Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Memristor chip-enabled adaptive neuromorphic decoder for co-evolutional brain-computer interfaces

Huaqiang Wu

wuhq@tsinghua.edu.cn

Tsinghua University https://orcid.org/0000-0001-8359-7997

Zhengwu Liu

Tsinghua University https://orcid.org/0000-0001-7968-9469

Jie Mei

Tianjin University

Jianshi Tang

Tsinghua University https://orcid.org/0000-0001-8369-0067

Minpeng Xu

Tianjin University

Bin Gao

Tsinghua University https://orcid.org/0000-0002-2417-983X

Kun Wang

Tianjin University

Sanchuan Ding

Suzhou Institute of Nano-Tech and Nano-Bionics

Qi Liu

Tsinghua University https://orcid.org/0009-0001-4269-3381

Qi Qin

Tsinghua University

Weize Chen

Tianjin University

Yue Xi

Tsinghua University

Yijun Li

School of Integrated Circuits, Tsinghua University

Peng Yao

Tsinghua University

Han Zhao

Tsinghua University https://orcid.org/0000-0001-6110-6800
Ngai Wong
The University of Hong Kong
He Qian
Tsinghua University https://orcid.org/0000-0003-3377-5366
Bo Hong
Tsinghua University https://orcid.org/0000-0003-2900-6791
Tzyy-Ping Jung
University of California, San Diego
Dong Ming
Tianjin University

Article

Keywords:

Posted Date: March 4th, 2024

DOI: https://doi.org/10.21203/rs.3.rs-3966063/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: There is **NO** Competing Interest.

Version of Record: A version of this preprint was published at Nature Electronics on February 17th, 2025. See the published version at https://doi.org/10.1038/s41928-025-01340-2.

Memristor chip-enabled adaptive neuromorphic decoder for co-evolutional brain-computer interfaces

Zhengwu Liu[1,#,†], Jie Mei[2,#], Jianshi Tang[1,3,*], Minpeng Xu[2,4,5,*], Bin Gao[1,3], Kun Wang[2,4,5], Sanchuang Ding[1], Qi Liu[1], Qi Qin[1], Weize Chen[2], Yue Xi[1], Yijun Li[1], Peng Yao[1], Han Zhao[1], Ngai Wong[6], He Qian[1,3], Bo Hong[7], Tzyy-Ping Jung[2,8], Dong Ming[2,4,5,*], Huaqiang Wu[1,3,*]

Affiliations:

[1]School of Integrated Circuits, Beijing Advanced Innovation Center for Integrated Circuits, Tsinghua University, Beijing, 100084, China.

[2]Academy of Medical Engineering and Translational Medicine, Medical School, Tianjin University, Tianjin, 300072, China.

[3]Beijing National Research Center for Information Science and Technology (BNRist), Tsinghua University, Beijing, 100084, China.

[4]Haihe Laboratory of Brain-computer Interaction and Human-machine Integration, Tianjin 300072, China

[5]State Key Laboratory of Advanced Medical Materials and Devices, Tianjin 300072, China

[6]Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong SAR, China

[7]Department of Biomedical Engineering, School of Medicine, Tsinghua University, Beijing 100084, China.

[8]Swartz Center for Computational Neuroscience, University of California, San Diego, CA, USA.

[†]Present address: Department of Electrical and Electronic Engineering, The University of Hong Kong; Hong Kong SAR, China

[#]These authors contributed equally: Zhengwu Liu, Jie Mei.

*Corresponding authors: jtang@tsinghua.edu.cn (J.T.), minpeng.xu@tju.edu.cn (M.X.), richardming@tju.edu.cn (D.M.), wuhq@tsinghua.edu.cn (H.W.).

Abstract:

To fulfill complex human-machine interactions, a brain-computer interface (BCI) must not only decipher brain signals but also dynamically adapt to brain fluctuations, ultimately co-evolving with the brain. This necessitates a novel decoder capable of flexible updates with energy-efficient decoding capabilities. In this work, we designed a co-evolutional BCI with a neuromorphic decoder enabled by a 128k-cell memristor chip. By interacting with the brain, the decoder continuously updates its parameters, leading to the successful real-time control of a drone in 4 degrees of freedom (4-DOF) and enabling it to navigate around obstacles. Our approach featured a hardware-efficient one-step memristor decoding strategy, enabling the neuromorphic chip-equipped BCI to achieve decoding performance equivalent to software-based methods. Notably, it accomplished this at three orders of magnitude lower energy consumption and two orders of magnitude higher normalized speed than a central processing unit (CPU). Moreover, employing an interactive update framework, we showed the co-evolution of the brain-memristor decoder over an extended interaction task involving ten subjects. This resulted in a remarkable enhancement of BCI performance by nearly 20%, showcasing the substantial potential of memristor decoders in advancing BCIs. The study results also showed that the decoder initially played a dominant role in the co-evolution, but the brain learned as the process progressed. Eventually, a dynamic balance between the two emerged for decision-making. These findings lay the groundwork for developing future human-centric hybrid intelligence systems.

Main text:

Science fiction's concept of hybrid intelligence—integrating the human brain with machines—has become increasingly feasible thanks to the advancements in neuroscience

and information technology. At the forefront of these developments are brain-computer interfaces (BCIs), establishing direct links between the human brain and external machines. BCIs play a crucial role as tools for integrating human-machine functions and for diagnosing and treating brain diseases. Recent advancements in this field include brain-to-text communication[1], speech synthesis[2-4], motor control[5], and spinal cord stimulation[6], enhancing abilities in both disabled and healthy individuals. However, during BCI use, the brain states and signals often fluctuate[7-17], influenced by learning or adaptation[5,7,9,10], recording electrode variations[14,15], and environmental changes[16,17]. Therefore, a practical BCI should be able to adapt to evolving brain states and interpret intentions during collaborative interactions, realizing a dynamic co-evolution between the brain and machines. Nevertheless, several key challenges remain. The first challenge concerns the passive and abrupt update scheme employed for the brain decoder. Current adaptive decoders primarily use data labels generated by pre-updated decoders[1,14,15], leading to a lack of active brain involvement[18] that impedes ongoing improvement of BCI performance. Moreover, abrupt updates of decoder parameters, regardless of the extent of changes in brain signals and decoders[6,7], clearly deviate from the human brain's natural, gradual adaptation[12] to the external world. The second critical challenge lies in the insufficient decoding capability of computing hardware. The rapid growth in brain-machine links[19-22], commands[23,24], and decoding algorithm complexity[3,4], raises critical issues for conventional Si chips based on von Neumann architecture[25] to decode an increasing volume of brain signals, particularly given the slowdown of Moore's law[26]. Therefore, a more energy-efficient brain signal decoder and a hardware-friendly update scheme are needed to realize co-evolutional BCIs.

Encouragingly, neuromorphic computing chip based on memristors, a typical two-terminal resistive switching device whose tunable conductance resembles the synaptic weight in biological neural network, might provide disruptive solutions to the abovementioned challenges. First, the non-von Neumann paradigm of in-memory computing can be seamlessly implemented in situ on analog memristor crossbar arrays by

applying fundamental physics laws[27-29]. This implementation can potentially enhance the decoding energy efficiency for BCIs substantially. Besides, the memristor chip also supports in-situ parameter tuning[30-32], enabling adaptive decoders with a gradual update capability for continuous brain-machine interaction. However, realizing such a memristor chip-enabled BCI with a brain-decoder co-evolution remains arduous. For example, the non-ideal device characteristics of memristors lead to a tradeoff between accuracy and energy efficiency. Besides, long-term brain-machine interaction necessitates memristor decoder updates using new brain signal data and self-generated labels, rather than using pre-collected dataset for memristor learning[31,32]. Consequently, it is necessary to establish a memristor-friendly decoder update framework that integrates brain involvement to enhance the BCI performance and mitigate the non-ideal impacts of memristor decoders.

In this work, we designed and demonstrated a memristor chip-enabled BCI (**Fig. 1**), achieving heightened performance through a collaborative evolution of the brain and memristor decoder. To achieve high accuracy, speed, and low energy consumption in decoding brain signals, we developed a one-step memristor decoding strategy, providing a valuable tool to analyze the sparsity of decoder parameters and the correlation between the brain signals and a specific task. In a complex real-time task involving brain-controlled drone flight, we demonstrated the feasibility and advantages of our BCI. It achieved decoding performance equivalent to software methods but with a remarkable 1643× lower energy consumption and 216× higher normalized speed than a CPU. Furthermore, we developed an interactive update scheme, allowing the memristor decoder to adapt to the evolving brain signals naturally. This showcased a task of brain-memristor decoder co-evolution, resulting in enhanced overall performance. Ten participants engaged in the co-evolution task, each undergoing ~6 hours of experiment. The results revealed that the BCI with brain-memristor decoder co-evolution achieved ~20% higher accuracy than that without co-evolution. Both the feature maps of brain signals and the conductance map of memristor decoders tended to stabilize with periodic updates, indicating brain learning and decoder adaptivity. This work facilitates seamless

human-machine integration in BCIs with long-term stability and adaptivity for medical treatment and brain rehabilitation. It could also provide a vital interface to explore how brain works and empower future human-centric hybrid intelligence systems.

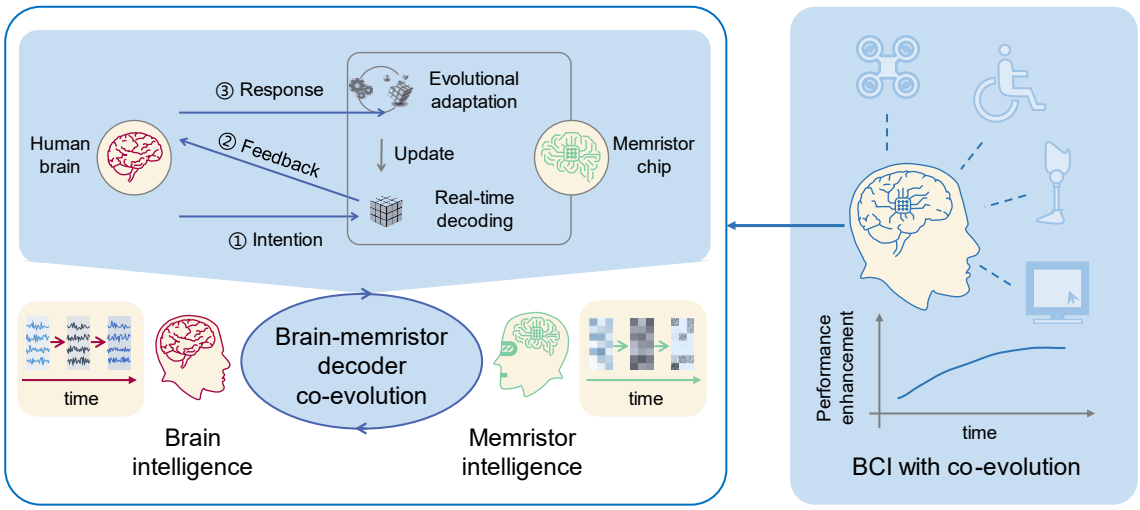


Fig. 1 | Schematic of BCI with brain-memristor decoder co-evolution. Enhanced BCI that integrates human brain intelligence with memristor chip-based machine intelligence (memristor intelligence) is enabled by brain-memristor decoder co-evolution. The process involves three key signals: intention, feedback, and response. Initially, the brain sends out a control intention through its activities. Subsequently, the memristor chip decodes it, transmitting the decoding result to the brain as feedback. After that, the brain responds to whether the decoding is correct, guiding the memristor chip update and decoding model transfer. The three sets of evolving multi-channel brain signals indicated by red arrows illustrate the fluctuations of brain activities. In contrast, the evolving decoder maps indicated by green arrows illustrate the adaptive memristor decoders. The right panel exemplifies typical BCI-controlled devices, such as a drone, wheelchair, prosthesis and mouse.

