Course Report for Human-Computer Interaction

VAN SONGHIENG
20CST
2020059118
songheingvan@gmail.com

ABSTRACT

Computer vision systems can identify and track objects, and they are becoming increasingly sophisticated. In the future, they may be able to interpret a person's gestures and facial expressions. This could enable a smartphone to be controlled with only the eyes. The technology would need to be paired with a screen that is transparent, so the user's eyes can be tracked. There are many potential benefits to using only one's eyes to control a smartphone. It would be quicker and more convenient than using one's hands. Additionally, it would be more immersive, as the user's hands would not be obscuring the screen. This could be especially beneficial for virtual reality applications. There are some challenges that need to be addressed before this technology can be implemented. The accuracy of eye-tracking systems needs to be improved. Additionally, screen resolution needs to be increased so that the user's eyes can focus on small objects on the screen. This technology is still in the early stages of development. However, it has the potential to revolutionize the way we interact with our smartphones.

Keywords

Gaze Gestures, Motion Gestures, Computer Vision, Human-Computer Interaction, Deep learning, Artificial Intelligence.

INTRODUCTION

As improved technology offers many advantages for human existence, it will also likely be the future of my original product, which I intend to expand into. For this paper, we will focus on enhanced technology. so that instead of what we have developed, we may grasp this new innovation better. Not least, this report will include detailed information regarding the project that was invented.

As established by the research, the product should have a sustainable competitive advantage. This is something that should be further looked into as technology improves and this product is enhanced. Additionally, as the research is further developed, this report will also be helpful in order to continuously monitoring and improving the product. In order to meet the research criteria, the product and all of its functions must be continuously assessed and updated with new information and findings. In order to demonstrate our findings, please refer to the product itself.

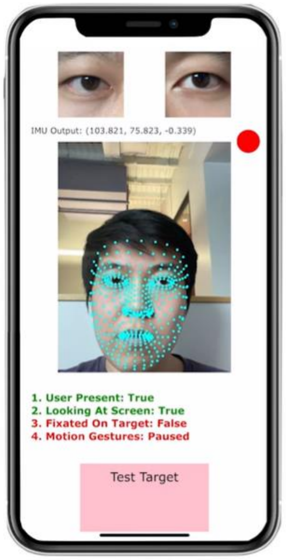
The product should be able to fulfill all of the necessary criteria with the help of technology. For example, our product should be able to have a method of detecting and responding to environmental changes. Additionally, the product should be able to automatically update itself to reflect the changes in the environment. The product should also be able to be monitored by a human, in order to properly assess the data that it is receiving.

SYSTEM and IMPLEMENTATION.

DETAIL PRODUCT FOR ORIGNAL PROGRAM

In the original project, determining whether or not there are any humans within the recorded camera's field of view is the first stage of the project. If so, the software will continue to execute and wait for user involvement. Otherwise, the software will be shut off automatically after a set amount of time if this is not done. The second stage will begin after the computer is able to recognize the user's pupil. At that point, it will be able to identify and analyze the coordinate that we are looking at.

To make the information clearer, the coordinate number will appear on the screen. If it cannot identify the user's pupils, it will display "none" in all other cases. The application may also recognize eye movements, such as blinking, looking up, down, left, right, or in the center, and show anything other than the default.



```
IMU Output: (103.821, 75.823, -0.339)
```

```
1. User Present: True
2. Looking At Screen: True
3. Fixated On Target: False
4. Motion Gestures: Paused
```

