Using Machine Learning to Differentiate Haptic Interaction with a Human or a Robot Partner using EEG Hyperscanning

Anonymous Authors[1]

Abstract

When carrying out common action with a partner, such as dancing or or handling a large furniture, the exchange of haptic information can improve performance. Although haptic communication has been well studied at the behavioural level, the underlying neural mechanisms remain unclear. Previously, a study has investigated dyads of participants, who performed a target tracking task, either on their own, relaxed and passively moved by a robot, or interacting with each other or a robot, while recording their cortical activity using a hyperscanning EEG system. Results showed that Solo was different from the other conditions but no strong differences between the human-human and the human-robot interaction in all metrics were observed. Here, the same dataset was used to examine how the different experimental conditions in this experiment could be classified using supervised machine learning algorithms on the previous data, and how this could distinguish the different haptic interaction types. Support vector machine, k-nearest neighbours, extreme gradient boosting decision tree (XGBoost), and StackNet were used to classify the seven experimental conditions, and the accuracies obtained ranged from 0.4 to 0.72 (chance-level of 0.14), with the StackNet and the XGBoost performing best. Although the models identified the Solo condition with great accuracy (over 0.8), the partner conditions were more difficult to distinguish (accuracy from 0.58 to 0.73) and often confused with each other.

1. Introduction

Physical interaction between humans is crucial for performing joint tasks, like moving a heavy piece of furniture, and learning new motor tasks, such as when a parent holds their child's hand to teach them how to walk. This transmission of force and/or haptic information is commonly referred to as haptic communication, where two individuals can exchange their motion plan and infer information about the task using the haptic feedback from their partner (Takagi et al., 2017). The behavioural characterisation of this haptic communication has been studied in previous work using a dual robotic interface where two individuals were haptically connected via a virtual spring while performing a target tracking task (Ganesh et al., 2014). The presence of this haptic connection can benefit both participants, even the more skilled one, facilitating task improvement and learning (Ivanova et al., 2022; Takagi et al., 2017; Beckers et al., 2020; Noccaro et al., 2023). Moreover, key aspects of motor control, such as performance, motion smoothness and energy, as well as interaction perception are different when a human interacts with a *human partner* and with *trajectory guidance*, traditional robotic assistance exactly following the target (Takagi et al., 2017; Ivanova et al., 2020; 2022). On the other hand, when interacting with a human-like *robot partner* (Takagi et al., 2017), the resulting individual's performance and learning are similar to when interacting with a human, although some motion and perception attributes differ (Ivanova et al., 2020; 2022). Hence, there are specific characteristics at the behavioural and perception level that can distinguish the different interaction partners.

The neural mechanisms underlying this complex sensorimotor integration, however, are not well-defined yet. Classifying different types of haptic feedback using recorded neural signals could help to improve our understanding and benefit the development of more natural human-robot interfaces, brain-computer interfaces and haptic feedback systems, for various applications such as prosthetic limb control or remote precision surgery. Previous work has used machine learning algorithms to distinguish several haptic interactions using electroencephalography (EEG) recordings. For instance, the presence of delay in the perceived haptic feedback during a visuo-haptic task has been classified with an accuracy over 75% using a Support Vector Machine (SVM) and Extreme Gradient Boosting decision tree (Alsuradi & Eid, 2023). Different textures could also be identified from EEG data collected during object interaction. The classifi-

[1] Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

cation of three roughness levels was correct over 85% of the time using an SVM (Eldeeb et al., 2020). A similar four-class problem in a study investigating time-domain and frequency-domain EEG features during rest or interaction with a rough object, a smooth object, and a liquid object obtained an accuracy of 64% with a k-Nearest Neighbours algorithm (kNN), and over 70% with random forest and SVM (Aspiotis et al.). Levels of cognitive conflict during a physical human-robot interaction were also correctly classified over 50% of the times in a three-class problem using supervised machine learning algorithms (Aldini et al., 2023). However, these studies did not explore EEG during haptic communication between individuals and how it differs from the interaction with a robot partner.

We have recently investigated cortical signals of dyads of participants, collected with EEG, while they interacted with each other, the robot partner and robotic guidance during a target-tracking task (Tong, 2024). Control conditions exploring several sensory modalities, i.e. vision, proprioception, haptic feedback, and volition were also tested. A classical EEG analysis of this data, including power spectral density, inter-brain synchrony and fuzzy entropy, of the EEG signal in the regions of interest has shown that the brains of the two participants synchronised when they were connected or when they received simultaneously a similar haptic feedback, while fuzzy entropy seems to be correlated with force transmission (Tong, 2024). These metrics did not indicate a difference between the human-human interaction and the human-robot interaction (Tong, 2024). Hence, this work aims to uncover more subtle differences between the investigated haptic interaction conditions by using supervised machine learning models, trained on features extracted from the whole collected EEG dataset.

2. Methods

2.1. Experiment and collected dataset

The study involved 24 right-handed individuals (12 dyads) without known impairments (mean age: 23.6 ± 1.4 years). Ethical approval was obtained from the Research Ethics Committee of Imperial College London (reference: 21IC6578).

The dyads of participants performed the same wrist target tracking task using the Hi5 dual robotic interface (Melendez-Calderon et al., 2011), while seated at individual monitors, displaying their wrist position and the moving target to follow (Fig. 1 A). Various conditions were presented to the participants, which incorporated different combinations of visual feedback, provided through changes in the screen display, and haptic feedback on the participants' wrist, by applying robot torque. During the experiment, participants were separated by a curtain and they were unaware of the partner's task and were asked to not to communicate with each other.

The experiment included three interaction conditions and four control conditions (Fig. 1 B). The interaction conditions were: (i) *human partner (HP)*, where the two participants in the dyad were connected via a virtual spring; (ii) *robot partner (RP)*, involving a reactive human-like algorithm from (Takagi et al., 2017) that replicated the initial participant's performance accuracy; and (iii) *trajectory guidance (TG)*, where a spring-like force connected the participant's wrist to the target, as typically used in robot-assisted rehabilitation and industrial applications. The control conditions were: (i) *Solo*, where participants performed the task independently without a partner; (ii) *Vision*, in which participants were asked to relax and follow the visual target displayed on the screen with their eyes while their hand stayed still in one position; (iii) *Hand*, involving passive wrist motion performed by the robot, with haptic stimuli but without any visual feedback; and (iv) *Passive*, involving passive wrist motion performed by the robot with the target visualised on the screen. Each condition included 23 trials lasting 15 seconds.

Using the hyperscanning technique, EEG data was recorded in both participants, at a sampling frequency of 500 Hz, while they tracked the moving target in the various conditions. Two eego mylab amplifiers (ANTneuro, Netherlands) were used to collect the data from 2*64 channels using the 10-10 international positioning system through Ag/AgCl active electrodes (waveguard cap ANTneuro, Netherlands), where electrode impedances were kept below 10 kΩ. CPz was used as the reference electrode and AFz as the ground electrode. The eemagine software (eemagine Medical Imaging Solutions GmbH, Germany) was used to acquire and save the data, and the pre-processing steps for the EEG data, performed using EEGLab (Delorme & Makeig, 2004), are outlined in the diagram in Fig. 1 C. The EEG dataset consists of 14 files (7 conditions * 2 participants) of 276 epochs (23 trials * 12 dyads) each, except for the Solo condition which has 3 epochs less. Further details on the experimental design, including the connection stiffness parameters and the equation of the target trajectory, as well as detailed descriptions for each pre-processing step are provided in (Tong, 2024).

2.2. Feature extraction

To extract relevant features from the collected EEG dataset, we used the comprehensive list from Stancin and colleagues (Stancin et al., 2021), who explored EEG metrics during driving. Their task involving visual and haptic feedback were similar to ours. The feature set consisted of features extracted from the cortical signals of each channel in the time-domain, frequency-domain and time-frquency domain.

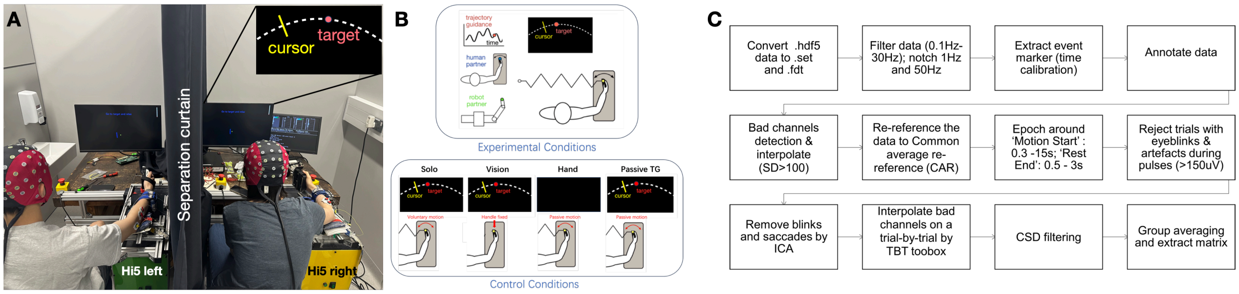

A
cursor
target
Separation curtain
Hi5 left
Hi5 right
B
trajectory guidance
time
cursor
target
human partner
robot partner
Experimental Conditions
Solo
Vision
Hand
Passive TG
cursor
target
Voluntary motion
Handle fixed
Passive motion
Passive motion
Control Conditions
C
Convert .hdf5 data to .set and .fdt
Filter data (0.1Hz-30Hz); notch 1Hz and 50Hz
Extract event marker (time calibration)
Annotate data
Bad channels detection & interpolate (SD>100)
Re-reference the data to Common average re-reference (CAR)
Epoch around 'Motion Start' : 0.3 -15s; 'Rest End': 0.5 - 3s
Reject trials with eyeblinks & artefacts during pulses (>150uV)
Remove blinks and saccades by ICA
Interpolate bad channels on a trial-by-trial by TBT toobox
CSD filtering
Group averaging and extract matrix

Figure 1: Experimental setup and protocol. **A.** Laboratory setup to study brain activity during human-robot and human-human interaction. Subjects were interacting using wrist flexion/extension. The cortical activity is recorded using an EEG hyper-scanning system. **B.** Three interaction conditions with a partner and four control conditions, investigating different sensory feedback modalities, are tested. **C.** EEG pre-processing pipeline.

