

A Novel Color Detection Model

Selin HACIOĞLU, Tahsin ÜSTÜNEL, Gökçe NUR YILMAZ *

¹ TED University, Faculty of Engineering, Computer Engineering, Turkey

Abstract

In today's world, the consumption of clothing and accessories is increasing day by day. In this consumption area, people sometimes have difficulty in finding a color outfit or jewelry they want. Especially in face-to-face shopping, people look at all the stores to find the color they are looking for and consume a lot of energy in those stores. Considering this increasing problem, a model is proposed in this study that detects color and then displays the images of the clothing and accessory data it contains according to the color it perceives. In this way, this model provides ease of finding what people are looking for. Therefore, it could be easily stated that this system turns the filtering process into a kind of visual filtering. The results derived using the proposed model prove its effectiveness for detecting the colors.

Keywords: *Clothes; Color Detection; Filtering.*

1. Introduction

In today's conditions, clothing and accessories consumption is increasing. This increased consumption can make people to have difficulty in finding a color outfit or jewelry they require which in return causes a lot of energy and time consumption. Considering this problem, a model detecting the color of the clothes and accessories required and then displaying the images of the clothing and accessory data it contains according to their color is developed in this study. It could be easily stated that color is the significant feature for our proposed model. Human eyes and brain work in coordination to convert light into color, and light receptors in the eyes transmit the signal to the brain, which in turn recognizes color. Adapting this strategy to the machines is the key factor in identifying the color names in the developed model and fulfilling its main task [1][2]. To express the color of some pictures, Red, Green, Blue (RGB) based color recognition system [3] is used. In this system, each color always consists of 3 values between 0 and 255 followed by RGB [3]. In the literature, there are some models relying on this system, for instance; Lee et. al. [4] applied the color identification technique for the NOKIA smart mobile device. This system uses phone cameras to recognize colors. This system's disadvantage is to only use three basic colors (red, green, blue) [5]. Duth et.al. shade and model images in the color detection system proposed in [3].

Contrary to the existing systems in the literature, using RGB in our proposed model is not efficient. The reason behind this is that it is desired and aimed to detect the color of a currently displayed image and to show its color in our model. Various filters have been tried in our model to make the RGB method compatible with the system, for example, Laplacian However, the desired yield could not be obtained. For the system to obtain images in the desired direction, the HSV in OpenCV method is used in our proposed study. There are red, green, and blue colors in the HSV method but in addition to these there are H, S, V. In HSV, color can be changed with 'H', saturation can be changed with 'S', value can be changed with 'V' [6]. This makes it easier to adapt the system to real life. Another reason for using this method is that the CCD camera used to define color in more complex systems is not used. The abbreviation for this camera is device with charging connection. Cameras with such image sensors perform well in low light conditions [7]. But in the proposed system, the webcam is used to detect the object. In the presented project, using the HSV method, the color ranges and the color of the objects are determined more efficiently.

In our proposed model, another important factor other than color is to save the data in the system and display it on the screen. The filing method was used to make this happen. With this simple and time-saving filing (dataset) method, the desired images are displayed on the screen according to the colors. As a result, a webcam is used to model real-life objects, the colors are determined by the HSV method to detect the colors of these objects and display on the screen, and a window is created using some libraries in Python. Colors can be printed on the screen with OpenCV. Finally, the requested data is displayed with the file system.

The remainder of the article is organized as follows. The model proposed is introduced in the second section. The proposed model is discussed in section 3. In section 4, the results are presented. Finally, in section 4, conclusions and considerations for future work are presented.

2. Proposed Model

Our proposed model's framework can be visualized in Figure 1. As can be observed from the framework, OpenCV and NumPy libraries have an important place in the working algorithm of our proposed system. The reason for using these libraries is that they are useful for image processing. NumPy is also widely used for multidimensional arrays and mathematical operations, which are frequently encountered in image processing.

