A ROBUST IMAGE COPY DETECTION METHOD USING MACHINE LEARNING

MAJOR PROJECT REPORT

Submitted in partial fulfillment of the requirements for the award of the degree of

MASTER OF COMPUTER APPLICATIONS

From

The University of Kerala, Thiruvananthapuram

By,

ANGEL MARIA RAJU

Reg.no:95520455011

Under the guidance of

Mr. PRIJI KURIAN ISAC

HOD & Associate Professor

DEPARTMENT OF COMPUTER APPLICATIONS

MAR THOMA INSTITUTE OF INFORMATION TECHNOLOGY

CHADAYAMANGALAM

2022

MAR THOMA INSTITUTE OF INFORMATION TECHNOLOGY

DEPARTMENT OF COMPUTER APPLICATIONS

CHADAYAMANGALAM

CERTIFICATE-I

Certified that this is the Bonafide Record of the Project work entitled

A ROBUST IMAGE COPY DETECTION METHOD USING MACHINE LEARNING

Submitted By,

ANGEL MARIA RAJU

In partial fulfillment for the award of ***Post Graduate Degree in Computer Applications from the University of Kerala****, carried out at****, Thiruvananthapuram*** *during the* ***Fourth Semester*** *academic under our supervision.*

Internal Guide	**Head of the Department**	**Principal**
Asso. Prof. Priji Kurian Isac	Asso. Prof. Priji Kurian Isac	Prof. Dr. K Jacob

MAR THOMA INSTITUTE OF INFORMATION TECHNOLOGY

DEPARTMENT OF COMPUTER APPLICATIONS

CHADAYAMANGALAM

CERTIFICATE-II

Certified that this is the Bonafide Record of the Project work done by the student in the partial fulfillment for the award of the ***Post Graduate Degree in Computer Applications*** *from the* ***University of Kerala, Thiruvananthapuram.***

Name of the Candidate : ANGEL MARIA RAJU

Reg. No : 955204550011

Place : AYUR

Date : 24.11.2022

Internal Guide
Asso. Prof. Priji Kurian Isac

Head of the Department
Asso. Prof. Priji Kurian Isac

Internal Examiner

External Examiner

ACKNOWLEDGEMENT

Let me have the opportunity to thank all those who have been directly or indirectly involved in making my project is a success. First of all, I am grateful to GOD Almighty, for his grace during toughest times.

I am thankful to **Prof. Dr. K. Jacob, Principal, Mar Thoma Institute of Information Technology, Ayur**, for his legal support and permission to do the project.

I am indebted to **Asso. Prof. Priji Kurian Isac, Head of the Computer Applications Department, Mar Thoma Institute of Information Technology, Ayur**, for his support, guidance and help in presenting my project.

I am immensely grateful to**, Mr. Vishnu Thampi. R, HIGBEC Pvt. Ltd Thiruvananthapuram,** where I have done my project, for his guidance, encouragement and support during the course of this project.

Last but not the least; I express my gratitude to my parents and friends who have given me inspiration, mental support and lots of help and encouragement for doing this project successfully.

ANGEL MARIA RAJU

ABSTRACT

In this present time, it is very easy to get a digital image through our smartphone we use everyday. Thus easilyour digital image can be processed and used by other people without our knowledge. The original image gets manipulated and processed for various reasons without the authentication of those who belong to it. Due to this reason image copy detection plays a significant role. This paper tries to present the image copy detection model for the two most common image tampering method which includes copy-image forgery detectionand Image Splicing. Also Feature extraction process is done through the Speeded Up Robust Features(SURF), and Scale Invariant Feature Transform(SIFT) Descriptor method for identifying the matching point. In the case of Image Splicing Detection, it is used for extracting the image edges of Y(brightness in luma), Cb(blue minus luma), and C_r (Red minus luma) of the image components. Gray Level Co-Occurrence Matrix (GLCM) is utilized for each and every edge internal image through which the feature vector is created. The created feature vectoris fed into a Support Vector Machine (SVM Classifier). In this paper, the proposed method for image copy detection represents an outcome showing that the Speeded Up Robust Feature is superior to the Scale InvariantFeature Transform. The proposed method simulates 80 percent accuracy for the detection of tampered images.The image processing technique using the YC_b C_r color model showed a significant result in image splicing detection. The outcome from this method simulates 98 percent positivity for the detection of image splicing.

LIST OF ABBREVIATIONS

DFD	-	Data Flow Diagram
UML	-	Unified Modeling Language
CNN	-	Convolutional Neural Networks
VGG	-	Visual Geometry Group
ReLu	**-**	Rectified Linear Unit

TABLE OF CONTENTS

CHAPTER 1

1. INTRODUCTION

1.1 ABOUT THE PROJECT

Image coping has become one of the easiest jobs to do since images are very easily available on the web. There is much software which easily available for image manipulation throughan original image that can be rechanged so that it appears as a brand new image. A recent report states that there are almost a billion similar digital images available on the web. And the usage of low-cost manipulated images and the dataset of these manipulated photos and videos easily makes the web an easy process for getting those images. For these kinds of image manipulation on the web, the **"A ROBUST IMAGE COPY DETECTION METHOD USING MACHINE LEARNING"** approach has been proposed. Managing digital images of an individual is always a daring thing in the presence of cyberspace society. Present generation due to the availability of smart devices they share their personal lifeday to day activity on social media websites such as Instagram, Twitter, Facebook either as photos or videos. So there is a chance for any random individual to access those photos and manipulated them. Image manipulation is present long back but in this scenario, it has gained more attention among IT (information technology), intelligence service, forensic investigation, movies, social media sites, etc. Image Copy Detection Method has now become an important research topic in various fields which includes computer technology, criminal investigation, biomedical, forensic, etc. this methodhas sought more attention and is considered as one of the challenging processes with the utilization of the latest software system. Social media is one of the places where the more digital image gets shared and viewed by many and thus security issue arises if someone manipulates the original image. The transistion from non-digital to digital life allows data to be stored in digital format and it is a daring task to maintain authentication of those data. The digitally stored files, videos, images, and data can be easily manipulated and forged without the knowledge of the user. The original image will be altered slightly which is termed as near-duplicate image detection. Feature extraction must be used to over image manipulation. Because it is an accurate step to detect a near-duplicate image. It is more critical because the features extracted directly influence the efficiency in the detection/retrieval. The motiva tion to propose efficient copy-move forgery detection methods with the use of optimization techniques and a classifier is discussed.

CHAPTER 2

2. SYSTEM REQUIREMENTS

2.1 HARDWARE REQUIREMENTS

Processor	:	Intel i3 and above
Processor speed	:	1.7GHZ
Random Access Memory	:	8 GB
Hard Disk Memory	:	512 GB
Monitor	:	Color Monitor

2.2 SOFTWARE REQUIREMENTS

Operating System	:	Windows /Linux
Front End	:	Python
Back End	:	Python

2.3 TECHNOLOGIES USED

A set of programs associated with the operation of a computer is called software. Software is the part of the computer system which enables the user to interact with several physical hardware devices. The minimum software requirement specifications for developing this project are as follows:

Operating System: Windows 10

Documentation Tool: MS Word

2.3.1 PYTHON

Python is a widely used high-level programming language for general-purpose programming, created by Guido van Rossum and first released in 1991. An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages suchas C++ or Java. The language provides constructs intended to enable writing clear programs on botha small and large scale.

Python features a dynamic type system and automatic memory management and supports multiple including object-oriented, imperative, functional programming, and procedural styles. It has a large and comprehensive standard library.

Python interpreters are available for many operating systems, allowing Python code to run on a wide variety of systems. CPython, the reference implementation of Python, is open source software and has a community-based development model,as do nearly all of its variant implementations. CPython is managed by the non-profit Python Software Foundation(PSF).

Python’s development is conducted largely through the Python Enhancement Proposal(PEP) process. The PEP process is the primary mechanism for proposing major new features, for collecting community-input on an issue, and for documenting the design decisions that have goneinto Python. Outstanding PEPs are reviewed and commented Age Estimation and Gender Recognition System up on by the Python community and by Van Rossum, the Python project’s Benevolent Dictator for Life.

2.3.2 NumPy

NumPy, which stands for Numerical Python, is a library consisting of multidimensional array objects and a collection of routines for processing those arrays. Using NumPy, mathematical and logical operations on arrays can be performed. NumPy is a Python package. It stands for 'Numerical Python'. It is a library consisting of multidimensional array objects and a collection of routines for processing of array. Numeric, the ancestor of NumPy, was developed by Jim Hugunin. Another package Numarray was also developed, having some additional functionalities. In 2005, Travis Oliphant created NumPy package by incorporating the features of Num array into Numeric package. There are many contributors to this open-source project.

