Total Knee Arthroplasty Recovery Monitoring Using a Wearable Device and Machine Learning Analysis

Alireza Borjali

(Team: Solo Project)

DGMD S-14 (34484)

Final Project

ABSTRACT

Trauma or degenerative joint diseases can affect the structure of the knee joint and deteriorate its function, causing pain and disability. Total knee arthroplasty (TKA), also known as total knee replacement (TKR), is the most common surgical intervention to treat severe degenerative knee joint diseases. Post TKA clinicians monitor the patient's progress and especially gait to evaluate the success of the surgery. These sessions usually include house visits that can be expensive and due to Covid-19 risky. The goal of this project was to design a wearable device that the patients can wear on their knees to track their progress post TKA surgery. We used SensorTile on a cradle and used a strap to attach it to a patient's knee. We used Bluetooth communication between the SensorTile and BLE sensor app to collect data and used python to analyze the data. We collected 101 data points from three participants simulating three types of gaits: 1) normal walking, 2) limping, and 3) running. We subsequently designed a 1-D convolutional neural network (1-D CNN) to analyze the gait data and categorize it into "normal (walking)" and "abnormal (limping and running)" gait. The CNN achieved 80% accuracy on the test data (30% of the whole dataset). Patients can use this device to track their progress post-TKA surgery and self-report their progress. Clinicians can analyze the patients' progress and only visit the ones that are not recovering as expected and continuing to have an abnormal gait after the expected recovery period of 2-3 weeks.

1.Introduction

1.1 Knee joint functionality

The knee joint is part of the musculoskeletal system and enables 3D movement of the leg while supporting the body’s weight. It is the largest load-bearing joint in the human body. The knee joint rotates around three axes and translates along these axes during normal gait. The three main rotations are defined as adduction/abduction, lateral/medial, and flexion/extension rotations. The adduction/abduction motion is a rotation of the knee joint, in the coronal plane, toward and away from the body, respectively. The lateral/medial rotation occurs in the transverse plane. A lateral rotation moves the ventral surface of the knee joint away from the midsagittal plane, whereas a medial rotation brings the ventral surface of the knee joint toward the midsagittal plane. Flexion/extension describes the rotation of the knee joint in the ventral and dorsal directions, respectively, parallel to the midsagittal plane. Although flexion/extension is the dominant rotation with the most extended range of motion compared to the other knee joint rotations, and because the leg swings forward during gait, all knee joint rotations are crucial to maintaining a normal gait.

1.2 Total knee arthroplasty (TKA)

Trauma or degenerative joint diseases can affect the structure of the knee joint and deteriorate its function, causing pain and disability. Clinicians initially recommend nonsurgical treatments such as weight loss, physical therapy, using a walking aid, and medication to patients suffering from such knee joint disorders. However, cases where non-surgical treatments fail to relieve pain and restore normal, pain-free knee joint function, may ultimately require surgical intervention. Total knee arthroplasty (TKA), also known as total knee replacement (TKR), is the most common surgical intervention to treat severe degenerative knee joint diseases. TKA replaces the worn or defective natural knee joint with a prosthetic knee implant to relieve pain and restore knee joint functionality and normal gait.

1.3 Post-op surveillance

Post TKA clinicians monitor the patients' progress post-op to evaluate the success of the surgery. These evaluations include routine physical therapy sessions to assess the quality of gait and measure the patient’s progress towards regaining a normal gait function. These sessions usually include house visits, especially for older patients, which can be expensive and increase the overall healthcare costs. Furthermore, in recent years due to the Covid-19 infection house visits have become increasingly more expensive and riskier especially for the vulnerable elderly population. An alternative solution that can track patients’ recovery could reduce overall healthcare costs and improve the quality of care delivered to patients.

1.4 Goal

Wearable devices are becoming an integrated part of modern life. Many of these devices are already in use delivering health analytics. The goal of this project was to design a wearable device that the patients can wear on their knees to track their progress post TKA surgery. This device can distinguish between a “normal” and “abnormal” gait. Patients can use this device to track their progress post-TKA surgery and self-report their progress. Clinicians can analyze the patients' progress and only visit the ones that are not recovering as expected and continuing to have an abnormal gait after the expected recovery period of 2-3 weeks.

1.5 Related work

There are many studies on human activity recognition, for instance using smartphone sensors [1,2], and other custom-designed wearables[3]. There are also commercial wearables specifically designed for monitoring TKA patients. For instance, TracPatch (https://tracpatch.com) is an orthopedic wearable for gait analysis for monitoring TKA patients among other functionalities. The present work would have a subset of TracPatch functionalities, but we would be able to custom fit the product for each patient as opposed to a mass-produced product such as TracPatch.

2. Materials and Methods

2.1 Sensor configuration

We used SensorTile as the main sensor to collect data (Fig. 1). Based on our clinical experience we decided to capture the gyroscope data to capture knee rotations.

