

A Mini Project Report

On

CURRENCY RECOGNITION USING IMAGE PROCESSING

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

Bachelor of Engineering

in

Computer Science and Engineering

By

VASPARI THARUN KUMAR 1608-20-733-022

Under the guidance of

Mr. V CHANDRA SEKHAR

(Assistant Professor)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Matrusri Engineering College

Accredited by NBA & NAAC

(Affiliated to Osmania University, Approved by AICTE)

Saidabad, Hyderabad-500059

(2022-2023)

Department of Computer Science and Engineering
Matrusri Engineering College

Accredited by NBA & NAAC

(Affiliated to Osmania University, Approved by AICTE)

Saidabad, Hyderabad-500059

(2022-2023)



CERTIFICATE

This is to Certify that A Mini Project report entitled “**Currency Recognition Using Image Processing**” is being submitted by Vaspari Tharun Kumar (1608-20-733-022), in partial fulfillment of the requirement of the award for the degree of Bachelor of Engineering in “Computer Science and Engineering” O.U., Hyderabad during the year 2022-2023 is a record of bonafide work carried out by him under my guidance. The results presented in this project have been verified and are found to be satisfactory.

Project Guide

H.O.D

Mr. V Chandra Sekhar

Assistant Professor,
Dept of CSE

Dr. P. Vijayapal Reddy

Professor & Head,
Dept. of CSE

External Examiner(s)

Department of Computer Science and Engineering
Matrusri Engineering College

Accredited by NBA & NAAC

(Affiliated to Osmania University, Approved by AICTE)

Saidabad, Hyderabad-500059

(2022-2023)



DECLARATION

I **Mr. Vaspari Tharun Kumar** bearing **H.T.No.1608-20-733-022** hereby certify that the minor project entitled “**CURRENCY RECOGNITION USING IMAGE PROCESSING**” is submitted in the partial fulfilment of the required for the award of the degree of **Bachelor of Engineering in Computer Science and Engineering**.

This is a record work carried out by us under the guidance of **Mr. V Chandra Sekhar**, Assistant Professor, CSE, Matrusri Engineering College, Saidabad. The results embodied in this report have not been reproduced/copied from any source. The results embodied in this report have not been submitted to any other university or institute for the award of any other degree or diploma.

V.THARUN KUMAR 1608-20-733-022

ACKNOWLEDGEMENT

It is our privilege and pleasure to express my profound sense of respect, gratitude and indebtedness to our guide Mr. V Chandra Sekhar, Assistant Professor, Department of Computer Science and Engineering, Matrusri Engineering College, for his inspiration, guidance, cogent discussion, constructive criticisms and encouragement throughout this dissertation work.

We express our sincere thanks to Project Coordinator Dr L Raghavendar Raju, Assistant Professor, Department of Computer Science and Engineering, Matrusri Engineering College, for his valuable suggestions and constant helping in completing the work.

We express our sincere gratitude to Dr. P. Vijayapal Reddy, Professor & Head of the Department of Computer Science and Engineering, Matrusri Engineering College, for his precious suggestions, motivation and co-operation.

We express our sincere thanks to Dr. D. Hanumantha Rao, Principal, Matrusri Engineering College, Saidabad, Hyderabad, for his encouragement and constant help.

We extend our sincere thanks to all the teaching and non-teaching staff of Computer Science and Engineering Department for their support and encouragement.

Last but not least, we wish to acknowledge my friends and family members for giving moral strength moral strength and helping us to complete this dissertation.

LIST OF ABBREVIATIONS

| S.NO | ABBREVIATION | EXPANSION |
|------|--------------|----------------------------------|
| 1. | DB | Database |
| 2. | ML | Machine Learning |
| 3. | CNN | Convolutional Neural Network |
| 4. | INR | Indian Rupee |
| 5. | USD | United States Dollar |
| 6. | CVIP | Computer Vision Image Processing |

ABSTRACT

We proposed an currency recognition system using image processing methodology. It mainly focuses on the recognition of currency by its image or photograph. It will help users to recognize details about currency like Currency Value, Currency Name, the value in INR, EURO and US Dollar. It works using the main characteristics of currency notes such as size, colour or printed text on it. We have considered INDIAN Rupee and US Dollar. This system works accurately and also able to quickly identify the currency notes.

According to the UN charter there are around 180 currencies around the globe. All these currencies are different in characteristics such as size, color and texture. In the era of rapidly increasing tourism all over the world, it becomes necessary to recognize each currency note correctly. Most of the local people are not familiar with other countries currency and also not familiar with the exchange rate of that currency. So, it becomes necessary to develop an automated system that helps in recognition notes easily, faster with efficiency. The limitations in the existing system is very critical to recognition and time taking process.

The proposed system is based on image processing and makes the process robust and automatic. This system is based on computer science technologies like Digital Image Processing, python and also a small step to implement in a system that is most important for industrial development. Edge detection and CNN algorithms are used.

LIST OF FIGURES

| S.No | Fig No. | Name of the Figure | Page No |
|-------------|----------------|-----------------------------------|----------------|
| 1 | 4.2.1 | System Architecture | 9 |
| 2 | 4.3.1 | UML Hierarchy Diagrams | 11 |
| 3 | 4.3.1.1 | Class Diagram | 12 |
| 4 | 4.3.2.1 | Use Case Diagram | 13 |
| 5 | 4.3.3.1 | Component diagram | 14 |
| 6 | 4.3.4.1 | Sequence Diagram | 15 |
| 7 | 4.3.5.1 | Activity Diagram | 16 |
| 8 | 5.3.1 | 5 Rupee note | 26 |
| 9 | 5.3.2 | 500 Rupee note | 27 |
| 10 | 5.3.3 | 10 \$(dollar) | 27 |
| 11 | 5.3.4 | 20 \$(dollar) | 27 |
| 12 | 7.1 | Upload Currency Image | 38 |
| 13 | 7.2 | Select & Load Image | 39 |
| 14 | 7.3 | Enter Exchange Rates | 40 |
| 15 | 7.4 | Displays Original & Trained Image | 41 |
| 16 | 7.5 | Image Recognized | 42 |

LIST OF TABLES

| S.No | Table No. | Name of the Table | PageNo |
|-------------|------------------|--------------------------|---------------|
| 1 | 2.1 | Literature Survey | 5 |
| 2 | 6.3.1 | Home page | 34 |
| 3 | 6.3.2 | Upload Image | 35 |
| 4 | 6.3.3 | Recognize | 36 |
| 5 | 6.3.4 | Exit | 37 |

CONTENTS

| | |
|---|----------|
| Declaration | iii |
| Acknowledgement | iv |
| List of Abbreviations..... | v |
| Abstract..... | vi |
| List of Figures. | vii |
| List of Tables | viii |
| | |
| 1. INTRODUCTION..... | 1 |
| 1.1 MOTIVATION..... | 1 |
| 1.2 PROBLEM STATEMENT..... | 1 |
| 1.3 PROJECT OBJECTIVE..... | 2 |
| | |
| 2. LITERATURE SURVEY..... | 3 |
| 2.1 EXISTING SYSTEM..... | 5 |
| 2.2 LIMITATIONS OF EXISITNG SYSTEM..... | 5 |
| | |
| 3. SOFTWARE REQUIREMENT SPECIFICATION..... | 6 |
| 3.1 SOFTWARE REQUIREMENTS..... | 6 |
| 3.2 HARDWARE REQUIREMENTS..... | 6 |
| 3.3 FUNCTIONAL REQUIREMENTS | 6 |
| 3.4 NON-FUNCTIONAL REQUIREMENTS..... | 7 |
| | |
| 4. SYSTEM DESIGN..... | 8 |
| 4.1 SYSTEM DESIGN..... | 8 |
| 4.2 SYSTEM ARCHITECTURE..... | 9 |
| 4.3 UML DESIGN..... | 9 |
| 4.3.1 CLASS DIAGRAM..... | 12 |
| 4.3.2 USECASE DIAGRAM..... | 13 |

| | |
|---------------------------------|-----------|
| 4.3.3 COMPONENT DIAGRAM..... | 14 |
| 4.3.4 SEQUENCE DIAGRAM..... | 15 |
| 4.3.5 ACTIVITY DIAGRAM..... | 16 |
| 4.4 TECHNOLOGY DESCRIPTION..... | 17 |
| 5. IMPLEMENTATION..... | 24 |
| 5.1 LIBRARIES..... | 24 |
| 5.2 MODULES..... | 25 |
| 5.3 DATASET | 26 |
| 5.4 EXECUTABLE CODE..... | 28 |
| 6. TESTING..... | 32 |
| 6.1 TESTING DEFINATION..... | 32 |
| 6.2 UNIT TESTING..... | 32 |
| 6.3 TEST CASES..... | 33 |
| 7.RESULTS..... | 38 |
| 8.CONCLUSION..... | 43 |
| 8.1 CONCLUSION..... | 43 |
| 8.2 FUTURE SCOPE..... | 43 |
| REFERENCES..... | 44 |

INTRODUCTION

1.1 MOTIVATION

According to the UN charter there are around 195 countries around the globe. In which 193 countries are members of the UN and two are observing states. According to The U.N., worldwide there are 180 currencies. All these currencies are different in characteristics such as size, colour and texture. In the era of rapidly growing levels of trade between countries and also tourism all over the world, it becomes necessary to recognize each currency note correctly. Now a days people travel to different countries, they use their native country currency in paying bills or buying stuffs and because most of the local people are not familiar with the currency other than their own country currency and also not familiar with the exchange rate of that currency in their own currency, it becomes necessary to develop an automated system that helps in recognition notes easily, faster with efficiency.

