THE ANDREW & ERNA VITERBI FACULTY OF ELECTRICAL ENGINEERING

SIPL Signal and Image Processing Lab.

TECHNION Israel Institute of Technology

Final Presentation

User Interface Design for Low Cost 3D Printed Prosthetic Hand

Students: Yahel Solomon, Marina Mearkovich

Supervisor: Shunit Polinsky

Semester: Winter, 2020

Date: 08/09/2020

In collaboration with: Haifa 3D

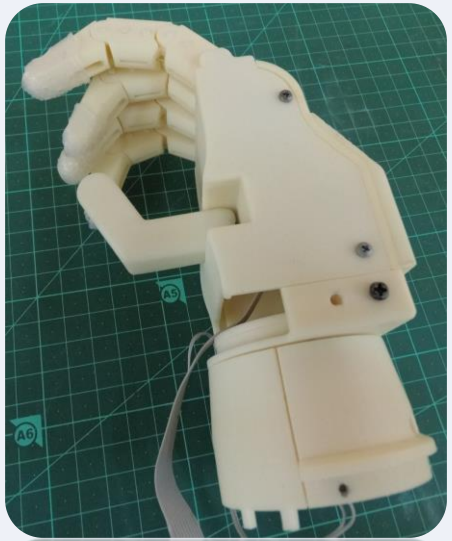
Outline

- Project goal
- Background
- Existing solutions
- Chosen Solution & Implementation
- Results
- Conclusions
- Future work

Project Goal

- Control of 3D printed electro-mechanical prosthetic hand through signals acquired from the leg.
  - Setting up a sensor system
  - Collecting data
  - Classification of at least three gestures

The prosthetic hand in use



Background

- Aspects impaired by hand loss:
  - Ability to perform daily life activities
  - Work
  - Social activities
- Drawbacks of current commercial solutions
  - Expensive
  - Not intuitive & long training
  - Required the residual limb muscle activation

➔Thus a need for a practical, reliable & intuitive hand prosthesis arises.

Needs of Prosthesis users

- Reliability of detection – low probability of unintended movements
- Intuitive control:
  - Short training period
  - Controlling grip force
  - Feedback
- Light weight

Existing Prosthetic Arms

Active

Body Powered

- Requires a lot of power
- Open and close gripper

Electro Mechanic: Most common– Surface EMG

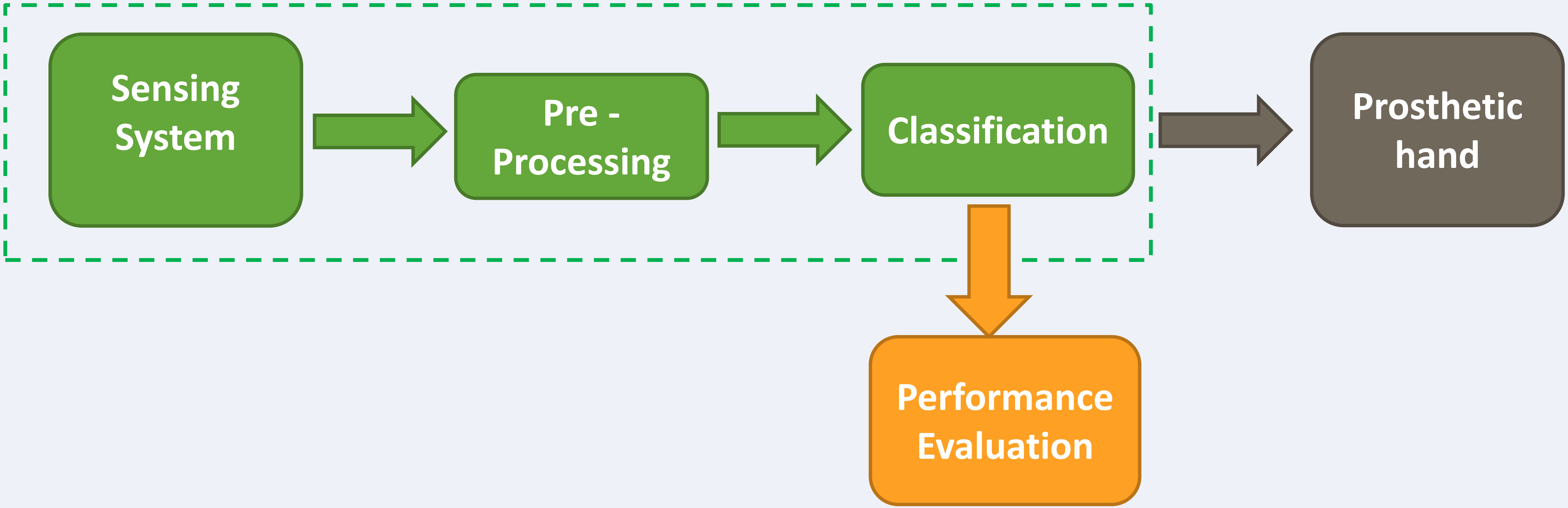
- Lots of research
- Noisy signal , unstable control.
- Phantom pain

Passive

- Esthetic
- Functional

Chosen Solution

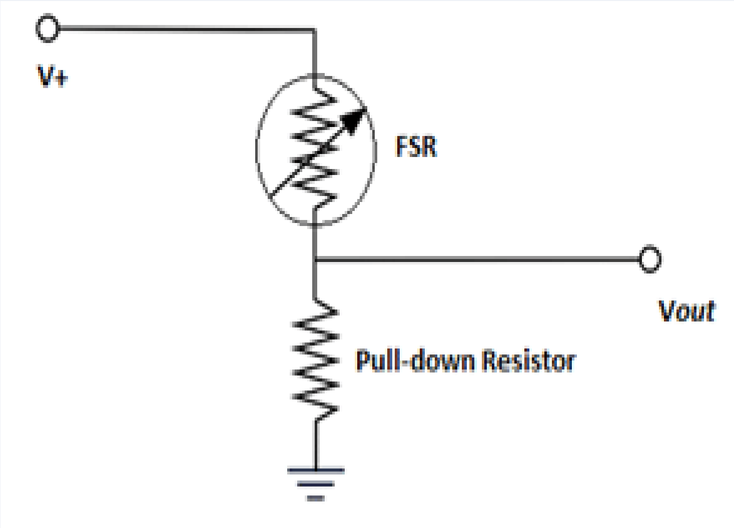
General Block Diagram

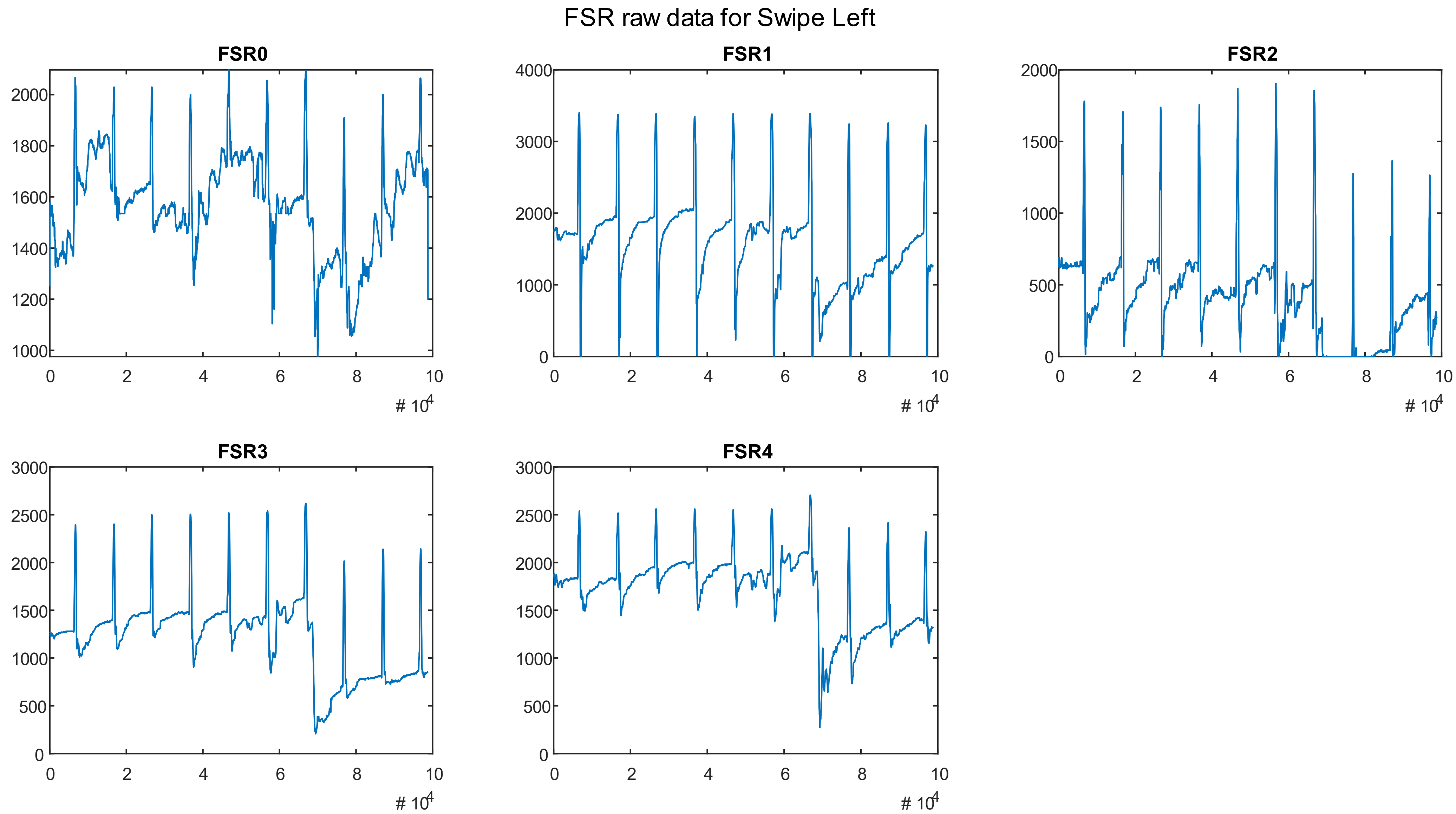


Sensor System

- FSR – force sensitive resistors

  Consists of two copper electrodes that contact a sheet of conductive polymer that decreases its resistance when pressure is applied.




