Currency Recognition System

A Project Report

Submitted by

Smriti Singh	**111508068**
Ameya Tammewar	**111508069**
Aanchal Chugh	**111510001**

in partial fulfillment for the award of the degree

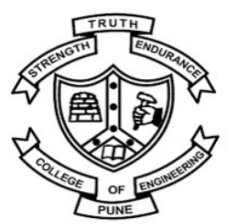
of

B.Tech Information Technology

Under the guidance of

Prof. S. P. Gosavi

College of Engineering, Pune



DEPARTMENT OF COMPUTER ENGINEERING

AND

INFORMATION TECHNOLOGY,

COLLEGE OF ENGINEERING, PUNE-5

May, 2019

DEPARTMENT OF COMPUTER ENGINEERING

AND

INFORMATION TECHNOLOGY,

COLLEGE OF ENGINEERING, PUNE

CERTIFICATE

Certified that this project, titled "Currency Recognition System" has been successfully completed by

Smriti Singh	**111508068**
Ameya tammewar	**111508069**
Aanchal Chugh	**111510001**

and is approved for the partial fulfillment of the requirements for the degree of "B.Tech. Information Technology".

SIGNATURE
Prof. S.P. Gosavi
Project Guide
Department of Computer Engineering and Information Technology, College of Engineering Pune, Shivajinagar, Pune - 5.

SIGNATURE
Dr. Vahida Attar
Head
Department of Computer Engineering and Information Technology, College of Engineering Pune, Shivajinagar, Pune - 5.

Abstract

Taking into consideration the various currencies all over the world, each of them can be differentiated from each other in terms of size, color, texture etc. The people who work in forex exchange, banks, financial institutions and organizations, etc have to detect and differentiate amongst all the types of currencies.

In our proposed system, our relevance is to focus on using image processing techniques and recognize the country as well as the denomination of the detected currency. Detection is done by using a hardware like webcam. Once the currency is detected, its features like colour, shape, size, depth, curvature, orientation, scale, etc are matched with the saved features. The currency dataset stores all the features of the different currencies beforehand. This helps in quick classification into class model of country and denomination. The country and denomination of the different currencies detected are extracted and stored and provides current exchange rate conversion to any selected forex rate. Therefore, the human effort to recognize the currency or hardware like UV currency recognizer, etc is computerized and made time-saving and efficient.

One of the main features of our system is to recognize the banknote from any side, orientation and environment (e.g. dim lights, bright background, etc). Banknotes maybe old or noisy or torn. It is not easy to recognize such notes. Training dataset teaches our system to recognize these as well and quite effectively.

Keywords

Image processing techniques, currency recognition, country, denomination, forex

Contents

List of Tables

List of Figures

List of Abbreviations

- SRS : Software Requirements Specifications
- SIFT : Scale Invariant Feature Transform
- FLANN : Fast Library For Approximate Nearest Neighbour
- GUI : Graphical User Interface
- UV : Ultra Violet
- IR : Infra Red
- MRSAR : Multi Resolution Simultaneous Auto Regressive
- SURF : Speeded Up Robust Feature
- RGB : Red Green Blue
- HSV : Hue Saturation Value
- HIS : Hue Intensity Saturation
- YIQ : Luminance(Y) In-phase Quadrature
- OS : Operating System

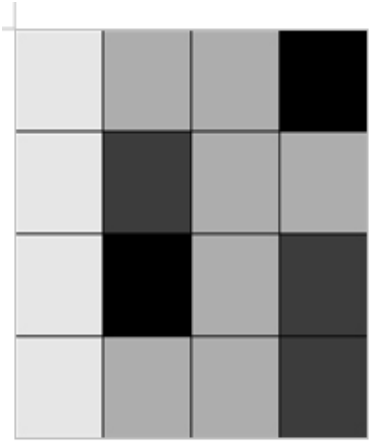
Chapter 1

Introduction

Taking into account the currencies all over the world, there might be more than 50 currencies with each one completely different from each other. For example, in terms of size, colour and pattern. It is not an easy task to differentiate between different types of currencies for the ones who work for financial institutions (e.g. in Forex Banks). Remembering the symbol of each currency is tedious and may lead to wrong recognition. Recognizing various currencies, ensuring convenience and efficiency is the main aim of our system. Currency Sorting Machines currently help bank staff to identify the various different types of currencies through capturing and analysis of image. It is an analogy of different calculations, pattern matching, fake currency detection and various protean techniques and is named "electronic, optical and mechanical integration" which is highly remarkable in terms of efficiency and accuracy. Our aim is to do the same in lesser time with a more user-friendly interface and provide better features like dynamic denomination conversion according to user selection.

1.1 Image

An image can be defined as a two dimensional signal. It is represented as F(x,y) where x is the horizontal coordinate and y is the vertical coordinate. The value of F(x,y) provides the pixel value of the point (x,y) at any time. F(x,y) can take a value between 0 to 255 and an image in nothing but a two dimensional array containing the values of all pixels. The dimension of an image would be the dimension of its two dimensional array.



```
{236, 189, 189, 0},
{236, 80,  189, 189},
{236, 0,   189, 80},
{236, 189, 189, 80}
```

1.2 Image processing

Image processing is an approach of converting an image to a digital structure to either enhance it or mine pragmatic information. Image input is like a signal disbursement from video or captured image. Output maybe another refined image or characteristics extracted. Image processing is one of the major sectors of core research areas of engineering today because of its various applications (eg. remote sensing, moving object sensing, defense surveillance, bio medical imaging techniques, etc) and great capitalization capacity.