We used INR and USD as an example to illustrate the technique. This system is based on our knowledge about computer science technologies like Digital Image Processing, python and also a small step to implement in a system that is most important for industrial development.

We used INR and USD as an example to illustrate the technique. This system is based on our knowledge about computer science technologies like Digital Image Processing, python and also a small step to implement in a system that is most important for industrial development.

1.2 PROBLEM STATEMENT

According to the UN charter there are around 195 countries around the globe. In which 193 countries are members of the UN and two are observing states. According to The U.N., worldwide there are 180 currencies. All these currencies are different in characteristics such as size, colour and texture. In the era of rapidly growing levels of trade between countries and also tourism all over the world, it becomes necessary to recognize each currency note correctly. Now a days people travel to different countries,

they use their native country currency in paying bills or buying stuffs and because most of the local people are not familiar with the currency other than their own country currency and also not familiar with the exchange rate of that currency in their own currency, it becomes necessary to develop an automated system that helps in recognition notes easily, faster with efficiency. The proposed system is based on image processing and makes the process robust and automatic.

1.3 PROJECT OBJECTIVE

First of all input a currency note image, which needs to be checked. Then system performs basic image processing techniques on input image and refined to convert it into a usable input for matching with present image in dataset. The system extracts the information from image based on features such as colour and texture. With the help of these features the system determines currency name and denomination of the currency note. After this the exchange rate of currency will be extracted from internet with the help of online exchange rate apiurl. At the end output displayed on the screen.

According to the survey conducted by the CIA [7], there are around 180+ currencies presently circulating in the world. Each of these currencies differs greatly in features such as size, color and texture. Unlike the olden times, the trade and commerce between countries has increased in all sorts of levels. The need for acquiring knowledge about all the currencies by the banks has been extremely important.

However for any human teller to recognize each note correctly is something that is not feasible. Thus the need for an efficient automated system that helps in recognizing notes is pivotal for the future. In this paper, we propose an automated system for currency recognition using Image processing techniques. Our system works for 20 of the most commonly used currencies. The nature of each image is firstly refined to convert it into a usable input to extract various pictorial information in the pre-processing phase. The system then extracts the region of interest (ROI) based on features such as size, color and text. Using these regions of interest, the system first determines the country of origin of the currency note. After this phase, the denomination of the currency note would be identified using.

LITERATURE SURVEY

[1] DIGITAL IMAGE PROCESSING 4th EDITION

The 4th Edition, which celebrates the book's 40th anniversary, is based on an extensive survey of faculty, students, and independent readers in 150 institutions from 30 countries. Their feedback led to expanded or new coverage of topics such as deep learning and deep neural networks, including convolutional neural nets, the scale-invariant feature transform (SIFT), maximally-stable extremal regions (MSERs), graph cuts, k-means clustering and superpixels, active contours (snakes and level sets), and exact histogram matching. Major improvements were made in reorganizing the material on image transforms into a more cohesive presentation, and in the discussion of spatial kernels and spatial filtering.

[2] IDENTIFICATION OF PAPER CURRENCY TECHNIQUES

A SURVEY: Paper currency identification is an image processing technique i.e. worn to recognize currency of different countries. The paper currencies of different countries are possibly interweaved collectively consequently rises ever more. It is a challenge for standard currency recognition systems. However, the main focus of most of the standard currency recognition systems and machines is on recognizing forged currencies. Hence there is very vital role of currency identification system and it is essential that the identification system should be very accurate. A thriving approach for paper currency identification depends upon preprocessing, feature extraction and classification of that currency image. In this paper, we have gone through different literature which describes different techniques of paper currency recognition. Finally we have concluded that if we apply some efficient pre-processing and feature extraction technique than we can improve the accuracy of identification system.

2.3. RECOGNITION SYSTEM FOR REAL TIME PAPER CURRENCY: Money number recognition refers to the money of the currency, the currency and authenticity recognition. Money number recognition system is the kernel module of self-service system, and the major applied range is cash-related equipments. In this paper we design a kind of money number recognition system. The quick positioning of money number

is achieved based on gray value accumulation. The edge line of money number area is detected using the least square method. Using geometrical rotation method and gray adjacent interpolation method to realize the number of tilt correction. Based on the character structure characteristic and the imaginary line and character of the point of intersection features, formation recognition judgment tree, realized the character recognition. The simulation experiment indicates that this algorithm has a high recognition accuracy under the condition of rejection.

[3] PAPER CURRENCY RECOGNITION

Paper currency recognition (PCR) is an important area of pattern recognition. A system for the recognition of paper currency is one kind of intelligent system which is a very important need of the current automation systems in the modern world of today. It has various potential applications including electronic banking, currency monitoring systems, money exchange machines, etc. This paper proposes an automatic paper currency recognition system for paper currency. A method of recognizing paper currencies has been introduced. This is based on interesting features and correlation between images. It uses Radial Basis Function Network for classification. The method uses the case of Saudi Arabian paper currency as a model. The method is quite reasonable in terms of accuracy. The system deals with 110 images, 10 of which are tilted with an angle less than 15°. The rest of the currency images consist of mixed including noisy and normal images 50 each. It uses fourth series (1984–2007) of currency issued by Saudi Arabian Monetary Agency (SAMA) as a model currency under consideration. The system produces accuracy of recognition as 95.37%, 91.65%, and 87.5%, for the Normal Non-Tilted Images, Noisy Non-Tilted Images, and Tilted Images respectively. The overall Average Recognition Rate for the data of 110 images is computed as 91.51%. The proposed algorithm is fully automatic and requires no human intervention. The proposed technique produces quite satisfactory results in terms of recognition and efficiency.

Table- 2.1 Literature Survey

| Year | Authors | Title | Methodology | Features | Disadvantages |
|------|--------------------|----------------------------|---|---|--|
| 2010 | K.K. Debnath et al | Currency Recognition | Ensemble Neural Network(ENN) with Negative Correlation Learning | Conversion of note image into greyscale, compression of image, ability to identify currency with noise and old notes. | Time taking Process |
| 2010 | J. Guo et al | Paper Currency Recognition | Local Binary Pattern (LBP) algorithm | Extraction of currency features, efficient recognition of currency notes, simplicity, and high speed. | Independent training leading to fewer chances of misclassification |

2.1 Existing System

Now a days people travel to different countries, they use their native country currency in paying bills or buying stuffs and because most of the local people are not familiar with the currency other than their own country currency and also not familiar with the exchange rate of that currency in their own currency, it becomes necessary to develop an automated system that helps in recognition notes easily, faster with efficiency.

2.2 Limitations of Existing System

- 1.very critical to recogition
- 2.time taken process

SOFTWARE REQUIREMENT SPECIFICATION

This chapter gives an overview of the software and hardware components required for our project.

3.1 SOFTWARE REQUIREMENTS

Operating System : Windows 8
Coding Language : Python 3.7

3.2 HARDWARE REQUIREMENTS

System : Minimum i3
Hard Disk : 40GB+
Ram : 4GB

3.3 FUNCTIONAL REQUIREMENTS

These are the requirements that the end user specifically demands as basic facilities that the system should offer. All these functionalities need to be necessarily incorporated into the system as a part of the contract. These are represented or stated in the form of input to be given to the system, the operation performed and the output expected. They are basically the requirements stated by the user which one can see directly in the final product, unlike the non-functional requirements.

- Data Collection
- Handling Missing Values
- Remove Duplicate rows
- Data Cleaning
- Data Labeling
- Data Scaling
- Finding Feature Importance
- Data Visualization
- Model Creating
- Model Training

- Hyper parameter tuning
- Model boosting and bagging
- Model Evaluation
- Model Deployment

3.4NON-FUNCTIONAL REQUIREMENTS

These are basically the quality constraints that the system must satisfy according to the project contract. The priority or extent to which these factors are implemented varies from one project to other. They are also called non-behavioral requirements.

They basically deal with issues like:

- Portability
- Security
- Maintainability
- Reliability
- Scalability
- Performance
- Reusability
- Flexibility

SYSTEM DESIGN

4.1 System Design

In this phase, the system and software design documents are prepared as per the requirement specification document. This helps define overall system architecture.

