A

Course end Project report on

**Fake Currency Detection using Morphological Extraction**

**A6413-Fundamentals of Machine Learning**

Submitted by Shiva Nagi Reddy Pulicharla

**Course Instructor**

**Mrs. S.Sujana**

**Assistant Professor, Dept of ECE**



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING** VARDHAMAN COLLEGE OF ENGINEERING, HYDERABAD

Autonomous institute, affiliated to JNTUH

**2021 – 22**

**🡪Contents**

* Abstract
* Introduction
* Proposed Method
* Results
* Conclusion
* References
* Code

**1.Abstract**

Malpractising has always been a serious challenge that resulted in a serious problem in society. The automation in technology creates a more copied currency that is entirely spread, resulting in reducing the economic growth of the country. The note detection is compulsory, and also necessary to be very consistent and reliable. The paper currency identification depends upon a number of steps, including edge detection, feature extraction, image segmentation, grayscale conversion, and comparison of images. This paper also consists of a literature survey consisting of different methodologies for detection. The review to detect malpractice concludes that whenever we apply some efficient preprocessing and feature extraction techniques, it helps in improving the algorithm as well as the detection system. Machine Learning techniques help in building tools that is required and necessary for the research work, and we can make computer learning design, implementation, and methods to have a difference between fake and genuine currency.

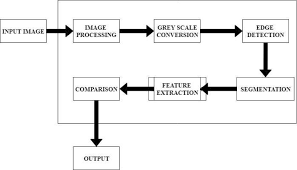
**2. Introduction**

* Technology is growing very fast these days. Consequently the banking sector is also getting modern day by day.
* This brings a deep need of automatic fake currency detection in automatic teller machine and automatic goods seller machine.
* Many researchers have been encouraged to develop robust and efficient automatic currency detection machine .
* Automatic machine which can detect banknotes are now widely used in dispensers of modern products like candies, soft drinks bottle to bus or railway tickets.
* The technology of currency recognition basically aims for identifying and extracting visible and invisible features of currency notes.
* Until now, many techniques have been proposed to identify the currency note. But the best way is to use the visible features of the note .
* For example, color and size. But this way is not helpful if the note is dirty or torn. If a note is dirty, its color characteristic are changed widely. So it is important that how we extract the features of the image of the currency note and apply proper algorithm to improve accuracy to recognize the note.
* Fake currency detection is a serious issue worldwide, affecting the economy of almost every country including India.
* Currency duplication also known as counterfeit currency is a vulnerable threat on economy. It is now a common phenomenon due to advanced printing and scanning technology.
* The possible solutions are to use either chemical properties of the currency or to use its physical appearance. The approach presented in this paper is based upon physical appearance of the Indian currency.
* Image processing algorithms have been adopted to extract the features such as security thread, intaglio printing (RBI logo) and identification mark, which have been adopted as security features of Indian currency.
* Hence, we propose a more user friendly and portable solution to this problem in form of an mobile app coupled with cloud storage.



**3.Proposed Method**

In proposed work, we will develop a system that would perfectly assess the features of fake note and real notes based on the paper by “Ms. Monali Patil, Prof. Jayant Adhikari, Prof. Rajesh Babu”. Our proposed system will be capable of performing real time detection of fake currency as we are using cloud storage for execution of our image processing logic simultaneously reducing the size of the smartphone app which plays crucial role in memory management of daily users. Also, our system will give live update of the identified currencies stock market values w.r.t other currencies around the globe.



For example, we have new Rs 500. We will discuss their feature one. First image is of Rs 500. Front – Side features. (These are the observe features) Rs 500 denomination bank notes are released in new series with inset letter ‘E’ in both the number panels and it also has the image of Mahatma Gandhi and signature of Governor Dr. Urjit R.Patel. First of all, we will discuss the features of Rs 500 notes. Color The color of Rs 500 note is stone gray. Size The new Rs 500 note is smaller than previous note (size of at 63mm x 150 mm). Bleed lines There are seven “bleed lines” on the side of Rs.2000 notes, and five lines on Rs 500 notes. Latent image In Rs 500 note there is latent image of the denomination in numeral. Denomination numeral in Devanagari font. Devanagari font is also used on the currency of Rs.500. Orientation In previous notes of Rs 500 and current notes there are some changes in orientation and relative position of Mahatma Gandhi. Windowed security thread Rs.500 notes contain a readable, fully embedded windowed security thread with the inscription “Bharat‟ (in Hindi), and “RBI‟. Which changes color from green to blue when note is tilted. Guarantee clause, Governor’s signature with promise clause and RBI emblem shifted towards right. Portrait The portrait has been changed in raised manner. The orientation and the portrait of Mahatma Gandhi has been changed and we can see the electrotype watermark also. Numerals In new currency notes numerals are mentioned in the increasing order of their size from top-left side to bottom right side. Ashoka pillar emblem On the right-side Ashoka pillar emblem is present. Special Features of New Currency

