

*Charlie Maclean  
King's  
cm927*

Diploma in Computer Science Project Proposal

## **Synthesis of Heart-Rate Detection Methods**

*December 18, 2019*

**Project Originator:** Dr Robert Harle

**Project Supervisor:** *Dr Robert Harle*

**Signature:**

**Director of Studies:** *Dr Timothy Griffin*

**Signature:**

**Overseers:** *Dr Anil Madhavapeddy and Professor John Daugman*

**Signatures:** *<no need to obtain Overseers' signatures yourself>*

# Introduction and Description of the Work

Heart-rate signals from watches are unreliable while exercising. Watches make use of photoplethysmography (PPG) sensors - sensors which detect the volume of blood in the skin and use variances in this to reconstruct a heart-rate. PPG sensors are preferred to the more accurate electrocardiogram (ECG) due to user comfort. However, the signals they provide are harder to process - I want to compare strategies to process these signals to extract heart rate.

There are several sources of noise within a PPG signal. There is often high frequency contamination caused by electrical interference or light from external sources. Additionally, there is a constant low frequency variation in the DC background of the signal, as a result of capillary density, blood volume and temperature variations.

In the context of running, motion caused by the arms swinging forward and back causes the sensor to slide along the skin, creating motion artifacts (MAs). These are particularly challenging as they can have a much larger amplitude than the pulse we are looking for. Additionally, they can be at the same frequency as the heart rate signal, making them challenging to filter out. Therefore, there is research [1][2] that suggests using accelerometer data in order to predict MAs. I will implement algorithms which find the MAs based on data from the accelerometer. Following this, I will look into the implementation of filters to remove these MAs.

In order to get accurate heart-rate measurements with which to compare the PPG signals, I will make use of an chest-mounted portable ECG. I will need to synchronize the signals received from the ECG and PPG, due to clock skew between the different devices. Potentially, there could be drift between the two clocks as well.

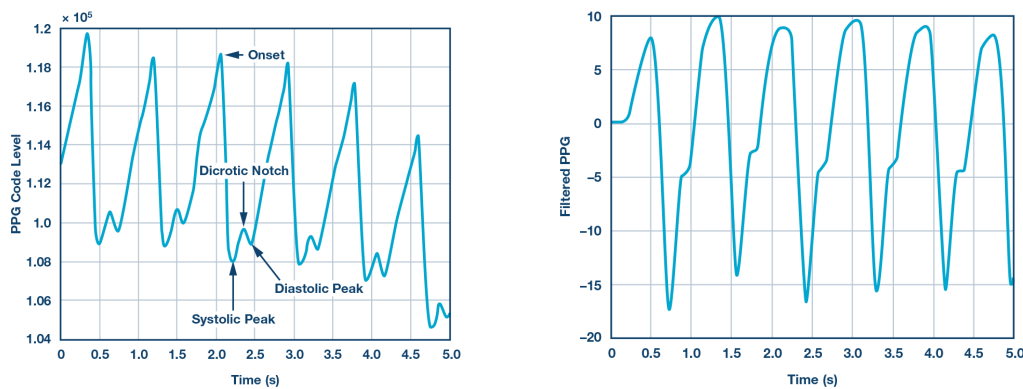


Figure 1: PPG signals before and after filtering [3]

# Starting Point

## Wearable Development

I will need to develop an application for the smart watch which will record PPG and motion signals without interruption. I will use Android Studio and Kotlin.

I have never used Kotlin before and have used Android Studio once before, but never to develop something for a wearable.

## Digital Signal Processing

Manipulation of the PPG signals I receive will require much digital processing, and there are two languages I am considering using: MATLAB and Python.

I have never used MATLAB before, but I am familiar with Python.

# Substance and Structure of the Project

## Core

1. Developing a wearable application to capture PPG signals and motion data. I would like to develop this using Android Studio and Kotlin. I will need to make use of wake locks in order to ensure that the application can continuously record data.
2. Collecting data using a PPG-enabled watch and a portable ECG. I will record my heartbeat over the course of several runs.
3. Synchronizing signals. Implementation of an algorithm to synchronize the data output from the watch records to the data output from the portable ECG. I will have to develop a program to find the lag between the two signals.
4. Removing noise. Due to the various sources of noise, I will investigate potential low and high pass filters, to remove both high and low frequency disturbance.
5. Peak finding - implementing algorithms to find the actual beat given a clean PPG signal. Some potential algorithms are:
  - (a) Adaptive threshold [4]
  - (b) Wavelet transformation [5]
6. Finding MAs - implement an algorithm which uses the accelerometer in order to detect segments of the PPG which are likely to have been affected by motion.

