Fault Level Thinking resaves a specific image at a specific error rate, such as 96 percent, and then looks for a virtual change; if one is found, it signifies the cells have hit their global minimum for loss at that quality level.

Machine learning is comparable to data mining in terms of how it works. Both systems sift through data in order to find patterns. Machine learning, on the other hand, analyses data to find patterns in data and change programme operations appropriately, rather than harvesting data for human interpretation as in data mining applications.

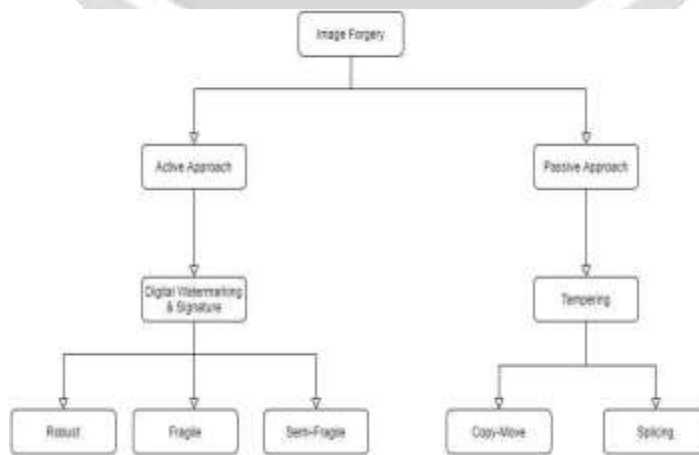


Fig-1: Types of image tampering techniques

IV. PROBLEM DEFINITION

The technology stores an image at 100 percent quality initially, then converts it to a 90 percent quality image. The distinction between the actual can be discovered using the difference approach. The output picture is the input image's needed accuracy level analysis (ELA) image. This image is now saved as a buffering image and delivered to the human brain to be processed further.

- 1) Make, Model, and Software
- 2) Image size
- 3) Timestamps
- 4) Types of metadata
- 5) Descriptions
- 6) Missing metadata
- 7) Altered Metadata

V. PROPOSED METHODOLOGY

1. Metadata Evaluation

In essence, a metadata analyzer is a tag selection and search algorithm. If keywords like Photoshop, Gimp, Adobe, and other similar terms appear in the text, the likelihood of it being tampered with increases.

2. Analysis of Error Levels

Error Level Analysis resaves a specific image at a specific error rate, such as 96 percent, and then looks for a virtual change; if one is found, it signifies the cells have hit their local minima for error at that quality level.

3. The Convolutional Neural Network (CNN) is a type of neural network that

A multilayer perceptron neural network with a few hidden layers on both the input and output levels. When an image is selected for review, it is first transformed from the Compression and Error Level Analysis stage to an ELA representation.

4. Transfer learning: improves the learner by transferring information from one domain to the necessary domain. It's a method of creating a model for one activity and then using it as a starting point for another.

5. Model VGG16 is a convolutional neural network design that focuses on having a stride 1 Convolution layer with the same padding and a stride 2 max pool layer.

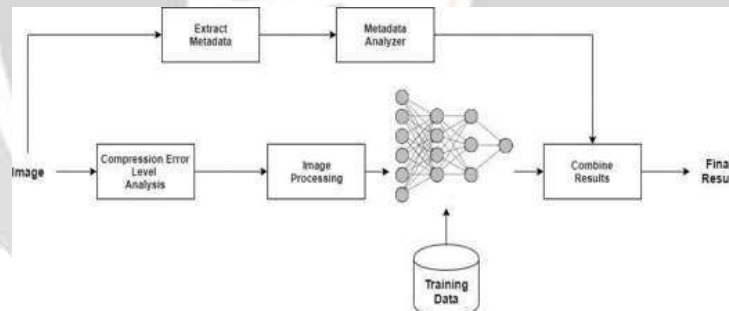


Fig-2: Proposed method architecture.

