2021 43rd Annual International Conference of the
IEEE Engineering in Medicine & Biology Society (EMBC)
Oct 31 - Nov 4, 2021. Virtual Conference

Data-efficient Causal Decoding of Spiking Neural Activity using Weighted Voting

Ali Marjaninejad[1], Christian Klaes[2], Francisco J. Valero-Cuevas[1]

Abstract— **Brain-Computer Interface systems can contribute to a vast set of applications such as overcoming physical disabilities in people with neural injuries or hands-free control of devices in healthy individuals. However, having systems that can accurately interpret intention online remains a challenge in this field. Robust and data-efficient decoding—despite the dynamical nature of cortical activity and causality requirements for physical function—is among the most important challenges that limit the widespread use of these devices for real-world applications. Here, we present a causal, data-efficient neural decoding pipeline that predicts intention by first classifying recordings in short sliding windows. Next, it performs weighted voting over initial predictions up to the current point in time to report a refined final prediction. We demonstrate its utility by classifying spiking neural activity collected from the human posterior parietal cortex for a cue, delay, imaginary motor task. This pipeline provides higher classification accuracy than state-of-the-art time windowed spiking activity based causal methods, and is robust to the choice of hyper-parameters.**

Clinical relevance— **We have tested our decoder during delayed imaginary grasp tasks on data from the human posterior parietal cortex—a relatively understudied region of the brain thought to contribute to motor intention. Our results provide new insight into the underlying neural dynamics of this region. In fact, the most discriminating information—and the greatest utility of voting—appear to occur during the early phases of the task. This makes our approach most useful to short-latency control of brain-computer systems such as neuroprosthetics.**

I. INTRODUCTION

Brain-Computer Interface (BCI) devices have the potential to effectively help with overcoming disabilities, especially spinal cord injuries [1]–[9], or to enhance healthy individuals with decoding nervous commands that can be used in different applications such as hands-free control [10]–[16]. There are, however, some major challenges to have BCI devices as reliable solutions for real-world problems. A successful BCI device to be used in real-world settings needs to provide accurate predictions in a robust and reliable fashion; that is, it needs to have high accuracy across different tasks and in the presence of noise and artifacts (such as external electromagnetic noise or motion artifacts) [10], [13], [17]. Additionally, ease of setting up and use are important factors for having widespread BCI applications. Moreover, due to the dynamical nature of the neural activity and challenges in collecting user/task specific neural data [1], [2], [7], data-efficient learning of the decoding algorithm is also a critical point in distinguishing a BCI device for real-world applications. Although some of these challenges can introduce inherent trade-offs (such as the signal quality vs. invasiveness), it is important for a successful BCI system to find ways to push the limits and provide accurate and reliable predictions even when facing some of these challenges.

Implanted sensors have seen much attention during recent years due to their ability in recording signals with high spatiotemporal resolution and high Signal-to-Noise Ratio (SNA) [1], [2]. Non-invasive methods might seem more interesting at first since they do not need surgery for sensor placements [13], [14], [17]; However, they face physical limitations such as low SNR and interference between signals recorded from individual neurons (low spacial resolution) [14]. They also have to deal with motion artifacts [14] and the recording headsets need to be set up on the user before each use. Therefore, implanted sensors are a great choice for an accurate and robust BCI system that offers great mobility.

Regardless of the recording method, collecting neural signals for a vast range of tasks is very challenging which calls for data-efficient learning-based decoders. Also, due to the highly dynamic nature of neural activities, parameters of the estimation model that is fitted to the data will need to be continually re-tuned. Although deep-learning methods that use a large number of model parameters have shown promising results in decoding offline datasets [18], [19], currently, these methods are less attractive for real-time applications with a large repertoire of tasks. The data-expensive and computationally heavy learning phase make them hard to be trained and run in real-time. Also, they are not able to efficiently generalize to a large number of tasks without having access to considerable amount of data for each case. Therefore, for general purpose online BCI applications it is of great importance that the decoder can be tuned in a data-efficient manner and can perform in a causal setting.

