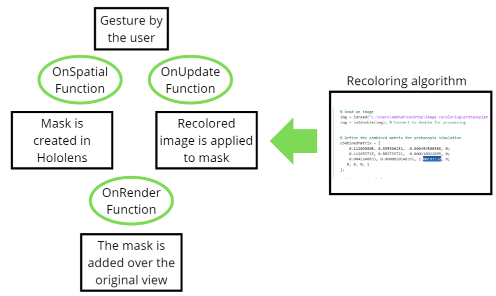
The implementation of the real-time correction of colors for a colorblind (daltonian) person using HoloLens(1st gen.)

Bauyrzhan Askarov, Rakhat Ospanov

***Abstract*—With the rapid advancement of digital technologies, colorblind individuals encounter increasing challenges in everyday visual tasks. While various image processing techniques exist to enhance color perception, their real-time integration into wearable augmented reality (AR) systems remains underexplored. This research focuses on incorporating existing color correction algorithms into an AR headset, specifically the Microsoft HoloLens platform, to demonstrate its potential as an assistive tool for colorblind users. This study will integrate the LMS daltonization algorithm into the first-generation HoloLens using Microsoft's library to provide a practical AR-based solution. Our implementation establishes a testing framework for future researchers to validate and refine color correction methods utilizing AR HoloLens for clinical trials.**



***Index Terms*—Image recolorization, Augmented reality, Color vision deficiency, Finger interaction.**

I. Introduction

Colorblindness, also referred to as color vision deficiency (CVD), is a widespread visual impairment that affects daily tasks that require accurate color perception. Approximately one in twelve men and one in two hundred women experience some form of CVD, characterized by an inability to perceive and differentiate colors.

This impairment arises from partial or complete dysfunction of the retina's cone cells at the back of the eye, which respond to long (L), medium (M), or short (S) wavelengths of light that humans can perceive. Depending on the extent and type of cone cell dysfunction, color vision deficiency can be broadly classified into three main types: monochromacy, dichromacy, and trichromacy. Monochromacy, the most severe form, results in complete color blindness, with individuals perceiving their environment solely in shades of gray. Dichromacy occurs when one of the three cone types is entirely nonfunctional, causing substantial color perception issues. In contrast, anomalous trichromacy involves partial functionality, resulting in impaired yet not entirely absent color discrimination. Among these types, red-green deficiencies linked to abnormalities in L or M cones are most common, whereas blue-yellow deficiencies, associated with S cones, are relatively rare (see Fig. 1 for a visual overview of these subtypes). Given their prevalence and the significant difficulties they pose in everyday activities, this research specifically targets dichromacy and anomalous trichromacy, providing a focused basis for algorithm integration in AR with real-time implementation.

Recent advancements in augmented reality (AR) technologies have provided promising opportunities to support individuals with color vision deficiency (CVD) through real-time assistive solutions. Existing AR-based systems, such as the commercially available Chroma platform, dynamically analyze users' surroundings to enhance color perception interactively. Additionally, research prototypes have developed specialized glasses targeting improved color differentiation; however, few studies have integrated established color correction algorithms directly into widely available AR headsets. Addressing this gap, our research specifically utilizes the first-generation Microsoft Hololens due to its robust real-time computing capabilities and flexibility in handling color correction algorithms. In this study, we incorporate the LMS daltonization algorithm into the Hololens environment to modify RGB and LMS values in real-time, aiming to improve color discrimination specifically for users experiencing dichromacy or anomalous trichromacy [1]. To systematically validate the effectiveness of our implementation, the widely recognized Ishihara color test plates will serve as an initial benchmark. The framework developed through this research provides a foundation for iterative experimentation, allowing future studies to evaluate and enhance AR-based color correction methodologies further.