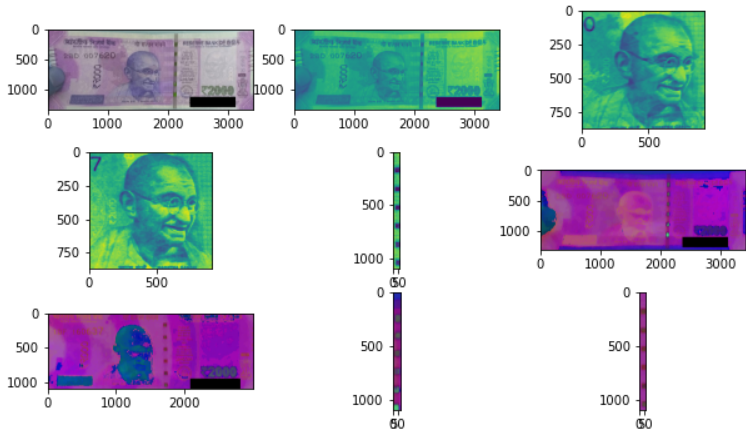
1. Swatch Bharat slogan with logo.
2. At the center there is a language panel.
3. 3.Red fort with Indian tri-color(flag).

4.At the right-side denomination numeral in Devanagari.

### **Methodology**

1. Image Acquisition: The image is provided to the model. There should be two images – a note which you want to detect and a corresponding real note.
2. RGB to GRAYSCALE: The RGB image is converted into a GRAYSCALE image.
3. Segmentation: The image is segmented to crop Gandhi Ji image and thin strip image.
4. Feature Measurement(**Morphological Extraction**): Feature measurement is done to measure the number of lines on a thin strip. This is a really lengthy process.
5. Finding Correlation: We find a correlation between Gandhi Ji’s image on the original note and a fake note. If the result is greater than 0.5 then we will consider it legitimate otherwise the currency is fake.
6. Classification: Finally, we will classify the image as real or fake.

**4.Results**



Observation :

* (1332,3416)  
  (1100,3000)

correlevance of transparent Gandhi>0.5

green strip is fake

**5.Conclusion**

In this Machine Learning project, we built a fake currency detection system. We used grayscale conversion, segmentation, and feature extraction under image processing in our system to compare real and fake notes. Currency use is a necessity for survival and hence it is always necessary to keep in track of its originality. Paper currencies are used much more in India and hence a system to detect the fake currency is needed. As the new currencies are used in the market, the proposed system seems to be useful to detect the currency to be genuine or not. This system compares more features for feature extraction than other proposed systems. It also shows where the differences are in the currencies instead of simply displaying the result. This system can be further implemented for foreign currencies like Dollars, Euros, Taka, etc. as a future scope.

**6. References**

* The requirement for this project is Python installed. You should also be familiar with libraries such as numpy and OpenCV. We will be using Jupyter Notebook for our project. You can use any other IDE for the project.
* The required modules for this project are –

OpenCV – pip install opencv-python

Numpy – pip install numpy

* The images we have used in this project are provided along with the fake currency detection project code in the following link: [**Fake Currency Detection Dataset & Code**](https://project-gurukul.s3.ap-south-1.amazonaws.com/machine-learning/fake-currency-detection.zip)

