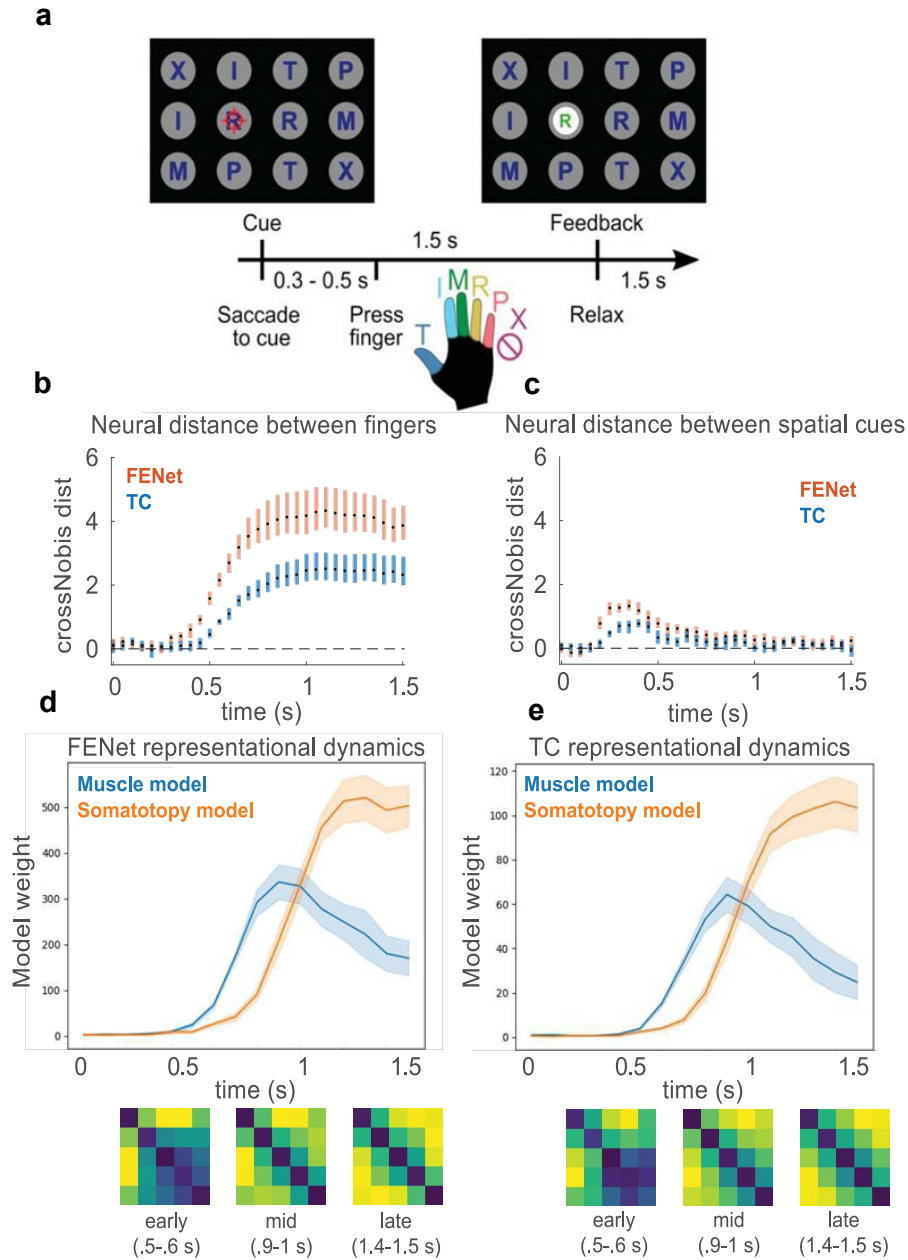
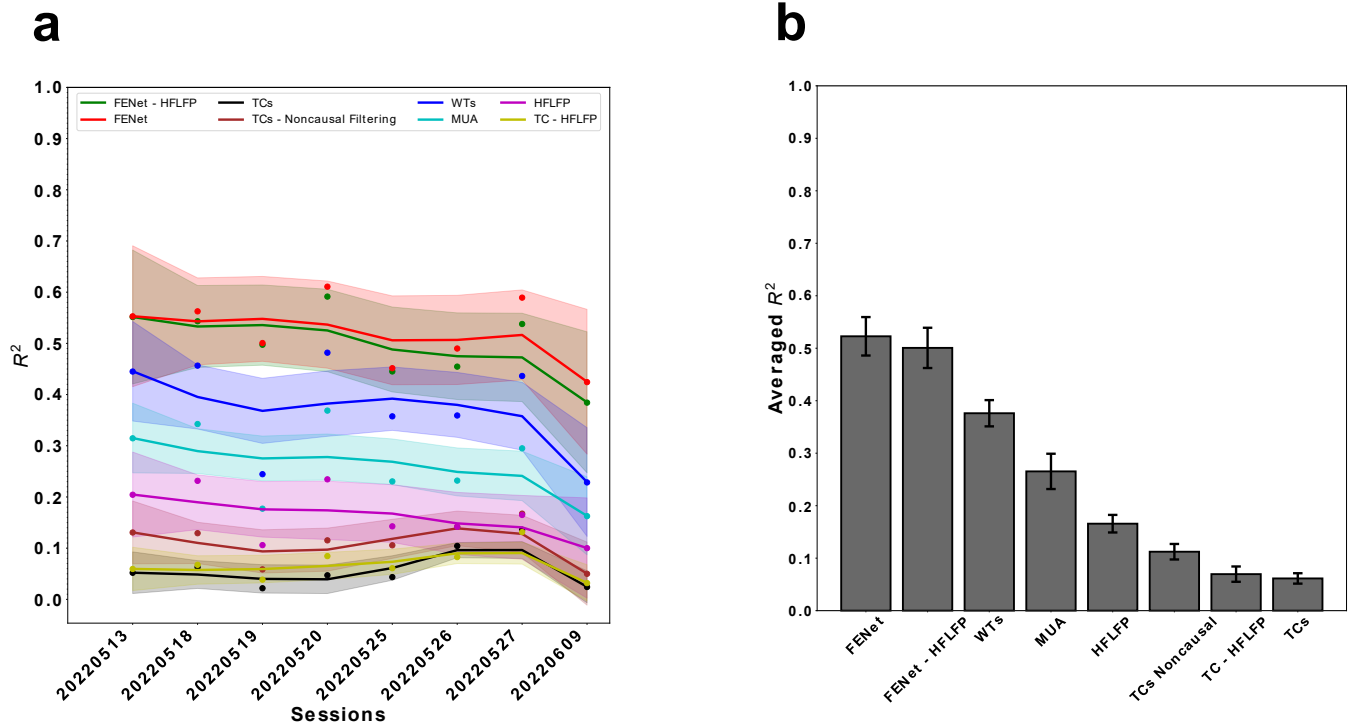


Enhanced control of a brain–computer interface by tetraplegic participants via neural-network-mediated feature extraction

In the format provided by the
authors and unedited



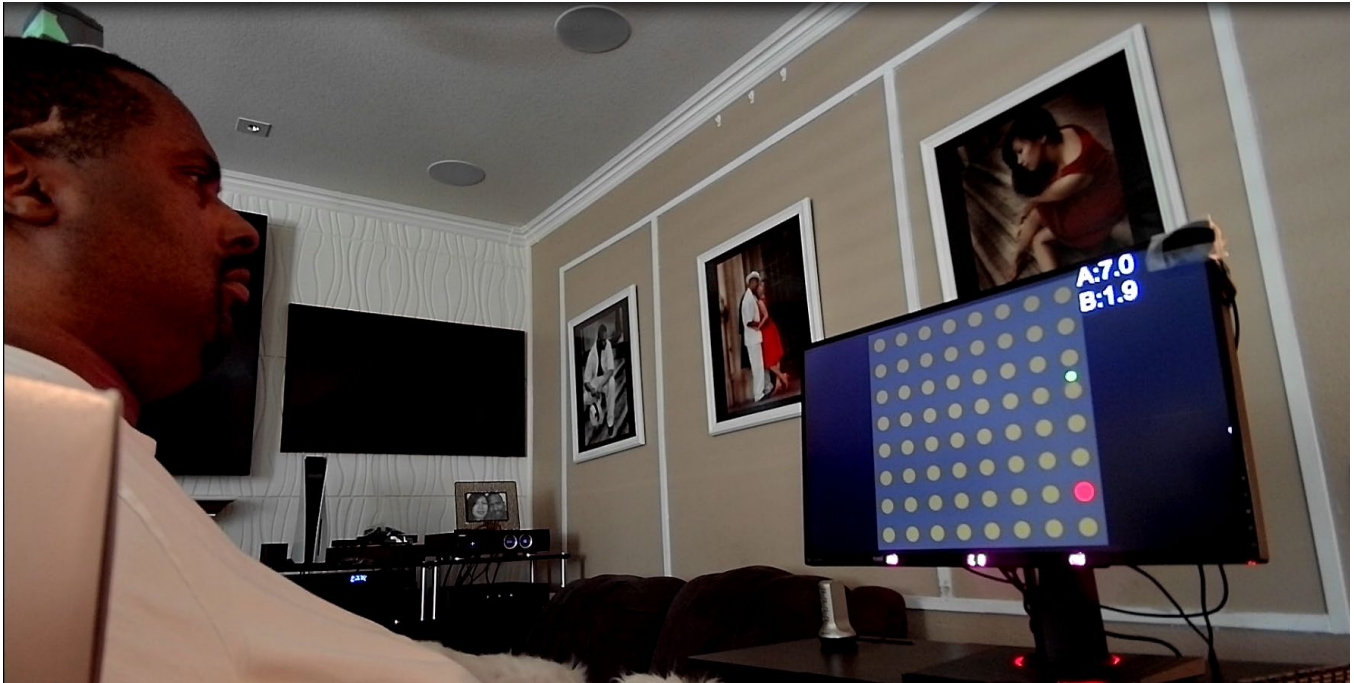
Supplementary Fig. 1 | Finger-grid task for testing generalizability of FENet over multiple tasks. Robust brain-machine interface (BMI) control of individual fingers. (a) Main finger flexion task. When a letter was cued by the red crosshair, the participant looked at the cue and immediately attempted to flex the corresponding finger of her right (contralateral) hand. We included a null condition ‘X’, during which the participant looked at the target but did not move her fingers. Visual feedback indicated the decoded finger 1.5 s after cue presentation. To randomize the gaze location, cues were located on a grid (three rows, four columns) in a pseudorandom