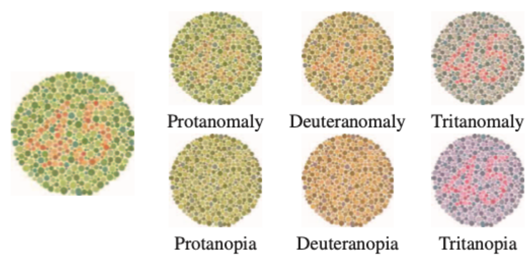


Fig. 1: Types of color blindness.

II. Related Works

Several studies have explored image-processing techniques aimed at assisting individuals with color vision deficiencies

Year	Authors	Work Description	Key Results
2006	Iaccarino et al.	Brightness adjustment	Improved color distinction
2008	Okhubo and Kobayashi	AI recoloring	Converted red tones to blue (and similarly other colors) tailored for different types of colorblindness.
2009	Huang et al.	Hue modification in HSV color space	Enhanced recoloring by considering local image characteristics rather than just brightness adjustments.
2010	Huang et al.	Probability distribution methods for refining recoloring	Further refined hue-based recoloring for improved accuracy.
2015	Kuhn et al.	Hue-based modifications preserving original colors	Improved visual fidelity while assisting colorblind viewers.
2019	Tanuwidjaja et al.	AR application ("Chroma") utilizing recoloring and segmentation	Significant improvement in user color differentiation test scores.
2020	Manaf et al.	AR glasses integrating ML and image processing	Interactive recoloring of specific colors upon user request.
2020	Sahih et al.	Comparative analysis of different AR approaches for colorblind assistance	Evaluated multiple AR-based methods for effectiveness.
2020	Li et al.	Adversarial networks for color correction and inversion	Introduced deep learning methods that correct colors and invert them back, maintaining consistency.

TABLE I: A table summarizing the work descriptions and results.

(CVD). This section discusses key research efforts chronologically (2006–2020), highlighting their contributions and relevance to our work.

In one of the earliest studies, Iaccarino et al. proposed an image-processing algorithm that adjusts brightness to enhance color perception for colorblind users [1]. Their experiments showed that boosting overall brightness could help individuals with CVD distinguish colors more effectively.

Following this initial breakthrough, researchers began implementing more sophisticated techniques. For example, Ohkubo and Kobayashi developed algorithms to recolor traffic lights for specific types of colorblindness using an early form of artificial intelligence [2]. In cases where a user could not perceive red, the system would transform red tones into blue, and similarly adapt other colors for various deficiencies.

Building on these foundations, Huang et al. introduced a novel approach that modified the hue component of the HSV color space based on local image characteristics instead of simply increasing brightness [3]. Their subsequent work integrated probability distribution methods to further refine this recoloring process [4]. Kuhn et al. similarly focused on hue-based modifications but emphasized preserving the original colors as much as possible, thus improving overall visual fidelity [5].

As algorithms became more accurate, industrial applications flourished. Tanuwidjaja et al. introduced an AR application called Chroma, which incorporated various image recoloring and segmentation methods to offer real-time assistance for CVD users [6]. Their user study revealed significant improvements in color differentiation test scores among participants who used Chroma.

Other industry-oriented efforts integrated machine learning into color correction. Manaf et al. combined ML and image-processing algorithms to develop AR glasses capable of altering specific colors upon the user's request [7]. This interactive design empowered users to selectively modify colors in their visual field.

More recently, Salih et al. compared different AR approaches for colorblind assistance [8], and Li et al. discussed adversarial network methods that invert the corrected colorblind image back to the original color space while preserving consistency [9]. Li's work not only summarized previous advancements but also introduced a new dimension of deep learning for color correction, demonstrating the ongoing innovation in this field.

In light of these developments, our research seeks to both build upon and extend existing techniques. We plan to experiment with advanced recolorization methods inspired by M. Riberio and A.J. Gomes [10]. Finally, leveraging insights from Anderson et al. regarding utilization of HoloLens for driving condition awareness [11], we aim to implement our color correction solution on Microsoft HoloLens 1.0. According to these findings, the HoloLens platform supports a robust ecosystem—incorporating Visual Studio, C#, and Unity3D—that will facilitate our system's development and real-time testing. Our initial experiments will be conducted using MATLAB before transitioning to a fully integrated AR environment.