There are two kinds of design documents developed in this phase:

High-Level Design (HLD)

- Brief description and name of each module
- An outline about the functionality of every module
- Interface relationship and dependencies between modules
- Database tables identified along with their key elements
- Complete architecture diagrams along with technology details

Low-Level Design(LLD)

- Functional logic of the modules
- Database tables, which include type and size
- Complete detail of the interface
- Addresses all types of dependency issues
- Listing of error messages
- Complete input and outputs for every module

4.2 SYSTEM ARCHITECTURE:

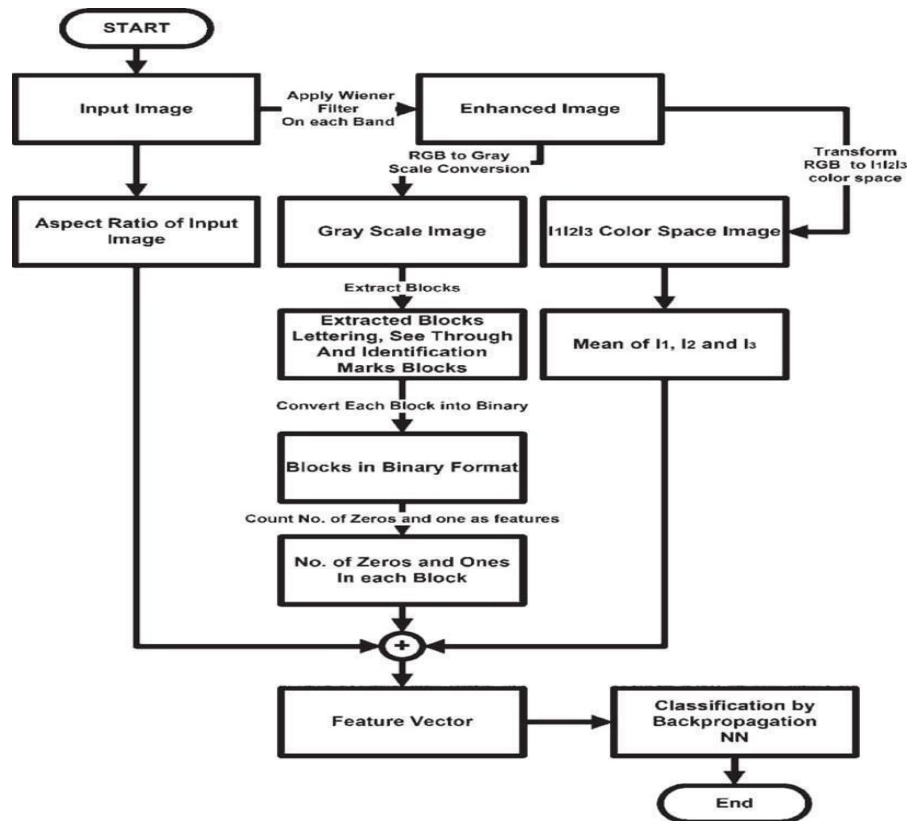


Figure-4.2.1 System Architecture

4.3 UML Design:

Unified Modeling Language (UML) is a general purpose modeling language. The main aim of UML is to define a standard way to visualize the way a system has been designed. It is quite similar to blueprints used in other fields of engineering.

UML is not a programming language; it is rather a visual language. We use UML diagrams to portray the behavior and structure of a system, UML helps software engineers, businessmen and system architects with modeling, design and analysis. The Object Management Group (OMG) adopted Unified Modeling Language as a standard in 1997. It's been managed by OMG ever since. International Organization

for Standardization (ISO) published UML as an approved standard in 2005. UML has been revised over the years and is reviewed periodically.

Do we really need UML?

- Complex applications need collaboration and planning from multiple teams and hence require a clear and concise way to communicate amongst them.
- Businessmen do not understand code. So UML becomes essential to communicate with non programmer's essential requirements, functionalities and processes of the system.
- A lot of time is saved down the line when teams are able to visualize processes, user interactions and static structure of the system.
- UML is linked with object oriented design and analysis. UML makes the use of elements and forms associations between them to form diagrams. Diagrams in UML can be broadly classified as:

The Primary goals in the design of the UML are as follows:

- Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
- Provide extendibility and specialization mechanisms to extend the core concepts.
- Be independent of particular programming languages and development process.
- Provide a formal basis for understanding the modeling language.
- Encourage the growth of OO tools market.
- Support higher level development concepts such as collaborations, frameworks, patterns and components.
- Integrate best practices.

Types of UML Diagrams:

Structural Diagrams:

Capture static aspects or structure of a system. Structural Diagrams include: Component Diagrams, Object Diagrams, Class Diagrams and Deployment Diagrams.

Behavior Diagrams:

Capture dynamic aspects or behavior of the system. Behavior diagrams include: Use Case Diagrams, State Diagrams, Activity Diagrams and Interaction Diagrams.