FSR raw data for Swipe Left
FSR0
FSR1
FSR2
FSR3
FSR4
10^4

Sensor System

- BNO055 - IMU

  Consists of three sensor units- accelerometer, gyroscope and magnetometer.

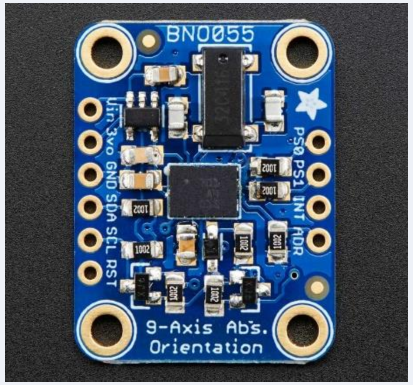
  <u>Measurements used:</u>

  – **Angular velocity:** Three axis (x,y,z) angular velocity data (gyro) in rad/s.

- ESP32 – controller board

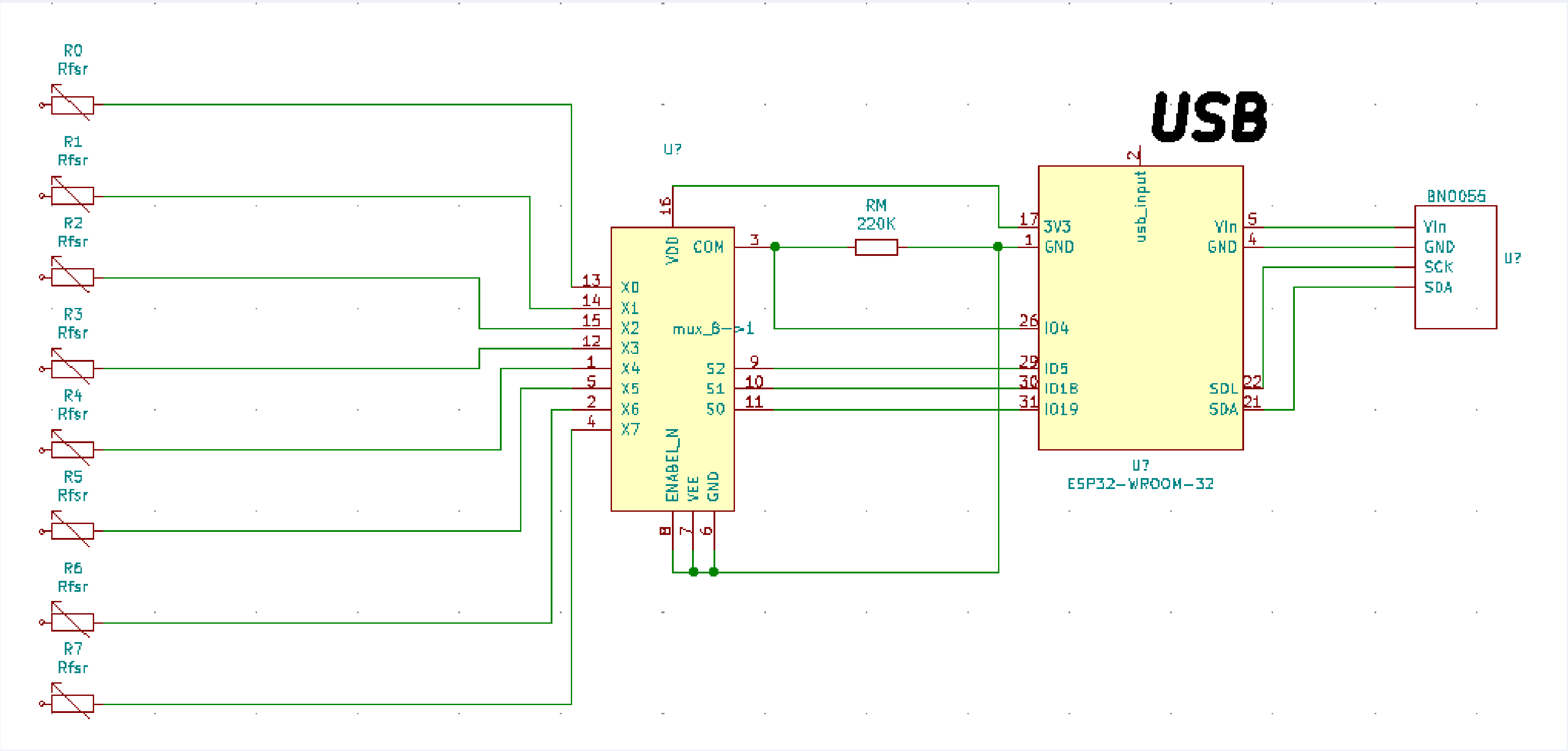
  – Receives input from IMU using I2C communication.

  – Arduino code written to receive Data at 50[Hz] sample rate.

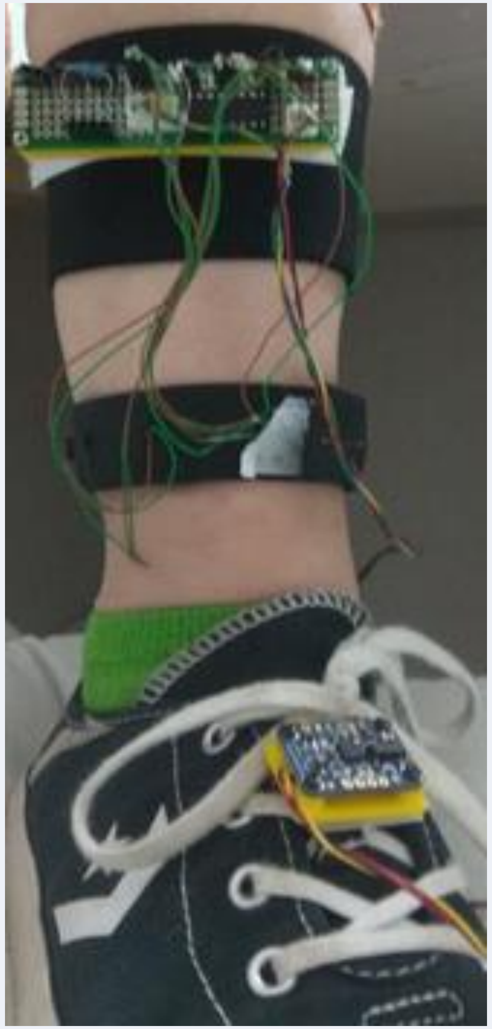


Adafruit©

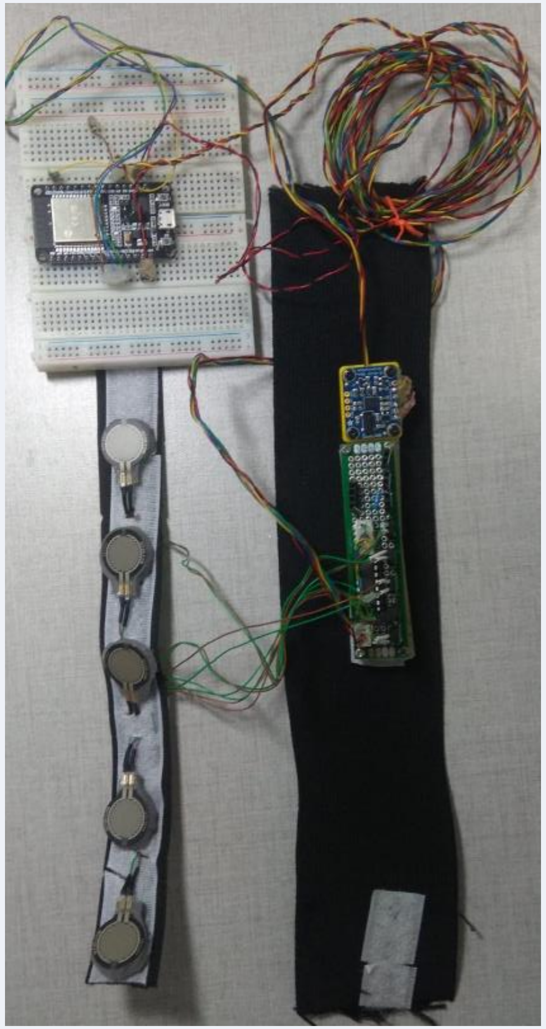
Electrical diagram



System for acquiring data from the leg:



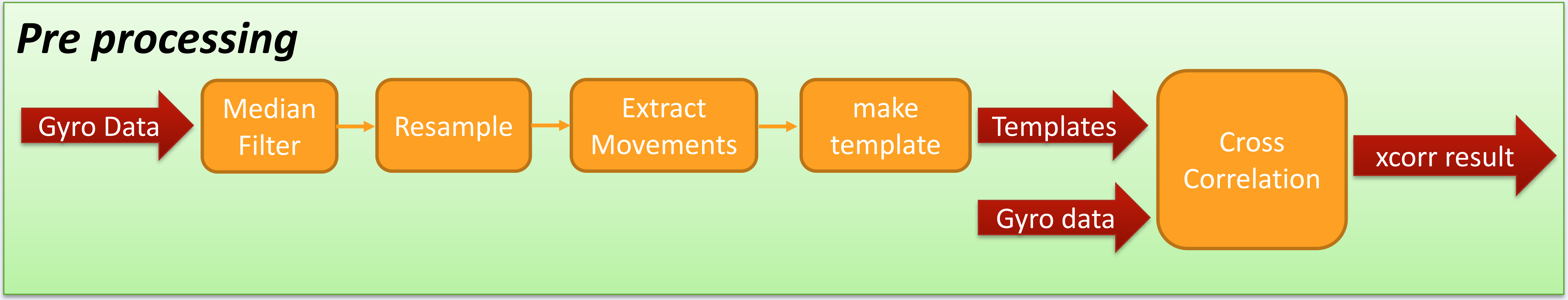
System on leg



System layout

Pre-Processing

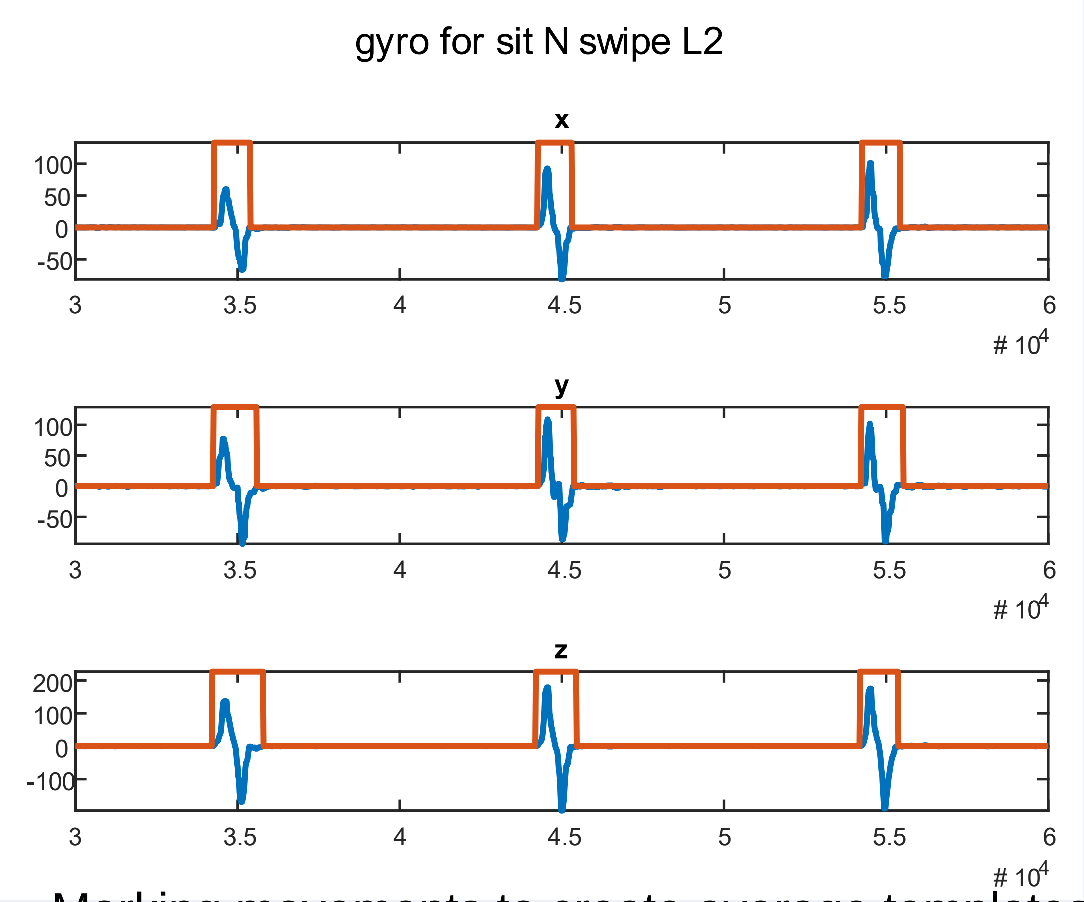
- Block diagram:



  - A series of measurements was conducted for 4 movements: Swipe Left , Swipe Right , Tap and Side Ankle flexion.
  - Raw data was filtered using a median filter and resampled to a constant sample rate.

Movement extraction

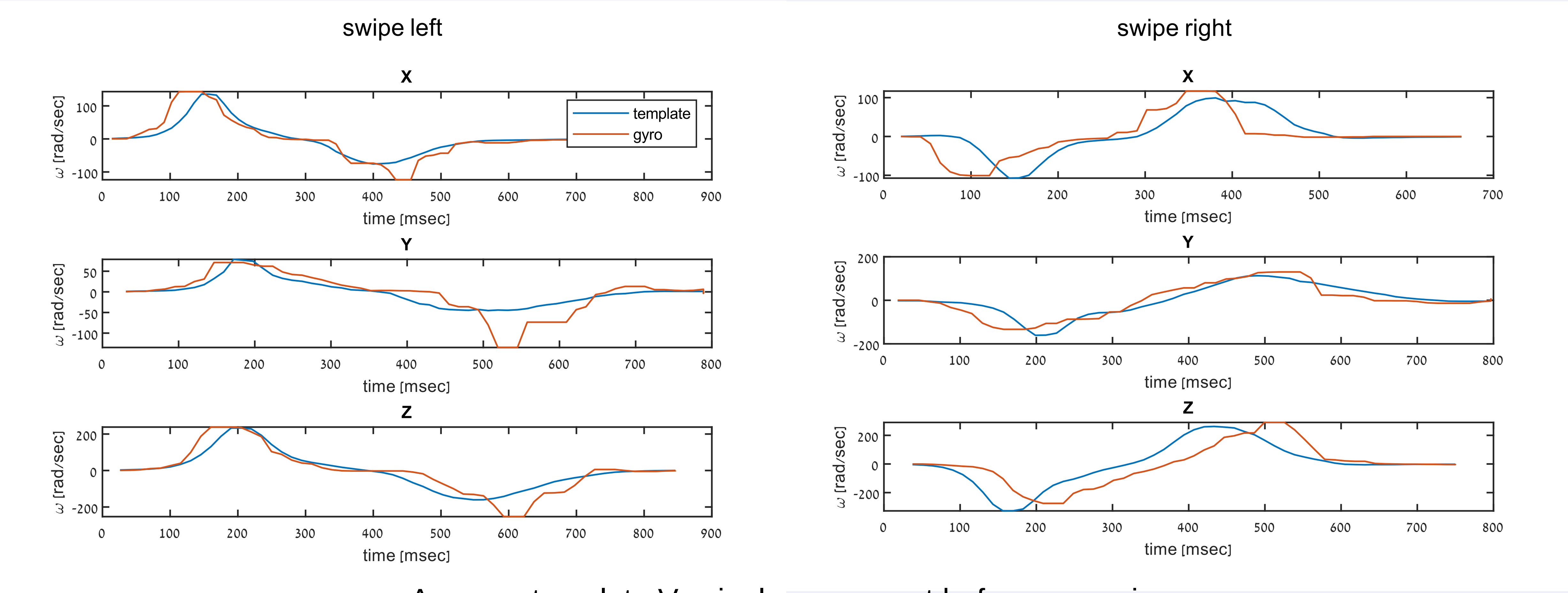
- Template matching method was chosen for classification due to the pattern exhibited by the gyro measurements.
- Movements were extracted to create average templates of each movement at each axis.



