SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V11I5P102

Volume 11 Issue 5, 12-19, May 2024

Original Article

Design of IoT Based Real-Time Color Recognition and Mixing Machine

Layak Ali[*1], Susmitha Dokur[1], A. Santhosha Kumar[2]

[1]*Department of Electronics and Communication Engineering, School of Engineering, Central University of Karnataka, Karnataka, India.*
[2]*Department of Electrical Engineering, School of Engineering, Central University of Karnataka, Karnataka, India.*

[*]*Corresponding Author : layakali@cuk.ac.in*

Received: 01 March 2024 Revised: 03 April 2024 Accepted: 01 May 2024 Published: 29 May 2024

Abstract - *Many industries and research organizations aim to design a color mixing system to increase efficiency and productivity and minimize human efforts. This paper presents the design of a fully automated color recognition and regeneration of an object's color. Further, it is augmented with remote operation. The designed prototype has a facility for manual operation to generate the required color by selecting various color options locally. It has sensors which detect the color of an object and generates the same color. These operations are controlled remotely via Internet of Things (IoT), where object color is detected remotely, and the same color is generated locally. Further, to meet the real-time constraints and parallel execution, the FreeRTOS is employed in the prototype. The prototype is built using cost-effective Arduino-based hardware and tested with multiple colors sensing and generation methods. The results suggest that the prototype is suggestive and can be made as a fine product with minor enhancements.*

Keywords - *IoT, Color recognition, Color mixer, FreeRTOS, ThingSpeak.*

1. Introduction

The color of an object improves its recognition and visualization and further plays an important role in human society and evolution [1, 2]. The three primary colors will produce many different colors and shades by mixing with appropriate proportions. Getting the required color and shade is very important and has lots of applications. There are many areas of human life where we need efficient color-mixing strategies [3, 4].

Many industries and research organizations are involved in designing a color mixing system to increase productivity and reduce human efforts. There are various kinds of color mixing strategies that can be seen in the literature [3-16] and [17-25]. The work presented in [5-7] shows the only object identification process. The work in [8] shows how Arduino and Matlab are used for color generation. The work in [9] shows PLC and HMI interface. Similarly, the works [10-26] show colour generation from primary colours. Though there are many color mixing schemes available [8-26], there is still a requirement for a cost-effective, fully automated, with parallel working and controlled remotely through the Internet.

This paper presents the design of a fully automated color recognition and regeneration prototype. The prototype supports manual selection and object color detection with a sensor. Further, the prototype has extended its operation to control and operate remotely via the Internet of Things (IoT) platform. It uses the IoT and Mobile App for remote access and automation. The parallel computation and real-time constraints are achieved by incorporating a Real-Time Operating System (RTOS) called FreeRTOS in the prototype. The prototype is built using cost-effective, open-source Arduino-based hardware and tested with multiple colors sensing and generation methods. The results suggest that the prototype is suggestive and can be made as a working product with minor enhancements.

The paper is arranged and represented as follows: Section 2 presents the Methodology and state of the art, Section 3 presents the design of prototypes, Section 4 presents the detailed results, and the paper is concluded in Section 5.

2. Methodology and State-of-the-Art

The world around us is very colorful, with lots of colors and shades. Thus, equivalent digital color visualization is an important area. However, modern technologies need a certain way of representing colors. Thus, color models have emerged. The color model describes how various colors are generated from primary colors. Different color models are being developed for various applications and each differs slightly from the other. The complete set of colors being generated by a specific color model is said to be color space. The simplest and most feasible color model is the RGB color model.