The image below shows the hierarchy of diagrams according to UML

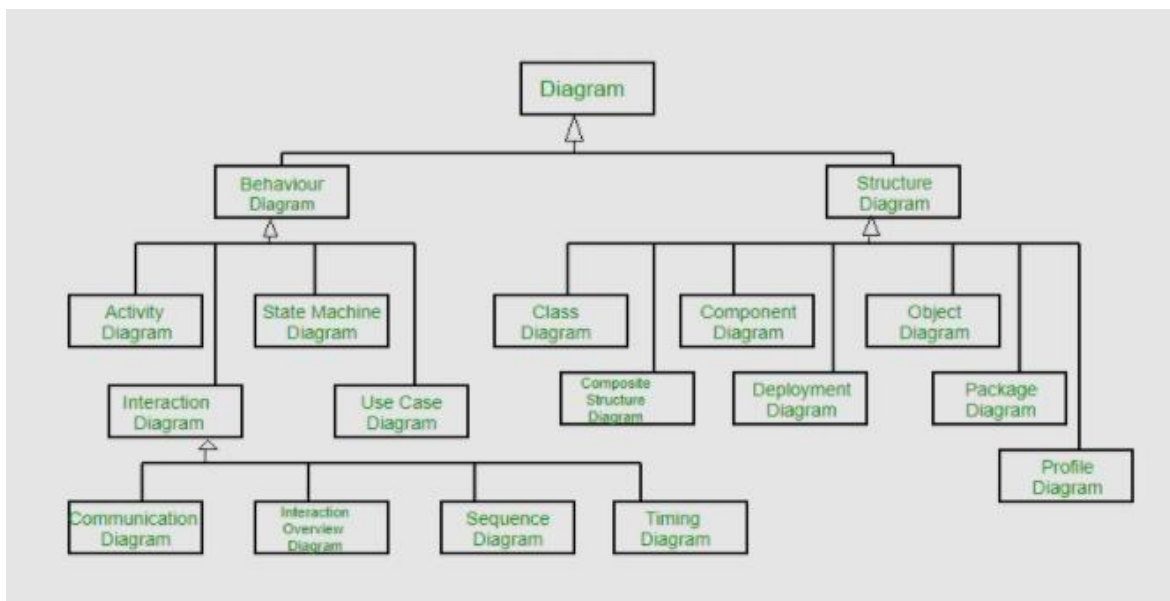


Figure-4.2.1 UML Hierarchy diagrams

4.3.1 CLASS DIAGRAM:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

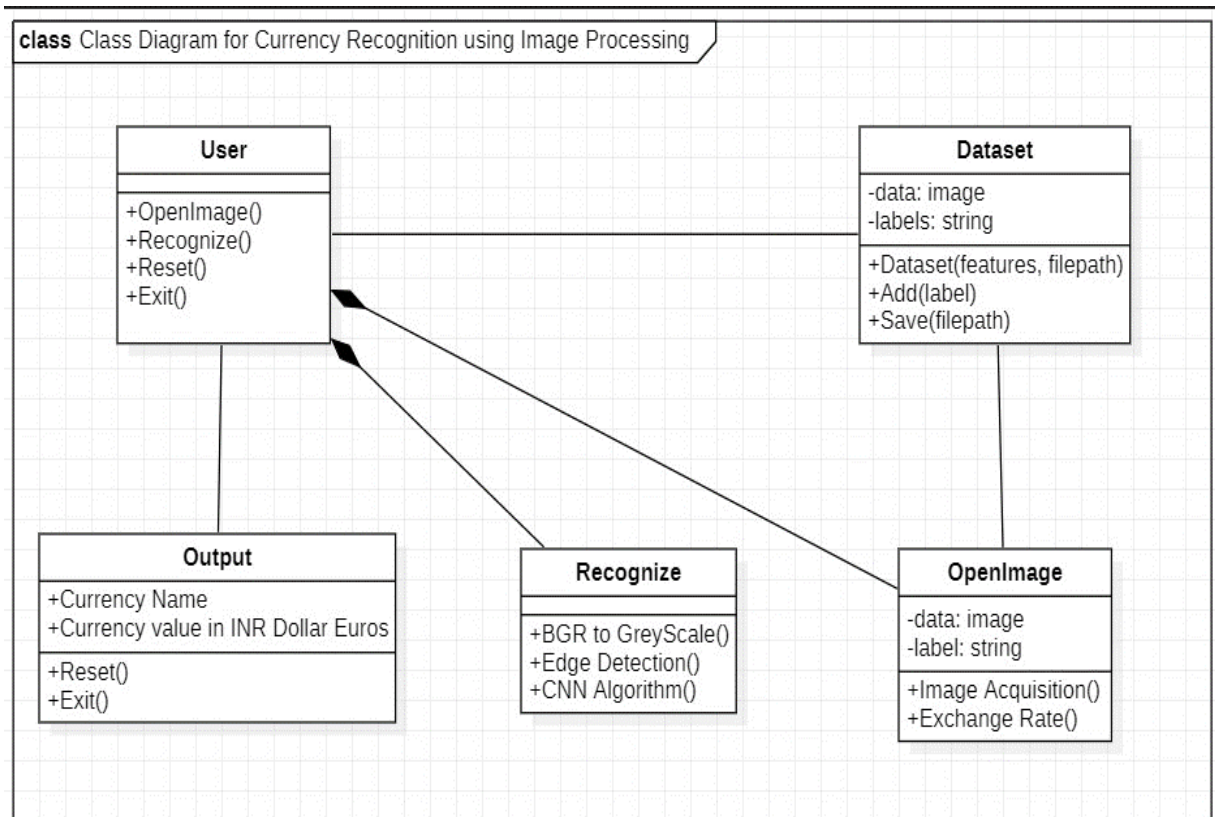


Figure-4.3.1.1 Class Diagram

4.3.2 USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted

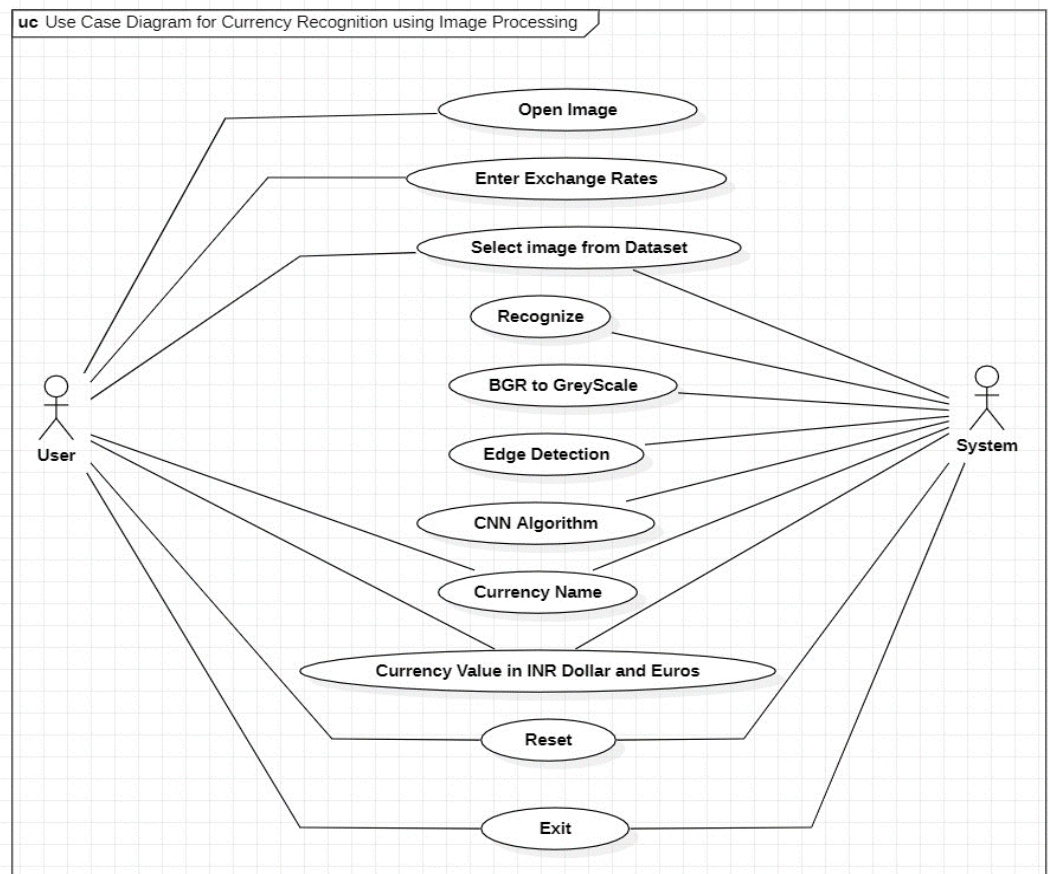


Figure-4.3.2.1 Use Case Diagram

4.3.3 COMPONENT DIAGRAM:

A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical components in a system. Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required functions is covered by planned development.