2.3.3 Keras

Keras is a deep learning API written in Python, running on top of the machine learning platform TensorFlow. It was developed with a focus on enabling fast experimentation. Being able to go from idea to result as fast as possible is key to doing good research. Keras is Simple but not simplistic. Keras reduces developer cognitive load to free you to focus on the parts of the problem that really matter. Keras is Flexible, Keras adopts the principle of progressive disclosure of complexity: simple workflows should be quick and easy, while arbitrarily advanced workflows should be possible via a clear path that builds upon what you've already learned. Keras is Powerful, Keras provides industry-strength performance and scalability: it is used by organizations and companies including NASA, YouTube, or Waymo.

2.3.4 Tensorflow

TensorFlow is an end-to-end platform that makes it easy for you to build and deploy ML models. TensorFlow offers multiple levels of abstraction so you can choose the right one for your needs. Build and train models by using the high-level Keras API, which makes getting started with TensorFlow and machine learning easy. If you need more flexibility, eager execution allows for immediate iteration and intuitive debugging. For large ML training tasks, use the Distribution .Strategy API for distributed training on different hardware configurations without changing the model definition. TensorFlow has always provided a direct path to production. Whether it's on servers, edge devices, or the web, TensorFlow lets you train and deploy your model easily, no matter what language or platform you use. Use TensorFlow Extended (TFX) if you need a full production ML pipeline. For running inference on mobile and edge devices, use TensorFlow Lite.

2.3.5 Machine Learning

Machine learning (ML) is a type of artificial intelligence (AI) that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so. Machine learning algorithms use historical data as input to predict new output values. Labeled data has both the input and output parameters in a completely machine-readable pattern, but requires a lot of human labor to label the data, to begin with. Unlabeled data only has one or noneof the parameters in a machine-readable form. This negates the need for human labor but requiresmore complex solutions. There are also some types of machine learning algorithms that are used in very specific use-cases,but three main methods are used today.

Supervised Learning - Supervised learning is one of the most basic types of machine learning. In this type, the machine learning algorithm is trained on labeled data. Even though the data needs to be labeled accurately for this method to work, supervised learning is extremely powerful when used in the right circumstances. In supervised learning, the ML algorithm is given a small training dataset to work with. This training dataset is a smaller part of the bigger dataset and serves to give the algorithm a basic ideaof the problem, solution, and data points to be dealt with. The training dataset is also very similar to the final dataset in its characteristics and provides the algorithm with the labeled parameters required for the problem.

The algorithm then finds relationships between the parameters given, essentially establishing a cause and effect relationship between the variables in the dataset. At the end of the training, the algorithm has an idea of how the data works and the relationship between the input and the output. This solution is then deployed for use with the final dataset, which it learns from in the same way as the training dataset. This means that supervised machine learning algorithms will continue to improve even after being deployed, discovering new patterns and relationships as it trains itself on new data.

Unsupervised Learning - Unsupervised machine learning holds the advantage of being able to work with unlabeled data. This means that human labor is not required to make the dataset machine-readable, allowing much larger datasets to be worked on by the program.

In supervised learning, the labels allow the algorithm to find the exact nature of the relationship between any two data points. However, unsupervised learning does not have labels to work offof, resulting in the creation of hidden structures. Relationships between data points are perceived by the algorithm in an abstract manner, with no input required from human beings. The creation of these hidden structures is what makes unsupervised learning algorithms versatile. Instead of a defined and set problem statement, unsupervised learning algorithms can adapt to thedata by dynamically changing hidden structures.

Reinforcement learning - Reinforcement learning directly takes inspiration from how human beings learn from data in their lives. It features an algorithm that improves upon itself and learns from new situations using a trial-and-error method. Favorable outputs are encouraged or ‘reinforced’, and non-favorable outputs are discouraged or ‘punished’. Based on the psychological concept of conditioning, reinforcement learning works by putting the algorithm ina work environment with an interpreter and a reward system. In every iteration of the algorithm, the output result is given to the interpreter, which decides whether the outcome is favorable or not. In every iteration of the algorithm, the output result is given to the interpreter, which decides whether the outcome is favorable or not.

In case of the program finding the correct solution, the interpreter reinforces the solution by providing a reward to the algorithm. If the outcome is not favorable, the algorithm is forced to reiterate until it finds a better result. In most cases, the reward system is directly tied to the effectiveness of the result.

In typical reinforcement learning use-cases, such as finding the shortest route between two points on a map, the solution is not an absolute value. Instead, it takes on a score of effectiveness, expressed in a percentage value. The higher this percentage value is, the more reward is given to the algorithm. Thus, the program is trained to give the best possible solution for the best possible reward.

CHAPTER 3

3. LITERATURE REVIEW

In this paper, the extracted image features and analyzing it todetect the forged images and also determine the type of the forgery whether it is copy-move or splicing. The present work is to test multiple datasets. In today's world, digital images are widely used in various domains such as; newspapers, scientific journals, magazines,and many other fields. Unfortunately, today's digital technology made it easy for digital images to be forged due to the availability of low-cost photo editing software.

Due to the huge development of technology, the usage of the digital image has been expanding day by day in our daily lives. Because of this forgery of the digital image has turned out to be increasingly straightforward and indiscoverable. At present technology where anything can be controlledor changed with the assistance of modern technology had started to disintegrate the authenticity of images counterfeiting and forgeries with the move to the Megapixels, which gives a new way for forgery. Imitation is not new to humankind but rather it is past generation issues. In the past it was limited to craftsmanship and composing yet did not impact the overall population. The ability to create image forgery is nearly as old as photographyitself. Over a two-decade, photography is a normal and fascinating art which turned out for creating portraits and by that portrait photographers can earn money by making forgery possible by enhancing deals by retouching their photographs. Image forgery detection had gained more attention and incredible investigation in various fields such as computer visualization, image processing, biomedical tools, immoral analysis, image forensics, etc. It gained further attention anddemanding due to advanced software's that become difficultto verify whether an image is influenced by naked eyes.

However, now because of the advanced software and development of numerous gadgets,a photo can be changed and altered. It prompts to exceedinglyworrying for people to distinguish the picture is authenticate or forged. The accessibility of digital image processing software includes Photo Shop, Adobe Photoshop makes it moderatelysimple to generate the forged image from several images. In any case, the image contains many sources of data, and the dependability of digital images is therefore turning into an imperative issue. A photo may worth a thousand words however, nearby, it might have scores of analysis. Images areused to elucidate extreme ideas and move us easily in each field. The work in this chapter reveals the importance of imageforgery detection in real-world applications. It is a difficult task to identify the forgery image. This can be possible by adopting image processing techniques for efficient detection. Due to the availability of image processing techniques, editing tools, the conversion of images has become so easy and accessible. Nowadays image processing techniques are widely used in all fields due to easy manipulation, low expensive, less computation time, and efficient result. In this research detection of image forgery by image processing techniques to differentiate the forgery and authenticate image. Image forensics is a well-developed field that analyzes the images of specific conditions to build up trust and genuine- ness. It is a quick and better-known domain due to several executions of real-time applications in numerous areas that incorporate intelligence, sports, legitimate administrations, news reporting, medical imaging, and protection assert investigations. An image can be faked by modifying the image features characteristics such as brightness, darkness, or image parameters or concealing data.

Image copy implies altering the digital image to some meaningful or valuable data. Simply it can define as the process of inserting or eliminating the specific features from an image without any proof of altering and to evade for malicious purposes. In some cases, it is complicated to recognize the altered image part from the authenticate image. The identification of a forged image is essential for originality and to preserve the truthfulness of the image. A forgery detection that endeavors the unobtrusive irregularities in the color shade of the illumination changes in images. To accomplish this by consolidating data from material science and statistical-based illuminate estimators on image regions to separate texture and edge-based features. The copy-move is defined by copying a region of an image and pasting it in another place in the same image, generally to hide unwanted parts of the image. On the other hand, image splicing is the process of copying a region of an image and pasting it in another place in another image. Thus, detection of tampered regions is done through searching for very similar regions in copy-move images and completely odd regions in spliced images.

CHAPTER 4

4. PROBLEM DEFINITION

4.1 INTRODUCTION

Image coping has become one of the easiest jobs to do since images are very easily available on the web. There is much software which easily available for image manipulation throughan original image that can be rechanged so that it appears as a brand new image. A recent report states that there are almost a billion similar digital images available on the web. And theusage of low-cost manipulated images and the dataset of these manipulated photos and videos easily makes the web an easy process for getting those images. For these kinds of image manipulation on the web, the “A ROBUST IMAGE COPY DETECTION METHOD USING MACHINE LEARNING” approach has been proposed. Managing digital images of an individual is always a daring thing in the presence of cyberspace society. Present generation due to the availability of smart devices they share their personal lifeday to day activity on social media websites such as Instagram, Twitter, Facebook either as photos or videos.

