ISSN: 2641-8835

Madridge Journal of Bioinformatics and Systems Biology

Research Article

Open Access

Color Blindness Correction using Augmented Reality

Sasikumar Gurumurthy[1]*, Renuka Devi Rajagopal[2] and **Anand AjayAshar[2]**

[1]*Department of Computer Science and Systems Engineering, Sree Vidyanikethan Engineering College, Tirupati, Andhra Pradesh, India*
[2]*Department of Software Engineering, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India*

Article Info

***Corresponding author:**
Sasikumar Gurumurthy
Professor
Department of Computer Science and Systems Engineering
Sree Vidyanikethan Engineering College
Tirupati, Andhra Pradesh
India
Tel: 8870919338
E-mail: mithrangurugsk@gmail.com

Received: February 28, 2019
Accepted: April 12, 2019
Published: April 22, 2019

Citation: Gurumurthy S, Rajagopal RD, AjayAshar A. Color Blindness Correction using Augmented Reality. *Madridge J Bioinform Syst Biol.* 2019; 1(2): 31-33.
doi: 10.18689/mjbsb-1000106

Published by Madridge Publishers

Abstract

To alter the hue of an image in real-time to correct color blindness using a mobile application.

Keywords: Augmented Reality; Colors; Hue shifts; Realtime.

Introduction

Augmented Reality provides a real-time world environment and allows the viewers to interact with game live. This happens with the help of various augmented factors such as audio, visual, computer graphics and even global positioning input. Augmented reality synchronizes the environment with the graphical structure to provide an ultimate virtual reality gaming experience.

Using the same technology, we can alter the saturation of an image in real time to print the correct color that can be perceived as it is by a colorblind person

Method and Procedures

The color correction application is built for an android device using Unity 2017.3.0f 3. Unity allows the developers to build different games, scenes, environments, augmented reality scenes, virtual reality scenes, etc. Vuforia package is used to access the augmented reality camera. Vuforia augmented reality is configured in the environment [1].

There exists a hierarchy in the Unity environment where the game objects are visible. The hierarchy of the Unity environment includes an Augmented Reality camera, and a canvas, that embeds the user interface components (Figure 1).

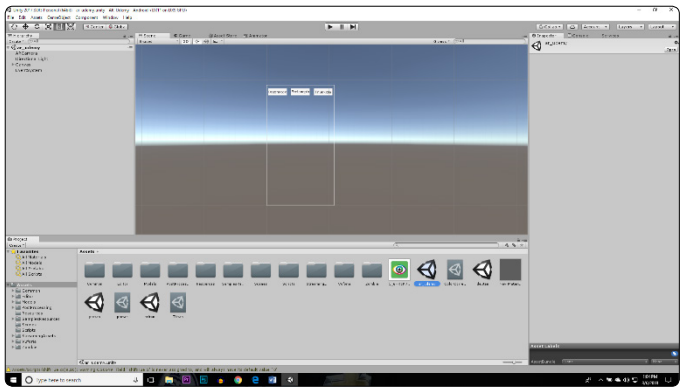

Figure 1. Main Menu.

User interface components include three buttons that loads filters for the three opias (Deuteranopia, Protanopia and Tritanopia) (Figures 2-4). Each button, on clicking changes the scene. Each scene has different color filters and the scene includes additional sliders to twitch the hue within the set range [2].

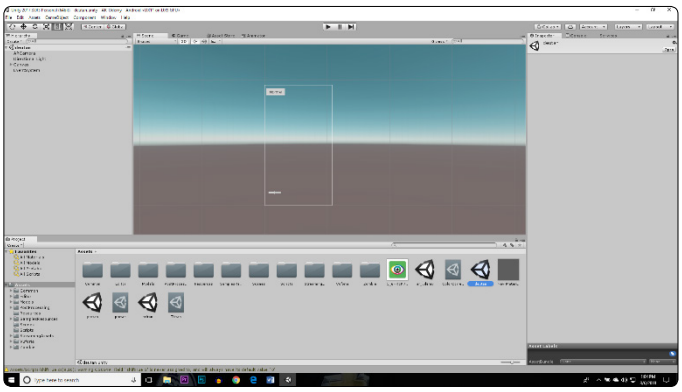

Figure 2. Deuteranopia Scene.