Framework of a memristor chip-enabled BCI

To construct a prototype memristor chip-enabled BCI, we used the steady-state visually evoked potential (SSVEP) as the intention, the decoding result from the memristor chip as the feedback, and the error-related potential (ErrP) as the response, respectively (**Fig. 2**). SSVEP, closely correlated with visual stimuli[33,34], is one of the most used brain signals to build BCI. ErrP, a neurophysiological signal associated with error processing in the brain[35-37], is evoked when the brain recognizes an error. **Fig. 2a** illustrates the framework of the memristor-enabled BCI. The proposed interactive update scheme of the memristor chip-based decoder is illustrated in the hybrid path. Here, ErrPs are used to accumulate new data in the online interaction process. More specifically, if the memristor decoder's decision does not elicit an ErrP, the corresponding SSVEP signal and the decoding result serve as a new training sample. The decoder undergoes updates during the interaction phase as these new training samples accumulate. We can use the updated decoder for the subsequent loop through this human-in-the-loop mechanism.

Fig. 2b is the memristor chip-based decoding platform, and the die photo of the memristor chip is shown in **Fig. 2a**. We elaborate on the chip fabrication process in **Methods**. The chip consists of a 1024×128 (i.e., 128k-cell) memristor array that supports massively parallel multiply-accumulate computations, and peripheral circuits for input and output buffers and drivers. Each cell in the array is in the form of one-transistor-one-resistor (1T1R). The memristor has a material stack of TiN/TaO_x/HfO_2/TiN, as shown in the cross-sectional transmission electron microscope (TEM) image in **Fig. 2c**. The memristor array exhibits excellent analog switching characteristics for high-efficiency in-memory computing as shown in **Extended Data Fig. 1**.

We used the representative ensemble task-related component analysis (TRCA) algorithm[38] to decode the brain signals. To deploy it efficiently on the memristor chip, here we propose a one-step decoding strategy that considers both accuracy and efficiency by minimizing the error accumulation from multiple memristor arrays/chips-based sequential computations. The key idea is to merge the original three-step operation, including preprocessing, feature extraction, and pattern recognition, into a

one-step operation (**Fig. 2d** and see **Methods** for the mathematical derivation). This method significantly reduces the computation complexity and the required number of memristor arrays. As an example, for a task with 8-channel signals and 12-class control commands, 6.5× fewer computations can be achieved using this one-step strategy compared with the conventional three-step implementation (**Fig. 2e**). The temporal filter, spatial filter, and SSVEP template signal for the conventional implementation are illustrated in **Figs. 2f-h**, respectively. The new strategy combines these components into one matrix and then maps them onto the memristor chip as the device conductance in differential pairs, as shown in **Fig. 2i**. We describe more details about the mapping process in **Methods**. As can be seen, this one-step strategy simplifies the operations, thus avoiding the accumulation of errors caused by multi-step operations on the memristor chip. More interestingly, it also increases the interpretability of the decoder, whose pattern could reflect the channel preferences of the brain and the sparsity of control parameters in the interactive BCI tasks, as to be investigated later. For example, the pattern in **Fig. 2i** suggests that the brain relies more on the signals from channels 7, 5, 1, and 3 than the others for this specific task since they have larger parameter magnitudes.

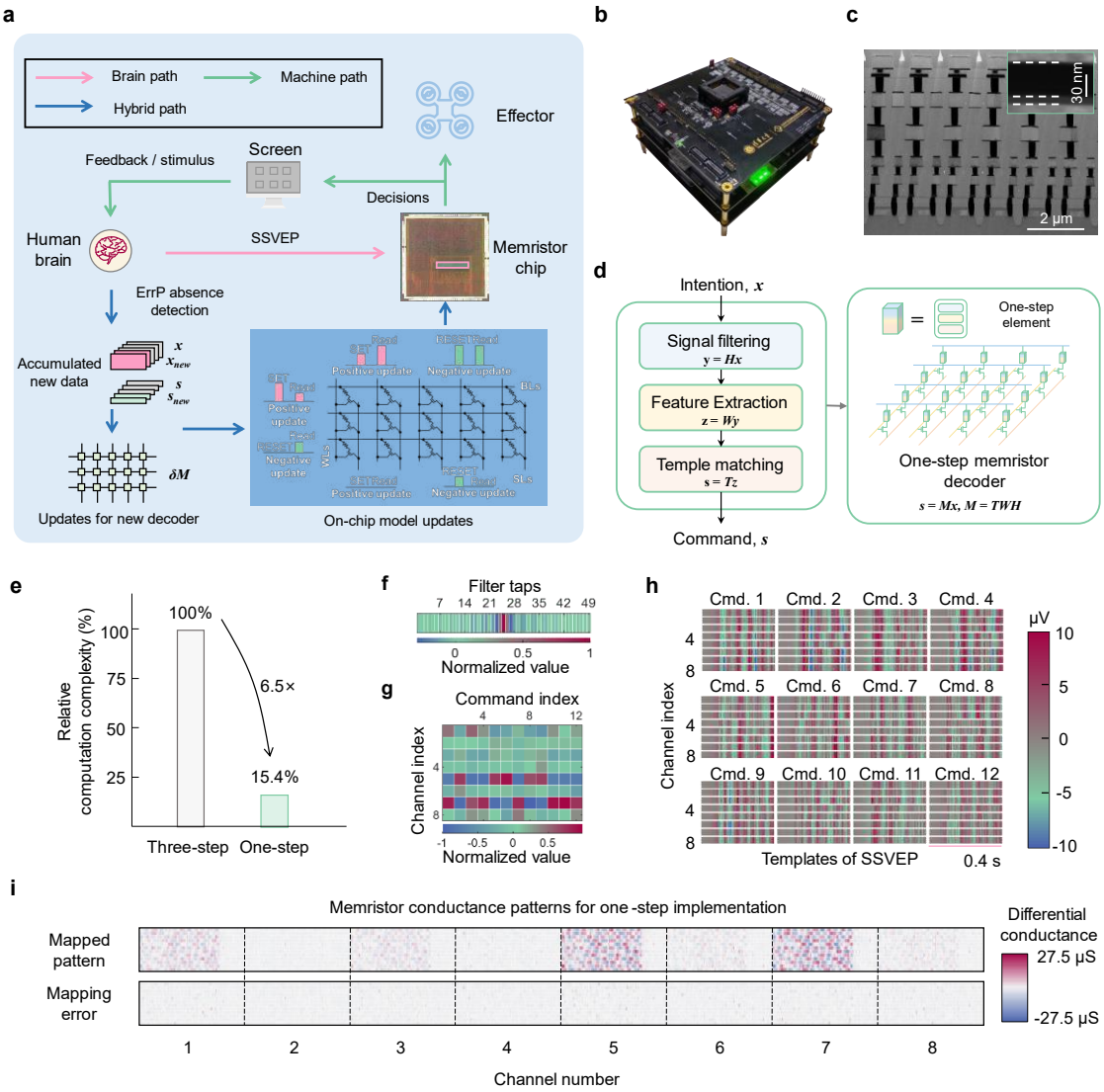


Fig. 2 | Design of a memristor chip-enabled BCI. a, Framework of the memristor chip-enabled BCI. The pink, green, and blue arrows represent the brain, decoder, and hybrid paths, respectively. In the brain and decoder paths, the memristor chip decodes the brain's SSVEP signals and then feeds the results back to the brain. In the hybrid path, ErrP responses from the brain are detected to accumulate new online training samples to update the decoder. SET/RESET pulses are then applied on the memristor chip to update the model. **b,** The photo of the memristor chip-based decoding platform. **c,** A cross-sectional TEM image of the memristor chip with 1T1R cells. The inset shows the TiN/TaO_x/HfO_2/TiN material stack of the memristor device. **d,** One-step implementation of the SSVEP decoding algorithm on the memristor array by merging the original preprocessing (temporal filtering), feature extraction (spatial filtering), and pattern

recognition (template matching) into one matrix computation. **e,** Comparison of the computational complexity for the three-step and one-step decoding algorithm implementations for an 8-channel and 12-command drone flight control task. **f-h,** Coefficient maps of the temporal filtering (**f**) and spatial filtering (**g**) template matching signals (**h**) for the conventional three-step implementation in this task. **i,** Conductance maps for the concise one-step implementation. The upper panel is the mapped differential conductance on the memristor array, while the lower is the corresponding mapping error.

Real-time brain-controlled drone flight with memristor chip-based decoder

To verify the feasibility of the memristor chip-based decoder for BCI, we first carried out a sophisticated online task of 4-DOF real-time brain-controlled drone flight (**Fig. 3a)**. Besides the memristor chip, the experimental setup also includes a brain signal acquisition system (Synamps2 system from Neuroscan Inc.), a drone (DJI RoboMaster TT), a screen showing the flight commands (see **Methods** and **Extended Data Table 1**), and artificial obstacles to guide the drone flight (see **Methods**). **Supplementary Video 1** shows a typical trial, where the drone successfully completed its intended three-dimensional (3D) trajectory around obstacles using the memristor chip-based decoder. **Fig 3b** illustrates the key frames of three flight nodes T_1-T_3.

To quantify the decoder performance, we performed the memristor chip-based simulated online experiments on five subjects (marked as A1-A5). **Fig. 3c** plots the differential conductance maps of the memristor decoder, indicating that different subjects could have distinct channel preferences. This observation stems from the commonly reported cross-subject variabilities[8] in BCIs. For instance, subject A1 primarily relies on signals from channels Oz, PO4, and PO6, while subject A5 favors channel POz. **Figs. 3d-e** compare the performance of BCIs based on the memristor chip and a conventional decoding hardware, i.e., Intel CPU, which exhibited similar information transfer rate (ITR,

123.58±10.33 bits/min for the memristor chip *versus* 126.26±8.40 bits/min for the CPU) and decoding accuracy (85.17±3.76% for the memristor chip *versus* 86.08±2.74% for the CPU). **Extended Data Table 2** lists detailed results for each subject. Meanwhile, the memristor chip-enabled BCI consumed 1643× less energy and had 216× higher normalized speeds than the CPU-based system (**Figs. 3f, g**). The benchmark is detailed in **Methods** and **Extended Data Table 3**. Such equivalent accuracies suggest that the decoding results of the memristor chip are in excellent consistency with those of the CPU, as compared in **Fig. 3h**. **Fig. 3i** further presents the confusion matrix of the memristor chip-based decoding results for different commands. The above results clearly validate the feasibility and advantages of the memristor chip-based decoder for BCI.

Previous research reported that sparsity of control parameters clearly exists in many BCI tasks[7,39]. To investigate the effects of decoding parameters with various magnitudes on the system performance, we carried out parameter-pruning experiments using binary masks (**Extended Data Fig. 2**). **Extended Data Table 4** summarizes the results. As we can see from the table, the decoders that retained the top 20% parameters in magnitude could achieve similar accuracy (<1.5% accuracy loss, as shown in **Extended Data Table 4a**) as the unpruned decoders. In comparison, removing only the top 5% of the parameters could result in a significant degradation in accuracy (>10.0%, as shown in **Extended Data Table 4b**). This result clearly indicated the sparsity of parameters, and the decoder parameters with higher magnitudes played dominant roles in decoding the brain signals. We further investigated the importance and preference of different brain recording channels for the specific task. **Fig. 3j** shows the average magnitude of parameters in each channel of the decoders. **Fig. 3k** presents the decoding accuracy based on a single-channel brain signal for every subject. It can be seen that one or a few individual channels contained most of the critical information for decoding (for example, channel Oz for subject A1, which alone already achieved a high accuracy of 90.0%), while the others contributed very little (such as channel PO4 for subject A3, which alone achieved a chance-level accuracy of 8.75%). Comparing the patterns of the average magnitude (**Fig. 3k**) and accuracy (**Fig. 3j**) from a

channel-wise perspective, they are found to have a high similarity of 0.783. This is because those channels that achieved a higher accuracy often had parameters with larger magnitudes. Thus, they were more important than the other channels in the brain-controlled drone flight task. The above results showcase the close correlation between the memristor-chip based decoder and the brain activities from both parameter-wise and channel-wise perspectives, which could offer a useful tool to study the 'credit assignments' to brain signals in the task. In addition, the sparse parameters of the decoder are also favorable for hardware implementation and system optimization, leading to lower decoding costs in terms of chip area and energy consumption.