III. Methodology

A. Hololens Setup

This research focuses on the integration of real-time image recolorization methods for color vision deficiency (CVD) using Microsoft HoloLens 1.0. To address the needs of colorblind users, there should be some configuration procedure of Hololens setup. Specifically, two primary methods exist for deploying applications onto the HoloLens device: a direct USB connection or wireless deployment via Wi-Fi. To simplify the implementation and testing processes, Wi-Fi deployment was utilized. The procedure involves pairing the development PC and the HoloLens by entering the headset's IP address, enabling efficient data transfer and application debugging. This was achieved by using Visual Studio 2017 as the preferred IDE because newer versions occasionally present connection stability issues with HoloLens 1.0.

To streamline direct image-processing tasks on the HoloLens 1.0 without dependency on third-party platforms such as Unity, the Microsoft-provided HoloLensforCV library was utilized. The primary objective of this library is to facilitate the use of the HoloLens Research Mode by providing direct access to various device sensors and camera frames.

Despite these benefits, the HoloLensforCV library imposes certain limitations. Specifically, it restricts direct modifications to core system functions, allowing developers only to extend or script additional functionalities rather than altering or removing existing ones. Therefore, although it provides crucial sensor access and function triggers, customization is constrained to external scripting.

For clarity, below is a concise overview of key functions within the HoloLensforCV library, central to this research:

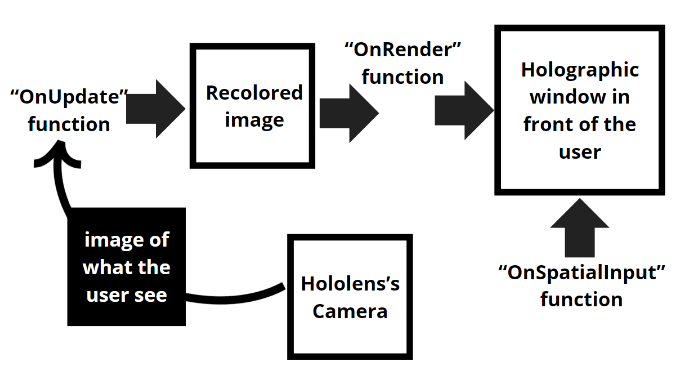


Fig. 2: HololensForCV scheme

(a) OnSpatialInput (App::main): Triggered by a recognized gesture ("two-finger tap"), this function provides spatial interaction parameters, including the user's head position, gaze direction, and device orientation. It initiates essential events, such as the creation of interactive application windows.

(b) OnUpdate (App::main): Executed at fixed, user-defined intervals, this function provides continuous access to camera frames captured by the HoloLens. Critical tasks such as image processing, computer vision algorithms, and color-correction calculations are implemented here.

(c) OnRender (App::main): Responsible for rendering updated images onto the HoloLens display. Frames processed and altered by the OnUpdate function are visually presented to the user through this rendering process.

Through utilizing the HoloLensforCV library and leveraging HoloLens Research Mode, the proposed project methodology successfully achieves real-time color correction tailored to users with color vision deficiencies. Figure 3 provides a visual summary of the implemented script pipeline.project pipeline.

According to this figure, user interaction begins when the OnSpatialInput function creates an initial window within the user's AR view. Next, the OnUpdate function processes real-time camera frames, applying the LMS daltonization algorithm to recolor the user's environment. Finally, the recolored images are presented in the AR window by the OnRender function, enabling enhanced color differentiation for colorblind users.

B. Algorithms of color recolorization

Our image processing algorithms were mainly based on the following work [12]. We used the LMS algorithm based on this paper since it is suited to all types of dichromacy (Protanopia, Deuteranopia, and Tritanopia) colorblindness and is the most convenient to use with Hololens glasses.