4.2 EXISTING SYSTEM

The work in this chapter reveals the importance of image forgery detection in real-world applications. It is a difficult task to identify the forgery image. This can be possible by adopting image processing techniques for efficient detection.Due to the availability of image processing techniques, editing tools, the conversion of images has become so easy and accessible. Nowadays image processing techniques are widely used in all fields due to easy manipulation, low expensive, less computation time, and efficient result. In this research detection of image forgery by image processing techniques to differentiate the forgery and authenticate image.

Image forensics is a well-developed field that analyzes the images of specific conditions to build up trust and genuineness. It is a quick and better-known domain due to several executions of real-time applications in numerous areas that incorporate intelligence,

sports, legitimate administrations, news reporting, medical imaging, and protection assert investigations. An image can be faked by modifying the image features characteristics such as brightness, darkness, or image parameters or concealing data.

4.2.1 LIMITATIONS OF EXISTING SYSTEM

- Time consuming.
- It is a difficult task to identify the forgery image.
- Record maintenance issues.

4.3 PROPOSED SYSTEM

This section is divided into two subsections; the copy-move detection technique, and the splicing detection technique. The explanation on the algorithm in both techniques, the workflow, and the datasets are represented in the block diagram.

A. **Proposed Method of Copy-Move Forgery Detection**

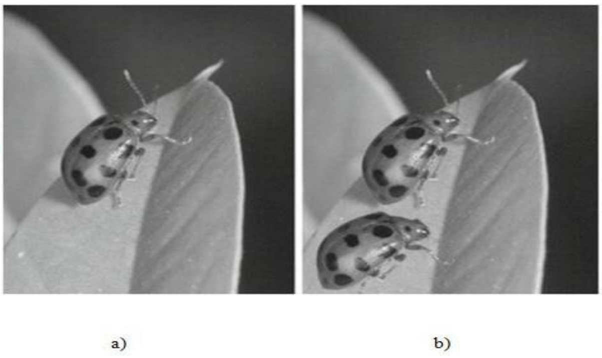


Figure 2. Example of Copy-Move Forgery a) original image, b) tampered image

1. Working Plan

In copy-move detection, based on (Vin-cent, et al 2011). Given an image, the detected regions are computed through the following steps:

Step 1: Convert the image from RGB to the gray-scale color model.

Step 2: Divide the image into 4 equal blocks andcalculate their integral features.

Step 3: Divide each of the 4 blocks into another four blocks of the same size and execute their features.

Step 4: Extract key-points of all blocks using SIFT and SURF.

Step 5: Calculate a feature vector for each key-point.

Step 6: Match each feature vector by comparingeach block's features executed with another block.

Step 7: The forgery is then detected according toa certain threshold among all blocks.

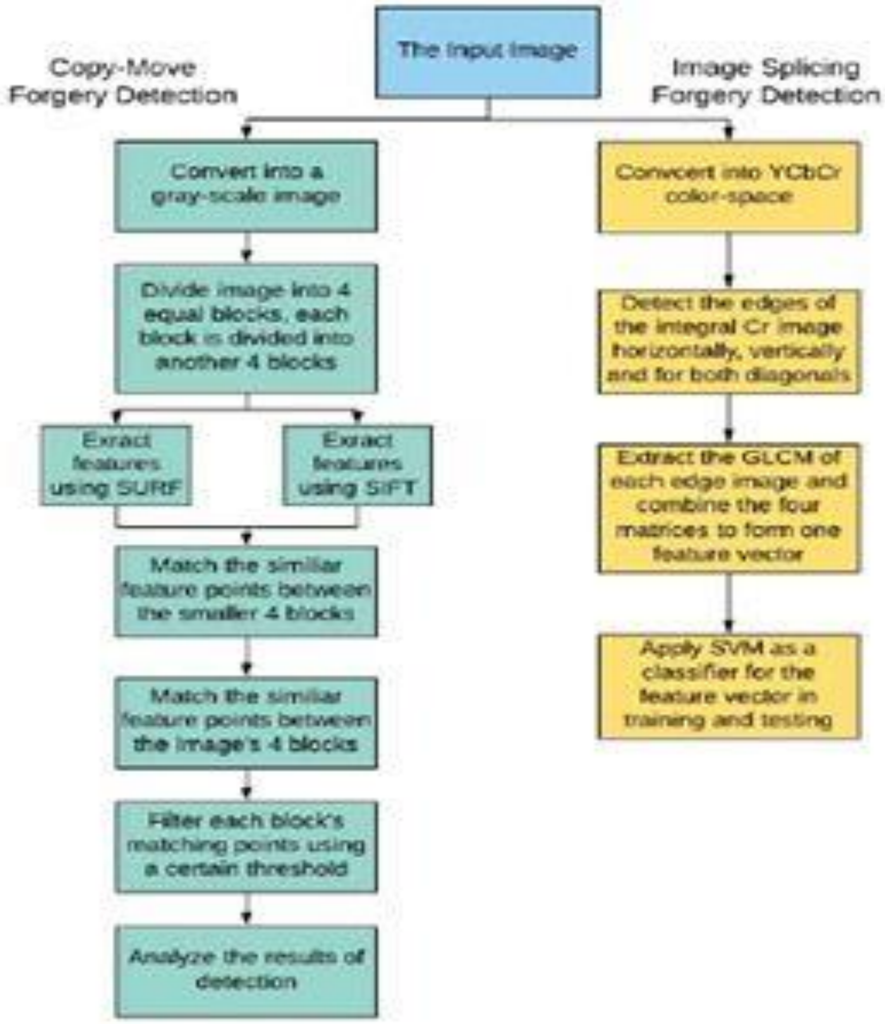
Step 8: The detected blocks are then displayedwith the common object plotted.

Pre-processing: In the beginning, the system was de- signed using MATLAB, where it requests an RGB image of any format, then the system converts it into a gray-scale. Then the image now is ready for the blocking process. A simple two stages algorithm is then used to divide an image into blocks. In the first stage, the image is divided into 4 equal blocks of the same size andangle. Similarly in the second stage, the system divideseach block into another 4 equal smaller blocks. This approach is called Multi Staged blocking. The blocking technique eases the feature extraction and matching processes that will be discussed later.

Features Extraction: For the features extraction, bothkey-point based methods were used; SIFT & SURF approaches for each block. SIFT Key-points based method: SIFT (Scale-InvariantFeature Transform) is an algorithm to detect and de- scribe local features in an image. The SIFT algorithm converts an image into a local feature vector called SIFT descriptors and these descriptors have powerful geometric transformations that are constant to scalingand rotation.

In addition to extracting the features using SIFT, Harris features on the gray image is used to find the corner points. This process is applied to each block of the image. As a result, it is to obtain the valid points for theneighboring features. SURF Key-points based method:similar to SIFT, SURF (Speed Up Robust Feature) is a descriptor used to recognize and locate objects. The values of Hessian determination for each pixel in the image are used to find the points of interest. Next, functions are constructed to be used to select extreme points.

Alternatively, we replace the SIFT step with the SURF.Then, we find the corner points using the Harris detection on the gray image. This process is performed on each block of the image.


The input image
Copy-Move Forgery Detection
Image Splicing Forgery Detection
Convert into a gray-scale image
Divide image into 4 equal blocks, each block is divided into another 4 blocks
Exract features using SURF
Exract features using SIFT
Match the similiar feature points between the smaller 4 blocks
Match the similiar feature points between the image's 4 blocks
Filter each block's matching points using a certain threshold
Analyze the results of detection
Convert into YCbCr color-space
Detect the edges of the integral Cr image horizontally, vertically and for both diagonals
Extract the GLCM of each edge image and combine the four matrices to form one feature vector
Apply SVM as a classifier for the feature vector in training and testing

Matching Points: After extracting the neighboring features of each block, the neighboring features are com- pared to features of another block to find the matched features. Successfully, the locations of the corresponding points for each block will be determined. Ultimately, the system allows the user to view the corresponding points. The system shows the two suspicious blocks where they exceeded the threshold of detected matchedpoints.

Filtering & Analyzing: The blocks are filtered according to a threshold for the number of matching points detected between two blocks. The threshold is calculated from the average number of matched points detected inthe datasets.

The system calculates a percentage of the forgery in the image based on the number of suspicious blocks. Accordingly, the percentage of forgery decides which key-point-based method works better on the datasets.