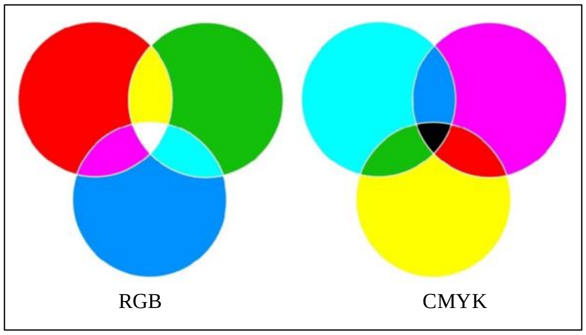

Fig. 1 RGB and CMYK models [30]

The RGB color model deals with basic colors like Red, Green, and Blue (RGB), and with an appropriate mixer of these colors, it generates lots of other secondary colors. as shown in Figure 1. The RGB color model is easy to understand and implement. It can give almost 16 million colors, which can be seen on the screen.

However, the colors generated by this model cannot be printed. Hence, another model called the CMYK color model is preferred. This model is very popular for printing devices. In the CMYK model, the colors are generated through the subtraction of the RGB model base colors from the white color. The primary colors in the CMYK model are Cyan, Magenta, and Yellow.

Kalskaya et al. have defined the word color model and presented a review of different color models, their descriptions, and their comparison [4]. An appropriate way of representing color is required to use color as a visual signal in multimedia, image processing, graphics, and computer vision applications. It discusses RGB, CMYK, HSV, and Lab color models as well as their areas of application [4]. Jose et. al. have addressed mainly two issues in [17]. The first is how color order systems relate to various kinds of chromatic mixtures, and the second is how opacity and transparency in the coloring media affect the resulting results beyond the additive, partitive, and subtractive mixtures.

Further, color mixing is an important process wherein different colors are mixed with different proportions and get the desired color. There can be two methods, viz. additive and subtractive color mixing. In additive color mixing, red, blue, and green play an important role, while in subtractive essential colors are yellow, maroon, and cyan, as shown in Figure 1.

The color mixing process can be traced back to the 13th century [1]; further details can be found in [2]. A comprehensive color model and computer visualization details can be found in [3]. The work in [5] presents a new color model for image and video processing. Further details of image processing may be found in [6, 7]. The mathematical model for color mixing and controlling using PI controllers can be found in [9, 10]. The work presented in [16] shows how automatic paint mixing can be implemented using software, especially in LabView. The work presented in [11] explains a color mixing system designed using a Programmable Logic Controller (PLC) for syrup production. The periodic task of liquid mixing and packing was automated and implemented for bottle filling using PLC [12]. Further, only bottle-filling automation can be found in [13] and [14]. The work presented in [15] deals with smooth color variations using a single layer “plasmonic” color pixel and a method for “nanophotonic structural” color mixing based on the RGB color model.

The authors in [18] proposed a new method of light mixing red, green, and blue lights for mixing, but this proposed method contains four colors that are red, green, blue, and white. Most of the research work presented uses hexadecimal, RGB, or CMYK values for color regeneration. The work presented in [19] used a color sensor to sense the color of an object, and accordingly, RGB colors were dispensed from their compartments with the help of a syringe. The work in [19] has emphasized mechanical parts, movements and additive colors (RGB) for mixing. Abdalftah H. Mohammed et al. used a Human-Machine Interface (HMI) for color selection [20]. It mixes primary (RGB) colors with a base to obtain the required color manually defined by the user and then pumps it through pipes into the painting machine. The output colors and their RGB proportions are defined beforehand and can be selected by HMI manually.

The authors in [8] designed a color mixing machine that is fully automated based on Arduino. The color selection is done manually through the Matlab graphical user interface. The automatic color mixing plant is presented in [21], which is generic and can be operated by non-technical users also. This work also generates a color by mixing the predefined ratio of basic colors that need to be selected with HMI. The final generated color is verified by using the color sensor. Ali Srour in [22] has implemented a color mixing machine with a PIC microcontroller. The work in [22] facilitates the user by providing the facility to choose a special color within a specific amount and time. A pneumatic conveyor robot for color detection and sorting is presented in [23]. It uses a color sensor to detect the color of the object and has a mechatronic conveyor system for transportation. The real-time environment monitoring system using ESP8266 and the ThingSpeak IoT platform is implemented in [24].