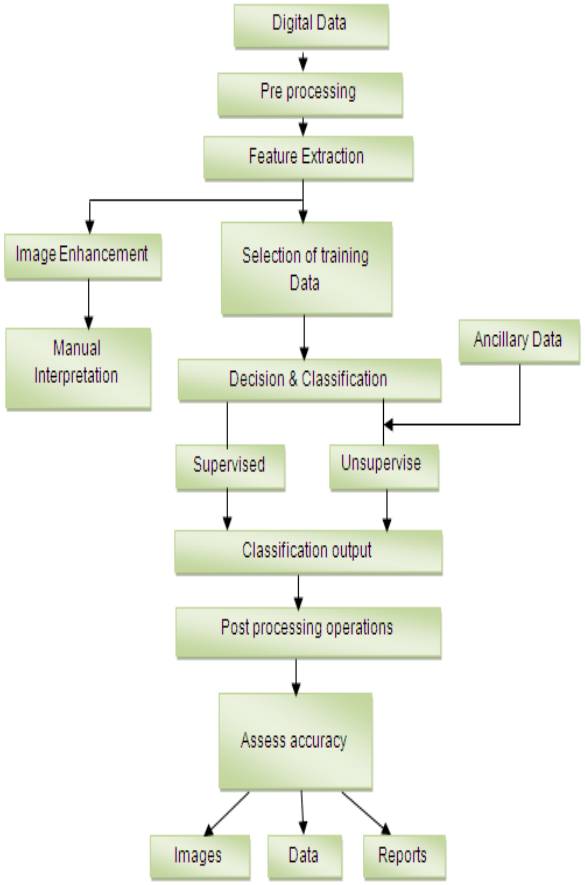
Three majors steps included are:

1) Receive an input image

2) Interpretation and manipulation of input image

3) Output as altered image or report based on image analysis

Below shown is the flowchart of different phases of digital image processing



There are two methods for image processing - analog and digital.

Analog method is mostly used for hard copies. The image is interpreted by an analyst and is purely based on his/her knowledge of visual techniques. Associating previously analyzed images to the input image is one of the major visual technique used.

Digital processing manipulates the input image because the raw input maybe deficient.The flaws are improved upon by applying various processing techniques.

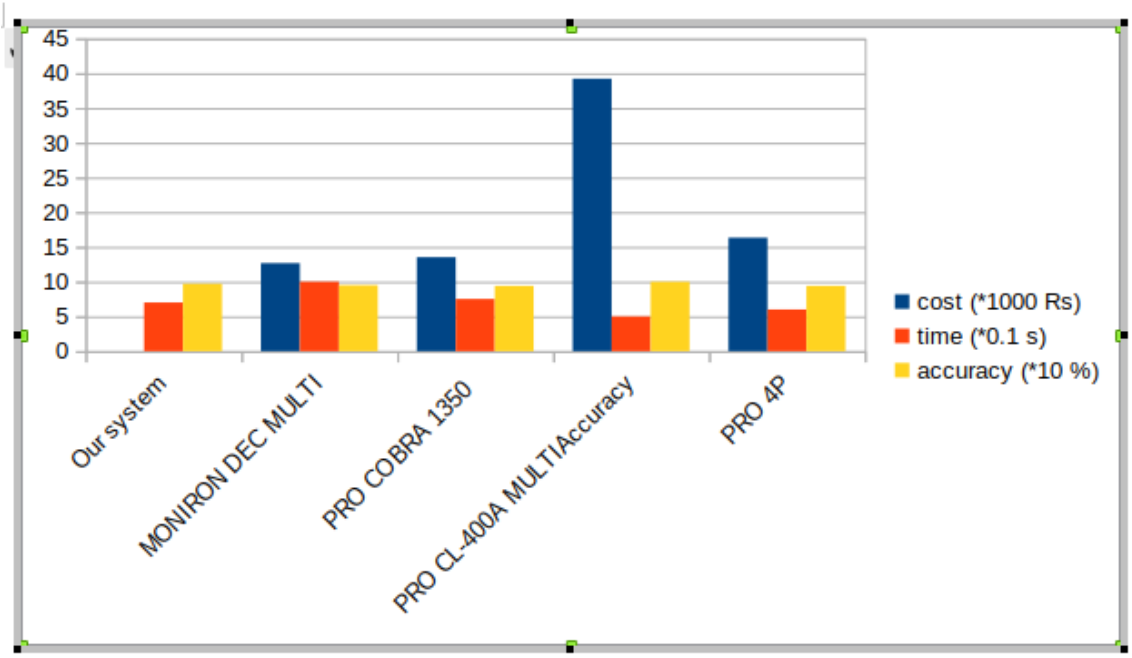
1.3 Problems with current system

Today, currency recognition is majorly done using currency detector hardwares (PRO CL 16 IR LCD, MONIRON DEC MULTI, PRO CL-400A MULTI, etc) in banks, forex exchanges, etc. These hardware use UV or IR technologies to scan the notes and similar notes at a time to give accurate results. But hardwares are not portable and incur maintenance cost, service cost, etc. If this system could be computerized like we propose, the usability of this system could improve and could be used in shopping malls, railway system, etc.

1.4 Relevance

We intend to focus on identification of different types of currencies using Image Processing. Building a fast and accurate system with ability to recognize notes from any side, any condition (e.g. old, torn), any angle, any scale and any kind of lighting.

A user-friendly interface so that the users don't have to go through too much trouble with the working of our system. Useful features like conversion of different currency denominations to one common and chosen denomination using accurate and updated Forex rates and also add up the denominations. Also, our system offers software based computerized solution which is portable and cost effective.



1.5 System Background

The proposed application will display currency denomination, side (front/back), country and if opted can ask for the country/currency to which the given currency is to be converted. The system goes through the following process- capturing the image through webcam, removing unnecessary noise from captured image, identifying feature points, and matching based on feature extraction. It will be verified for different denominations e.g, 50, 100, different country, old notes and different light conditions.

1.6 Future scope

- Fake currency detection along with currency detection
- Android application to improve usability of this system
- Link the system to cloud to store the data

Chapter 2

Problem Statement

2.1 Aim

When people travel or send money across countries, the notes need to be identified correctly and converting them to appropriate receiver denomination. The main intent of our system is to help people get solution to the above condition. Some currency recognition systems are inefficient since they focus entirely on analysis of images. The system that we intend to provide uses image processing, making process automatic and robust.

2.2 Features

- Receiving image or webcam input from a user system
- Processing the input image to diagnose the country and denomination of the currency
- Required denomination exchange rate is determined using Forex rates at the instant
- Total amount in selected denomination of various different types of currencies can be calculated

Chapter 3

Literature Survey

3.0.1 Texture based Recognition Technique

Texture is the feel, appearance, or consistency of a surface or a substance. Texture caters the extent of properties like regularity, smoothness and coarseness. The texture of a region is described using three dominant approaches in image processing - structural, spectral and statistical.