It should be noted that the LMS algorithm is based on the use of LMS color space, where each letter defines the specific range of waves a human eye can perceive: L-long, M-medium, and S-short. The transformation of RGB color space to LMS color space is processed through matrix multiplication.

Throughout the algorithm, the pipeline can be divided into two sections: colorblindness simulation and colorblindness correction. In the first part, the simulation of the specific colorblindness type is processed through matrix algorithms. Consequently, these simulated images are then used to correct the colors of the image for the Colorblind user.

Description of the LMS Daltonization algorithm [1iteimage13

Input: RGB input image

Output: RGB color-corrected image

Step 1. The first step is to convert all RGB pixels of the image to LMS color space. It is achieved by using the matrix(1) provided below:

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 17.8824 & 43.5161 & 4.11935 \\ 3.45565 & 27.1554 & 3.86714 \\ 0.0299566 & 0.184309 & 1.46709 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{1}$$

Step 2. The next step is to apply specific matrices for each type of Dichromacy colorblindness: Protanopia(2), Deuteranopia(3), and Tritanopia(4). These matrices will transform the LMS images into the spectrum of certain diseases:

$$\begin{bmatrix} L_P \\ M_P \\ S_P \end{bmatrix} = \begin{bmatrix} 0 & 2.02344 & -2.52581 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \tag{2}$$

$$\begin{bmatrix} L_D \\ M_D \\ S_D \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0.49421 & 0 & 1.24827 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \tag{3}$$

$$\begin{bmatrix} L_T \\ M_T \\ S_T \end{bmatrix} = \begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ -0.395913 & 0.801109 & 0.0 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \tag{4}$$

Step 3. After acquiring images for each type of disease in LMS color space, we return the images back to RGB space using the matrix (5):

$$\begin{bmatrix} R_i \\ G_i \\ B_i \end{bmatrix} = \begin{bmatrix} 0.0809444479 & -0.130504409 & 0.116721066 \\ 0.113614708 & -0.0102485335 & 0.0540193266 \\ -0.000365296938 & -0.00412161469 & 0.693511405 \end{bmatrix} \begin{bmatrix} L_i \\ M_i \\ S_i \end{bmatrix} \tag{5}$$

Step 4. At that point, we have 2 images: the first is the original image, and the second is a simulated image for each type of color deficiency. Now we have to subtract each RGB pixel of the acquired image from the original to find the difference using (6),(7),(8). **Note:** The simulation images are important, and, even if they are not displayed, they should be calculated for future steps:

$$D_{R(i)} = R - R_i \tag{6}$$

$$D_{G(i)} = G - G_i \tag{7}$$

$$D_{B(i)} = B - B_i \tag{8}$$

Step 5. After all matrix multiplications and subtraction till step 4, some small errors might occur. Therefore, to be able to see the new images we need to shift colors to visible spectrum. It is completed via multiplying by error matrices for Protanopia(9), Deuteranopia(10), and Tritanopia(11):

$$\begin{bmatrix} R_{map(P)} \\ G_{map(P)} \\ B_{map(P)} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0.7 & 1 & 0 \\ 0.7 & 0 & 1 \end{bmatrix} \begin{bmatrix} D_{R(P)} \\ D_{R(P)} \\ D_{R(P)} \end{bmatrix} \tag{9}$$

$$\begin{bmatrix} R_{map(D)} \\ G_{map(D)} \\ B_{map(D)} \end{bmatrix} = \begin{bmatrix} 1 & 0.7 & 0 \\ 0 & 0 & 0 \\ 0 & 0.7 & 1 \end{bmatrix} \begin{bmatrix} D_{R(D)} \\ D_{R(D)} \\ D_{R(D)} \end{bmatrix} \tag{10}$$

$$\begin{bmatrix} R_{map(T)} \\ G_{map(T)} \\ B_{map(T)} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0.7 \\ 0 & 1 & 0.7 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} D_{R(T)} \\ D_{R(T)} \\ D_{R(T)} \end{bmatrix} \tag{11}$$

Step 6. In step 6, to find the final simulated image as seen by people of one of the aforementioned diseases, we have to add each RGB channel of shifted colors to the RGB channels of the original image via equation (12), (13), (14):

$$R_{F(i)} = R + R_{map(i)} \tag{12}$$

$$G_{F(i)} = G + G_{map(i)} \tag{13}$$

$$B_{F(i)} = B + B_{map(i)} \tag{14}$$

In the following Figure 3, the simulation and corrected images using the aforementioned LMS Daltonization algorithm can be seen.

