Understanding Motor Intentions Using EEG Brain Wave Data

Jamshaid Ali

August 3rd

1 General Overview

Our goal is to build a system based on machine learning and statistical forecasting that is able to understand when someone wants to move their left hand versus their right hand just by looking at their brain signals (using EEG). And to also understand which portions of the brain (left vs. right) are responsible for different motor functions. For instance, we want to see how much of an impact the left side of the brain will have on the brain compared to the right side when it comes to our data. The biggest issue and where a lot of our statistical methods will come into play will be in handling the excessive noise that ineveitably comes from EEG data. And one of our primary goals mathemetically is signal seperation and cleaning data.

My plan at the moment is to model the EEG data as follow:

$$X(t, s) = S_{\text{motor}}(t, s) + N(t, s) \tag{1}$$

where $X \in \mathbb{R}^{T \times S}$ is what we actually measure, $S_{\text{motor}} \in \mathbb{R}^{T \times S}$ is the clean motor intention signal we want, and $N \in \mathbb{R}^{T \times S}$ is everything else (noise, artifacts, other brain activity). The whole point is to design algorithms that can extract S_{motor} reliably. I'm hoping that we can develop a way to do this within>90% accuracy in under 100ms. However this is ambitious and an arbitray benchmark I've set.

We're training models to minimize this objective:

$$\Phi^* = \arg\min_{\Phi} \mathbb{E}_{(X,y)\sim\mathcal{D}}[L(y, \Phi(X))] + \lambda\|\theta\|_2^2 \tag{2}$$

This just says we want to minimize classification errors across all our data while preventing overfitting with the regularization term.

2 Why the Brain Makes This Problem Solvable

The machine learnign portion of thie project comes from exploiting pecific patterns we find in the EEG data that our model will be trained on. I believe the most important one for our purposes is contralateral control: where your left brain controls your right body and vice versa.

Our goal is to find predicitable spacial patters that we can model mathematically, where we can seperate the left and right limbs from each other into matheamtical equations that should look something like this:

$$S_{\text{left}}(s) = A_L \exp\left(-\frac{\|s - s_{C4}\|^2}{2\sigma_s^2}\right) \tag{3}$$

$$S_{\text{right}}(s) = A_R \exp\left(-\frac{\|s - s_{C3}\|^2}{2\sigma_s^2}\right) \tag{4}$$

This is just a prediction from some sources that I was looking at, and they are subject to change. But my hypothesis is essentially that motor signals are strongest at specific electrodes (C3 for right-hand movements, C4 for left-hand movements) and fall off exponentially as you move away. The $\sigma_s \approx 1.5$cm parameter comes from the physical spacing of EEG electrodes which is arbitray atm.

The reason why I made the spatial localization so large is because we want motor signals and noise to have have different spatial signatures that we can separate algorithmically.

3 Frequency Domain Analysis

Motor intention signals manifest most clearly in specific frequency bands rather than across the entire EEG spectrum. This frequency selectivity is crucial for our signal separation problem because it allows us to isolate motor-related neural activity from the broadband noise that dominates raw EEG recordings.

Decades of motor cortex research have identified two frequency bands that reliably encode movement preparation and execution:

$$P_\mu(t,s) = \sum_{f=8}^{12} |X(t,f,s)|^2 \quad \text{(mu band, 8-12Hz)} \tag{5}$$

$$P_\beta(t,s) = \sum_{f=13}^{30} |X(t,f,s)|^2 \quad \text{(beta band, 13-30Hz)} \tag{6}$$

The mu rhythm represents the brain's motor cortex in an idling state. For our purposes this rhythm exhibits event-related desynchronization (ERD) during movement planning—the power in this band decreases when motor neurons prepare for action. This creates a reliable neurophysiological marker we can exploit for classification:

$$\text{ERD}(t) = \frac{P_{\text{baseline}} - P_{\text{active}}(t)}{P_{\text{baseline}}} \times 100\% \tag{7}$$

By focusing on these specific frequency bands I've laid out, we dramatically improve the signal-to-noise ratio of our motor intention detection. This frequency domain filtering will serve as the first stage of our signal separation pipeline, and hopefully remove a lot of the non-motor brain activity and other noise that would otherwise confound our spatial pattern analysis.

Just from some research I'm guessing the baseline period (-1000 to -500ms before movement cues) shoudl be our starting point fo ERD calculations. This is known to be best for capturing the resting state before any movement preparation begins and this temporal specificity will hopefully be best for our purposes of measuring genuine motor-related changes rather than noise we aren't interested in.

4 Dataset Selection and Training Data

We need to find existing EEG datasets that are actually suitable for motor intention classification. After looking at what's available, I've noticed a major problem with most EEG datasets: they use stimulus cues instead of actual movement timing for labels. Most of the data I've looked ad showed that there's usually only a 200-500ms delay between when someone sees a cue and when they actually intend to move. So we will have to find a workaround to that or have less ambitous goals. **The data sets we train on must meet these reqs for our model:**

- Motor imagery tasks with clear left vs right hand instructions
- Datasets that recorded actual movement timing (EMG or motion sensors) alongside EEG
- Focus on C3/C4 electrode coverage over motor cortex
- At least 500+ trials per condition for statistical power
- Sampling rates of 250Hz or higher to capture mu/beta rhythms properly

5 Signal Preprocessing and Artifact Removal

Regardless of the data sets we choose to use, we know that the Raw EEG recordings will contain substantial contamination from physiological and environmental sources that must be systematically removed before analysis. We need to develop a preprocessing pipeline that addresses these contamination sources through a sequence of targeted filtering operations, each designed to preserve motor-related neural activity. I have come up with some ideas on what we can do...

