I. **Title:** Paper Bill Color Detector

II. Abstract

The researchers have devised a MATLAB software to aid individuals with visual impairments in handling Philippine banknotes. The program utilizes a capturing device to take a photo of the bill, which is then processed through a digital image processing code that identifies the RGB and HSV values of the image. Based on these values, the program determines the value of the banknote and conveys it to the user through audio feedback. Through rigorous testing, the researchers have determined that a higher-quality capturing device is essential for flawless performance. In addition, in-depth knowledge of digital image processing and proficiency in MATLAB functions are crucial for minimizing errors, especially when dealing with multiple variables.

III. Introduction

Individuals with visual impairment often require assistance in their daily lives due to their disability. Various systems have been developed to aid these individuals and enhance their overall quality of life. To address the issue of visually-impaired individuals struggling with handling paper currency, researchers have created a program to assist them in this task. Despite the numerous technological advancements made, visually-impaired individuals, especially in the Philippines, are still left behind. Unfortunately, many people take advantage of their disability to commit fraudulent activities. Existing studies, such as those conducted by Sudeep [1] and Lang [2], have developed currency detection programs, but these are only limited to the currency of their respective countries. The researchers in this project aim to develop a currency detection program specifically for the Philippine currency. Using MATLAB and the concept of digital image processing, the program will detect the color of the bill and inform the user of its amount. By developing this program, visually-impaired individuals will become more self-sufficient and confident in handling money. Additionally, this project will raise awareness about the plight of people with disabilities and hopefully inspire other researchers to create or enhance programs/devices to improve their quality of life.

IV. Theoretical Consideration

1. Difference between RGB and HSV

Color processing, a fundamental concept in digital image processing, involves the utilization of RGB (red, green, blue) and HSV (hue, saturation, value) concepts. In the RGB model, an image is composed of three distinct image planes, each representing one of the primary colors. By manipulating the quantities of these color components, specific colors can be generated [3]. In addition to RGB, the HSV model also comprises three elements: hue, saturation, and value. Hue measures the wavelength associated with the dominant color perceived by the human eye [4], while saturation denotes the amount of white light mixed with the hue, influencing its intensity.

2. Color Segmentation

Color segmentation is the process of breaking down an image into its individual items or pieces. Edge detection, border detection, thresholding, and region-based segmentation are a few examples of common procedures. It has a number of color spaces, including the RGB and HSV, which have been widely used. The most popular color space, RGB, is straightforward and easy to understand, but it is extremely sensitive to variations in lighting. Due to their capacity to distinguish between color information and intensity as well as their resilience to variations in lighting, HSV and Lab color spaces are well-liked [5].

V. Methodology

For this project, the researchers will be utilizing image processing concepts available in MATLAB. The objective of the project is to provide aid to visually-impaired individuals by processing the image of a paper bill. The main concepts involved in the project are RGB, HSV, and programming loops. Although these concepts are relatively new to the researchers, they conducted research on MATLAB programming which provided valuable insight for the project. The project was able to operate properly with the help of MathWorks and the MATLAB community. The mentioned concepts were all implemented with some necessary modifications made to further optimize their functionality.

a. Initialization

The researchers utilized pre-existing images of Philippine banknotes and loaded them into MATLAB's Color Thresholder program. The HSV feature was then utilized to perform color segmentation, where the hue, saturation, and value (intensity) were adjusted to isolate the specific colors that distinguish a particular paper bill. For example, the colors of a Php 100 bill were separated to retain only the eye-visible purple tint. The HSV values obtained from this color-segmented image were converted into code by the researchers to determine threshold values. These code-derived values will be used in subsequent stages of the project.