As can also be observed from Figure 1, after importing the OpenCV and NumPy libraries, a video capture method is created in our proposed framework. The reason for this is to be able to capture and detect the images in the video recording. While creating this, the size of the frame of the capture screen is also adjusted. After this stage, methods are defined to detect colors. Following these defined methods, it is necessary to open a color palette to define the colors. However, before the Color palette is defined, the HSV technology, which is used to better detect the color of the objects, is created. By looping this technology, HSV ranges are defined, and masks are created. Later, the created management system is tested with every color registered. After this stage, the palette for the colors to be used in the system is opened using the cv2 color method. Then, it is time to determine how the image we want to be detected would be detected by the camera. For this, a point is specified in the middle of the screen. The camera is defined as a round value and set to the middle of the camera. In this way, the objects shown to the point appearing in the middle of the screen can be detected. After detecting the object with the camera, the process of detecting colors is started. Numbers between 0 and 255 corresponding to the colors are used here. The range is determined for each color. Colors corresponding to numbers could be defined thanks to the cycles defined in the system. After determining the color of the detected objects, the putText method is used to display the detected colors on the screen. When it comes to the final stage of the project, it is requested to display a visual on the screen according to the colors. For this, images are displayed on the screen by using the circle method. Finally, the system is terminated with the release and destroyAllWindows methods so that the video from which the video is taken will not continue.

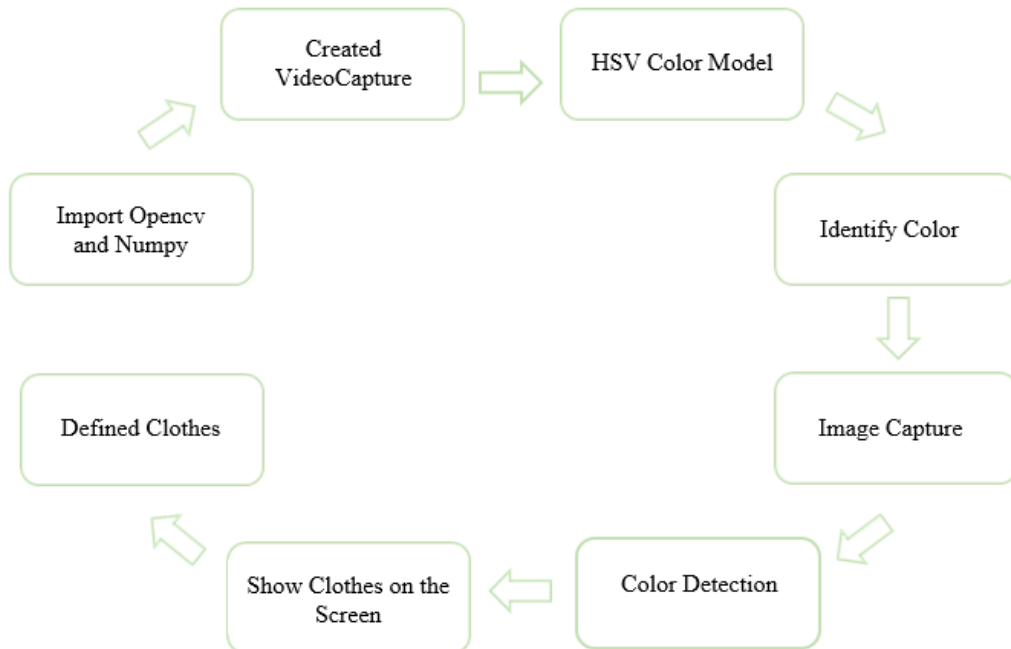


Figure 1. The framework of the proposed model

In addition to all these processes, the idea of transferring data from files is studied when displaying the data as seen in Figure 2. Then, a dataset is created as seen in Figure 3. The pictures that are requested to be included in this created data set are positioned according to their colors and then called from the system. In this way, the desired images are displayed on the screen when the colors are detected.

blue	25.05.2022 00:19	Dosya klasörü
green	25.05.2022 01:29	Dosya klasörü
orange	25.05.2022 01:43	Dosya klasörü
red	25.05.2022 00:19	Dosya klasörü
violet	25.05.2022 01:45	Dosya klasörü
yellow	25.05.2022 01:40	Dosya klasörü