Primary Filtering Operations:

- Bandpass filtering (0.5-40Hz) removes low-frequency drift and high-frequency muscle artifacts
- Notch filtering (50/60Hz) will eliminate power line interference that is common in EEG processing
- Fourth-order Butterworth implementation can help ensure optimal frequency selectivity without phase distortion

Removal with Statistical Methods: Now I found through some research papers that FastICA decomposition might be our best bet. In short this method will exploit the statistical independence between neural signals and physiological artifacts. The papers I saw looked applied this to eye blinks and muscle contractions, which exhibit characteristically different temporal and spatial signatures compared to motor cortex activity. And that artifacts typically present as high-amplitude, temporally discrete events with broad spatial distribution, while motor signals manifest as sustained oscillatory patterns with focal spatial localization. **Spatial Referencing:** Common average referencing removes signals present across all electrodes while preserving locally generated motor activity:

$$X_{\text{ref}}(t, s) = X(t, s) - \frac{1}{S} \sum_{i=1}^{S} X(t, s_i) \tag{8}$$

This equations should help eliminate distant noise sources while maintaining the spatial specificity needed for hemispheric lateralization analysis.

6 Spatial Pattern Optimization

Common Spatial Patterns (CSP) will be the cornerstone of our filtering method, providing optimal linear combinations of electrode signals that maximize discriminability between motor conditions. The algorithm below will hopefully be close to an accurate representation of what it takes to extract motor intention signals from the complex spatial distribution of EEG activity. CSP optimization begins with computation of normalized covariance matrices for each movement class:

$$C_1 = \frac{1}{N_1} \sum_{i:y_i=1} \frac{X_i X_i^T}{\text{tr}(X_i X_i^T)} \; C_2 \qquad = \frac{1}{N_2} \sum_{i:y_i=2} \frac{X_i X_i^T}{\text{tr}(X_i X_i^T)} \tag{9}$$

The trace normalization ensures that spatial pattern differences, rather than overall amplitude variations, drive the optimization process. The algorithm then solves the generalized eigenvalue problem:

$$C_1 w = \lambda (C_1 + C_2) w \tag{10}$$

This formulation identifies spatial filters w that maximize the variance ratio:

$$\frac{w^T C_1 w}{w^T (C_1 + C_2) w} \tag{11}$$

The resulting filters will automatically discover multi-electrode patterns that should distinguish between left and right motor intentions. The idea is that we will cause specific movements to have high variance while keeping the total variance controlled, that way the extreme eigenvalues will hopefully give the bets spatial filters. The reason why CSP shoudl work is because motor signal patterns will be inheretly complex and there will be too many different channels to look through, so this should be the most efficient and effecitve solution.

7 Feature Engineering and Integration

This feature extraction strategy will combine complementary information sources to create a comprehensive representation of motor intention states. This approach takes advantage of both established neuro-physiological markers and data-driven spatial patterns to maximize classification performance. We will than be left with a final feature vector integrates spectral and spatial information:

$$\phi(X) = [P_\mu(C3), P_\mu(C4), P_\beta(C3), P_\beta(C4), \log\big(\mathrm{var}(w_1^T X)\big), \ldots, \log\big(\mathrm{var}(w_k^T X)\big)]^T \tag{12}$$

Feature Categories:

- **Spectral Features:** Mu and beta band power at key electrode locations provide direct measures of motor cortex activation states which is why this is the basic neurophysiology everyone uses everyone uses
- **Spatial Features:** CSP-derived variance measures capture complex multi-electrode patterns not evident in single-channel analysis
- **Logarithmic Transformation:** Addresses the skewed distribution characteristics typical of EEG power measurements, improving classifier performance

I'm expecting this hybrid approach to give us the best chance at detecting individual differences in brain anatomy and signal characteristics while maintaining interpretability through neurophysiologically grounded features.

8 Classification Methodology

The reason why the classification framework will employ multiple algorithmic approaches is because there is are diverse characteristics of EEG data across subjects and recording conditions. So a multi-method strategy should provide us with both baseline performance metrics and complex pattern recognition. **Support Vector Machine Implementation:** SVMs provide great performance for high-dimensional, noisy data through margin maximization, returning support vectors through this optimization:

$$\max_{\alpha} \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \text{ subject to } \quad \sum_{i=1}^{N} \alpha_i y_i = 0, \quad 0 \leq \alpha_i \leq C \tag{13}$$

The resulting decision function depends only on support vectors, providing inherent robustness to outliers and noise:

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i \exp\bigl(-\gamma |x_i - x|^2\bigr) + b \tag{14}$$

Probabilistic Classification: Logistic regression provides calibrated probability estimates essential for real-time decision making:

$$L = -\sum_{i=1}^{N} [y_i \log \pi_i + (1 - y_i) \log(1 - \pi_i)] + \lambda |w|_2^2 \tag{15}$$

where $\pi_i = \sigma(w^T x_i + b)$ Probability estimates enable confidence-based control strategies, allowing the system to defer decisions when classification certainty is insufficient.