order. The red crosshair was jittered to minimize visual occlusion. (b) Neural distance between fingers in time for FENet and TCs. (c) Neural distance between targets in time for FENet and TCs. (d), (e) RDA of the measured RDM over time shows an early fit to the muscle model and a late fit to the somatotopy model (see methods). Confidence intervals indicate $\pm$ standard error of the mean (SEM) bootstrapped across 10 sessions. Gray shaded region indicates the approximate onset time of the saccade to cue (interquartile range across trials). Difference in model start time (170 ms) was significant ($P = 0.002$, two-sided Wilcoxon signed rank test). RDM snapshots (bottom, each max-scaled to [0, 1]) intuitively visualize the change in representational structure over time from muscle-like to somatotopic. Part of the figure and legend text have been reproduced from Fig. 1 and Fig. 2 of ref. 50. The original image and legend text are published under the terms of the Creative Commons Attribution-Non-commercial 4.0 International license (CC BY-NC 4.0) <https://creativecommons.org/licenses/by/4.0>



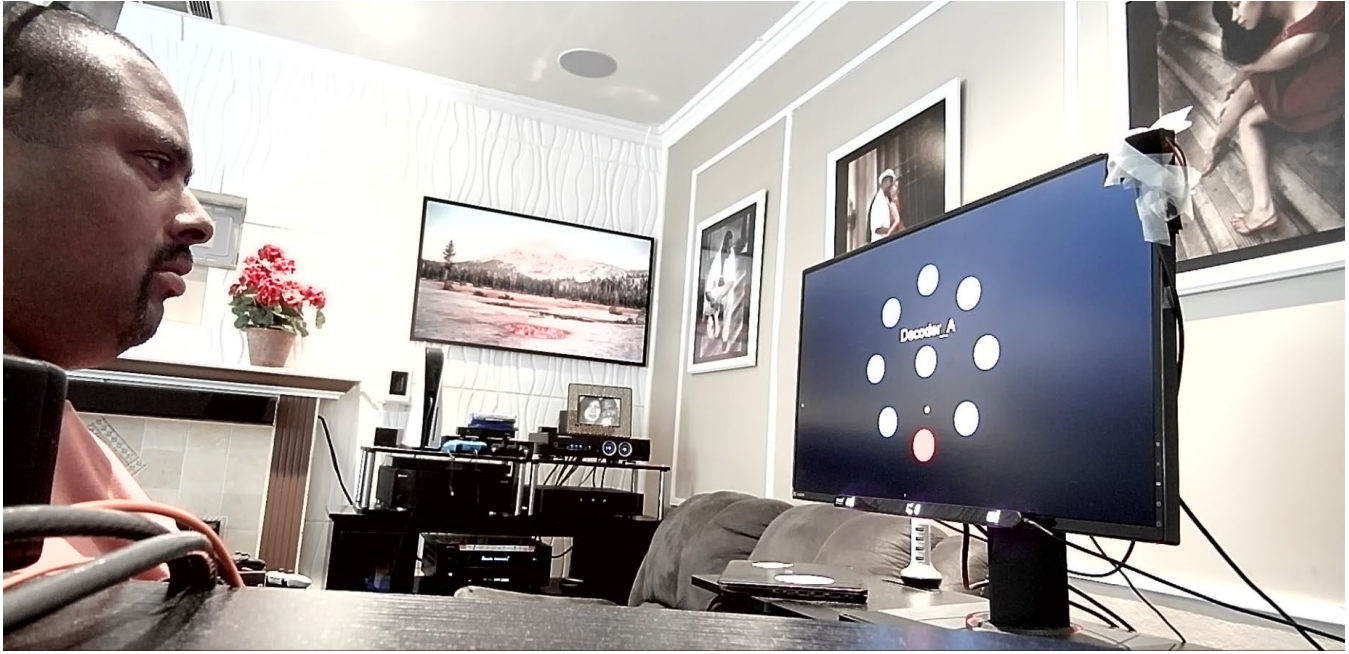
Supplemental Fig. 2 | Decoder performance comparison across eight sessions using different feature extraction techniques. (a) The session-wise R^2 values of the linear decoder operating feature extraction techniques applied over the course of eight sessions. Each line represents a different feature extraction method: FENet (red), FENet + HFLFP (green), TCs - Noncausal Filtering (black), WT (blue), MUA (cyan), HFLFP (purple), and TC + HFLFP (pink). FENet consistently achieves the highest performance, while Threshold-Crossing (TC) techniques, both with and without non-causal filtering, lag behind. Notably, applying non-causal filtering to the TC features improves the decoder's performance slightly, but the performance still remains far behind the FENet method. The band in each time series shows the range of its 95% confidence interval of a LOESS^{85,86} fit. (b) Averaged R^2 performance. The bar plot demonstrates the average R^2 values across all sessions for each feature extraction technique, with error bars indicating the standard error of the mean. FENet provides the highest overall averaged performance, with an average R^2 of 0.5228 on these eight sessions, notably outperforming the TC-based methods. The TCs - Noncausal Filtering method improves the decoder's performance (average R^2 of 0.1123) compared to standard TCs (0.0614), however, there remains a substantial gap between this method and the FENet decoder. This figure highlights the relative strengths of different feature extraction approaches in enhancing decoder performance, with FENet emerging as the most effective technique. The non-causal filtering applied to TCs improves performance but is still notably less effective than FENet in terms of decoding accuracy.