Statistical approach deals with smoothness, coarseness, grainyness and so on. Structural techniques deal accord with regularly spaced lines based texture. Spectral techniques are established on properties of Fourier spectrum and used to detect global periodicity of an image. Some texture based image retrieval tools are edge histogram, Gabor texture feature, Tamura features, pyramid-structured, MRSAR and tree-structured wavelet transform. Gabor extracts information at multiple scales and orientations. Tamura coarseness finds mean coarseness at each pixel inside the texture region. Tamura features finds the coarseness, directionality and contrast of an image. MRSAR (Multi Resolution Simultaneous Auto Regressive) provides results which are helpful in distinguishing different texture patterns. The performance of these features are not optimum since they compute directly from input images

3.0.2 Pattern based Recognition Techniques

Pattern based recognition encompasses collocation of objects based on sets of images. Experimentation is required to obtain preceding knowledge about objects to arrive at a conclusion. It also requires a large agglomeration of data. These techniques create histogram models using vector quantization. Vector quantization is a sampling method where each point xi in a set of data in k-dimensional space is replaced by N prototype points. The prototype points are elected such that sum of distances (i.e. distortion) from each data point xi to its nearest prototype point is curtailed.

The process includes data collection with different backgrounds. Next, segmentation and cropping is done. In this step, Segmentation of image from its background is done using Nechbas code. After locating the edges of the objects in an image, it can be cropped. The features are then extracted by writhing texture templates with each image. Next step is to get high accuracy by training the program with as much data as possible. This method provides 94 percent accuracy.

It is found that recognition of paper currency is quite simple, easy to be realized and efficient after a lot of experimentation. Denomination numerals are used for identification which can be extracted easily from paper currency. Then, extracted numerals are matched and create a bucket of output. The next feature is then matched to create next bucket until an exact match is found.

Ensemble neural network consist of a number of neural networks which are trained independently. Negative correlation learning is how neural networks are trained. The system is able to recognize highly noisy or old image and reduces the wrong classification of notes.

Faiz M. Hasanuzzaman proposed a component based framework for cur-

rency recognition by using Speeded Up Robust Features (SURF). It divides the image into components and then matches test image with original image component. The SURF features remain constant under conditions of changes of scaling, image rotation, viewpoint, illumination and image occlusion. The above algorithm achieves 100 percent accuracy.

3.0.3 Colour based Recognition Techniques

Many colours are used in different currencies but distinction requires only one or two assertive colors in each currency. The image inputted from the camera is in RGB format. Color spaces are representation formats of electronic processing of the colour signals (Red, Green and Blue). There are variety of color spaces like YUV, YCrCb, YIQ, HIS and HSV and each format has its own pros. The performance of procedures such as segmentation are greatly influenced by the colour space.

Amplitude of data is reduced after inputting the image. Conversion to color space suitable for processing from RGB color space takes place. After processing, the features are extracted, matched and denomination is detected.

In 2012, a group converted an image to HSV color space from RGB color space before feature extraction. HSV color space is close to human notion of understanding of color and was hence used. Color histogram, hue, saturation and intensity value was scrutinized for feature extraction. Saved and input images' comparison took place and currency denomination was found when divergence in threshold value was greater than the stated value.

Histogram counts the number of pixels of each color and characterizes the global color distribution in an image. When range of color and prominent color have to be segregated, color histogram method best suits. A drawback of this algorithm includes the fact that a color histogram describes in what

quantity and which colors are present in the image but does not provide spatial information.

Color indexing technique is used to extract features based on color content of the image. Color regions in the image is set using back-projection of binary colors in this algorithm. It conquers some of the problems with color histogram.

Recognition of an old note whose color faded or changed due to rigorous circulation is what is desired from colour based recognition.

3.0.4 Currency Localization Techniques

A blended technique in which image of a currency note is localized and application of different algorithms based on thresholds is applied on it, in order to get the denomination of the note. Input is taken from webcam which is the major advantage of this technique. There is no need of importable hardware. Normalization, noise reduction and contrast magnification, etc are some of the pre-processing techniques which are to be applied in order to enhance the image. Performing gray conversion from RGB and edge detection comes under the background subtraction step. After edge detection, scan line algorithm is to be applied on image in order to carry out localization of currency note. Scanning of the image is to be done line by line and from top to bottom and left to right. Now, highlight the line containing number of pixels greater than the threshold set. A rectangle will be formed by the intersection of these lines and is called as a key point space. In order to recognize the currency, color matching technique is then applied.

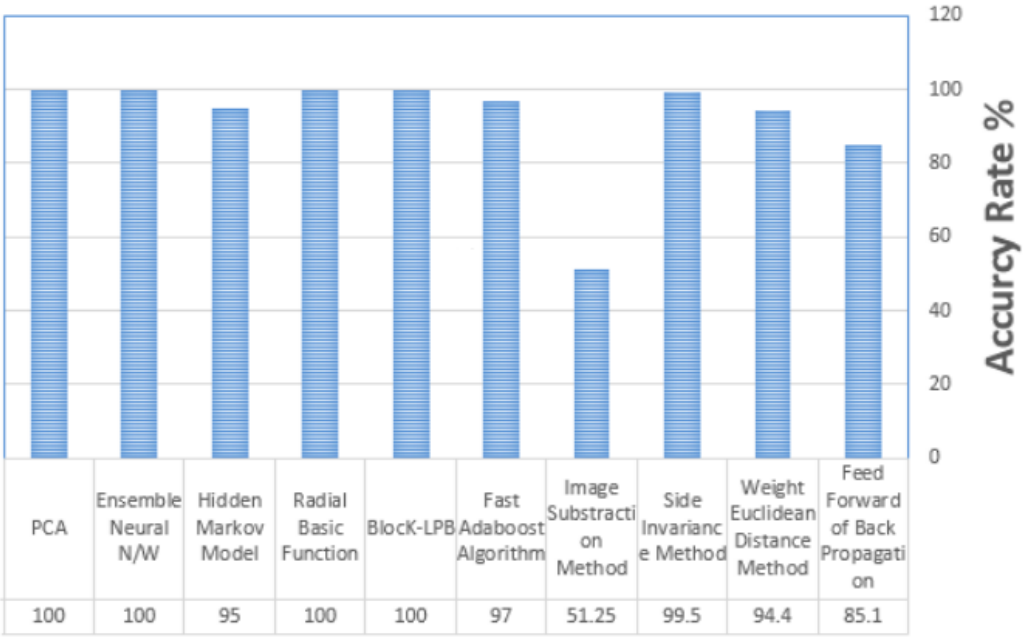


Figure : comparison between Accuracy of Various Paper Currency Recognition Techniques.