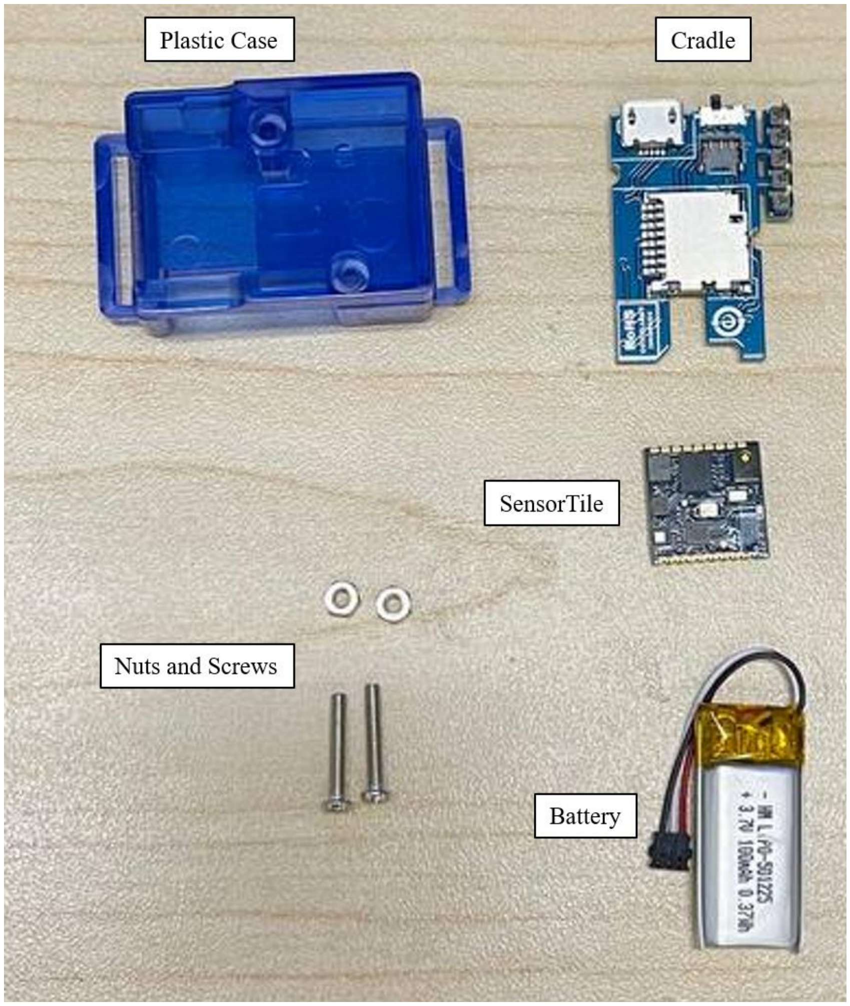


Fig. 1 SensorTile assembly

In the first step, the SensorTile was glued to the Cradle and the corresponding pins were solder together. Next, the battery was connected to the cradle and all the components were put into the plastic case. Subsequently, the SensorTile was flashed with the firmware from Tutorial 8. (please refer to ref. 1). In the last step, we used a strap to attach the plastic case to a patient’s knee (Fig. 2).