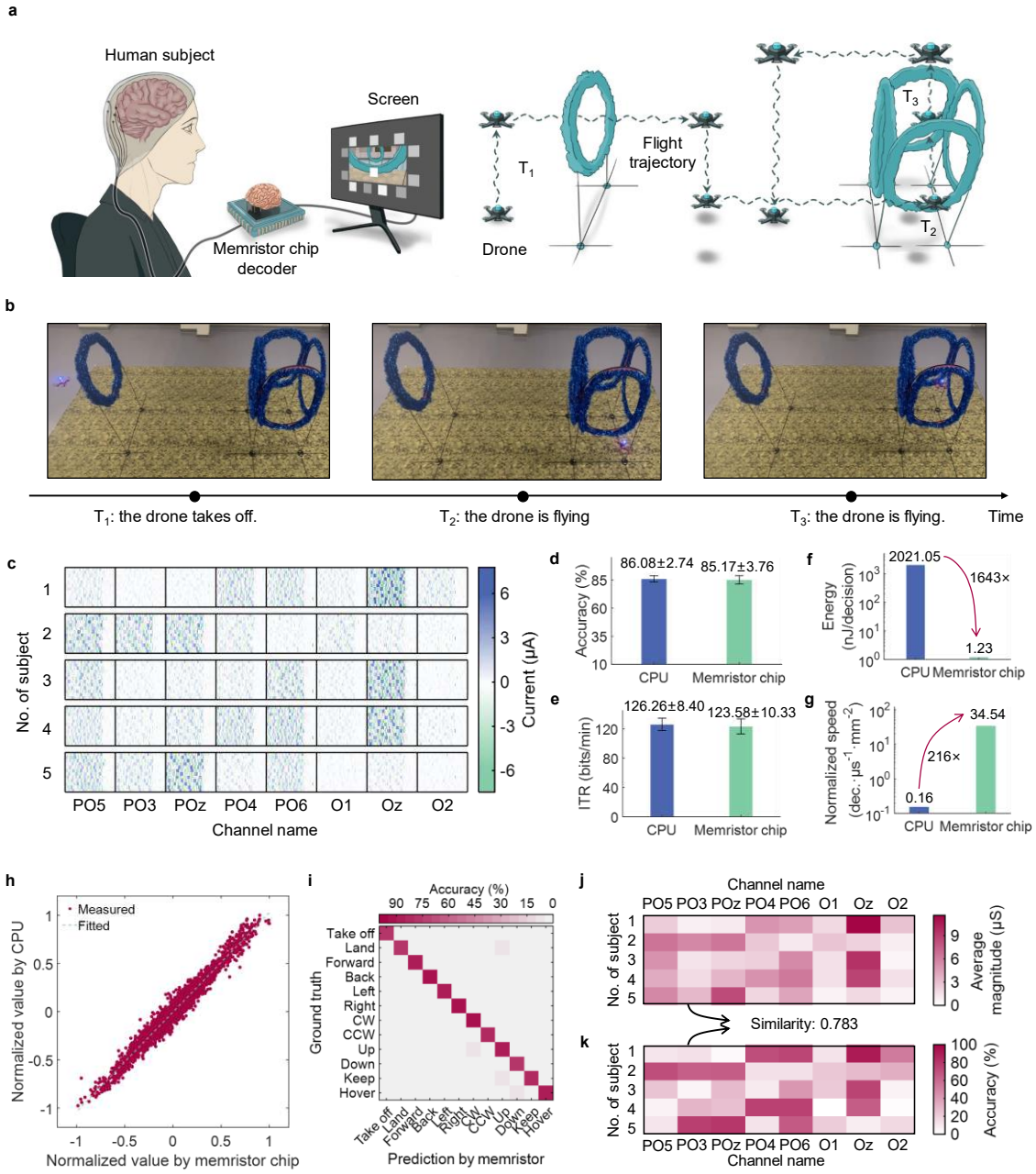


Fig. 3 | Real-time brain-controlled drone flight with memristor chip-based decoder. **a,** Schematic of the experimental setup. The left panel illustrates that the subject controls the drone flight through the memristor chip-based decoder. The screen shows 12 encoded stimuli of flight commands along with a real-time video stream of the drone. The right panel shows the flight scene and trajectory in the online experiment. T_1-T_3 represent several key time points during the online drone flight. **b,** Key frames of the memristor chip-based online decoding experiment. **c,** Differential conductance maps of the memristor decoder for five different subjects. **d-g,** Performance comparisons between

CPU and the memristor chip, including (**d**) accuracy, (**e**) ITR, (**f**) energy consumption, and (**g**) normalized decoding speed. **h,** Comparison of the decoding results from the CPU and the memristor chip in simulated online experiments, showing a linear relationship and high consistency. **i,** Confusion matrix of the memristor chip-based decoding experiment. **j**, Average amplitude of the decoder parameters for the five subjects. **k**, Accuracy map of single-channel brain decoding experiments for the five subjects.

Brain-memristor decoder co-evolution for enhanced BCI

To demonstrate the co-evolution of the brain-memristor decoder for an enhanced BCI, we conducted two experiments: the first one is a Simulated Update experiment (denoted as 'SU'), and the other one is a Human-in-the-Loop Update experiment (denoted as 'HLU'). In both experiments, we initiated the decoder by training it with an offline dataset and subsequently updated it with the newly acquired data. Finally, we evaluated the performance of the updated decoder in the extended experiment. The major difference between these two experiments was how to label new samples. The SU experiment used ground truths for labeling to validate whether the decoder update improved the BCI's performance. In contrast, the HLU experiment used ErrPs in online setups for labeling, intending to explore the feasibility of interactive brain-memristor decoder co-evolution.

In the SU experiment (on subject A3), following five updates, the decoding accuracy exhibited a notable improvement of more than 5% with the brain-memristor decoder co-evolution compared to its non-co-evolving counterpart (see **Extended Data Fig. 3a)**. This outcome unequivocally demonstrates that the memristor chip-based decoder, coupled with the interactive update scheme, possesses the ability to adapt to evolving brain signals, thereby enhancing the decoding performance over an extended period. To track the change of SSVEP signals, the decoder map underwent periodic updates, tending to converge to a stable state. This convergence is evident in the correlation coefficients,

which progressively increased from 0.81 initially to 1.0 at the conclusion of the experiment (see **Extended Data Figs. 3b-c**).

To further demonstrate the brain-memristor decoder co-evolution, we then conducted the HLU experiment on ten subjects (marked as B1-B10), in which subject-dependent ErrP detectors were first established to accumulate online data (see **Methods**). To achieve high-accuracy detection, we carried out an ErrP calibration experiment. **Extended Data Fig. 4** shows the performance of a typical detector. We studied the effects of the training set size and the algorithm threshold on the ErrP detection performance. Increasing the training set size generally enhanced the performance (**Extended Data Fig. 4a**). An optimal threshold of about 0.5 achieved the highest balanced accuracy of 82.6% (**Extended Data Fig. 4b**). **Extended Data Figs. 4c, d** detail the ErrP detection using this threshold. Our next step was to conduct an online ErrP detection using the same workflow, observing how the brain responded to the correct or incorrect decoding result obtained by the memristor chip. When incorrect decoding occurred during the online experiments, an apparent positive spike was observed at around 200-300 ms following the stimulus presentation (see **Extended Data Fig. 4e**).

We further analyzed the BCI performance with the memristor chip-enabled adaptive decoder. Throughout the experiment, the decoding accuracy of the BCI, coupled with brain-memristor decoder co-evolution, consistently outperformed its counterpart without co-evolution across all subjects (see **Extended Data Fig. 5**). Notably, the performance gap between the two approaches widened progressively over time. By the 9th update, corresponding to a ~6-hour experiment duration, we observed a remarkable ~20% improvement in the average accuracy (see **Fig. 4a**). **Fig. 4b** and **Extended Data Figs. 6-7** illustrate the evolution of the decoding accuracy of each command in BCIs with and without brain-memristor decoder co-evolution, respectively. These figures suggest that the decoder acquired new information during interaction and consistently improved accuracy with each update.

To reveal the continuous fluctuations of brain signals and the change of brain control state for the task, we analyzed the evolution of subject-averaged accuracies (**Fig. 4c**, see **Methods**) using the unsupervised algorithm, filter bank canonical correlation analysis[33] (FBCCA). The increasing trend shows the gradually improved brain control state during the 6 hour-long experiment. Throughout the co-evolution process, the task-related components progressively adopted a waveform that resembled a combination of sinusoidal signals synchronized with the stimulus frequency. Computational analysis revealed a notable enhancement in the signal-to-noise ratio (SNR) of these components (**Fig. 4d**). We also observed the evolution of similarity between the extracted feature components of brain signals and the template signals (**Fig. 4e** and **Extended Data Fig. 8**). For all subjects, similarity increased, presenting a trend of stabilization with gradually saturated values. **Fig. 4f** shows the temporal waveforms of brain signal components after every decoder update. These results from all ten subjects illustrate how the task-related brain signals changed during the interaction.

We further analyzed the evolution of the adaptive memristor decoders. **Fig. 4g** plots the adaptive memristor decoder map of subject B1 in the initial state and after nine consecutive updates, while **Extended Data Fig. 9** plots the decoder maps of other subjects. The evolving pattern for all subjects showed a trend of stabilization and convergence. To verify this, we calculated the correlation coefficients between the patterns of the present and the final decoder maps. For example, **Fig. 4h** shows the evolution of the correlation coefficients for the decoder of subject B1 and each channel (insets), which continuously increased from ~0.74 to 1.00. **Extended Data Fig. 10** shows comparable results for other subjects. To study the credit assignments in this task, i.e., the importance of different channels and their changes, we then analyzed the evolving direction of the decoder by calculating the fitted slope of the parameter magnitudes. **Extended Data Fig. 11** plots the slopes of all eight channels for subject B1. **Extended Data Fig. 12** shows the evolution of the positions of the top 15% parameters. Different channels can evolve in distinct directions. As an example, channel PO3's decoder map

exhibited a downward trend in magnitudes (**Fig. 4i**). In contrast, channel Oz's showed an upward trend (**Fig. 4j**). To study the evolution trends for different control commands, here we plotted the distributions of the decoder parameters for each command (**Fig. 4k** for subject B1 and **Extended Data Fig. 13** for all subjects). As the update progressed, the decoder parameters became more and more distributed, as the standard deviations increased from ~3.3 μS to ~5.8 μS. In addition, we observed no noticeable difference in the update rules for different commands.