Algorithm	Scale & Rotation	Blur	Illumination	Affine	Time Cost
SIFT	Best	Good	Better	Good	Better
PCA-SIFT	Better	Better	Better	Good	Better
GSIFT	Good	Best	Best	Good	Better
CSIFT	Best	Better	Good	Better	Good
SURF	Common	Common	Common	Common	Best
ASIFT	Good	Common	Common	Best	Common

Sl.no	Texture and Shape	Properties	Advantage	Disadvantage	Application
1	LBP (Local Binary Pattern)	Two dimensional surface textures can be described by local spatial patterns and gray scale contrast.	Its usage in interest region is tolerance against illumination changes Computational simplicity Very fast to compute Do not require many parameters to be set	The recognition speed is slow on large-scale database; Under some certain circumstance it miss the local structure Sensitive to noise Large number of dimensions is required.	Computer vision Face recognition Background subtraction Biometrics Gait recognition Facial age classification
2	SIFT (Scale Invariant Feature Representation)	Gradient orientation and magnitude-based feature is extracted	Robust to change of intensity Illumination, scaling and rotation. Easy to localize the pixels	Slight variations in luminance will produce for key points similar to those produced by large variations	Object recognition, Image stitching, 3D modelling Video tracking
3	SURF (Speeded Up Robust Feature)	It a detector and a descriptor for points of interest in images where the image is transformed into coordinate, using the multi-resolution pyramid technique.	Extraction of interest points is faster Best for scale and Illumination variation Easy to localize and find correspondence between frames Robust to large, unpredictable inter frame motion	View dependent Extraction of sufficient number of key frames is difficult in 3D environment Time consuming Appearance of key point can change over frames	Object Recognition and Tracking. Robot Localization and Mapping Image Registration. Image Retrieval
4	GLOH (Gradient Location and Orientation Histogram)	Extension of the SIFT descriptor Histogram based gradient orientation is used	Best on the structured scene Robust to illumination changes	It perform worse for computed on scale invariant regions The edges disappear in the case of a strong blur	Object Recognition and Tracking. Image Registration. Image Retrieval
5	Gabor Filters	Frequency and orientation representations based filter. Similar to those of the human visual system.	Robustness against varying brightness Invariant to Contrast of images, Illumination variation and rotation	When Gabor filters are applied to each pixel of the image, the dimension of the filtered vector can be very large. Expensive computation storage cost.	Face recognition Traffic measurement and management, Traffic surveillance,
6	Schmid	Similar to those of the human visual system.	Reduction in computational complexity.	Only keep track of the effect of the bias on the true error distribution.	Traffic surveillance, Pedestrian detection
7	HOG (Histogram of Oriented Gradient)	The local object appearance and shape of an image can be described Based on intensity gradients or edge directions. The HOG descriptor is thus particularly suited for human detection in images.	Invariance to local geometric and photometric transformations Better able to deal with moving body parts and handle occlusions, overlaps	Translations or rotations make little difference if they are much smaller that the local spatial or orientation bin size.	Person detection Object Detection Pedestrian detection

Chapter 4

System Requirement Specifications

4.1 Intended Audience

Our system can be used in banks, currency exchanges, business firms, railways, shopping malls, departmental stores, government organizations and other financial institutions.

4.2 Software Requirements

OS : LINUX/Windows

FrontEnd : OpenCV library, Python, Tkinter

4.3 Hardware Requirements

Hard Disk : 1 GB Hard-drive space

Ram : 2 GB RAM

Web cam

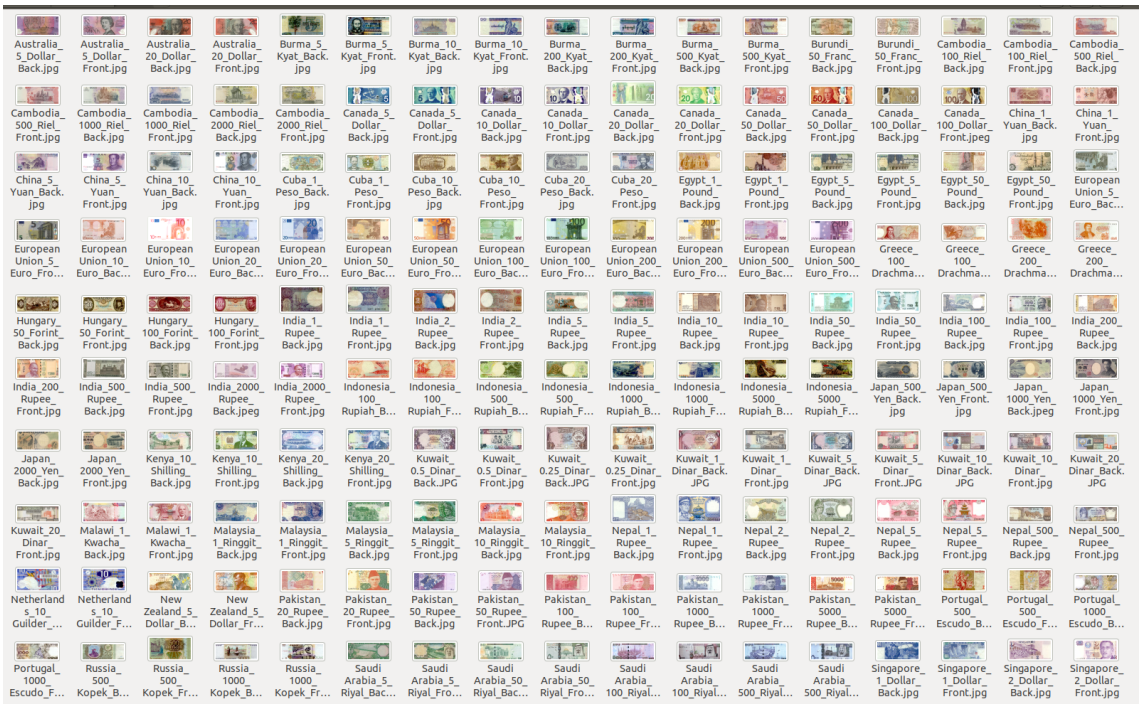
4.4 User Documentation

The application is a python file to be run from terminal in LINUX or command line in Windows. You can also download python and run the file in it.