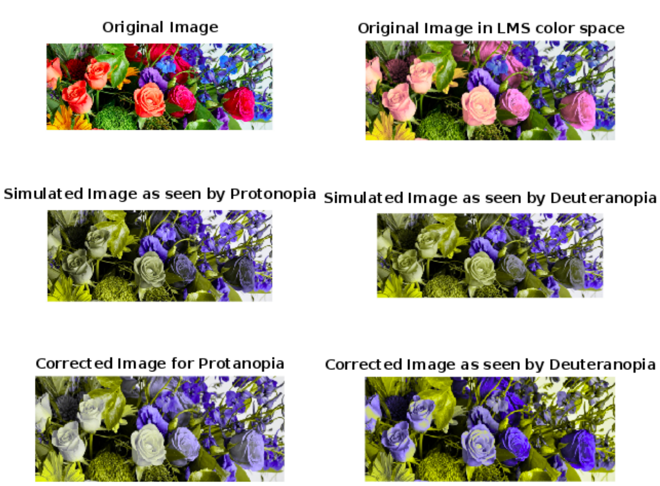


Fig. 3: Simulation and corrected images for Protanopia and Deuteranopia types of colorblindness in Matlab.

C. Additional Considerations

Some notions regarding the algorithm implementation in Hololens devices must be included. First, the colorspace used in this device is BGRA (not a common RGB configuration), where B is the blue channel, G is the green channel, R is the red channel, and A is the alpha channel.

Therefore, in addition to the described algorithm above, we should add the colorspace transformation from BGRA to RGB in the beginning and RGB to BGRA at the end.

Furthermore, as the spatial data is only accessed through the ”OnSpatialInput” function, we cannot process the data obtained every second. Therefore, the position of the frames should be constant all the time, until a user does not use ”Gesture.”

IV. Experiments and Results

It should be noted that the two primary goals of our experiment are to represent the color vision of color-deficient patients using the Augmented Reality Hololens 1.0 device and to develop a color correction tool that will allow colorblind people to better distinguish their ”difficult” colors.

As a result, our team conducted two types of experiments as part of our project, the first of which was Hololens Recoloring Simulation, which was based on the research of generating the view of people with some degree of color deficiency for normal people using Augmented Reality and the LMS recoloring algorithm. The second part of our experiments focused on color correction for people who are colorblind.

Furthermore, in the realities of our experiments, we considered three types of participants (colorblind people), with each having a high degree of red-green color deficiency. Thus, our experiments were based on protonopia simulation and correction.

A. Hololens Recoloring Simulation of Protanopia

The methodology section of our experiment details the LMS recoloring algorithm, which aims to replicate the visual perception of an individual with colorblindness. It involves the manipulation of the original image's matrix, which is typical for normal vision. In this matrix, each pixel comprises varying degrees of the primary colors red, green, and blue. To achieve the recoloring effect, the original matrix must be multiplied by three distinct matrices. The first of these matrices is responsible for converting RGB values to LMS values. The second matrix varies based on the type of colorblindness: protanopia, deuteranopia, or tritanopia. The final matrix converts the LMS values back to RGB values. Therefore, it is clear that for the representation of the protanopia view, we are required to multiply the original image from the Hololens camera by these three matrices. However, for the better computational power economy of the Hololens first-generation device, we decided to reduce the number of matrix multiplications from three to one by preliminary multiplying three constant matrices.