Marking movements to create average templates

Creating templates

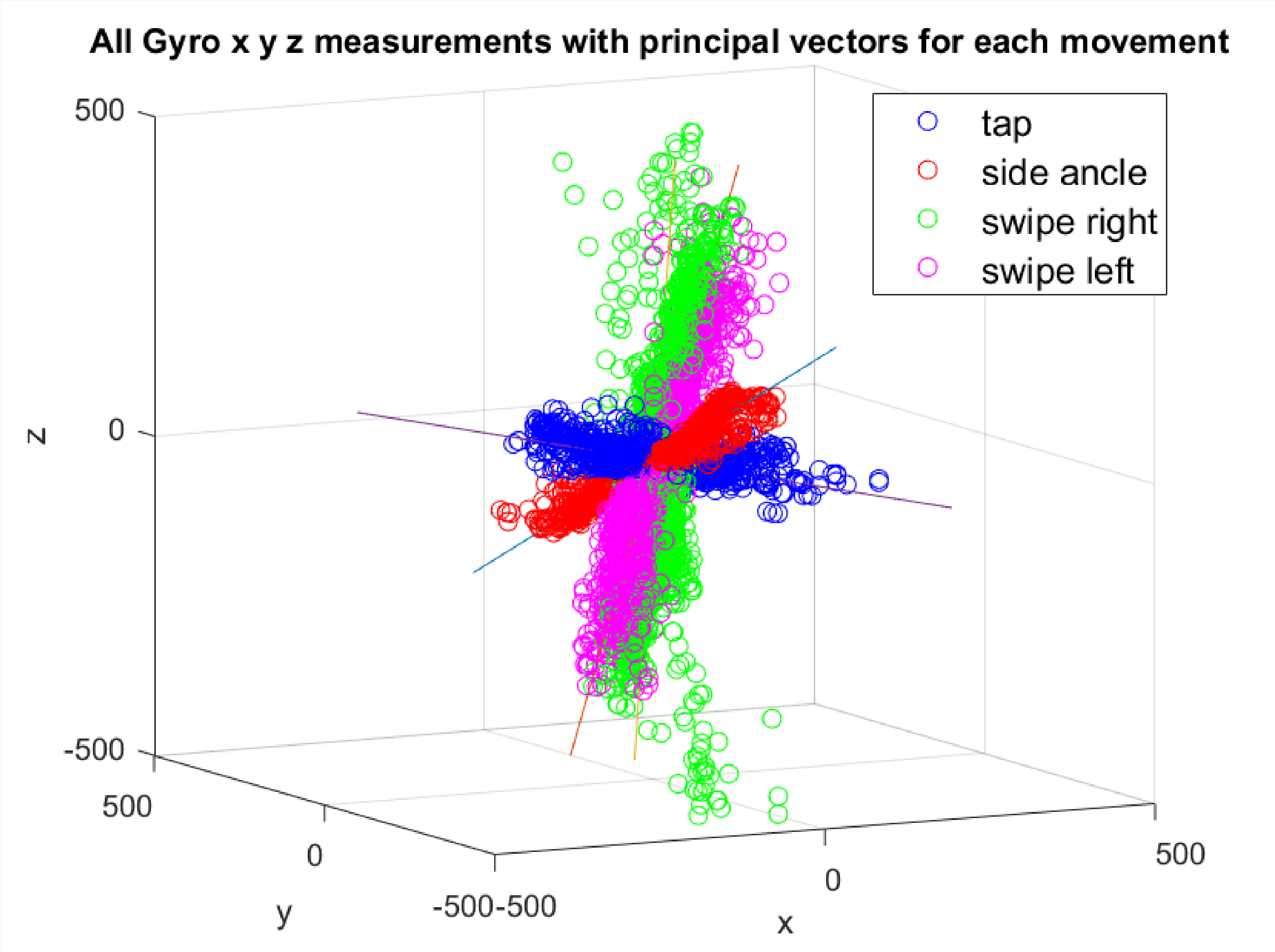
- Movements were aligned and clipped to create average templates of each axis for each movement:



Average template Vs single movement before averaging

Movement properties

- We chose 4 foot movements for classification : Swipe Left , Swipe Right, Tap, Foot Inversion (side ankle)
- PCA analysis was preformed to the measured 3 axis gyro data, to find principal vectors for all 4 movements.
- This shows the Movements are distinct by their principal vector direction.



Template matching

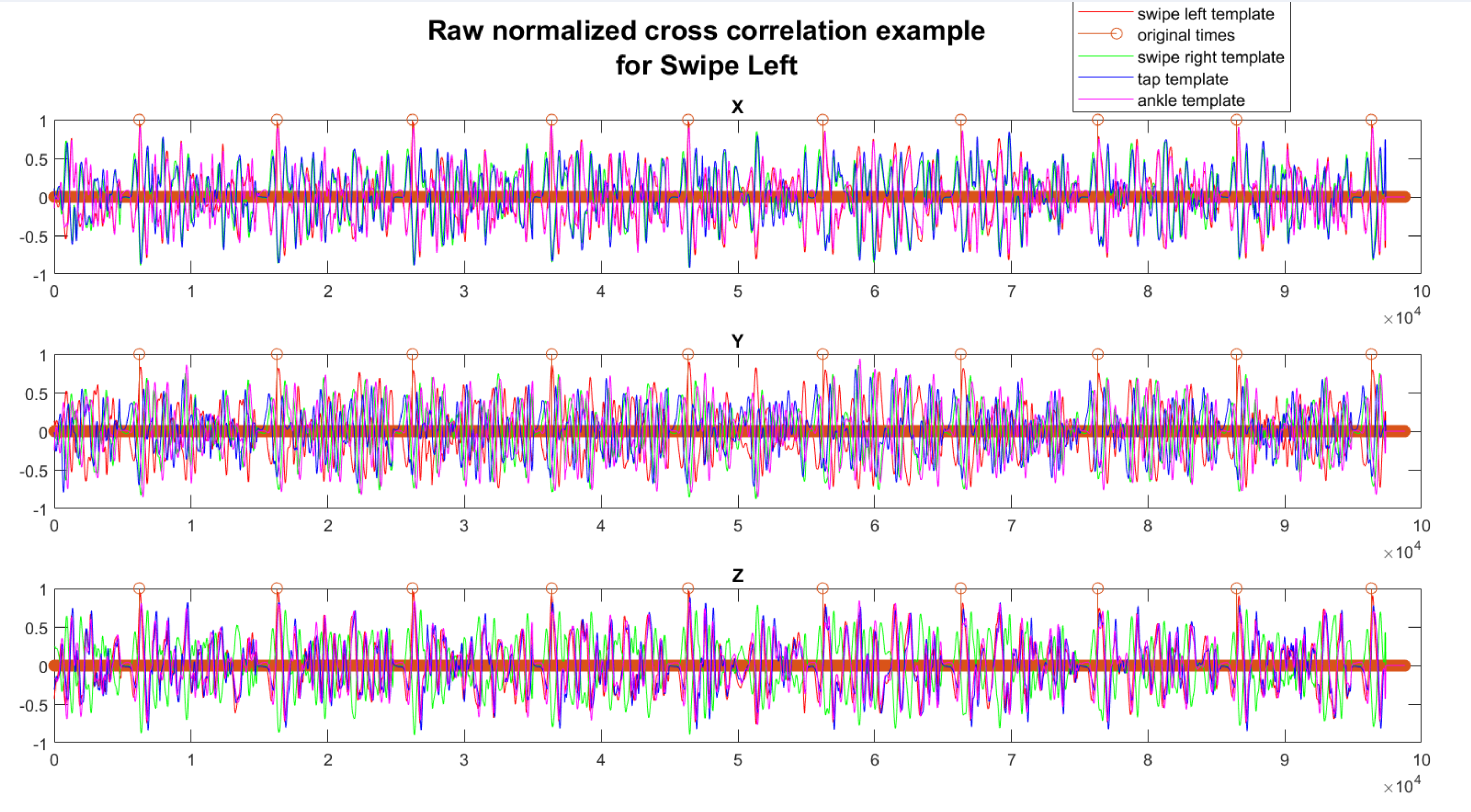
- Cross - correlation between the input data and each of the 4 movement templates :

$$R_{xy,normalized}(0) = \frac{1}{\sqrt{R_{xx}(0)R_{yy}(0)}} \sum_{n=0}^{N-1} x_n y_n$$

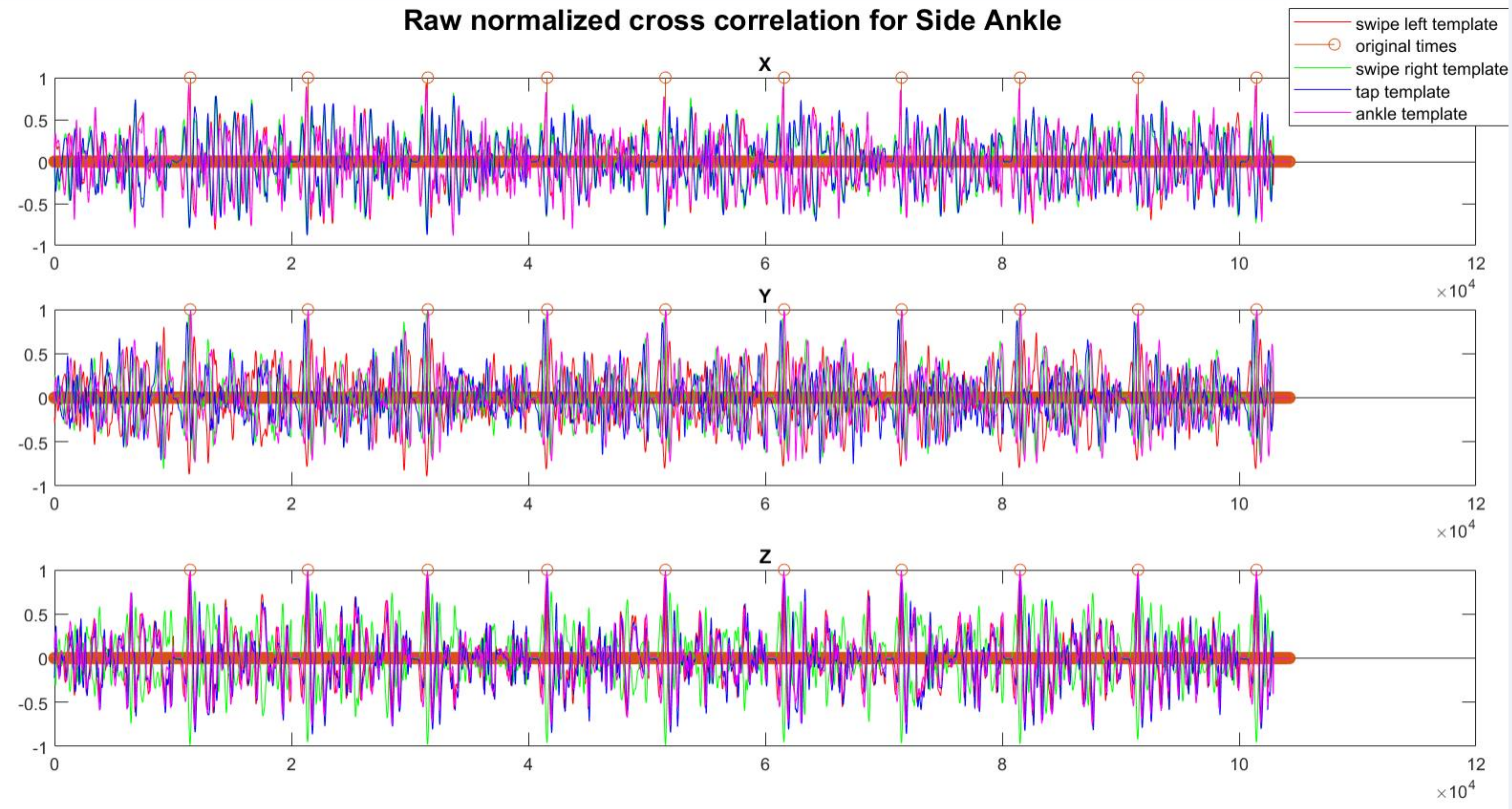
  - $y_n - template, R_{yy}(0) = template\ energy$
  - $x_n - input\ data,\ R_{xx}(0) = data\ energy$
  - $N - template\ len$