A. **Proposed Method of Image Splicing Forgery Detection**Regarding the image splicing forgery detection the algorithmis based on the Gray Level Co-occurrence Matrix (GLCM) for feature extraction similar to and the Support Vector Machine (SVM) for classification.

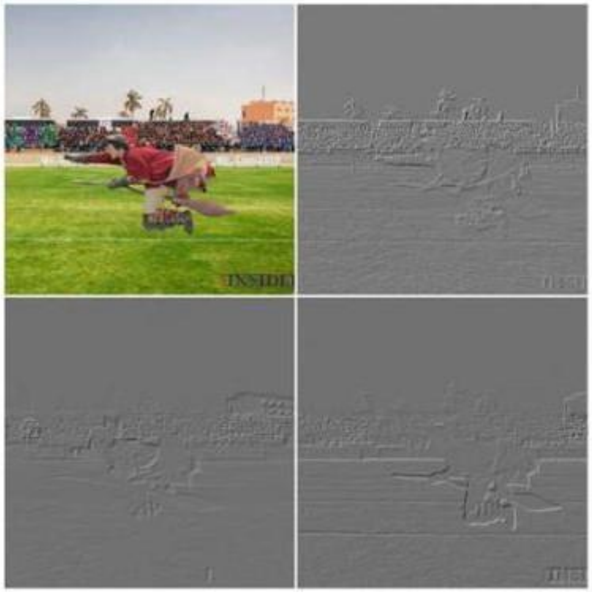


Figure 4. An example of spliced image and the DiagonalEdge detection of RGB

Working Plan: Given an RGB image as an input, the system runs as follows:

Step 1: Convert the RGB image to the YCbCrim-age component.

Step 2: Extract each color channel.

Step 3: Edge detection is performed on each individual color channel image resulting in edge images. The edges are detected horizontally, vertically and both combined.

Step 4: Gray Level Co-occurrence Matrix (GLCM)is calculated for each edge, holding the features of the edge image.

Step 5: These features are given to the Support Vector Machine (SVM) to decide whether a forgery is detected or not.

Review on the System Algorithm: The present algorithm assumes that the images are colored as colors encode relevant information and sensitive to lighting condition at the moment of image acquisition. There- fore, it is expected to have homogeneous color distribution in case of image splicing. Unlike the copy-move forgery detection, the used YCbCr color model instead of gray-scale images. Y is the component of luminescence that contains most of the image content. Cband Cr are the component of chroma blue-difference and red-difference.

4.3.1 MERITS OF PROPOSED SYSTEM

- Better accuracy
- Less time consuming
- More images can be predicted

4.4 FEASIBILITY STUDY

The main objective of the feasibility study is to test the economical, technical and operational feasibility while developing the system. This analysis is done by investigatingthe existing system in the area under generating an idea about the new system. Feasibility study is a test of proposed system regarding its workability, impact on the organization, ability to meet the needs and effective use of resources. Thus, when a new project is proposed, it normally goes through a feasibility study before it is approved for development. The study is made to see if the project on completion will serve the purpose of the organization for the amount of work, effort and the time that is spend on it. Three key considerations involved in the feasibility analysis are:

- Technical Feasibility
- Operational Feasibility
- Economic Feasibility

4.4.1 TECHNICAL FEASIBILITY

The site must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the site requirement in the terms of input, output, programs and procedures. Technical feasibility centers around computer system and to what extend it can support the proposed addition. For example, if the current computer system is operating at 80% capacity then running another application could overload the system or requires additional hardware. This involves the financial considerations to accommodate the additional technical enhancements. If budget is not a serious constraint, then the project is judged technically feasible. Since no further addition of hardware or software is needed, the proposed system Reading aid for the blind people using Tesseract and AlexNet.

4.4.2 OPERATIONAL FEASIBILITY

The operational feasibility depends up on whether system performed in the expected way or not. The application developed is so simple and user friendly there is no special user training is required. So this application can be said to be operationally feasible. The proposed system is very much user friendly and operations on it can be done very easily. The language used inis English and every people can operate it reading options. Quick responses for the queries are available. The website is focused on providing a user-friendly environment so that any users can do what they like. The processes like registration, login all are simple processes. There is no need of special training required for the users.

4.4.3 ECONOMIC FEASIBILITY

The developing system must be justified cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest. This deals with whether expected cost saving, increase the profits and reductions in required investment, and other benefits exceed the cost of developing and operating the proposed system. Its preliminary investigation is concentrated on costs of hardware and software. The online application entitled Reading aid for the blind people using Tesseract and AlexNetis economicallyfeasible because it reduces the network traffic and the need for high band width mediums.

CHAPTER 5

5. SYSTEM DESIGN

5.1 INTRODUCTION

System Design develops the architectural details required to build system or product. The system design process encompasses the following activities:

- Partition the analysis model into subsystems
- Identify concurrency that is dictated by the problem.
- Develop design for the user interface.
- Choose a basic strategy or implementing data management.
- Identify global resources and the control mechanisms required to access them
- Design an appropriate control mechanism for the system, including task management.

5.2 ADOPTION OF MODULES

The application consist of Four Modules

- Preprocessing
- Deep Learning model
- Model training and core relation analysis
- Model testing and result analysis

5.2.1 Preprocessing

Data preprocessing is a data mining technique which is used to transform the raw data in a useful and efficient format. Steps Involved in Data Preprocessing:

1. Data Cleaning:

The data can have many irrelevant and missing parts. To handle this part, data cleaning is done. It involves handling of missing data, noisy data etc.

(a). Missing Data:

This situation arises when some data is missing in the data. It can be handled in various ways. Some of them are:

Ignore the tuples:

This approach is suitable only when the dataset we have is quite large and multiple values are missing within a tuple.

Fill the Missing values:

There are various ways to do this task. You can choose to fill the missing values manually, by attribute mean or the most probable value.

(b). Noisy Data:

Noisy data is a meaningless data that can’t be interpreted by machines.It can be generated due to faulty data collection, data entry errors etc. It can be handled in following ways :

Binning Method:

This method works on sorted data in order to smooth it. The whole data is divided into segments of equal size and then various methods are performed to complete the task. Each segmented is handled separately. One can replace all data in a segment by its mean or boundary values can be used to complete the task.

Regression:

Here data can be made smooth by fitting it to a regression function.The regression used may be linear (having one independent variable) or multiple (having multiple independent variables .

5.2.2 Machine Learning Model Building from Scikit learn

Scikit-learn is an open source Python library that is used to build machine learning model using various algorithms. There are several models that you can choose according to the objective that you might have you will use algorithms of classification, prediction, linear regression.

5.2.3 Model Training and Correlation Analysis

Model training is done by training the datasets to run smoothly and see an incremental improvement in the prediction rate. Remember to initialize the weights of your model randomly the weights are the values that multiply or affect the relationships between the inputs and outputs-which will be automatically adjusted by the selected algorithm the more you train them Correlation analysis is used to find the correlation between data points on data set there by improve prediction accuracy .

5.2.4 Model Testing and Result Analysis

Model testing is the process of testing the learned properties. The model does not know and verify the precision of your already trained model. So, we use testing data set having 20% data fortesting. after the testing we can conclude the model accuracy and if the accuracy is low we can retrain the model either minimize error or choose another algorithm for training .

5.3 DATA FLOW DIAGRAM

A dataflow diagram is a graphical technique that depicts information and transforms that are applied as data move from input to output. The DFD is used to represent increasing information flow and functional details. A level-0 DFD is also called a fundamental system model represents the entire software elements as a single bible with input and output indicated by incoming 3and outgoing arrows respectively.

Additional process and information flow parts are represented in the next level, i.e., level 1 DFD. Each of the process represented at level 1 are sub functions of overall system depicted in the context model. Any processes that are complex in level 1 level will be further represented into sub functions in the next level, i.e., level 2

5.3.1 ADVANTAGES

- Users easily understood these simple notations.
- Users can make suggestions for modifications.
- They can also spot problem quickly
- If analyst wants to overview the overall system late, they use the higher overview

5.3.2 RULES FOR CONSTRUCTING A DATA FLOW DIAGRAM

- Arrows should not cross each other.
- Squares, circles and files must bear names.
- Decomposed data flow squares and circles can have same names.
- Choose meaningful names for data flow.
- Draw all data flows around the outside of the diagram.

5.3.3 COMPONENTS OF DATA FLOW DIAGRAM

Data Flow Diagram (DFD) is an important tool used by system analyst. DFD provide an overview of what data a system would process, what transformation of data are done, what files are used and where the results flow. The graphical representation of the system makes it a good communication tool between the user and the analyst. Analysis model help us to understand the relationship between different components in the design.