The *time-domain* features included the mean, median, variance, standard deviation, kurtosis and peak amplitude values of the signals. In the *frequency-domain*, features were extracted from the Power Spectral Density (PSD) of the signal, a spectral representation of the power distribution of a signal's frequency components. From the PSD, the power of typical EEG frequency bands were extracted, namely the delta (1-4 Hz), theta (4-8 Hz), alpha (8-13 Hz), beta (13-30 Hz) and gamma (30-50 Hz) bands. To obtain features in the *time-frequency domain*, the Discrete Wavelet Transformation (DWT) of each channel was computed, as it is suitable for non-stationary signals like EEG, and the mean, median, variance, standard deviation, maximum and minimum values were extracted. Daubechies Wavelet with four levels of decomposition was applied as it is the most commonly used and strikes a balance in capturing details of both low and high frequency components. A second set of time-frequency features was also extracted from the Short-Time Fourier Transform (STFT) of the signals as it is also suitable for non-stationary signals. STFT uses an overlapping window to segment a signal over time and computes the Fourier Transform. The mean, median, variance, standard deviation and peak amplitude of the STFT output were calculated and also used as time-frequency features.

The final feature set after feature extraction consisted of 3861 samples (24 participants * 23 trials * 7 conditions, except 3 rejected trials in the Solo condition) each having 3172 features. To avoid overfitting, a loss in performance and high computational costs, due to the curse of dimensionality, Principal Component Analysis (PCA) was used. PCA is a statistical method that transforms the correlated features of the dataset into a set of linearly uncorrelated features – principal components (PCs).

2.3. Classification algorithms

We used Support Vector Machine (SVM), k-Nearest Neighbours (kNN), Extreme Gradient Boosting (XGBoost) and StackNet to classify the seven conditions tested in our experiment (with chance level of 14%), as these algorithms exhibited good classification performance on EEG data in previous works (Alsuradi & Eid, 2023; Eldeeb et al., 2020; Aspiotis et al.; Schaabova et al., 2015; Bilucaglia et al., 2021) .

For each algorithm, the classification was carried out in two phases. First, the whole training set of extracted features (i.e. without PCA) was used to train each model with the default hyperparameters to obtain the models' baseline performances on the test set. In the second phase, hyperparameter tuning was carried out using the training set to find the optimal hyperparameters for each model, including the optimal number of PCs. The range of PCs explored was from 50 to 300, as these capture between 80% and 95% of variance in the data. With each model, the 80:20 ratio was used for the training-test sets splitting strategy. Stratified five-fold cross validation was used on the training set for the hyperparameter tuning for the second phase.

The Tree-structured Parzen Estimator (TPE) (Bergstra et al., 2011) from the HyperOpt library was used to explore the search space. TPE is a Sequential Model-Based Optimization (SMBO) algorithm suitable for high-dimensional spaces and balances exploration and exploitation by intelligently selecting the next set of hyperparameters for evaluation. TPE uses the negative cross-validation accuracy as the objective function to define two Probability Density Functions (PDF) estimated using Kernel Density Estimation (KDE). The next set of configurations is selected by maximizing the ratio of the two sets, and the whole process is repeated for 50 evaluations.

Support Vector Machine (SVM) is a non-parametric supervised machine learning algorithm that classifies the data according to the optimal line or hyperplane maximising the distance between each class in an N-dimensional plane. Given the domain knowledge that the EEG signals are highly non-linear and complex, Radial Basis Function (RBF) with kernel $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$, was used in the first phase of experimentation with the whole feature set without PCA. The model was trained with regularization parameter C = 1 and γ = 0.000315, calculated as $1/(N_{features} * var(X))$ (Appendix, Table 7). In the second phase with hyperparameter tuning, two kernels suitable for EEG data, namely the Polynomial kernel, $K(x_i, x_j) = (\gamma x_i \cdot x_j + r)^d$ and the RBF kernel, were used to transform non-linearly separable data to a higher dimension that is linearly separable. Table 1 summarizes the search space defined for the hyperparameter tuning of SVM.

k-Nearest Neighbours (kNN) is a supervised machine learning algorithm that uses proximity neighbours and a voting system to make a prediction. It is a non-parametric method used for classification and is a lazy learner as no model is trained but instead, the data is memorised. In the first phase, all features were used to train the classifier, and baseline hyperparameters were selected as follows: $k = 5$, uniform voting weights, and the Euclidean distance (Appendix, Eq. 1, Table 7), while in the second phase, PCA was used and the search space for each hyperparameter is displayed in Table 2.

Table 1: SVM Search Space

Parameter	Range	Optimal
C (Regularization)	0.1 – 1000	115.85
Gamma γ	0.0001 – 1	0.0009
Kernel	Polynomial, RBF	RBF
Principal Components	50 – 300	300

Table 2: kNN Search Space

Parameter	Range	Optimal
k Neighbours	1 – 30	4
Voting Weights	Uniform, Distance	Distance
P	1 (Manhattan), 2 (Euclidean), 3 – 5	2
Principal Components	50 – 300	300

Table 3: XGBoost Search Space

Parameter	Range	Optimal
Boosting Rounds (n_estimators)	50 – 100	100
Max Tree Depths	3 – 10	9
Learning Rate (η)	0.001 – 0.3	0.083
γ Regularization	0 – 5	0.0027
Principal Components	50 – 300	130

Extreme Gradient Boosting (XGBoost) is a Gradient Boosting Decision Tree (GBDT) ensemble learning algorithm that uses multiple decision trees and the weighted sum of each prediction for the final output (Chen & Guestrin, 2016). The algorithm improves the performance of weak learners or trees by sequentially training the next model using the residual errors of the previous tree using the update rule (Appendix, Eq. 2). This algorithm is highly scalable and minimizes bias by combining multiple weak learners. It reduces underfitting using deeper trees in the ensemble and overfitting through regularization techniques. In phase one, XGBoost was trained on the whole feature set without PCA with the number of boosting rounds (n_estimators) set to 100, the maximum tree depth to 10, the learning rate η to 0.1 and the γ regularization factor to 1 (Appendix, Table 9). The search space for the hyperparameter tuning used in phase 2 is displayed in Table 3.

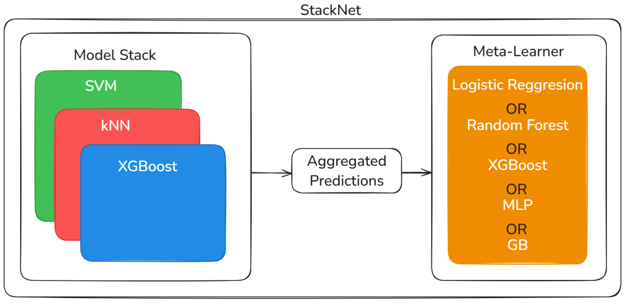


Figure 2: StackNet Model

StackNet is an advanced machine learning model that combines multiple machine learning models as a stack where the predictions from each model are used to make the final prediction. It is a type of ensemble learning and consists of two parts, namely a model stack and a meta-learner. The model stack contains two or more models, the base-learners, trained on the input features set and generating predictions, called the meta-features. The meta-features of each model are aggregated together and used as input for the final layer, called the meta-learner. The meta-learner can be any machine learning model that will be trained on the meta-features and produce the final prediction.

The base-learners used here are the SVM, kNN and XGBoost models, with their corresponding optimal hyperparameters (Tables 1, 2, 3), where the baseline model (without PCA and with default parameters) was used if outperforming the tuned one. Five different meta-learners, namely Logistic Regression (LR), Random Forest (RF), XGBoost, Multilayer Perceptron (MLP), and Gradient Boosting (GB), were used.

2.4. Performance measures

To evaluate the performance of each model, we computed the accuracy and F1-score on the test set, using the below equations. The accuracy of the validation sets was also calculated in the five-fold cross-validation.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 - score = \frac{2 * Precision * Recall}{Precision + Recall},$$

where TP stands for True Positive, TN for True Negative, FP for False Positive, and FN for False Negative.

3. Results

3.1. Support Vector Machine

Using SVM with the whole feature set to classify the seven experimental conditions, an accuracy and F1-score of 0.4 were obtained, with a chance level of 0.14. This poor performance was likely due to the high number of samples to classify (3861), which the feature space of 3172 features could not adequately cover, leading to poor generalisability.

In the second phase of classification with SVM, hyperparameter tuning and PCA were carried out. The optimal hyperparameters, obtained after 50 evaluations using the TPE algorithm, are displayed in Table 1. Using PCA in the pipeline, 300 components were used as features in the optimal model. The model performance improved compared to the baseline test, obtaining an accuracy and F1-score of 0.58. This was likely due to the reduction in the dimension of the feature set. The confusion matrix (Appendix, Fig. 7) indicates that most experimental conditions were correctly classified 50% of the time with 'Solo' being the highest and 'TG' being the lowest.

3.2. k-Nearest Neighbours

After fitting the classifier on the whole feature set, an accuracy and F1-score of 0.48 were obtained. Similarly, as for the SVM, this poor performance was likely caused by the high dimensionality of the feature set, especially since the kNN model relies on the distance metric and typically performs poorly on a highly sparse data space.

Using the optimal hyperparameters (Table 2), the kNN classifier performed slightly better than the previous, with an accuracy and F1-score of 0.5. This indicates that the hyperparameter tuning phase and the PCA did not improve the model's performance, suggesting this method may not be suitable. The confusion matrix (Appendix, Fig. 8) indicates that only three out of seven labels were correctly classified about 50% of the time with Solo being the highest at 61.98%, while the Passive label was correctly classified only 43.12% of the time, although this is much higher than the chance level of 14%.

3.3. Extreme Gradient Boosting

Using the default parameters, the XGBoost model yielded an accuracy and F1-score of 0.71, indicating a good performance compared to chance (0.14) and the other models tested. The nature of the algorithm is such that it can handle high-dimensional and sparse datasets, hence it seems that it could use all features without much loss in performance. The confusion matrix (Appendix, Fig. 9) shows that five out of seven labels were correctly classified above 65% of the time with Solo being the highest with 81.82%, while the classification of the HP condition was lowest with 58.33%.

Table 4: StackNet Results

Meta-Learner	Accuracy	F1-score
Logistic Regression	0.71539	0.72
Random Forest	0.72316	0.72
XGBoost	0.70505	0.71
MLP	0.69470	0.69
GB	0.69858	0.70

In the second phase including hyperparameter tuning and PCA, the model performed less well, with an accuracy and F1-score of 0.49. Table 3 reports the optimal hyperparameters obtained during the tuning phase, indicating that PCA reduced greatly the dimensionality of the feature set (130 PCs), although it may have removed useful information for the generalisability of the model. Indeed, the linear transformation of the PCA could lose the non-linear patterns present in the EEG features and fail to capture their variability.