Here, we have proposed a data-efficient classification pipeline that performs weighted voting across predictions of different time windows of the recorded neural data and has shown its state-of-the-art accuracy in classifying spiking neural activity compared to not using the temporal information (neural activity in prior time windows). Using sliding windows, we first perform spike counting to transform spiking neural activity into feature vectors. Next, for features coming from each time window, we apply a classifier to perform classification corresponding to that time window. We

[1]A. Marjaninejad and F. J. Valero-Cuevas (corresponding author*) are the Department of Biomedical Engineering, University of Southern California, Los Angeles, CA 90089 USA marjanin@usc.edu, valero@usc.edu

[2]C. Klaes is with Department of Neurosurgery University hospital Knappschaftskrankenhaus, Bochum, 44892 Bochum, Germany christian.klaes@googlemail.com

978-1-7281-1178-0/21/$31.00 ©2021 IEEE

have studied two of the go to algorithms for the classifier: the K-Nearest Neighbor (KNN; model-free) [1] and the Poisson decoder (model-based) [4]. Next, we have performed weighted voting in a causal fashion (from the beginning of the recording to the current time) and have shown that utilizing this temporal information using the voting layer significantly increases the classification accuracy. We have tested our proposed algorithm on recordings from human Posterior Parietal Cortex (PPC), which is an area that has recently attracted more attention due to its aptitude in controlling neuroprosthetic devices and has exhibited promising results in getting early and accurate classifications during imagined motor tasks.

II. Methods

A. Data collection and experimental setup

To test our proposed pipeline, we have used data collected using a 96 channel UTAH array (Neuroport, Blackrock Microsystems, Salt Lake City, UT) placed on the left anterior intraparietal (AIP) area of the PPC of a male tetraplegic human participant (E.G.S.) reported on [1]. Recorded raw signals then used to perform spike sorting (using the first two principal components of the detected waveforms) and extract spiking activity. For each electrode, maximum of two units were distinguished (using manually selected cluster centroids) and all other signals were considered as noise (maximum of 192 units in total). In total, the data consists of 10 sessions with 50 trials in each session (10 per hand shape).

The task consisted of three phases. During the first phase, "Cue" (200 ms), one of the five symbols corresponding to the task were shown to E.G.S. These symbols were: Rock, Paper, Scissors, Spock, and Lizard. It was followed by the "Delay" phase (1000 ms) during which, no symbols were shown and E.G.S. was asked not to perform any tasks. Finally, during the "Response" phase (2300 ms), E.G.S. was asked to perform one of the five imaginary hand shapes corresponding to the symbols he saw during the "Cue" phase. For further details on the data-collection, pre-processing, and experiment details, please refer to [1].

B. Feature extraction & Classifiers

For each unit and time window, the number of the spiking activity for that window was selected and provided to the classifiers as the feature vector to perform the classification on. We have used a sliding window of 500 ms (width) with sliding steps of 100 ms over the length of the experiment. The classifier predictions for each sliding window were then sent to the voter to perform the final prediction (see Fig. 1). Alternatively, instead of incorporate temporal information by having sliding windows and voting over their corresponding predictions, we have also assessed the classification accuracy when the window width is increased as time passes (instead of the window sliding in time domain, the window extends from the beginning of the experiment to the time of prediction) which we refer to here as the cumulative window (see Fig. 2). In this case, all of the temporal information for each unit is smashed into one single feature (spike count number) and no-voting layer was used for this case.

We have used two of the commonly used classifiers in the state-of-the-art online BCI studies for implanted electrodes: KNN decoder (model free) [1] and Poisson decoder (model based) [4]. For either of the classifiers, we use an 80% train, 20% test split.

a) KNN classifier: Feature vectors of length 192 (total number of spiking for each unit in that time window) were fed into the classifier and its class was predicted by its proximity (Euclidean distance) to the samples in the training data. Similar to [1], we have used a KNN classifier with K = 4. In an event of a tie, the closest neighbor's vote was used as the tiebreaker.

b) Poisson classifier: For each unit and time window, average spiking across the feature vectors (of length 192) of training data were used as the lambda (λ) parameter (expected number of spikes on that unit and during that specific time window) for the underlying Poisson distribution. The probability of having class C_k for the trained Poisson naive Bayes classifier is defined as:

$$p(C_k|x) = \frac{p(C_k)p(x|C_k)}{p(x)} \tag{1}$$

where p(C_k) is the prior probability of class C_k and p(x) is the probability of the evidence (the feature vector). Since the prior probability of each class is equal in our case, we can rewrite (1) as:

$$p(C_k|x) = \alpha \prod_{i=1}^{192} p(x_i|C_k) \tag{2}$$

where α is a constant equal to $1/p(x)$ and i is the unit index. Based on (2) and by applying the Maximum A Posteriori (MAP) decision rule, the predicted class ($\hat{C_p}$) is calculated as:

$$\hat{C_p} = argmax_{k\in[1,5]} \prod_{i=1}^{192} p(x_i|C_k) \tag{3}$$

Please note that since the prior probability of each class is the same, MAP estimation will transform into Maximum Likelihood Estimation (MLE).