The analysis modeling must achieve three primary objectives.

- To establish a basis for creation of software design.
- To describe what the user requires.
- To define set of requirements that can be validated once the software us build.

A data flow diagram is a graphical technique that depicts information flow and transforms that are applied as data move from input to output. The DFD is used to represent increasing information flow and functional details. A level 0 DFD also called fundamental system model represents the entire software elements as single bubble with input and output indicated by incoming and outgoing arrow respectively. To construct the data flow diagram we use arrows, circle, and rectangles. A Data Flow Diagram (DFD) is a graphical representation of the "flow" of data through an information system, modeling its process aspects.

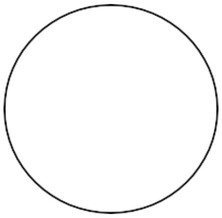
Components of Data Flow Diagram

There are only four symbols that are used in the drawing of data flow diagrams.
These are explained below together with the rules that apply to them.

External Entities

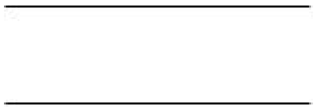
External entities represent the sources of data that enter the system or the recipients of data that leave the system.

Process



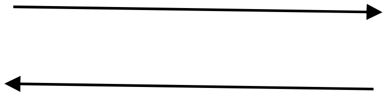
Processes represent activities in which data is manipulated by being stored or retrieved or transformed in some way.

Data Stores



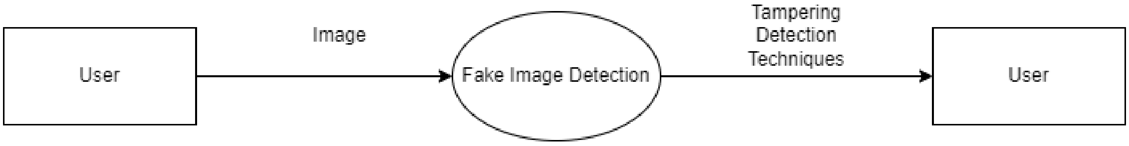
Data stores represent stores of data within the system.

Data Flow

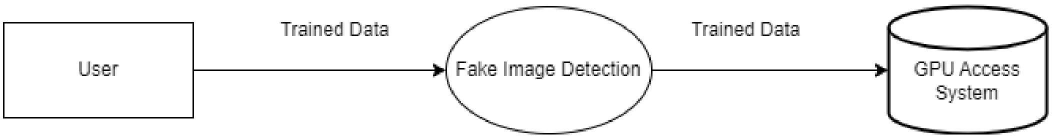


A data flow shows the flow of information from its source to its destination. A line represents a data flow, with arrowheads showing the direction of flow.

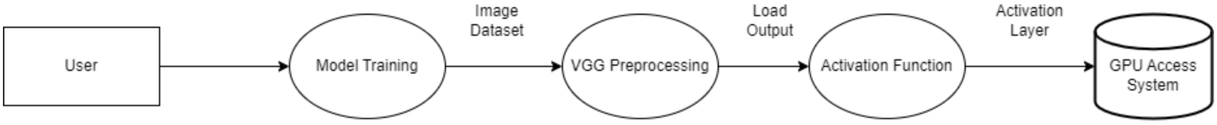
5.3.4 CONTEXT LEVEL DIAGRAM



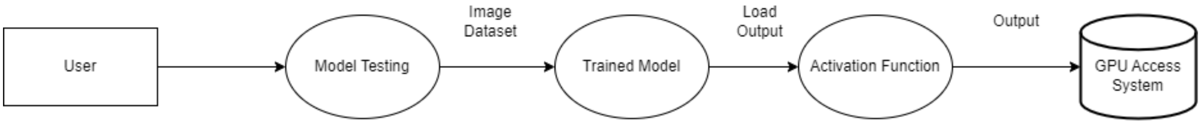
5.3.4.1 LEVEL 1



5.3.4.2 LEVEL 2



5.3.4.3 LEVEL 3



5.4 UML DIAGRAM

The Unified Modeling Language (UML) is a graphical language for visualizing, specifying, constructing and documenting the artifacts of a software-intensive system. The UML offers a standard way to write a systems blueprints, including conceptual things such as business processes and system functions as well as concrete things such as programming language, database schemas, and reusable software components. Here three UML diagrams are specified. They are:

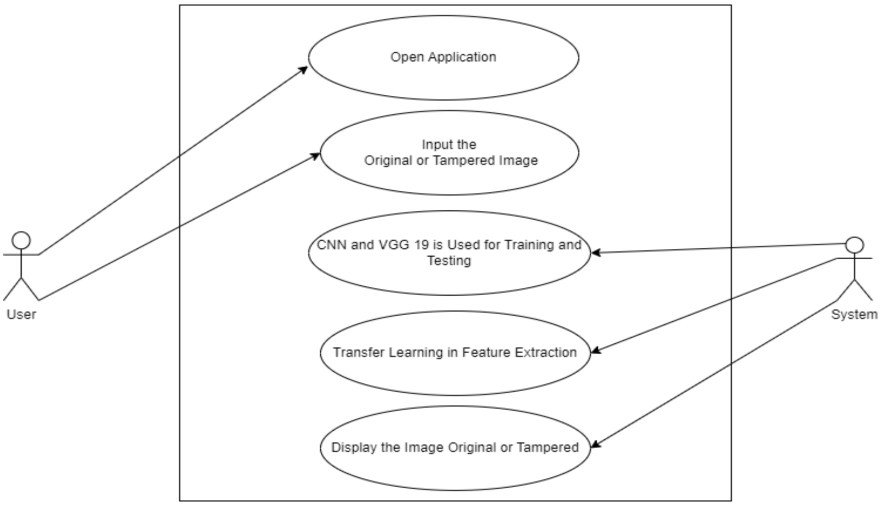
Use Case diagram

Sequence diagram

Modular Diagram

5.4.1 USE CASE DIAGRAM

A use case is the set of scenarios that describing an interaction between a user and a system. A use case diagram displays relationship among actors and use cases. The two main components of a use diagram are use case and act. A user is an external view of the system that represents some action the user might perform in order to complete a task. The actor can be a human external system.


Open Application
Input the Original or Tampered Image
CNN and VGG 19 is Used for Training and Testing
Transfer Learning in Feature Extraction
Display the Image Original or Tampered
User
System

5.4.2 MODULAR DIAGRAM

Modular diagram is used to represent the modules in the software. In the diagram rectangles and arrows are used to represent the working of modules.

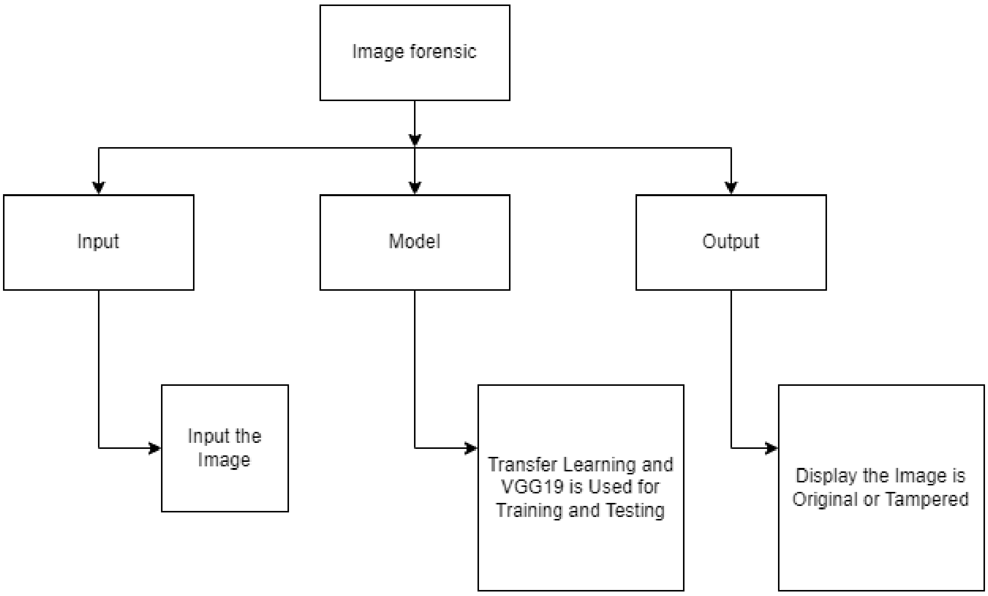
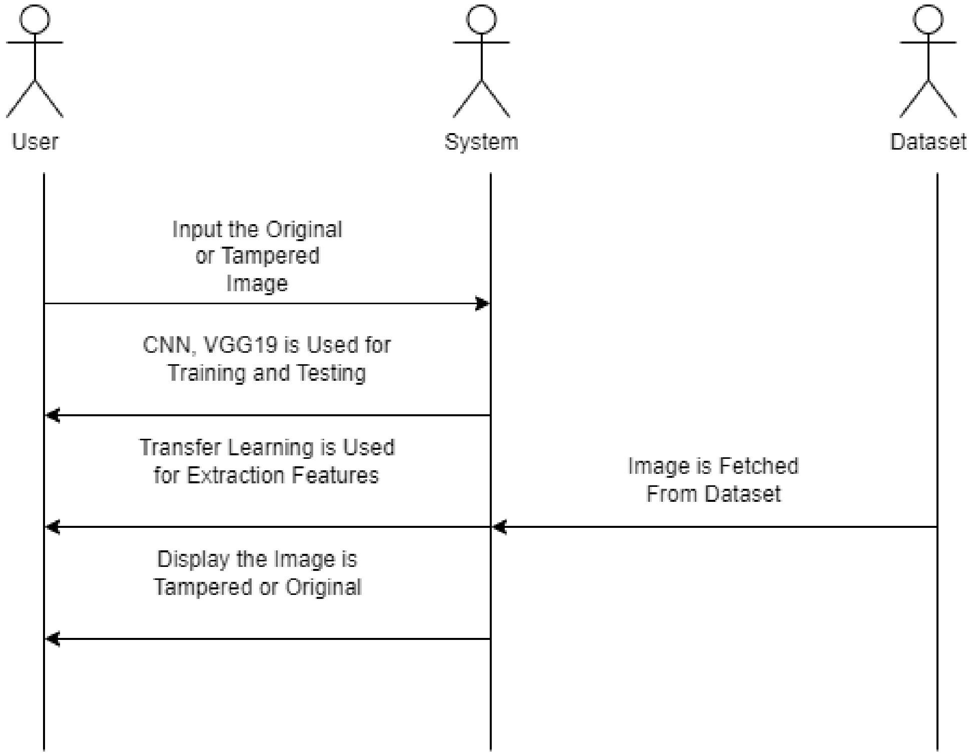