Fig. 1

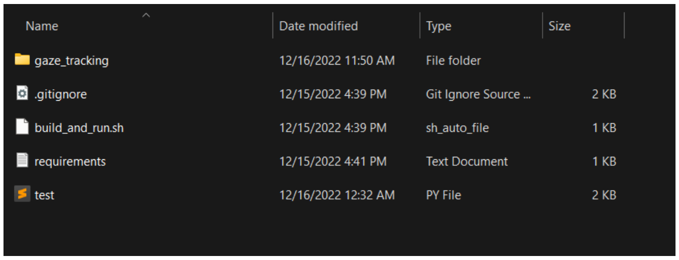
PLANNED TECHNOLOGY FOR ORIGNAL PROGRAM

For the original which I have created I used DeepLabCut and Gazenet. DeepLabCut is a deep convolutional network that combines pretrained ResNets with deconvolutional layers, two essential components from techniques for object identification and semantic segmentation. The network is made up of a ResNets variation whose weights were developed using the popular, extensive ImageNet object recognition benchmark, on which it performs exceptionally well. We demonstrate the versatility of this framework by tracking various body parts in multiple species across a broad collection of behaviors. The package is open source, fast, robust, and can be used to compute 3D pose estimates. GazeNet is an innovative framework for developing event detectors that do not require custom signal characteristics or signal thresholding. It uses an end-to-end deep learning technique to classify raw eye-tracking data into fixations, saccades, and post-saccadic oscillations. As a result, our strategy calls into question an established implicit assumption that hand-crafted features are required in the design of event detection algorithms.

CODE DESCRIPTION

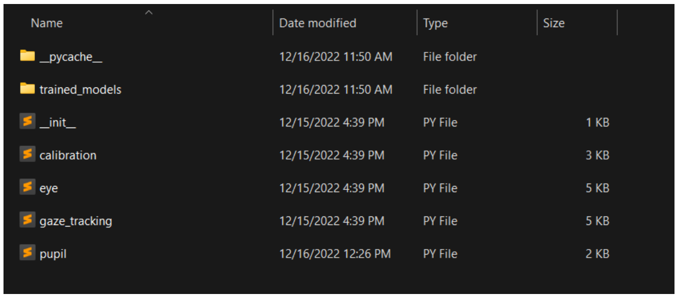
In this section, a detailed description of the main function will be provided to help better understand how this algorithm works in real-life applications.

Noted: We need to install according to the requirements, which are provided in the requirements.txt file, or we can run "pip install -r requirements.txt" in the command prompts in the window.



Name	Date modified	Type	Size
gaze_tracking	12/16/2022 11:50 AM	File folder	
.gitignore	12/15/2022 4:39 PM	Git Ignore Source ...	2 KB
build_and_run.sh	12/15/2022 4:39 PM	sh_auto_file	1 KB
requirements	12/15/2022 4:41 PM	Text Document	1 KB
test	12/16/2022 12:32 AM	PY File	2 KB

Fig 2. Main Folder



Name	Date modified	Type	Size
__pycache__	12/16/2022 11:50 AM	File folder	
trained_models	12/16/2022 11:50 AM	File folder	
__init__	12/15/2022 4:39 PM	PY File	1 KB
calibration	12/15/2022 4:39 PM	PY File	3 KB
eye	12/15/2022 4:39 PM	PY File	5 KB
gaze_tracking	12/15/2022 4:39 PM	PY File	5 KB
pupil	12/16/2022 12:26 PM	PY File	2 KB

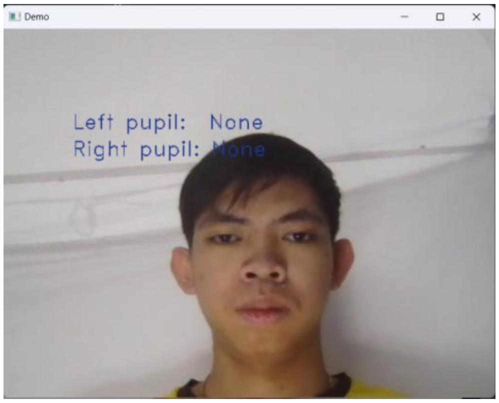
Fig 3. In gaze_tracking Folder

test.py (the main file)