Version: 2.7.5 or greater

Command: python3 currency_detection.py

The libraries have been called in the file and although different functions are stored in different files, they are called in the main currency_detection.py file. The dataset is a set of images of different currencies of the world with front and back and denomination stored along with the country.

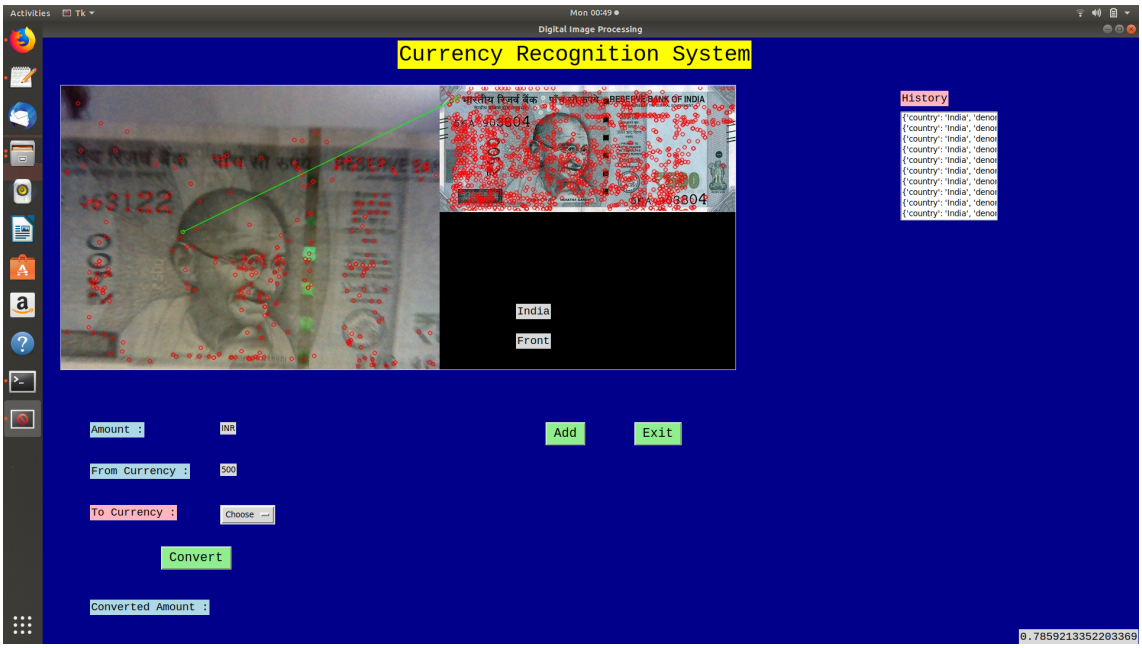


The names of dataset images are stored as country_denomination_name-of-currency_front/back. Training set is created by stripping down the name and storing the different properties.

SIFT algorithm finds the features in each dataset image and stores them. The features are linked to their training set information to create the class model. When input image is taken, SIFT finds its features and FLANN algorithm matches it to the stored features as fast as possible. The linked information is outputted to the user.

The GUI is created using Tkinter library of python. It shows the input, detected keypoints, output information, button to select denomination, button to add currency, output of selected denomination, output of added currency in selected denomination.

The denomination conversion is done according to current exchange rates for which python library forex-python is used.



Chapter 5

Timeline

Task Name	**Duration**	**Start**
Topic discussion	15d	02-08-18
Topic finalization	5d	20-08-18
Abstract	10d	27-08-18
Literature Survey	30d	15-09-18
SRS	45d	22-10-18
Installation	20d	10-01-19
Coding	55d	1-02-19
Testing	30d	27-03-19

Appendix A

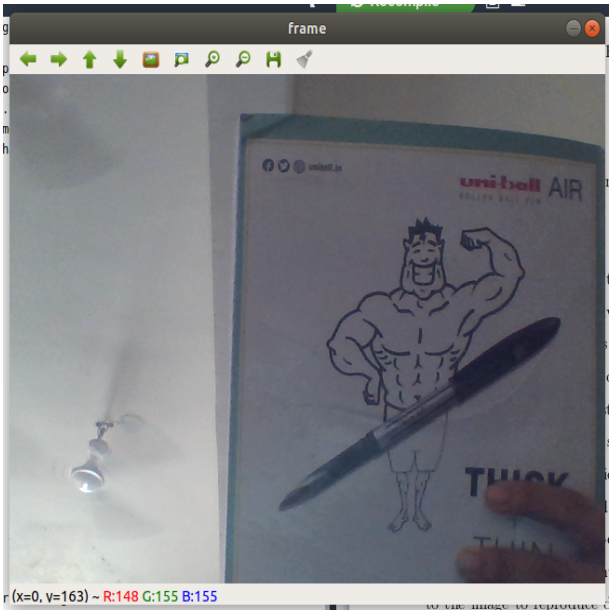
Algorithm

Two algorithms are used SIFT and FLANN.

Methodology

Step1 :Image Acquisition

First step is capturing the image of currency note using web-camera of a computer or giving stored image of a currency note as a input for currency detection.