C. Voting layer

Once we have the prediction for each time window up to the current time, we can look back and create a weighted polling between the prediction of each sliding window to determine and report the final predicted class. As seen on Fig. 1, we multiply outputs of the classifiers for each sliding window by their corresponding weight and send them to the voter. The voter will then report the most frequent (after incorporating weights) prediction as the final prediction. Please note that in the case of all weights being equal, the voting layer will be equivalent to a causal majority voter.

The next question is how to select these weights and how they will affect the final prediction. To study this problem, we

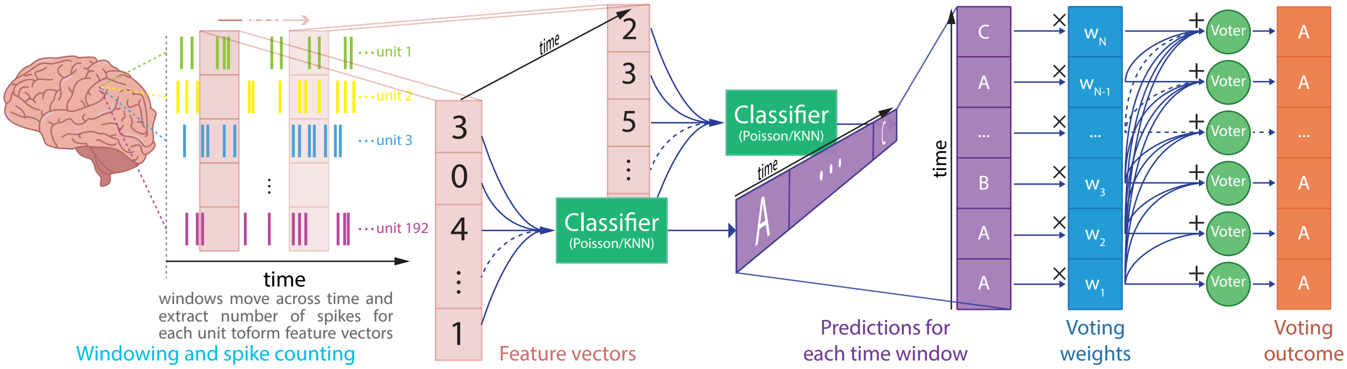

Fig. 1. The entire decoding pipeline with the spike counting using sliding windows on the left, classifier on the center and the voting layer on the right.

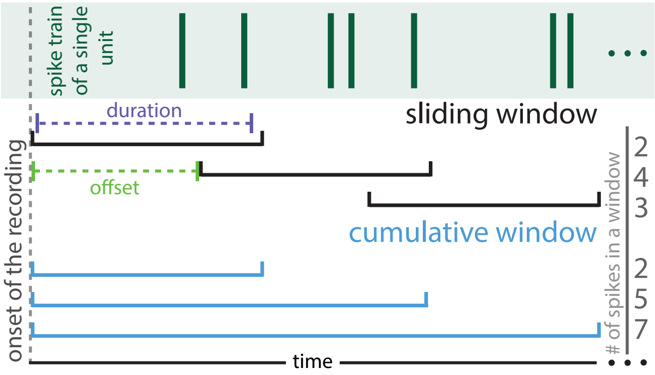

Fig. 2. Spike counting using the sliding window and cumulative window methods.

have selected a number of weight sets (Fig. 3) and reported the results for each. The studied weights on this paper are as follows:

a) Uniform: In this case, all weights are equal which turns the proposed weighted voting into a simple causal majority voter.

b) Ramp function: In this weight set, there is a linear relationship between weight value and its time index (discounted in time as we move toward more prior samples). An offset was also added to all weight values to make sure weights do not start from zero and therefore smoothens the differences in their ratios.

c) Gaussian curve: This weight set is very similar to the ramp function in implementation and rationale, however, instead of a linear function, here we have used the Gaussian distribution curve to form the weight.

d) Overall accuracy-based weights: The idea behind this weight set is to have higher weights for the areas that are expected to provide more accurate predictions. Therefore, we have performed an offline classification across all data and calculated the overall accuracy. This accuracy will be an indicator of how trustable each time window will be in informing the real intended hand gesture (class). Therefore, the overall accuracy curve can be a great candidate for being used as the voting-weights. Please note that since this pattern is relatively consistent across all experiments, there is no need for recalculating this pattern for each recording which will introduce anticausality. Simply having access to a training dataset or utilizing prior information about the expected accuracy vs. time pattern would be enough for a reasonable assumption of the overall accuracy-based weights.