1. The code starts by importing the necessary libraries.
2. It then creates a GazeTracking object and sets it to be used for the duration of the program.
3. Next, it opens up a video file from which we will get frames at regular intervals.
4. The code then starts an infinite loop that waits for new frames from the webcam and sends them to GazeTracking.
5. The next line is where we start getting our first frame from webcam.
6. We send this frame to GazeTracking , which analyzes it and returns us an annotated_frame .
7. This is what we see on screen when looking through the camera's viewfinder.
8. Next, if there are any blinking eyes in this frame, they will appear as text in red with "Blinking" written underneath them.
9. If there are no blinking eyes or right eye pupils present in this frame, they will appear as text with "Looking right" written underneath them instead.
10. If left eye pupils are present in this frame, they will be displayed as text with "Looking left" written under them instead; if both pupil types are present but not blinking (i.e., just looking), they'll show up as text with "Looking center."
11. The code is used to read frames from the webcam and send them to GazeTracking.
12. The code then goes on to analyze the image with GazeTracking, which will produce a new frame that has text overlaid onto it.

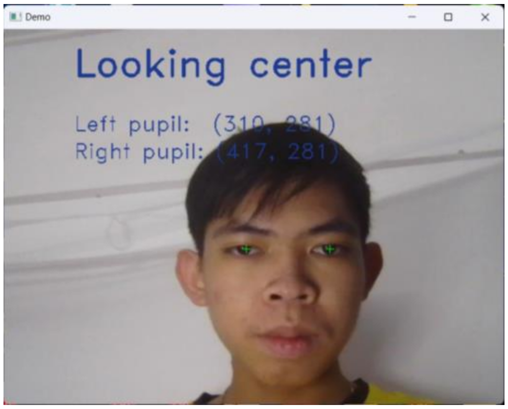
RESULT

As you can see in the demo figure, in figure 1, I am looking a bit away from the webcam, so the program is not fixated on the eyes and nothing happens. You can see on the screen that it says the left pupil and the right pupil coordinate are both none. After that, as shown in Figure 2, the program can detect my eyes and calculate the coordinates I am staring at on the screen; you can see two green dots on both of my pupils, confirming the detection. For the same figure, you can notice that my left pupil is located at (310, 281) on the screen, and the right pupil is located at (417, 281). But if I move my eyes a very little bit on Figure 3, then you can see that the coordinates that have been detected have changed immediately, from the left pupil (310, 281) and the right pupil (417, 281) to the left pupil (294, 274) and the right pupil (402, 271).



```
Left pupil:  None
Right pupil: None
```

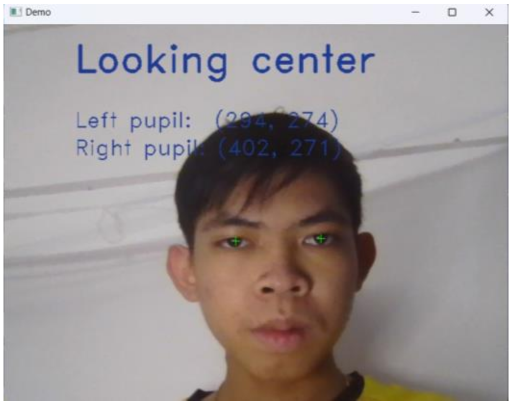
Fig4. Result 1



```
Looking center

Left pupil:  (310, 281)
Right pupil: (417, 281)
```

Fig5. Result 2



```
Looking center

Left pupil:  (294, 274)
Right pupil: (402, 271)
```

Fig6. Result 3

DETAIL PRODUCT FOR ENHANCED PROGRAM

Although the new technology would be amazing, human engineering would still be superior. As I've previously indicated, the improved technology would be similar to how smart gadgets can communicate with one another without the use of hands or fingers. People may easily engage with those gadgets by merely using their eyes to control anything on them. People may use simply hand movements and their eyes to communicate with their phones, especially the phone, for something like to access social media or call someone.

For instance, when someone pulls their phone close to them, the APP will instantly detect it and open. Previously, they could only look at the APP on the phone. In contrast, the user may just push the phone away from him when he has finished using the APP, and it will immediately close. There are numerous other possibilities scenarios, of which this is only one.

It shows how a multi-level design process can create deeper engagement by matching the right level of interaction process to the task users need to complete.