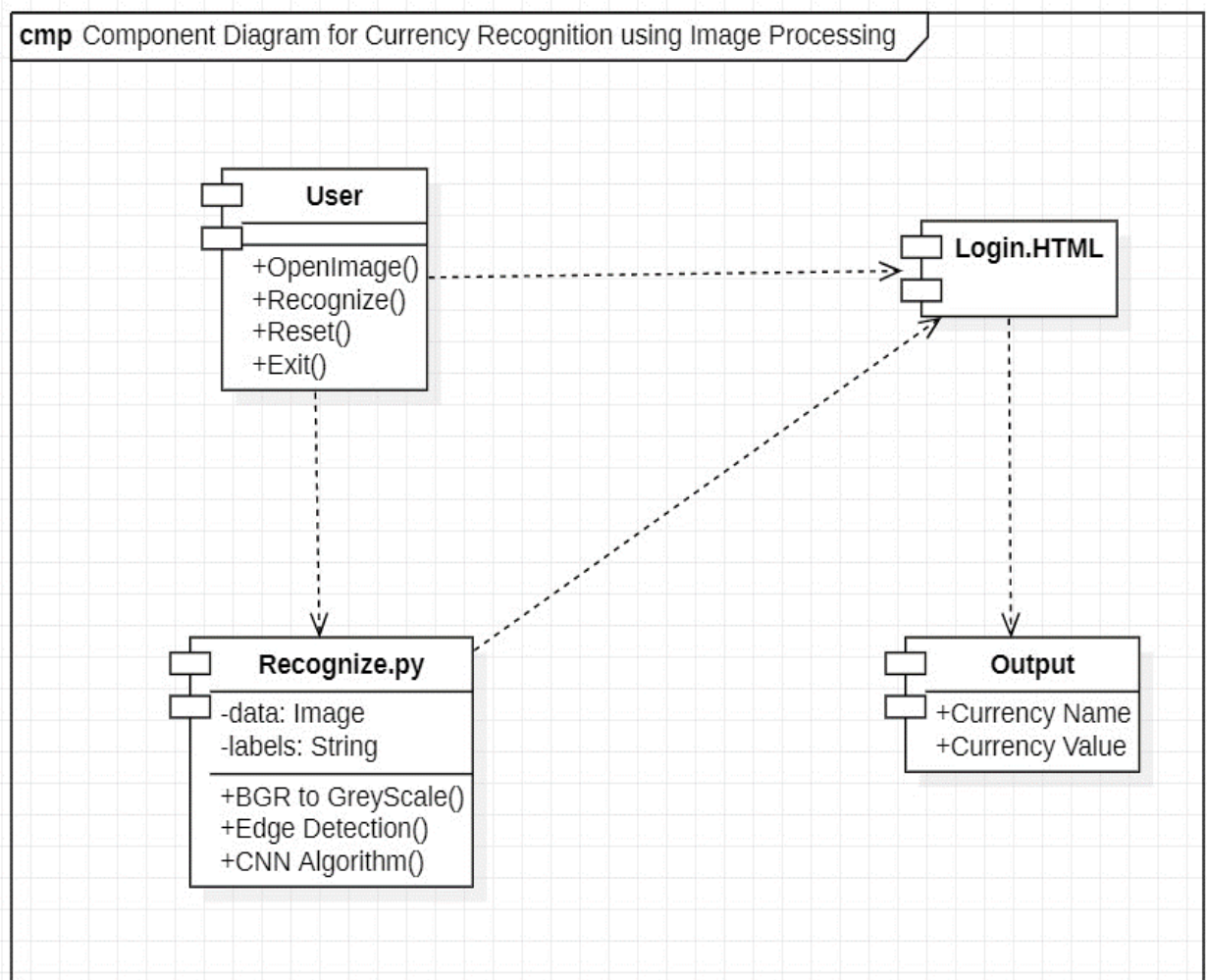


Figure-4.3.3.1 Component Diagram

4.3.4 SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

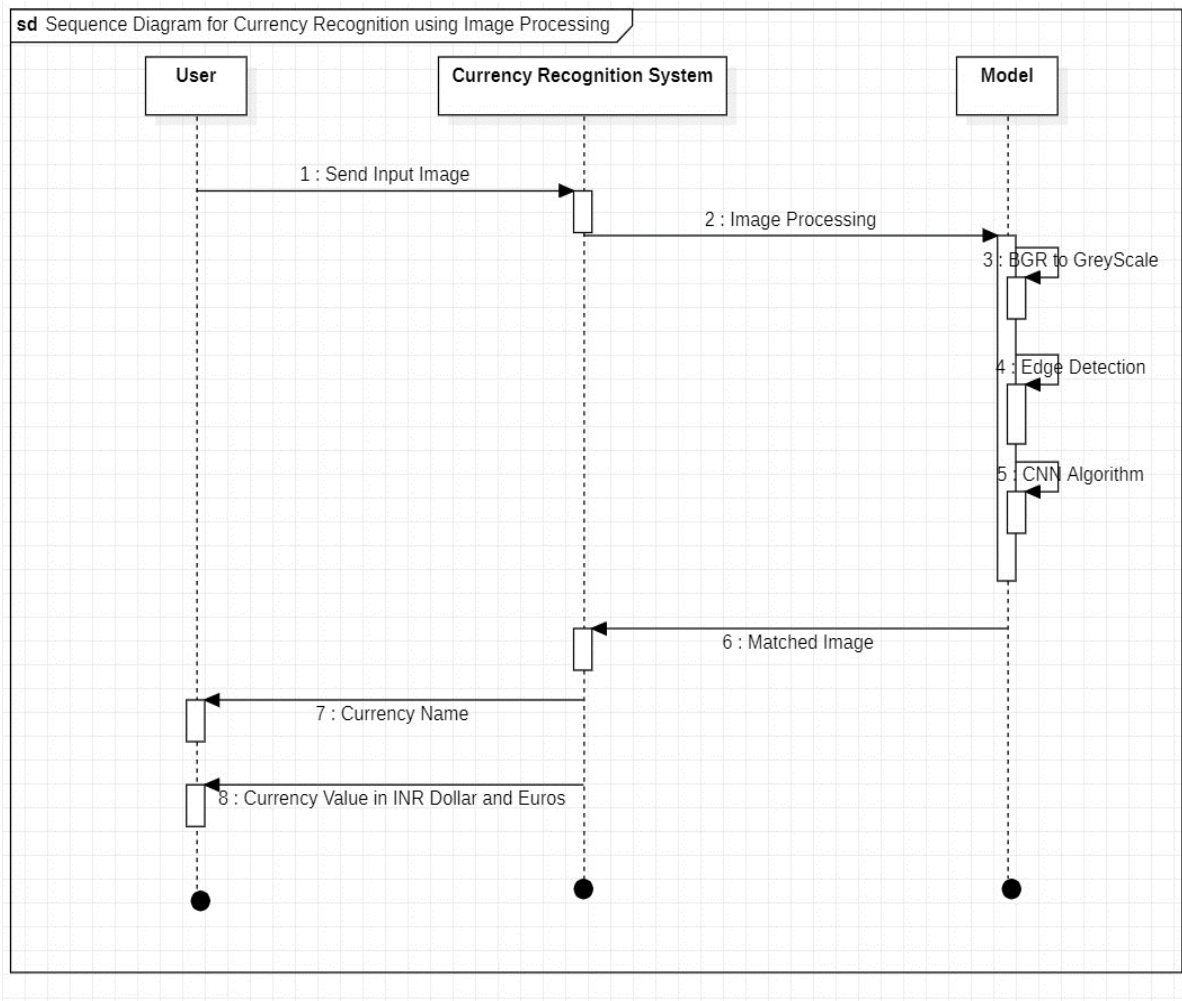


Figure-4.3.4.1 Sequence Diagram

4.3.5 ACTIVITY DIAGRAM:

In UML, an activity diagram is used to display the sequence of activities. Activity diagrams show the workflow from a start point to the finish point detailing the many decision paths that exist in the progression of events contained in the activity.

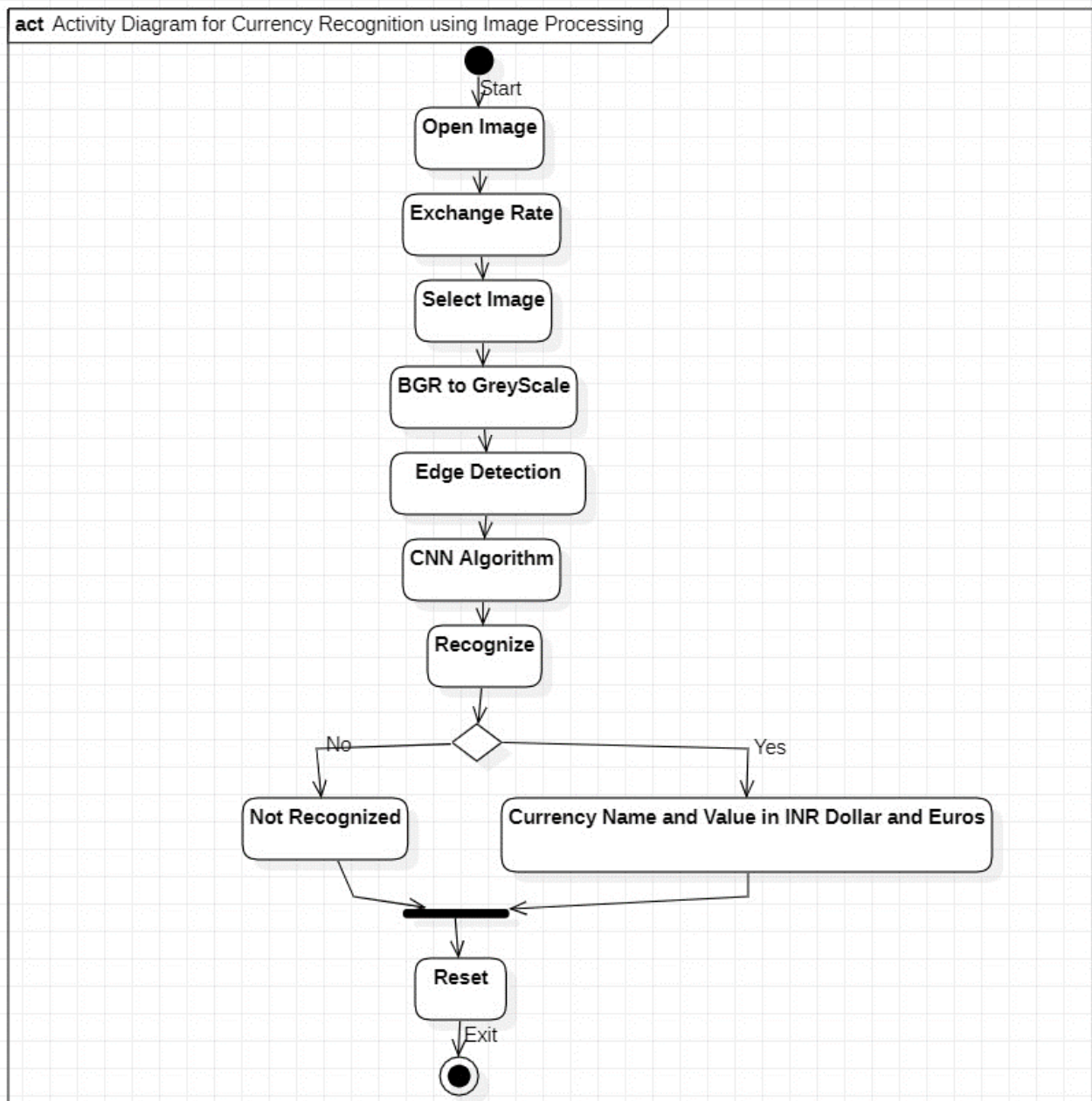


Figure-4.3.5.1 Activity Diagram

4.4 TECHNOLOGY DESCRIPTION

4.4.1 Python :-

Below are some facts about Python.

Python is currently the most widely used multi-purpose, high-level programming language.

Python allows programming in Object-Oriented and Procedural paradigms. Python programs generally are smaller than other programming languages like Java.

Programmers have to type relatively less and indentation requirement of the language, makes them readable all the time.

Python language is being used by almost all tech-giant companies like – Google, Amazon, Facebook,

Instagram, Dropbox, Uber... etc.

The biggest strength of Python is huge collection of standard library which can be used for the following

- MachineLearning
- GUI Applications (like Kivy, Tkinter, PyQt etc.)
- Web frameworks like Django (used by YouTube, Instagram, Dropbox)
- Image processing (like Opencv, Pillow)
- Web scraping (like Scrapy, BeautifulSoup, Selenium)
- Test frameworks
- Multimedia

4.4.2 What is Machine Learning : -

Before we take a look at the details of various machine learning methods, let's start by looking at what machine learning is, and what it isn't. Machine learning is often categorized as a subfield of artificial intelligence, but I find that categorization can often be misleading at first brush. The study of machine learning certainly arose from research in this context, but in the data science application of machine learning methods, it's more helpful to think of machine learning as a means of building models of data.

Fundamentally, machine learning involves building mathematical models to help understand data. "Learning" enters the fray when we give these models tunable parameters that can be adapted to observed data; in this way the program can be considered to be "learning" from the data. Once these models have been fit to previously seen data, they can be used to predict and understand aspects of newly observed data. I'll leave to the reader the more philosophical digression regarding the extent to which this type of mathematical, model-based "learning" is similar to the "learning" exhibited by the human brain. Understanding the problem setting in machine learning is essential to using these tools effectively, and so we will start with some broad categorizations of the types of approaches we'll discuss here.

Categories Of Machine Learning :-

At the most fundamental level, machine learning can be categorized into two main types: supervised learning and unsupervised learning.

Supervised learning involves somehow modeling the relationship between measured features of data and some label associated with the data; once this model is determined, it can be used to apply labels to new, unknown data. This is further subdivided into classification tasks and regression tasks: in classification, the labels

are discrete categories, while in regression, the labels are continuous quantities. We will see examples of both types of supervised learning in the following section.

Unsupervised learning involves modeling the features of a dataset without reference to any label, and is often described as "letting the dataset speak for itself." These models include tasks such as clustering and dimensionality reduction. Clustering algorithms identify distinct groups of data, while dimensionality reduction algorithms search for more succinct representations of the data. We will see examples of both types of unsupervised learning in the following section.

Need for Machine Learning

Human beings, at this moment, are the most intelligent and advanced species on earth because they can think, evaluate and solve complex problems. On the other side, AI is still in its initial stage and haven't surpassed human intelligence in many aspects. Then the question is that what is the need to make machine learn? The most suitable reason for doing this is, "to make decisions, based on data, with efficiency and scale".