- We repeat the calculation above for N-length windows each time delayed by one sample

Cross correlation results

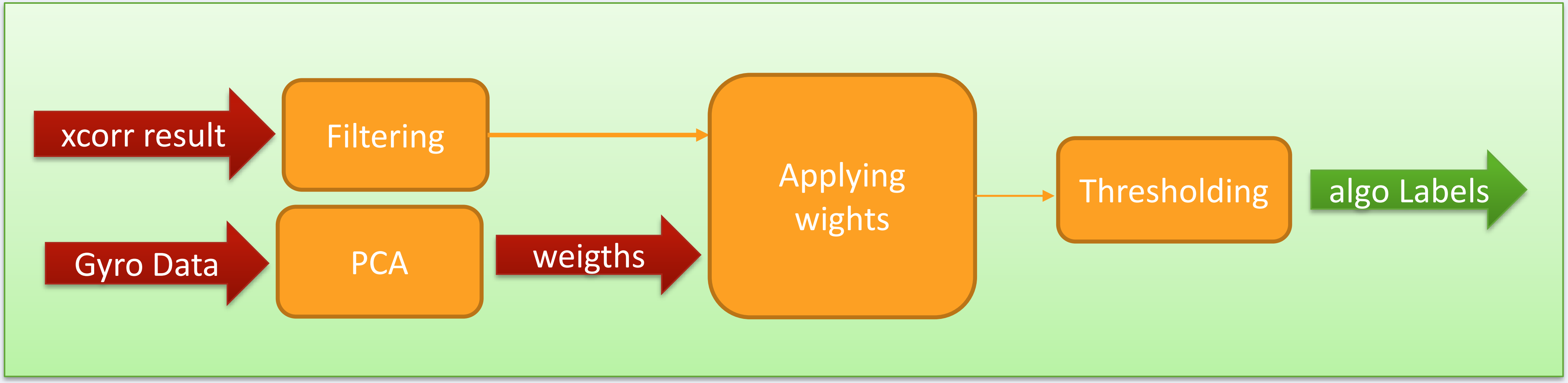


Cross correlation results



Classification

- The main classification method – Template matching
- Block Diagram:

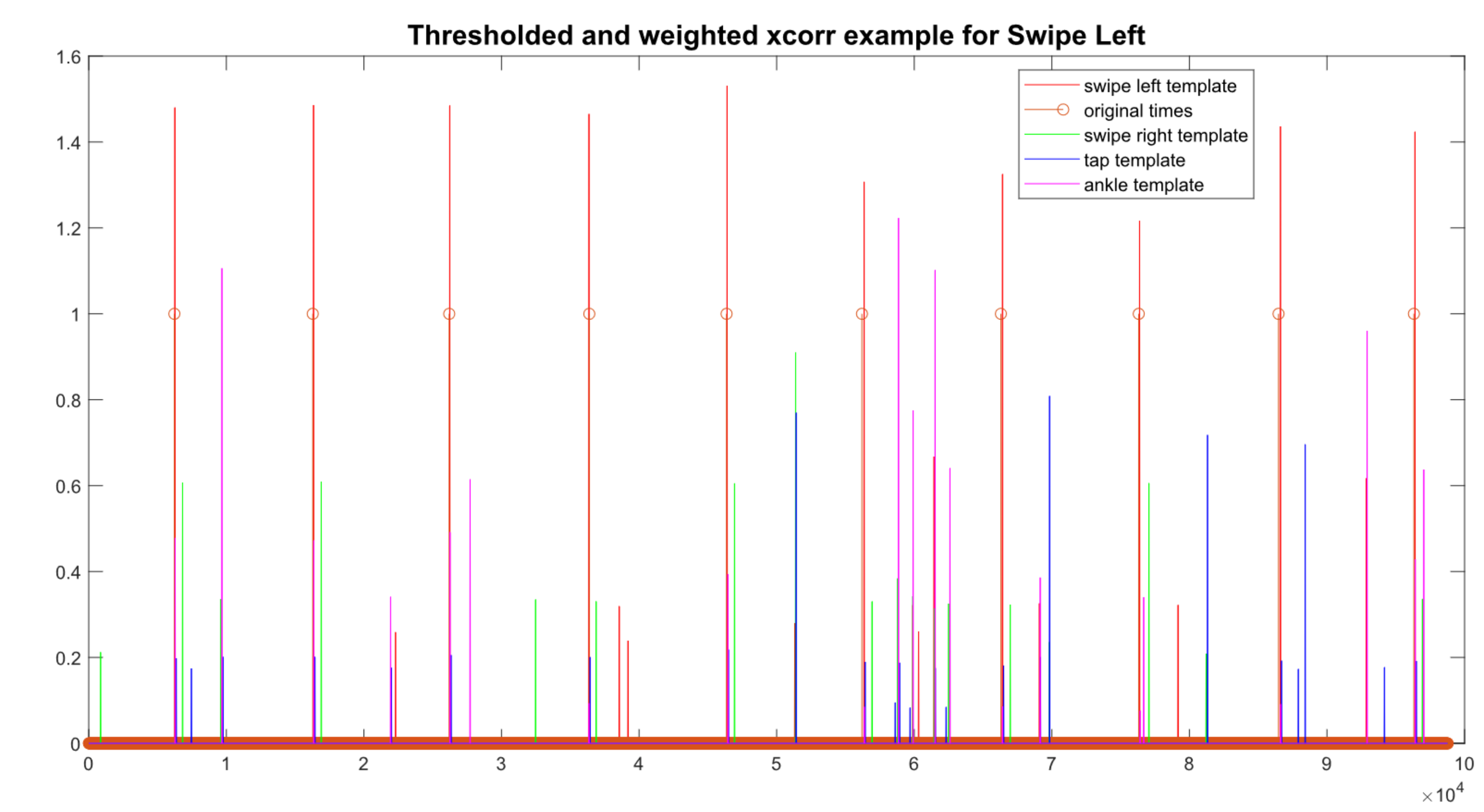


- Thresholds were applied to filter out noise from the cross-correlation results.
- PCA weights were used to combine 3 axis of cross correlation to one axis.