The authors in [25] developed a robotic arm with a color sensor and servo motor, a color sensor for sorting, and a servo motor for pickup and putting objects in the container. There are various color regeneration systems and strategies available which are based on additive or subtractive color strategies. However, there is a requirement for cost-effective, parallel processing, a fully automated and remotely controlled system for color recognition and generation.

3. Proposed Prototype Design

The following section describes the hardware and software used in the prototype design.

3.1. Hardware Details

The prototype is built using the open-source Arduino platform shown in the figure below. The details of Arduino UNO and its applications can be found in [27-29].

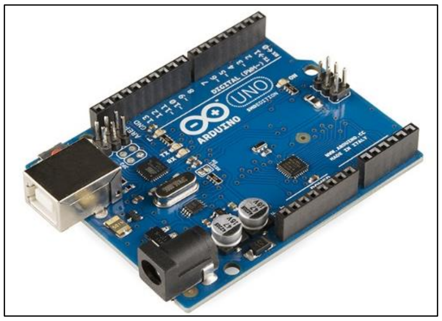

Fig. 2 Arduino Uno
(Source: https://en.wikipedia.org/wiki/Arduino_Uno)

The color pumping is done through a 5V-operated DC pump, as shown in the below figure.

Fig. 3 DC pump 5v
(Source: https://www.mr100rb.store/)

The relay modules, as shown in the below figure, are used for controlling the color pumps ON and OFF actions.

Fig. 4 Relay module 5V DC
(Source: https://www.superbtech.in/)

The following keypad of 3x3 is used in select colors for a local color generation.

Fig. 5 Keypad 3x3
(Source: http://sriembeddedsolutions.com/)

3.2. Software Details

The software used for developing applications is Arduino IDE and BlynkApp. For parallel processing, FreeRTOS [31] is integrated, and for remote access, ThingSpeak [32] cloud service is used. The circuit development and simulation are carried out on Proteus software. The following sections give the detail implementation circuit and simulation setup.

Table 1 shows the proportions of RGB colors for getting the required color. Similarly, Table 2 gives the percentage and amount of color to be mixed in the base color to get the required color. Further, it gives the theoretical timing requirement for corresponding DC pumps to operate. Further, Table 3 gives practical timing requirements of corresponding DC pumps.

Table 1. RGB decimal code

S. No.	Colour	RGB Decimal Code
1	Maroon	(128,0,0)
2	Purple	(128,0,128)
3	Sky Blue	(135,206,235)
4	Indigo	(75,0,130)
5	Orange	(255,165,0)
6	Violet	(238,130,238)
7	Navy	(0,0,128)
8	Pink	(255,192,203)
9	Chacolate	(98,52,18)

Table 2. RGB decimal code and time requirement

Colour	RGB Decimal Code	RGB in %	RGB in ml	Pump Timing (ms)
Maroon	(128,0,0)	(50,0,0)	(17,0,0)	(251,0,0)
Purple	(128,0,128)	(50,0,50)	(17,0,17)	(251,0,251)
SkyBlue	(135,206,235)	(52,80,92)	(18,28,31)	(266,414,458)
Indigo	(75,0,130)	(29,0,50)	(10,6,17.3)	(148,89,256)
Orange	(255,165,0)	(100,64,0)	(33.33,22,0)	(493,326,0)
Violet	(238,130,238)	(93,50,93)	(31.7,17,31.7)	(469,251,469)
Navy	(0,0,128)	(0,0,50)	(0,0,17)	(0,0,251)
Pink	(255,192,203)	(100,75,79)	(33.33,25,27)	(493,370,400)
Chocolate	(98,52,18)	(38,20,7)	(13,7,3)	(192,100,44)

