Non-invasive Eye Tracking using CNN

Wei-Hsiang, Shen
tommyrpg1010@gmail.com

Yo-da, Shih
york3816877@yahoo.com.tw

Yi-Cheng, Pan
pan010941@gmail.com

Abstract

Conventional eye tracking system consists of hardware devices that use near infra-red light to track the gaze of the user's eye. The system is widely used in gaming or research usage. However, the equipment is costly and hard to integrate into smartphones.

In this project, we proposed a non-invasive eye tracking system. No external light sources are used for tracking the eyes. A convolutional neural network are trained to determine the eyes' vision.

We evaluate the system by using L2 error distance between the predicted and the ground truth gaze location.

1. Introduction

Computer vision based eye tracking has become a dominant method in several domains including cognitive science, psychology, medical, diagnoses, market research, identity authentication and eye-based human-computer interaction.

Traditionally, eye tracking systems were implemented by feature-based and appearance-based method, of which feature-based method is the most popular method. Multiple cameras and external infra-red light would be used to detect the corneal-reflection signal and tracks the eye movement. Feature-based tracking method were used in systems like Tobii and SMI eye tracker. By using external infra-red lights, the eye movement can be tracked precisely since the pupil center and the glint can be easily extracted to calibrate the errors caused by head movements.

However, high cost and ill-health caused by lights are unavoidable. Moreover, infra-red light based systems are not reliable when used in outdoor conditions and smartphones. Therefore, we propose an eye tracking method which is low cost, non-invasive, high precision and easier operation.

We first detect the face and eyes of the user by a cascade object detector and KLT object tracker. Then, a convolutional neural network (CNN) waw trained to detect the eyes' gaze.

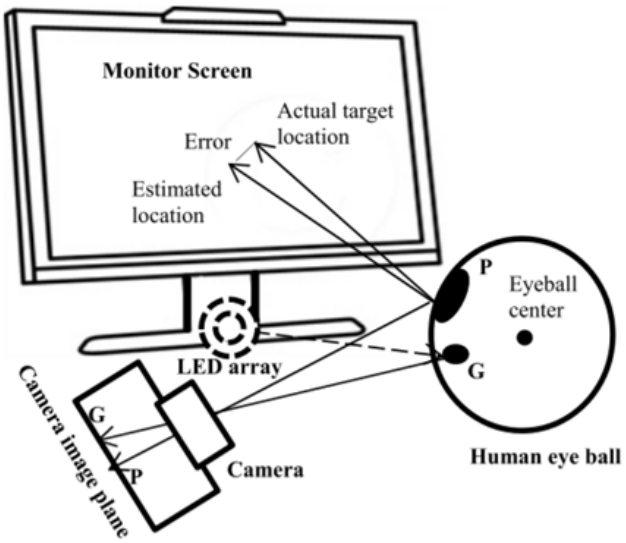

Figure 1. Pupil center corneal reflection

2.1. Previous Work

Previous method used in eye tracking is the well-known pupil center corneal reflection techniques (PCCR, fig. 1) . It uses an external near infra-red light to produce a glint on the eye cornea surface. The position of the glint is fixed and does not change with eyeball movement. Therefore, the gaze can be calculated by the relative position between the glint and pupil. System like the Tobii Eye Tracker has make successful products.[1] However, the cost of the external near infra-red light is high (around 200 USD at 2018).

Another attempt[2] uses visible light instead of near infra-red light, which is significantly cheaper. The method is comparable or even better than those using infra-red light and it runs real-time running in 25 frames/s. However, the performance is dependent on the ambient light condition, which may lower the performance of the system.

2.2. Our Work

We give up the PCCR detection method, and try different approach that does not require any external light source. Convolutional neural network (CNN) has shown great performance in image classification task. We train a CNN to

classify (predict) the gaze of one's eyes, as suggested in [3].

3.1. Training Dataset

The training dataset we used to train the CNN would be "GazeCapture"[3]. The dataset consists of 1,490,959 image frames from 1474 people. Comparing the dataset with existing datasets, "GazeCapture" is significant larger and contains more variation (different races, light sources, head posisions) which provide us to train the eye tracking CNN robustly.

