Towards a Practical EMG Gesture Recognition System

UNIVERSITY OF TORONTO

Sebastian Kmiec, Yuhao Zhou

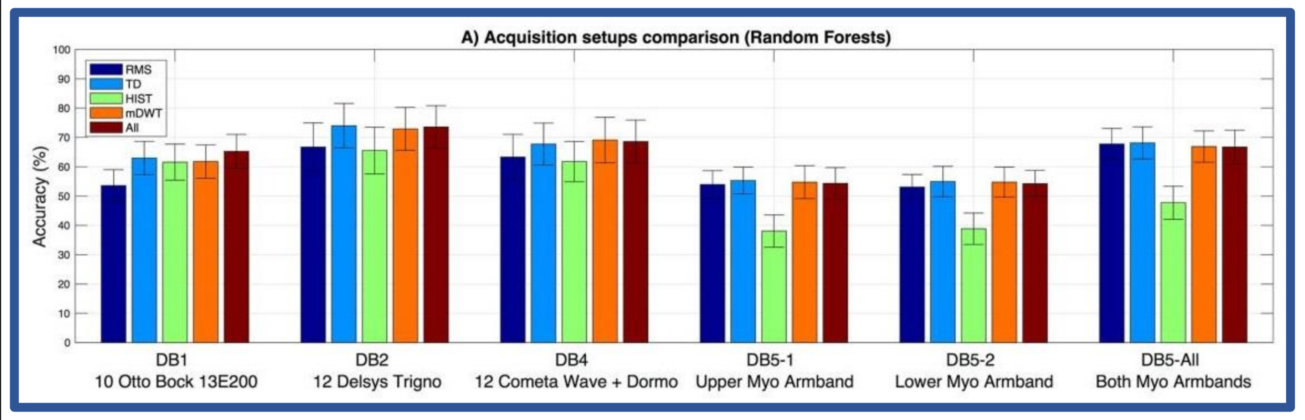
Supervisor: Stark Draper, **Admin**: Khoman Phang

ECE496 Capstone Design

GitHub

Motivation and Background

Motivation: Previous surface electromyography (sEMG) gesture recognition systems require expensive hardware setups and offer limited functionality.



ETH Zurich Research Results (2017):

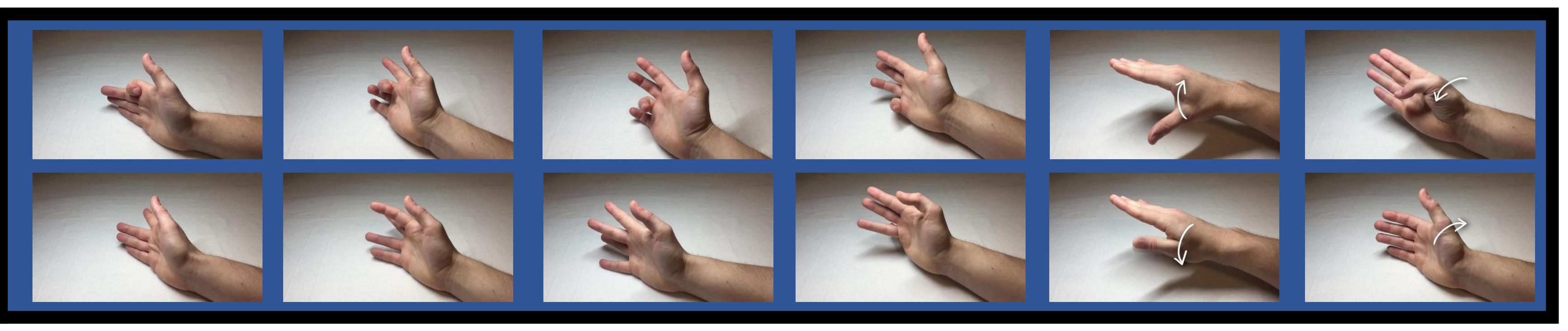
- Inexpensive hardware setups can produce similar results (**left**: *DB5-all*) [1]
- Creation of the NinaPro dataset, discussed below

Goal: Build a pipeline to perform real-time gesture recognition on low-cost hardware.