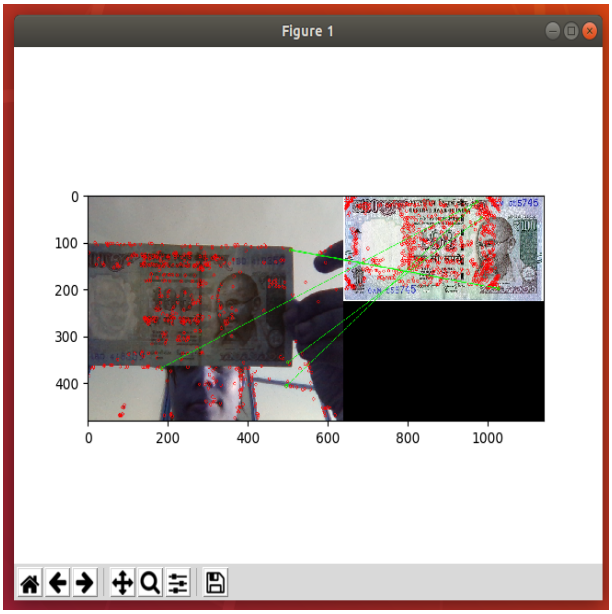
Step2 :Feature Extraction

Feature points extraction is the most challenging task for image recognition.

It is done by using Scale Invariant Feature Transform (SIFT) descriptor algorithm. SIFT algorithm is used for keypoint detection and description. A set of images that are stored in a database are used to extract keypoints of different objects. SIFT distinguishes features which are Invariant to location, background, rotation, scale, noise and Also invariant to linear changes in lighting that is illumination.

Following steps are involved for building a SIFT-descriptor.

1.key-point detection and localization in scale-space Searching for features in scale-space to obtain scale invariance. Gaussian blurring is repeatedly applied to the image to reproduce decreasing scale. Each image is subtracted from its direct neighbors so that Difference-of-Gaussian series images are created. That images are a close approximation to Laplacian-of-Gaussian (LoG). Now three different scales are used to detect keypoint candidates. Each point is compared to its neighboring 8 points on the same scale and each of its 9 neighbors from one scale up and one scale down. Keypoint candidate is every point that is smaller or bigger than every neighbor. Poor contrast candidates are eliminated and also candidates which are located on the edge.

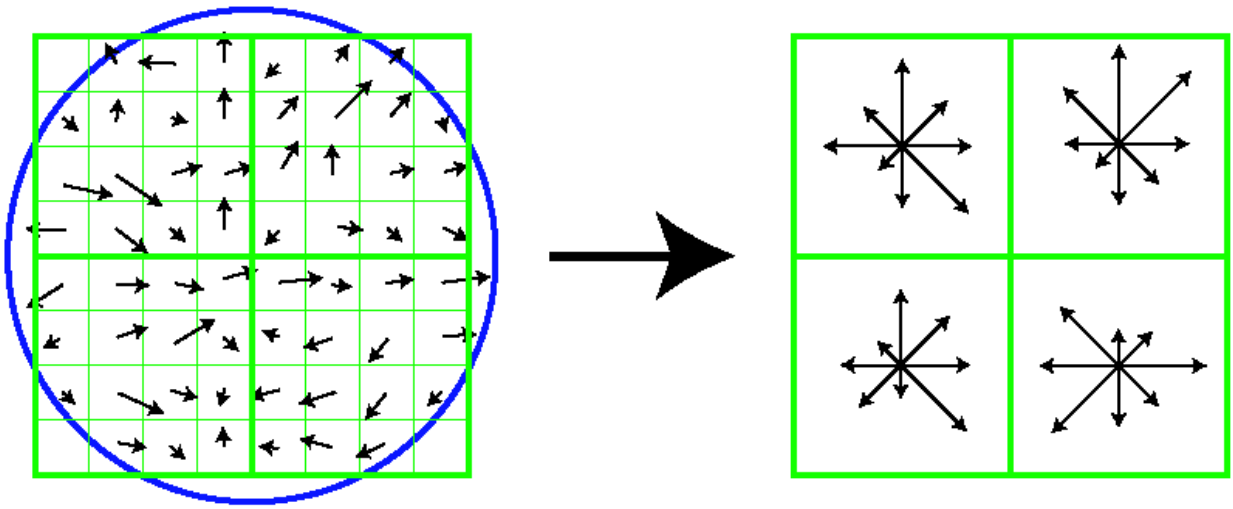


2.Assigning orientations

Image rotation invariance is obtained by assigning orientation to each keypoint. Depending on the scale, a neighbourhood is to be considered around the location of the keypoint ,computing the direction and gradient magnitude in that neighbourhood area. Creation of histogram which will be having 36 bins and will cover 360 degrees. This histogram is called orientation histogram. In computation of the orientation, histogram is analysed to get it highest peak and also consider the peak which is above 80 percent of it. Therefore, it leads to the formation of the keypoints having same scale and location varying in terms of direction.

3.Generating Feature Descriptor

In this step with respect to the keypoint, calculations using orientation histogram of vicinity is done and accordingly a principle orientation is to be given to each keypoint. It will be used for representing each keypoint irrespective of rotation of image and with respect to its orientation. Seeing the orientation histogram, maximum value is to be obtained. Interpolation of the peaks in histogram via adjacent points is done so that each keypoint can be alloted a precise orientation. In order to make feature vector unvarying to rotation change, there is a standard coordinate(original) which is rotated in accordance to main orientation.Computation of orientation and magnitude of gradient at each sample point is done considering the region around the key point, each keypoint vector is established. At last, each feature point gets a feature descriptor having 128 elements.



Step3 :Matching Algorithm

Fast Library For Approximate Nearest Neighbour (FLANN) is used as matching algorithm. It is used for approximate nearest neighbor search. Features that are extracted from pictures which are real time are matched with the features from objects so that objects are finally identified. In order to get features certain pictures from the object to be detected is scanned. Each feature extracted is considered as a point in multi-dimensional space.