Figure 2. *Filling Method*

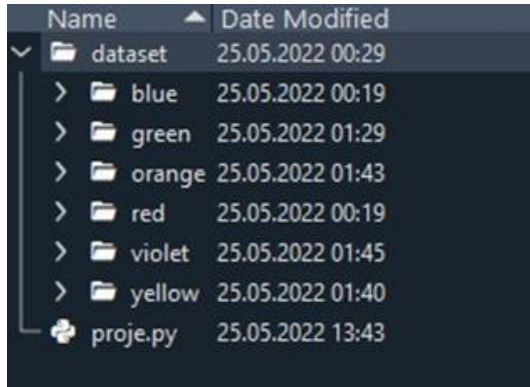


Figure 3. *The dataset in the proposed system*

3.Results and Discussions

In order to derive results, the RGB method is first used for the color detection. With this method, different and inefficient results are obtained for our system as exemplified in Figure 4. For example, as seen in Figure 4, when run for orange color, data [0 94 254] is received. So 'Green' is 94. In this method, each pixel has the same value. Although there is no problem with still images, it is not useful as the aim of the project is to capture real-life images at that moment. Because this method simply mixes the colors and identifies the presence of each by showing the resulting color. So, RGB method is converted to the HSV method to obtain results with our proposed method.

```
import cv2
img = cv2.imread("orange_background.webp")
print(img)
cv2.imshow("Tma", img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```



Figure 4. The end obtained from the RGB method

In our proposed model, another important thing to mention is the camera point, which plays an important role in perceiving the color of the image. The biggest disadvantage with this is that the color detection capacity of the camera is different due to the image quality. For example, it may detect a distant white object as blue due to camera quality. Therefore, it can actually display images of unwanted colors. For example, when it detects a white wall in a distant and illuminated environment, it can print it in blue on the screen. This is a negative result we get. However, to minimize this and get efficient results, we convert RGB to HSV and we receive the results we want. The results screen of our proposed model can be seen in Figure 5. With the HSV Color Space method used later, we use a filtering method so that the system can more easily focus on the object whose color is desired to be detected. In this way, the screen is blacked out of the object and become more