Table 3. Theoretical and practical time requirements

Colour	RGB Decimal Code	Theoretical Timing (ms)	Practical Timing (ms)
Maroon	(128,0,0)	(251,0,0)	(251,50,0)
Purple	(128,0,128)	(251,0,251)	(150,50,250)
Sky blue	(135,206,235)	(266,414,458)	(100,400,600)
Indigo	(75,0,130)	(148,89,256)	(150,300,500)
Orange	(255,165,0)	(493,326,0)	(500,100,0)
Violet	(238,130,238)	(469,251,469)	(100,200,500)
Navy	(0,0,128)	(0,0,251)	(10,10,350)
Pink	(255,192,203)	(493,370,400)	(500,50,50)
Chocolate	(98,52,18)	(192,100,44)	(250,250,200)

The three basic prototypes were designed that is based on the RGB color model, and finally, all were combined for remote control and parallel operations.

3.3. RGB-Based Prototype: Keypad

This prototype is based on an RGB color model and has a human interface. The interface is provided through a keypad to select a required color for color regeneration locally. The water pumps are used to dispense colors, which are connected to Arduino through relays with 10, 11 and 12 pins of Arduino, as shown in Figure 6.

The pumps dispense Red, Green, and Blue colors in the required quantities taken from Tables 1, 2 and 3. As per the circuit diagram, every key has been assigned a specific color. When the key is pressed, the appropriate delay of color mapping will be assigned to DC pumps. The required delays for turning ON the DC pumps are calculated as shown in Table 3. The corresponding color DC pumps will dispense colors as per the delays. This gives the required target color.

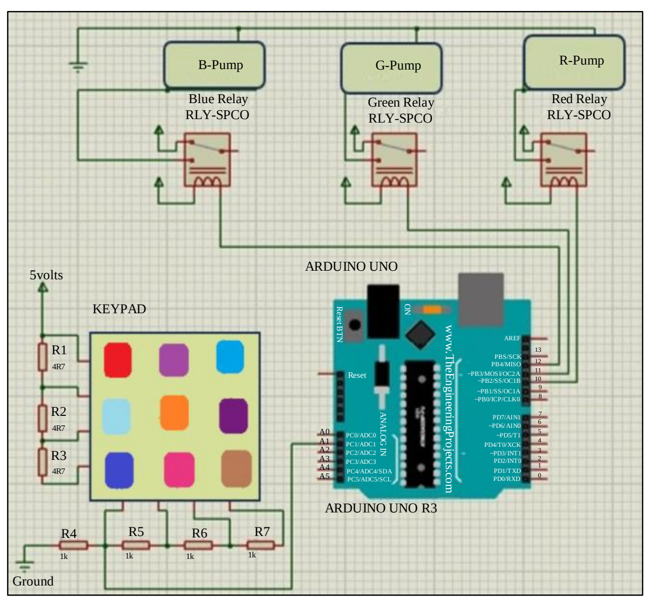

Fig. 6 RGB-based prototype with keypad

3.4. RGB-Based Prototype: Sensor

This prototype is again based on the RGB color model, and it has the Sensor interfaced (namely TCS3200). The separate color pallets were prepared to test the design. These pallets will be sensed by the color sensor when they are brought under the sensor area. The details of color sensor TCS3200 are shown in Figure 7. Figure 8 shows the sensor-based prototype.

The color sensor is connected to pins 5 to 9 of Arduino; the relays are controlled by pins 10, 11, and 12 of Arduino, which runs the submersible pumps. These DC pumps dispense Red, Green, and Blue colors in required quantities to obtain the desired color as defined by the user. This prototype uses a color sensor to detect the object color and gives RGB values based on wavelength; according to the RGB values range, appropriate delays will be assigned to pumps to dispense the RGB colors into the mixing container to produce the target color as per the requirement. The nine color pallets were prepared to test the prototype.