3.4. StackNet

For the SVM and kNN models, the tuned model performed better than the baseline, hence PCA and the hyperparameters in Tables 1 and 2 were used, while the baseline XGBoost model performed better than the tuned one, hence the whole feature set and default model parameters (Appendix, Table 9) were used. Table 4 reports the results of the StackNet classifier with each meta-learner. The RF meta-learner outperformed all other models with an accuracy and F1-score of 0.72, closely followed by the LR meta-learner with the same F1-score and an accuracy of 0.71. The confusion matrix shown in Fig. 3 indicates that all labels were correctly classified over 55% of the time. The Hand condition was correctly classified with the highest accuracy (82.44%), while HP had the lowest accuracy of 58.33%.

4. Discussion

4.1. Classification of experimental conditions with EEG activity

All evaluated ML models resulted in well above chance level (0.14) classification of the seven experimental conditions, with accuracies on the test set from 40% to 72%. This indicates that EEG data contain sufficient information to differentiate when a participant performs a task alone, with a robot or human partner, or when they are relaxed and receive different sensory feedback modalities. The ensem-

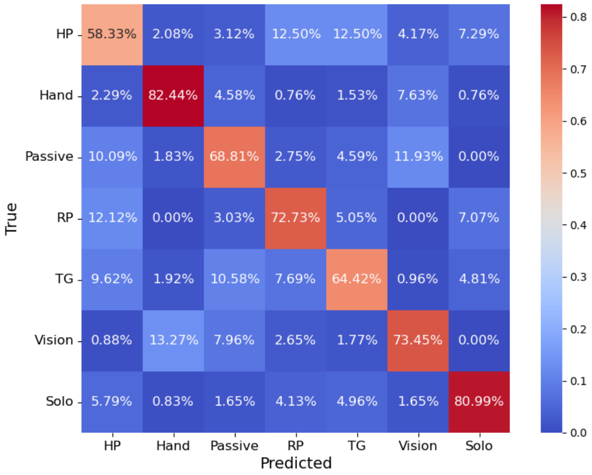


Figure 3: StackNet RF Confusion Matrix

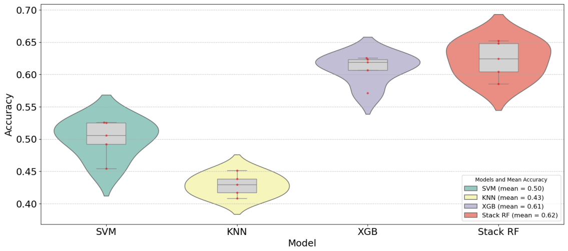


Figure 4: Accuracy of the four supervised ML models tested, across the five cross-validation folds

ble models, XGBoost and StackNet, achieved the highest classification accuracies, exceeding 70% overall. On the other hand, the accuracy of each model across the cross-validation folds (Fig. 4) shows the limited performance of simpler algorithms, such as SVM and kNN (with mean validation accuracies of 44% and 51%, respectively)), and their greater variability across cross-validation folds. Indeed, these algorithms may not be suited to capture non-linear relationships present in the EEG data.In contrast, XGBoost and StackNet with RF meta-learner, which are better suited for handling high-dimensional, noisy data, demonstrated higher and more consistent performance in the cross-validation folds, with narrower interquartile ranges. Thanks to the iterative improvement of the weak learner during the training of XGBoost, the non-linear relationships of EEG may be better captured, while the stacking of several machine learning methods in the StackNet model can provide a more stable classification result, as the strengths of several base-learners are used to compensate for their individual limitations.

The overall accuracy of the two best-performing models is also comparable to previous studies investigating EEG-based classification problems with similar algorithms, during haptic or tactile interaction, when identifying the presence of haptic delays from robot feedback (Alsuradi & Eid, 2023), object textures (Eldeeb et al., 2020; Aspiotis et al.; Cho et al., 2021) or the level of trust in a digital human-robot interaction (Xu et al., 2024). This similarity in results is noteworthy given the added complexity of our seven-class problem.

4.2. Identifying distinct and ambiguous conditions

Across all models, the Solo and Hand conditions consistently achieved the highest classification performance, with an accuracy greater than 80% in the StackNet with RF model. This aligns with findings from previous analysis on this dataset, of power spectral density (PSD), inter-brain synchrony (IBS), and fuzzy entropy showing that Solo differed significantly from other conditions (Tong, 2024). The high accuracy for Hand could also be due to the sole presence of haptic feedback without visual input to confound the signal. Similarly, the Vision condition, the other condition where only one sensory modality is used, also achieved good classification accuracy (over 70% in the two best models).

Conversely, the TG and HP conditions were the least distinguishable and were frequently confused with each other and with RP. RP was also confused with HP in above 10% of cases in the XGBoost and StackNet models, although it achieved an overall higher accuracy of 73% compared to the other two interactive conditions. This is expected as these interactive conditions all involve multisensory integration of visual, proprioceptive, and haptic feedback, coupled with active movement required to complete the task. The fact that RP is also misclassified as HP can be justified by the design of this algorithm to be human-like (Takagi et al., 2017), highlighting the purpose of this partner to engage the participants' behavioural and neural activities in a similar manner as when interacting with a human. The confusion matrices (Fig. 3, Appendix Fig. 9) indeed corroborate the results of previous analysis on this dataset, that exhibited no significant differences between these interaction conditions with the investigated metrics (Tong, 2024).

The Passive condition presented moderate classification accuracy (below 70%) compared to the other relaxed control conditions, and was often misclassified as Vision or Hand, likely due to the shared passive nature of these conditions, and the design of these control conditions, where Passive includes both Vision (tracking the dot visually) and Hand (perceiving the haptic feedback and the wrist movement produced by the robot). The observable statistical differences in PSD between these three control conditions were also not numerous and focused on specific areas such as the occipital lobe (Tong, 2024). Interestingly, Passive was occasionally predicted as HP, possibly reflecting the strong activation of sensory feedback areas in both conditions despite the difference in volitional effort to be provided.

5. Conclusion

This study explored the potential of supervised machine learning algorithms to classify EEG data collected during a target-tracking task involving haptic interaction with a human or a robot, and various control conditions featuring specific sensory modalities. The results demonstrated that EEG data could contain sufficient information to distinguish between the seven experimental conditions, with ensemble models such as XGBoost and StackNet achieving the highest classification accuracies. When focusing on individual conditions, the high classification accuracy of Solo, Hand and Vision could indicate the presence of distinct neural signatures during tasks involving single sensory modalities or independent task execution. Conversely, the models presented difficulty in distinguishing interactive conditions like TG, HP, and RP, which could reflect the shared mechanisms in these conditions, where multisensory integration into the motion plan is required, as well as the human-like nature of the RP. These findings align with prior EEG analysis on this dataset (Tong, 2024), also showing clear differences between Solo and other conditions, but similar responses among interactive conditions. Overall, this study highlights the potential of machine learning in decoding complex neural data, which can complement an analysis of the neural mechanisms of haptic communication based on sensorimotor metrics (Tong, 2024).

Impact Statement

This work aimed at using supervised machine learning to provide insights in the neural processes underlying human-human and human-robot interaction, which are crucial for applications such as robot-assisted surgery, rehabilitation or intelligent prosthetics.

References

Aldini, S., Singh, A. K., Leong, D., Wang, Y.-K., Carmichael, M. G., Liu, D., and Lin, C.-T. Detection and Estimation of Cognitive Conflict During Physical Human–Robot Collaboration. *IEEE Transactions on Cognitive and Developmental Systems*, 15(2):959–968, 2023.

Alsuradi, H. and Eid, M. EEG-Based Machine Learning Models to Evaluate Haptic Delay: Should We Label Data Based on Self-Reporting or Physical Stimulation? *IEEE Transactions on Haptics*, 16(4):524–529, 2023.

Aspiotis, V., Peschos, D., Tzimourta, K. D., Tsipouras, M. G., Abosaleh, A. H. S., Antoniou, E., Giannakeas, N., Tzallas, A. T., and Glavas, E. Active touch classification using EEG signals. In *2021 6th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference*, pp. 1–5.

Beckers, N., Van Asseldonk, E. H. F., and Van Der Kooij, H. Haptic human–human interaction does not improve individual visuomotor adaptation. *Scientific Reports*, 10 (1):19902, 2020.

Bergstra, J., Bardenet, R., Bengio, Y., and Kégl, B. Algorithms for Hyper-Parameter Optimization. In *Advances in Neural Information Processing Systems*, volume 24, 2011.

Bilucaglia, M., Duma, G. M., Mento, G., Semenzato, L., and Tressoldi, P. E. Applying machine learning EEG signal classification to emotion-related brain anticipatory activity. 9(173), 2021.

Chen, T. and Guestrin, C. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016.

Cho, J.-H., Jeong, J.-H., Kim, M.-K., and Lee, S.-W. Towards Neurohaptics: Brain-Computer Interfaces for Decoding Intuitive Sense of Touch. In *International Winter Conference on Brain-Computer Interface*, pp. 1–5. IEEE, 2021.

Delorme, A. and Makeig, S. Eeglab: an open source toolbox for analysis of single-trial eeg dynamics including independent component analysis. *Journal of neuroscience methods*, 134(1):9–21, 2004.

Eldeeb, S., Weber, D., Ting, J., Demir, A., Erdogmus, D., and Akcakaya, M. EEG-based trial-by-trial texture classification during active touch. *Scientific Reports*, 10(1): 20755, 2020.

Ganesh, G., Takagi, A., Osu, R., Yoshioka, T., Kawato, M., and Burdet, E. Two is better than one: Physical interactions improve motor performance in humans. *Scientific Reports*, 4(1):1–7, 2014.

Ivanova, E., Carboni, G., Eden, J., Krüger, J., and Burdet, E. For Motion Assistance Humans Prefer to Rely on a Robot Rather Than on an Unpredictable Human. *IEEE Open Journal of Engineering in Medicine and Biology*, 1: 133–139, 2020.

Ivanova, E., Eden, J., Carboni, G., Krüger, J., and Burdet, E. Interaction with a reactive partner improves learning in contrast to passive guidance. *Scientific Reports*, 12(1): 15821, 2022.