Deep Learning Architecture: Incoproarting Deep Learning feels promising since EEGNet provides end-to-end learning capabilities through biologically inspired convolution operations which can look like something like this:

$$\mathbb{R}^{1\times C\times T} \xrightarrow{\text{Conv2D}} \mathbb{R}^{8\times C\times T} \quad \xrightarrow{\text{DepthConv2D}} \mathbb{R}^{16\times 1\times T} \xrightarrow{\text{SepConv2D}} \mathbb{R}^{16\times 1\times T'} \quad \xrightarrow{\text{Dense}} \mathbb{R}^2 \tag{16}$$

Each layer corresponds to traditional EEG processing operations: temporal convolution performs frequency filtering, depthwise convolution implements spatial filtering analogous to CSP, and separable convolution extracts temporal patterns.

9 Training and Validation Strategy

From my research it is obvious that EEG data presents unique challenges. And that we require specialized training approaches that account for biological variability, noise characteristics, and individual differences in neural signatures. This potential validation framework will address both within-subject learning and cross-subject generalization.

Cross-Validation Design:

- Within-subject 5-fold validation assesses individual pattern learning capability
- Leave-one-subject-out validation evaluates generalization across anatomical and physiological differences

Optimization Parameters: Adam optimization with learning rate decay ($\text{lr}(t) = 0.001 \cdot 0.95^{\lfloor t/100 \rfloor}$) and early stopping (20-epoch patience) balance convergence speed with overfitting prevention. **Data Augmentation Strategy:**

- Additive Gaussian noise ($X' = X + \epsilon$, $\epsilon \sim \mathcal{N}(0, 0.1\sigma_X^2)$) simulates variability
- Channel dropout (10% probability) simulates electrode failures and prevents over-reliance on specific sensors

Performance Metrics:

$$\text{Accuracy} = \frac{TP + TN}{N} \tag{17}$$

$$\text{Cohen's } \kappa = \frac{p_o - p_e}{1 - p_e} \tag{18}$$

$$\text{AUC-ROC} = \int_0^1 \text{TPR}(\text{FPR}^{-1}(x))dx \tag{19}$$

Cohen's kappa provides essential correction for temporal correlations that can artificially inflate standard accuracy metrics in time-series data.

10 Real-Time Processing Architecture

The Exoskeleton team's goal is to deliver real-time motor intention detection, so we emphasize temporal features in our models and data, and we hope the data we collect will help us achieve this objective. As you can see, we're hoping to achieve this through creating a processing architecture that can meet stringent latency requirements through algorithmic optimization and efficient memory management. Although the latency below is subject to change, it would be optimal

Latency Breakdown:

$$t_{\text{total}} = \underbrace{20\text{ms}}_{t_{\text{acquisition}}} + \underbrace{30\text{ms}}_{t_{\text{preprocess}}} + \underbrace{25\text{ms}}_{t_{\text{FFT+CSP}}} + \underbrace{15\text{ms}}_{t_{\text{classify}}} = 90\text{ms} \tag{20}$$

Temporal Windowing: The plan is to have 250ms analysis windows with 50ms updates provide optimal frequency resolution for mu/beta detection while maintaining 20Hz control rates suitable for responsive brain-computer interfaces.

Adaptive Feature Updates: We have limited computing power so I'm expecting to use exponential smoothing to reduce computational overhead while maintaining responsiveness:

$$P_\mu^{(n+1)} = 0.9 \cdot P_\mu^{(n)} + 0.1 \cdot P_\mu^{\text{current}} \; P_\beta^{(n+1)} \quad = 0.9 \cdot P_\beta^{(n)} + 0.1 \cdot P_\beta^{\text{current}} \tag{21}$$

I believe a 0.9 forgetting factor will optimally balance noise rejection with system responsiveness **Memory Management:** Circular buffer architecture

with 1000-sample capacity (1 second at 1kHz) eliminates dynamic memory allocation during real-time operation, ensuring consistent latency performance.

11 Recap

Our primary objective centers on processing EEG data and returning reliable hemispheric motor intention classification. One of our largest obstacles is signal separation to reveal the underlying motor-based brain-computer interfaces. With the hope of establishing the mathematical and algorithmic foundation for more complex analysis of the realtiaionship between brainwaves and body-moto-functions.

Implementation Priorities:

- Synchronized EEG and inertial sensor data collection protocols
- Robust CSP implementation with comprehensive cross-validation
- Baseline SVM performance benchmarking
- Real-time processing pipeline with latency optimization
- Comprehensive performance evaluation across multiple metrics

The focus remains on practical implementation rather than creating more theory. The algorithm we create should demonstrate reliable performance under realistic operating conditions.