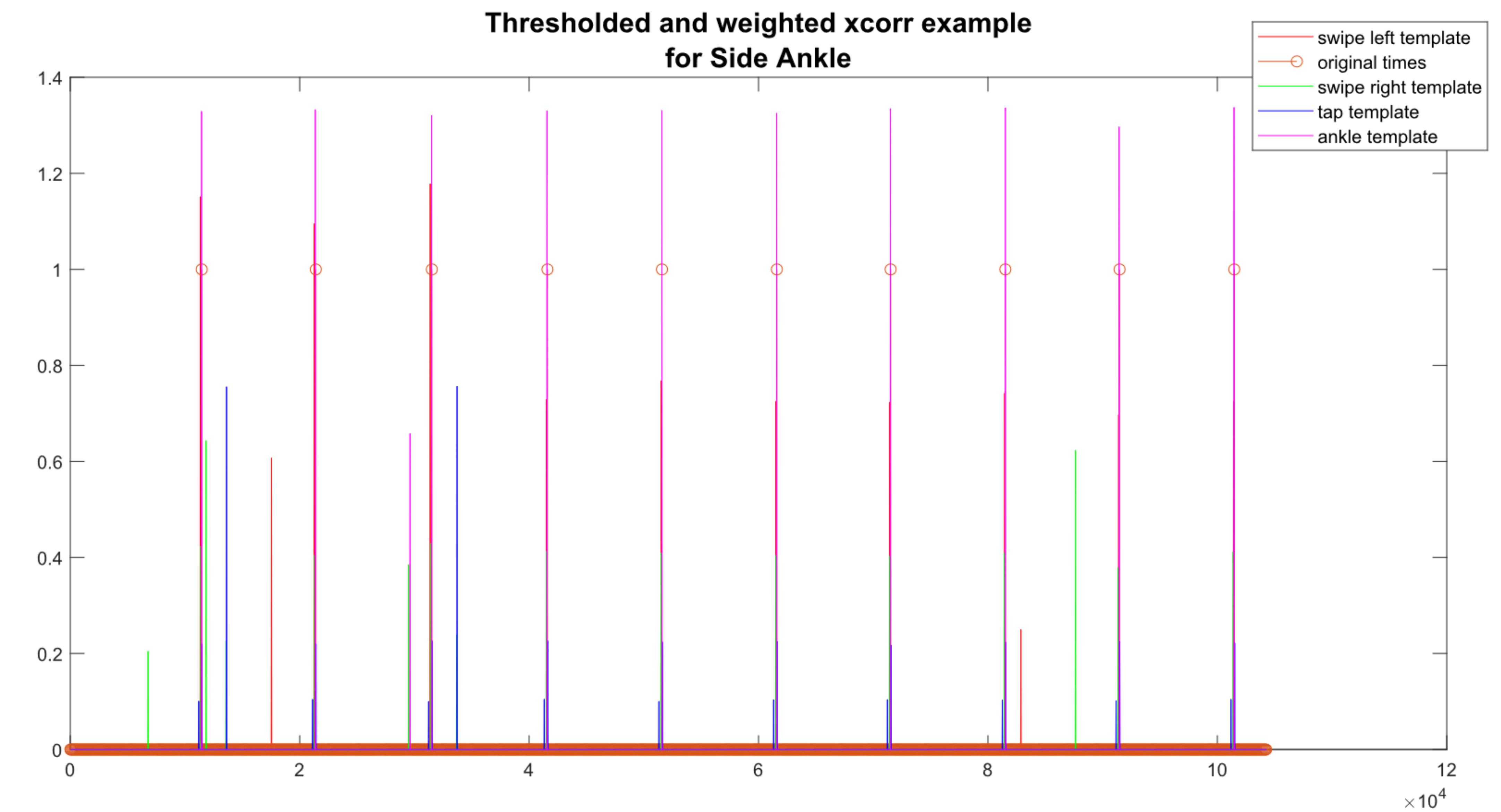
Filtering Cross - Correlation

- Two Filtering methods were used to filter cross correlation results:
  - Filtering peaks lower than selected threshold – th1.
  - Filtering peaks that appear above th1 less than a selected time threshold – th2.
- Use PCA principle vectors as weights to each 3-axis cross correlation result.
  - Combines 3 axis cross – correlation to one result
  - Make results more distinguishable between different movement types.

Filtering Cross - Correlation



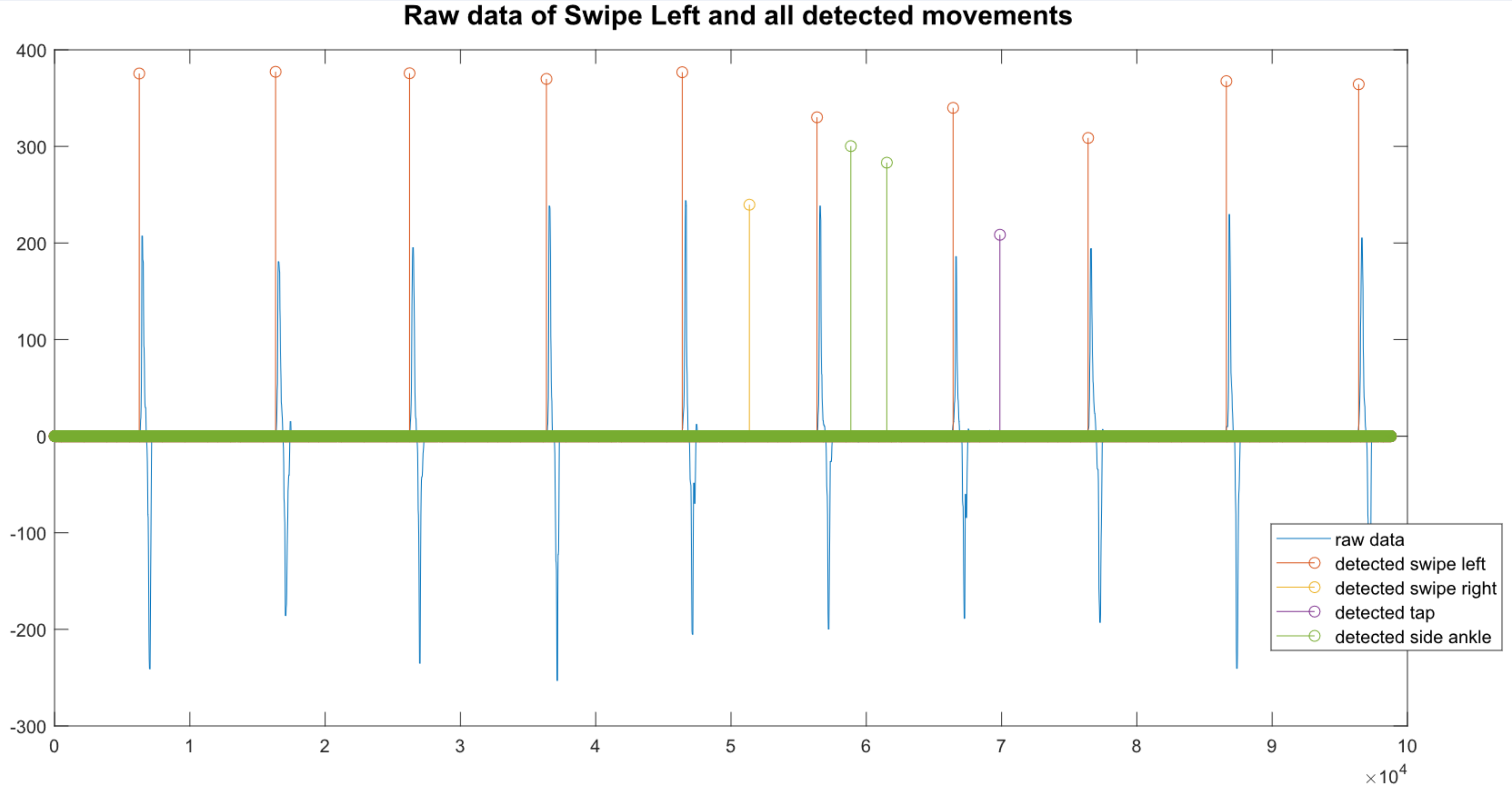
Filtering Cross - Correlation



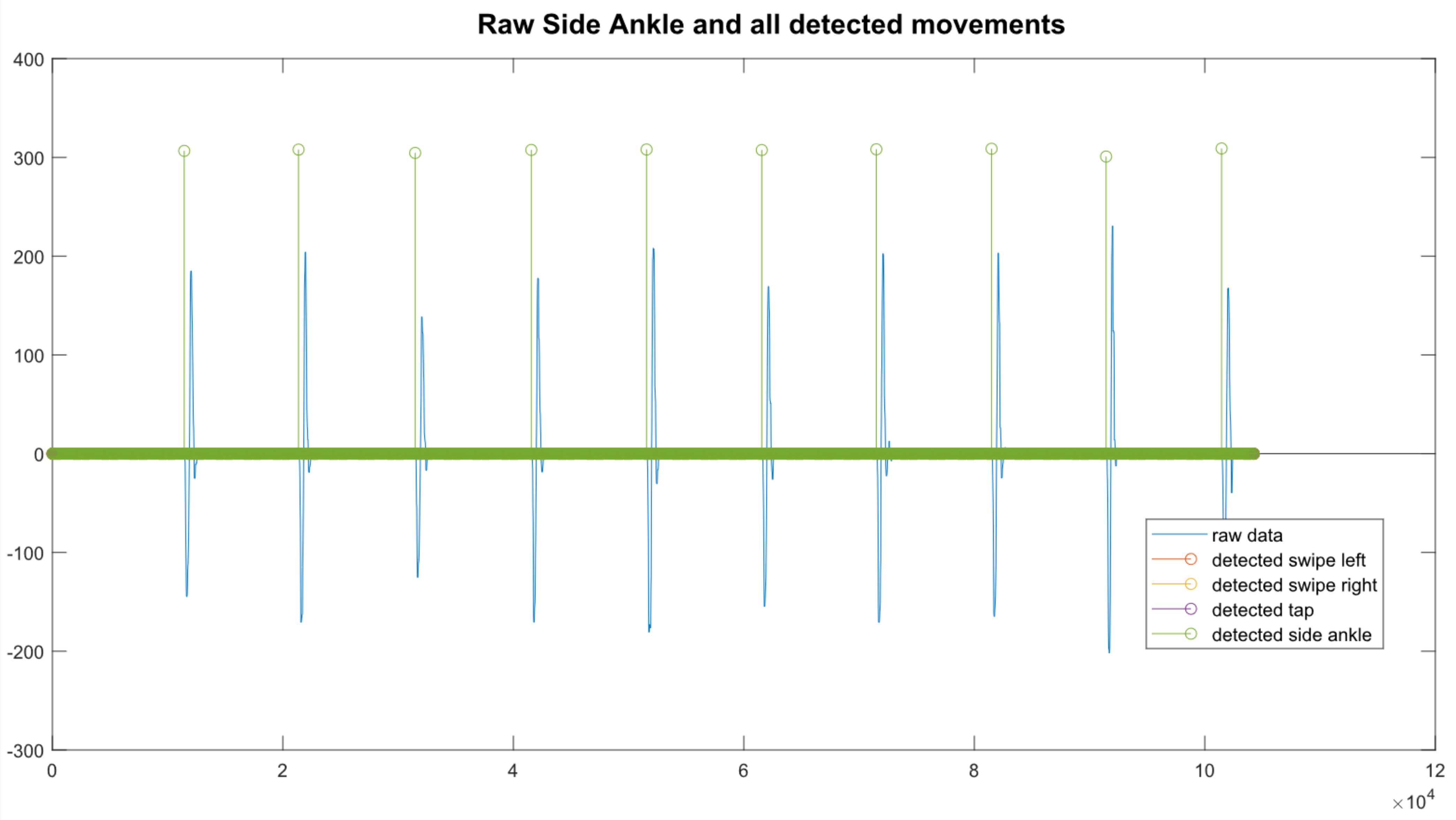
Classification Results

- Providing Binary output label to each timestep in the input vector:
  - Applying thresholding to weighted data from previous step
  - Picking the class of each timestep by choosing the maximal result from the previous step.
  - Comparing the current timestep result with results from a predetermined number of previous time steps:
    - If there is a larger result - zero the current label
    - If the current result is largest among the previous results - zero the previous results.