As a result, our protanopia simulation algorithm can be described in only one equation:

SimulatedProtanopia =

$$\begin{bmatrix} 0.112090805 & 0.885306131 & -0.000454506368 \\ 0.112653715 & 0.889739721 & -0.000110022065 \\ -0.0045248835 & 0.0000820148395 & 1.00191532 \end{bmatrix} \tag{15}$$

$*$ OriginalRGBMatrix

Therefore, the overall implementation pipeline of the Hololens Protanopia Simulation is described in Fig. 4.

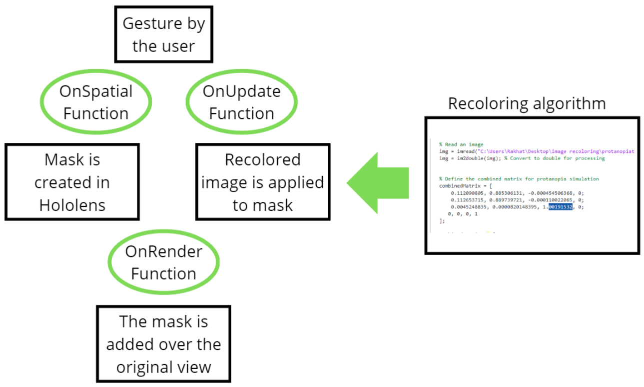


Fig. 4: Hololens Protanopia Simulation pipeline

According to this Protanopia Simulation implementation, we obtained the following results:
Figure 5 presents the rgb gradients seen using Hololens first generation and our protanopia simulation algorithm.

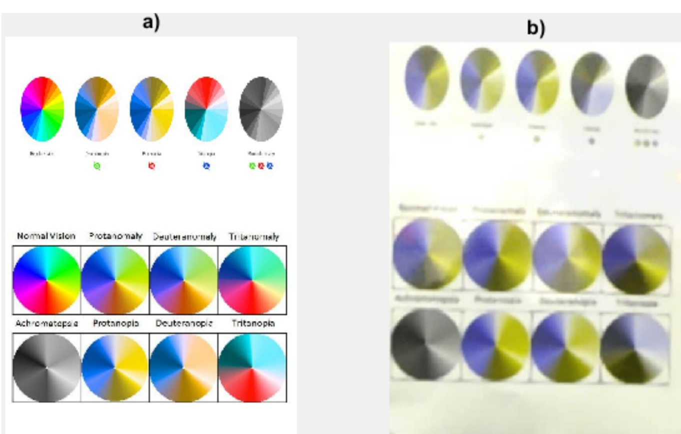


Fig. 5: Simulation of Protanopia from Hololens camera. a) Original Photo, b) The image seen in Hololens

Therefore, it is clear that our Hololens Protanopia Simulation solution is able to transform the view of a normal user regarding the Protanopia view. As is seen in Fig.5 all the red and green colors became much less distinguishable.
Nevertheless, the question ”To what extent are the results accurate?” still remains.

B. Hololens Protanopia Correction

In the second part of our experiments, we evaluated the accuracy of our Hololens Protanopia Correction Solution.

To find the answer, our team first conducted a ”small” experiment, where we first found some objects that were mostly red and green colors (Fig. 6 a). Next, we simulated the view of a protanopia person through the Hololens implementation and verified that this kind of colorblind person is not able to distinguish two objects with red and green colors. For our case, a red cap and a sweater were chosen (Fig. 6 b). By implementing the simulation of protanopia disease with the image correction, we found that a protanopia person now could distinguish these two specific items from one another. Now, the cap has become blue, while the sweater remains green Fig. 6 c).