Supplemental Video 1 | Closed-loop test for center-out task. This video shows the second and third step of our closed-loop recording pipeline (Extended Figure 1a). All the movements are performed in real-time. We re-train the decoders by recording a block of closed-loop center-out data for the total of 64 trials. Next, we record a block of double-blind center-out data for closed-loop control performance evaluation, which each includes 96 trials. We switch between the feature extraction techniques A and B every 16 trials for all steps. In the middle, we ask the subject's preference regarding the feature extraction techniques A and B. We have used the double-blind testing to capture quantifiable and subjective performance metrics of the algorithms being tested for each of the standard feature types (TCs and FENet). Each run lasted approximately 3-5 minutes. We performed the runs in each session with at least a ten-minute break between and alternate the starting algorithm. The participant and researchers had been told which algorithm was being used ("A" or "B") but not what A or B were. After each batch, we queried the participant to capture subjective experience and preference in each session. In this stage of the closed-loop run, participants is controlling the movements completely. In this video, decoder A is using TCs, and decoder B is using FENet as the feature extraction technique. The difference between A and B in the video is clearly showing that the FENet-based decoder has superior performance compared to the TCs-based decoder.



Supplemental Video 2 | Closed-loop test for the grid task. This video shows the fourth step of our closed-loop recording pipeline (Extended Figure 1a). All the movements are performed in real-time. We record a block of double-blind grid data for closed-loop control performance evaluation, which each includes 96 trials. We switch between the feature extraction techniques A and B every 16 trials for all steps. In the middle, we ask the subject's preference regarding the feature extraction techniques A and B. We have used the double-blind testing to capture quantifiable and subjective performance metrics of the algorithms being tested for each of the standard feature types (TCs and FENet). We performed the runs in each session with at least a ten-minute break between and alternate the starting algorithm. The participant and researchers had been told which algorithm was being used ("A" or "B") but not what A or B were. After each batch, we queried the participant to capture subjective experience and preference in each session. In this stage of the closed-loop run, participants is controlling the movements completely. In this video, decoder A is using FENet, and decoder B is using TCs as the feature extraction technique. The numbers show the bitrates. The difference between A and B in the video is clearly showing that the FENet-based decoder has superior performance compared to the TCs-based decoder.



Supplemental Video 3 | Closed-loop test for center-out task. This video shows the second and third step of our closed-loop recording pipeline (Extended Figure 1a). All the movements are performed in real-time. We re-train the decoders by recording a block of closed-loop center-out data for the total of 64 trials. Next, we record a block of double-blind center-out data for closed-loop control performance evaluation, which each includes 96 trials. We switch between the feature extraction techniques A and B every 16 trials for all steps. In the middle, we ask the subject's preference regarding the feature extraction techniques A and B. We have used the double-blind testing to capture quantifiable and subjective performance metrics of the algorithms being tested for each of the standard feature types (WTs and FENet). Each run lasted approximately 3-5 minutes. We performed the runs in each session with at least a ten-minute break between and alternate the starting algorithm. The participant and researchers had been told which algorithm was being used ("A" or "B") but not what A or B were. After each batch, we queried the participant to capture subjective experience and preference in each session. In this stage of the closed-loop run, participants is controlling the movements completely. In this video, decoder A is using WTs, and decoder B is using FENet as the feature extraction technique. The difference between A and B in the video is clearly showing that the FENet-based decoder has superior performance compared to the WTs-based decoder.



Supplemental Video 4 | Closed-loop test for the grid task. This video shows the fourth step of our closed-loop recording pipeline (Extended Figure 1a). All the movements are performed in real-time. We record a block of double-blind grid data for closed-loop control performance evaluation, which each includes 96 trials. We switch between the feature extraction techniques A and B every 16 trials for all steps. In the middle, we ask the subject's preference regarding the feature extraction techniques A and B. We have used the double-blind testing to capture quantifiable and subjective performance metrics of the algorithms being tested for each of the standard feature types (WTs and FENet). We performed the runs in each session with at least a ten-minute break between and alternate the starting algorithm. The participant and researchers had been told which algorithm was being used ("A" or "B") but not what A or B were. After each batch, we queried the participant to capture subjective experience and preference in each session. In this stage of the closed-loop run, participants is controlling the movements completely. In this video, decoder A is using WTs, and decoder B is using FENet as the feature extraction technique. The numbers show the bitrates. The difference between A and B in the video is clearly showing that the FENet-based decoder has superior performance compared to the WTs-based decoder.