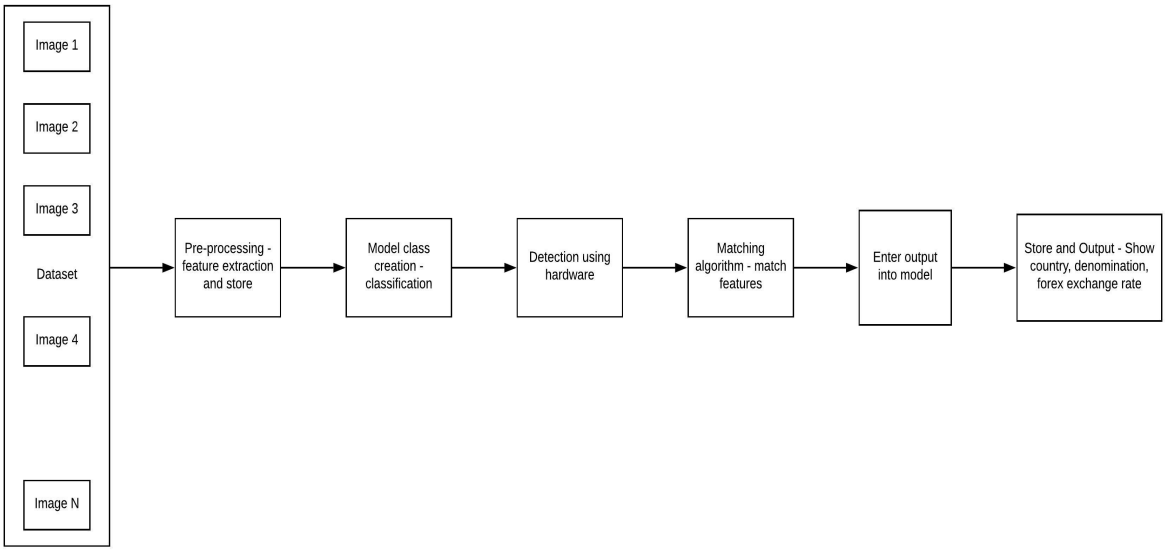
Database using all the points is created using kd-trees as data structure.

First the kd-tree contain one node with all points. If the node contains more than one point then it is split such that maximum dimension spread present at the median. While splitting, points that lie on a line can be considered in any of the adjacent node.

