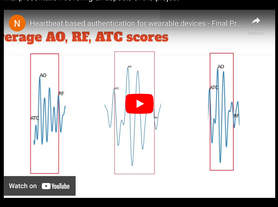
Heartbeat Based Authentication in Wearable Devices

Introduction

As part of COMP6733 (Internet of Things (IoT) Design Studio), our major project was a heartbeat based authentication system. This was a group project undertaken with my friend Solomon and Nick. The project involved using a sensortag (pictured) to authenticate a person based on their heartbeat.

This page only summarises the project so for a full picture of the project please see the final report or final presentation in the video below. You can view the final report here.

The video below is our final presentation covering all aspects of the project



Background

The project was designed with home quarantine during the COVID-19 pandemic in mind. To ensure a person in home quarantine stays at home, you can place a wearable tracking device on them which alerts authorities if home quarantine is breached. But how do you know that the person hasn't simply taken the device off or handed the device to someone else in the house to wear? This is where the heartbeat based authentication comes in. Our system can be used in conjunction with a wearable tracking device to authenticate the person wearing it. This project is based on the paper Unlock with Your Heart: Heartbeat-based Authentication on Commercial Mobile Phones [1].

Project Specification

In this project we created a heartbeat-based authentication system that can be used in a wearable device (the "wearable device" for this project is a CC1350 SensorTag (pictured below). The system attempts to accurately authenticate a person based on their heartbeat pattern.

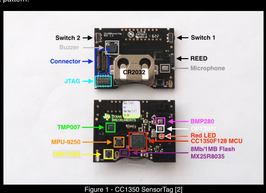


Figure 1 - CC1350 SensorTag [2]

To authenticate a person, their heartbeat pattern is collected by the accelerometer (the MPU-9250) on the sensortag, by pressing it against the user's chest. The collected pattern then undergoes various data pre-processing steps (including feature extraction - more on this below) and is then inputted to a support vector machine (SVM) model to compare it against the person's heartbeat pattern as well as other heartbeats. The SVM model will give the likelihood that the given heartbeat belongs to the owner or not and if this likelihood is above a certain threshold, then the person is successfully authenticated.

For the project, there are 2 use cases of the system: training and authentication.

Training

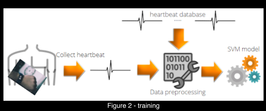


Figure 2 - training

Training involves using the collected heartbeat pattern to train the support vector machine (SVM) model.

Authentication

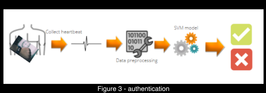


Figure 3 - authentication

Authentication involves using the collected heartbeat pattern as an input to the support vector machine (SVM) model. The SVM model which will give the likelihood of whether an unknown heartbeat signal belongs to the owner or not.

Implementation

The implementation method follows the process outlined in the paper this project is based off [1].

Collection

As described before, the sensortag is used to take heartbeat samples. With the sensor tag we have been able to collect a 7 sec sample at 250hz of the z-axis of a heart beat and transfer the data to a computer to process with python. There were some changes required to the MPU-9250 contiki driver as the device casted the reading from unsigned integer to a floating point back to a signed int losing precision as the least significant bit is quite important for our use.

The sample collected was precise enough to detect heartbeat with enough precision which we believe to do the heartbeat authenticatication. Collection and networking (sending the data from the sensortag to a computer) was a difficult process and further details can be found in the final report.

Heartbeat segmentation

As the sensor tag collects heartbeat in a continuous manner, we have to separate it into individual heartbeats.

Since an average person has a heartbeat ranging from 50bpm to 120bpm, we divide the collected heartbeat into sections of 2 seconds to guarantee that at least 1 full heartbeat cycle can be collected in each section. Figure 4 below shows a heartbeat pattern being separated into 2 second intervals.