Melendez-Calderon, A., Bagutti, L., Pedrono, B., and Burdet, E. Hi5: A versatile dual-wrist device to study human-human interaction and bimanual control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2578–2583, 2011.

Noccaro, A., Buscaglione, S., Eden, J., Xiaoxiao, C., Stefano, N. D., Pino, G. D., Burdet, E., and Formica, D. Robot-mediated asymmetric connection between humans can improve performance without increasing effort. *Authorea Preprints*, 2023.

Schaabova, H., Krajca, V., Sedlmajerova, V., Bukhtaieva, O., and Petranek, S. Supervised learning used in automatic EEG graphoelements classification. In *E-Health and Bioengineering Conference*, pp. 1–4, Iasi, Romania, 2015.

Stancin, I., Cifrek, M., and Jovic, A. A Review of EEG Signal Features and Their Application in Driver Drowsiness Detection Systems. *Sensors*, 21(11):3786, 2021.

Takagi, A., Ganesh, G., Yoshioka, T., Kawato, M., and Burdet, E. Physically interacting individuals estimate the partner's goal to enhance their movements. *Nature Human Behaviour*, 1(3):1–6, March 2017.

Tong, L. A humanoid partner can also cause social haptic interaction. Master's thesis, Imperial College of London, 2024. Unpublished master's thesis.

Xu, C., Zhang, C., Zhou, Y., Wang, Z., Lu, P., and He, B. Trust Recognition in Human-Robot Cooperation Using EEG. In *IEEE International Conference on Robotics and Automation*, pp. 7827–7833, 2024.

A. Appendix

A.1. Description of Experimental Conditions

Table 5: Summary of Control and Experimental Conditions

Condition Type	**Condition Name**	**Description**	**Modality**
4*Control	**Solo**	Participants trace the target on the screen with the handle, experiencing no haptic feedback (0 N/m stiffness) and movement is voluntary.	Visual, Voluntary Motion
	Passive TG	Participants relax their arm while the robot leads the movement. No voluntary movement, full haptic feedback from the robot is provided.	Somatosensory, Visual
	Vision	Participants receive only visual stimuli without any hand movement and the dot is moved to the target with eye tracking.	Visual
	Hand	Participants experience passive hand motion haptic similar to passive TG, without any visual stimuli.	Somatosensory
3*Experimental	**TG (Trajectory Guidance)**	Participants move the handle to follow a target on the screen, receiving both visual information and haptic feedback from resistance due to the stiffness (0.0498 N/m).	Visual, Somatosensory, Voluntary Motor
	HP (Human Partner)	Two participants receive haptic feedback from each other's movements, cooperating to control the cursor close to the target.	Visual, Somatosensory, Voluntary Motor
	RP (Robotic Partner)	Participants interact with a humanoid algorithmic robot that provides haptic feedback.	Visual, Somatosensory, Voluntary Motor

A.2. Classification Results

Table 6: Classification Results for All Models

Model	Accuracy	Metric	HP	Hand	Passive	RP	TG	Vision	Solo
SVM	0.40	Precision	0.30	0.62	0.31	0.24	0.31	0.45	0.64
		Recall	0.24	0.49	0.21	0.55	0.21	0.48	0.57
		F1-Score	0.27	0.55	0.25	0.34	0.25	0.46	0.61
SVM (tuned, PCA)	0.58	Precision	0.46	0.70	0.53	0.54	0.47	0.60	0.67
		Recall	0.50	0.73	0.52	0.54	0.41	0.51	0.77
		F1-Score	0.48	0.72	0.53	0.54	0.44	0.56	0.72
KNN	0.48	Precision	0.36	0.65	0.44	0.36	0.42	0.49	0.74
		Recall	0.57	0.57	0.43	0.45	0.41	0.38	0.50
		F1-Score	0.44	0.61	0.43	0.40	0.42	0.43	0.60
KNN (tuned, PCA)	0.50	Precision	0.39	0.64	0.41	0.43	0.45	0.47	0.72
		Recall	0.44	0.52	0.43	0.55	0.46	0.45	0.62
		F1-Score	0.41	0.57	0.42	0.48	0.45	0.46	0.67
XGBoost	0.71	Precision	0.57	0.76	0.69	0.66	0.70	0.77	0.77
		Recall	0.58	0.79	0.66	0.74	0.60	0.73	0.82
		F1-Score	0.57	0.77	0.68	0.70	0.65	0.75	0.80
XGBoost (tuned, PCA)	0.49	Precision	0.34	0.62	0.50	0.42	0.33	0.46	0.71
		Recall	0.34	0.60	0.50	0.44	0.31	0.48	0.69
		F1-Score	0.34	0.61	0.50	0.43	0.32	0.47	0.70
Stack RF	0.72	Precision	0.56	0.83	0.69	0.69	0.68	0.73	0.83
		Recall	0.58	0.82	0.69	0.73	0.64	0.73	0.81
		F1-Score	0.57	0.83	0.69	0.71	0.66	0.73	0.82
Stack LR	0.72	Precision	0.56	0.82	0.63	0.68	0.64	0.76	0.86
		Recall	0.59	0.82	0.66	0.74	0.61	0.73	0.81
		F1-Score	0.58	0.82	0.65	0.71	0.62	0.75	0.83
Stack XGB	0.71	Precision	0.57	0.80	0.61	0.64	0.69	0.74	0.84
		Recall	0.60	0.79	0.64	0.72	0.60	0.73	0.82
		F1-Score	0.59	0.80	0.63	0.68	0.64	0.73	0.83
Stack MLP	0.69	Precision	0.56	0.78	0.65	0.67	0.64	0.73	0.76
		Recall	0.49	0.81	0.61	0.75	0.59	0.73	0.83
		F1-Score	0.52	0.79	0.63	0.71	0.61	0.73	0.80
Stack GB	0.70	Precision	0.54	0.79	0.69	0.66	0.59	0.73	0.82
		Recall	0.55	0.83	0.63	0.73	0.53	0.73	0.83
		F1-Score	0.55	0.81	0.66	0.69	0.56	0.73	0.82

A.3. Tree-structured Parzen Estimator (TPE)

The Tree-structured Parzen Estimator (TPE) (Bergstra et al., 2011) from the HyperOpt library will be used for exploring the search space. TPE is a Sequential Model-Based Optimization (SMBO) algorithm that is suitable for high dimensional spaces and balances exploration and exploitation by intelligently selecting the next set of hyperparameters for evaluation. The objective function is the negative cross-validation accuracy and is represented as $y = f(x)$.

TPE then uses two Probability Density Functions (PDF), equation $l(x) = p(x \mid y < \gamma)$ which corresponds to the probability that configuration x is providing an objective value y which is less than γ and this is set as the good configurations. The second equation $g(x) = p(x \mid y \geq \gamma)$ corresponds to the probability that configuration x is providing an objective value greater than γ and this is set as the bad configurations. Using the two sets, the next set of configurations is selected by maximizing the ratio of $a(x) = l(x)/g(x)$. The densities $l(x)$ and $g(x)$ are estimated using Kernel Density Estimation (KDE) and the whole process is iteratively repeated for 50 evaluations.

A.4. Model Baseline Parameters

Table 7: SVM Baseline Parameters

Parameter	Range
C (Regularization)	1
Gamma γ	0.000315
Kernel	RBF

Table 8: kNN Baseline Parameters

Parameter	Range
k Neighbours	5
Voting Weights	Uniform
P	2 (Euclidean)

Table 9: XGBoost Baseline Parameters

Parameter	Value
Boosting Rounds (n_estimators)	100
Max Tree Depths	10
Learning Rate (η)	0.1
γ Regularization	1

A.5. Euclidean Formula

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{1}$$

A.6. XGBoost Formula

XGBoost uses the following formula as update rule to improve weak learners.

$$F(m) = F(m-1) + \eta \cdot \left(- \frac{\sum_{i=1}^{n} g_i}{\sum_{i=1}^{n} h_i + \lambda} \right) \tag{2}$$

where
η: Learning rate
$\sum_{i=1}^{n} g_i$: Summation of gradients over all data points.
$\sum_{i=1}^{n} h_i$: Summation of Hessians over all data points.
λ: Regularization parameter that penalizes the model complexity.

