Identify Problematic Regions in Field of View for Aiding People with Vision Impairments

—Comp562 Project Report—

Yiyin Gu

May 13, 2021

1 Introduction

Some visual impairments, such as age-related macular degeneration (AMD), have a sustained impact on visual function and can lead to central vision loss.[1] This can cause potential dangerous situations on the road, for example, if they cannot see the car or road signs in the area that was influenced by AMD. This project trained a neural network to help identify the problematic regions so that it can be combined it with Augmented Reality Head-Mounted Display to dynamically generate hints for patients in an outdoor environment.

2 Visual Impairment Simulations

For data generation, we should first have a working model of wet AMD visual impairment simulations. XREye is a system that run visual impairment simulations run in VR or video–see-through AR and can also be experienced when viewing 360° images, using an HTC Vive Pro HWD with a Pupil Labs 200Hz binocular eye tracker add-on.[1] The system can also generate from 2d images, which is what we used to train the network.

We download our initial street view dataset called Cityscapes Dataset from its website.[2] Then we generated the wet AMD influenced field of view with XREye for the 438 street view images that were taken in the cities Aachen, Weimar and Zurich. Please see Figure 1 for an example.

(a) Street View From Aachen

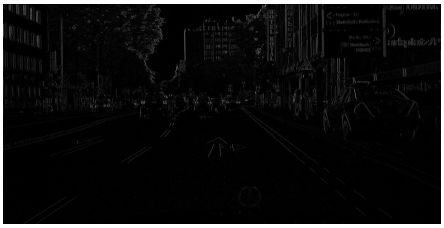
(b) Simulated View of an AMD patient

Figure 1: A Comparison of the original image and the simulated AMD version

3 Quantification of Influenced Degree

Although we got the simulated images in the previous section, we can only subjectively decide whether a part of the image is influenced by AMD or not. This section is dedicated to find an appropriate way to calculate the influenced degree pixel-wise for the target images.

We represent each pixel with an energy value. The algorithm used is called forward energy. Briefly stated, the calculation process is adding the $L1$-norm of the gradient of the pixel value and the energy change of the newly adjacent pixels when the target pixel is dismissed. More details could be found in the fifth section in the paper cited.[3] Figure 2 shows the visualization of the energy of the images, the brighter the color, the higher the energy. Just for the purpose of demonstration, the highest energy in each image is correlated with a color of pure white, so the comparison between image with the color is invalid.



(a) Energy Visualization of Original Image

(b) Energy Visualization of Simulated Image

Figure 2: Visualizations of the energy of original and simulated images

There are two rules that we adapted to decide whether a pixel is significantly influenced. First, the pixel must be influenced, which means that the energy calculated in the original image and in the simulated image for the same pixel must have an above threshold (THRESHOLD_DIFFERENCE_RATIO) difference ratio. Second, the pixel must be itself important, or it would be meaningless to address it. Thus the original energy must be above another threshold (THRESHOLD_EDGE). Check every pixel if it satisfies the conditions above, and polarize the image with white as significantly influenced and black as uninfluenced. Figure 3 shows the significantly influenced pixels in white of the sample image, which is the same image featured in Figure 1 and Figure 2. The constants used are $THRESHOLD_DIFFERENCE_RATIO = 0.8$ and $THRESHOLD_EDGE = 20$.

$$\begin{cases} \frac{|E_{original} - E_{simulated}|}{E_{original}} >= THRESHOLD_DIFFERENCE_RATIO \\ E_{original} >= THRESHOLD_EDGE \end{cases}$$

Figure 3: Visualization of Significantly Influenced Pixels of the Sample Image

The next step is to turn significantly influenced pixels into areas. By splitting the original image into square grids with side length $GRID_SIZE$, and polarize the grid if there is more than $THRESHOLD_IN_GRID_INFLUENCED$ pixels in the grid that are significantly influenced. Figure 4 shows the significantly influenced areas in white of the sample image. The constants used are $GRID_SIZE = 64$ and $THRESHOLD_IN_GRID_INFLUENCED = 20$. The original dimension of the image is 2048×1024.

Figure 4: Visualization of Significantly Influenced Areas of the Sample Image

Generating Images for the visualization of significantly influenced areas for all 438 street view images to create a dataset for later training.

4 Image-to-Image Translation with Conditional GANs

We used the Pytorch implementation of our method for high-resolution photo-realistic image-to-image translation by Nvidia [4] for the training process. Because we are not generating images from a label image, we used the "–label_nc 0" and "–no_instance" option, so the network will try to use the input of the original street view and turn it into a visualization of significantly influenced areas. We resized the original images from 2048×1024 to 1024×512 before the training to save on computing power. 400 pairs of the images are used for training and the 38 pairs are left for testing. It was trained for 200 epoches. Figure 5 is an example of an image from the test set, on the left is the original image, the middle is the images generated by neural network, and the right is the image we generated from our algorithm stated in the previous sections.

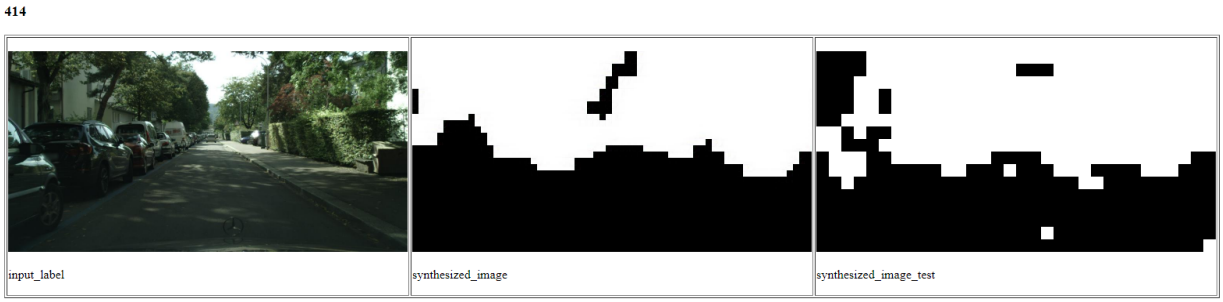


Figure 5: Examples of Training Results

5 Conclusions and Future Works

The result of the neural network are satisfying, being generally consistent with the algorithms. It even has its advantages because it is not restricted to a all-or-none grid.

With the result of the neural network, we can export the trained network into ONNX so it can be imported into Unity and other game engines which can be connected to different head-mounted displays. In addition, researchers can held users study to adjust the thresholds mentioned to get a more practical result.

6 Acknowledgments

The authors wish to thank Katharina Kroesl for providing the original street view images and Simulated View using XReye and Praneeth Kumar Chakravarthula for the oppotunity to participant in this study. This work is part of a bigger project in collaboration with Kroesl and Chakravarthula.

References

[1] Katharina Krösl, Carmine Elvezio, Matthias Hürbe, Sonja Karst, Steven Feiner, and Michael Wimmer. Xreye: Simulating visual impairments in eye-tracked xr. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pages 831–832. IEEE, 2020.

[2] Cityscapes dataset. https://www.cityscapes-dataset.com/. Accessed: 2021-05-13.

[3] Michael Rubinstein, Ariel Shamir, and Shai Avidan. Improved seam carving for video retargeting. *ACM Transactions on Graphics (SIGGRAPH)*, 27(3):1–9, 2008.

[4] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.