Towards practical applications, previous studies suggest that a BCI would achieve optimal performance when both the brain and the decoder can adapt to new scenarios[7,40,41]. In line with this, our system demonstrated this dynamic interplay, as evidenced by the brain learning over time and the memristor decoder adapting accordingly. To gain a deeper understanding of the enhanced performance of our co-evolutional BCI, we further explored the individual contributions of these two components. **Fig. 4l** plots the accuracy of the co-evolutional BCI ($A_{\mathbf{co-evo}}$, **Fig. 4a**), the brain control state ($A_{\mathbf{brain}}$, **Fig. 4c**), and the memristor decoder's effectiveness ($A_{\mathbf{decoder}}$) for this task (see **Methods**). A linear regression analysis on the z-scores (**Fig. 4m**) revealed that the slope of $A_{\mathbf{co-evo}}$ was greater than that of both $A_{\mathbf{brain}}$ and $A_{\mathbf{decoder}}$. In simpler terms, the co-evolutional BCI exhibited faster performance improvements compared to the human brain and the memristor decoder. This accelerated improvement can be attributed to the synergistic evolution of the brain and the decoder, collectively enhancing the performance of the co-evolutional BCI. As a first-order approximation, we can decompose their individual contributions using the following linear relationship:

$$A_{\mathbf{co-evo}} = \alpha A_{\mathbf{brain}} + \beta A_{\mathbf{decoder}}$$

$$\alpha, \beta \geq 0$$

where α and β denote the fitting coefficients associated with the brain and the decoder, respectively. We can explore the evolution of α and β by fitting this model in each update. Our findings reveal that the contribution of the decoder ($\alpha/(\alpha+\beta)$) decreased, while that

of the brain ($\beta/(\alpha+\beta)$) increased as the evolution progressed (**Fig. 4n**). This suggests that, in the initial stage of co-evolution, the decoder played a dominant role. However, as the co-evolution progressed, the brain gradually learned, leading to a dynamic balance between the decoder and the brain for decision-making. Such a dynamic shift may indicate a complex interaction between the human brain and the decoder, likely influenced by factors[40,42,43] such as the brain's plasticity, the decoder's adaptivity, and the human-in-the-loop feedback. Unlike conventional decoders, our memristor chip-based neuromorphic decoder, with inherently adaptive features, holds the potential to revolutionize brain-machine interaction.

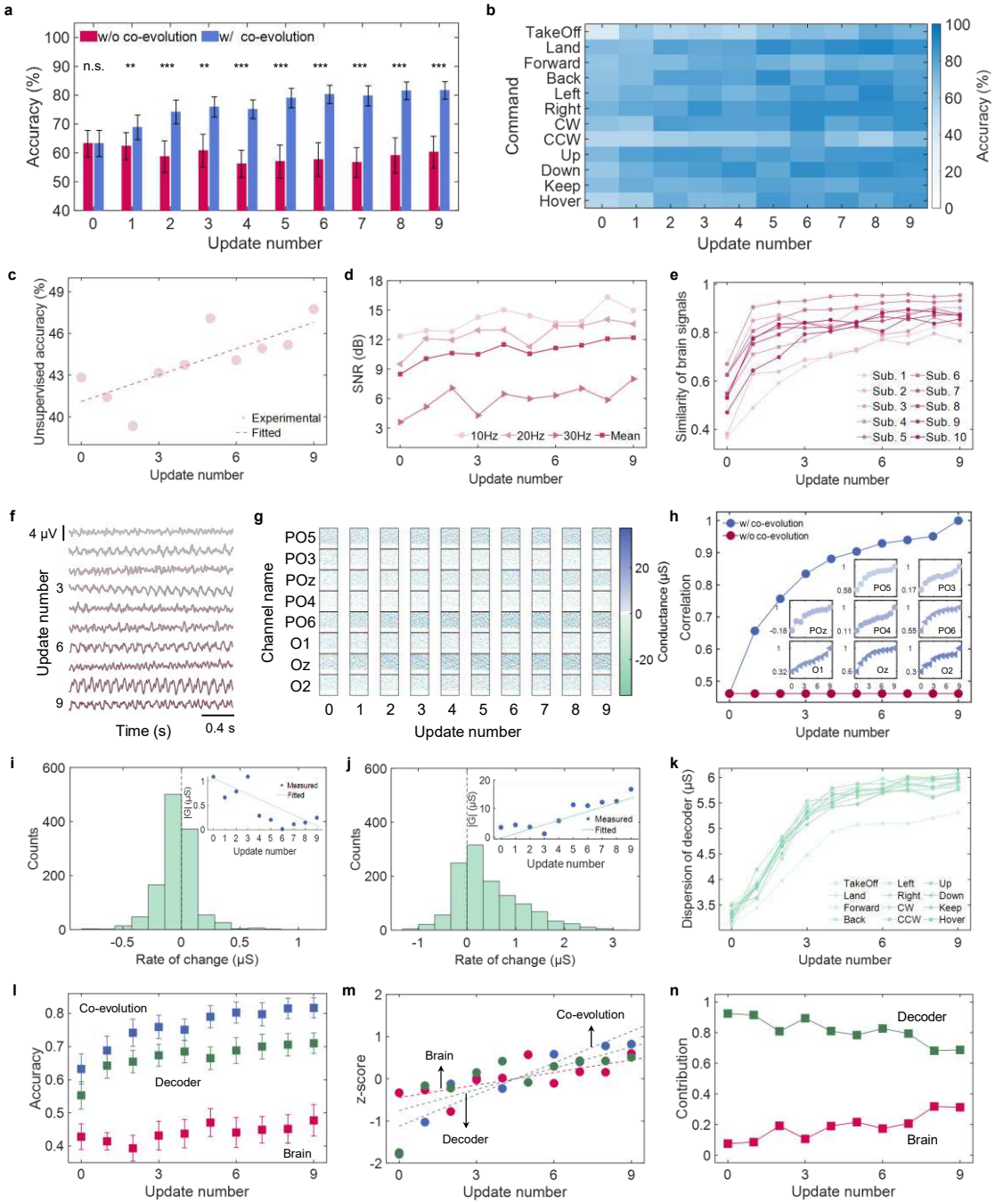


Fig. 4 | Online brain-memristor decoder co-evolution experiments with HLU. a, Decoding accuracies versus update cycles for BCI with (w/) and without (w/o) co-evolution for ten subjects. Mean in bar ± s.e.m. in error bar. Paired-samples t test: **: p<0.01, ***: p<0.001.**b,** The evolution of the decoding accuracy for 12 different control commands. Each pixel represents the average result of 10 subjects. **c**, Accuracies calculated using FBCCA after different updates. Each dot represents the average result of

ten subjects, and the dash line is the result of linear fitting (R^2=0.588, p<0.01). **d,** Progressive enhancement of SNR for the task-related components at 10 Hz and its harmonics during evolution. **e,** The similarities of these components between the test and template signals for all subjects. **f,** Temporal waveforms of task-related components after the decoder update. **g,** Adaptive memristor decoder maps after each update cycle for eight recording channels. **h,** Correlation coefficient between the decoder maps after each update and the final one. Insets: Correlation coefficient for each channel of the decoder. **i-j,** Two representative histograms of the slope of the decoder parameter change during the co-evolution process. The magnitude of the decoder parameters shows a decreasing (**i**, channel PO3 for subject B1) or increasing (**j**, channel Oz for subject B1) trend. Insets: examples of positive (**i**) and negative (**j**) fitted slopes. **k,** The evolution of the parameter distribution of the decoder for each command with the number of updates, evaluated by standard deviations. The data for **h**-**k** is from subject B1. Similar results could be observed in the other nine subjects, as shown in **Extended Data Figs. 9, 10** and **12** show similar results for the other nine subjects. **l,** The evolution of the accuracy of co-evolutional BCI (blue), the brain control state (red), and the memristor decoder's effectiveness (green), respectively. Mean in dot ± s.e.m. in error bar. **m**, The z-score of the three metrics in **l** averaged over all ten subjects. The dots are the measured data, and the dash lines are the results of linear fitting. For the blue line (co-evolution), R^2=0.801, p<0.001. For the green line (decoder), R^2=0.528, p<0.05. For the red line (brain), R^2=0.570, p<0.05. **n,** The evolution of the individual contribution of the brain ($\alpha/(\alpha+\beta)$, red) and the decoder ($\beta/(\alpha+\beta)$, green) to the enhanced co-evolutional BCI performance.

Conclusions

In summary, we have designed and implemented a neuromorphic BCI empowered by a memristor chip, fostering a symbiotic evolution with the human brain. A sophisticated brain-controlled 4-DOF drone flight task was successfully demonstrated

using a memristor chip-based adaptive decoder, achieving a remarkable software-equivalent high accuracy of 85.17% ± 3.76% and an ITR of 123.58±10.33 bits/min. Impressively, our approach operates at three orders of magnitude lower energy consumption and two orders of magnitude higher normalized speed than conventional CPU-based systems. Notably, we have developed a unique one-step memristor decoding strategy, which not only significantly reduced the computational complexity and hardware costs but also serve as a valuable tool for studying credit assignments in subject-dependent BCI tasks via analyzing the parameter sparsity of the memristor decoder. Furthermore, we have successfully showcased the brain-memristor decoder co-evolution using an interactive decoder update scheme based on ErrP, which achieved ~20% accuracy improvement in a 6-hour experiment. During this HLU experiment, both the brain signals and the memristor chip-based decoder gradually converged to their stabilized states. The decomposition of their individual contributions revealed an increasing role of the brain and a decreasing contribution of the decoder. This dynamic shift indicates the brain's learning and the decoder's adaptivity, enabling seamless co-evolution.

These results confirm that the memristor chip could provide a highly efficient and adaptive platform for decoding brain signals, and the brain-memristor decoder co-evolution feature significantly improved the overall BCI performance. Notably, the dynamic evolution of brain activities is not confined to scalp electroencephalogram recordings but is also evident in invasive recordings such as spikes[14] and electrocorticography[44]. Therefore, the proposed framework of brain-memristor decoder co-evolution can be generalized to develop various types of BCIs with long-term stability and highly adaptive capability for medical treatment and rehabilitation. In addition, this work also explored the close interaction and evolution between the biological brain and neuromorphic memristor chip, paving the road toward the future human-centered hybrid intelligence.

Methods:

Memristor chip fabrication and characteristics

Design and fabrication of 128k-cell memristor chip. The chip comprises a 1024×128 1T1R memristor array and Si complementary metal-oxide semiconductor (CMOS) peripheral circuits, including input buffers and drivers to support read/write and computation operations. Each cell in the array consists of an access transistor and a memristor with a material stack of TiN/TaO_x/HfO_2/TiN in the size of 0.5 μm × 0.5 μm. We fabricated the peripheral circuits and Si transistors in the array using a standard 130nm Si CMOS process. The 8nm HfO_2 layer was deposited by atomic layer deposition (ALD) to serve as the resistive switching layer, and the TaO_x and TiN layers were deposited by sputtering to serve as the thermal enhanced layer and top/bottom electrodes, respectively.

Memristor chip-based decoding platform. The platform mainly consists of three customized printed circuit boards (PCBs): a field-programming-gate-array (FPGA) board, a power board, and a socket board. The FPGA board is used for control. The power board generates various voltage signals through the integrated digital-to-analog converters (DACs) for the memristor chip mounted on the socket board. Besides the memristor chip, the socket board also contains eight analog-to-digital converters (ADCs) for array readout.

Mapping the decoder model onto the memristor chip. We used a standard write-verify programming scheme to map the decoder model onto the memristor chip. Each memristor device has two basic steps when mapping it to the target conductance. Firstly, read the device to obtain its current conductance, and calculate the difference between the target and current states. Secondly, if the difference is positive (negative), a SET (or RESET) pulse is applied to the device. The above two steps are repeated until the difference is within a pre-set margin.