A.7. Additional Figures and Plots

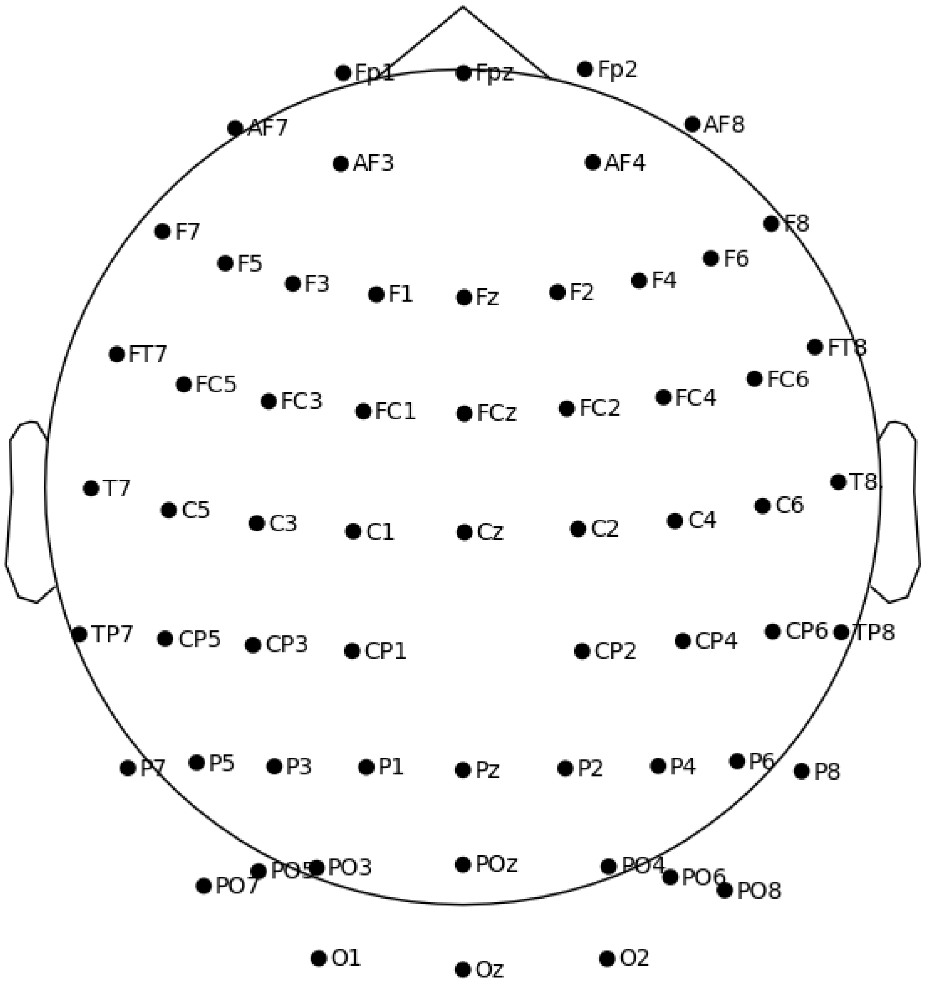


Figure 5: Scalp EEG Channels

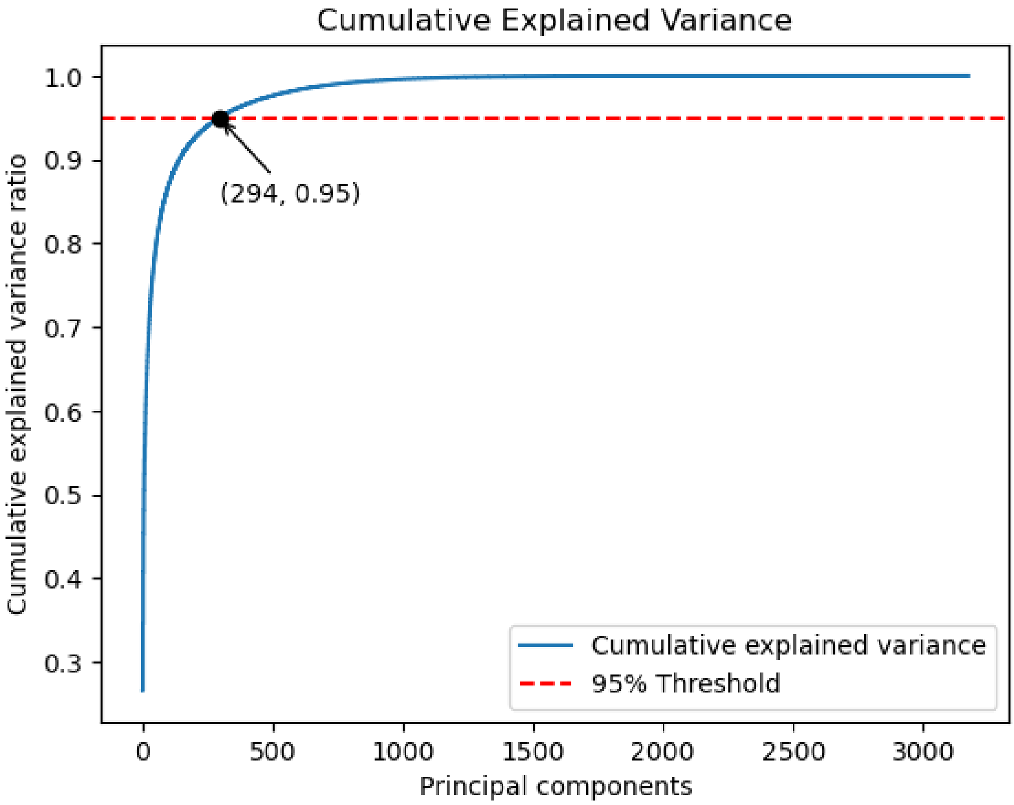


Figure 6: Cumulative Variance Plot

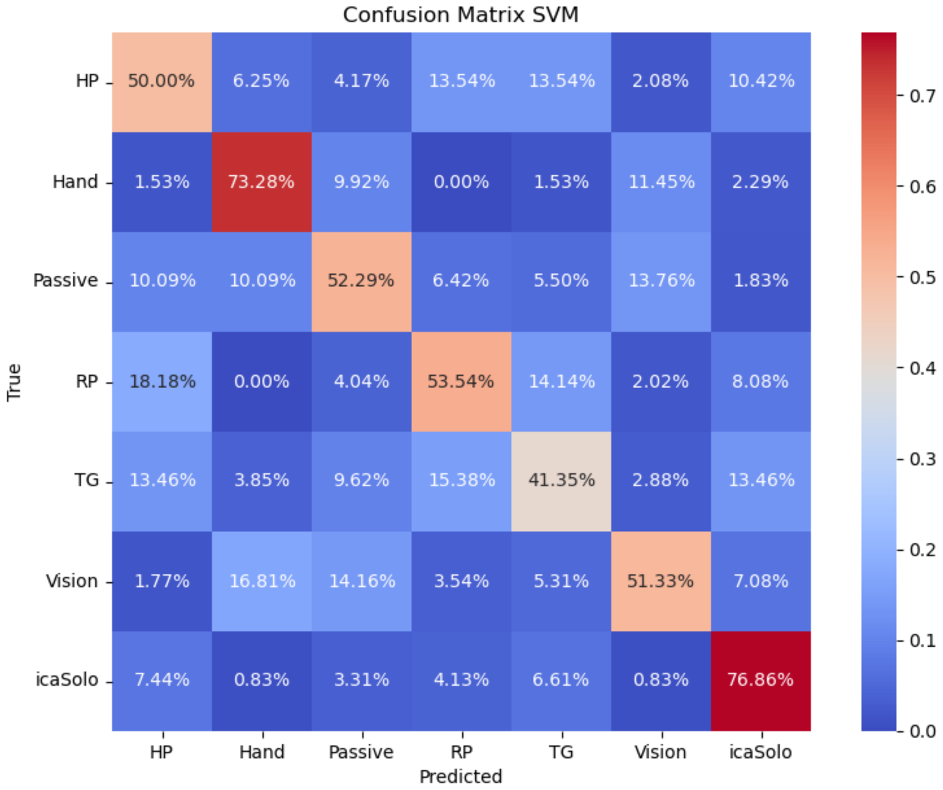


Figure 7: SVM Confusion Matrix

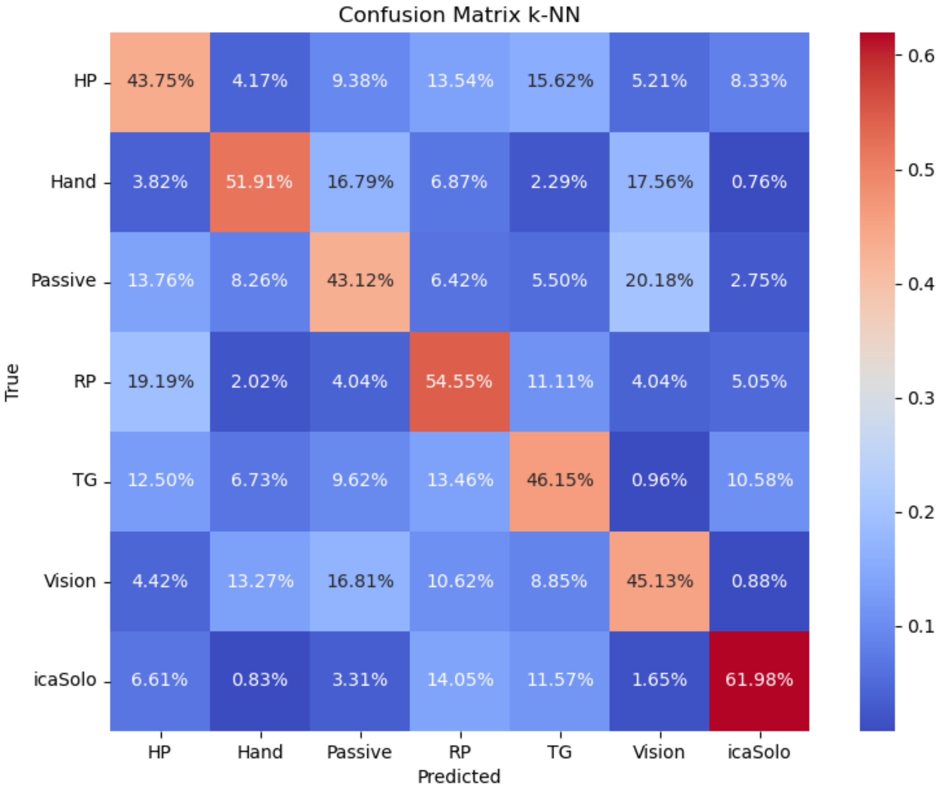
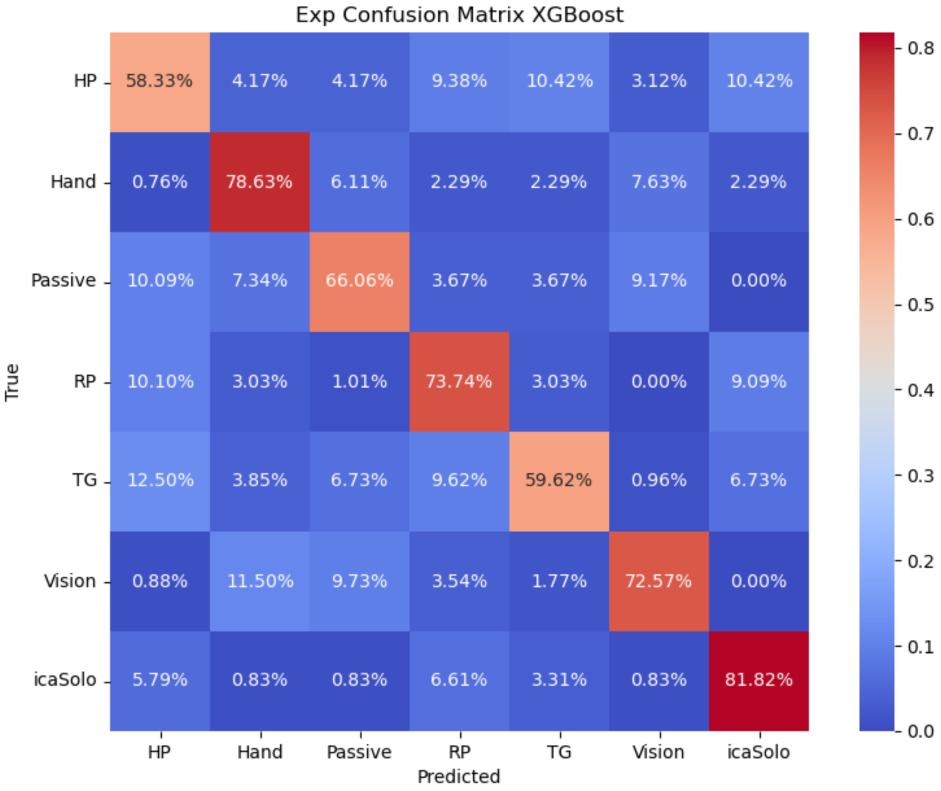