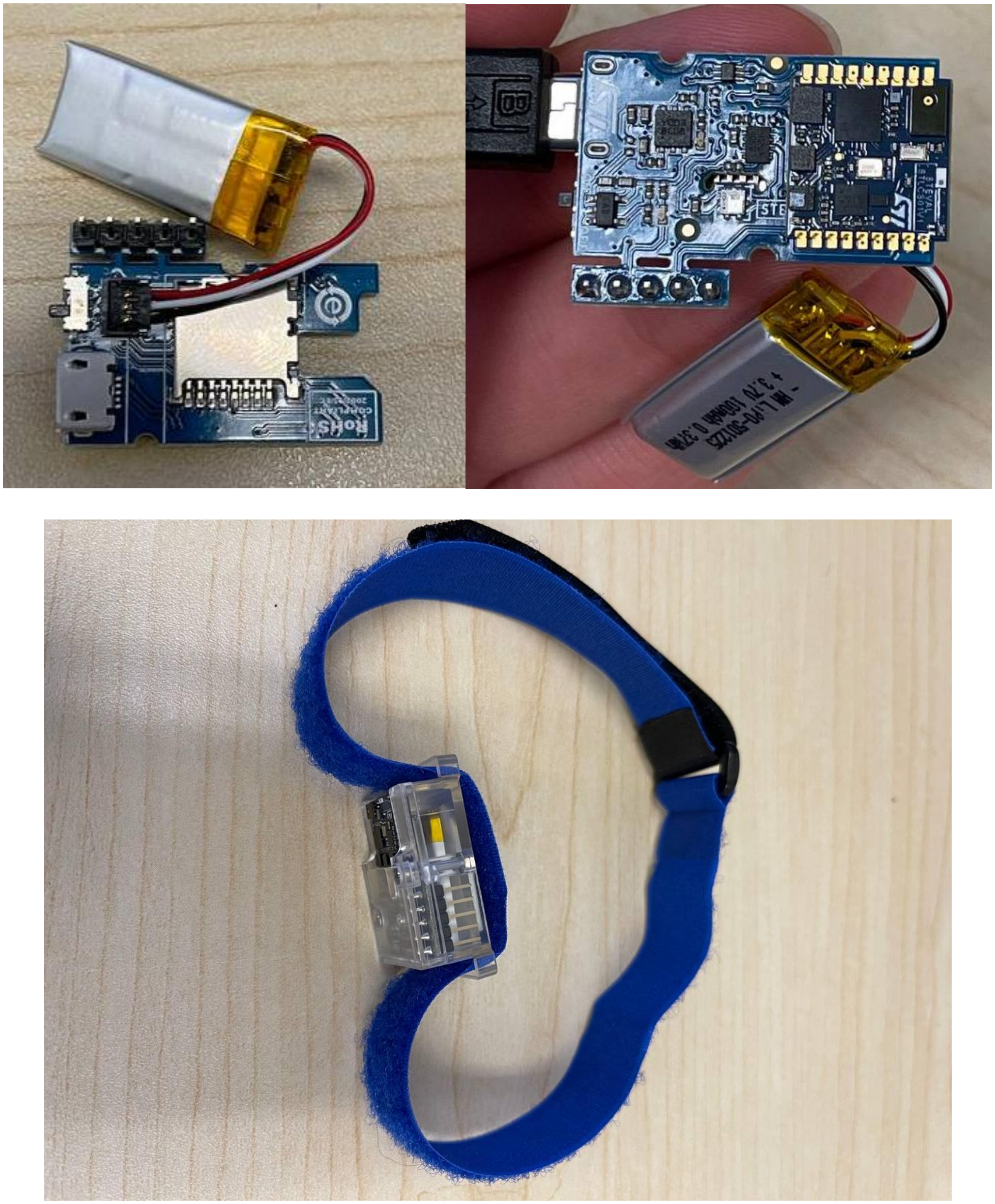


Fig.2 SensorTile assembly

The whole data acquisition system comprised of a single SensorTile strapped to the patient's knee below the anterior horn of the tibia to accurately capture gait. SensorTile communicated via Bluetooth to a smart phone using ST BLE Sensor app and the data were sent via email to a PC for analysis (Fig. 3).

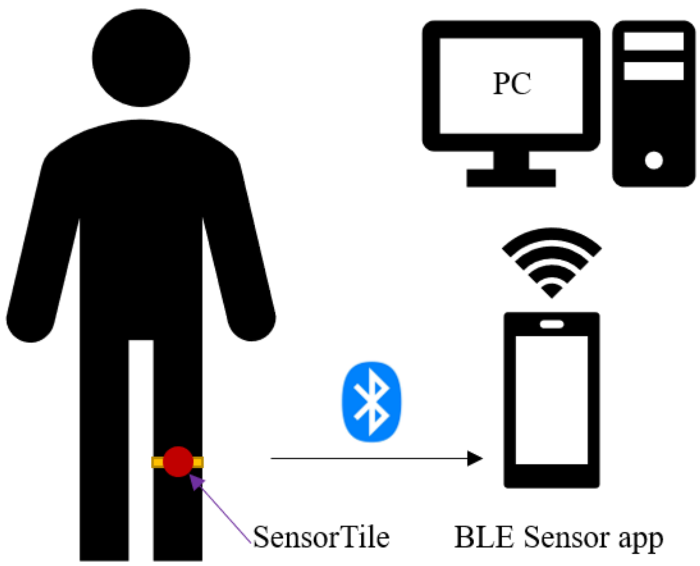


Fig. 3 System architecture

2.2 Data collection, description, and visualization

Three participants with normal knee function volunteered to simulate the dataset. Participants' demographics are shown in Table 1.

Table 1. Participants' demographics

Participant	Gender	Age	BMI	Knee Function
1	Female	30	19.1	Normal
2	Female	32	22.4	Normal
3	Male	30	24.3	Normal

The SensorTile was attached to the participants' knee using a strap band below the anterior horn of the tibia to accurately capture gait motion. Each participant was asked to simulate three distinct gait cycles for ~10 seconds as follows: 1) limping, 2) walking, and 3) running. Each activity was repeated ~10 times to collect a dataset of ~90 datapoints (Fig 4).

Participant 1	10x	10x	10x
Participant 2	10x	10x	10x
Participant 3	10x	10x	10x

Fig.4 Experiment setup

We collected knee rotations using the gyroscope sensor. Figure 5 shows and example of the raw data collected.