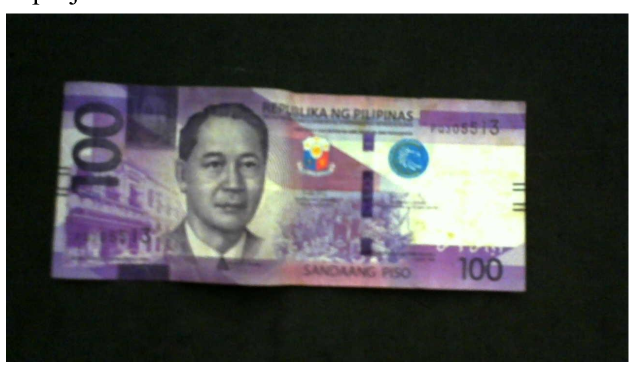

Figure 1. Pre-existing Php 100 banknote used for thresholding

Figure 2. Detectable pixels for Php 100 after thresholding

```
ThMin1h = 0.750;
TsMin1h = 0.105;
TvMin1h = 0.246;

ThMax1h = 0.853;
TsMax1h = 0.531;
TvMax1h = 0.887;
```

Figure 3. Threshold values for Php 100 in code form

b. Input

The webcam integrated into MATLAB will be used to capture the input. To enable this functionality, the researchers downloaded and installed the Image Acquisition Toolbox and MATLAB Support Package for USB Webcams, which were necessary add-ons for the program. The RGB values of the captured image will be extracted and then converted to HSV for precise color segmentation. The hue, saturation, and value (intensity) of the image will be isolated for further comparison.

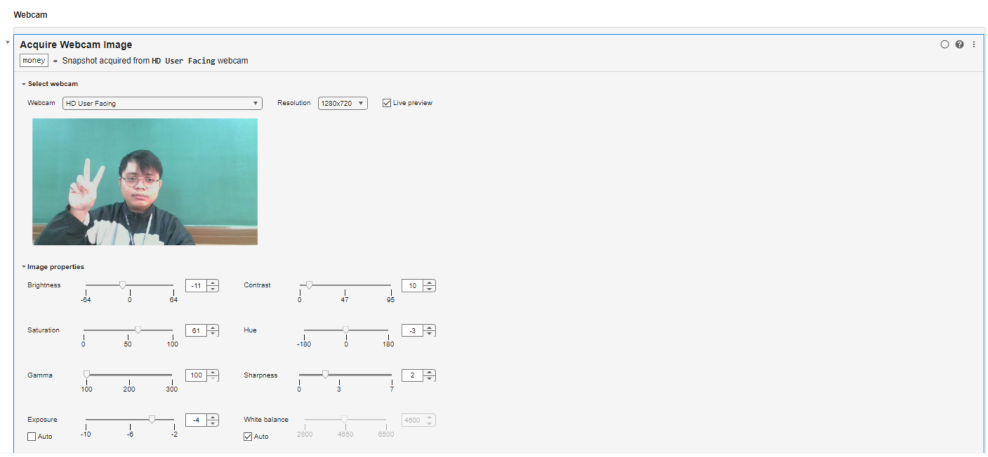

Figure 4. Webcam Function

RGB to HSV converter

```
hsvimg=rgb2hsv(money);
hue=hsvimg(:,:,1);
saturation=hsvimg(:,:,2);
value=hsvimg(:,:,3);
```