Image forensic
Input
Model
Output
Input the Image
Transfer Learning and VGG19 is Used for Training and Testing
Display the Image is Original or Tampered

5.4.3 SEQUENCE DIAGRAM

A Sequence diagram shows interaction among arranged in a time sequence. It shows the objects participating in the interaction by their life lines and the messages they exchange, arranged in a time sequence. The sequence diagram has two dimensions; the vertical dimension represents time; the horizontal dimension represents different objects. The vertical line is called objects lifeline. The life line represents the objects existence during the interaction


User
System
Dataset
Input the Original or Tampered Image
CNN, VGG19 is Used for Training and Testing
Transfer Learning is Used for Extraction Features
Image is Fetched From Dataset
Display the Image is Tampered or Original

CHAPTER 6

6. SYSTEM IMPLEMENTATION

6.1 INTRODUCTION

The creation of the designed system takes place in the implementation phase. Development phase overview, preparation of implementation, computer program development, development phase report and overview. It also performs activities like writing, testing, debugging and documenting the programs. This is to review the performance of the system and to evaluate against standard or criteria. A study is conducted for measuring the performance of the system against pre-defined requirements. A system implementation is the process which involves defining how the information system should be built . Ensuring that the information system is operational and used. And also ensuring that the informationsystem meets quality standards.

STEPS

1. Initialize Application.

2. Input Dataset.

3. Select image.

4. Load Convolutional Neural Network.

5. Load VGG19, transfer learning.

6. Create a model.

7. Input an image.

8. Predict the image fake or original

System Implementation consisting the following contents.

Environment setup

CNN and transfer learning take longer time and more memory to process the images during model preparation. Therefore, the experiment is carried out on the Google Colaboratory with 118 GB drive storage, 4 GB RAM, and run-time on-demand GPU support. Google Colaboratory notebooks are just like Jupyter notebooks but hosted on the cloud rather than the local machine. These notebooks use python version 3.6.9, and libraries such as Keras and TensorFlow are employed for image pre-processing and execution of CNN, VGG19. Thetime a model takes to learn depends on the number of layers involved, and thus, runtime GPUsupport is beneficial as it further expedites the execution time of these models. All data is uploaded on Google Drive and accessed in notebooks by mounting Google Drive using python library available for this purpose. Image pre-processing, transformations and, upsampling are done using Keras libraries in python

Data Handling

All images are read and loaded into the directories from Google Drive. Different directories were made according to train, test and validation data belonging to normal and abnormal classes. Furthermore, images were split into two folders namely, Training and Testing, containing images of both the categories. Image data Generator class from the Keras library is further used for up-sampling the datasets during run-time. Various parameters are used and modified to improve the overall model performance.

6.2 ARCHITECTURE USED

All the libraries and packages required to implement TensorFlow using Keras were imported into the Colab notebooks and uninstalled packages were installed using pip command. After setting up the environment with all needed libraries and the mounted drive containing pre-processing data, it was time to build the architecture of the different models.

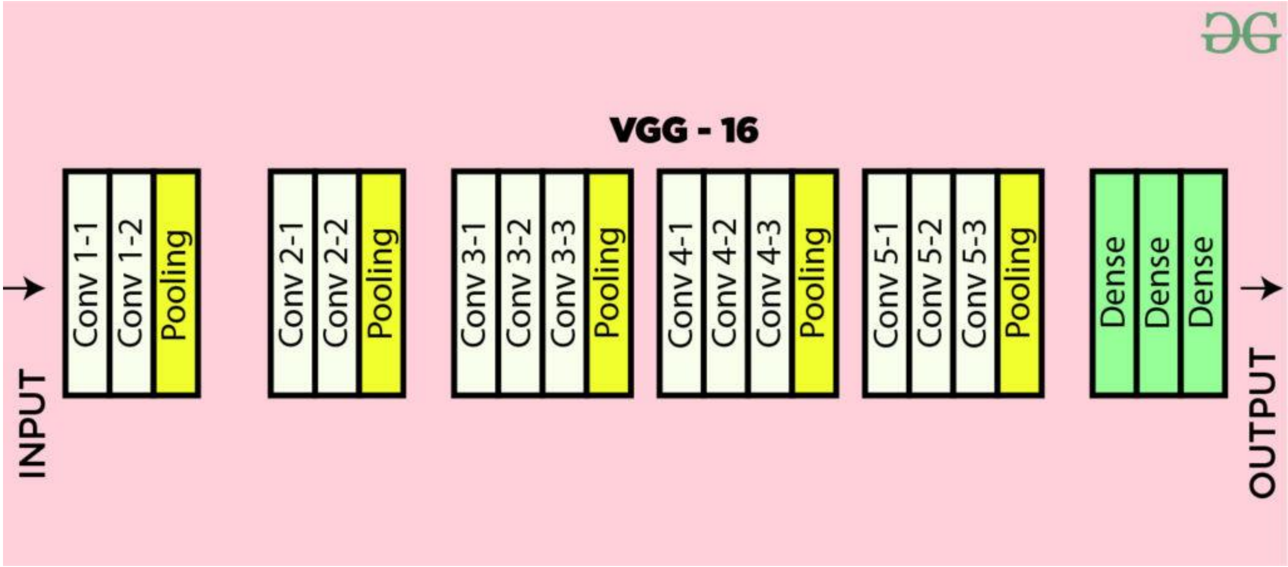
6.2.1 CONVOLUTION NEURAL NETWORK

The model is the Convolution Neural Network based on deep learning. The architecture is built using 3 convolution layers of filter size of 16, 32 and 64 with kernel size (3,3). The first convolution layer requires the input shape, and the activation function used in every convolution layer is Rectified Linear Unit (ReLu). After each convolution layer, there is a Max Pooling layer with pool size (2,2) to down-sample the input representation by taking maximum value defined by pool size. Next layer is the Flatten layer that converts or flattens the input to the one-dimensional space without affecting the batch size. Then, a dense layer with 512 neuron units, activation function ReLu and L2 regularizer, is added to the sequentialmodel. Finally, the classification layer is built using Dense layer with 2 units Sigmoid activation function is added to perform the binary classification. The model is compiled with Adam optimizer.

6.2.2 VGG

The VGG model stands for the Visual Geometry Group from Oxford. VGG-16 is a convolutional neural network that is 16 layers deep. You can load a pretrained version of the network trained on more than a million images from the ImageNet database [1]. The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals.