Figure 8: kNN Confusion Matrix



True \ Predicted	HP	Hand	Passive	RP	TG	Vision	icaSolo
HP	58.33%	4.17%	4.17%	9.38%	10.42%	3.12%	10.42%
Hand	0.76%	78.63%	6.11%	2.29%	2.29%	7.63%	2.29%
Passive	10.09%	7.34%	66.06%	3.67%	3.67%	9.17%	0.00%
RP	10.10%	3.03%	1.01%	73.74%	3.03%	0.00%	9.09%
TG	12.50%	3.85%	6.73%	9.62%	59.62%	0.96%	6.73%
Vision	0.88%	11.50%	9.73%	3.54%	1.77%	72.57%	0.00%
icaSolo	5.79%	0.83%	0.83%	6.61%	3.31%	0.83%	81.82%

Figure 9: Experimental XGB Confusion Matrix

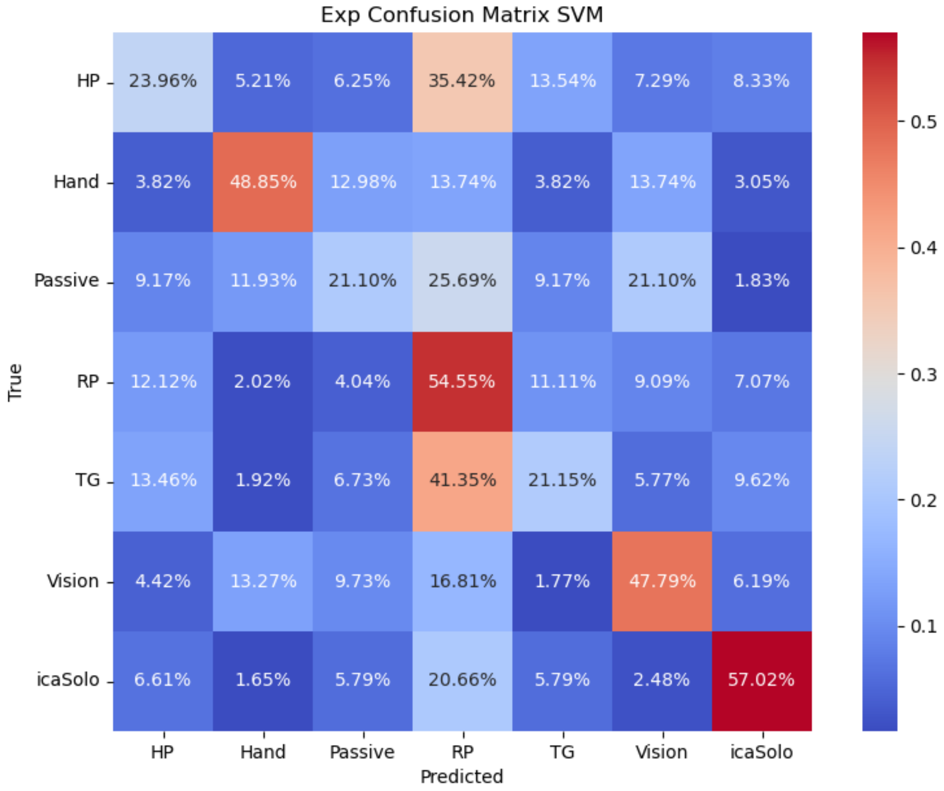
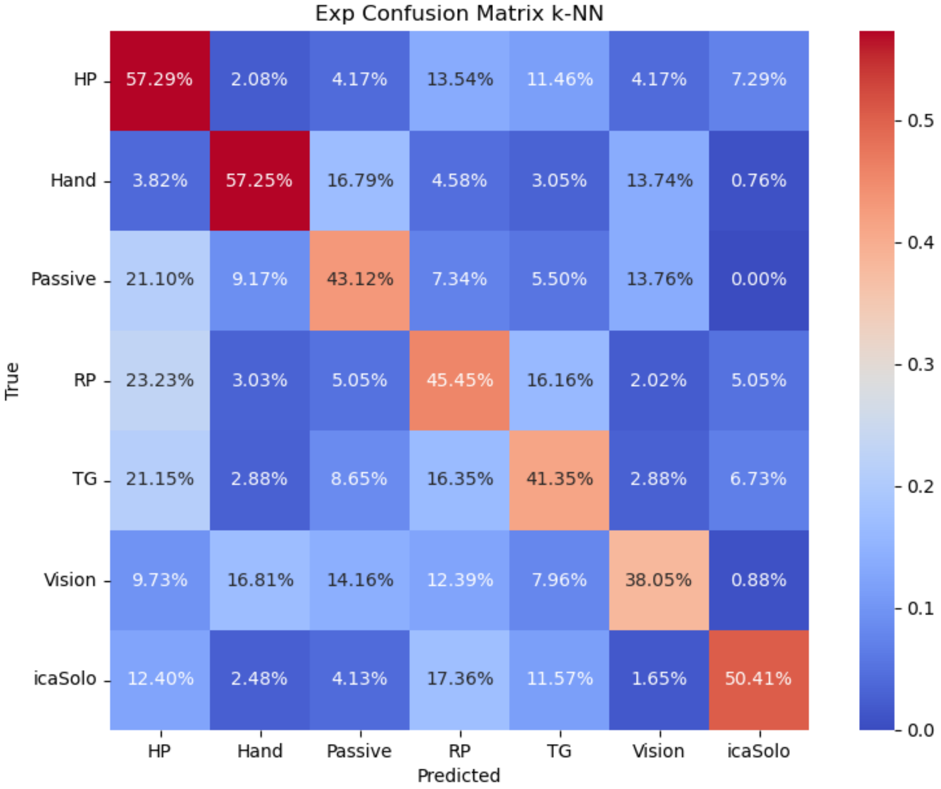
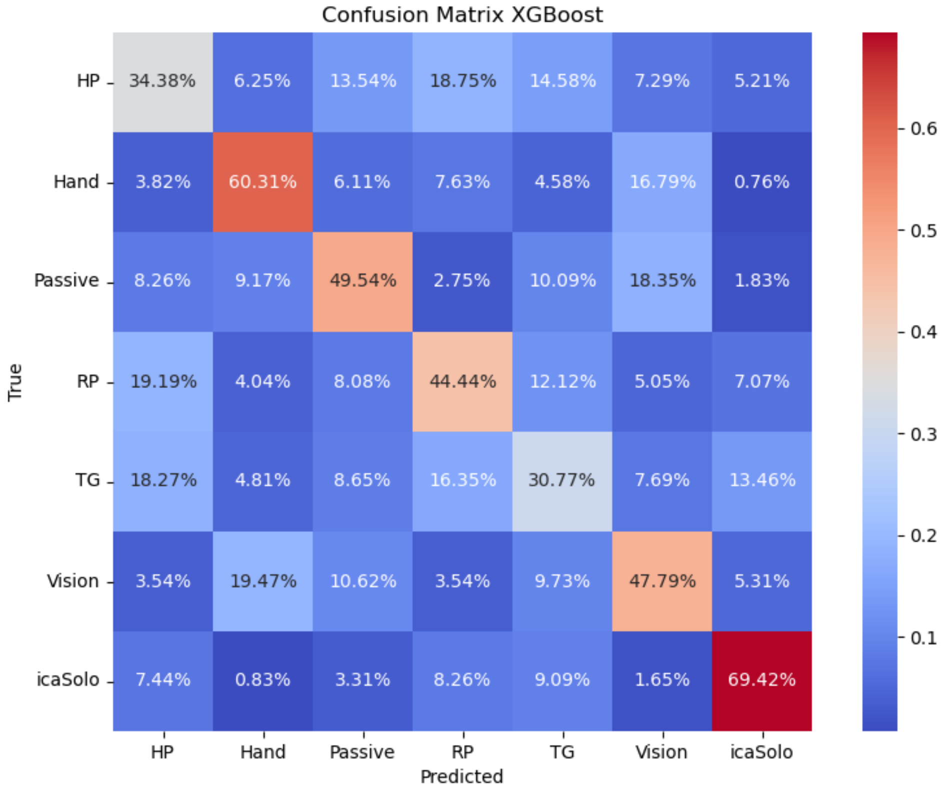


