MOTIVATION

- Enhancing the live concert experience by incorporating interactive visuals.
- It is believed that visuals that reflect the mood and emotions of the audience improves the audience-musician interaction.
- The feasibility of readily available and easy to use digital devices that are capable of externalizing the proposed concept.

INTRODUCTION

- PULSE is an infrastructure for collecting audience heartbeats during a live musical performance and using this data to generate real-time visualizations.
- The prototype uses a specially designed pulse sensor to detect a user’s heart rate. The sensor is a small device having a uniquely identifying QR code that comfortably fits on the tip of a finger. It also boasts an accelerometer and Bluetooth that helps the sensor pair with a mobile device.
- The mobile application scans the QR code and connects with the sensor. It then starts receiving a stream of heartbeat data from the sensor in real-time. It also periodically pushes the collected data to a Server. The user can also see his live heart rate on the application.
- The pulse server that is hosted centrally collects the pulse data from all the mobile devices which it has connected to. The server maintains a database schema to systematically store this influx of data. It exposes this data via web API’s so that the visualization application can query for it.
- The visualization application on receiving this data uses a software that transforms the raw heart beat data into amazing visuals.

OBJECTIVES

- To focus on enhancing the pulse sensing module and making it robust.
- Improve the pulse detection algorithm to eliminate noise interference that leads to unreliable pulse data.
- Implement mechanisms to dynamically adjust the pulse sampling rate and data sending rate in order to improve efficiency.
- Integrate with the Android application and Server modules.

IMPLEMENTATION

ACCELEROMETER BIASED ACCURACY:

- The pulse sensor contains an inbuilt accelerometer that is used to detect substantial movement; a scenario where the sensor is susceptible to noise.
- The algorithm checks for successive sensor positions to find an instance where the axis readings cross a certain pre-defined threshold. It then skips the corresponding heart beat readings taken at that instance.

LOW PASS FILTER:

- A filter that detects and eliminates sudden spikes in the heart rate in real-time.
- It refines the readings by detecting anomalies from the regular pattern of a user’s heart beat.

EXPOSING API’S:

- The sensor module exposes API’s that the Server can use via the mobile application to tweak the sampling and transfer rate for necessary behavior.
- The API’s accept certain parameters and dynamically sets the requested rate within the sensor module.

INTEGRATION WITH OTHER MODULES:

- Supporting authentication using device-code hash comparison as required by the mobile application.
- Allowing the Server a functionality to dynamically adjust different rate parameters.

RESULTS

- In the original algorithm, a substantial movement of the pulse sensor causes the heart rate to deviate from its ongoing range, rendering it unreliable.
- The improved algorithm overcomes this inconsistency, it maintains a stable heart rate in response to significant sensor movements.

CONCLUSION AND FUTURE WORK

- The improved pulse detection algorithm eliminates noise interference up to a considerable extent, hence, producing more accurate readings.
- Future work could involve capturing the raw signal data and using it to construct real-time heart rate waveforms on the mobile application.
- Features can be extracted from the raw signal data in order to perform correlations with the music waveform.

REFERENCES

1. Joe Geigel et. al. “PULSE: An Infrastructure for collection of audience heartbeats for music visualization”.