Classification Results

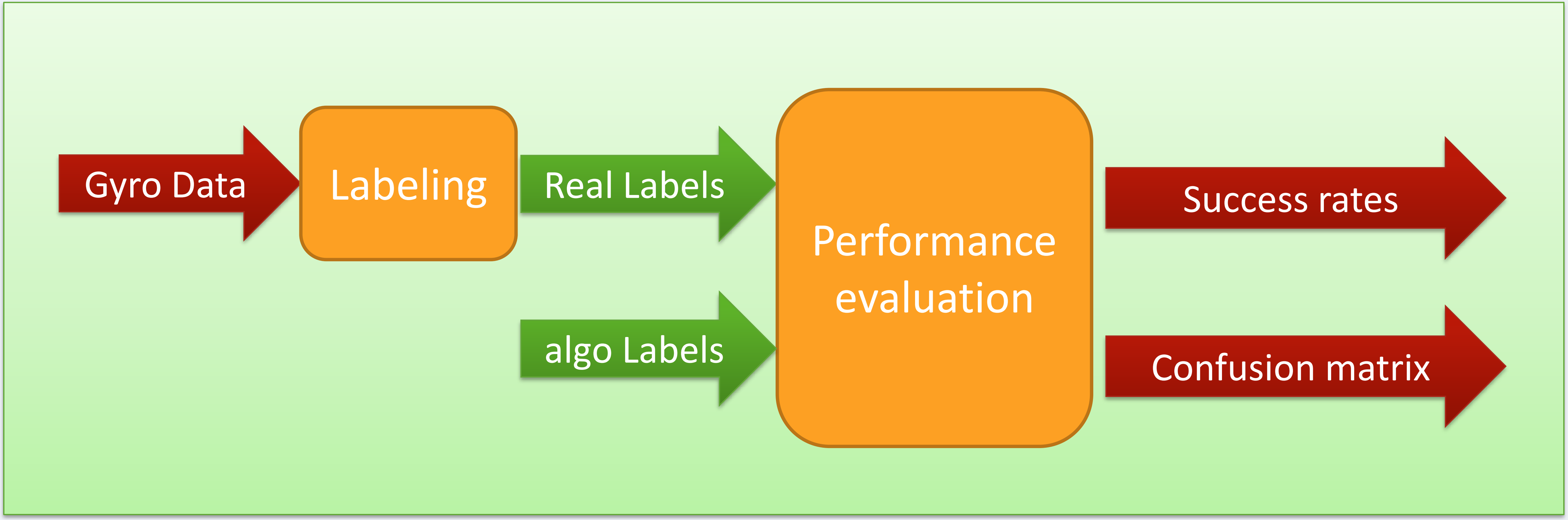


Classification Results



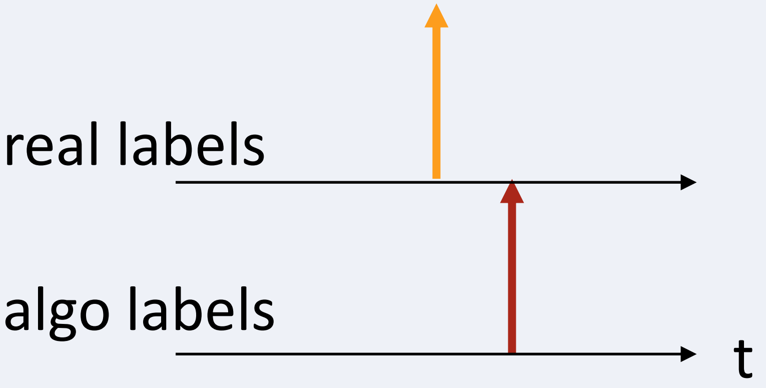
Performance evaluation

- Block Diagram:

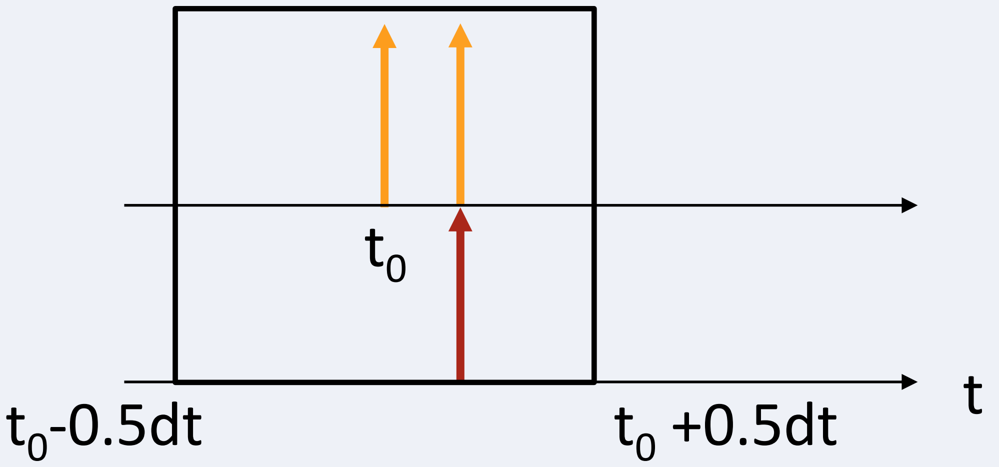


Confusion matrix

Problem:
Delay between labels



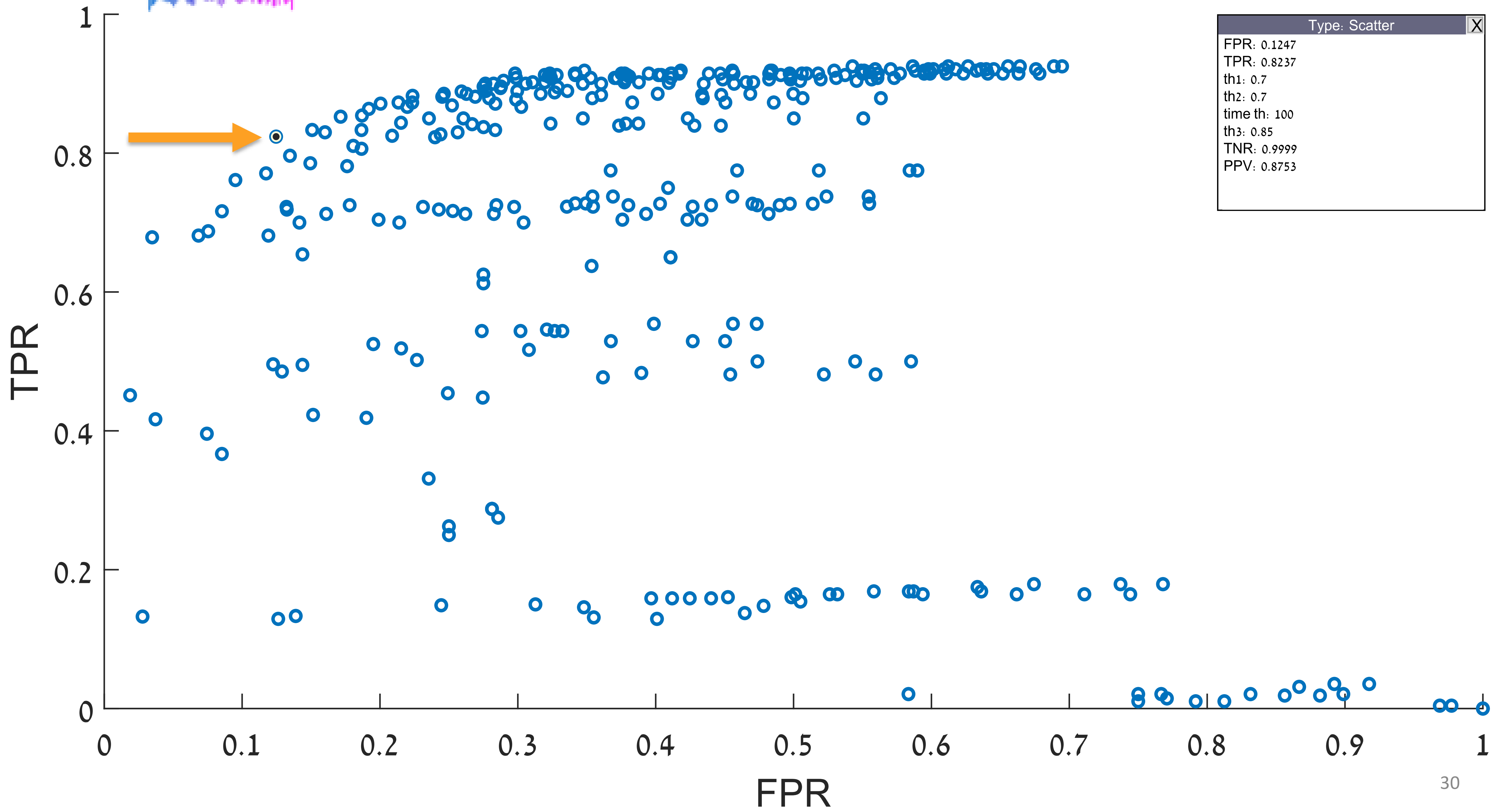
Solution:
Moving real label to be at same timestep as algorithm label