VGG Architecture: The input to the network is an image of dimensions (224, 224, 3). The first two layers have 64 channels of a 3*3 filter size and the same padding. Then after a max pool layer of stride (2, 2), two layers have convolution layers of 128 filter size and filter size (3, 3). This is followed by a max-pooling layer of stride (2, 2) which is the same as the previous layer. Then there are 2 convolution layers of filter size (3, 3) and 256 filters. After that, there are 2 sets of 3 convolution layers and a max pool layer. Each has 512 filters of (3, 3) size with the same padding. This image is then passed to the stack of two convolution layers. In these convolution and max-pooling layers, the filters we use are of the size 3*3 instead of 11*11 in AlexNet and 7*7 in ZF-Net. In some of the layers, it also uses 1*1 pixel which is used to manipulate the number of input channels. There is a padding of 1-pixel (same padding) done after each convolution layer to prevent the spatial feature of the image.


VGG - 16
INPUT
Conv 1-1
Conv 1-2
Pooling
Conv 2-1
Conv 2-2
Pooling
Conv 3-1
Conv 3-2
Conv 3-3
Pooling
Conv 4-1
Conv 4-2
Conv 4-3
Pooling
Conv 5-1
Conv 5-2
Conv 5-3
Pooling
Dense
Dense
Dense
OUTPUT

CHAPTER 7

7. SYSTEM TESTING

7.1 INTRODUCTION

Testing involves a series of operation of a system of application under controlled conditions and subsequently evaluating the result. The controlled condition should include both normal and abnormal conditions. It is planned and monitor for each testing level. The various testing performed are unit testing, integration testing, validation testing, output testing and system testing.

7.2 TEST PROCEDURE

Testing involves a series of operation of a system of application under controlled conditions and subsequently evaluating the result. The controlled condition should include both normal and abnormal conditions. It is planned and monitor for each testing level. The various testing performed are unit testing, integration testing, validation testing, output testing and system testing.

7.2.1 UNIT TESTING

A level of the software testing process where individual units of a software are tested. The purpose is to validate that each unit of the software performs as designed. The first level of testing, unit testing, is the most micro-level of testing. It involves testing individual modules or pieces of code to make sure each part or “unit” is correct. A “unit” can be a specific piece of functionality, a program, or a particular procedure within the application. Unit testinghelps verify internal design and internal logic, internal paths, as well as error handling. The unit testing level includes a single type of testing; unit testing. Unit tests are done by the developer who wrote the code.

7.2.2 INTEGRATION TESTING

A level of the software testing process where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units. Integration testing is done after unit testing. This level tests how the units work together. Individual modules are combined and tested as a group. Its one thing if units work well on their own, but how do they perform together? Integration testing helps you determine that, and ensures your application runs efficiently. It identifies interface issues between modules. There are a few techniques that can be used for conducting integration testing:

- Big Bang Testing
- Top Down Approach
- Bottom Up Approach

Big bang testing involves testing the entire set of integrated components together simultaneously. Because everything is integrated together and being tested at one time, this approach makes it difficult to identify the root cause of problems. The top down approach starts by testing the top-most modules and gradually moving down to the lowest set of modules one-by-one. The bottom up approach starts with testing the lowest units of the application and gradually moving up one-by-one.

7.2.3 SYSTEM TESTING

This level of testing is the first level that tests the entire application as a whole. It is often done in a mirrored production environment. This level of testing is actually a series of tests whose purpose is to test the application end-to-end. In this testing process where a complete, integrated system is tested. The purpose of this test is to evaluate the systems compliance with the specified requirements. System testing is particularly important because it verifiesthe technical, functional, and business requirements of the software. System testing is the last level of testing before the user tests the application. There are dozens of types of system testing, including usability testing, regression testing and functional testing. This level of testing is typically done by the testing team and includes a combination of automated testing and manual testing.

7.2.4 ACCEPTANCE TESTING

A level of the software testing process where a system is tested for acceptability. The purpose of this test is to evaluate the system’s compliance with the business requirements andassess whether it is acceptable for delivery.

The final level of testing, acceptance testing, or UAT (user acceptance testing), determines whether or not the software is ready to be released. Let’s face it, requirements change through out the development process. It’s important that the user verifies the business needs are met before the software is released into production. Are the functional requirements met? Are the performance requirements met? These are the questions that are answered during acceptance testing level. UAT is the final say as to whether the application is ready for use in real life or not. This phase also involves change control managing requested modifications and new feature requests. Acceptance testing should be done by the business user / end-user.

7.3 TESTING TECHNIQUES

7.3.1 BLACK-BOX TESTING

The technique of testing without having any knowledge of the interior workings of the application is called black-box testing. The tester is oblivious to the system architecture and does not have access to the source code. Typically, while performing a black-box test, a tester will interact with the system's user interface by providing inputs and examining outputs without knowing how and where the inputs are worked upon.

7.3.2 WHITE-BOX TESTING

White-box testing is the detailed investigation of internal logic and structure of the code. White-box testing is also called glass testing or open-box testing. In order to perform whitebox testing on an application, a tester needs to know the internal workings of the code. The tester needs to have a look inside the source code and find out which unit/chunk of the code is behaving inappropriately.

7.3.3 GREY-BOX TESTING

Grey-box testing is a technique to test the application with having a limited knowledge of the internal workings of an application. In software testing, the phrase the more you know, the better carries a lot of weight while testing an application. Mastering the domain of a system always gives the tester an edge over someone with limited domain knowledge. Unlike black-boxtesting, where the tester only tests the application's user interface; in grey-box testing, the tester has access to design documents and the database. Having this knowledge, a tester can prepare better test data and test scenarios while making a test plan.

7.4 TEST CASE AND OUTPUT

The test case is a document that describes an input, action, or event and an expected responseto determine if a feature of an application is working correctly. A test case should contain particulars such as test case identifiers, test case, name, objectives, test conditions, input data requirements steps and expected results. This suggests the need for retesting and to discover the source of differences. The major document produced by the system analysis at the end of the system study stage. It provides complete details of the analyst's proposed solution to the problem outlined in terms of references. This is the description of the proposed new computer system in great deal; it specified how the system would do it. The system specification describes the hardware and software specification to develop the software.

7.4.1 TEST CASES

Test objectives: In order to classify real and fake reconstructed images.Test data: Images

Step No	Steps	Data	Expected Results	Actual Results
1	Select Tampered Image	Image	Display theImage is Tampered.	Process successful
2	elect Original Image	Image	Display the Image is Original	Process successful

CHAPTER 8

8. CONCLUSION AND FUTURE ENHANCEMENT

8.1 CONCLUSION

In this work, it presents a general framework for detecting two challenging forgery techniques, the copy-move and splicing. In particular, the system can detect the manipulated regions in the image. The results show that a key-point based methodbased on the SURF features can be more efficient for copy- move forgery detection than SIFT. Its main advantage is the remarkably low computational load, combined with good performance and detection of scaled or rotated objects. The quantified the performance of splicing forgery detection using SVM model with the RBF kernel, which gives outstanding results when applied on the Cr component of the image. Thefurther must serve as an initial building block to improve thesecurity of images on the web. It is believed that insights would help forensics professionals with more concrete decisions.

8.2 FUTURE ENHANCEMENT

Given more time, expanding on this work to multi-modal Deepfakes would be interesting. The existing model could be used within the context of an CNN for learning cross-frame relationships. Similar CNN-based approaches could be used to detect audio tampering, and the results of these two models could be synthesized for a complete end-to-end Deepfake video detection model.