Lately, organizations are investing heavily in newer technologies like Artificial Intelligence, Machine Learning and Deep Learning to get the key information from data to perform several real-world tasks and solve problems. We can call it data-driven decisions taken by machines, particularly to automate the process. These data-driven decisions can be used, instead of using programming logic, in the problems that cannot be programmed inherently. The fact is that we can't do without human intelligence, but other aspect is that we all need to solve real-world problems with efficiency at a huge scale. That is why the need for machine learning arises.

Challenges in Machine Learning :-

While Machine Learning is rapidly evolving, making significant strides with cybersecurity and autonomous cars, this segment of AI as whole still has a long way to

go. The reason behind is that ML has not been able to overcome number of challenges. The challenges that ML is facing currently are –

Quality of data – Having good-quality data for ML algorithms is one of the biggest challenges. Use of low-quality data leads to the problems related to data preprocessing and feature extraction.

Time-Consuming task – Another challenge faced by ML models is the consumption of time especially for data acquisition, feature extraction and retrieval.

Lack of specialist persons – As ML technology is still in its infancy stage, availability of expert resources is a tough job.

No clear objective for formulating business problems – Having no clear objective and well-defined goal for business problems is another key challenge for ML because this technology is not that mature yet.

Issue of overfitting & underfitting – If the model is overfitting or underfitting, it cannot be represented well for the problem.

Curse of dimensionality – Another challenge ML model faces is too many features of data points. This can be a real hindrance.

Difficulty in deployment – Complexity of the ML model makes it quite difficult to be deployed in real life.

Applications of Machines Learning :-

Machine Learning is the most rapidly growing technology and according to researchers we are in the golden year of AI and ML. It is used to solve many real-world complex problems which cannot be solved with traditional approach. Following are some real-world applications of ML –

- o Emotion analysis
- o Sentiment analysis

- o Weather forecasting and prediction
- o Stock market analysis and forecasting
- o Speech synthesis
- o Speech recognition
- o Customer segmentation
- o Object recognition
- o Fraud detection
- o Fraud prevention
- o Recommendation of products to customer in online shopping.

Types of Machine Learning

- **Supervised Learning** – This involves learning from a training dataset with labeled data using classification and regression models. This learning process continues until the required level of performance is achieved.
- **Unsupervised Learning** – This involves using unlabelled data and then finding the underlying structure in the data in order to learn more and more about the data itself using factor and cluster analysis models.
- **Semi-supervised Learning** – This involves using unlabelled data like Unsupervised Learning with a small amount of labeled data. Using labeled data vastly increases the learning accuracy and is also more costeffective than Supervised Learning.
- **Reinforcement Learning** – This involves learning optimal actions through trial and error. So the next action is decided by learning behaviors that are based on the current state and that will maximize the reward in the future.

Advantages of Machine learning :-

1. Easily identifies trends and patterns -

Machine Learning can review large volumes of data and discover specific trends and patterns that would not be apparent to humans. For instance, for an e-commerce website like Amazon, it serves to understand the browsing behaviors and purchase histories of its users to help cater to the right products, deals, and reminders relevant to them. It uses the results to reveal relevant advertisements to them.

2. No human intervention needed (automation)

With ML, you don't need to babysit your project every step of the way. Since it means giving machines the ability to learn, it lets them make predictions and also improve the algorithms on their own. A common example of this is anti-virus softwares; they learn to filter new threats as they are recognized. ML is also good at recognizing spam.

3. Continuous Improvement

As **ML algorithms** gain experience, they keep improving in accuracy and efficiency. This lets them make better decisions. Say you need to make a weather forecast model. As the amount of data you have keeps growing, your algorithms learn to make more accurate predictions faster.

4. Handling multi-dimensional and multi-variety data

Machine Learning algorithms are good at handling data that are multi-dimensional and multi-variety, and they can do this in dynamic or uncertain environments.

5. Wide Applications

You could be an e-tailer or a healthcare provider and make ML work for you. Where it does apply, it holds the capability to help deliver a much more personal experience to customers while also targeting the right customers.

Disadvantages of Machine Learning :-

1. Data Acquisition

Machine Learning requires massive data sets to train on, and these should be inclusive/unbiased, and of good quality. There can also be times where they must wait for new data to be generated.

2. Time and Resources

ML needs enough time to let the algorithms learn and develop enough to fulfill their purpose with a considerable amount of accuracy and relevancy. It also needs massive resources to function. This can mean additional requirements of computer power for you.

3. Interpretation of Results

Another major challenge is the ability to accurately interpret results generated by the algorithms. You must also carefully choose the algorithms for your purpose.

IMPLEMENTATION

5.1 Libraries:

Tensorflow

TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks. It is a symbolic math library, and is also used for machine learning applications such as neural networks . It is used for both research and production at Google .

TensorFlow was developed by the Google Brain team for internal Google use. It was released under the Apache 2.0 open-source license on November 9, 2015.

Numpy

Numpy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.

It is the fundamental package for scientific computing with Python. It contains various features including these important ones:

- ✦ A powerful N-dimensional array object
- ✦ Sophisticated (broadcasting) functions
- ✦ Tools for integrating C/C++ and Fortran code
- ✦ Useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, Numpy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined using Numpy which allows Numpy to seamlessly and speedily integrate with a wide variety of databases.

Pandas

Pandas is an open-source Python Library providing high-performance data manipulation and analysis tool using its powerful data structures. Python was majorly used for data munging and preparation. It had very little contribution towards data analysis. Pandas solved this problem. Using Pandas, we can accomplish five typical

steps in the processing and analysis of data, regardless of the origin of data load, prepare, manipulate, model, and analyze. Python with Pandas is used in a wide range of fields including academic and commercial domains including finance, economics, Statistics, analytics, etc.

Matplotlib

Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter Notebook, web application servers, and four graphical user interface toolkits. Matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, error charts, scatter plots, etc., with just a few lines of code. For examples, see [thesample plots and thumbnail gallery](#).

For simple plotting the pyplot module provides a MATLAB-like interface, particularly when combined with IPython. For the power user, you have full control of line styles, font properties, axes properties, etc, via an object oriented interface or via a set of functions familiar to MATLAB users. **Scikit – learn**

Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. It is licensed under a permissive simplified BSD license and is distributed under many Linux distributions, encouraging academic and commercial use.

5.2 MODULES:

1.Image-Acquisition:

This module is responsible for capturing the image of the currency note. The image can be obtained from various sources such as cameras, scanners, or other image capturing devices.

2.Feature-Extraction:

This module is responsible for extracting the relevant features of the currency note. The features can include colour, texture and shape.

3.Classification:

This module is responsible for identifying the denomination of the currency note. The classification module uses a trained classifier to identify the note based on its features.

4.Output:

This module is responsible for presenting the results of the recognition process. The output can be in the form of a text message or an image.

5.3DATASET



Figure-5.3.1 5rs note



Figure-5.3.2 500rs note



Figure-5.3.3 10\$ note



Figure-5.3.4 20\$ note

5.4 EXECUTABLE CODE

```
from tkinter import *
import tkinter
from tkinter import filedialog
from tkinter.filedialog import askopenfilename
import numpy as np
import os
import cv2
from tkinter import messagebox

main = tkinter.Tk()
main.title("Currency Recognition System Using Image Processing")
main.geometry("1200x1200")

global filename

def uploadImage():
    global filename
    text.delete('1.0', END)
    filename = filedialog.askopenfilename(initialdir="testImages")
    text.insert(END, str(filename) + " Dataset Loaded\n\n")
    pathlabel.config(text=str(filename) + " Dataset Loaded\n\n")

def recognizedCurrency():
    text.delete('1.0', END)
    global filename
    img_rgb = cv2.imread(filename)
    img_gray = cv2.cvtColor(img_rgb, cv2.COLOR_BGR2GRAY)
    maxVal = 0
    flag = 0
    temps = None
    recognized = None
    currency = None
    for root, dirs, directory in os.walk('Dataset'):
        for j in range(len(directory)):
            name = os.path.basename(root)
```

```

if 'Thumbs.db' not in directory[j]:
    template = cv2.imread(root+"/"+directory[j],0)
    w, h = template.shape[::-1]
    img_gray = cv2.resize(img_gray,(w,h))
    img_rgb = cv2.resize(img_rgb,(w,h))
    res
= cv2.matchTemplate(img_gray,template,cv2.TM_CCOEFF_NORMED)
    threshold = 0.4
    loc = np.where( res >= threshold)
    for pt in zip(*loc[::-1]):
        #cv2.rectangle(img_rgb, pt, (pt[0] + w, pt[1] + h), (0,0,255), 2)
        (minVal, maxVal, minLoc, maxLoc) = cv2.minMaxLoc(res)
        if minVal > maxVal:
            maxVal = minVal
print(root+"/"+directory[j]+" "+str(minVal)+" "+str(maxVal)+" "+str(minLoc)+"
    "+str(maxLoc)+" "+str(res.ravel()))
    (startX, startY) = maxLoc
    endX = startX + template.shape[1]
    endY = startY + template.shape[0]
    cv2.rectangle(img_rgb, (startX, startY), (endX, endY), (255, 0, 0), 3)
    temps = template
    flag = 1
    recognized = os.path.basename(root)
    currency = os.path.basename(os.path.dirname(root))

if flag == 1:
    img_rgb = cv2.resize(img_rgb,(600,400))
    xi = float(input("Enter the currency rate for rupee"))
    xd = float(input("Enter the currency rate for dollar"))
    xe = float(input("Enter the currency rate for euro"))
    cv2.putText(img_rgb, 'Currency Recognized as : '+currency+" "+recognized,
(10, 25), cv2.FONT_HERSHEY_SIMPLEX,0.7, (0, 0, 255), 2)
    text.insert(END, 'Currency Value : '+recognized+"\n")
    text.insert(END, 'Currency Name : '+currency+"\n")
    if currency == 'INR':