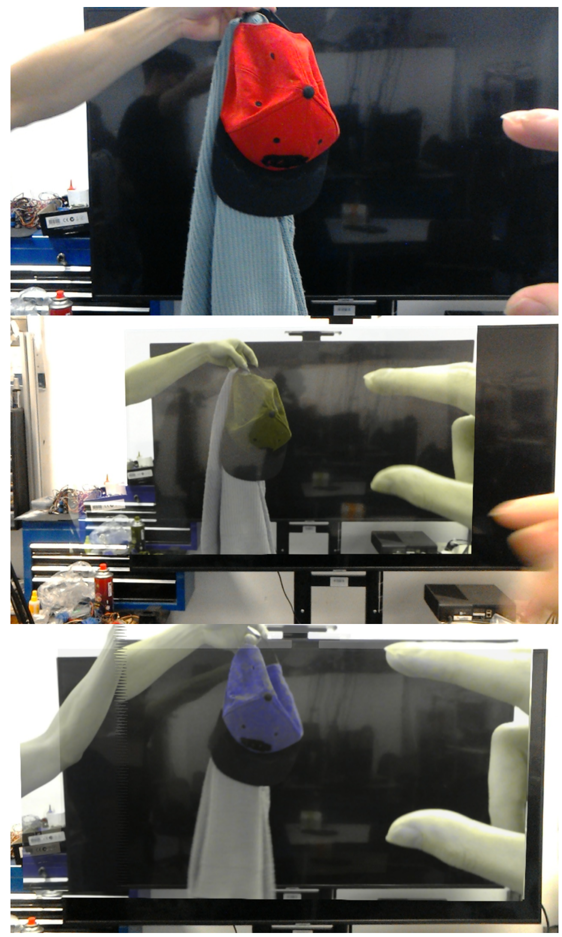


Fig. 6: Demo implementation of the LMS-based recoloring algorithm for Protanopia. a) Original image, b) Protanopia Simulation of the original image, c) Corrected image in Protanopia simulation

For the result, it was clear that our correction solution through Hololens recoloring worked fine, as the red color became blue, which is distinguishable for protanopia colorblind people.

The demonstration of our color correction Hololens solution from their view perspective can be seen in Fig.7, where b) demonstrates their representation of the Ishihara test, and c) represents their view with the use of our correction. It is noticable that the red color of the number became blue, while the red-black colors of the background became black-gray.

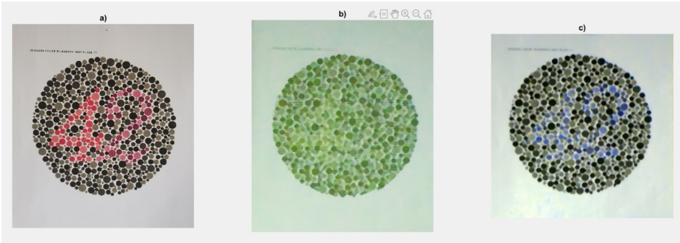


Fig. 7: Hololens color Correction of Ishihara test. From top to bottom: a) Original image, b) Protanopia Simulation of the original image, c) Corrected image in Protanopia simulation

V. Discussion and Conclusion

Through the use of HololensForCV and Hololens' research mode presented by Microsoft Team, we could implement different image recoloring methods to simulate the view of a colorblind person and create a correction tool for these color-deficient patients.

Through several experiments, we obtained a quantitative proof of our approaches, where we observed that the results on colorblind simulation and correction parts are very high.

1) Image recolorization: Regarding the image processing part, it should be stated that in the scope of this research, we investigated the LMS recolorization method, which is widely popular among different sources.

As this method directly works with the representation of colors in terms of humans' eyes, it could provide us with good results and accurate representations.

2) Hololens implementation: This part required the majority of our time, as different approaches have been considered. Nevertheless, eventually, it can be stated that we succeeded in the implementation.

Through the use of the HololensForCV library, we could hardcode our implementation directly to the Hololens device without the use of any third-party applications.

Regarding the restrictions of this approach, we could not obtain a real-time permanent mask, as even if it translates the real-time data, the position of the mask is restricted to 3D coordinates (2 meters in front of the user after the "Gesture" was recognized) and is not updated as the user moves throughout the 3D coordinate system. Therefore, currently, to have the desired effect, the user should be stable in regards to the real world space system.