Brain signal recording and decoding

Brain signal recording and preprocessing. In this study, brain signals were acquired using a Synamps2 system (Neuroscan Inc.) at a sampling rate of 1kHz and were resampled to 250Hz before feeding to the memristor chip. 64-channel whole-head signals were recorded. Eight of these channels over the parietal and occipital areas (PO5, PO3, POz, PO4, PO6, O1, Oz, and O2) were used as SSVEP signals. 32 channels of them over the other areas (FP1, FP2, F7, F3, Fz, F4, F8, FC5, FC1, FC2, FC6, T7, C3, Cz, C4, T8, TP7, CP5, CP1, CP2, CP6, TP8, P7, P3, Pz, P4, P8, PO7, POz, PO8, O1, O2) were used to represent ErrP-related signals. During the recording, we kept the impedance of all electrodes below $10\mathrm{k}\Omega$ to ensure good signal quality. The reference electrode was placed at the vertex. A total of 15 subjects (five males and ten females), aged between 22 and 26 (average: 23.80, standard deviation: 1.21), participated in the experiments of this study, which followed an experimental protocol approved by the Institutional Review Board at Tianjin University. Complying with the Declaration of Helsinki, we fully informed all subjects of all procedures, understood all possible consequences of the study, and signed an informed contest form.

Ensemble TRCA decoding algorithm. Ensemble TRCA is a supervised method and requires user calibration data to train the decoder model[38]. The model contains the individual template $\boldsymbol{\chi}^{N_f}$ calculated by averaging the inter-frequency trials and the ensemble spatial filter $\boldsymbol{U_e} = [u_1, u_2, \dots u_{N_f}]$, where $\boldsymbol{u_k}$ represents the spatial filter trained with k-th stimulus data. $\boldsymbol{u_k}$ is obtained by maximizing the covariance among inter-frequency trials. For a testing trial $\boldsymbol{x'}$, the target is identified by finding the maximum correlation coefficient between $\boldsymbol{x'U_e}$ and $\boldsymbol{\chi_k U_e}$:

$$\boldsymbol{r}(k) = \rho(\boldsymbol{x'U_e}, \boldsymbol{\chi_k U_e}) \quad (1)$$

where $\boldsymbol{r}$ is the decision vector of the testing trial $\boldsymbol{x'}$. $\rho(\boldsymbol{a}, \boldsymbol{b})$ refers to the Pearson correlation analysis between matrices $\boldsymbol{a}$ and $\boldsymbol{b}$.

Derivation of one-step decoding strategy on the memristor array. The **Eq. 1** above can be viewed as three-step computations, which can be merged into a one-step operation. Firstly, a temporal filtering for multi-channel brain signals $\boldsymbol{x_k}(k = 1,2,\dots,N_c)$ can be mathematically expressed as:

$$y_k(n) = \sum_{j=0}^{J} x_k(n-j)h(j) \tag{2}$$

Here, N_c is the number of signal channels, and $\boldsymbol{y_k}(k = 1,2,\dots,N_c)$ are the multi-channel filtered signals. J is the order of the filter, and $\boldsymbol{h}$ represents the response function of the filter. **Eq. 2** can be rewritten as follows:

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} \tag{3}$$

$$\boldsymbol{x} = [\boldsymbol{x}_1 \quad \boldsymbol{x}_2 \quad \dots \quad \boldsymbol{x}_{N_c}]^T \tag{4}$$

$$\boldsymbol{x_k} = [x_k(1) \quad x_k(2) \quad \dots \quad x_k(N_s)](k = 1,2,\dots,N_c) \tag{5}$$

$$\boldsymbol{y} = [\boldsymbol{y}_1 \quad \boldsymbol{y}_2 \quad \dots \quad \boldsymbol{y}_{N_c}]^T \tag{6}$$

$$\boldsymbol{y_k} = [y_k(1) \quad y_k(2) \quad \dots \quad y_k(N_s)](k = 1,2,\dots,N_c) \tag{7}$$

$$\boldsymbol{H} = diag(\boldsymbol{F},\boldsymbol{F},\dots,\boldsymbol{F}) \tag{8}$$

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{F_1} & \boldsymbol{O}_{(J+1)\times(N_s-J-1)} \\ & \boldsymbol{F_2} \end{bmatrix} \tag{9}$$

$$\boldsymbol{F_1} = \begin{bmatrix} h(0) & 0 & \boldsymbol{O}_{1\times(J-2)} & 0 \\ h(1) & h(0) & \boldsymbol{O}_{1\times(J-2)} & 0 \\ \vdots & \vdots & \ddots & \boldsymbol{O}_{(J-2)\times 1} \\ h(J) & h(J-1) & \cdots & h(0) \end{bmatrix} \tag{10}$$

$$\boldsymbol{F_2} = \begin{bmatrix} \boldsymbol{O}_{1\times 1} & \boldsymbol{h} & \boldsymbol{O}_{1\times(N_s-J-2)} \\ \boldsymbol{O}_{1\times 2} & \boldsymbol{h} & \boldsymbol{O}_{1\times(N_s-J-3)} \\ & & \vdots \\ & \boldsymbol{O}_{1\times(N_s-J-1)} & \boldsymbol{h} \end{bmatrix} \tag{11}$$

$$\boldsymbol{h} = [h(J) \quad h(J-1) \quad \dots \quad h(0)] \tag{12}$$

where N_s is the number of sampling points in each channel, and $\boldsymbol{O}_{m\times n}$ represents a zero matrix with m rows and n columns. In **Eq. 8**, the number of $\boldsymbol{F}$ is N_c.

Secondly, the feature (task-related component) extraction can be expressed as:

$$\boldsymbol{Z} = \boldsymbol{W}_0\boldsymbol{Y} \tag{13}$$

$$\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{z}_1 \\ \boldsymbol{z}_2 \\ \vdots \\ \boldsymbol{z}_{N_{class}} \end{bmatrix} \tag{14}$$

$$\boldsymbol{Y} = \begin{bmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \vdots \\ \boldsymbol{y}_{N_c} \end{bmatrix} \tag{15}$$

$$\boldsymbol{W}_0 = \begin{bmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \\ \vdots \\ \boldsymbol{w}_{N_{class}} \end{bmatrix} = \begin{bmatrix} w_{1,1} & \cdots & w_{1,N_c} \\ \vdots & \ddots & \vdots \\ w_{N_{class},1} & \cdots & w_{N_{class},N_c} \end{bmatrix} \tag{16}$$

$$\boldsymbol{w_k} = [w_{k,1} \quad w_{k,2} \quad \cdots \quad w_{k,N_c}](k = 1,2,\dots,N_{class}) \tag{17}$$

where $\boldsymbol{W}_0$ is the spatial filter matrix, and $\boldsymbol{w_k}(k = 1,2,\dots,N_{class})$ is the spatial filter for the k-th class signals. $\boldsymbol{z_k}(k = 1,2,\dots,N_{class})$ represents the task-related component of $\boldsymbol{y_k}(k = 1,2,\dots,N_c)$. **Eqs. 13-17** can be rewritten as follows:

$$\boldsymbol{z} = \boldsymbol{W}\boldsymbol{y} \tag{18}$$

$$\boldsymbol{W} = \begin{bmatrix} \boldsymbol{D_{1,1}} & \cdots & \boldsymbol{D_{1,N_c}} \\ \vdots & \ddots & \vdots \\ \boldsymbol{D_{N_{class},1}} & \cdots & \boldsymbol{D_{N_{class},N_c}} \end{bmatrix} \tag{19}$$

$$\boldsymbol{D_{p,q}} = diag\left(\left[w_{p,q}, w_{p,q}, \dots, w_{p,q}\right]\right)(p = 1,2,\dots,N_{class}; q = 1,2,\dots,N_c) \tag{20}$$

$$\boldsymbol{z} = [\boldsymbol{z}_1 \quad \boldsymbol{z}_2 \quad \cdots \quad \boldsymbol{z}_{N_{class}}]^T \tag{21}$$

$$\boldsymbol{z_k} = [z_k(1) \quad z_k(2) \quad \dots \quad z_k(N_s)](k = 1,2,\dots,N_{class}) \tag{22}$$

where $\boldsymbol{W}$ is the one-step implementation matrix. In **Eq. 20**, the number of $w_{p,q}$ is N_s.

Thirdly, the template matching can be expressed as:

$$l = argmax(\boldsymbol{r}) \tag{23}$$

$$\boldsymbol{r} = [r_1 \quad r_2 \quad \cdots \quad r_{N_{class}}]^T \tag{24}$$

$$r_k = \frac{(\boldsymbol{z} - \bar{\boldsymbol{z}}) \cdot (\boldsymbol{t_k} - \bar{\boldsymbol{t_k}})}{\sqrt{(\boldsymbol{z} - \bar{\boldsymbol{z}}) \cdot (\boldsymbol{z} - \bar{\boldsymbol{z}})}} \approx \frac{\boldsymbol{z} \cdot \boldsymbol{t_k}}{\sqrt{(\boldsymbol{z} - \bar{\boldsymbol{z}}) \cdot (\boldsymbol{z} - \bar{\boldsymbol{z}})}} \propto \boldsymbol{z} \cdot \boldsymbol{t_k} \tag{25}$$

$$\boldsymbol{t_k} - \bar{\boldsymbol{t_k}} = \frac{\left(\boldsymbol{t_{0,k}} - \overline{\boldsymbol{t_{0,k}}}\right)}{\sqrt{\left(\boldsymbol{t_{0,k}} - \overline{\boldsymbol{t_{0,k}}}\right) \cdot \left(\boldsymbol{t_{0,k}} - \overline{\boldsymbol{t_{0,k}}}\right)}} \tag{26}$$

where $\boldsymbol{t_{0,k}}$ and $\boldsymbol{t_k}$ represent the template and normalized template for the k-th class signals, respectively, and have the same matrix size as $\boldsymbol{z}$. r_k is the correlation coefficient between $\boldsymbol{z}$ and $\boldsymbol{t_k}$. Here $\boldsymbol{z}$ and $\boldsymbol{t_k}$ are the filtered signals, so the mean vectors $\bar{\boldsymbol{z}}$ and $\bar{\boldsymbol{t_k}}$ are roughly equal to zero vectors. **Eqs. 23-26** can be rewritten as follows:

$$l = argmax(\boldsymbol{s}) \tag{27}$$

$$\boldsymbol{s} = [s_1 \quad s_2 \quad \cdots \quad s_{N_{class}}]^T \tag{28}$$

$$\boldsymbol{s} = \boldsymbol{T}\boldsymbol{z} \tag{29}$$

$$\boldsymbol{T} = [\boldsymbol{t}_1 \quad \boldsymbol{t}_2 \quad \cdots \quad \boldsymbol{t}_{N_{class}}]^T \tag{30}$$

Finally, the above three-step operations can be merged as follows:

$$\boldsymbol{s} = \boldsymbol{M}\boldsymbol{x} \tag{31}$$

$$\boldsymbol{M} = \boldsymbol{T}\boldsymbol{W}\boldsymbol{H} \tag{32}$$

To implement it on the memristor array, we can use $\boldsymbol{G}$, $\boldsymbol{v}$ and $\boldsymbol{i}$ to represent $\boldsymbol{M}$, $\boldsymbol{x}$ and $\boldsymbol{s}$ respectively:

$$\boldsymbol{i} = \boldsymbol{G}\boldsymbol{v} \tag{33}$$

where $\boldsymbol{G}$, $\boldsymbol{v}$ and $\boldsymbol{i}$ are the differential conductance matrix, the input voltage vector, and the output current vector, respectively.

Brain-controlled drone flight experiment

Visual stimulus presentation. We presented the visual stimuli using a sampled sinusoidal stimulation method encoded with joint frequency and phase modulation strategy[33]. The stimulus sequence $\boldsymbol{s}(f, \varphi, i)$ was generated by modulating the luminance of the screen using the following equation:

$$\boldsymbol{s}(f, \varphi, i) = \frac{1}{2}\left\{1 + \sin\left[2\pi f\left(\frac{i}{RR}\right) + \varphi\right]\right\} \tag{34}$$

where f and φ are the frequency and phase of the stimulus, respectively, i indicates the frame index in the stimulus sequence, and RR represents the refresh rate of the screen. The value of $\boldsymbol{s}$ ranges from 0 to 1, where 0 represents dark, and 1 represents the highest luminance. In this study, 12 commands for 4-DOF drone flight were encoded to the frequencies and phases of stimulus as listed in **Extended Data Table 1**.