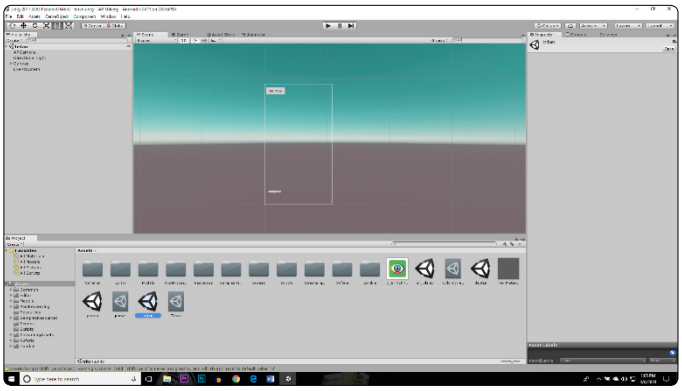

Figure 3. Tritanopia Scene.

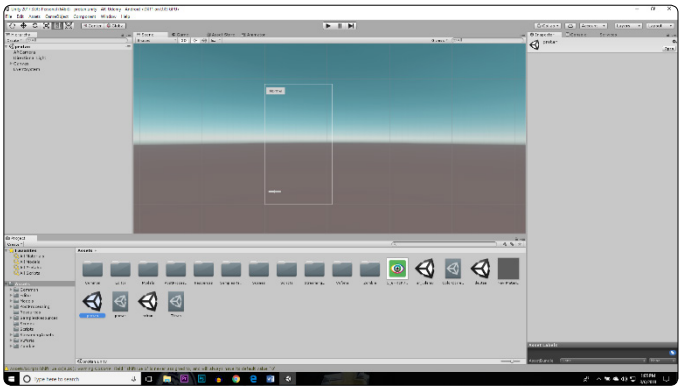

Figure 4. Protanopia Scene.

The application hosts a post processing utility tool that is available on Unity asset store. The post processing utility tool is an additional package that gives us full control to adjust the graphics of any scene in unity. Some of its attributes are color grading, anti-aliasing, chromatic aberration, motion blur, and many others.

Hue shift is an attribute of color grading [3]. A post processing profile is created. Also, an additional script is created that loads the created post processing profile to access the color grading component.

The sliders on canvas (for individual scene) control the hue. The additional script created to control the post processing profile is linked with the sliders that give us the control to change hue on the user interface (Figure 5).

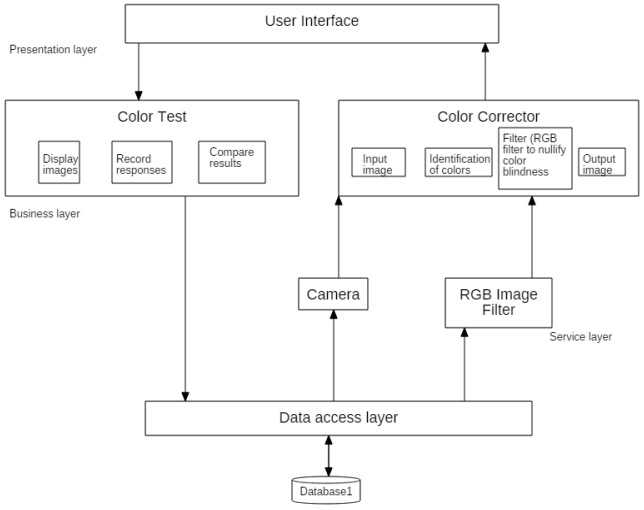

Figure 5. Architecture diagram.

All the scripts are embedded in to the AR Camera game object [4]. The sliders are linked with the script that fetches the hue from the post processing utility tool. Initial hue levels are set and a range is calculated for the sliders (after testing).

Testing and Analysis

Unit testing is shown in table 1 and integration testing is shown in table 2.

Table 1. Unit Testing.

Test Case Id	Test Scenario	Test Steps	Pre Condition	Input Data	Expected Output	Actual Output	Priority	Status
01	To add User Information	Enter the type of color blindness	Type of color blindness is known	Deuteranopia	Deuteranopia filters	Deuteranopia filters	High	Pass
02	To test the color correction filters	Hue shift	Color filters loaded	Slide shifter	Corrected image	Corrected image	High	Pass
03	To test the color correction filters	Hue shift	Color filters loaded	Slide shifter	Corrected image	Unreadable results	High	Fail
04	To revert to normal image	Hit the normal vision button	Loaded filters	Normal	Normal vision	Normal vision	Medium	Pass
05	To revert to normal image	Hit the normal vision button	Loaded filters	Normal	Normal vision	Error, System crash	Medium	Fail
06	To add User Information	Enter the type of color blindness	Type of color blindness is known	Deuteranopia	Deuteranopia filters	Protanopia filters	High	Fail
07	To add User Information	Enter the type of color blindness	Type of color blindness is known	Protanopia	Protanopia filters	Protanopia filters	High	Pass
08	To test the color correction filters	Hue shift	Color filters loaded	Slide shifter	Corrected image	Corrected image	High	pass
09	To test the color correction filters	Hue shift	Color filters loaded	Slide shifter	Corrected image	Unreadable results	High	fail