A. Future work

As the experiment has mostly succeeded, there is still some work to be done.

1) Hololens implementation: In terms of Hololens implementation, it is required to create a stable mask not in terms of 3D Real World coordinates, as it is currently stated, but in terms of the user monitor. So once the mask is created in front of the user, the position of the mask (frames) should be updated based on the updates in the position of the user. Thus, the mask should be always in front of the user.

2) Image processing: Also, to broaden the number of methods that can be utilized by AR glasses, we intend to further work on our skillet-based recoloring algorithm for dichromats described in the work of Ribeiro M. G. and Gomes A. J. [13]. Furthermore, as the rest of the work is approximately done, our team can direct the majority of the power for the research of some other recoloring algorithms. Therefore, we also consider the various increases in the number of analyzed methods through AR glasses.

References

[1] G. Iaccarino, D. Malandrino, M. Percio, and V. Scarano, "Efficient edge-services for colorblind users," 05 2006, pp. 919–920.

[2] T. Ohkubo and K. Kobayashi, "A color compensation vision system for color-blind people," in *2008 SICE Annual Conference*, 2008, pp. 1286–1289.

[3] J.-B. Huang, S.-Y. Wu, and C.-S. Chen, "Enhancing Color Representation for the Color Vision Impaired," in *Workshop on Computer Vision Applications for the Visually Impaired*. Marseille, France: James Coughlan and Roberto Manduchi, Oct. 2008. [Online]. Available: https://inria.hal.science/inria-00321936

[4] J.-B. Huang, C.-S. Chen, T.-C. Jen, and S.-J. Wang, "Image recolorization for the colorblind," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009, pp. 1161–1164.

[5] G. R. Kuhn, M. M. Oliveira, and L. A. F. Fernandes, "An efficient naturalness-preserving image-recoloring method for dichromats," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1747–1754, 2008.

[6] E. Tanuwidjaja, D. Huynh, K. Koa, C. Nguyen, C. Shao, P. Torbett, C. Emmenegger, and N. Weibel, "Chroma: A Wearable Augmented-reality Solution for Color Blindness," in *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, ser. UbiComp '14. New York, NY, USA: ACM, 2014, pp. 799–810. [Online]. Available: http://doi.acm.org/10.1145/2632048.2632091

[7] A. S. Manaf and R. F. Sari, "Color recognition system with augmented reality concept and finger interaction: Case study for color blind aid system," in *2011 Ninth International Conference on ICT and Knowledge Engineering*, 2012, pp. 118–123.

[8] A. E. Salih, M. Elsherif, M. Ali, N. Vahdati, A. K. Yetisen, and H. Butt, "Ophthalmic wearable devices for color blindness management," *Advanced Materials Technologies*, vol. 5, no. 8, p. 1901134, 2020. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/admt.201901134

[9] J. Li, X. Feng, and H. Fan, "Saliency-based image correction for colorblind patients," *Computational Visual Media*, vol. 6, pp. 169–189, 06 2020.

[10] M. Ribeiro and A. J. P. Gomes, "Recoloring algorithms for colorblind people: A survey," *ACM Comput. Surv.*, vol. 52, no. 4, aug 2019. [Online]. Available: https://doi.org/10.1145/3329118

[11] R. Anderson, J. Toledo, and H. ElAarag, "Feasibility study on the utilization of microsoft hololens to increase driving conditions awareness," in *2019 SoutheastCon*, 2019, pp. 1–8.

[12] L. A. Elrefaei, "Smartphone based image color correction for color blindness," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 12, no. 3, p. pp. 104–119, Jul. 2018. [Online]. Available: https://online-journals.org/index.php/i-jim/article/view/8160

[13] M. G. Ribeiro and A. J. P. Gomes, "A skillet-based recoloring algorithm for dichromats," in *2013 IEEE 15th International Conference on e-Health Networking, Applications and Services (Healthcom 2013)*, 2013, pp. 702–706.