Drone actuation. This study develops a novel first-person perspective user interface (UI) that integrates a video stream from the drone into visual stimuli to enable the control of the drone's flight in 4 DOFs. The UI displays 12 stimuli corresponding to 12 flight commands, which include "take-off", "land", "hover", "keep", "clockwise", "counterclockwise" and rectilinear motions in six directions. The decoding results of brain signals are utilized to yield a flight command. Direct commands, such as "take-off", "land", "hover", and "keep" are executed immediately. In contrast, the other commands, known as motion control commands (i.e., "clockwise", "counterclockwise" and rectilinear motions), are converted into a four-element speed vector $\boldsymbol{v}_n$. The vector's elements represent pairs of opposite directions, which are "forward-back", "left-right", "up-down" and "clockwise-counterclockwise", respectively. For instance, if the currently decoded control command is "forward", then a unit speed value would be set into the speed vector $\boldsymbol{v}_n$, i.e., $\boldsymbol{v}_n = [1,0,0,0]$, where n is the speed vector index. The successive m speed vectors will be summed up and determine the movement speed in different dimensions $\boldsymbol{V}_n$ of the drone:

$$\boldsymbol{V}_n = \sum_{n-m+1}^{n} \boldsymbol{v}_n, (n > m-1) \tag{35}$$

Offline and simulated online experiments. We conducted an offline experiment to collect training samples for training a user-specific decoder of brain signals. The experiment comprised six blocks, each containing 12 trials corresponding to 12 control commands presented in a random order. Before each trial, we presented a visual cue for 0.5s that filled the target stimulus with red. Subjects were instructed to shift their gaze to the cue-indicated target as quickly as possible. Following the offset of the visual cue, all stimuli flickered for 2s, divided into five consecutive epochs of 0.4s each. We conducted a simulated online experiment comprising two blocks to evaluate the decoding performance. All stimuli flickered by repeating the 0.4s epochs within each block. To acquire labeled data for evaluating the decoding accuracy, a red-filled circle positioned beneath the simulation square served as a visual cue to guide the subjects. After every ten epochs, the visual cue changed from one stimulus to another, demanding the subjects to gaze at the cue-indicated stimulus. The simulated online experiment comprises 240 epochs, each target being cued twice.

Online brain-controlled drone flight experiment with memristor chip-based decoder. We set up the online experiment scenario in a spacious room that was well-lit, quiet, and comfortable. As shown in **Supplementary Video 1**, a ring-shape obstacle with a diameter of 0.6m and a cube with edges of 0.6m long, supported by their respective base, were placed on a blanket on the floor. We asked the subject to control the drone to complete a 3D trajectory tracking task. In the online experiment, the stimuli flicker continuously with 0.4s epochs, while the 8-channel brain signals were decoded in real-time through a memristor chip-based decoder for drone flight control. Throughout the online experiments, we carefully monitored the drone's status to ensure the safety of the subject and devices.

Performance evaluation

ITR and its calculation. The ITR is commonly used to evaluate the performance of BCI[34], and we defined it as:

$$ITR = \left[\log_2 N + P\log_2 P + (1-P)\log_2 \frac{1-P}{N-1}\right] \times \left(\frac{60}{T}\right) \quad (36)$$

where N is the number of commands, P is the accuracy and T is the time taken to make each decision. For the brain-controlled drone flight task, $T = 1.2s$ as three consecutive epochs were used to make one decision. For the co-evolution task, $T = 0.4s$ as 1 epoch was used to make one decision.

Energy benchmark. A 1×800 vector was used as the input data in the brain-controlled drone flight experiment. The decoding process computed the multiplication of the input vector and the decoder matrix, whose size is 800×12. This step consumes ~1.23 nJ for the memristor chip (as detailed in **Extended Data Table 3**). For a representative Intel® Core™ i9-12900 CPU, whose energy efficiency is about 9.5 GOPS/W, the above step consumes 800×12×2OPs/9.5GOPS/W = 2021.05 nJ. As a result, the energy consumption of the memristor chip was ~1643× lower than that of the CPU.

Benchmark of normalized decoding speed. Each 1-bit input took ~50 ns on the memristor chip. In the brain-controlled drone flight experiment, with 10-bit input data, the computation time for the task was 50ns×10 = 500ns. With the chip area being 0.0579 mm^2 (as detailed in **Extended Data Table 3**), we estimated the normalized decoding speed (i.e., the number of decoding decisions per unit time per unit area) as 1/500 ns/0.0579 mm^2 = 34.54 decisions·μs^{-1}·mm^{-2}. For comparison, we still used the Intel® Core™ i9-12900 CPU with a chip area of 200 mm^2 and computing power of 600 GOPS; therefore, the computation time by CPU was 800×12×2/600 GOPS = 3.2 ns for the task. We then estimated the normalized decoding speed as 1/3.2 ns/200 mm^2 = 0.16 decisions·μs^{-1}·mm^{-2}. Consequently, regarding the normalized decoding speed, our memristor chip showed an advantage of approximately 216 times over the CPU.

Brain-memristor decoder co-evolution experiment

ErrP calibration experiment. To train the ErrP detector, we asked subjects to first participate in a calibration experiment to collect target and non-target brain signals as training samples. We modified the user interface from the previous drone flight control experiment and used the same basic stimuli settings. The experiment comprised five blocks, each containing 36 trials (i.e., going through all 12 stimuli three times). A 1.5s break occurred between two consecutive trials, during which we presented a visual cue to guide the subject to switch gaze to the next stimulation. Each trial had 10 stimulation epochs, with visual feedback presented after the offset of each stimulation epoch. There was a 20% chance that the visual feedback was inconsistent with the stimulus indicated by the visual cue, constituting an error occurrence. The brain signals collected during error occurrence epochs were labeled as targets. Conversely, the remaining 80% of epochs classified as "correct" were labeled as non-targets. All the target and non-target trials were collected to train the ErrP detector.

Data preprocessing. The signals collected from the calibration experiment were firstly re-referenced by averaging TP7 and TP8 channels and then extracted into trials with the interval [-400, 600] ms that was time-locked to the feedback onset. These trials were filtered between [1, 10] Hz using a Chebyshev Type I infinite impulse response filter.

ErrP detector training and online ErrP detection. This study used the discriminate canonical pattern matching[45] (DCPM) to detect ErrP features. DCPM comprises three key components: the construction of discriminative spatial patterns (DSPs), the construction of CCA patterns, and pattern matching. We used the signals as a training set to differentiate between target and non-target trials. We calculated each class's DSP spatial filter and template using the training data. Next, we applied these DSP spatial filters to the test trial and templates to enhance distinguishability after transformation. Then, we used the CCA algorithm to reveal the underlying correlation between the processed test trial and the templates. We determined the test trial's class by comparing its similarities with the

training template. We used true positive rate (TPR), true negative rate (TNR), and balanced accuracy (BACC) to evaluate the performance of our detector (**Extended Data Fig. 4**). The first two metrics represented the chances of successful predictions in the "target" class (Error, positive sample) and the "non-target" class (Correct, negative sample), respectively. The BACC was for evaluating the learning process with class imbalance. We defined them as follows:

$$TPR = \frac{TP}{TP + FN} \tag{37}$$

$$TNR = \frac{TN}{TN + FP} \tag{38}$$

$$BACC = \frac{TPR + TNR}{2} \tag{39}$$

where TP, FN, TN and FN are the number of true positive, false negative, true negative, and false negative samples, respectively.

Long-term brain-memristor decoder co-evolution experiment. Before conducting the HLU experiment, we conducted a separate calibration experiment to train the SSVEP decoder, following the same procedure as the offline experiment in the above brain-controlled drone flight experiment. In the HLU experiment, we asked the subjects to complete the cue-following task, which contained ten online blocks and lasted ~6 hours. For each block, the online experiment paradigm remained the same as the ErrP calibration experiment. However, the visual feedback of each 8-channel SSVEP epoch was determined by the online memristor chip decoding result. We synchronized the onset of visual feedback with brain signals to extract the ErrP epoch. Using the ErrP epoch, one could evaluate the subject's response to the visual feedback of the SSVEP epoch, i.e., whether it was correct. We considered the SSVEP epoch to be correctly decoded when the ErrP detector did not identify an ErrP response. Subsequently, we stored the signals of that epoch and used them to update the memristor decoder.

Estimation of brain control state. During co-evolution experiments, we monitored the SSVEP component in the participants' brain signals, considering it to indicate their brain control state. We collected online brain signals to form a test sample set at each update. We used the FBCCA method on this set to predict sample labels. The accuracy of the FBCCA represented the current brain control state. High accuracy suggested a stable and effective brain control state, while low accuracy indicated an ineffective control state. We could observe changes in the brain control state by monitoring the classification accuracy over time. This provided insights into the evolution of brain control ability and its effect on the performance of the co-evolution BCI.

Estimation of the memristor decoder's effectiveness. We generated a set of test signals to track the memristor decoder's effectiveness throughout the co-evolution experiment. Specifically, the online data for each update is denoted as $x_{n,f}^{s}$, where n represents the update index, f is the index of target and s is the subject index. We defined the individual template for each subject as

$$\overline{X}_f^s = \sum_{n=1}^{K} x_n^s \tag{40}$$

where K represents the total number of updates. The sin-cos references signals Y_f can be represented as follows:

$$Y_f = \begin{bmatrix} sin(2\pi ft) \\ \cos(2\pi ft) \\ \vdots \\ sin(2\pi N_h ft) \\ \cos(2\pi N_h ft) \end{bmatrix} \tag{41}$$

where f is the stimulation frequency and N_h is the number of harmonics. By applying the source aliasing matrix estimation method[46], the estimate aliasing matrix $\widehat{\boldsymbol{\Phi}}$ can be obtained by minimizing the mean squared error between $\overline{X}_f^s$ and Y_f. The test signal $\tilde{X}_f^s$ can be constructed as follows:

$$\tilde{X}_f^s = \widehat{\mathbf{\Phi}}\overline{X}_f^s + \alpha W_{noise}^m \quad (42)$$

where W_{noise}^m is random Gaussian noise, $m \in 1,2,\ldots,N_a$, and N_a is the number of constructed signals. The parameter α controls the intensity of noise. Subsequently, these test signals were decoded using the memristor decoder obtained from each update, and the decoding accuracy served as an indicator of the decoder's performance at each update. We repeated the above process ten times to obtain statistically meaningful results, resulting in the average accuracy that represents the decoder's effectiveness. In this study, K is 10, N_h was set to 5, N_a was set to 5 and α was set to 1.8.

Reference

1 Willett, F. R., Avansino, D. T., Hochberg, L. R., Henderson, J. M. & Shenoy, K. V. High-performance brain-to-text communication via handwriting. *Nature* **593**, 249-254 (2021). https://doi.org:10.1038/s41586-021-03506-2

2 Anumanchipalli, G. K., Chartier, J. & Chang, E. F. Speech synthesis from neural decoding of spoken sentences. *Nature* **568**, 493-498 (2019). https://doi.org:10.1038/s41586-019-1119-1

3 Metzger, S. L. *et al.* A high-performance neuroprosthesis for speech decoding and avatar control. *Nature* **620**, 1037-1046 (2023). https://doi.org:10.1038/s41586-023-06443-4

4 Willett, F. R. *et al.* A high-performance speech neuroprosthesis. *Nature* **620**, 1031-1036 (2023). https://doi.org:10.1038/s41586-023-06377-x

5 Flesher, S. N. *et al.* A brain-computer interface that evokes tactile sensations improves robotic arm control. *Science* **372**, 831-836 (2021).