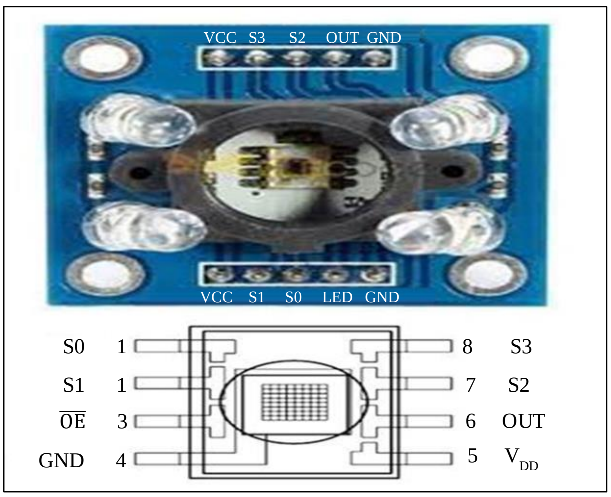

Fig. 7 Color sensor and pins of TCS3200

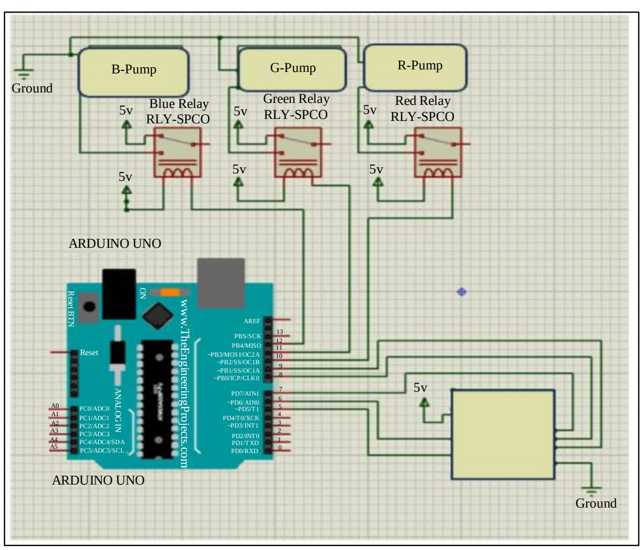

Fig. 8 RGB-based prototype with sensor

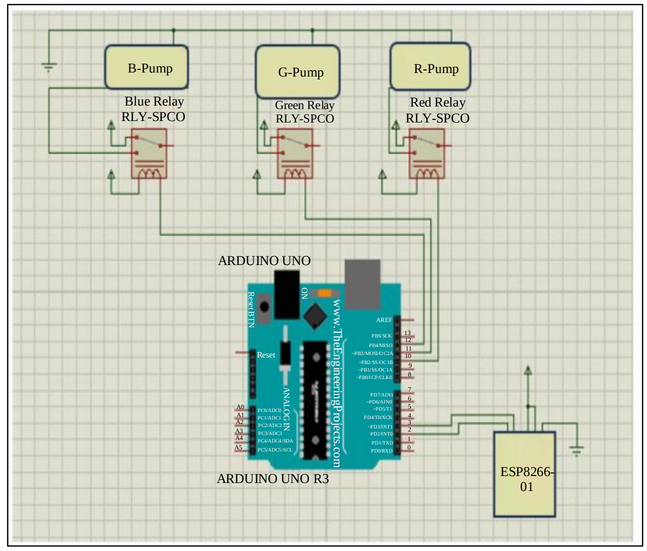

Fig. 9 RGB-based prototype with IoT

3.5. RGB-Based Prototype: BlynkApp

This prototype is augmented to the above-designed prototypes and is implemented for remote operation. The circuit in Figure 9 shows that the RGB-based prototype with the "BlynkApp" and Wi-Fi is an IoT-based module, i.e., the device can be controlled from anywhere in the world. The Wi-Fi is configured using the ESP8266 module.