Figure 10: Experimental SVM Confusion Matrix



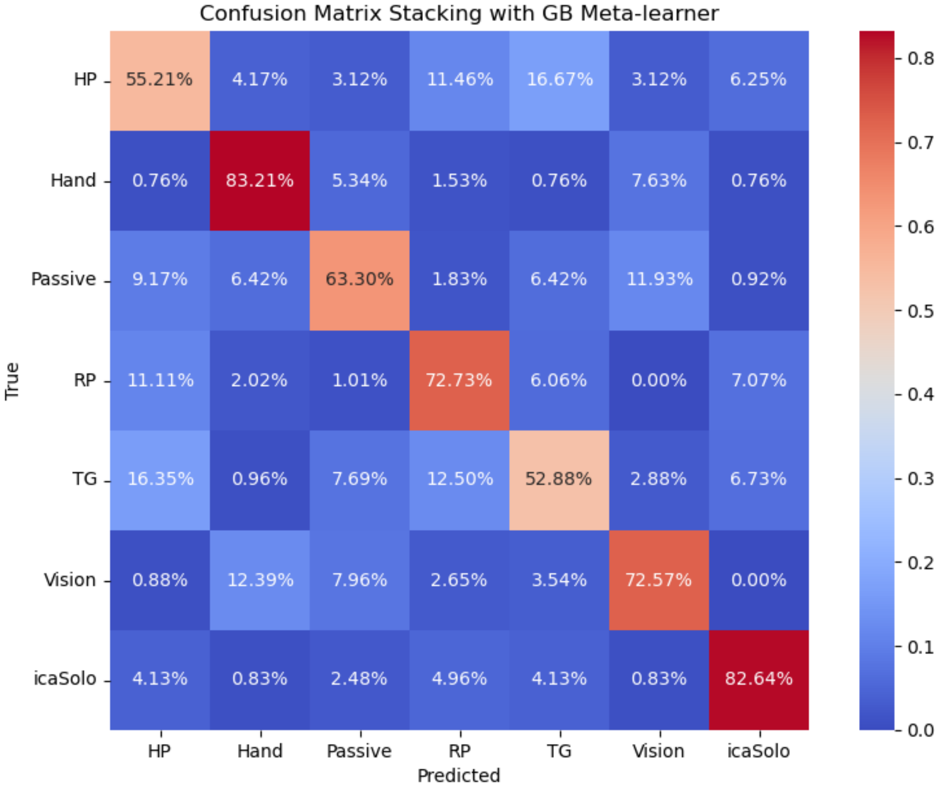
	HP	Hand	Passive	RP	TG	Vision	icaSolo
HP	57.29%	2.08%	4.17%	13.54%	11.46%	4.17%	7.29%
Hand	3.82%	57.25%	16.79%	4.58%	3.05%	13.74%	0.76%
Passive	21.10%	9.17%	43.12%	7.34%	5.50%	13.76%	0.00%
RP	23.23%	3.03%	5.05%	45.45%	16.16%	2.02%	5.05%
TG	21.15%	2.88%	8.65%	16.35%	41.35%	2.88%	6.73%
Vision	9.73%	16.81%	14.16%	12.39%	7.96%	38.05%	0.88%
icaSolo	12.40%	2.48%	4.13%	17.36%	11.57%	1.65%	50.41%

Figure 11: Experimental kNN Confusion Matrix



	HP	Hand	Passive	RP	TG	Vision	icaSolo
HP	34.38%	6.25%	13.54%	18.75%	14.58%	7.29%	5.21%
Hand	3.82%	60.31%	6.11%	7.63%	4.58%	16.79%	0.76%
Passive	8.26%	9.17%	49.54%	2.75%	10.09%	18.35%	1.83%
RP	19.19%	4.04%	8.08%	44.44%	12.12%	5.05%	7.07%
TG	18.27%	4.81%	8.65%	16.35%	30.77%	7.69%	13.46%
Vision	3.54%	19.47%	10.62%	3.54%	9.73%	47.79%	5.31%
icaSolo	7.44%	0.83%	3.31%	8.26%	9.09%	1.65%	69.42%

Figure 12: XGB Confusion Matrix



True \ Predicted	HP	Hand	Passive	RP	TG	Vision	icaSolo
HP	55.21%	4.17%	3.12%	11.46%	16.67%	3.12%	6.25%
Hand	0.76%	83.21%	5.34%	1.53%	0.76%	7.63%	0.76%
Passive	9.17%	6.42%	63.30%	1.83%	6.42%	11.93%	0.92%
RP	11.11%	2.02%	1.01%	72.73%	6.06%	0.00%	7.07%
TG	16.35%	0.96%	7.69%	12.50%	52.88%	2.88%	6.73%
Vision	0.88%	12.39%	7.96%	2.65%	3.54%	72.57%	0.00%
icaSolo	4.13%	0.83%	2.48%	4.96%	4.13%	0.83%	82.64%