The vision-based interaction system they propose would work by pointing the phone at the user and using the phone’s camera to follow the user’s eyes. The user’s eyes would then be used to control the phone, eliminating the need for hand-based interaction.

The system would be built on top of a deep learning platform, which would be used to interpret the user’s eye movements. While the system is still in the conceptual phase, the authors believe that it has the potential to revolutionize the way we interact with our devices. It would not only be more effective and simpler to use, but it may also enhance the immersiveness of using a smartphone.

PLANNED TECHNOLOGY FOR ENHANCED PROGRAM

But for the enhanced technology I think it is would be more effective and convenient if we could use the Convolution Neural Network which is also known as CNN. From my past experience the CNN is even advance and high accuracy the VGG even though the process is a little bit complicated.

The core of our model is a Convolutional Neural Network (CNN) adapted from the state-of-the-art tracking model presented by Valliappan et al. We used TensorFlow to train this modified CNN model from and estimates the 2D estimate of the gaze in screen coordinate to train a device-calibrated support vector, we first create a gaze feature vector representing each frame of the iPhone's camera. This feature vector includes the output of the final three layers of our CNN, to which we append the following facial features: head yaw, pitch, and roll, the area of the face with respect to the frame, and the on-screen coordinates of the left and right eye corners.

SYSTEM DESIGN

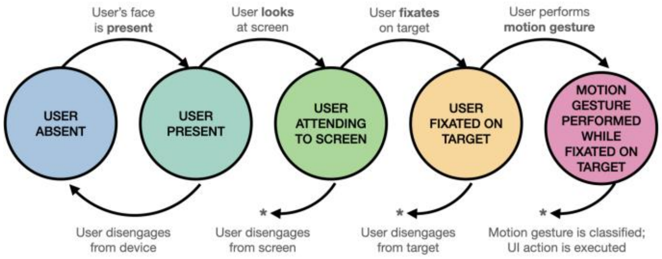

User's face is present
User looks at screen
User fixates on target
User performs motion gesture
USER ABSENT
USER PRESENT
USER ATTENDING TO SCREEN
USER FIXATED ON TARGET
MOTION GESTURE PERFORMED WHILE FIXATED ON TARGET
User disengages from device
* User disengages from screen
* User disengages from target
* Motion gesture is classified; UI action is executed

Fig 7. System Design

The figure above represents the behavior diagram, which represents the entire interaction of the product's operation. We categorized it into five major stages, such as;

1. USER ABSENT

   This stage, which often detects the user's presence, is in charge of the user's absence. It will advance to the next step if the user's face is visible. Otherwise, it will disengage from the devices or keep looping at the same stage.

2. USER PRESENT

   This stage's duty is to check whenever the user looks at the screen. If the user is physically present in front of the screen but does not look at it. This stage is still not activated, so wait until the user looks at the screen. Otherwise, it will disengage from the screen.

3. USER ATTENDING TO SCREEN.

   The program's task at this point is to identify what the user intends to do with their eyes, such as which app they wish to launch. Does the user seem to be fixated on the target? If not, the software must stop interacting with the screen.

4. USER FIXATED ON TARGET.

   After that, the program can detect which app is the target of the user now. The program will wait for the user's command to perform a motion gesture. Which of these actions do these users want to take? Otherwise, the user will disengage from the target if the program cannot understand the user's command.

5. MOTION GESTURE PERFORMED WHILE FIXATED ON TARGET.

   The last stage is in charge of the program's logic, therefore getting to it requires successfully completing the previous four stages without making any mistakes. This indicates that the motion gesture has been classified and the UI action is prepared to be carried out in accord with the command order.