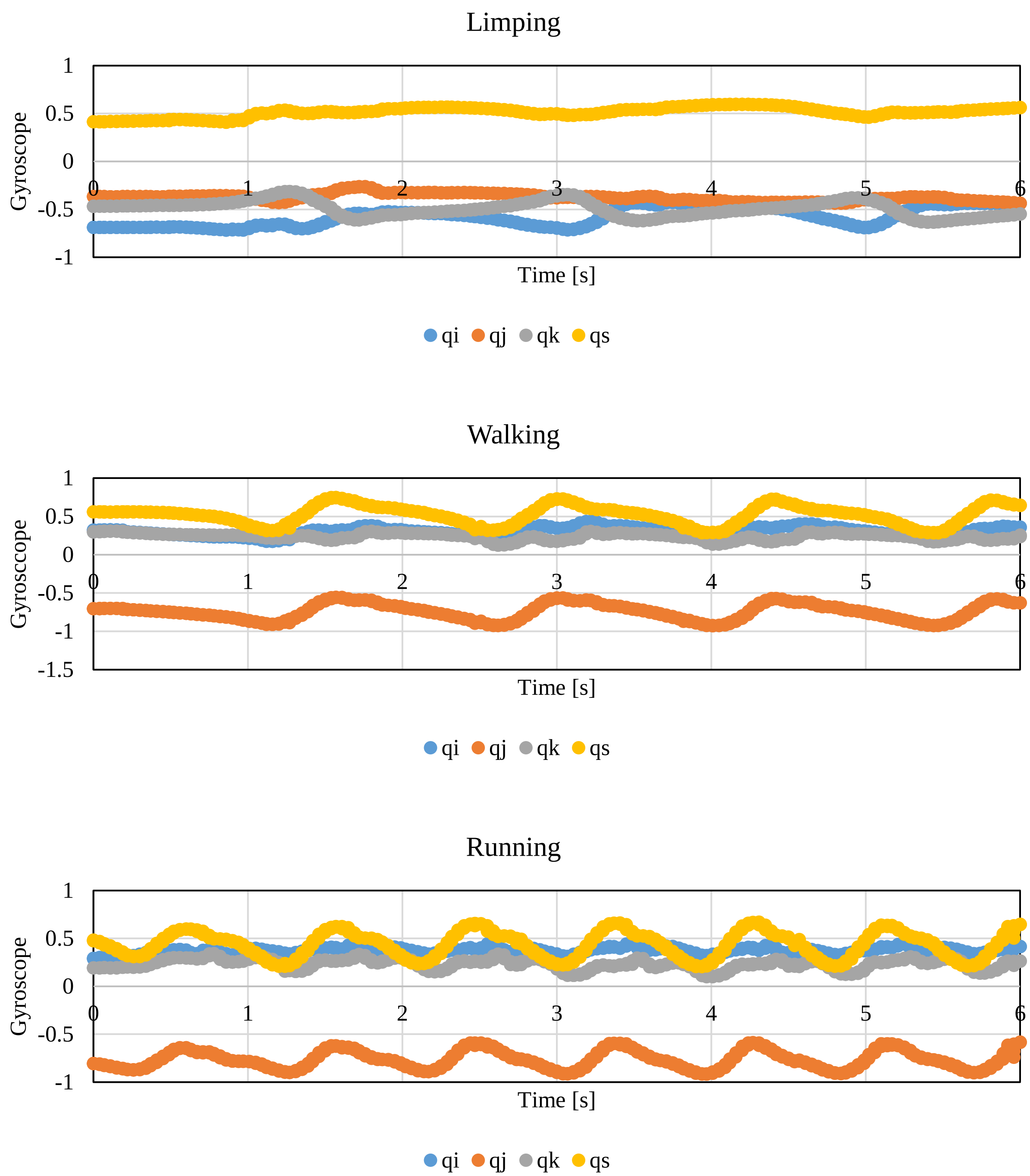


Fig. 5 Examples of three gait types and the gyroscope output for the Participant 1

We considered both “running” and “limping” as “abnormal” gait and “walking as “normal” gait for training the artificial intelligence model.

2.3 Understanding the data with Fast Fourier Transform

We used Fast Fourier Transform (FFT) to understand the data and gait frequency. Figure 6 shows an example of FFT for different type of gaits.