10	To revert to normal image	Hit the normal vision button	Loaded filters	Normal	Normal vision	Normal vision	Medium	pass
11	To add User Information	Enter the type of color blindness	Type of color blindness is known	Tritanopia	Tritanopia filters	Deuteranopia filters	High	Fail
12	To add User Information	Enter the type of color blindness	Type of color blindness is known	Tritanopia	Tritanopia filters	Tritanopia filters	High	Pass
13	To test the color correction filters	Hue shift	Color filters loaded	Slide shifter	Corrected image	Unreadable results	High	Fail
14	To test the color correction filters	Hue shift	Color filters loaded	Slide shifter	Corrected image	Corrected image	High	Pass
15	To turn the camera on		System is attached to a camera	Load filters	Camera functions properly	Camera functions properly	high	Pass

Table 2. Integration Testing.

Test ID	Test Case	Test Case description	Expected result
1	Check the functionality of deuteranopia button	Click the deuteranopia button on the UI	On clicking, will load deuteranopia filters
2	Check the functionality of protanopia button	Click the protanopia button on the UI	On clicking, will load protanopia filters
3	Check the functionality of tritanopia button	Click the tritanopia button on the UI	On clicking, will load tritanopia filters
4	Check the functionality of Normal button in Deuteranopia scene	Click the normal button on the UI	On clicking, will load main menu
5	Check the functionality of Normal button in Tritanopia scene	Click the normal button on the UI	On clicking, will load main menu
6	Check the functionality of Normal button in Protanopia scene	Click the normal button on the UI	On clicking, will load main menu
7	Check the functionality of the camera	Run the app	Camera should work properly
8	Check the slider in Deuteranopia scene	Slide to change hue	Hue shifts
9	Check the slider in Protanopia scene	Slide to change hue	Hue shifts
10	Check the slider in Tritanopia scene	Slide to change hue	Hue shifts

Results and Discussion

A. Preliminary test

The Ishihara plates were shown to both sets of test subjects. The numbers inside the plates (7, 10, 4, 2, 8 and 45) are not visible to any of the four red-green color-blind subjects whereas all four normal vision subjects could correctly identify all six values [5]. The designing and testing of various filters and image enhancement techniques were used on the images. As a simple approach, the images were processed to remove the green and blue components leaving only the red component. This can be done shifting the hue.

B. Test results post execution of the application

Once the application was running on the device, the Ishihara plates were shown again to the test subjects. The normal vision subjects could identify the numbers on the plates [6]. Using the app, the color-blind subjects were also able to identify the numbers on the Ishihara plates and hence cleared the test.

Conclusion

Color vision deficiency is a very common disease in today's world, yet there are no specific corrective measures for the same. This paper suggests one method to correct color-blindness. By shifting the hue of a real-time image, the colors can be inspected, and perceived by a color-blind individual.

Acknowledgment

The author would like to thank Mrs. Renuka Devi for her guidance throughout the development of the application.

References

1. Mellilo P, Riccio D, Di Perna L, et al. Wearable Improved vision system for color vision deficiency correction. *IEEE Journal of Translational Engineering in Health and Medicine*. 2017; 5: 1-7. doi: 10.1109/JTEHM.2017.2679746
2. Badeche M, Benmohamed M. Real-time tracking for Augmented reality. *International Conference on Information & Communication Technologies*. 2006: 1773-1778. doi: 10.1109/ICTTA.2006.1684654
3. Agusanto K, Li L, Chuangui Z, Sing NW. Photorealistic rendering for augmented reality using environment illumination. *The Second IEEE and ACM International Symposium on Mixed and Augmented Reality*. 2003: 208-216. doi: 10.1109/ISMAR.2003.1240704
4. Poret S, Dony RD, Gregori S. Image processing for color blindness correction. *IEEE Toronto International Conference Science and Technology for Humanity (TIC-STH)*. 2009: 539-544. doi: 10.1109/TIC-STH.2009.5444442
5. Halic FT, Kockara SS, Bayrak TC, Iqbal FK, Rowe FR. Two-way Semi-Automatic Registration in Augmented Reality System. *IEEE International Conference on Systems, Man and Cybernetics*. 2007: 3390-3395. doi: 10.1109/ICSMC.2007.4414000
6. Bostanci E, Clark AF, Kanwal N. Vision-based User Tracking for Outdoor Augmented Reality. *IEEE Symposium on Computers and Communications (ISCC)*. 2012: 000566-000568. doi: 10.1109/ISCC.2012.6249356