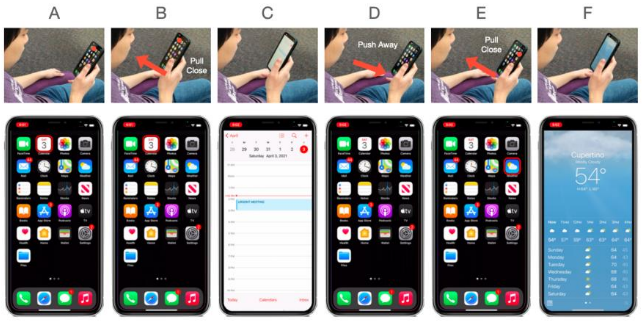
EMOTIONAL INTERACTION

The emotional interaction mechanism defines how emotions interact with one another. Emotions can have an additive or subtractive influence on other emotions. When one emotion is stimulated, it may influence the behavior of another emotion, and this affection is referred to as emotional interaction. Anger, surprise, disgust, delight, fear, and sadness are examples of fundamental emotions.

Using this technology allows you to track the user's emotion as well as their gaze gesture. Deep learning is used to determine the user's current emotional state, and content recommendations are made based on that information. If a user is grinning and appears cheerful, the material should have more joyful content; nevertheless, even if a user appears unhappy, we should still give motivational information.

USE CASE

Scenarios 1:



Home Screen: In this demo, a user looks at the calendar icon on their home screen (A) and pulls the phone closer (B) to open it (C). The calendar is minimized with a push action (D). The user then fixates on the weather icon and pulls the phone closer (E) to open the app (F).

Scenarios 2:

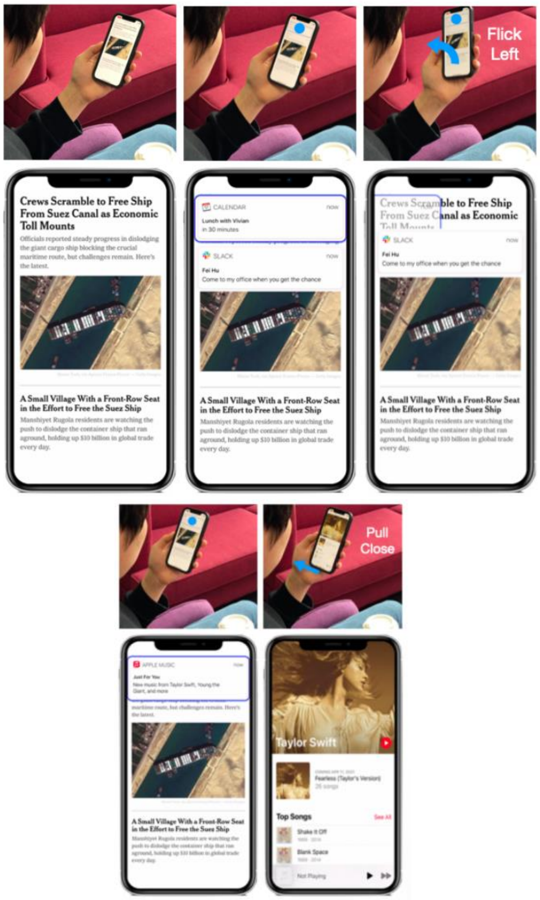


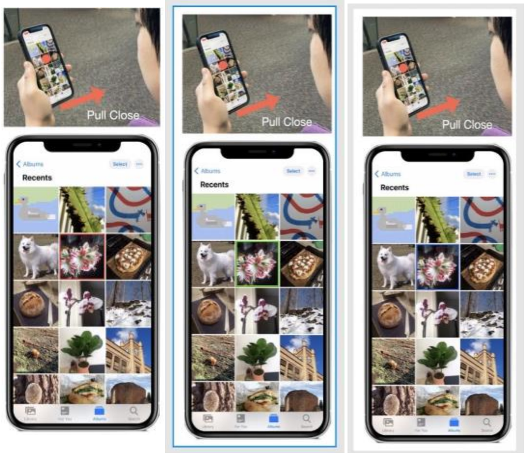
Fig.A Fig.B Fig.C

Fig.D Fig.E

When two alerts arrive, a user is reading the news (A) (B). They read the first one and then swipe left to dismiss it (C). The user then reads the other notice and swipes right to snooze it (D). A third notice appears as the user takes a drink of their coffee (E). The user draws their phone closer (G) to activate the app after reading it and wanting to see more (F). The User associated application the user has never had to touch the screen at any point.

