Boston University

Electrical & Computer Engineering

EC464 Senior Design Project

Final Prototype Testing Report

Team 20 SwingOn

Team Members:

Yoel Beyene

Tingru Lian

Hanlin Mai

Jessica Martinez Marquez

Equipment

The final prototype testing consists of assessing the application's hardware and software elements. The user of the app should have a device of iPhone 6S or newer. The system requirement is iOS 15 or iPadOS 15, as well as another device running on macOS Catalina. Both devices should be set up to run and simulate the SwingOn app.

The software part of the prototype testing consists of XCode 13 as the development environment. We choose Swift as the app's programming language, and SwiftUI as the user interface toolkit for designing the app presented in prototype testing.

The SwingOn app estimates the spatial location of a user's 17 key body joints from live video using the PoseNet pose detection model running on the iOS CoreML framework. Additionally, a tripod is used for the stabilization of our device for testing the app. One of our team members acts as the user to perform some movements in front of the camera to demonstrate the model's results.

Setup

The frontend and backend of the code is implemented in XCode13. Therefore, to test it on an iPhone or other iOS devices, devices need to be connected to the Mac that the project is written in. Then, they will download the app from the Mac computer to the corresponding device. The device then can be adjusted on a tripod to ensure a stable position. In order for their full body to be captured, the phone needs to be placed at least 2 meters away. This is to maintain a consistent laboratory environment to achieve the best result accuracy when performing the test of detecting the user from a background with many people. Also, videos of example golf swings

need to be downloaded to the local files or captured to photo albums in order to test the upload function of the app.

Measurements

The SwingOn app should be able to successfully run on any iOS device that is updated to the latest version, specifically, on the iPhone that is being used to test. Each screen and all of its elements should be displayed with a clear organization. The main screen should have the "SwingOn" logo, a welcome message, a map showing the user's location, and three buttons–one for the camera screen, one to upload videos from the phone's Photo Library, and one to upload videos from the phone's Files. If the user presses the "Open Camera" button, the camera screen should appear along with a "Back" button that allows the user to return to the main screen. The camera screen should ask the user for permission to access and open the camera. When permission is granted, the camera screen should display a person's body pose in real-time (>30fps), as well as allow users to make adjustments to the model. There should be a total of 18 key points detected on a person's body, including nose, left and right eye, left and right ear, left and right shoulder, left and right elbow, left and right wrist, left and right hip, left and right knee and left and right ankle as well as the centroid. The centroid should also be calculated based on the analysis of other points and displayed. The coordinates should be shown on the camera screen and indicate the location where a keypoint exists in the video. The goal is to have a real-time analysis of a golfer's motion using a camera as well as enable users to upload a video for analysis. Therefore, the upload feature of the app should allow the user to upload pre-recorded videos both from local files and the Photo Library to the app and analyze them by applying the pre-trained PoseNet model.

Different methods of centroid analysis should be shown to professors. Which includes the graph of centroid over time of several videos of both good and bad swings. The distance of the centroid changes over time compared to the original position and the degree the change should be calculated and will be shown in the testing. Several videos of both good swings and bad swings, which are from reliable sources, have been collected for analysis and shown.

Conclusion

For the final prototype testing, some requirements needed to be implemented for the app to perform more designated tasks, such as the threshold or classification have not been set to determine good or bad swing. Even though the centroids have been calculated and stored, the threshold has not been set yet. This threshold will be calculated through analysis of 100+ good/bad golf swing videos, and be based on the algorithm provided. The threshold will be the acceptable number range based on our calculations. Another part that needed to be added is the feedback for users. When the app detects users are doing a swing that is not balanced, the users will receive feedback on their swing based on the analysis and calculations. According to the feedback given, the user will be able to adjust their swinging technique and mechanics accordingly.

Different machine learning methods should be tried for the classification of whether it is a good or bad swing. For example, k-mean clustering is a useful method for grouping points and finding clusters in the data set. Convolution Neural Network (CNN) is a deep learning neural network that has hidden layers in contrast to traditional techniques, which can be very useful in

terms of pattern recognition. Both of these methods will be developed later to aid the classification of good or bad swings for our app.

After the update of iOS 15, the user interface of the app needs to be completely written in SwiftUI. However, part of our app's user interface is written in Swift. Thus, our app may not run smoothly on the newest iOS version. We are still trying to switch the code written in Swift to SwiftUI.