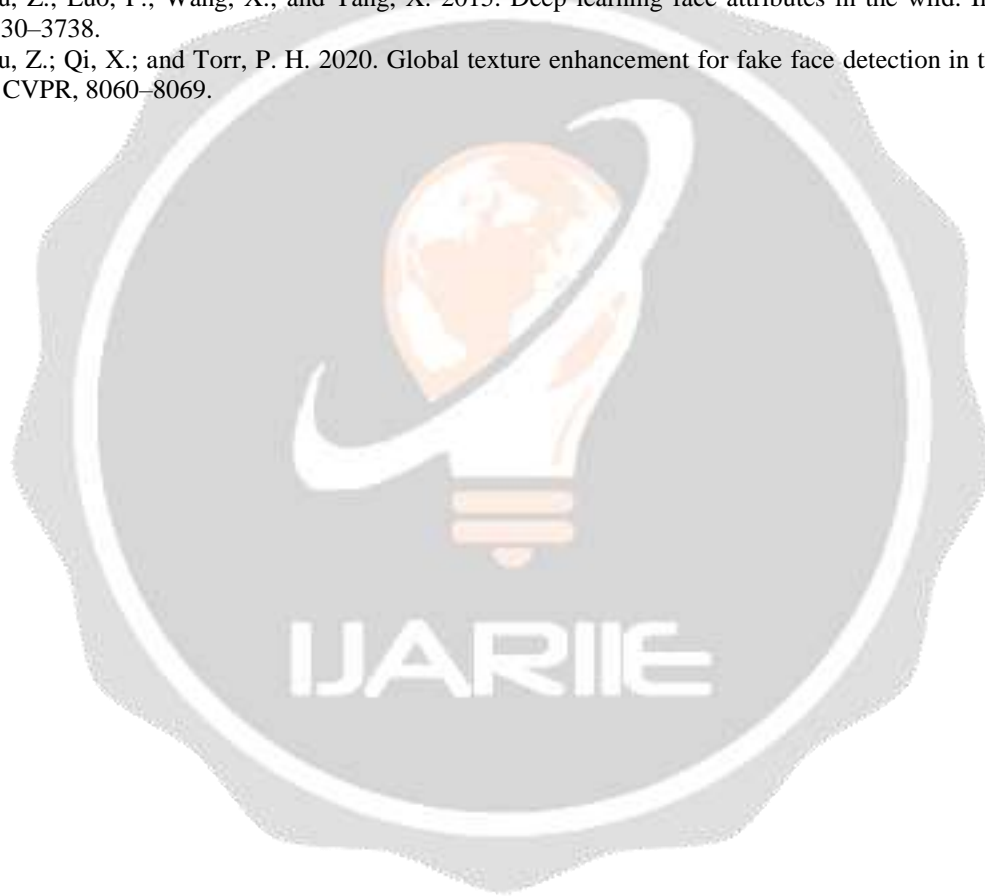
Spark is a fast and general-purpose cluster computing system for large-scale in-memory data processing. Spark has a similar programming model to Map Reduce but extends it with a data-sharing abstraction called Resilient Distributed Datasets or RDD. A Spark was designed to be fast for iterative algorithms, support for in-memory storage and efficient fault recovery. Spark Core consists of two APIs which are the unstructured and structured APIs. The unstructured API is RDDs, Accumulators, and Broadcast variables.

Processing: Large-scale datasets are frequently noisy, duplicated, and contain a variety of data kinds, posing significant hurdles to knowledge discovery and data modelling. In general, intrusion detection algorithms work with one or more forms of raw input data, such as the SVM algorithm, which exclusively works with numerical data. As a result, we prepare the data and transform the dataset's categorical data to numerical data.