```

```

text.insert(END, 'Currency in Euro : '+str(float(recognized) / xe)+"\n")
text.insert(END, 'Currency in USD : '+str(float(recognized) / xd)+"\n")
cv2.putText(img_rgb, 'Currency in Euro : '+str(float(recognized) / xe), (10,
55), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
cv2.putText(img_rgb, 'Currency in USD : '+str(float(recognized) / xd), (10, 85),
cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
if currency == 'USD':
text.insert(END, 'Currency in Euro : '+str(float(recognized) / 1.1)+"\n")
text.insert(END, 'Currency in INR : '+str(float(recognized) * xd)+"\n")
cv2.putText(img_rgb, 'Currency in Euro : '+str(float(recognized) / 1.1), (10, 55),
cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
cv2.putText(img_rgb, 'Currency in INR : '+str(float(recognized) * xd), (10, 85),
cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
if currency == 'EURO':
text.insert(END, 'Currency in USD : '+str(float(recognized) * 1.1)+"\n")
text.insert(END, 'Currency in INR : '+str(float(recognized) * xe)+"\n")
cv2.putText(img_rgb, 'Currency in USD : '+str(float(recognized) * 1.1), (10, 55),
cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
cv2.putText(img_rgb, 'Currency in INR : '+str(float(recognized) * xe), (10, 85),
cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
text.update_idletasks()
cv2.imshow("original Image",img_rgb)
cv2.imshow("matched Image",temps)
cv2.waitKey(0)
else:
messagebox.showinfo("Unable to recognized", "Unable to recognized")
def close():
main.destroy()
font = ('times', 14, 'bold')
title = Label(main, text='Currency Recognition System Using Image Processing')
title.config(bg='DarkGoldenrod1', fg='black')
title.config(font=font)
title.config(height=3, width=120)
title.place(x=5,y=5)
font1 = ('times', 13, 'bold')

```



```

uploadButton = Button(main, text="Upload Currency Image",
command=uploadImage)
uploadButton.place(x=50,y=100)
uploadButton.config(font=font1)
pathlabel = Label(main)
pathlabel.config(bg='brown', fg='white')
pathlabel.config(font=font1)
pathlabel.place(x=560,y=100)
matchButton = Button(main, text="Run Template Matching Currency Recognition",
command=recognizedCurrency)
matchButton.place(x=50,y=150)
matchButton.config(font=font1)
exitButton = Button(main, text="Exit", command=close)
exitButton.place(x=50,y=200)
exitButton.config(font=font1)
font1 = ('times', 12, 'bold')
text=Text(main,height=25,width=100)
scroll=Scrollbar(text)
text.configure(yscrollcommand=scroll.set)
text.place(x=520,y=150)
text.config(font=font1)
main.config(bg='LightSteelBlue1')
main.mainloop()

```

TESTING

6.1 TESTING DEFINATION:

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

6.2 UNIT TESTING

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Since, the grey box testing includes access to internal coding for designing test cases. Grey box testing is performed by a person who knows coding as well as testing.

6.3 Test case

A test case is the individual unit of testing. It checks for a specific response to a particular set of inputs.

Test suite

A test suite is a collection of test cases, test suites, or both. It is used to aggregate tests that should be executed together.

Test runner

A test runner is a component which orchestrates the execution of tests and provides the outcome to the user. The runner may use a graphical interface, a textual interface, or return a special value to indicate the results of executing the tests.

Outcomes Possible:

There are three types of possible test outcomes:

- OK – This means that all the tests are passed.
- FAIL – This means that the test did not pass and an `AssertionError` exception is raised.
- ERROR – This means that the test raises an exception other than `AssertionError`.

Test cases

User Requirements:

Home:

Table:6.3.1

| | |
|-----------------------|--|
| Use case ID | Currency-Recognition-System-Using-Image-Processing |
| Use case Name | Home button |
| Description | Display home page of application |
| Primary actor | User |
| Precondition | User must open application |
| Post condition | Display the Home Page of an application |
| Frequency of Use case | Many times |
| Alternative use case | N/A |
| Use case Diagrams | |
| Attachments | N/A |

Upload Image:

Table:6.3.2

| | |
|-----------------------|--|
| Use case ID | Currency-Recognition-System-Using-Image-Processing |
| Use case Name | Upload Image button |
| Description | Show window For selection of image |
| Primary actor | User |
| Precondition | User must open application |
| Post condition | Display the window for selecting image |
| Frequency of Use case | Many times |
| Alternative use case | N/A |
| Use case Diagrams | |
| Attachments | N/A |

Recognize:

Table:6.3.3

| | |
|-----------------------|---|
| Use case ID | Currency-Recognition-System-Using-Image-Processing |
| Use case Name | Recognize button |
| Description | This button is used for currency recognition. After pressing this button currency recognition will start and you will get output in few seconds on the application window |
| Primary actor | User |
| Precondition | User must open application |
| Post condition | This button is used for currency recognition. After pressing this button currency recognition will start and you will get output in few seconds on the application window |
| Frequency of Use case | Many times |
| Alternative use case | N/A |
| Use case Diagrams | |
| Attachments | N/A |

Exit:

Table:6.3.4

| | |
|-----------------------|--|
| Use case ID | Currency-Recognition-System-Using-Image-Processing |
| Use case Name | Exit button |
| Description | This works as simple exit button. |
| Primary actor | User |
| Precondition | User must open application |
| Post condition | This works as simple exit button. |
| Frequency of Use case | Many times |
| Alternative use case | N/A |
| Use case Diagrams | |
| Attachments | N/A |

RESULTS

- To run project double click on 'run.bat' file to get below screen.
- In above screen click on 'Upload Currency Image' button to upload currency images like below screen.