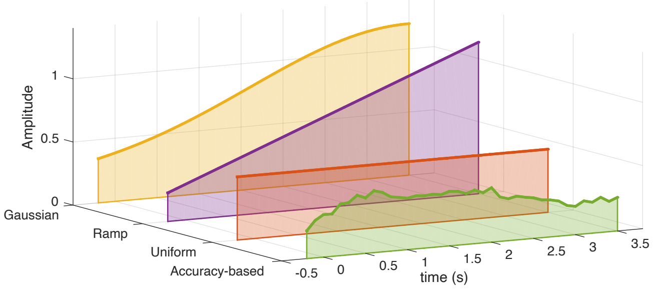

Fig. 3. Different voting weight sets studied in this paper.

III. Results

Results using either classifier (KNN and Poisson) were similar and generally followed the same patterns. Therefore, here we have mainly reported the results from the pipeline using one of them (KNN) and provided a complete report of results for both classifiers in the supplementary information (available at `https://github.com/marjanin/weighted_voting_BCI`).

Fig. 4a shows the performance of the weighted voting-based classifier averaged across all sessions for all studied weight sets (color coded). Also, we have applied one-way ANOVA analysis on average (over time) accuracy values of all sessions across the weight sets which shows there is no significant difference between outcomes of different weight sets. This result is encouraging in that it shows minimal dependence of the proposed method to the selection of weight. However, what really matters is the comparison of the proposed method versus the without voting (and especially with the sliding window case which does not incorporate temporal information from previous time spans).

Fig. 4b shows the performance of the weighted voting classifier (with accuracy-based weights) in green and predictions without the proposed voting layer (sliding window results), in dark blue. As can be seen on Fig. 4b, accuracy curves have a very similar start but as we go forward in time, the accuracy of the no-voting curve starts to decline whereas the accuracy of the proposed voting-based decoder keeps climbing. We start seeing this divergence between the two curves starting around 500 ms.

As discussed in the methods section, to have an alternative approach on utilizing temporal information across the entire trial, we have also introduced the idea of the cumulative window where we use a single window that extends from the beginning of the experiment to the time of prediction and reports the cumulative

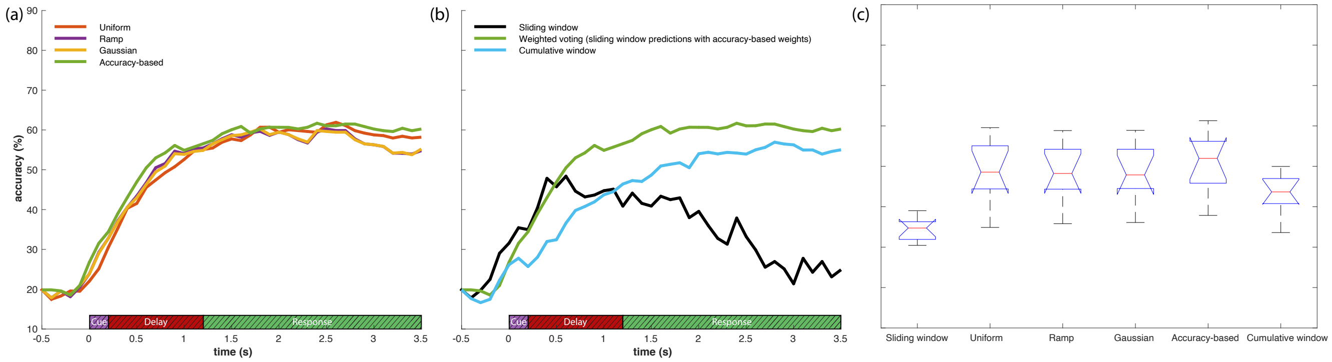

Fig. 4. (a) Accuracy for the entire experiment across different voting weight sets. (b) Accuracy for the entire experiment across no-voting (sliding window), cumulative window, and voting-based (with accuracy-based weights). (c) Box plots of mean accuracy for all of the curves shown on parts a and b across all recording sessions.