6 Lorach, H. *et al.* Walking naturally after spinal cord injury using a brain–spine interface. *Nature*, 1-8 (2023).

7 Silversmith, D. B. *et al.* Plug-and-play control of a brain–computer interface through neural map stabilization. *Nature Biotechnology* **39**, 326-335 (2021). https://doi.org:10.1038/s41587-020-0662-5

8 Chiang, K.-J., Wei, C.-S., Nakanishi, M. & Jung, T.-P. Boosting template-based SSVEP decoding by cross-domain transfer learning. *Journal of Neural Engineering* **18**, 016002 (2021). https://doi.org:10.1088/1741-2552/abcb6e

9 Pascual-Leone, A., Amedi, A., Fregni, F. & Merabet, L. B. The plastic human brain cortex. *Annual review of neuroscience* **28**, 377-401 (2005). https://doi.org:10.1146/annurev.neuro.27.070203.144216

10 Luppi, A. I. *et al.* A synergistic core for human brain evolution and cognition. *Nature Neuroscience* **25**, 771-782 (2022). https://doi.org:10.1038/s41593-022-01070-0

11 Abbott, L. F. & Nelson, S. B. Synaptic plasticity: taming the beast. *Nature Neuroscience* **3**, 1178-1183 (2000). https://doi.org:10.1038/81453

12 Sur, M. & Rubenstein, J. L. R. Patterning and Plasticity of the Cerebral Cortex. *Science* **310**, 805-810 (2005). https://doi.org:doi:10.1126/science.1112070

13 Holz, E. M., Botrel, L., Kaufmann, T. & Kübler, A. Long-Term Independent Brain-Computer Interface Home Use Improves Quality of Life of a Patient in the Locked-In State: A Case Study. *Archives of Physical Medicine and Rehabilitation* **96**, S16-S26 (2015). https://doi.org:https://doi.org/10.1016/j.apmr.2014.03.035

14 Sussillo, D., Stavisky, S. D., Kao, J. C., Ryu, S. I. & Shenoy, K. V. Making brain–machine interfaces robust to future neural variability. *Nature Communications* **7**, 13749 (2016). https://doi.org:10.1038/ncomms13749

15 Wen, S. *et al.* Rapid adaptation of brain–computer interfaces to new neuronal ensembles or participants via generative modelling. *Nature Biomedical Engineering* **7**, 546-558 (2023). https://doi.org:10.1038/s41551-021-00811-z

16 Renart, A. & Machens, C. K. Variability in neural activity and behavior. *Current Opinion in Neurobiology* **25**, 211-220 (2014). https://doi.org:https://doi.org/10.1016/j.conb.2014.02.013

17 van Praag, H., Kempermann, G. & Gage, F. H. Neural consequences of enviromental enrichment. *Nature Reviews Neuroscience* **1**, 191-198 (2000). https://doi.org:10.1038/35044558

18 Wolpaw, J. R. Brain–computer interfaces as new brain output pathways. *The Journal of Physiology* **579**, 613-619 (2007). https://doi.org:https://doi.org/10.1113/jphysiol.2006.125948

19 Musk, E. An Integrated Brain-Machine Interface Platform With Thousands of Channels. *J Med Internet Res* **21**, e16194 (2019). https://doi.org:10.2196/16194

20 Sahasrabuddhe, K. *et al.* The Argo: a high channel count recording system for neural recording in vivo. *Journal of Neural Engineering* **18**, 015002 (2021). https://doi.org:10.1088/1741-2552/abd0ce

21 Steinmetz, N. A. *et al.* Neuropixels 2.0: A miniaturized high-density probe for stable, long-term brain recordings. *Science* **372** (2021). https://doi.org:10.1126/science.abf4588

22 Grover, P. & Venkatesh, P. An information-theoretic view of EEG sensing. *Proceedings of the IEEE* **105**, 367-384 (2016).

23 Han, J. *et al.* A high-speed hybrid brain-computer interface with more than 200 targets. *Journal of Neural Engineering* (2023).

24 Chen, Y. *et al.* Implementing a calibration-free SSVEP-based BCI system with 160 targets. *Journal of Neural Engineering* **18**, 046094 (2021).

25 Horowitz, M. in *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*. 10-14 (IEEE).

26 Williams, R. S. What's Next? [The end of Moore's law]. *Computing in Science & Engineering* **19**, 7-13 (2017). https://doi.org:10.1109/MCSE.2017.31

27 Ambrogio, S. *et al.* Equivalent-accuracy accelerated neural-network training using analogue memory. *Nature* **558**, 60-67 (2018). https://doi.org:10.1038/s41586-018-0180-5

28 Wan, W. *et al.* A compute-in-memory chip based on resistive random-access memory. *Nature* **608**, 504-512 (2022). https://doi.org:10.1038/s41586-022-04992-8

29 Xia, Q. & Yang, J. J. Memristive crossbar arrays for brain-inspired computing. *Nature Materials* **18**, 309-323 (2019). https://doi.org:10.1038/s41563-019-0291-x

30 Guo, Z. *et al.* An Emerging NVM CIM Accelerator with Shared-Path Transpose Read and Bit-Interleaving Weight Storage for Efficient On-Chip Training in Edge Devices. *IEEE Transactions on Circuits and Systems II: Express Briefs* (2023).

31 Gao, B. *et al.* Memristor-based analogue computing for brain-inspired sound localization with in situ training. *Nat Commun* **13**, 2026 (2022). https://doi.org:10.1038/s41467-022-29712-8

32 Zhang, W. *et al.* Edge learning using a fully integrated neuro-inspired memristor chip. *Science* **381**, 1205-1211 (2023). https://doi.org:10.1126/science.ade3483

33 Chen, X. *et al.* High-speed spelling with a noninvasive brain–computer interface. *Proceedings of the National Academy of Sciences* **112**, E6058-E6067 (2015). https://doi.org:doi:10.1073/pnas.1508080112

34 Mahmood, M. *et al.* Fully portable and wireless universal brain–machine interfaces enabled by flexible scalp electronics and deep learning algorithm. *Nature Machine Intelligence* **1**, 412-422 (2019). https://doi.org:10.1038/s42256-019-0091-7

35 Chiang, K.-J. *et al.* in *2021 10th International IEEE/EMBS Conference on Neural Engineering (NER).* 487-490 (IEEE).

36 Chavarriaga, R., Sobolewski, A. & Millán, J. d. R. Errare machinale est: the use of error-related potentials in brain-machine interfaces. *Frontiers in neuroscience*, 208 (2014).

37 Chavarriaga, R. & Millan, J. d. R. Learning From EEG Error-Related Potentials in Noninvasive Brain-Computer Interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **18**, 381-388 (2010). https://doi.org:10.1109/TNSRE.2010.2053387

38 Nakanishi, M. *et al.* Enhancing Detection of SSVEPs for a High-Speed Brain Speller Using Task-Related Component Analysis. *IEEE transactions on bio-medical engineering* **65**, 104-112 (2018). https://doi.org:10.1109/tbme.2017.2694818

39 Gulati, T., Guo, L., Ramanathan, D. S., Bodepudi, A. & Ganguly, K. Neural reactivations during sleep determine network credit assignment. *Nat Neurosci* **20**, 1277-1284 (2017). https://doi.org:10.1038/nn.4601

40 Müller, J. S. *et al.* A mathematical model for the two-learners problem. *Journal of Neural Engineering* **14**, 036005 (2017). https://doi.org:10.1088/1741-2552/aa620b

41 Orsborn, A. L. *et al.* Closed-loop decoder adaptation shapes neural plasticity for skillful neuroprosthetic control. *Neuron* **82**, 1380-1393 (2014). https://doi.org:10.1016/j.neuron.2014.04.048

42 Madduri, M. M., Burden, S. A. & Orsborn, A. L. Biosignal-based co-adaptive user-machine interfaces for motor control. *Current Opinion in Biomedical Engineering* **27**, 100462 (2023). https://doi.org:https://doi.org/10.1016/j.cobme.2023.100462

43 Shanechi, M. M. *et al.* Rapid control and feedback rates enhance neuroprosthetic control. *Nature Communications* **8**, 13825 (2017). https://doi.org:10.1038/ncomms13825

44 Ahmadipour, P., Yang, Y., Chang, E. F. & Shanechi, M. M. Adaptive tracking of human ECoG network dynamics. *Journal of Neural Engineering* **18**, 016011 (2021).

45 Xiao, X. *et al.* Discriminative canonical pattern matching for single-trial classification of ERP components. *IEEE Transactions on Biomedical Engineering* **67**, 2266-2275 (2019).

46 Luo, R. *et al.* Data Augmentation of SSVEPs Using Source Aliasing Matrix Estimation for Brain-Computer Interfaces. *IEEE transactions on bio-medical engineering* **70**, 1775-1785 (2023). https://doi.org:10.1109/tbme.2022.3227036

Acknowledgements. This work was in part supported by the STI 2030-Major Projects 2022ZD0210200, National Natural Science Foundation of China (Nos. 92264201, 62025111, 82330064, 62122059, 81925020, 62206198), the XPLORER Prize, the Seed Fund for Basic Research for New Staff (No. 102010271) from the University of Hong Kong, and the Center of Nanofabrication, Tsinghua University.

Author contributions. J.T., M.X., D.M., and H.W. supervised the project. Z.L., J.T., and H.W. conceived the idea. Z.L., J.M., J.T., and M.X. designed the experiments and wrote the manuscript. Q.L., J.T., B.G., H.Q., and H.W. contributed to the design and fabrication of the memristor chip. Z.L., S.D., and Q.Q. designed the chip test system. Z.L., Y.X., and Y.L. tested and characterized the memristor devices. J.M., K.W., and W.C. designed the stimulus interfaces for BCI experiments. J.M. collected brain signals for the simulated online experiments. Z.L. optimized and implemented the decoding models on memristor chips. Z.L. and J.M. performed the memristor chip-based brain-controlled drone flight experiments. Z.L., J.M., and W.C. performed the brain-memristor decoder coevolution experiments. P.Y., H.Z., N.W., B.H., and T.-P.J. discussed the results and benchmark. All authors commented on the manuscript.

Competing interests. The authors declare no competing interests.

Correspondence. Correspondence requests should be addressed to J.T., M.X., D.M., and H.W.

Extended Data:

Extended Data Figure 1 | Analog switching characteristics of memristors.

Extended Data Figure 2 | Binary masks for parameter prune experiments.

Extended Data Figure 3 | Results of SU experiments for brain-memristor co-evolution.

Extended Data Figure 4 | Overall performance of ErrP detector.

Extended Data Figure 5 | Accuracy evolution comparison for BCIs with and without co-evolution for 10 subjects.

Extended Data Figure 6 | Accuracy changes for each control command in BCIs without co-evolution for 10 subjects.

Extended Data Figure 7 | Accuracy changes for each control command in BCIs with co-evolution for 10 subjects.

Extended Data Figure 8 | Similarity evolution of brain signals under different update numbers for 10 subjects.

Extended Data Figure 9 | Adaptive memristor decoder maps under different update cycles for 10 subjects.

Extended Data Figure 10 | Correlation of decoder maps for 10 subjects.

Extended Data Figure 11 | The fitted slopes of all eight channels for subject B1.

Extended Data Figure 12 | The evolution of the positions of top 15% parameters in the adaptive decoders for subject B1.