7. Removal of MAs - implementation of filters to remove the previously detected MAs.

## Possible Extensions

- Investigating PPG-enabled earbuds
  - Evaluation of the quality of heart-rate provided by earbuds
  - Exploring the potential to merge signals from a smartwatch and earbuds in order to provide a higher quality signal.
- There is research [6] to suggest gyroscope information is also helpful in filtering out MAs. I could include gyroscope data in my MA filtering technique.
- Comparing more heart-rate detection algorithms:
  - Digital filters
  - Adaptive filters
  - Singular value decomposition
  - Empirical mode decomposition
  - Spectrum analysis

## References

- [1] Z. Zhang. Photoplethysmography-based heart rate monitoring in physical activities via joint sparse spectrum reconstruction. *IEEE Transactions on Biomedical Engineering*, 62(8):1902–1910, Aug 2015.
- [2] Z. Zhang, Z. Pi, and B. Liu. Troika: A general framework for heart rate monitoring using wrist-type photoplethysmographic signals during intensive physical exercise. *IEEE Transactions on Biomedical Engineering*, 62(2):522–531, Feb 2015.
- [3] Foroohar Foroozan. Music-based algorithm for on-demand heart rate estimation using photoplethysmographic (ppg) signals on wrist, 2018.
- [4] Ivaylo I. Christov. Real time electrocardiogram qrs detection using combined adaptive threshold. *BioMedical Engineering OnLine*, 3(1):28, 2004.
- [5] Jake D. Campbell, Christopher G. Pretty, J. Geoffrey Chase, and Phillip J. Bones. Near-real-time detection of pulse oximeter ppg peaks using wavelet decomposition. *IFAC-PapersOnLine*, 51(27):146 – 151, 2018. 10th IFAC Symposium on Biological and Medical Systems BMS 2018.

- [6] Alexander J. Casson, Arturo Vazquez Galvez, and Delaram Jarchi. Gyroscope vs. accelerometer measurements of motion from wrist ppg during physical exercise. *ICT Express*, 2(4):175 – 179, 2016. Special Issue on Emerging Technologies for Medical Diagnostics.

## Success Criteria

The following should be achieved:

- Develop an application which records and stores PPG signals on a watch.
- Create program which synchronises ECG signals with PPG signals.
- Implement at least two filtering algorithms, demonstrate filtering works by displaying signals before and after filtering.
- Implement at least two peak finding algorithms, demonstrate they work by comparing peaks on the PPG signal to peaks on the ECG signal.
- Implement a MA detection algorithm.
- Be able to remove MAs, demonstrating that the peak finding algorithm is not affected by signals caused by motion.

## Resources Required

I will use my own laptop. I will regularly backup my project to GitHub and an external HDD, so that I can recover data in the event of hardware failure. I accept full responsibility for this machine and and I have made contingency plans to protect myself against hardware and/or software failure.

I will use a smart watch, and a chest-mounted ECG sensor in order to collect data.

## Timetable and Milestones

### Weeks 1-2 (28/10/19 - 10/11/19)

- Project set-up:
  - Installation and setup of Android Studio.

- Investigate and install an IDE for Python or MATLAB.
- Use GitHub to set up a backup system for the project.
- Learn to use Kotlin for the wearable app.
- Create a wearable app to record and store PPG and motion data, using Android Studio and Kotlin. Research into wake logs and interface design.
- Additionally, test application by recording data while running.

### **Weeks 3-4 (11/11/19 - 24/11/19)**

- Collect data from runs.
- Build program to synchronize data between ECG and PPG (based on either heart rate variation or motion information).
- Researching and then implementing filters to remove both high and low frequency noise from the PPG signal.

### **Weeks 5-6 (25/11/19 - 08/12/19)**

- Implement program to detect peaks in the signal.
- Begin writing dissertation document.

I will take the next week off.

### **Weeks 7-8 (16/12/19 - 29/12/19)**

- Implement MA detection algorithm.
- Continue writing dissertation.

### **Weeks 9-10 (30/12/19 - 12/01/20)**

- Implement MA removal algorithm.
- Continue writing dissertation.

### **Weeks 11-12 (13/01/20 - 26/01/20)**

- Begin extension work, testing earbud HR detection against wrist-watches and portable ECGs.
- Continue writing dissertation.

**Friday 31st January - Progress Report due**

### **Weeks 13-14 (27/01/20 - 09/02/20)**

- Extension work, testing whether earbud HR can be merged with wristwatch signal to create a more accurate output.
- Continue writing dissertation.

### **Weeks 15-16 (10/02/2020 - 23/02/2020)**

- Extension work, including gyroscope feedback into the MA detection algorithm.
- Continue writing dissertation.

### **Weeks 17-18 (24/02/2020 - 08/03/2020)**

- Extension work, including gyroscope feedback into the MA detection algorithm.
- Continue writing dissertation.

### **Weeks 19-20 (09/03/2020 - 22/03/2020)**

- Writing dissertation.

### **Weeks 20-21 (23/03/2020 - 05/04/2020)**

- Writing dissertation, aim to have first draft by second week of Easter holiday.

## **Weeks 22-26 (06/04/2020 - 08/05/2020)**

- Continue improving dissertation.
- Leaving time for exam preparation.

**Final deadline for dissertation 08/05/2020**