The Wi-Fi module is connected to Arduino through Tx and Rx pins for serial communication. The relays are controlled by 10, 11, and 12 pins of Arduino, which runs pumps. The "BlynkApp" is customized to the prototype and implemented. The slider facility of "BlynkApp" is used for deciding about the required color. This information is sent over the ThingSpeak cloud and received by the main module, which generates the required color.

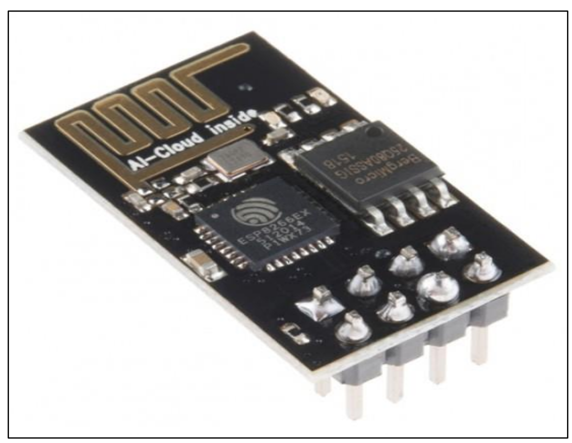

Fig. 10 ESP8266 module
(Source: https://techdelivers.com/)

This prototype comprises two modules; one senses the pallet color and sends it to the cloud, and the other receives information from the cloud and generates equivalent color.

3.6. Complete Prototype

The final circuit diagram and working model of the prototype with all the options mentioned above are shown in Figures 11 and 12, respectively. Figure 11 shows the circuit, and the implemented module is shown in Figure 12.

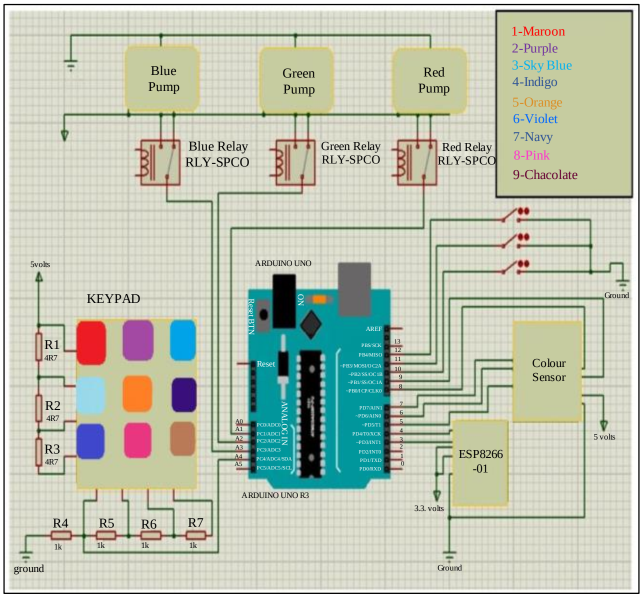

Fig. 11 Circuit diagram of the complete model

Fig. 12 Final working model

4. Results and Discussion

The results are presented in detail for all three prototypes in the following section. The results of all the proposed prototypes are shown in Figures 13 to 15. To check the color regeneration, nine reference colors were used, which are shown beside the results image from Figures 13 to 15. Figure 13 shows the results obtained with Keypad.

Here, each key corresponds to a unique color; in this prototype, only nine keys are used for nine reference colors. Once the key is pressed, the appropriate corresponding color will be generated, as shown in Figure 13. Figure 14 shows the color obtained with the color sensor module. The color pads are prepared with a specific color. These nine color pads, one after the other, are placed in front of the color sensor. The color sensor detects the color of the pad, and the corresponding color will be generated, as shown in Figure 14.

Fig. 13 Result with keypad model

The third prototype is based on IoT and implemented using a mobile App (BlynkApp). In the App, the color pallets are made for each color (nine). These color pallets are pressed one after the other, and the corresponding color will be regenerated on a remote machine. The reproduced color through the IoT platform is shown in Figure 15.