VI. REFERENCE

- [1] Luo, Weiqi, Jiwu Huang, and Guoping Qiu. "Robust detection of region-duplication forgery in digital image." *Pattern Recognition*, 2006. ICPR 2006. 18th International Conference on. Vol. 4. IEEE, 2006.
- [2] S. Gholap and P. K. Bora, Illuminant colour based image forensics, in Proc. IEEE Region 10 Conf. 2008.
- [3] Leida Li, Shushang Li, Hancheng Z -*Journal of Information Hiding and Multimedia Signal Processing*, Vol. 4, No. 1, pp. 46-56, January 2013.
- [4] Tiago and Christian et al Exposing Digital Image Forgeries by Illumination Color Classification. *IEEE Transactions on Information Forensics and Security* (Page: 1182 1194)Year of Publication: 2013.
- [5] Reshma P.D and Arunvinodh C IMAGE FORGERY DETECTION USING SVM CLASSIFIER Conference on Innovations in Information, Embedde and Communication Systems (ICIIECS), 2015.
- [6] S.Shaid."TypesofImageForgery."Internet:<http://csc.fsksm.utm.my/syed/research/image-forensics/11-types-of-mageforgery.html>, Feb.08, 2010 12:17 [Dec. 4, 2012].
- [7] Z. He, W. Sun, W. Lu, and H. Lu. "Digital image splicing detection based on approximate run length," *Pattern Recogn .Lett.*, vol. 32, pp. 1591-1597, 2011.
- [8] A picture's worth, *Digital Image Analysis and Forensics*, N Krawetz - 2007 Ph D, Hacker Factor Solutions.
- [9]] <http://imagej.net/Welcome> ImageJ is an open source image processing program designed for scientific multidimensional images.J. Clerk Maxwell, *A Treatise on Electricity and Magnetism*, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
- [10]] <http://forensics.idealtest.org/> CASIA v2.0 CASIA V2.0 is with larger size and with more realistic and challenged fake images by using post-processing of tampered regions. It contains 7491 authentic and 5123 tampered color images.
- [11]] <http://neuroph.sourceforge.net/> Neuroph Framework Neuroph is lightweight Java neural network framework to develop common neural network architectures. It contains well designed, open source Java library with small number of basic classes which correspond to basic NN concepts.
- [12] <https://github.com/drewnoakes/metadata-extractor> Metadata-extractor is a straightforward Java library for reading metadata from image files.
- [13]] <https://www.github.com/afsalashyana/FakeImageDetection> GitHub repositor for fake image detector desktop application written in javafx.
- [14] Bellemare, M. G.; Danihelka, I.; Dabney, W.; Mohamed, S.; Lakshminarayanan, B.; Hoyer, S.; and Munos, R. 2017. The cramer distance as a solution to biased wasserstein gradients. *arXiv preprint arXiv:1705.10743*.
- [15] Binkowski, M.; Sutherland, D. J.; Arbel, M.; and Gretton, A. 2018. ' Demystifying MMD GANs. In *ICLR*.
- [16] Chai, L.; Bau, D.; Lim, S.-N.; and Isola, P. 2020. What makes fake images detectable? Understanding properties that generalize. In *ECCV*, 103–120. Springer.
- [17] Chandrasegaran, K.; Tran, N.-T.; and Cheung, N.-M. 2021. A Closer Look at Fourier Spectrum Discrepancies for CNNgenerated Images Detection. In *CVPR*, 7200–7209. Chen, T.; Zhai, X.; Ritter, M.; Lucic, M.; and Houlsby, N. 2019. Self-supervised gans via auxiliary rotation loss. In *CVPR*, 12154– 12163.
- [18] Durall, R.; Keuper, M.; and Keuper, J. 2020. Watch your upconvolution: Cnn based generative deep neural networks are failing to reproduce spectral distributions. In *CVPR*, 7890–7899.
- [19] Frank, J.; Eisenhofer, T.; Schonherr, L.; Fischer, A.; Kolossa, D.; " and Holz, T. 2020. Leveraging frequency analysis for deep fake image recognition. In *ICML*, 3247–3258. PMLR.
- [20] Haliassos, A.; Vougioukas, K.; Petridis, S.; and Pantic, M. 2021. Lips Don't Lie: A Generalisable and Robust Approach To Face Forgery Detection. In *CVPR*, 5039–5049.
- [21] Huh, M.; Liu, A.; Owens, A.; and Efros, A. A. 2018. Fighting fake news: Image splice detection via learned self-consistency. In *ECCV*, 101–117.
- [22] Jeon, H.; Bang, Y. O.; Kim, J.; and Woo, S. 2020. T-GD: Transferable GAN-generated Images Detection Framework. In *ICML*, 4746–4761. PMLR.
- [23] Joslin, M.; and Hao, S. 2020. Attributing and Detecting Fake Images Generated by Known GANs. In *2020 IEEE Security and Privacy Workshops (SPW)*, 8–14. IEEE.
- [24] Karras, T.; Aila, T.; Laine, S.; and Lehtinen, J. 2017. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*.
- [25] Karras, T.; Laine, S.; and Aila, T. 2019. A style-based generator architecture for generative adversarial networks. In *CVPR*, 4401– 4410.