Figure 5. RGB to HSV Conversion

c. Comparison

The HSV values of the image are compared against the defined threshold for each bill using a combination of a for-loop function and if-else statements. In the first if-else statement, each HSV value is checked against the predetermined threshold values for each bill, thanks to the for loop function. The second if statement determines if all the HSV values match the threshold values and if so, the image is labeled with the corresponding color for that bill. Once the for loop completes, another if statement is used to compare the recorded colors and determine the dominant color for each bill, which indicates its monetary value.

```
for i=1:480
    for j=1:640

        %1k comparator
        if ((hue(i,j)>=ThMin1k)&(hue(i,j)<=ThMax1k))
            ch1k = 1;
            else
                ch1k = 0;
        end

        if ((saturation(i,j)>=TsMin1k)&(saturation(i,j)<=TsMax1k))
            cs1k = 1;
            else
                cs1k = 0;
        end

        if ((value(i,j)>=TvMin1k)&(value(i,j)<=TvMax1k))
            cv1k = 1;
            else
                cv1k = 0;
        end

        if ((ch1k == 1) & (cs1k == 1) & (cv1k==1))
            BP = BP + 1;
        end
```

```
%5H comparator
if ((hue(i,j)>=ThMin5h)&(hue(i,j)<=ThMax5h))
    ch5h = 1;
    else
        ch5h = 0;
end

if ((saturation(i,j)>=TsMin5h)&(saturation(i,j)<=TsMax5h))
    cs5h = 1;
    else
        cs5h = 0;
end

if ((value(i,j)>=TvMin5h)&(value(i,j)<=TvMax5h))
    cv5h = 1;
    else
        cv5h = 0;
end

if ((ch5h == 1) & (cs5h == 1) & (cv5h==1))
    YP = YP + 1;
end
```

```
%2H comparator
if ((hue(i,j)>=ThMin2h)&(hue(i,j)<=ThMax2h))
    ch2h = 1;
    else
        ch2h = 0;
end

if ((saturation(i,j)>=TsMin2h)&(saturation(i,j)<=TsMax2h))
    cs2h = 1;
    else
        cs2h = 0;
end

if ((value(i,j)>=TvMin2h)&(value(i,j)<=TvMax2h))
    cv2h = 1;
    else
        cv2h = 0;
end

if ((ch2h == 1) & (cs2h == 1) & (cv2h==1))
    GP = GP + 1;
end
```

```
%1H comparator
if ((hue(i,j)>=ThMin1h)&(hue(i,j)<=ThMax1h))
    ch1h = 1;
    else
        ch1h = 0;
end

if ((saturation(i,j)>=TsMin1h)&(saturation(i,j)<=TsMax1h))
    cs1h = 1;
    else
        cs1h = 0;
end

if ((value(i,j)>=TvMin1h)&(value(i,j)<=TvMax1h))
    cv1h = 1;
    else
        cv1h = 0;
end

if ((ch1h == 1) & (cs1h == 1) & (cv1h==1))
    PP = PP + 1;
end
```

```
%50 comparator
if ((hue(i,j)>=ThMin50)&(hue(i,j)<=ThMax50))
    ch50 = 1;
    else
        ch50 = 0;
end

if ((saturation(i,j)>=TsMin50)&(saturation(i,j)<=TsMax50))
    cs50 = 1;
    else
        cs50 = 0;
end

if ((value(i,j)>=TvMin50)&(value(i,j)<=TvMax50))
    cv50 = 1;
    else
        cv50 = 0;
end

if ((ch50 == 1) & (cs50 == 1) & (cv50 == 1))
    RP = RP + 1;
    end
```

```
        %20 comparator
        if ((hue(i,j)>=ThMin20)&(hue(i,j)<=ThMax20))
            ch20 = 1;
            else
                ch20 = 0;
        end

        if ((saturation(i,j)>=TsMin20)&(saturation(i,j)<=TsMax20))
            cs20 = 1;
            else
                cs20 = 0;
        end

        if ((value(i,j)>=TvMin20)&(value(i,j)<=TvMax20))
            cv20 = 1;
            else
                cv20 = 0;
        end

        if ((ch20 == 1) & (cs20 == 1) & (cv20 == 1))
            OP = OP + 1;
        end

        j=j+1;
    end
    i=i+1;
end
```

Figure 6. Pixel Color Comparator Code

d. Output

The application will produce the output by using MATLAB's text-to-speech feature after processing the image of the paper bill and giving it a monetary value. As a result, the results can be spoken out, which is very helpful for people with visual impairment.

