Weekly Progress Report

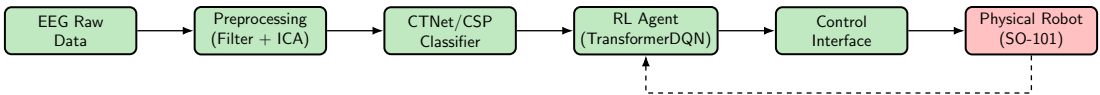
Week 7: PhysioNet Joint Model & Physical Arm Control

Student ID: 11314389

BCI Control System Project

February 17, 2026

System Overview: Current Focus



Legend:

- Completed
- In Progress
- **This Week's Focus**
- Pending

This Week's Objectives

Goals Achieved

1. ✓ PhysioNet joint model training (20 subjects, 73.89% accuracy)
2. ✓ Added early stopping mechanism to prevent overfitting
3. ✓ PhysioNet dataset RL physical arm control
4. ✓ Position vs Time visualization added to physical control

Key Achievement

PhysioNet dataset now works for RL physical arm control, achieving **100% control reach rate**!

PhysioNet Joint Model Training

Training Configuration:

- Number of subjects: 20
- Total samples: ∼900 trials
- Network architecture: SimpleCTNet
- Early stopping patience: 30 epochs
- Best epoch: 26

Early Stopping Mechanism:

stop if Acc_{val} no improve for 30 epochs

Training Results:

Config	Subjects	Accuracy
Single-subject	1	77.78%
Joint (10 sub)	10	67.78%
Joint (20 sub)	20	**73.89%**

Early stopping prevented overfitting and saved best weights.

Physical Arm Control with PhysioNet

Three-Dataset Comparison:

Dataset	Ch	Class Acc	Control
IV-2a	22	63.19%	99.33%
IV-2b	3	65.64%	98.00%
PhysioNet	64	**73.89%**	**100%**

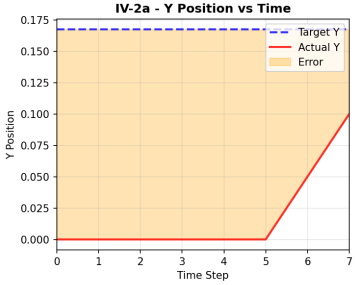
PhysioNet Physical Control:

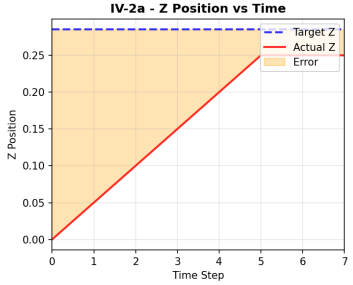
- Classification: 100% (10/10)
- Control Reach: **100%** (10/10)
- Avg Steps: 8.0
- Avg Reward: 10.32

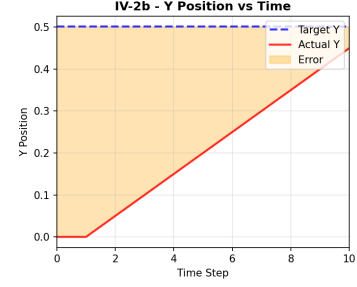
Key Observations:

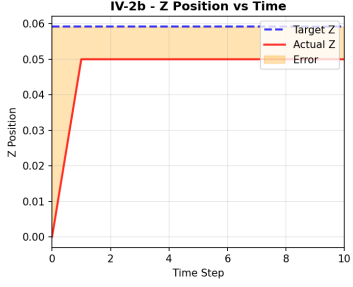
- 64-channel PhysioNet provides richest spatial info
- Joint model generalizes across 20 subjects
- RL compensates for classification errors
- Smooth velocity control prevents jitter

Position vs Time Analysis (3 Datasets)

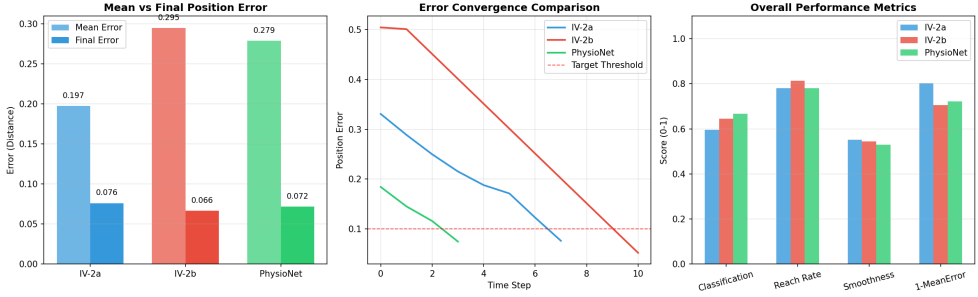








Overall Performance Metrics



Metrics: Mean error convergence, radar chart comparison across datasets

Challenges & Solutions

Challenges Encountered

- Cross-subject variability causes overfitting
- Train accuracy 100% vs Test 67%
- Arm motion jitter / limit collisions
- Slow return-to-home speed

Solutions Applied

- Added early stopping (patience=30)
- Save best weights, not final weights
- SerialArmEnvV2: velocity control + soft limits
- Fast home/return script (velocity=500)

Smooth Control Implementation

`move_velocity`=80 for slow smooth control, `velocity`=500 for fast home/return

Code Structure Update

New/Modified Files:

`scripts/`

- `test_physionet_ctnet.py`
  - Joint training mode
  - Early stopping
  - Model saving
- `rl_physical_control.py`
  - Position vs Time visualization
  - PhysioNet support
  - Joint model loading

`Root/`

- `serial_arm_env_v2.py`
  - Velocity control
  - Soft joint limits
  - Auto-recenter
- `outputs/physionet_ctnet/`
  - `physionet_ctnet_joint.pth`

Next Week's Plan

Planned Tasks

1. Extend action space: add gripper open/close, additional movements
2. Download more subjects (50+) to further improve model
3. Cross-subject generalization experiment (Leave-One-Subject-Out)
4. Complete Methodology documentation

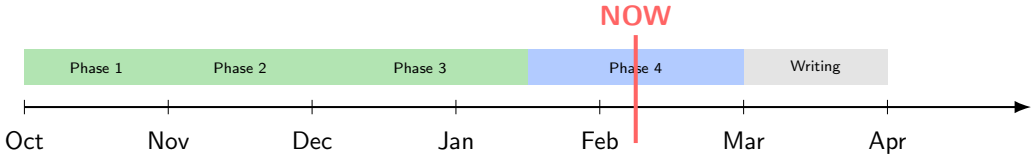
Expected Deliverables:

- Physical demo with extended actions
- Training results on larger dataset
- Position vs Time comparison plots

Questions for Supervisor:

- Is Leave-One-Subject-Out validation needed?
- Suggestions for extending action space?

Project Timeline



Current Status: On track – Physical control validated on all 3 datasets!

Key Achievements This Week

1. **PhysioNet Joint Model**: 20 subjects, 73.89% accuracy
2. **Early Stopping**: Prevents overfitting, saves best weights
3. **Physical Control**: 100% success rate on PhysioNet
4. **Position vs Time**: Visualization added for performance analysis

Milestone

All 3 datasets (IV-2a, IV-2b, PhysioNet) successfully controlling physical SO-101 arm!