Fig. 14 Result with color sensor model

Fig. 15 Result with IoT model

5. Conclusion

The design of the color mixing system is of paramount importance. There are lots of systems available for color mixing; however, there is still a requirement for a cost-effective, real-time and remotely controlled system. This paper presents the detailed design of an IoT-based and real-time colour detection and regeneration system. The prototype is implemented using cost-effective and open-source hardware, viz-Arduino platform. The prototype is designed and implemented for the RGB color model. A user can press the key and get the required color.

A user can show the object color to the system, and it generates the same color. All three prototypes were tested thoroughly for generating various colors. The results are presented lucidly. The results suggest that, with little modifications, a working color recognition and mixing system can be developed for real-time applications.

References

[1] Hannah E. Smithson et al., "A Three-Dimensional Color Space from the 13th Century," *Journal of the Optical Society of America A*, vol. 29, no. 2, pp. A346-A352, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[2] Eric Kirchner et al., "Color Theory and Color Order in Medieval Islam: A Review," *Color Research and Application*, vol. 40, no. 1, pp. 5-16, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[3] Noor A. Ibraheem et al., "Understanding Color Models: A Review," *ARPN Journal of Science and Technology*, vol. 2, no. 3, pp. 265-275, 2012. [Google Scholar] [Publisher Link]

[4] K.Y. Aleksandrovna, *Overview of Various Color Models of Computer Graphics*, LLC Publishing House 'Azhur', pp. 538-543, 2020. [Google Scholar] [Publisher Link]

[5] Rashad J. Rasras, Ibrahiem M.M. El Emary, and Dmitriy E. Skopin, "Developing a New Color Model for Image Analysis and Processing," *Computer Science and Information Systems*, vol. 4, no. 1, pp. 43-55, 2007. [CrossRef] [Google Scholar] [Publisher Link]

[6] Konstantinos N. Plataniotis, and Anastasios N. Venetsanopoulos, *Color Image Processing and Applications*, Springer Berlin, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[7] Barry R. Masters, Rafael C. Gonzalez, and Richard Woods, "Book Review: Digital Image Processing, Third Edition," *Journal of Biomedical Optics*, vol. 14, no. 2, 2009. [CrossRef] [Google Scholar] [Publisher Link]

[8] S. Brindha et al., "Automated Color Mixing Machine Using Arduino," *International Journal of Engineering Research and Technology*, vol. 6, no. 4, pp. 1-4, 2018. [Google Scholar]

[9] Laurentiu Schiop, and Marian Gaiceanu, "Mathematical Modelling of Color Mixing Process and PLC Control Implementation by Using Human Machine Interface," *2010 3rd International Symposium on Electrical and Electronics Engineering (ISEEE)*, Galati, Romania, pp. 165-170, 2010. [CrossRef] [Google Scholar] [Publisher Link]

[10] Majdi Salman, "*Modelling and Experimental Investigation of Paint Mixing Process Dynamics*," Thesis, Master of Mechatronics, Palestine Plytechnic University, pp. 1-68, 2020. [Google Scholar] [Publisher Link]

[11] Husam Salih, Hammam Abdelwahab, and Areej Abdallah, "Automation Design for a Syrup Production Line Using Siemens PLC S7-1200 and TIA Portal Software," *2017 International Conference on Communication, Control, Computing and Electronics Engineering (ICCCCEE)*, Khartoum, Sudan, pp. 1-5, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[12] Mini Sreejeth, and Shilpa Chouhan, "PLC Based Automated Liquid Mixing and Bottle Filling System," *2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES)*, Delhi, India, pp. 1-5, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[13] A.W. Pradana, and M. Somantri, "Bottle Filling Simulation System Using Programmable Logic Control," *IOP Conference Series: Materials Science and Engineering*, vol. 384, pp. 1-7, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[14] K. Thiyagarajan, R. Meenakshi, and P. Suganya, "Vision Based Bottle Classification and Automatic Bottle Filling System," *2016 International Conference on Advances in Human Machine Interaction (HMI)*, Kodigehalli, India, pp. 1-3, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[15] Claudio U. Hail et al., "A Plasmonic Painter's Method of Color Mixing for a Continuous Red-Green-Blue Palette," *ACS Nano*, vol. 14, no. 2, pp. 1783-1791, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[16] Mohamed A. Muftah, Abdulgani M. Albagul, and Abdullah M. Faraj, "Automatic Paint Mixing Process Using LabView," *Mathematics and Computers in Science and Industry*, vol. 31, pp. 233-238, 2014. [Google Scholar] [Publisher Link]