**7.Code**

import cv2  
import matplotlib.pyplot as plt  
import numpy as np  
A = cv2.imread(r'C:/Users/91944/Downloads/fake-currency-detection/Real.jpg')  
P = cv2.imread(r'C:\Users\91944\Downloads\fake-currency-detection\fake.jpg')  
plt. Subplot(531)  
plt.imshow(A)  
a = cv2.cvtColor(A, cv2.COLOR\_BGR2GRAY)  
p = cv2.cvtColor(P, cv2.COLOR\_BGR2GRAY)  
plt.subplot(532)  
plt.imshow(a)  
a2tr = a[330:1200, 1016:1927]  
plt.subplot(533)  
plt.imshow(a2tr)  
b2tr = p[170:1040, 716:1627]  
plt.subplot(534)  
plt.imshow(b2tr)  
print(a.shape)  
a2\_str = a[5:1100, 2080:2151]  
plt.subplot(535)  
plt.imshow(a2\_str)  
print(p.shape)  
p2\_str = p[5:1100, 1666:1729]  
plt.imshow(p2\_str)  
hsvImageReal = cv2.cvtColor(A, cv2.COLOR\_BGR2HSV)  
hsvImageFake = cv2.cvtColor(P, cv2.COLOR\_BGR2HSV)  
plt.subplot(536)  
plt.imshow(hsvImageReal)  
plt.subplot(537)  
plt.imshow(hsvImageFake)  
croppedImageReal = hsvImageReal[5:1100, 2080:2151]  
plt.subplot(538)  
plt.imshow(croppedImageReal)  
croppedImageFake = hsvImageFake[5:1100, 1666:1729]  
plt.subplot(539)  
plt.imshow(croppedImageFake)  
satThresh = 0.3  
valThresh = 0.9  
g = croppedImageReal[:,:,1]>satThresh  
h = croppedImageReal[:,:,2] < valThresh  
g1 = croppedImageFake[:,:,1]>satThresh  
h1 = croppedImageFake[:,:,2] < valThresh  
BWImageReal = g&h  
BWImageFake = g1&h1  
def bwareaopen(img, min\_size, connectivity=8):  
     
    # Find all connected components (called here "labels")  
    num\_labels, labels, stats, centroids = cv2.connectedComponentsWithStats(  
        img, connectivity=connectivity)  
     
    # check size of all connected components (area in pixels)  
    for i in range(num\_labels):  
        label\_size = stats[i, cv2.CC\_STAT\_AREA]  
         
        # remove connected components smaller than min\_size  
        if label\_size < min\_size:  
            img[labels == i] = 0  
             
    return img  
binr = cv2.threshold(a2\_str, 0, 255, cv2.THRESH\_BINARY+cv2.THRESH\_OTSU)[1]  
   
# define the kernel  
kernel = np.ones((3, 3), np.uint8)  
   
# invert the image  
invert = cv2.bitwise\_not(binr)  
   
# use morph gradient  
BWImageCloseReal = cv2.morphologyEx(invert, cv2.MORPH\_GRADIENT, kernel)  
binr2 = cv2.threshold(p2\_str, 0, 255, cv2.THRESH\_BINARY+cv2.THRESH\_OTSU)[1]  
   
# define the kernel  
kernel2 = np.ones((3, 3), np.uint8)  
   
# invert the image  
invert2 = cv2.bitwise\_not(binr2)  
   
# use morph gradient  
BWImageCloseFake = cv2.morphologyEx(invert2, cv2.MORPH\_GRADIENT, kernel2)  
areaopenReal = bwareaopen(BWImageCloseReal, 15);  
areaopenFake = bwareaopen(BWImageCloseFake, 15);  
bw = areaopenReal  
   
labels = np.zeros(bw.shape)  
countReal = cv2.connectedComponentsWithStats(bw, labels,8);  
bw2 = areaopenFake  
   
labels2 = np.zeros(bw2.shape)  
countFake = cv2.connectedComponentsWithStats(bw2, labels2,8);  
def corr2(A, B):  
     
    A\_mA = A - A.mean(1)[:, None]  
    B\_mB = B - B.mean(1)[:, None]  
   
    # Sum of squares across rows  
    ssA = (A\_mA\*\*2).sum(1)  
    ssB = (B\_mB\*\*2).sum(1)  
 # Finally get corr coeff  
    return np.dot(A\_mA, B\_mB.T) / np.sqrt(np.dot(ssA[:, None],ssB[None]))  
co=corr2 (a2tr, b2tr)  
   
if (co.any()>=0.5):  
    print ('correlevance of transparent gandhi > 0.5')  
    if (countReal[0] == countFake[0] ):  
        print ('currency is legitimate')  
    else:  
        print ('green strip is fake')  
else:  
    print ('correlevance of transparent gandhi < 0.5')  
    print ('currency is fake')

**THANK YOU**