Grid Search

- **Reducing unvoluntary hand movements** (FP) was considered the most important performance aspect, followed by **sensitivity** (TP).
  - Choosing optimal threshold values was based primarily on FPR, and TPR.
- The grid search loop provided the algorithm parameters that met two conditions:
  - FPR was less than 0.15
  - Maximal TPR

ROC scatter


Type: Scatter
X
FPR: 0.1247
TPR: 0.8237
th1: 0.7
th2: 0.7
time th: 100
th3: 0.85
TNR: 0.9999
PPV: 0.8753
TPR
FPR
0
0.2
0.4
0.6
0.8
1
0.1
0.3
0.5
0.7
0.9

SIPL

Example 1

Xcorr Th = 0.7, Time Th = 100, Th3 = 0.85

Movement classification confusion matrix

True Class \ Predicted Class	no movement	side ankle	swipe left	swipe right	tap		
no movement	99569	7	6	7	2	100.0%	0.0%
side ankle		40				100.0%	
swipe left	10		50			83.3%	16.7%
swipe right	5			35		87.5%	12.5%
tap	15	1			24	60.0%	40.0%
	100.0%	83.3%	89.3%	83.3%	92.3%		
	0.0%	16.7%	10.7%	16.7%	7.7%		

TPR

FPR

FP

FN

SIPL

Example 2:
Xcorr Th = 0.5, Time Th = 60, Th3 = 0.9

Movement classification confusion matrix

True Class \ Predicted Class	no movement	side ankle	swipe left	swipe right	tap		
no movement	99466	78	22	23	2	99.9%	0.1%
side ankle		40				100.0%	
swipe left	3		57			95.0%	5.0%
swipe right				40		100.0%	
tap	2	9			29	72.5%	27.5%
						TPR	
FP	100.0%	31.5%	72.2%	63.5%	93.5%		
FN	0.0%	68.5%	27.8%	36.5%	6.5%	FPR	

SIPL

FPR

Ex.1	16.7%	10.7%	16.7%	7.7%
Ex.2	68.5%	27.8%	36.5%	6.5%

TPR

Ex.1	Ex.2
100.0%	100.0%
83.3%	95.0%
87.5%	100.0%
60.0%	72.5%

Test Results

Movement classification confusion matrix

True Class \ Predicted Class	no movement	side ankle	swipe left	swipe right	tap		
no movement	24170	2				100.0%	0.0%
side ankle	2	12				85.7%	14.3%
swipe left	3		11			78.6%	21.4%
swipe right				13		100.0%	
tap	13				1	7.1%	92.9%
	99.9%	85.7%	100.0%	100.0%	100.0%		
	0.1%	14.3%					

TPR = 0.6785

FPR = 0.0357

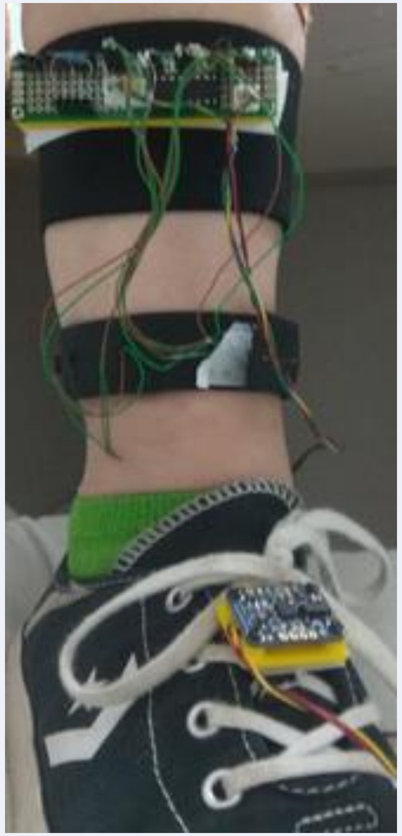
Discussion

- This project demonstrates classification of foot mounted IMU signal, of intuitive 3 movements using template matching method.
- The idea of using IMU & FSR signals from the leg is also under-development at DEKA Arm
  - Their solution at an advanced developments stage and has vast and robust functionality

DEKA Arm©

Conclusion

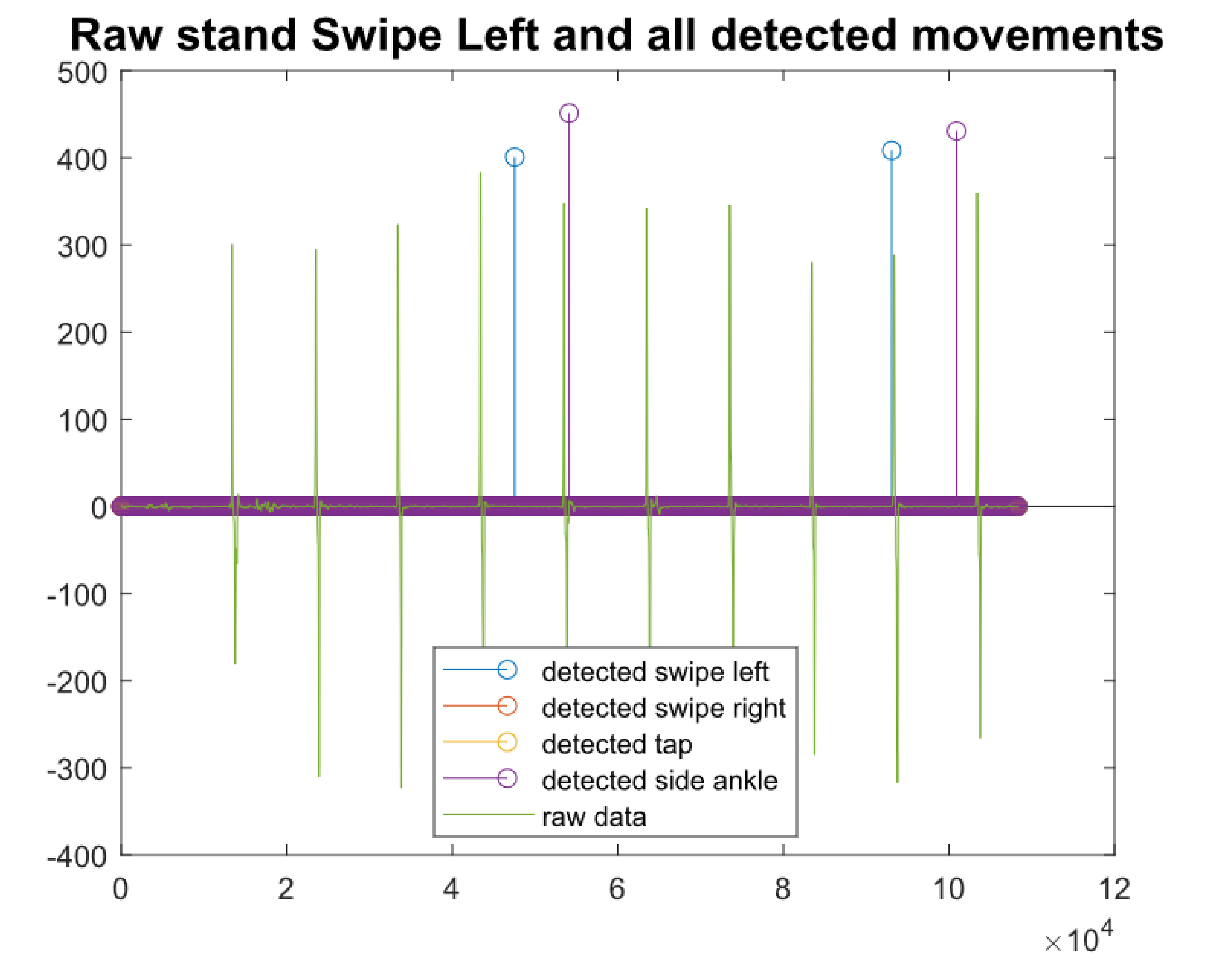
- Implementing system for acquiring data from the leg.
- Implementing 3 movements classifier
  - Intuitive movements
  - Templates were created
- Success rates of algorithm for chosen thresholds:
  - TPR = 0.881
  - FPR = 0.047



System on leg

Conclusion

- The classification method we used was template matching:
  - Performance may be reduced because the template made are specific for data acquired while sitting.
  - Can be improved by collecting more diverse data.
- Additional data that was already collected, can be used
  - Acceleration
  - FSR



Future Work

- Using FSR or acceleration to determine state of user (i.e walking/sitting/standing) then using designated template for each case.
- Using FSR data for determine intensity or duration of movement.
- Using quaternions to determine orientation of leg

References

1. Shunit Polinsky, Yair Herbst, and Dr. Yoav Medan "Interface Design for a Low-Cost 3D Printed Electro-Mechanical Prosthetic Hand"

2. Linda Resnik,Shana L Kinger, Katherine Etter "The DEKA Arm: Its features, functionality, and evolution during the Veterans Affairs Study to optimize the DEKA Arm" PubMed, Comparative Study, Prosthet Orthot Int.2014 Dec; 38(6):492-504. doi: 10.1177/0309364613506913