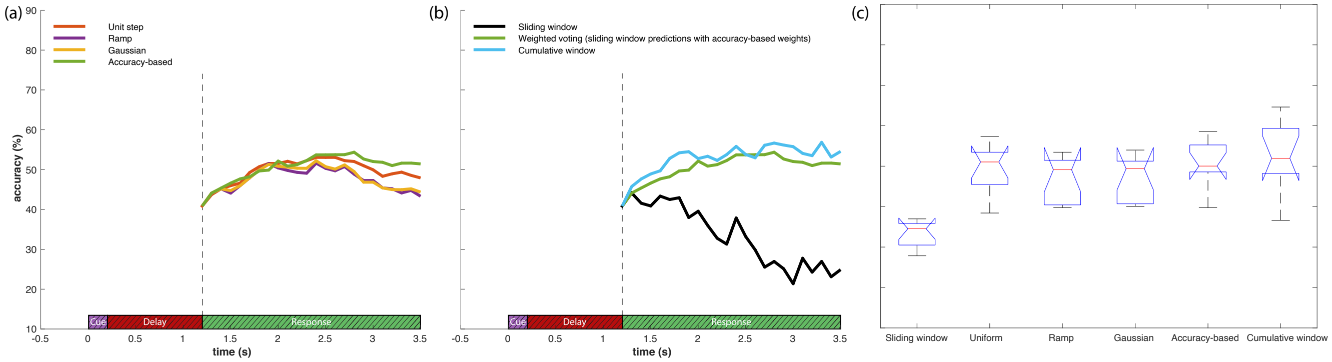

Fig. 5. (a) Accuracy for the response phase across different voting weight sets. (b) Accuracy for the response phase across no-voting (sliding window), cumulative window, and voting-based (with accuracy-based weights). (c) Box plots of mean accuracy for all of the curves shown on parts a and b across all recording sessions.

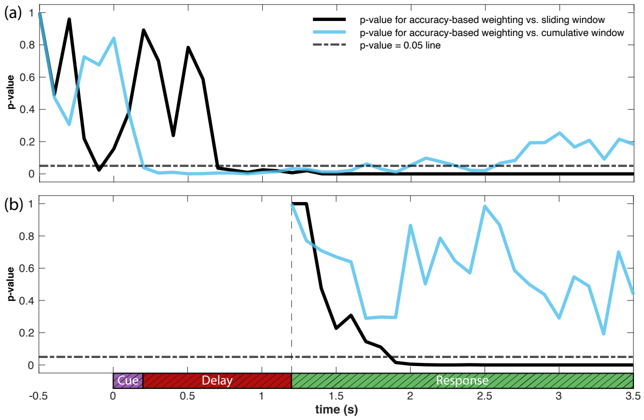

Fig. 6. p-value curves of the one-way ANOVA analysis across sessions for each time-point between voting-based (accuracy-based weighting) and no-voting (sliding window) accuracy values and between voting-based (accuracy-based weighting) and cumulative window accuracy values for the entire trial (a) and the response phase only (b).

number of spikes to the classifier (smashing all spikes across the time domain). Since in this approach, time history is accounted for by extending the window, we will not have a voting layer on this approach. The accuracy curve of this approach is also shown on Fig. 4b in light blue. As you can see, in general, the voting-based approach still shows superior performance compare to this approach. Nevertheless, we were surprised by the high accuracy of this approach since by smashing all temporal information into a single feature (cumulative number of spikes), this method cannot entertain any temporal distinctive aspect of the data.

Fig. 4c shows box plots of mean accuracy for all of the curves shown on Fig. 4a and Fig. 4b. across all sessions. In addition, we have performed One-way ANOVA analysis of these mean accuracy values across sessions. We found statistical significance ($p < 0.05$) between all proposed approaches that utilize past history versus the no-voting sliding window approach with accuracy-based weighing being the most significant one (please see supplementary information for more details and the interactive multiple comparison of means user interface). Also, the time discounted weights show relatively lower average accuracy values compared to the other two toward the end of the experiment.