OUTPUT

```
if ((YP>GP) & (YP>BP) & (YP>PP) & (YP>OP) & (YP>RP))
    disp('yellow')
    Speak(speaker,'The image you have scanned is five hundred philippine pesos')
end

if ((GP>BP) & (GP>YP) & (GP>PP) & (GP>OP) & (GP>RP))
    disp('green')
    Speak(speaker,'The image you have scanned is two hundred philippine pesos')
end

if ((RP>GP) & (RP>BP) & (RP>PP) & (RP>OP) & (RP>YP))
    disp('red')
    Speak(speaker,'The image you have scanned is fifty philippine pesos')
end

if ((PP>GP) & (PP>BP) & (PP>YP) & (PP>OP) & (PP>RP))
    disp('purple')
    Speak(speaker,'The image you have scanned is one hundred philippine pesos')
end

if ((OP>GP) & (OP>BP) & (OP>PP) & (OP>YP) & (OP>RP))
    disp('orange')
    Speak(speaker,'The image you have scanned is twenty philippine pesos')
end

if ((BP>GP) & (BP>YP) & (BP>PP) & (BP>OP) & (BP>RP))
    disp('blue');
    Speak(speaker,'The image you have scanned is one thousand philippine pesos')
end
```

Figure 7. Output Code with Speak function

VI. Results

For the results, the researchers will simulate how the flow the program will operate if the user was to input a picture of Php 100 bill. The researchers will first initialize the threshold of the paper bill using the Color Thresholder granted that there is pre-existing picture of the said banknote. Then, the user will input the picture with a Php 100 bill, the program takes its RGB values, converts it to HSV, runs the values through the for loop algorithm which is also the comparator, and lastly, the program will output the value of the bill based on the dominant pixel color audibly using the speak function. The embedded youtube video in the description showcases the results for the other the project starting at the timestamp of 10:50.