Figure 13: StackNet GB Confusion Matrix

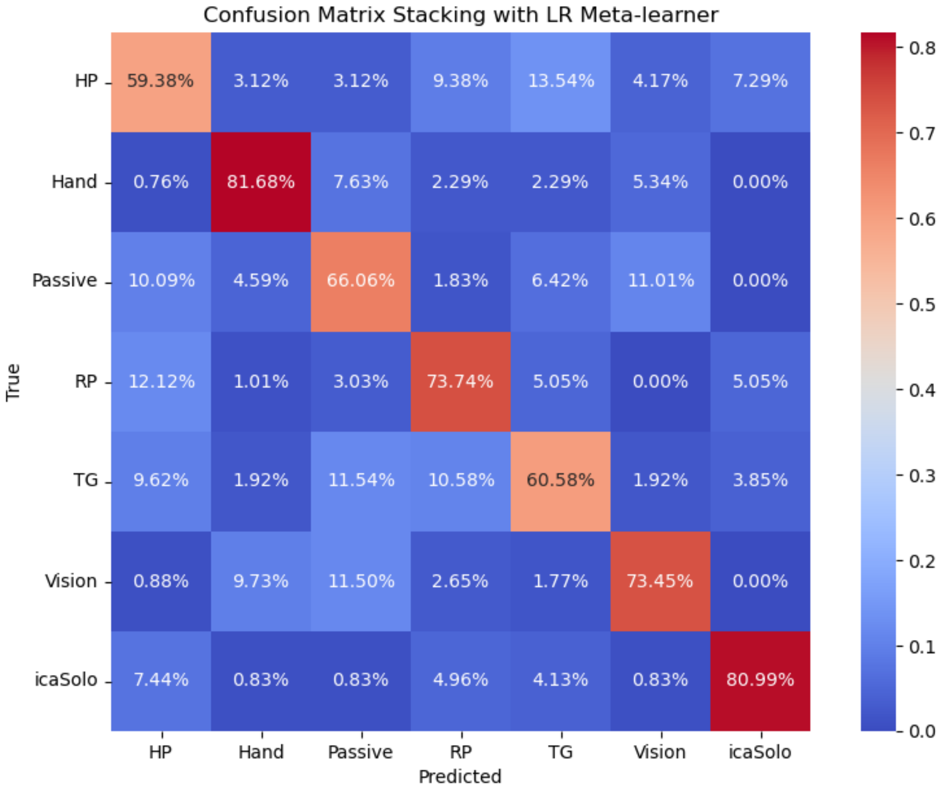


Figure 14: StackNet LR Confusion Matrix

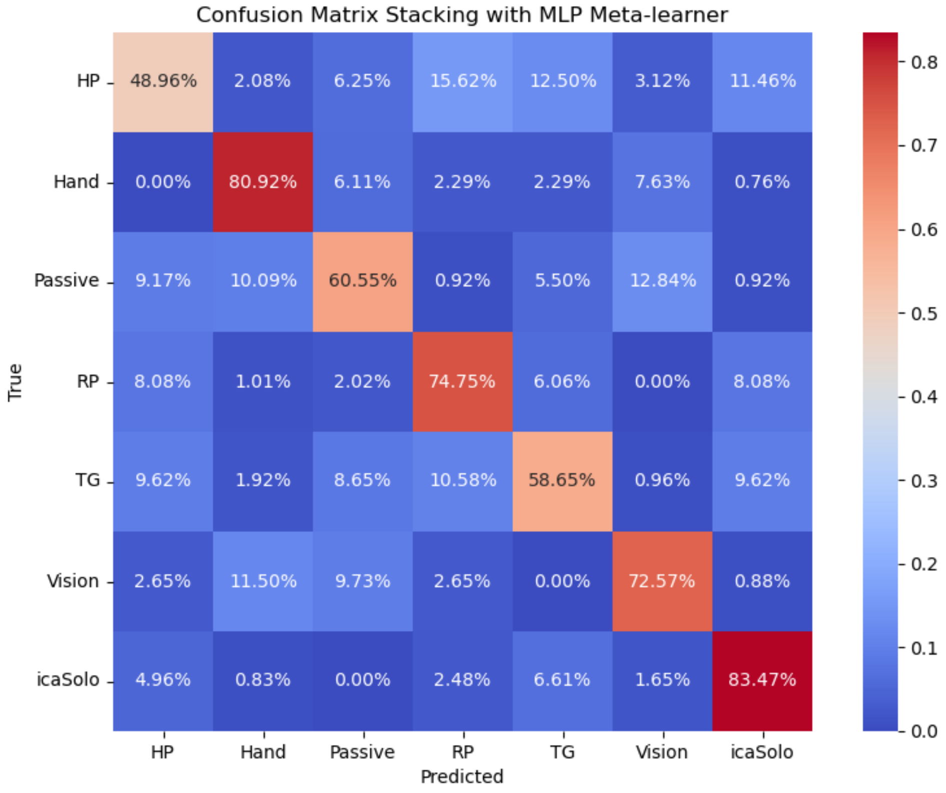


Figure 15: StackNet MLP Confusion Matrix

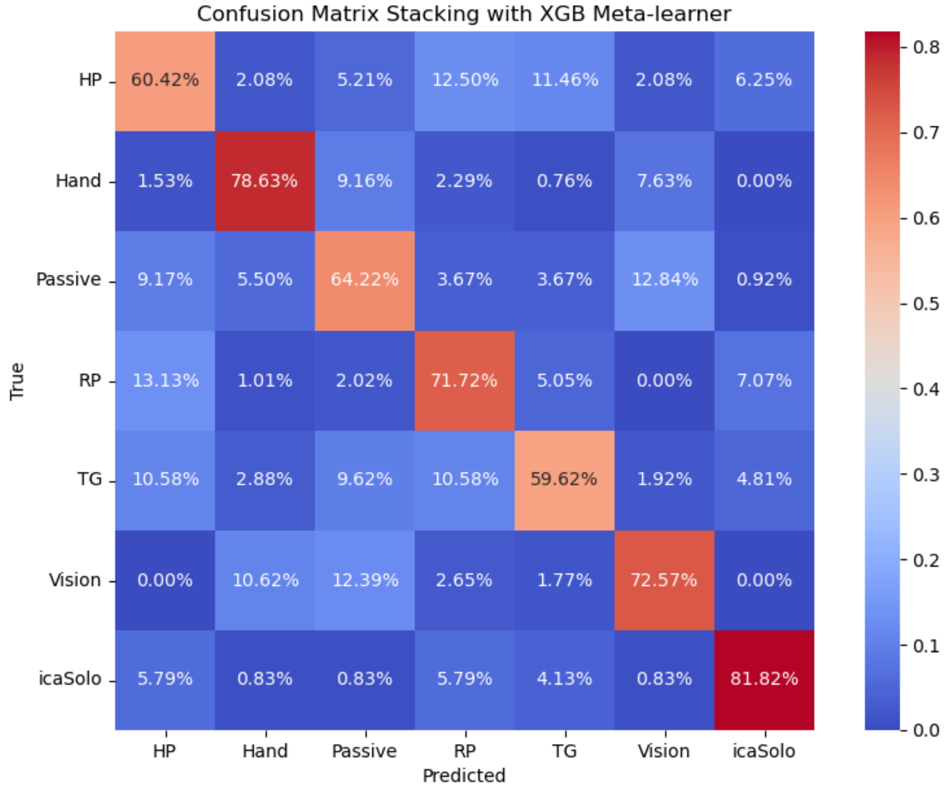


Figure 16: StackNet XGB Confusion Matrix

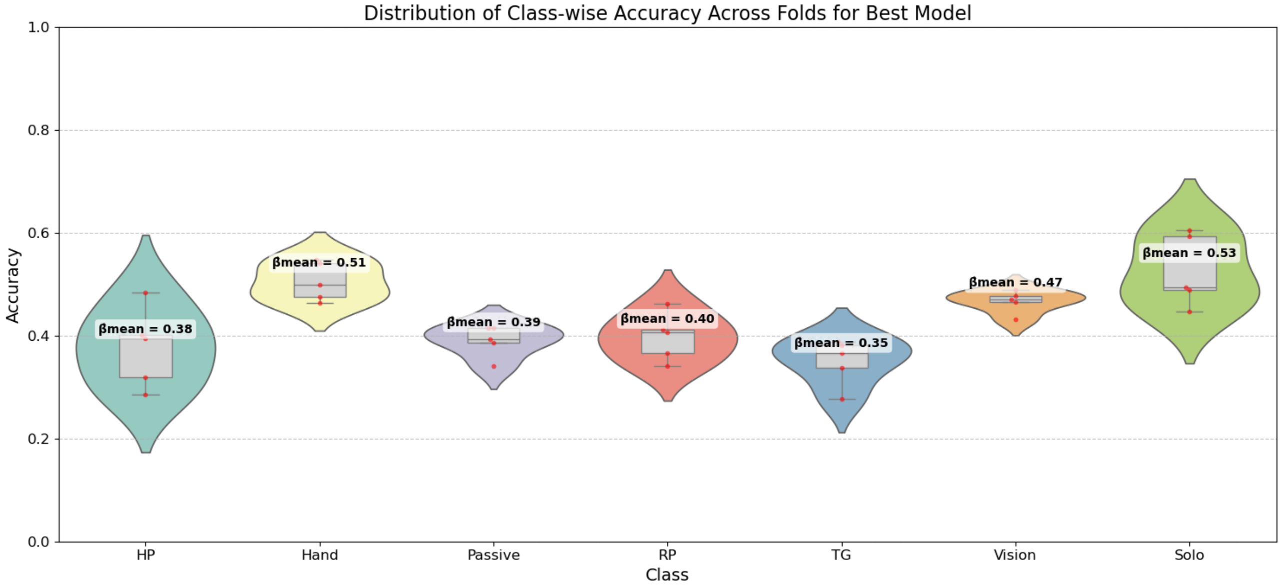


Figure 17: kNN Class-wise Accuracy

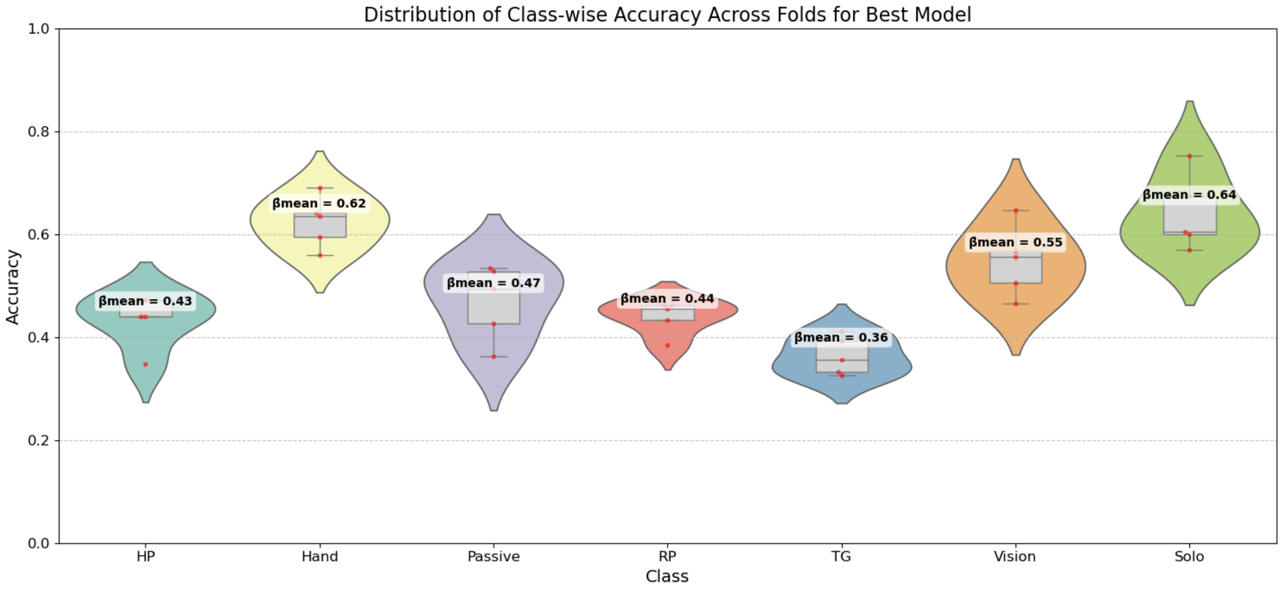


Figure 18: SVM Class-wise Accuracy

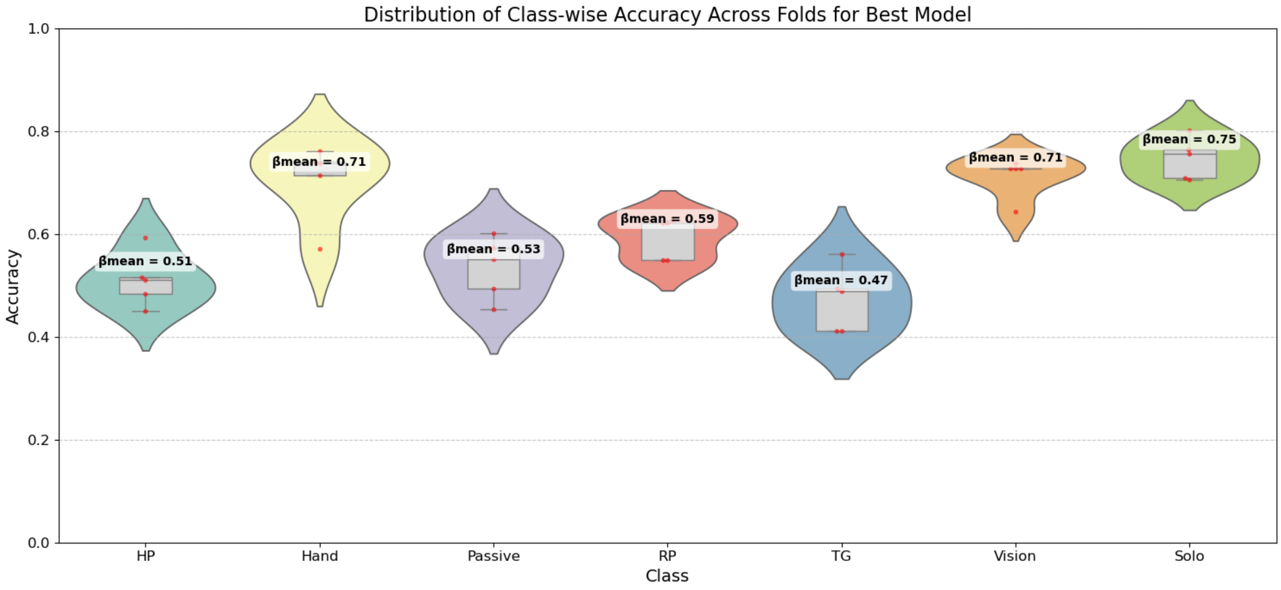


Figure 19: XGB Class-wise Accuracy

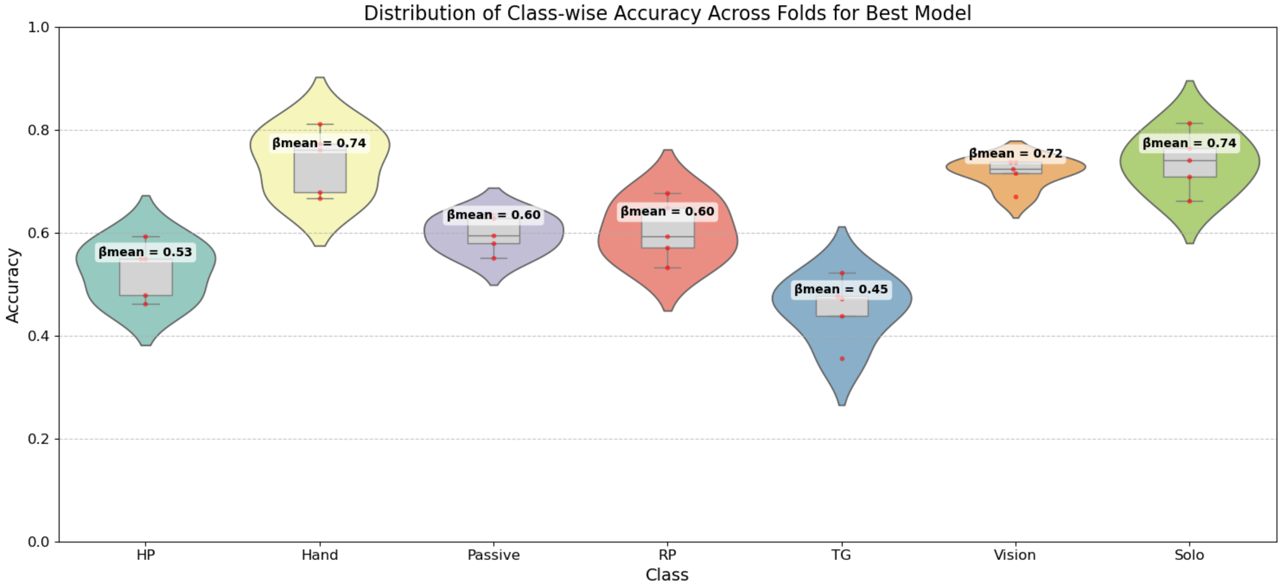


Figure 20: StackNet RF Class-wise Accuracy