It is important to note that in Fig. 4 results, both voting based and cumulative window methods utilize temporal information by having access to all different phases of the experiment (cue, delay, and imagined motor response); this means that the decoder can potentially utilize information in earlier phases (such as the visual information) to form the final prediction for the imaginary motor response. For real-world applications, however, it is also important to have a high accuracy decoder when only having access to the motor phase (since voluntary movements are not necessarily preceded by a visual cue). For this reason, we re-ran analysis reported on Fig. 4 on only the imagined motor response phase (1200ms - 3500ms of the experiment). This means that the data recorded in the cue and delay phases (0 ms - 1200 ms) had no contribution in the predictions. The results are reported on Fig. 5.

Similar to the results shown on Fig. 4, Fig. 5 shows that the effects of changing the voting weights are relatively insignificant (with accuracy-based and uniform weights still having higher performance toward the end compared to the time discounted ones; Fig. 5a) and both methods utilizing time history (voting-based and cumulative window) outperform the sliding window method (Fig. 5b). Box plots of mean accuracy for all of the curves shown on Fig. 5a and Fig. 5b. across all sessions are also shown on Fig. 5c).

Furthermore, we also performed One-Way ANOVA analysis for accuracy values across sessions for each time-point between voting-based (accuracy-based weights) and no-voting sliding window accuracy values and also between voting-based (accuracy-based weights) and cumulative window accuracy values. The results of these analyses for the entire experiment (corresponding to Fig. 4b curves) and the response phase only (corresponding to Fig. 5b curves) are shown on Fig. 6a and Fig. 6b, respectively. Fig. 6a shows that the accuracy of the weighted voting-based method (with accuracy-based weights) is significantly higher (statistically significant with $p < 0.05$) than the traditional no-voting based sliding window method (starting around 700 ms till the end of the experiment; black line) and than the cumulative window method (starting as early as the end of the cue phase–200 ms–but becoming less significant towards the end of the experiment–after around 2500 ms; blue line). Fig. 6b also shows that the differences between the voting-based and sliding window methods are significant (after early predictions: once we get more voters involved; black line). In this case, both approaches that utilize time history (voting-based and cumulative window; blue line) are resulting in similarly high accuracies (not a statistical significance in the deference).

IV. Discussion

We present a causal, data-efficient and accurate spiking neural decoder for BCI that utilizes weighted voting across prior time windows to estimate motor intention. We demonstrate it has better performance compared to classifiers that do not include long-term time history (longer than the window of interest) information in current predictions (e.g., [1] and [4]). By incorporating time history (both voting-based sliding windows and cumulative window approaches), as opposed to analyzing each time window independently, we show significant improvement in prediction accuracy (see Fig. 4, Fig. 5, and Fig. 6).

The modular nature of our pipeline enables us to inherent the advantages of the classifier of choice. In particular, we used two popular classifiers (KNN and Poisson) that are very data-efficient compared to some other alternatives such as data-heavy deep learning models [20]. That is why, for each session, we were able to provide high accuracy with around only 140 seconds of training recordings (80% of the available 175 seconds of data from fifty trials each lasting 3500 ms). Moreover, since the voting layer does not require an additional learning/training phase, the proposed voting based method inherits the data-efficiency of the classifier.

Computational cost is also an important factor for practical real-time BCI applications, and the weighted voting layer adds minimal additional computational overhead. Specifically, for each class, N sliding windows add N multiplications, $N(N+1)/2$ summations, and one categorical argmax operation (see Fig. 1).

Another important aspect of our BCI pipeline is its relative insensitivity to the specifics of the classifier or voting weights. We implemented two commonly used classifiers (KNN and Poisson) and four different sets of voting weights, and demonstrated high performance in all cases. Moreover, although we reported the results for a 500 ms window width, we also used other window widths between 400 and 1000 ms and did not see major changes in performance. Similarly, in the KNN algorithm we used k values between three and six without observing major changes in performance. Note that this sensitivity analysis was made possible by the modularity of our pipeline that also enables easy switching between alternative feature extraction methods, classifiers, and voting weights.

A high rate of improvement of prediction accuracy (reaching to high accuracy in early phases) is the goal of on-line BCI applications such as neuroprosthetic control to reduce the decision-to-action delays. Importantly, the weighted voting classifier is able to make accurate predictions early on in both the entire trial (Fig. 4b, Fig. 6a) and in the response phase only (Fig. 5b, Fig. 6b). Also note that at -500 ms, when the decoding window does not overlap with the experiment at all, as expected, all accuracy curves start from around 20% which is the chance level (five classes with equal priors).