- [26] Karras, T.; Laine, S.; Aittala, M.; Hellsten, J.; Lehtinen, J.; and Aila, T. 2020. Analyzing and improving the image quality of stylegan. In CVPR, 8110–8119.
- [27] Kendall, A.; Gal, Y.; and Cipolla, R. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In CVPR, 7482–7491.
- [28] Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; and Krishnan, D. 2020. Supervised contrastive learning. arXiv preprint arXiv:2004.11362.
- [29] Kim, C.; Ren, Y.; and Yang, Y. 2020. Decentralized Attribution of Generative Models. arXiv preprint arXiv:2010.13974.
- [30] Lee, K. S.; Tran, N.-T.; and Cheung, N.-M. 2021. InfoMax-GAN: Improved adversarial image generation via information maximization and contrastive learning. In WACV, 3942–3952.
- [31] Li, H.; Li, B.; Tan, S.; and Huang, J. 2020. Identification of deep network generated images using disparities in color components. *Signal Processing*, 174: 107616.
- [32] Liu, H.; Li, X.; Zhou, W.; Chen, Y.; He, Y.; Xue, H.; Zhang, W.; and Yu, N. 2021. Spatial-phase shallow learning: rethinking face forgery detection in frequency domain. In CVPR, 772–781.
- [33] Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2015. Deep learning face attributes in the wild. In ICCV, 3730–3738.
- [34] Liu, Z.; Qi, X.; and Torr, P. H. 2020. Global texture enhancement for fake face detection in the wild. In CVPR, 8060–8069.



ANNEXURE


ANNEXURE 2: PUBLISHED PAPER

New Submission	Submission 61	ADCIS 2022	Conference	News	EasyChair
----------------	---------------	------------	------------	------	-----------

ADCIS 2022 Submission 61

- [Update information](#)
- [Update authors](#)
- [Update file](#)

The submission has been saved!

Submission 61	
Title	FAKE IMAGE DETECTION USING MACHINE LEARNING
Paper:	 (Jun 04, 09:22 GMT)
Author keywords	Fake Image detector Photoshop Edited Fake Image Processing Detection Meta Data Analysis
Abstract	<p>Image editing is now so widespread because to the availability of image processing tools like Adobe Photoshop or GIMP. Detecting such phone photos is unavoidable if image-based cybercrime is to be exposed. Because of its ubiquity, a photograph captured with a digital camera or smartphone is frequently saved in the JPEG format. The JPEG method works with 8x8 pixel picture grids that are compressed individually. Unmodified photos have a comparable amount of inaccuracy. Due to a comparable quantity of faults over the whole picture, each block should deteriorate at about the same pace during the resaving procedure. Error Level Analysis detects that the compression ratio of this false picture differs from that of the genuine image.</p> <p>Our paper's goal is to create a picture forensics programme that can detect any type of photo modification. The vertical and horizontal histograms of the error level analysis image were then used to determine the site of the alteration. The suggested method was able to recognize the changed picture while also displaying the specific position of the adjustments, according to the results.</p>
Submitted	Jun 04, 09:22 GMT
Last update	Jun 04, 09:22 GMT
Status of using third-party material in your article	I am not using third-party material for which formal permission is required

Authors						
first name	last	email	country	affiliation	Web	corresponding?

	name				page	
Vaibhav	Kishore	vaibhavkanaujiya@gmail.com	India	Department Of Technical Education		✓
Dr Mohhamad	Suaib	suaib@iul.ac.in	India	Integral University, Lucknow, Uttar Pradesh		✓

Copyright © 2002 – 2022 EasyChair

ANNEXURE

ANNEXURE 3 : PLAGIARISM REPORT

RE-2022-22270-(2)-plag-report

ORIGINALITY REPORT

15%	22%	17%	14%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	thesai.org Internet Source	4%
2	dspace.bracu.ac.bd:8080 Internet Source	3%
3	www.ijcrt.org Internet Source	3%
4	ijariie.com Internet Source	2%
5	blog.paperspace.com Internet Source	1%
6	Submitted to Program Pascasarjana Universitas Negeri Yogyakarta Student Paper	1%
7	Submitted to British University in Egypt Student Paper	<1%
8	Submitted to Camarines Sur Polytechnic Colleges Student Paper	<1%
9	Submitted to Coventry University	

10

Submitted to South Bank University

Student Paper

<1 %

11

Arpita Dhar, Likhan Biswas, Prima Achariec, Shemonti Ahmed, Abida Sultana, Dewan Ziaul Karim, Mohammad Zavid Parvez. "DFCatcher: A Deep CNN Model to Identify Deepfake Face Images", TENCON 2021 - 2021 IEEE Region 10 Conference (TENCON), 2021

Publication

<1 %

12

www.hindawi.com

Internet Source

<1 %

13

Submitted to Griffith College Dublin

Student Paper

<1 %

14

Submitted to Yakın Doğu Üniversitesi

Student Paper

<1 %

15

ieee.nitk.ac.in

Internet Source

<1 %

16

i2msupdecocameroun.com

Internet Source

<1 %