efficient. Color-focused results can be observed in Figure 6 and Figure 7. As a result, when we bring an object other than Camera quality to the screen, the system works properly, with our last try, it focuses on the object better, the color of the piece corresponding to the point is displayed on the screen, and the relevant pictures are displayed on the screen. The result is about the registered colors we add to our filling system.

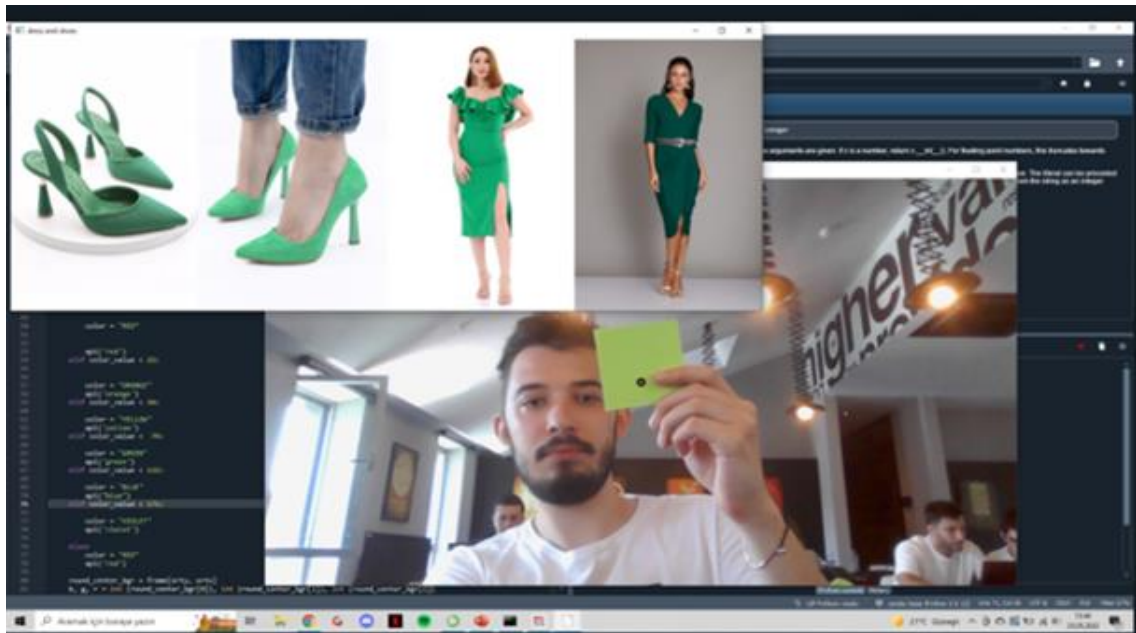


Figure 5. First trial screen

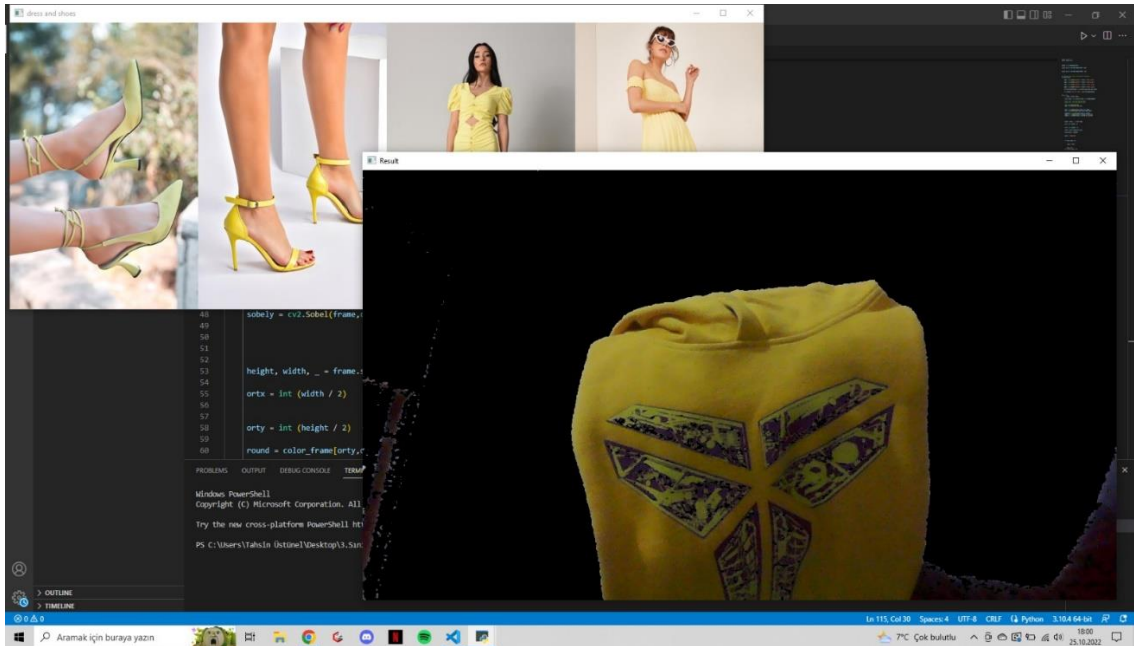


Figure 6. Color-focused result with HSV

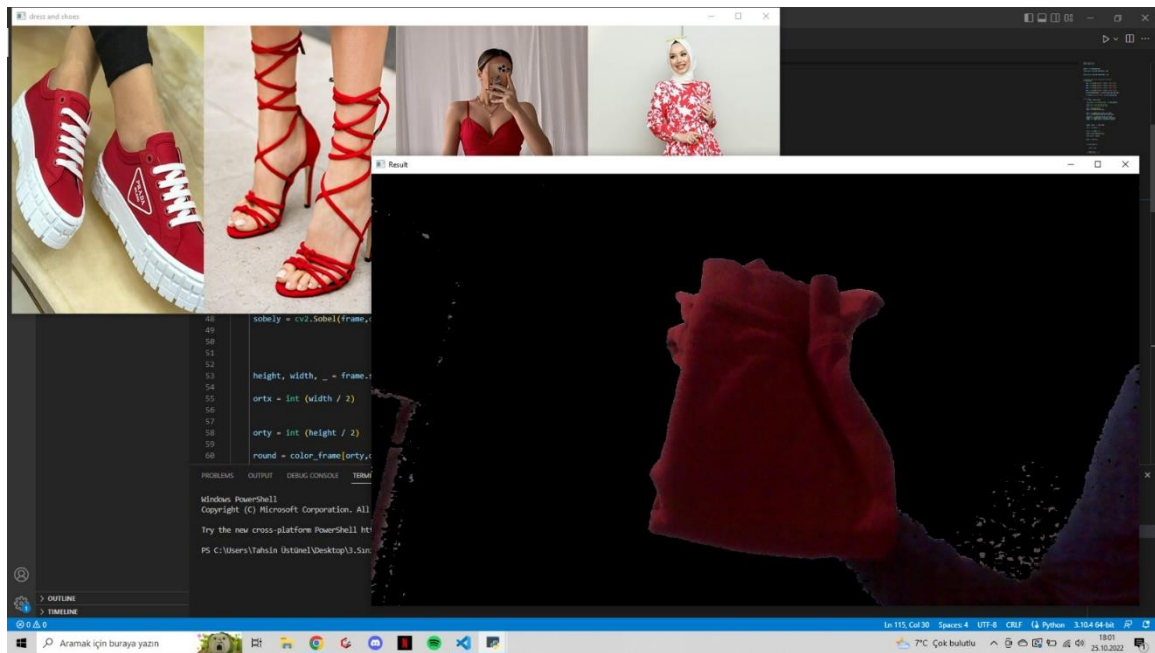


Figure 7. Color-focused result with HSV

4. Conclusions and Future Work

In this study, a model that brings images of clothes and accessories according to the desired colors has been proposed. The wealth of the Python-OpenCV libraries have been used in the proposed study. In addition, the differences and usage areas of the methods used in color perception and the areas where they are efficient have attracted attention. Along with the results of the proposed model, different color detection models and filtering processes are also observed to prove the effectiveness of the proposed model. The results of the proposed model have been observed as desired compared to the different color perception models.

There are many ways that can be done in the future to make this app useful in real life. For this application, high resolution cameras that are more sensitive and adaptable to environmental conditions can be used. In order to expand the color range, more color ranges can be added to the system and tested one by one. In addition to these, the data system can be supported with database systems so that it can be integrated into a more complex system. Thanks to this database support, it can even be transformed into a system that proposes combinations for users who use the application. At the same time, different filter features can be tried to add color to