A potentially surprising finding was the high accuracy of the cumulative window approach (i.e., where the classifier compares windows of increasing length, all sharing the same initial time point), as one would expect it to wash-out important distinctive temporal information. This is likely because most of the discriminative information is available early in the task (see Fig. 4b and Fig. 5b where the accuracy curves start rising early on during the trial and start going down for the sliding window method and flatten for the other two around 2 seconds). Moreover, the cumulative window had a relatively worst performance on the entire trial (Fig. 4b-c) compared to the response phase only (Fig. 5b-c). One potential explanation for this drop in the performance can be the changes in the dynamics of the neural activity across different phases (Cue, Delay, and the Response). Some of the distinctive temporal features specific to each phase might have been washed out using the cumulative bin approach, however, this was not significant enough to substantially impact the decoding process since it was still able to provide reasonable accuracy (Fig. 4b-c). We interpret these observations as a possible consequence of the brain region from which we recorded. In fact, these results can be supporting evidence of the role of PPC in generating motor intention—as opposed to other areas such as premotor and motor cortices (e.g., PMd, M1) [1], [2], [7], [21]. It is therefore of interest to repeat this analysis with signals from those other areas to further reveal the distinct roles different cortical areas play in the generation of motor intention vs. motor planning vs. motor execution.

Another potentially fruitful follow-up study would be to assess the performance of this pipeline in experiments where the participant changes their motor intention midway through the task. We would expect that the sliding window approach *without* voting would be the least disrupted in case of a change in the intention midway in the trial and the quickest to adapt as it is the least dependent on prior temporal information (although it might still have lower accuracy values before the change in the intention compare to the other methods incorporating temporal information). Cumulative method might be the most susceptible to error or delayed adaptation in this case since it will neither throw out or reduce the weight of the prior information (before the change in the decision). We hypothesize that the weighted voting method using weight sets that prioritize the most recent time windows might have better performance in these cases compared to the cumulative window or other weight sets, however, a more complete evaluation needs to be able to fully assess performance of each approach, which is beyond the scope of this study and can be a good avenue for future research.

V. Conclusion

Supported by the results reported here, we believe that the proposed data-efficient, causal spiking neural decoding pipeline can greatly improve the state-of-the-art performance in BCI algorithms for neuroprosthetic control by increasing reliability early on, and decreasing delays in the decoding process.

SUPPLEMENTARY INFORMATION

The code and the supplementary files can be accessed through project's Github repository at:
`https://github.com/marjanin/weighted_voting_BCI`

ACKNOWLEDGMENT

We would like to thank Andersen Lab at California Institute of Technology for providing the data. We would also like to thank Dr. Richard A. Andersen and Dr. Spencer Kellis for their guidance and feedback during the early phases of this work. Research reported in this publication was supported in part by the Department of Defense CDMRP Grant MR150091, and Award W911NF1820264 from the DARPA-L2M program, as well as National Institutes of Health under the award number R21-NS113613 to F.J.V.-C. This study was also partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under project number KL 2990/1-1—Emmy Noether Program. This work does not necessarily represent the views of the funding agencies.

REFERENCES

[1] C. Klaes, S. Kellis, T. Aflalo, B. Lee, K. Pejsa, K. Shanfield, S. Hayes-Jackson, M. Aisen, C. Heck, C. Liu, *et al.*, "Hand shape representations in the human posterior parietal cortex," *Journal of Neuroscience*, vol. 35, no. 46, pp. 15 466–15 476, 2015.

[2] T. Aflalo, S. Kellis, C. Klaes, B. Lee, Y. Shi, K. Pejsa, K. Shanfield, S. Hayes-Jackson, M. Aisen, C. Heck, *et al.*, "Decoding motor imagery from the posterior parietal cortex of a tetraplegic human," *Science*, vol. 348, no. 6237, pp. 906–910, 2015.

[3] R. A. Andersen, T. Aflalo, and S. Kellis, "From thought to action: The brain–machine interface in posterior parietal cortex," *Proceedings of the National Academy of Sciences*, vol. 116, no. 52, pp. 26 274–26 279, 2019.

[4] M. M. Shanechi, R. C. Hu, M. Powers, G. W. Wornell, E. N. Brown, and Z. M. Williams, "Neural population partitioning and a concurrent brain-machine interface for sequential motor function," *Nature neuroscience*, vol. 15, no. 12, pp. 1715–1722, 2012.