ALTERNATIVE DESIGN

Since the key aspect of this project is the logical approach in which this application is intelligent enough to connect with smart devices, not much user interface is needed for it. In view of this, "alternative designs" don't exist. The color that surrounds the identified app or the color scheme for the entire interface, however, are the only potential replacements for it. As seen in the picture below, the color ringed around the app that is being identified is red. The user has several options to select their favorite color for the different design concepts as a consequence.



BENEFIT OF THE PROJECT

I chose this topic for this research because I can clearly understand the huge benefits of this innovative idea. No matter how well-designed a project is, there will always be a negative. It is also rather common for technology to have drawbacks. The disadvantage, however, is insignificant when compared to the benefits of this innovation.

Advantage:

1. Help Disable people
2. Increase the accessibility
3. Convenient to use
4. Rapid interaction between the phone and the users.
5. Let the user control the device without touching it.
6. Has huge advantage on disabled and people who is not convenience to use finger.
7. Reduce distraction.

Disadvantage:

1. The price might be a little bit increase.
2. Might have some difficulties for eyes.
3. Technology not be well-known by people yet
4. Users need to get used to this kind of technology

For my personal growth:

1. To ensure that the capabilities match the current standards in the market.
2. To maintain one's credibility in the job market.
3. To enhance knowledge and improve the skills needed to deliver the best service
4. To ensure the knowledge and skills are up to date with the existing market conditions
5. To make a useful contribution to the growth of the organization

EXPECTED IMPACT

Global: I cannot assure that this has big impact on the Global stages, but I am pretty sure that this also help disability people be for more access to the smart phone much better than before.

Economy: Increasing the useable will undoubtedly have a significant influence on the economy, increasing employment opportunities as well as sales. 15% of the world's population, or one billion individuals, are disabled in some way. Therefore, it must be estimated that at least a few hundred million individuals will not be able to use a smart phone. nonetheless, the world economy will benefit greatly from this approach. [7]

Environmental: The positive thing about the environment is that it won't harm or have an influence on it. Because it requires just one unit of machine learning code, it can already be used on mobile devices.

CONCLUSION

Artificial intelligence (AI) and eye interaction have a long history, both with a great future ahead. Several researchers have been working in this direction to evolve the interaction between humans and devices into a natural one. Within this segment, the area of exploitation is vast, and the potential use cases range from human-computer interaction to human resource management.

ACKNOWLEDGMENTS

I would like to express my gratitude to my professor, Ph.D. LONG Jin Yi, who guided me throughout this research or the report paper and also thank for his awesome and perfect lecture performance. I would also like to thank my friends and family who supported me and offered deep insight into the study.

REFERENCES

[1] Computer vision - Wikipedia. Computer Vision - Wikipedia 2018. https://en.wikipedia.org/wiki/Computer_vision

[2] Convolutional neural network - Wikipedia. Convolutional Neural Network - Wikipedia 2019. https://en.wikipedia.org/wiki/Convolutional_neural_network

[3] yihuacheng. GitHub – yihuacheng Itracker: Gaze estimatin code. The Pytorch Implementation of “Eye Tracking for Everyone”. GitHub 2021. https://github.com/yihuacheng/Itracker.

[4] Eye Tracking for Everyone. Eye Tracking for Everyone n.d. https://gazecapture.csail.mit.edu/

[5] EyeMU Interactions: Gaze + IMU Gestures on Mobile Devices. YouTube 2021. https://www.youtube.com/watch?v=-HwcmWRAsaA

[6] Flores J. Training a TensorFlow model to recognize emotions. Medium 2018. https://medium.com/@jsflo.dev/training-a-tensorflow-model-to-recognize-emotionsa20c3bcd6468

[7] Disability Inclusion Overview. World Bank n.d. https://www.worldbank.org/en/topic/disability