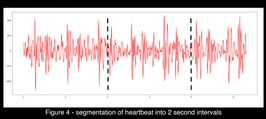


Figure 4 - segmentation of heartbeat into 2 second intervals

We then try to segmentate out a heartbeat in a section. The heartbeat cycle contains features such as the AO and RF. These features can be used to correctly segment the heartbeat pattern. In each 2 second interval - we will extract a single heartbeat. We do this by identifying the AO and RF using the shortest distance between 2 peaks that's greater than 200ms starting from the highest peak. This performs well in trying to find the AO and RF as AO to RF intervals are larger than 200ms when the heart rate is slower than 120bpm. Now that we have obtained the AO to RF distance, we simply multiply 0.5 to it to obtain the ATC to AO distance and from that we can obtain a full heartbeat cycle.

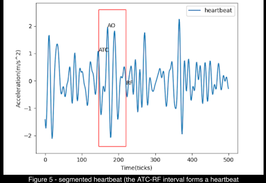


Figure 5 - segmented heartbeat (the ATC-RF interval forms a heartbeat

here is a comparison between a heartbeat pattern from the paper and one of our segmentated heartbeats. As you can see our algorithm captures a heartbeat just as described in the paper.

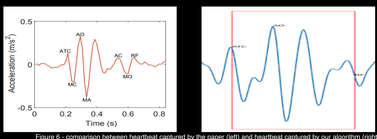


Figure 6 - comparison between heartbeat captured by the paper (left) and heartbeat captured by our algorithm (right)

Feature extraction

Now that we extracted heartbeat samples, the next task was to extract features from these samples to be used in the support vector machine (SVM) model.

The paper outlined numerous methods of feature extraction. Our team tried variations of these methods. The first method that was implemented was taking the average amplitude of all heartbeats during the different stages of a heartbeat cycle - however this gave poor results.

The feature extraction method that gave best results was discrete wave transform (DWT) - implemented using the pywavelets implementation. The DWT decomposes SCG signals into five levels with level 1 to 5 represent signal components in the frequency range of 25 ~ 50 Hz, 12.5 ~ 25 Hz, 6.25 ~ 12.5 Hz, 3.13 ~ 6.25 Hz and 1.56 ~ 3.13 Hz, respectively. To reduce noise in the SCG signal, we only use the detailed coefficients from the second level to the fourth level as the feature vector for heartbeat authentication.

The result of this was a 56 dimensional feature vector.

SVM training and testing

Our team used the sklearn implementation of the SVM. The SVM was created with the kernel parameter "linear" and all other parameters default. Each 56 dimensional vector created from each heartbeat formed a row inside a Pandas dataframe. For both training and authentication, all feature columns were passed to the SVM as data and the last column called "target" was passed as the label column.

Our team collected heartbeat patterns to be used as a heartbeat database for SVM training (forming negative cases and allow the SVM something to compare the given heartbeat sample against to make a prediction).

For authentication, the user passes in a new heartbeat pattern and this data is passed to the SVM predict attribute which will return a value of 0 (meaning this is the wrong person) or 1 (meaning this is the correct person and they are successfully authenticated).

Evaluation and conclusion

5-fold cross validation was used to evaluate the accuracy of the SVM model on each of the feature extraction methods. The method cross_val_score was used from sklearn to calculate the 5-fold cross validation score.

The accuracy returned was **76%**.

With an accuracy of 76% this system needs further improvements before it can be used for authentication as this accuracy will not protect the system against attackers and will also deny legitimate users at a rate too high. An accuracy closer to 95% would be appropriate before this system can be used for authentication as attackers would be denied and legitimate users blocked. Some improvements that can be made are: more data, more consistent data and further SVM fine tuning. With these improvements I am confident an accuracy closer to 95% can be achieved.

For more details on any aspect of the project please read the final report.

References

[1] Lei Wang, Kang Huang, Ke Sun, Wei Wang, Chen Tian, Lei Xie, and Qing Gu. 2018. Unlock with Your Heart: Heartbeat-based Authentication on Commercial Mobile Phones. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 3, Article 140 (September 2018), 22 pages. DOI:https://doi.org/10.1145/3264950

[2] SensorTag image source