3.2. System Design

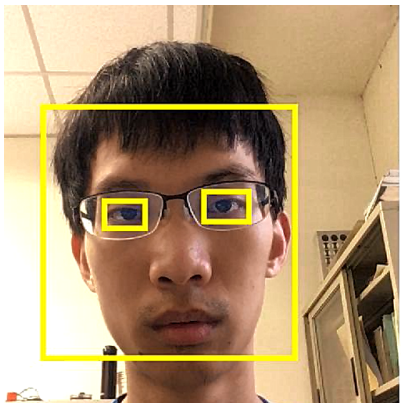

Figure 2. Face and eyes detection by VJ object detector

First, we found the face region and eye regions of the input frame using Viola-Jones object detector[4]. Once the face and eyes are detected, we use KLT object tracker to track them, so we do not need to perform detection on every frame and it enables the system to be real-time.

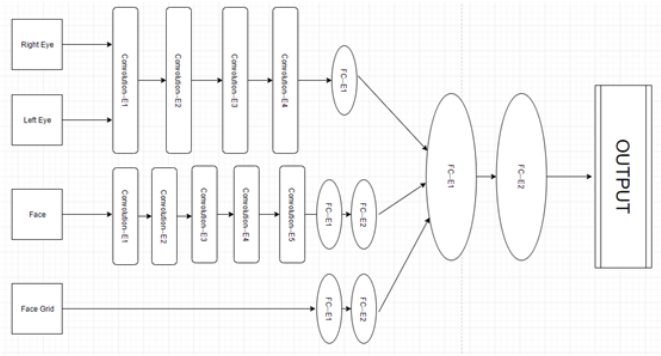

Figure 3. CNN Structure

The images of the right eye and left eye are fed into the convolutional layer of CNNs. We assume that the input right eyes and left eyes should have similar characteristic, so the convolutional layers share the same weighting (fig. 3).

We separate the input data to left eye, right eye, and the face image. Another input is the face grid, which is a binary mask used to indicate the location and size of head within the frame. With face grid, data gathered from different devices can be normalized and feed to the same CNN.

The output of the CNN is the x-y coordinate of the gaze. Since different data are obtained by different devices with different screens size. We normalize the output space to in the range of [0,1].

3.3. Training Details

Although the dataset consists of around 1.4M frames of images from 1474 people, we only choose 10% of the total frames also from 1474 people. Since each training epoch is running way too long if the entire dataset is used. Many frames are identical to each other, so we think that downsampling the frame counts would be acceptable.

The dataset is split into training, validation, and testing in the ratio of 80%, 10%, 10%.

3.4. Evaluation

The normalized output space is transformed back to absolute position according to the screen size. Then, the L2 distance between the ground truth position and predicted position is used as the evaluation metric.

The average L2 distance error on the testing set is 4.75cm, while in [3] they achieved average error of 1.04cm.

4. Visualization Result

We draw a red dot on the screen to represent the position that the person is looking at.

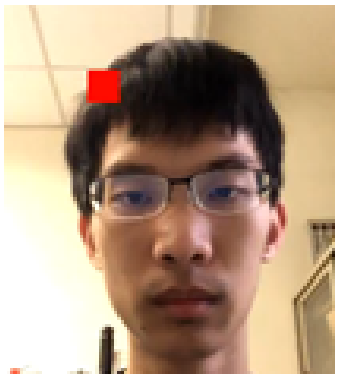

Figure 4. Visualization result

The result video demonstration and source code can be downloaded at the private repo in GitHub.

5. Conclusion and Discussion

In this project, we use train a CNN to predict the gaze location and the average L2 distance error on the screen in 4.75cm. By using CNN, we do not need external light source and is less sensitive to ambient light conditions.

When the user is looking at places outside the screen, CNN would predict wrong position since all the people in the dataset is gazing at positions in the screen. Therefore, we think that by appending the dataset with people looking outside the screen would solve the issue.

References

[1] Tobii Eye Tracking Product, https://www.tobii.com/group/about/this-is-eye-tracking/

[2] Sigut, J.; Sidha, SA. "Iris center corneal reflection method for gaze tracking using visible light". IEEE Transactions on Bio-medical Engineering. 58 (2): 411–9, February 2011.

[3] K. Krafka et al., "Eye tracking for everyone", Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 2176-2184, Jun. 2016.

[4] P. Viola, M. Jones, "Robust Real-time Object Detection", IJCV 2001.

[5] R. Konrad et al., "Near-Eye Display Gaze Tracking Via Convolutional Neural Networks", Computer Science Department of Stanford University.