Design Requirements

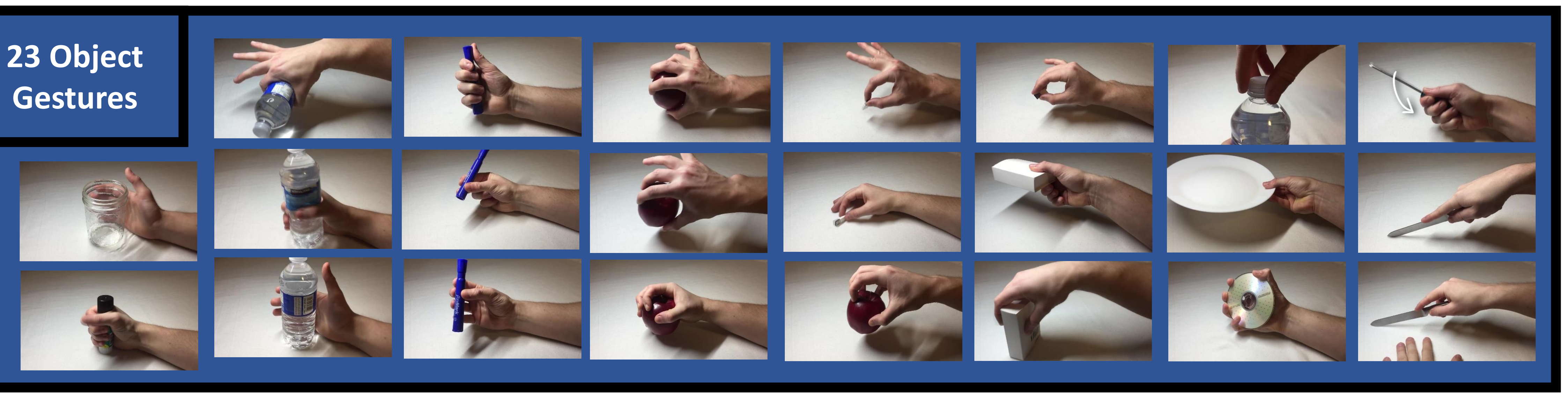
NinaPro: A new dataset, standardizing sEMG-based, gesture recognition algorithm benchmarks.