APPENDICES

APPENDIX A

SCREENSHOTS

Home page

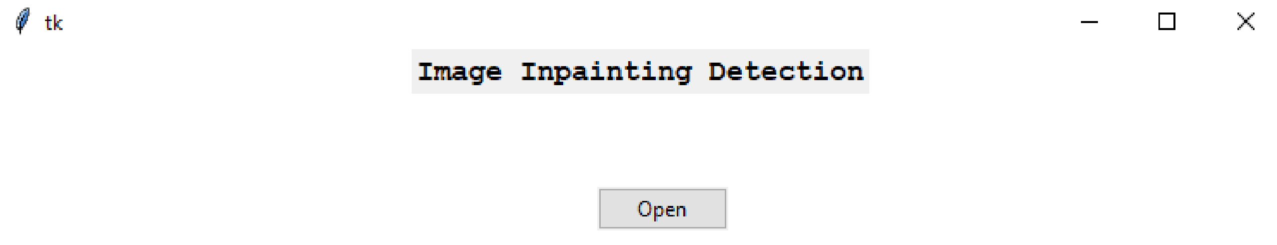


Image Inpainting Detection

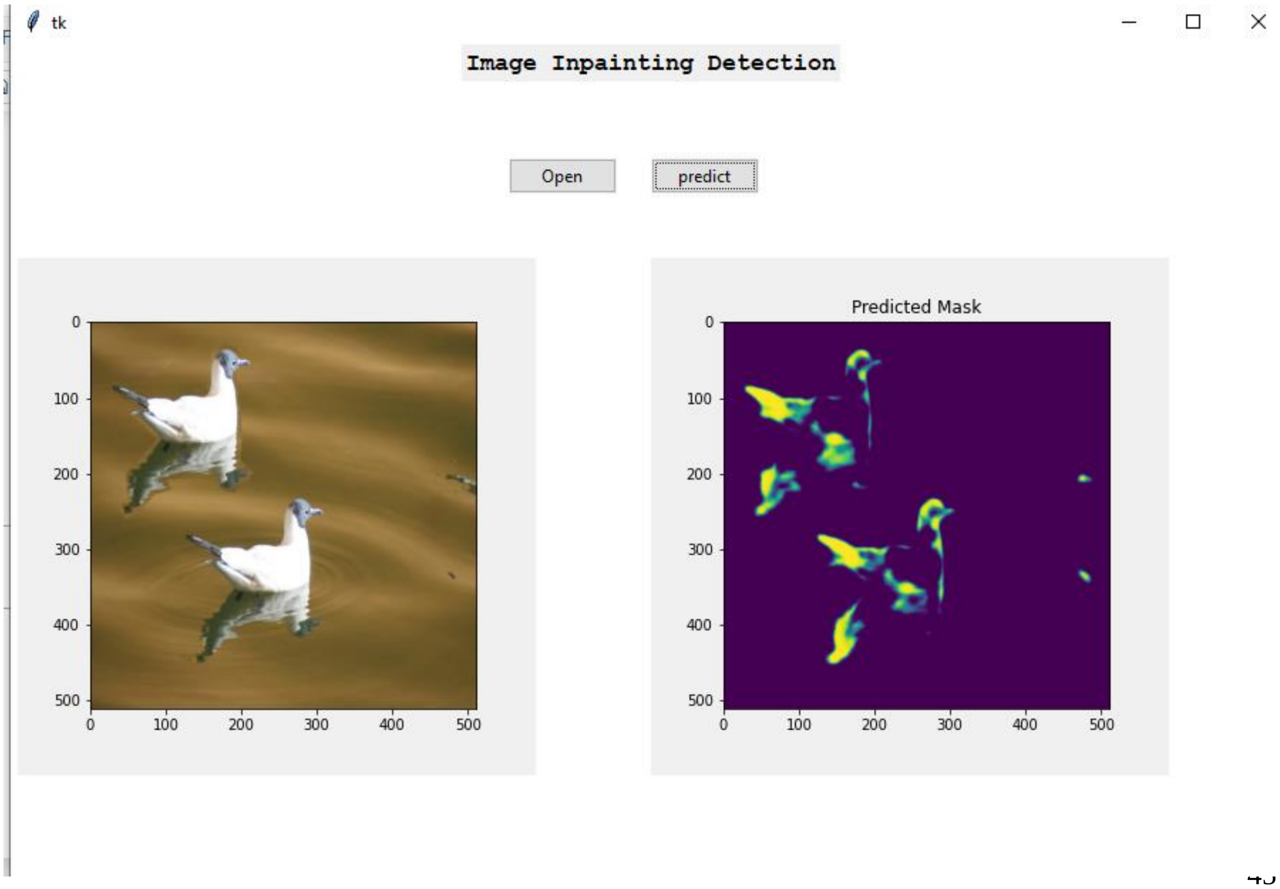
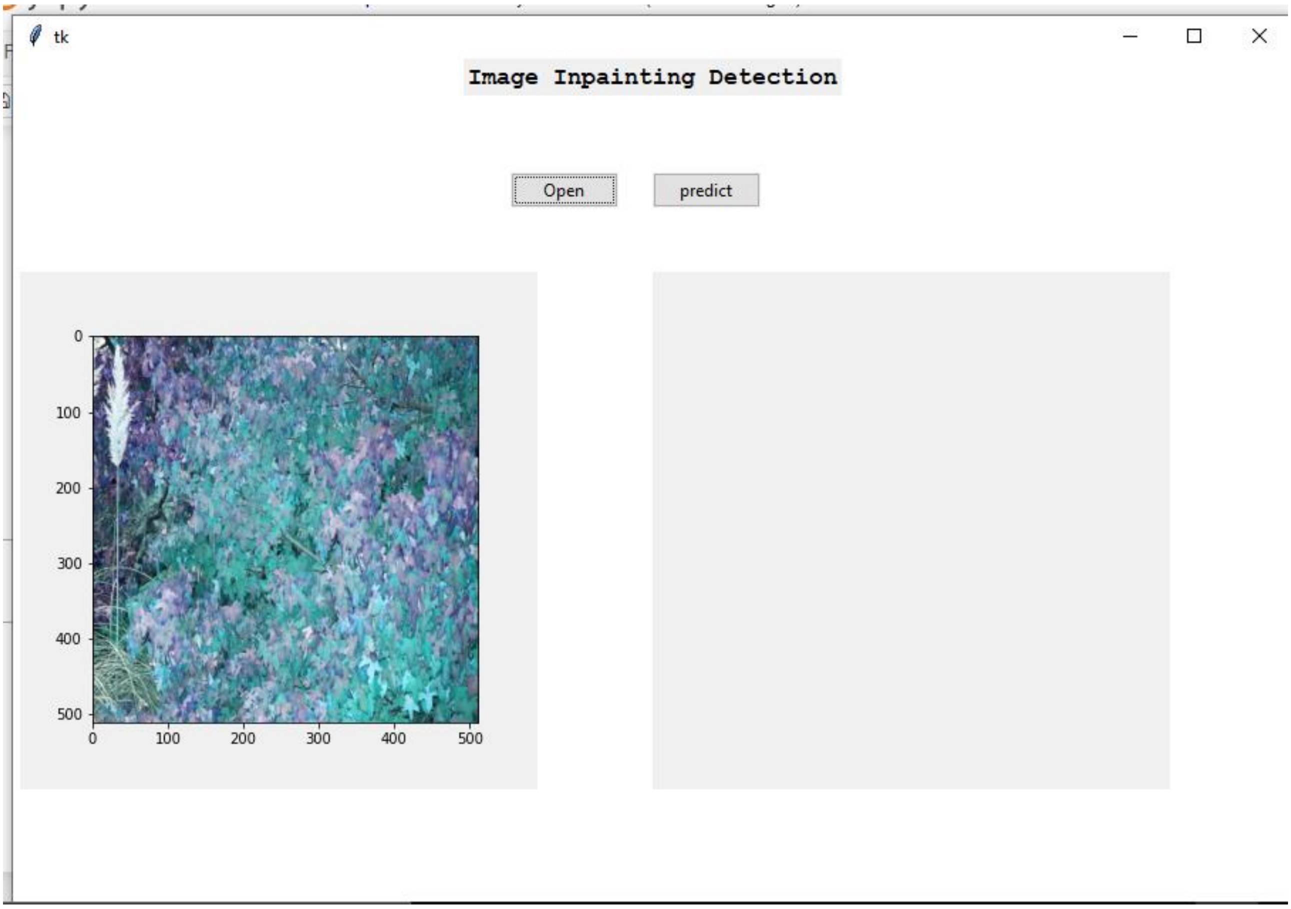


Image Inpainting Detection



tk
Image Inpainting Detection
Open
predict
0
100
200
300
400
500
0
100
200
300
400
500

REFERENCES

] William, Youssef, SherineSafwat, and Mohammed A- M. Salem. *Robust Image Forgery Detection Using PointFeature Analysis.* 2019 Federated Conference on Com- puter Science and Information Systems (FedCSIS). IEEE, 2019.

[2] Malík, Peter, Štefan Krištofík, and Kristína Knapová. *In- stance Segmentation Model Created from Three Semantic Segmentations of Mask,* Boundary and Centroid Pixels Verified on GlaS Dataset. 2020 15th Conference on Computer Science and Information Systems (FedCSIS). IEEE, 2020.

[3] Al-Berry, M. N., et al. Directional Multi-Scale Stationary Wavelet-Based Representation for Human Action Classification. *Handbook of Research on Machine Learning Innovations and Trends.* IGI Global, 2017. 295-319.

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

[5] Kuznetsov, Andrey, and Vladislav Myasnikov. A new copy-move forgery detection algorithm using image pre-processing procedure. *Procedia engineering.* 201(2017),436-444.

Websites:

[1]. https://www.learnpython.org

[2]. www.codecademy.com/learn/learn-python

[3]. www.python.org

[4]. w3schools.Com