Appendix B

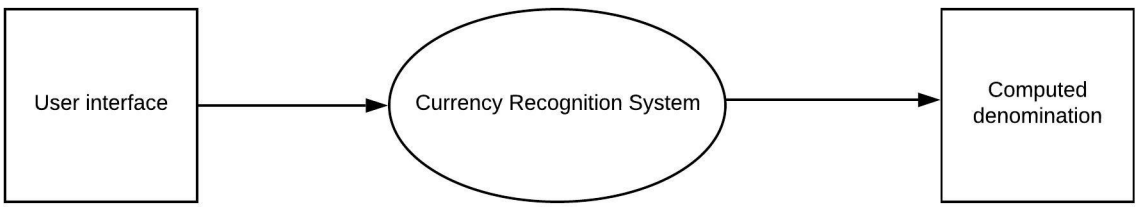
System Design

B.1 System design diagram

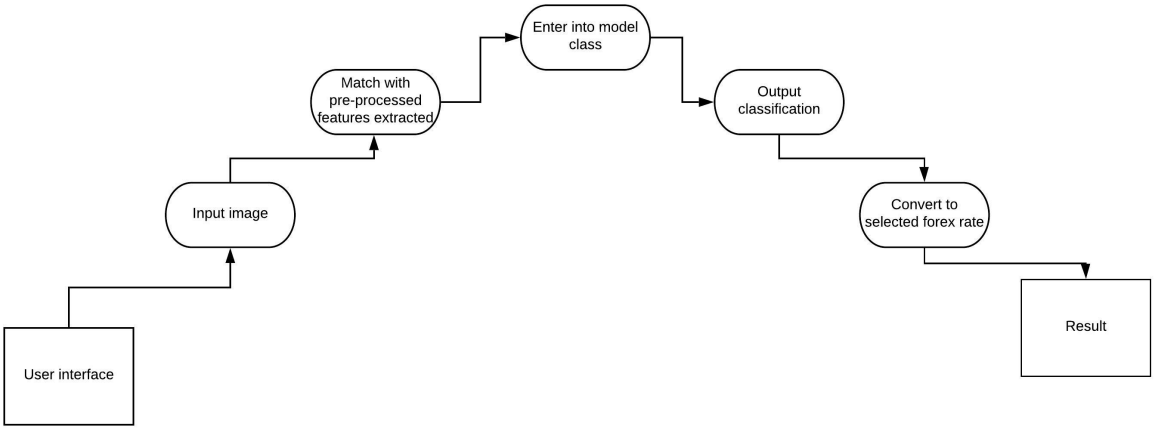


B.2 Data flow diagram

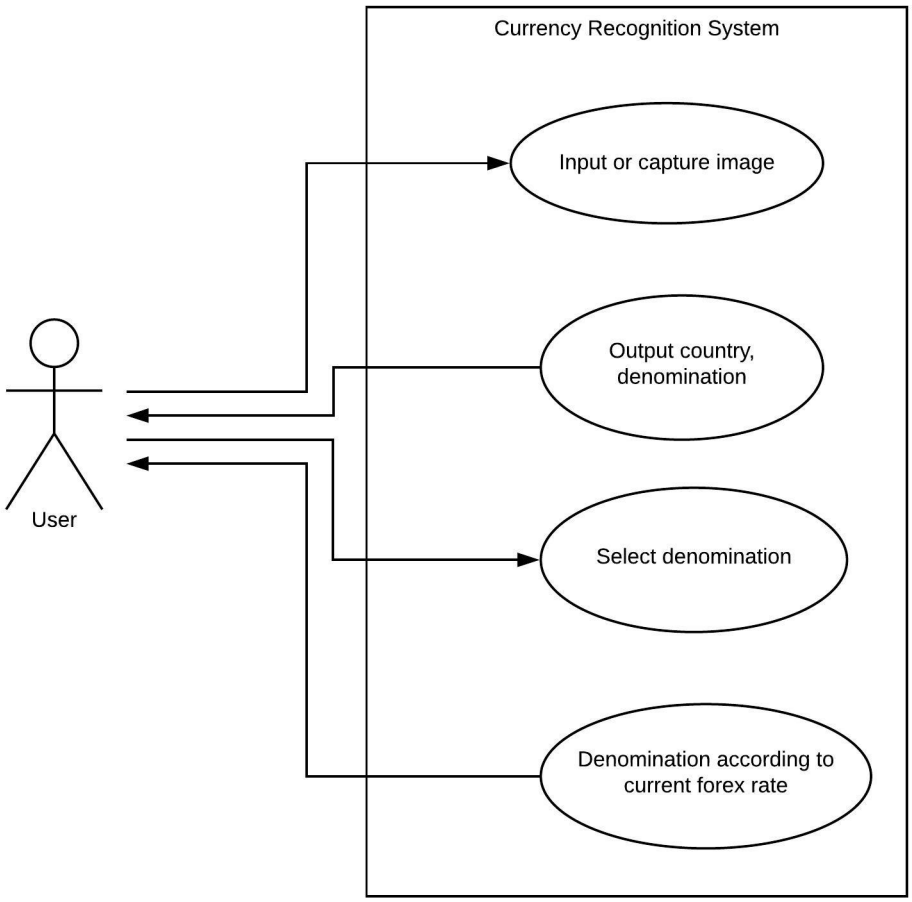
B.2.1 Level 0



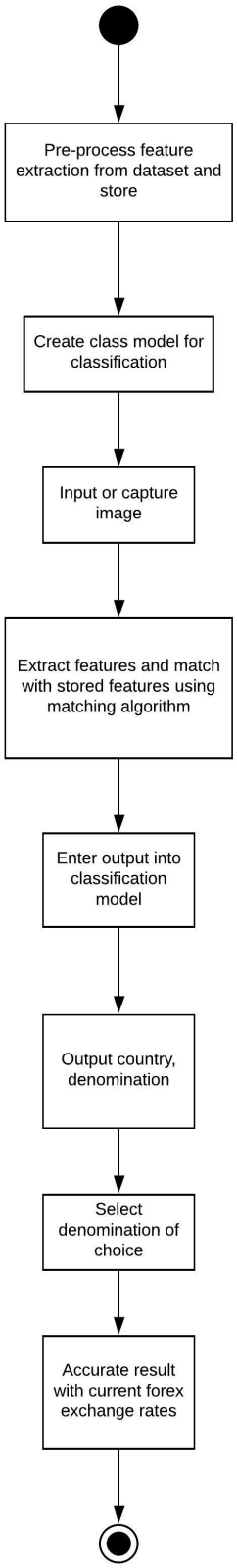
B.2.2 Level 1



B.3 Use case diagram



B.4 Activity diagram



Appendix C

Libraries

C.1 OpenCV

Version: 3.3.1

It is an open source computer vision library. It can run on LINUX as well as windows. It is a bulk library with a modular structure which includes many shared or static libraries. Modules:

- 2D Features Framework (features2d) - descriptors, and descriptor matching, salient feature detectors
- Image Processing (imgproc) - includes color space conversion, geometrical image transformations (generic table-based remapping, resize, affine and perspective warping), histograms, linear and non-linear image filtering and so on
- Object Detection (objdetect) - detection of predefined class objects (eg chair, people's faces, apple, bottle, car, etc)
- Video Analysis (video) - object tracking, background subtraction, motion estimation
- Video I/O (videoio) - video codecs and video capturing

- Core functionality (core) - defines basic data structures
- Camera Calibration and 3D Reconstruction (calib3d) - stereo correspondence algorithms, object pose estimation, multiple-view geometry algorithms, elements of 3D reconstruction, single and stereo camera calibration
- High-level GUI (highgui) - simple UI expertise

C.2 forex-python

Version: 1.5

Command : pip install forex-python

or clone repo

python setup.py install

Forex-python is a free python library. Its features include:

- Get historical rates for any day since 1999
- List of all currency rates
- Convert amount (10 USD to INR)
- Currency symbols
- BitCoin price for all currencies
- Currency names
- Conversion rate (eg USD to INR)
- Converting amount to BitCoins

C.3 Tkinter

Code:

```
import Tkinter
top = Tkinter.Tk()
top.mainloop()
```

It is a python library used to create standard GUI. Tk GUI toolkit has a powerful object-oriented interface because of Tkinter. Creating a GUI with Tkinter and bundled with python is very fast, easy and efficient:

- Import Tkinter library
- Design GUI main window
- Add widgets to the application
- Enter main event loop so that action can be taken for every triggered event

Various widgets that can be used in Tkinter are frame, button, listbox, menu, canvas, text, entry, spinbox, label, etc. Some of the standard attributes related to design are cursors, anchors, dimensions, fonts, colors, etc. The widgets have geometry management methods so that they can be placed according to the user's wish on the main window (eg left, right, fit window, etc). Advantage of this library is that it provides a robust and independent platform for windowing toolkit. It is a thin layer on top of Tcl/Tk and even though its documentation is very weak, good material is available to work on it (eg. tutorials, references, books, etc). Tk widgets are contrapted as Python classes as tkinter is a set of wrappers.

Appendix D

References

D.0.1 Research papers

[1] https://www.scribd.com/doc/36253092/Currency-Recognition-System-Using-Image-Processing

[2] https://acadpubl.eu/jsi/2017-114-7-ICPCIT-2017/articles/11/2.pdf[3] https:http://www.ipol.im/pub/art/2014/82/?utm_source=doi

[4] https://www.ijecs.in/index.php/ijecs/article/download/1277/1171/

[5] https://www.ijraset.com/fileserve.php?FID=7884

D.0.2 Image processing

[6] https://www.geeksforgeeks.org/digital-image-processing-basics/

[7] https://en.wikipedia.org/wiki/Digital_image_processing

[8] https://sisu.ut.ee/imageprocessing/book/1

D.0.3 OpenCV

[9] https://www.learnopencv.com/image-recognition-and-object-detection-part1/

[10] https://pythonprogramming.net/loading-images-python-opencv-tutorial/

[11] https://opencv.org/

[12] https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_tutorials.html

[13] https://medium.com/@ariesiitr/image-processing-with-opencv-45c3a5cefd10

D.0.4 Libraries

[14] https://pypi.org/project/forex-python/

[15] https://www.learnopencv.com/install-opencv3-on-ubuntu/