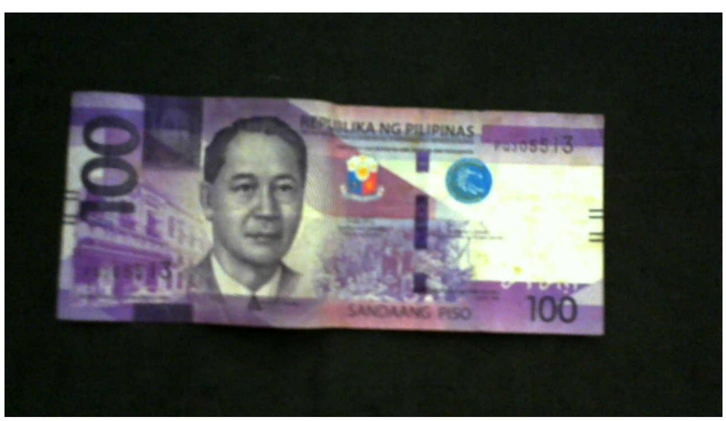

Figure 8. Pre-existing Php 100 banknotes used for thresholding

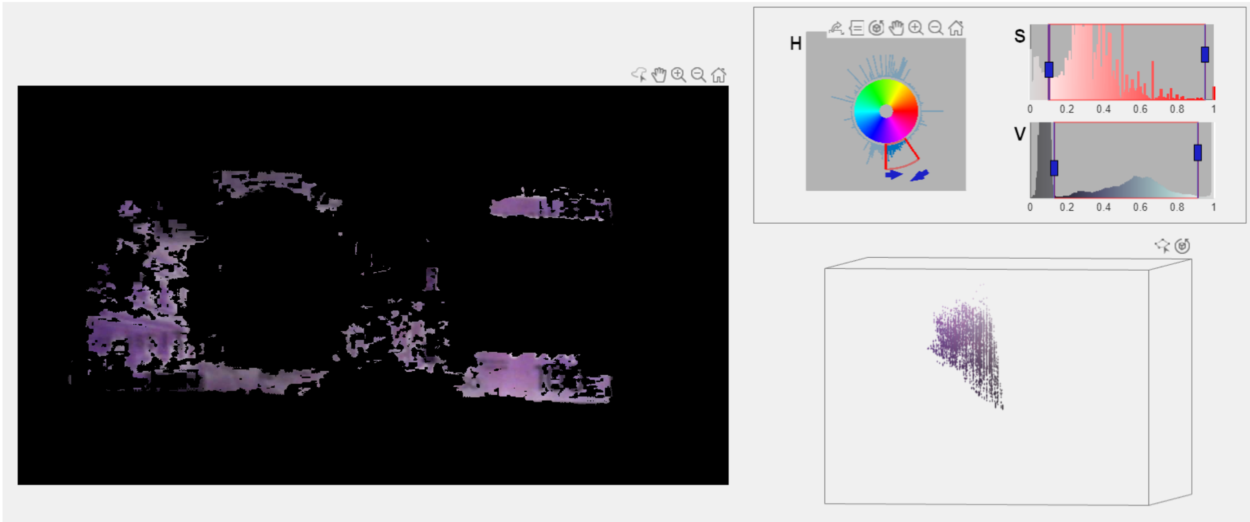

Figure 9. Color Thresholding

```
ThMin1h = 0.750;
TsMin1h = 0.105;
TvMin1h = 0.246;

ThMax1h = 0.853;
TsMax1h = 0.531;
TvMax1h = 0.887;
```

Figure 10. Threshold values for Php 100 in code form

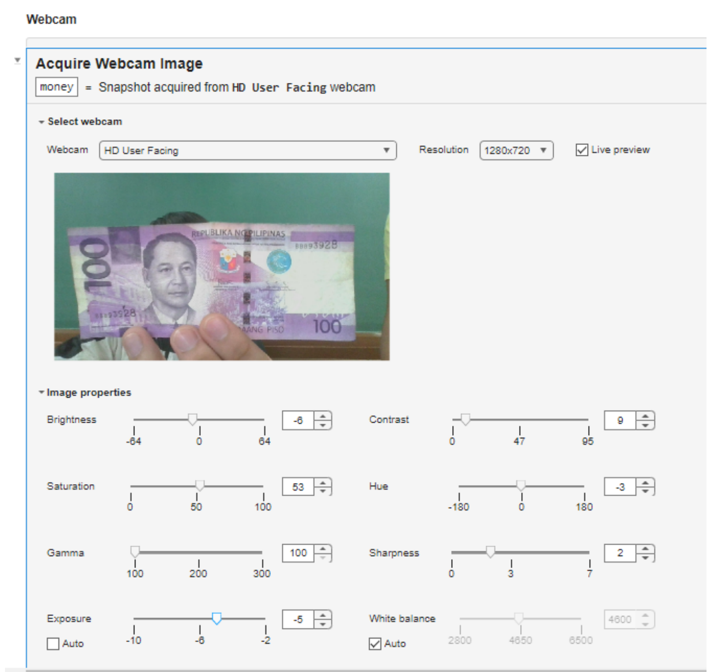

Figure 10. User inputting a picture

```
if ((PP>GP) & (PP>BP) & (PP>YP) & (PP>OP) & (PP>RP))
    disp('purple')
    Speak(speaker,'The image you have scanned is one hundred philippine pesos')
end
```

Figure 11. Expected resulting condition

VII. Discussion

The researchers will detail the process of acquiring the thresholds to the scanning to the outputting of the Php 100. First, a base picture of the pre-existing Php 100 bill will be scanned using the Color Thresholder app as seen in Figure 8. then the researchers manually identified the threshold in detecting only the purple pixel. Figure 9., the detectable pixels for a Php 100 can be seen and the program will strictly use this as a basis for the color comparator. This process is done for all available banknotes. The threshold codes for the Php 100 bill can be seen in Figure 10.

With the necessary thresholds integrated into the program, the program can now run as expected. The user may input a picture of a paper bill, as seen in Figure 10, in this case, the picture is that of a Php 100 bill. The program will take this as input, determine its RGB values, and convert these to HSV values. After this, these values will run through the for loop algorithm which is the comparator as seen in Figure 6.