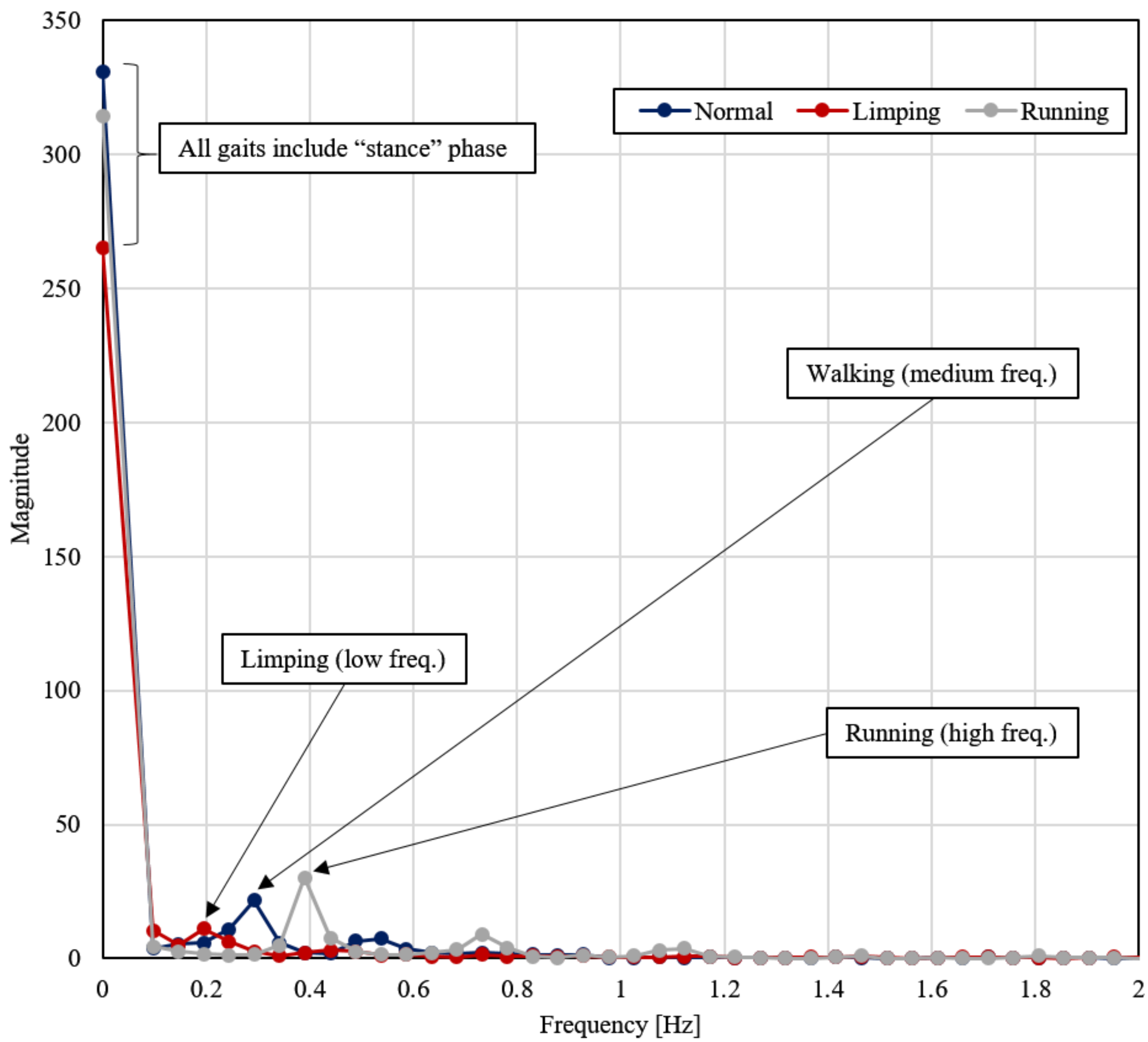


Fig.6 Different gaits Fast Fourier Transform (FFT)

As expected Fig. 6 shows that limping results in a low-frequency movement while normal walking and running results in higher frequency. Furthermore, all gaits include a stance phase.

2.4 Artificial intelligence method

2.4.1 Feature extraction

Upon careful analysis of the data, we observed that different activities can be distinguished by the roll motion around the knee axis (qs). This is in line with the clinical finding since the main indicator of a knee function is the flexion-extension motion around the knee axis and other rotations such as vargus/valgus play a smaller role in diagnosing an injured knee (Fig. 7). Based on this fact we decided to only analyze qs representing the knee flexion-extension motion (Fig. 8). Since we used a deep learning approach to analyze the time-series data no further feature selection was performed.

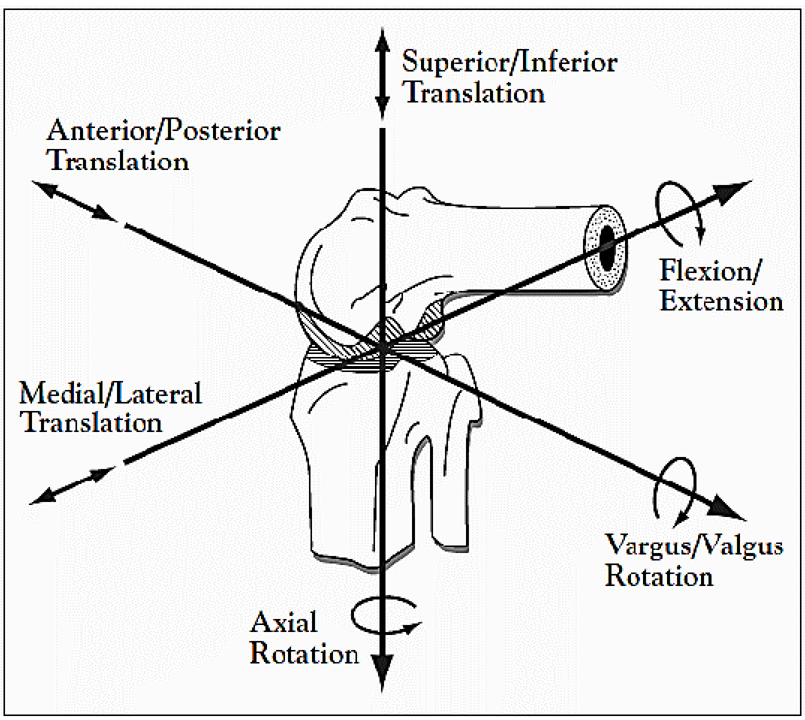


Fig. 7 Knee joint main motions. Source: Fathy and El Messiry, J Textile Sci Eng 2016, 6:2