[5] M. M. Shanechi, *Brain–Machine Interfaces*. Cham: Springer International Publishing, 2018, pp. 197–218. [Online]. Available: https://doi.org/10.1007/978-3-319-71976-4_8

[6] ——, "Brain–machine interfaces from motor to mood," *Nature neuroscience*, vol. 22, no. 10, pp. 1554–1564, 2019.

[7] G. Santhanam, S. I. Ryu, M. Y. Byron, A. Afshar, and K. V. Shenoy, "A high-performance brain–computer interface," *nature*, vol. 442, no. 7099, pp. 195–198, 2006.

[8] G. H. Wilson, S. D. Stavisky, F. R. Willett, D. T. Avansino, J. N. Kelemen, L. R. Hochberg, J. M. Henderson, S. Druckmann, and K. V. Shenoy, "Decoding spoken english from intracortical electrode arrays in dorsal precentral gyrus," *Journal of Neural Engineering*, vol. 17, no. 6, p. 066007, 2020.

[9] S. T. Albert, A. M. Hadjiosif, J. Jang, A. J. Zimnik, D. S. Soteropoulos, S. N. Baker, M. M. Churchland, J. W. Krakauer, and R. Shadmehr, "Postural control of arm and fingers through integration of movement commands," *Elife*, vol. 9, p. e52507, 2020.

[10] A. Marjaninejad, B. Taherian, and F. J. Valero-Cuevas, "Finger movements are mainly represented by a linear transformation of energy in band-specific ecog signals," in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2017, pp. 986–989.

[11] J. R. Stieger, S. Engel, H. Jiang, C. C. Cline, M. J. Kreitzer, and B. He, "Mindfulness improves brain–computer interface performance by increasing control over neural activity in the alpha band," *Cerebral Cortex*, vol. 31, no. 1, pp. 426–438, 2021.

[12] A. Sohrabpour, Z. Cai, S. Ye, B. Brinkmann, G. Worrell, and B. He, "Noninvasive electromagnetic source imaging of spatiotemporally distributed epileptogenic brain sources," *Nature communications*, vol. 11, no. 1, pp. 1–15, 2020.

[13] K. LaFleur, K. Cassady, A. Doud, K. Shades, E. Rogin, and B. He, "Quadcopter control in three-dimensional space using a noninvasive motor imagery-based brain–computer interface," *Journal of neural engineering*, vol. 10, no. 4, p. 046003, 2013.

[14] B. He, H. Yuan, J. Meng, and S. Gao, *Brain–Computer Interfaces*. Cham: Springer International Publishing, 2020, pp. 131–183.

[15] O. Ali, M. Saif-ur Rehman, S. Dyck, T. Glasmachers, I. Iossifidis, and C. Klaes, "Improving the performance of eeg decoding using anchored-stft in conjunction with gradient norm adversarial augmentation," *arXiv preprint arXiv:2011.14694*, 2020.

[16] K.-R. Müller and J. M. Carmena, "Editorial ieee brain initiative special issue on bmi/bci systems," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 10, pp. 1685–1686, 2017.

[17] C. G. Coogan and B. He, "Brain-computer interface control in a virtual reality environment and applications for the internet of things," *IEEE Access*, vol. 6, pp. 10 840–10 849, 2018.

[18] B. Allahgholizadeh Haghi, S. Kellis, S. Shah, M. Ashok, L. Bashford, D. Kramer, B. Lee, C. Liu, R. Andersen, and A. Emami, "Deep multi-state dynamic recurrent neural networks operating on wavelet based neural features for robust brain machine interfaces," *Advances in Neural Information Processing Systems*, vol. 32, pp. 14 514–14 525, 2019.

[19] Z. Tayeb, J. Fedjaev, N. Ghaboosi, C. Richter, L. Everding, X. Qu, Y. Wu, G. Cheng, and J. Conradt, "Validating deep neural networks for online decoding of motor imagery movements from eeg signals," *Sensors*, vol. 19, no. 1, p. 210, 2019.

[20] B. Haghi, S. Kellis, S. Shah, M. Ashok, L. Bashford, D. Kramer, B. Lee, C. Liu, R. A. Andersen, and A. Emami, "Deep multi-state dynamic recurrent neural networks operating on wavelet based neural features for robust brain machine interfaces," *bioRxiv*, p. 710327, 2019.

[21] E. R. Kandel, J. H. Schwartz, T. M. Jessell, S. Siegelbaum, A. J. Hudspeth, and S. Mack, *Principles of neural science*. McGraw-hill New York, 2000, vol. 4.