In the algorithm, the resolution of the webcam is initialization with i being the number of vertical pixels and j being the number of horizontal pixels. For example, if the resolution of the camera is 640x480, i=480 and j=640 therefore the total number of pixels scanned will be 307200. It will now run through the for loop algorithm and utilize the 1h comparator, meaning that it will use the comparator for purple pixels which is the Php 100 bill. Per each pixel in the picture, if the purple shade for the hue, saturation, and intensity (value) is within the threshold of Php 100, it detects and confirms it as a purple pixel and increments the number of purple pixels. This is true for other pixel colors so for example, if a pixel is confirmed to be green, it increments the number of green pixels. This pixel count is stored in respective variables and the dominant pixel count is identified using this process.

In the output function, the dominant pixel count is compared with all the other pixel counts. In this case, since the input is that of a Php 100 bill, the dominant pixel will be purple therefore, the conditional statement for if the dominant pixel color is purple as seen in Figure 11. runs in the program and the program outputs the value of the bill by audibly outputting through the device speakers “The image you have scanned is one hundred Philippine pesos”.

Although the testing of the program almost has no errors, initially, the researchers found that the output can be inaccurate due to the parameters of the webcam such as brightness, saturation, contrast, etc. However, this issue has now been solved by getting the average parameter values for different webcams.

VIII. Conclusion

The researchers were able to successfully implement the code accurately, however, errors in the initial testing phase were encountered due to the webcam parameters. To compensate for this, the researchers ran the program through different devices with different third-party and integrated webcams. The average for these parameters is applied to the program and this solution was observed to be relevant as the results were all determined to be accurate and the program did not output wrong bill values.

A possible recommendation for this project is the development of an app where its color-segmentation-based algorithm may be applied for it to be fully realized as a tool for visually impaired people. To conclude, this project was a successful implementation of a prototype that advocates for helping people with optical disabilities.

The researchers successfully created a software program as part of their project, which assists individuals with visual impairments in managing and recognizing Philippine currency. By utilizing digital image processing techniques and incorporating RGB and HSV concepts into the MATLAB code, the program is able to accurately identify and provide information to the user about the denomination of the bill.

IX. Author's Contribution

Chavez, Julian Carlos	Paper (Abstract, Introduction, Theoretical Considerations)
Escamilla, Philip Martin Emmanuel	Code (Conditional statements), Data collection, Results
Payuyo, John Louie	Code (webcam function, text-to-speech function, thresholding), Project testing, Paper (Methodology, Results, Discussion, Conclusion)

X. References

[1] "DSpace at my university: Home - 14.99.188.242:8080." [Online]. Available: http://14.99.188.242:8080/jspui/. [Accessed: 10-April-2023].

[2] L. Lang, N. F. Gazcón, and M. Larrea, "An open source solution for money bill recognition for the visually impaired user using smartphones," CIC Digital - Home, 01-Oct-2018. [Online]. Available: https://digital.cic.gba.gob.ar/handle/11746/8991. [Accessed: 10-April-2023].

[3] Informatics Homepages Server. [Online]. Available: https://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/OWENS/LECT14/lecture12.html#:~:text=In%20the%20RGB%20model%2C%20an,of%20the%20primary%20components%20present. [Accessed: 10-April-2023].

[4] Jeefri A Moka, "Image classification with HSV Color Model Processing," Data Science Central, 16-Oct-2017. [Online]. Available: https://www.datasciencecentral.com/image-classification-with-hsv-color-model-processing/. [Accessed: 10-April-2023]

[5] Matt Maulion, "Color Image Segmentation - Image Processing" Medium, 1-Geb-2021. [Online]. Available: https://mattmaulion.medium.com/color-image-segmentation-image-processing-4a04eca25c0. [Accessed: 10-April-2023]