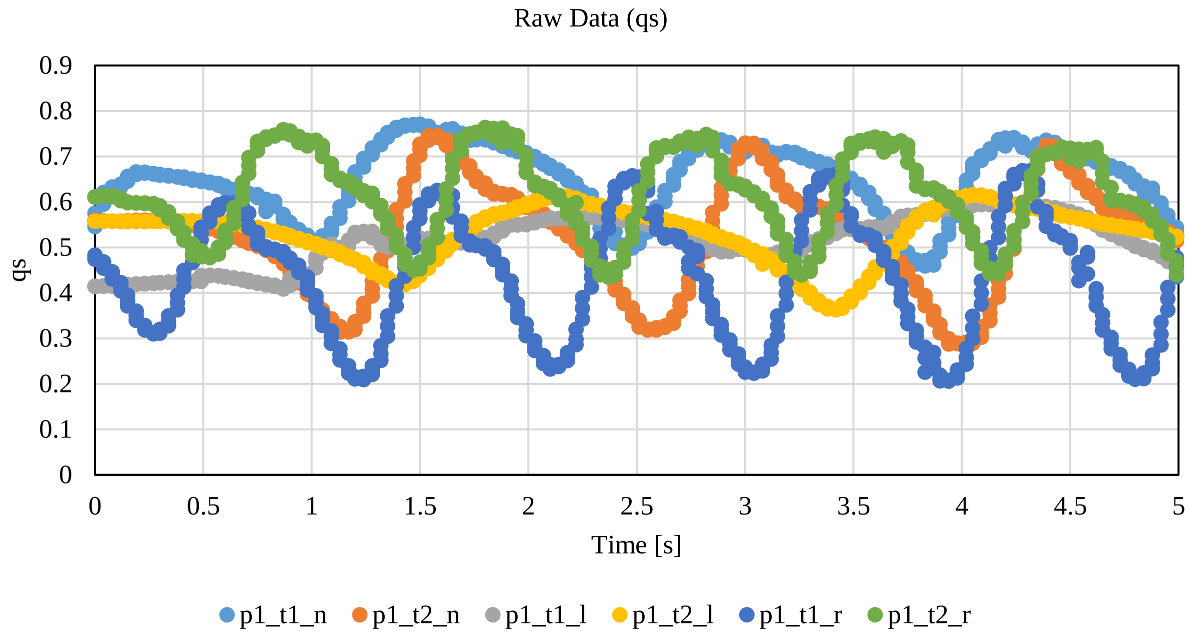


Fig. 8 Raw data (qs). Participant 1 (p1) try 1 (t1), normal (n), limping (l), running (r).

We subsequently normalized the data to focus on the gait pattern and frequency (Fig. 9).

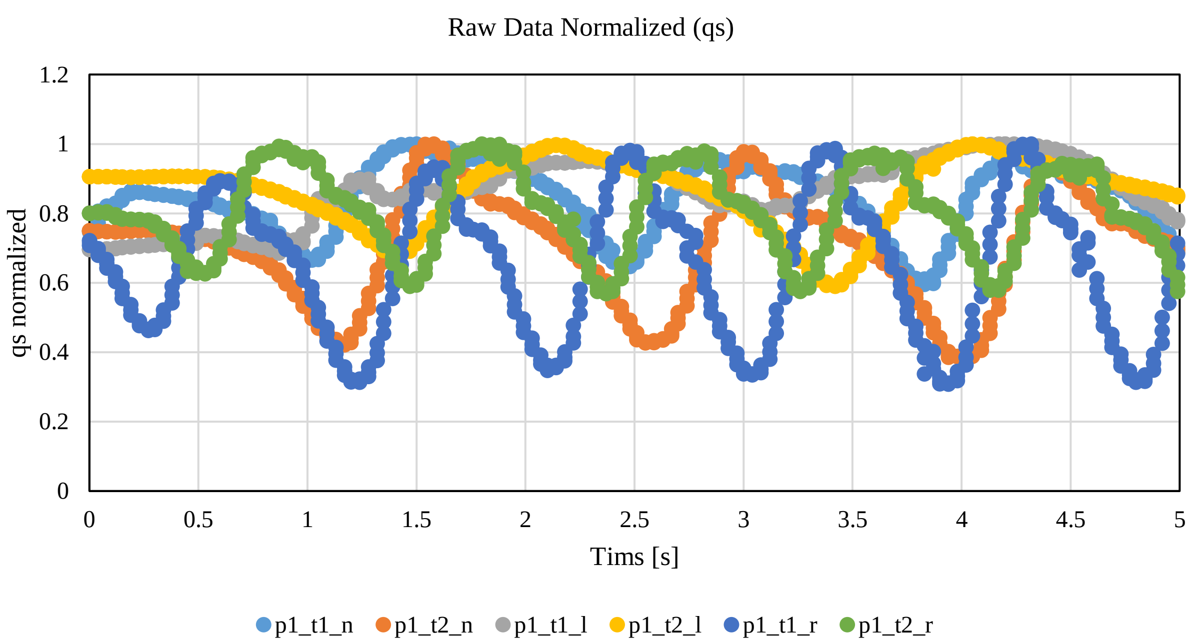


Fig. 9 Raw data normalized (qs). Participant 1 (p1) try 1 (t1), normal (n), limping (l), running (r).

2.4.2 Convolutional neural network

We implemented a 1-D convolutional neural network (CNN) model to analyze the time-series data. This network consisted of 80 filters with 100 kernel size convolving over the time-series data. We added a dropout layer (0.5) to reduce the chance of overfitting. Subsequently, a fully connected layer with 128 nodes with ReLu activation and a final classifier with 2 nodes with sigmoid activation were implemented. We used cross-entropy binary loss function with 'adam' optimizer. (Fig. 10).

```
model = Sequential()
model.add(Conv1D(filters=80, kernel_size=100, activation='relu', input_shape=data_x.shape[1:]))
model.add(Dropout(0.5))

model.add(Flatten())
model.add(Dense(128, activation='relu'))
model.add(Dense(2, activation='sigmoid'))

model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])
```

Fig. 10 1D- convolutional neural network (CNN)

We used 70% of the data for training and 30% of the data for validation using validation_split = 0.3 and trained the model using a batch size of 16 for 6 epochs. (Fig. 11)

```
history = model.fit(data_x, onehot_encoded, epochs=6, batch_size=16,validation_split=0.3,verbose=1,shuffle=False)
```