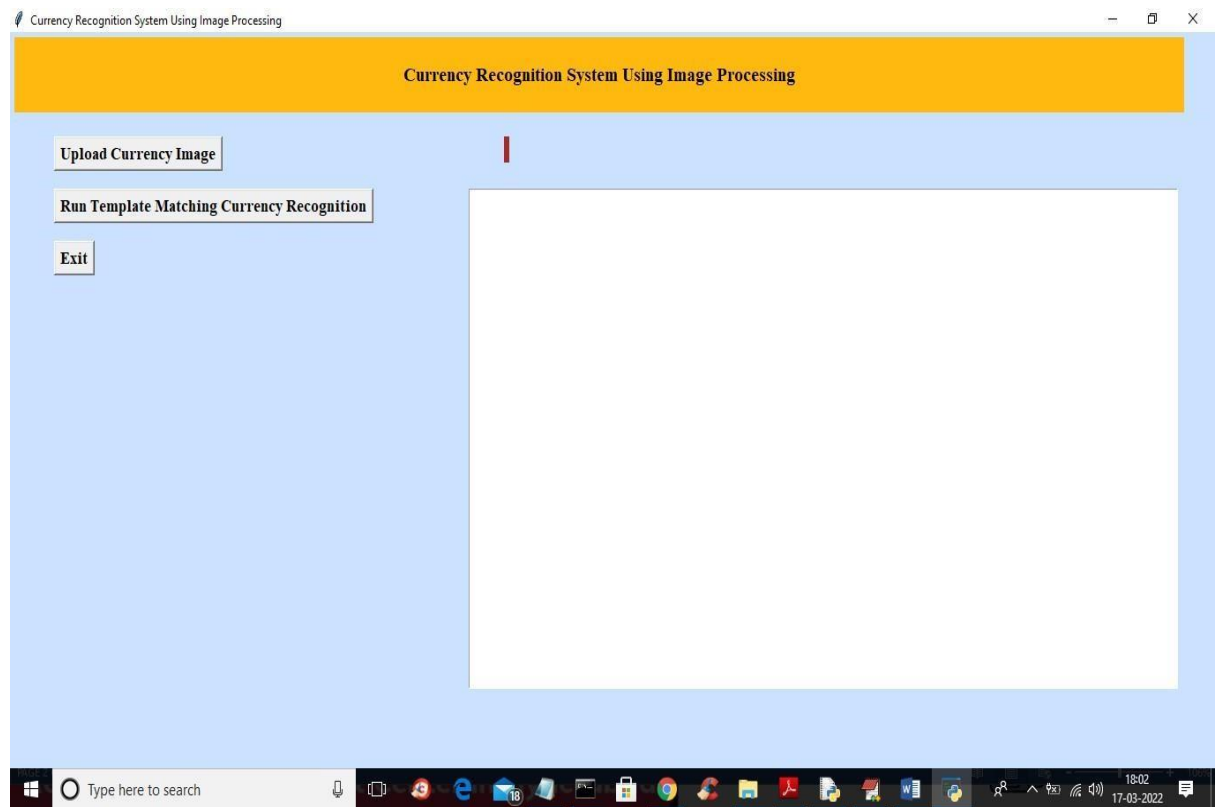


Figure-7.1upload currency image

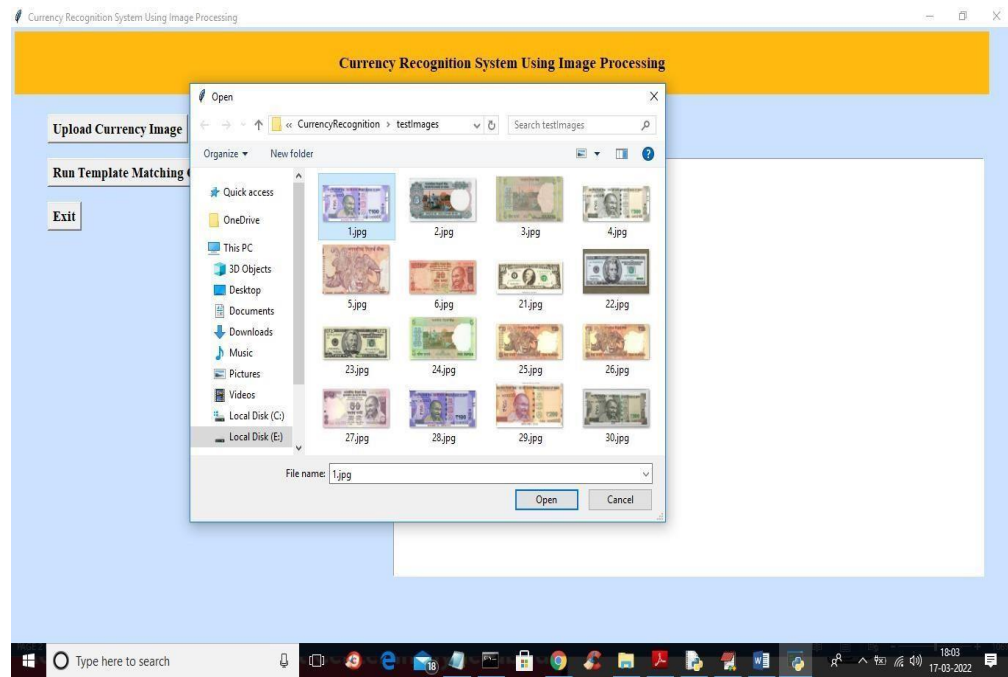


Figure-7.2 select and load image

In above screen selecting and uploading '1.jpg' file and then click on 'Open' button to load image and then click on 'Run Template Matching Currency Recognition' button to get below output.

Currency Recognition System Using Image Processing

```
Select C:\Windows\py.exe
Dataset\INR\100/100.jpg 0.6765773296356201 0.6765773296356201 (0, 0) (0, 0) [0.6765773]
Enter the currency rate for rupee1
Enter the currency rate for dollar82.26
Enter the currency rate for euro90.79
```

Figure-7.3 Enter the Exchange Rates



Figure-7.4 displays the original image and training image

In above screen first image is the training image template and second image is the original image so by applying template matching algorithm we can predict correct currency and that currency is recognized as INR 100. Similarly you can upload other image and test.

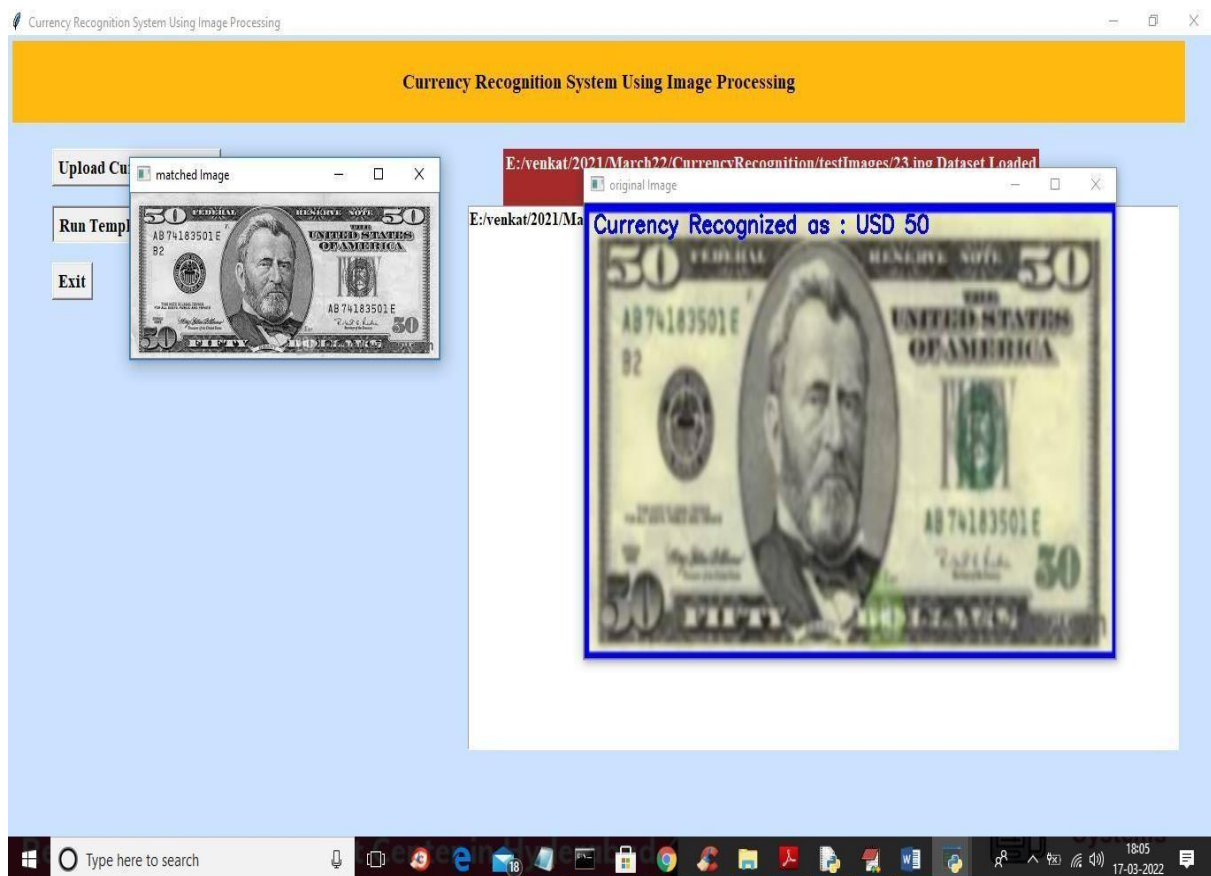


Figure-7.5 image recognized in blue colour text

In above screen in blue colour text we can see currency identified as USD 50

CONCLUSION & FUTURE SCOPE

8.1 CONCLUSION

In conclusion, we have designed a system that accurately identifies both the country of origin and the denomination of a given banknote. Our system currently supports twenty of the most common currencies, but can easily be extended to more countries based on the method we have previously described. When compared with the crude algorithm of pixel by pixel comparison, our algorithm is considerably more accurate, and takes less time. We have thus learned that our proposed algorithm is able to identify currency and denomination in an average of 5.3 seconds, which is a considerable improvement over the crude algorithm. However, our proposed system only considers a limited number of currencies. There are 180+ currencies that can be included in the system, and we have chosen to only do for 20 of the most common ones. Also, the system should be effective in identifying notes that are mutilated. Our system is not effective under this consideration. This can be worked on in the future..

8.2FUTURE SCOPE

- The future study will be done by applying different filters.
- In this thesis the images were scanned horizontally in the future the images will be scanned with different angles.
- Different currencies could be used for recognition like Indian Rupee, US dollar, EURO etc.

Similarly different features can be used for recognition

REFERENCES

Pearson Book on Digital Image Processing 4th Edition By Gonzalez (2018).

- [1]. Programming and Problem Solving with Python 1st Edition by 'Ashok Namdev Kamthane' and 'Amit Ashok Kamthane' 2017
- [2]. Identification of Paper Currency Techniques: A Survey by S. Sahu and T. Verma International Journal of Science Technology & Engineering (2016) 4. Recognition system for real time paper currency. IEEE 2012.
- [3]. Extraction of serial number on bank notes. IEEE.
- [4]. Digital Image Processing And Analysis: Human And Computer Vision Applications With Cviptools, 2nd Edition by Scott E Umbaugh, T and F India.
- [5]. An intelligent system for paper currency recognition - robust features Journal of Intelligent & Fuzzy Systems - A. Sargano, M. Sarfrazb, and N. Haq, 2014.
- [6]. Paper Currency Recognition - Journal of Advanced Computer Science and Technology Research, 'Althafiri E, Sarfraz M', 'Alfarras M. Bahraini'.
- [7]. Study of Printing Identification Based on Multi spectrum Imaging Analysis, Proceedings of the International Conference on Computer Science and Software Engineering, By Liu Q, Tang L. p, 229 – 232, 2008.