[17] José Luis Caivano, "Color Order Systems, Color Mixtures, and the Role of Cesia," *Color Research and Application*, vol. 46, no. 6, pp. 1169-1179, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[18] Hui Ren et al., "Study on LED Color Mixing for Stage Lighting Based on Locus Fitting of Blackbody," *2018 IEEE 3rd International Conference on Cloud Computing and Internet of Things (CCIOT)*, Dalian, China, pp. 177-181, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[19] Tapas Raj, and Ashirvad Jena, "*Design and Development of an Automated Paint Mixing Machine*," Thesis, National Institute of Technology, Rourkela, pp. 1-38, 2014. [Google Scholar] [Publisher Link]

[20] Abdalftah Hamed Mohammed Ali, "Design and Implementation of Color Mixing and Painting Automatic Machine," *Engineering Sciences and Technology*, vol. 10, no. 1, pp. 13-19, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[21] Sanjay Kumar et al., "Automatic Color Mixing Plant," *International Journal of Advances in Science, Engineering and Technology*, vol. 6, no. 4, pp. 15-19, 2018. [Publisher Link]

[22] Oussama Moustafa, and Ali Srour, "Automated Color Mixing Machine Based on Microcontroller," *Manufacturing Engineer*, 2022.

[23] Mohammadreza Lalegani Dezaki et al., "A Pneumatic Conveyor Robot for Color Detection and Sorting," *Cognitive Robotics*, vol. 2, pp. 60-72, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[24] Debapriya Parida et al., "Real-Time Environment Monitoring System Using ESP8266 and ThingSpeak on Internet of Things Platform," *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, Madurai, India, pp. 225-229, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[25] Gregorio Imanuel Efraim Panie, and Achmad Benny Mutiara, "Development of Robotic Arm for Color Based Goods Sorter in Factory Using TCS3200 Sensor with a Web-Based Monitoring System," *2018 Third International Conference on Informatics and Computing (ICIC)*, Palembang, Indonesia, pp. 1-6, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[26] Dhivya Priya E.L. et al., "IoT Based Compact, Low-Cost Dyeing Machine for Medium Scale Industrial Environment Applications," *Research Square*, pp. 1-18, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[27] Docs, Hardware, UNO R3. [Online]. Available: https://docs.arduino.cc/hardware/uno-rev3/

[28] Dlnya Abdulahad Aziz, "Design of Smart House System Based on Webserver Architecture Control," *International Journal of Computer Applications*, vol. 181, no. 8, pp. 43-49, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[29] Mostefa Kermadi, El Madjid Berkouk, and Ali Ben Achour, "Overview of Arduino Development Platform and Its Applications in Control Systems," *The 2nd International Conference on Power Electronics and their Applications (ICPEA)*, 2015. [Google Scholar]

[30] Vilma Kriauciunaite-Neklejonoviene, Juraj Muzik, and Stanislav Hodas, "Visual Orthophoto Classification Application," *IOP Conference Series: Earth and Environmental Science*, vol. 906, pp. 1-5, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[31] FreeRTOS, Real-Time Operating System for Microcontrollers. [Online] Available: https://www.freertos.org/

[32] ThingSpeak. [Online]. Available: https://thingspeak.com/