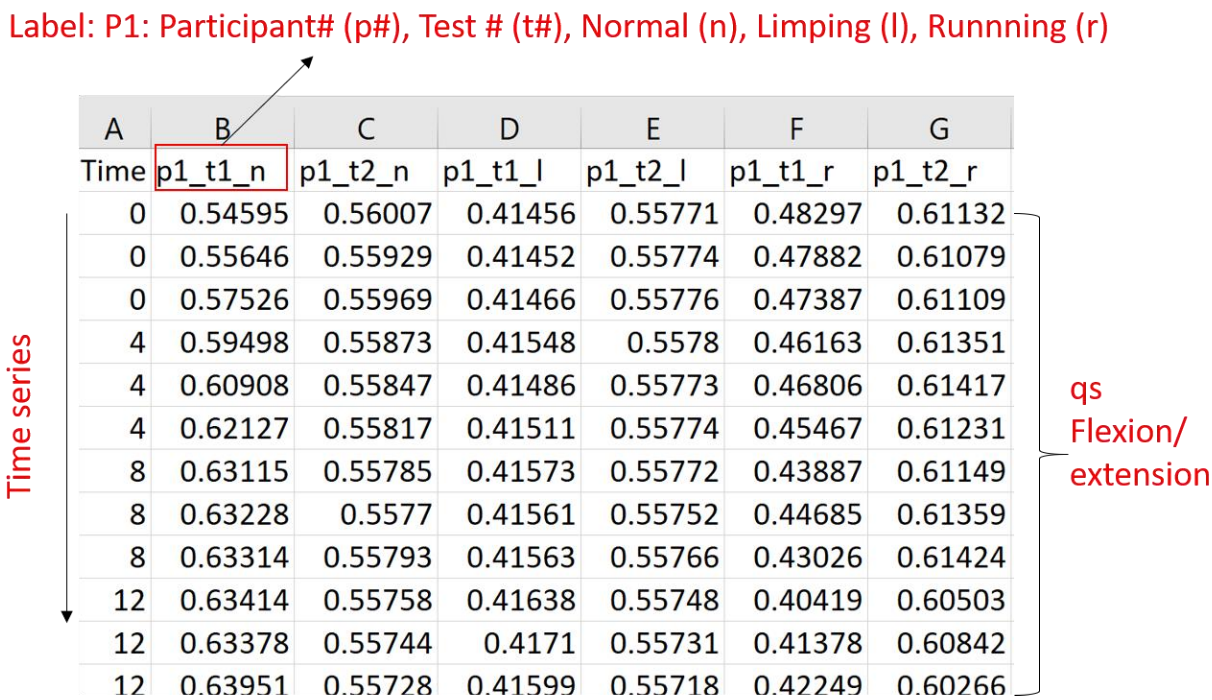
Fig. 11 Training the convolutional neural network (CNN)

2.4.3 Desktop setup

The work was performed on a PC running Microsoft Windows 10 Education, Processor Intel(R) Core(TM) i5-2400 CPU @ 3.10GHz, 3101 Mhz, 4 Core(s), 4 Logical Processors, Installed Physical Memory (RAM) 16.0 GB, with Python 3.6, and Keras 1.6.

3. Results and discussion

We collected 101 data points with almost 50-50% normal vs. abnormal gaits and filtered only the flexion/extension direction gyroscope output (Fig. 12).



A	B	C	D	E	F	G
Time	p1_t1_n	p1_t2_n	p1_t1_l	p1_t2_l	p1_t1_r	p1_t2_r
0	0.54595	0.56007	0.41456	0.55771	0.48297	0.61132
0	0.55646	0.55929	0.41452	0.55774	0.47882	0.61079
0	0.57526	0.55969	0.41466	0.55776	0.47387	0.61109
4	0.59498	0.55873	0.41548	0.5578	0.46163	0.61351
4	0.60908	0.55847	0.41486	0.55773	0.46806	0.61417
4	0.62127	0.55817	0.41511	0.55774	0.45467	0.61231
8	0.63115	0.55785	0.41573	0.55772	0.43887	0.61149
8	0.63228	0.5577	0.41561	0.55752	0.44685	0.61359
8	0.63314	0.55793	0.41563	0.55766	0.43026	0.61424
12	0.63414	0.55758	0.41638	0.55748	0.40419	0.60503
12	0.63378	0.55744	0.4171	0.55731	0.41378	0.60842
12	0.63951	0.55728	0.41599	0.55718	0.42249	0.60266

Fig. 12 Database including 101 data points available online in the github.

The network achieved 80% percent accuracy on the validation subset classifying 25/31 gaits in the validation subset correctly (Fig. 13).

```
In [20]: history = model.fit(data_x, onehot_encoded, epochs=6, batch_size=16,validation_split=0.3,verbose=1,shuffle=False)

Train on 70 samples, validate on 31 samples
Epoch 1/6
70/70 [==============================] - 0s 5ms/step - loss: 2.5542 - accuracy: 0.4357 - val_loss: 0.6205 - val_accuracy: 0.548
4
Epoch 2/6
70/70 [==============================] - 0s 3ms/step - loss: 0.8232 - accuracy: 0.6357 - val_loss: 0.5112 - val_accuracy: 0.806
5
Epoch 3/6
70/70 [==============================] - 0s 4ms/step - loss: 0.7153 - accuracy: 0.5857 - val_loss: 0.5237 - val_accuracy: 0.806
5
Epoch 4/6
70/70 [==============================] - 0s 4ms/step - loss: 0.6357 - accuracy: 0.6143 - val_loss: 0.6052 - val_accuracy: 0.806
5
Epoch 5/6
70/70 [==============================] - 0s 3ms/step - loss: 0.5236 - accuracy: 0.8143 - val_loss: 0.6294 - val_accuracy: 0.790
3
Epoch 6/6
70/70 [==============================] - 0s 4ms/step - loss: 0.4243 - accuracy: 0.9000 - val_loss: 0.5616 - val_accuracy: 0.806
5
```