12 Finger Gestures



17 Hand Gestures

23 Object Gestures

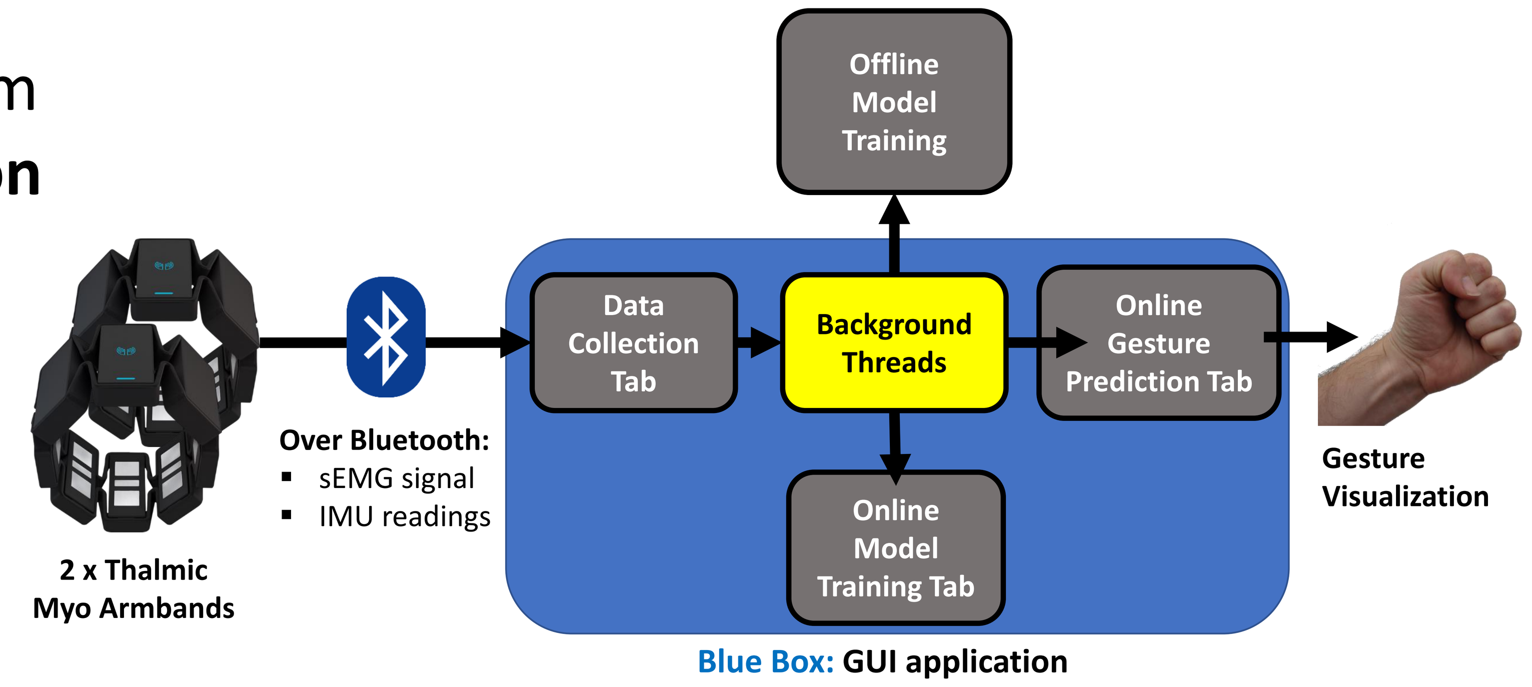


Constraints/ Objectives	1. **Design Cost**	Under 1000 CAD
	2. **Classification Accuracy**	(70/80)% on NinaPro dataset
	3. **Inference Time**	(125/300) ms per prediction
	4. **Training Data**	Six training samples per class

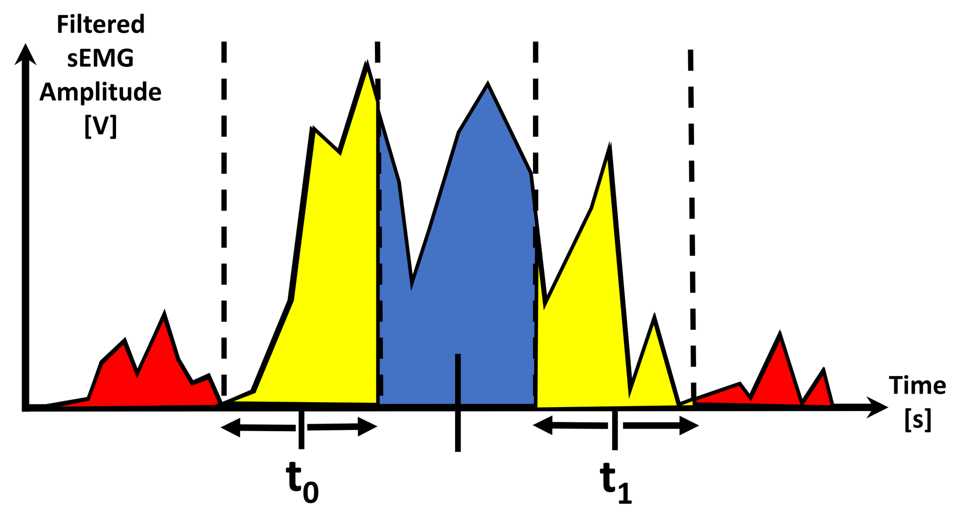
Final Design

A high-level block diagram of our **gesture recognition system**:

i. Bluetooth Library
ii. GUI
iii. Offline Modeling



A. Gesture Detection

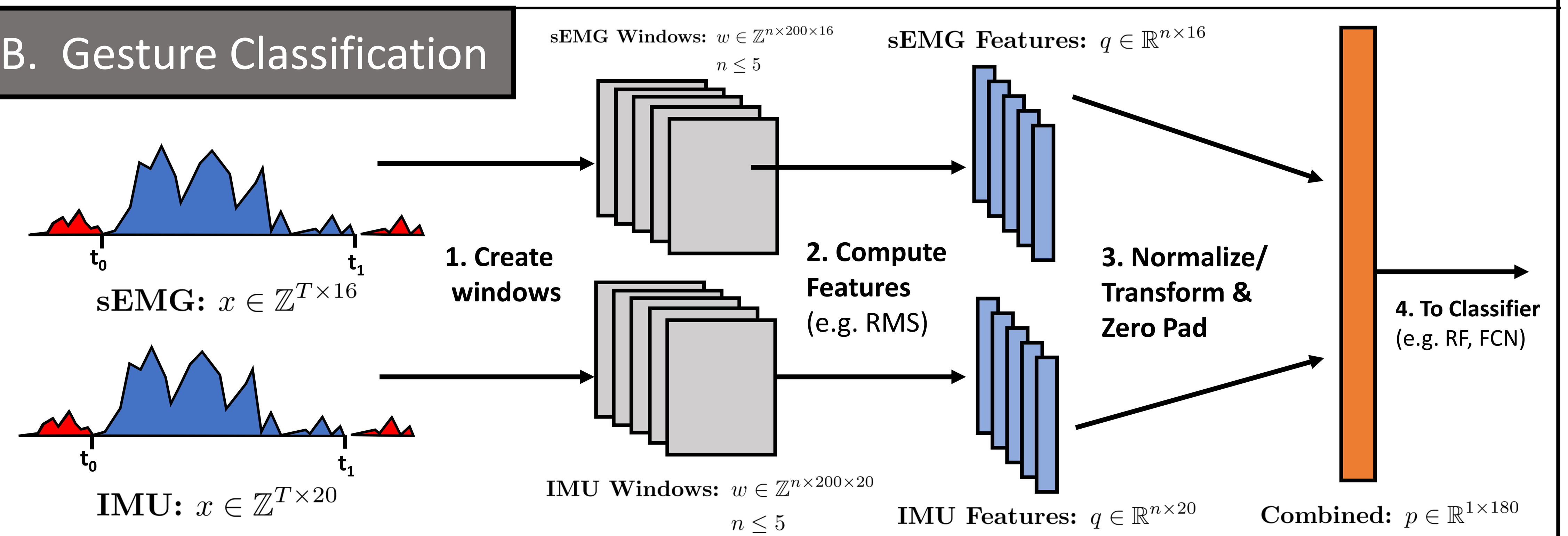


1. Compute preliminary t_0, t_1 from Lidierth window-threshold algorithm.
2. Refine t_0, t_1 by maximizing:

$$\sum_{i=1}^{t_0-1} \ln p_{\theta_0}(\mathbf{y}_i) + \sum_{j=t_0}^{t_1-1} \ln p_{\theta_1}(\mathbf{y}_j) + \sum_{k=t_1}^{T} \ln p_{\theta_0}(\mathbf{y}_k)$$

- T: Length of entire window
- $\mathbf{y}$: Multivariate (16 channel) sEMG data
- p_{θ_0}: A noise model fit on the data to the left of t_0 and right of t_1.
- p_{θ_1}: A signal model fit on the data between t_0 and t_1.

B. Gesture Classification



Results & Future Work

Features	Classifier	NinaPro	Own Data
Zero Padded RMS Windows	Random Forrest	82.2% (**7ms**)	89.5% (**7ms**)
	FCN	84.6% (35ms)	92.1% (33ms)
PCA, Normalized, Zero Padded, AM Features	Random Forrest	86.1% (9ms)	91.7% (9ms)
	FCN	**88.4%** (37ms)	95.4% (35ms)
Directly Above + IMU Features	FCN	N / A	**96.2%** (41ms)

1. Create synthetic training data (generative adversarial networks, data augmentation)
2. Use IMU data to switch between prediction models, for different resting forearm positions

[1] S. Pizzolato, L. Tagliapietra, M. Cognolato, M. Reggiani, H. Müller, and M. Atzori. Comparison of six electromyography acquisition setups on hand movement classification tasks. Plos One, 2017.