the application. Like outlining the clothes other than dimming screen. As a result of all these, it can be turned into a system that can be used in stores in the future. In other words, a user who shows the color he wants to the camera can see all the clothes with that color and even the content that is compatible with it and can easily choose it.

Declaration of Interest

The authors declare that there is no conflict of interest

Acknowledgements

An earlier version of this paper was presented at the ICADA 2022 Conference and was published in its Proceedings (Title of the conference paper: “Color Detection with Clothes”).

References

- [1] Gokce Nur, H Kodikara Arachchi, Safak Dogan, Ahmet M Kondoç, “Seamless video access for mobile devices by content-aware utility-based adaptation,” *Multimedia Tools and Applications*, vol. 70, 2014.
- [2] DVSX De Silva, Gokce Nur, Erhan Ekmekcioglu, A Kondoç, “QoE of 3D Media Delivery Systems,” *Media Networks: Architectures, Applications, and Standards*, CRC Press Taylor and Francis Group, 2012.
- [3] Google Inc., “What does RGB even mean? Find out this and more with a brief overview of what RGB is and how you can use it in your web designs.” 2022. [Online]. Available: <https://www.shutterstock.com/blog/rgb-definition-design-work>. [Accessed 13 June 2022].
- [4] Hsiao Ping Lee, Jun-Te Huang, Chien-Hsing Chen*, Tzu-Fang Sheu “Building a Color Recognizer System on the Smart Mobile Device for the Visually Challenged People” *The Sixth International Multi-Conference on Computing in the Global Information Technology –2011*
- [5] P. Sudharshan Duth, M. Mary Deepa, “Color Detection in RGB-modeled images using MATLAB” *Research Paper-2018*
- [6] Hüseyin Bilal Macit, “Python ve OpenCV ile RGB Üzerinde HSV Renk Kodu Tespiti” 2020. [Online]. Available:<https://www.hbmacit.com/2020/09/15/python-ve-opencv-ile-rgb-uzerinde-hsv-renk-kodu-tespiti/>. [Accessed 13 June 2022].
- [7] CCD information: <https://www.guvenlikonline.com/makale/413/ccd-kamera.html>