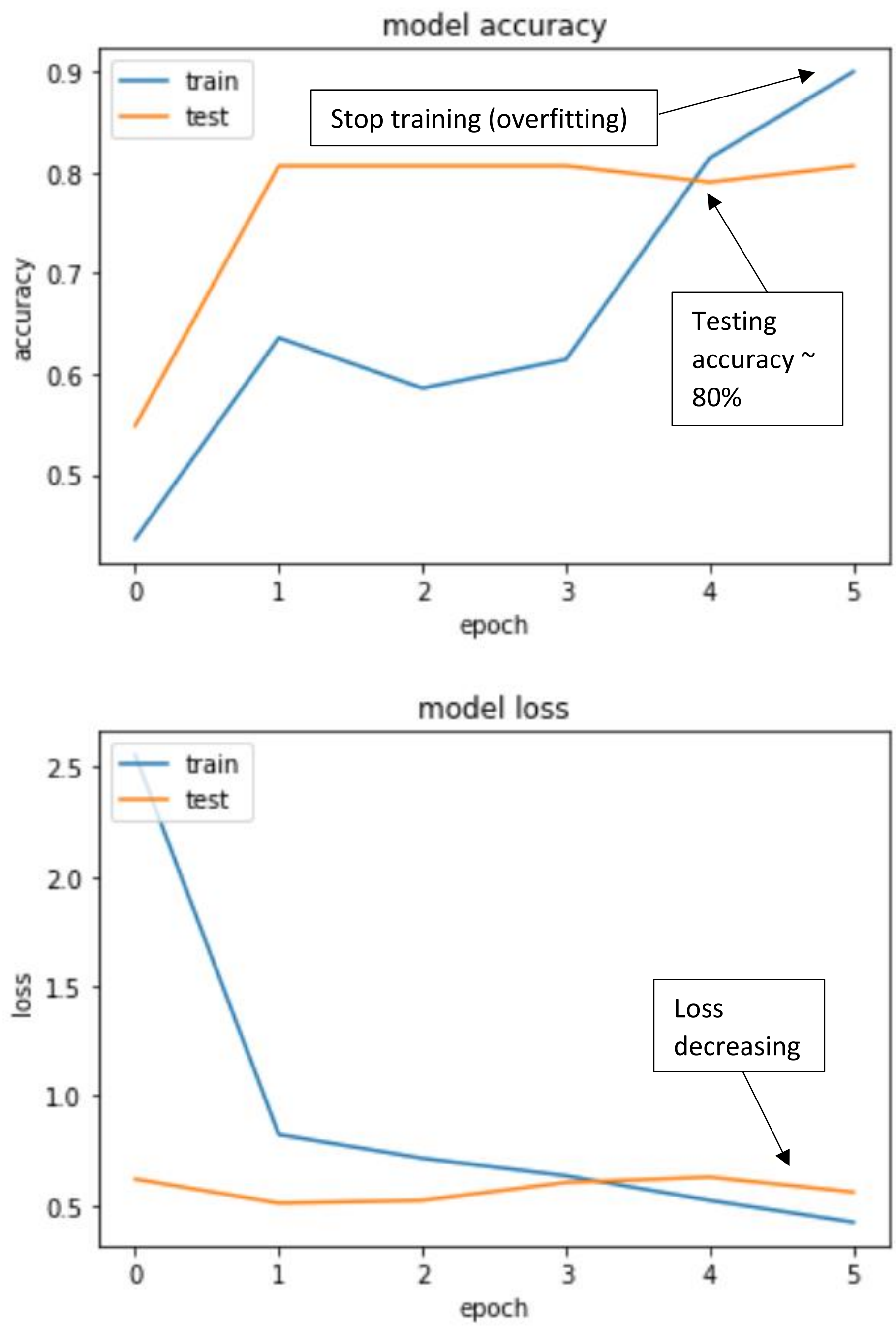


Fig 13. Model performance

4. Demonstration

https://youtu.be/u2KNyBKBTR4

5. Conclusions

We successfully developed a wearable device to capture gait data and an accompanying AI using 1-D CNN network to classify a gait function as "normal" or "abnormal." This device can be implemented in tele-medicine as well as in precision medicine and patient-centric care. This can eventually reduce the need for in-person follow up visits saving patients money and time. It can also result in a more accurate and continues monitoring of the patients post TKA surgery to identify patients with abnormal recovery for more patient-centric care and alternative interventions.

6. Source code

Source code and the raw data can be found here:

https://github.com/Borjali/WearableDevice

7. Future work

In the future, we will collect more data to improve tha network accuracy. Also with more data, we can perform categorical analysis instead of only binary. Furthermore, we will move from using BLE Sensor app to BeagleBone. The SensorTile will connect to the BeagleBone via Wifi and the AI network will run online to calculate the results. We will add a small LCD monitor to show the results eliminating the need for a PC. With more data, we can also try other more advanced deep learning methods such as LSTM, and CNN-LSTM.

8. References

1. Hassan, M. M., Uddin, M. Z., Mohamed, A. & Almogren, A. A robust human activity recognition system using smartphone sensors and deep learning. *Futur. Gener. Comput. Syst.* **81**, 307–313 (2018).
2. Ronao, C. A. & Cho, S. B. Human activity recognition with smartphone sensors using deep learning neural networks. *Expert Syst. Appl.* **59**, 235–244 (2016).
3. Chen, Y. & Xue, Y. A Deep Learning Approach to Human Activity Recognition Based on Single Accelerometer. in *Proceedings - 2015 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2015* 1488–1492 (Institute of Electrical and Electronics Engineers Inc., 2016). doi:10.1109/SMC.2015.263