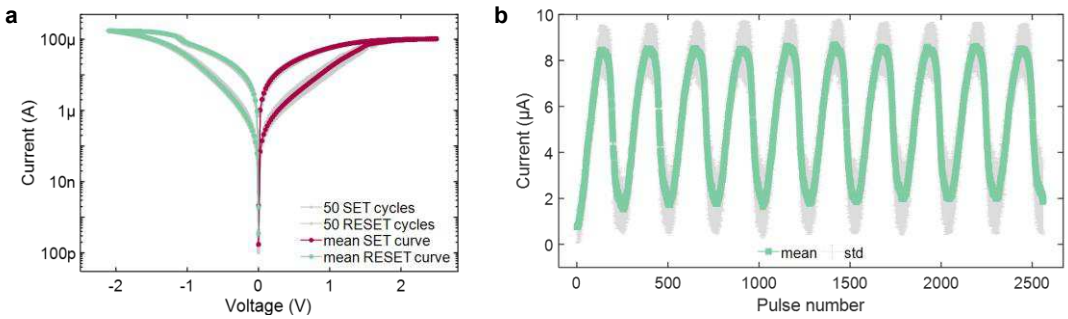
Extended Data Figure 13 | Distribution evolution of decoder parameter magnitudes for 10 subjects.

Extended Data Table 1 | Command encoding for brain-controlled drone flight.

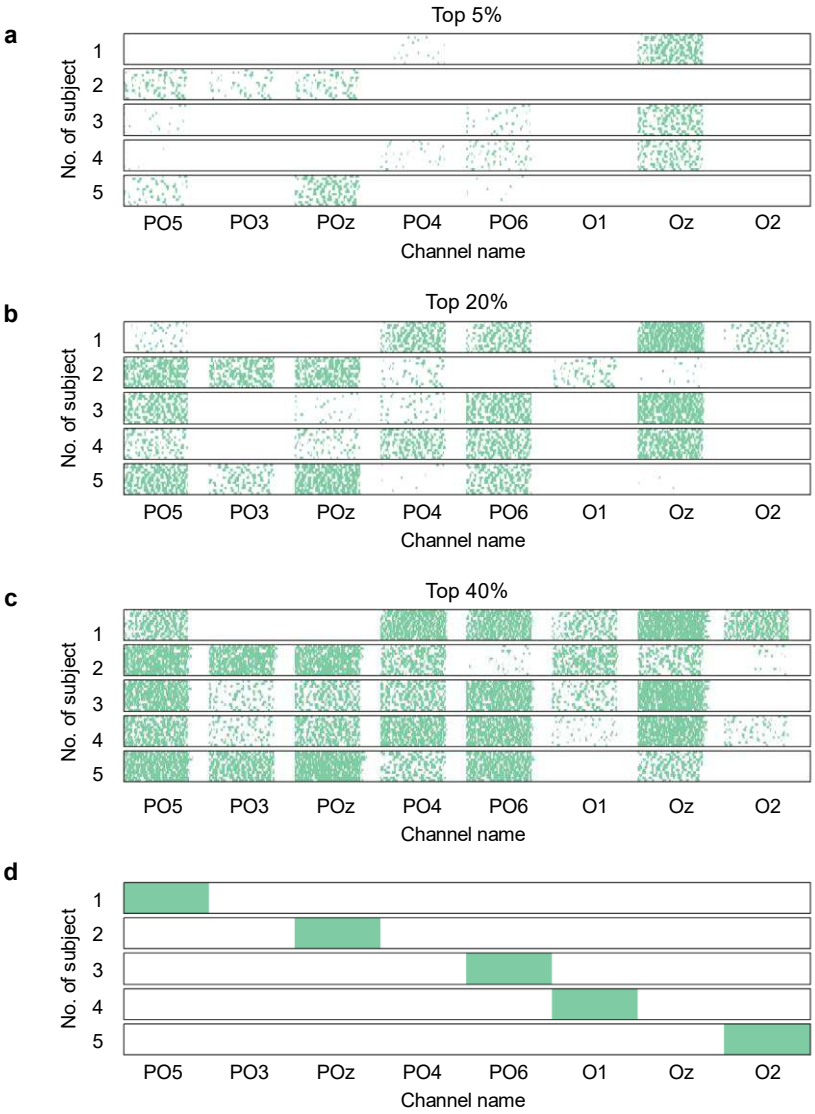
878 Extended Data Table 2 | Decoding results with CPU and memristor chip for 5 different
879 subjects.

880 Extended Data Table 3 | Energy consumption benchmark for memristor chip-based
881 decoding.

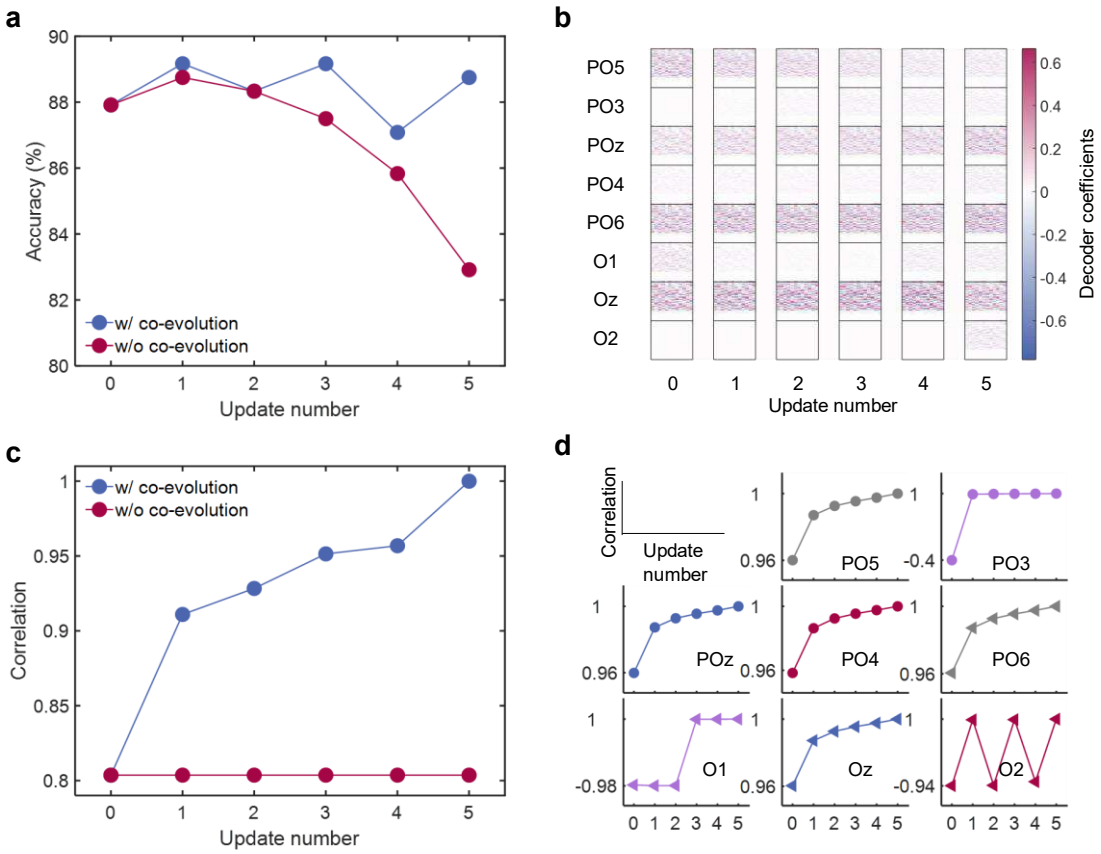
882 Extended Data Table 4 | Decoding results for experiments with masks on decoder
883 parameters.



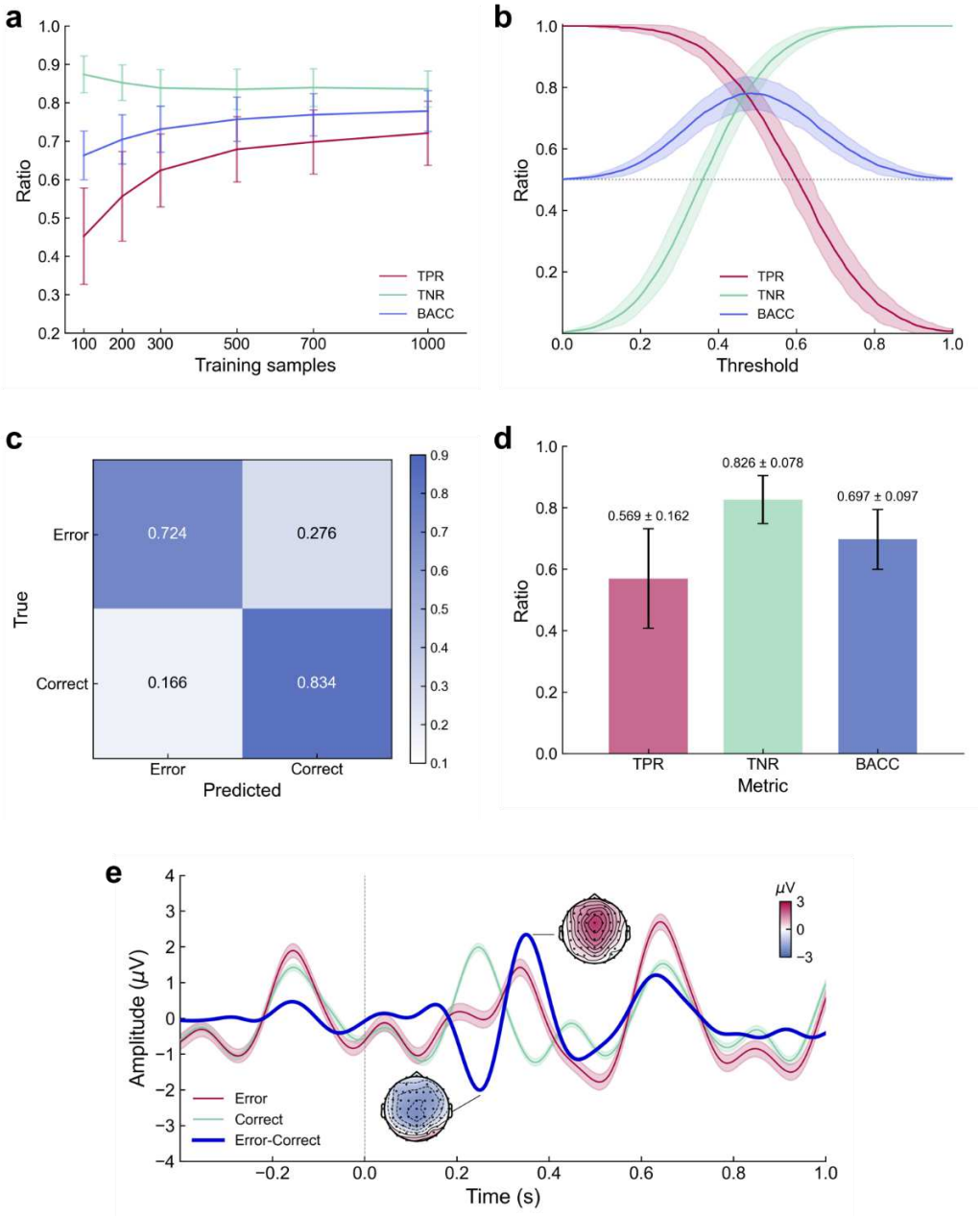
Extended Data Figure 1 | Analog resistive switching characteristics of memristors. a, Direct-current (DC) I-V curves of a typical memristor device. The red and green lines plot the mean values of 50 consecutive switching cycles (grey lines). **b,** Analog resistive switching characteristics for 128 representative memristor devices under 10 consecutive pulsed SET and RESET cycles. The read voltage is fixed at 0.2 V.



Extended Data Figure 2 | Binary masks for parameter pruning experiments. a, Mask for parameters whose magnitudes are at the top 5%, **b,** Mask for parameters whose magnitudes are at the top 20%, **b,** Mask for parameters whose magnitudes are at the top 40%, **d,** Mask examples for single-channel decoding experiments. Here the activated channels for subjects A1-A5 are channels PO5, POz, PO6, O1 and O2, respectively.

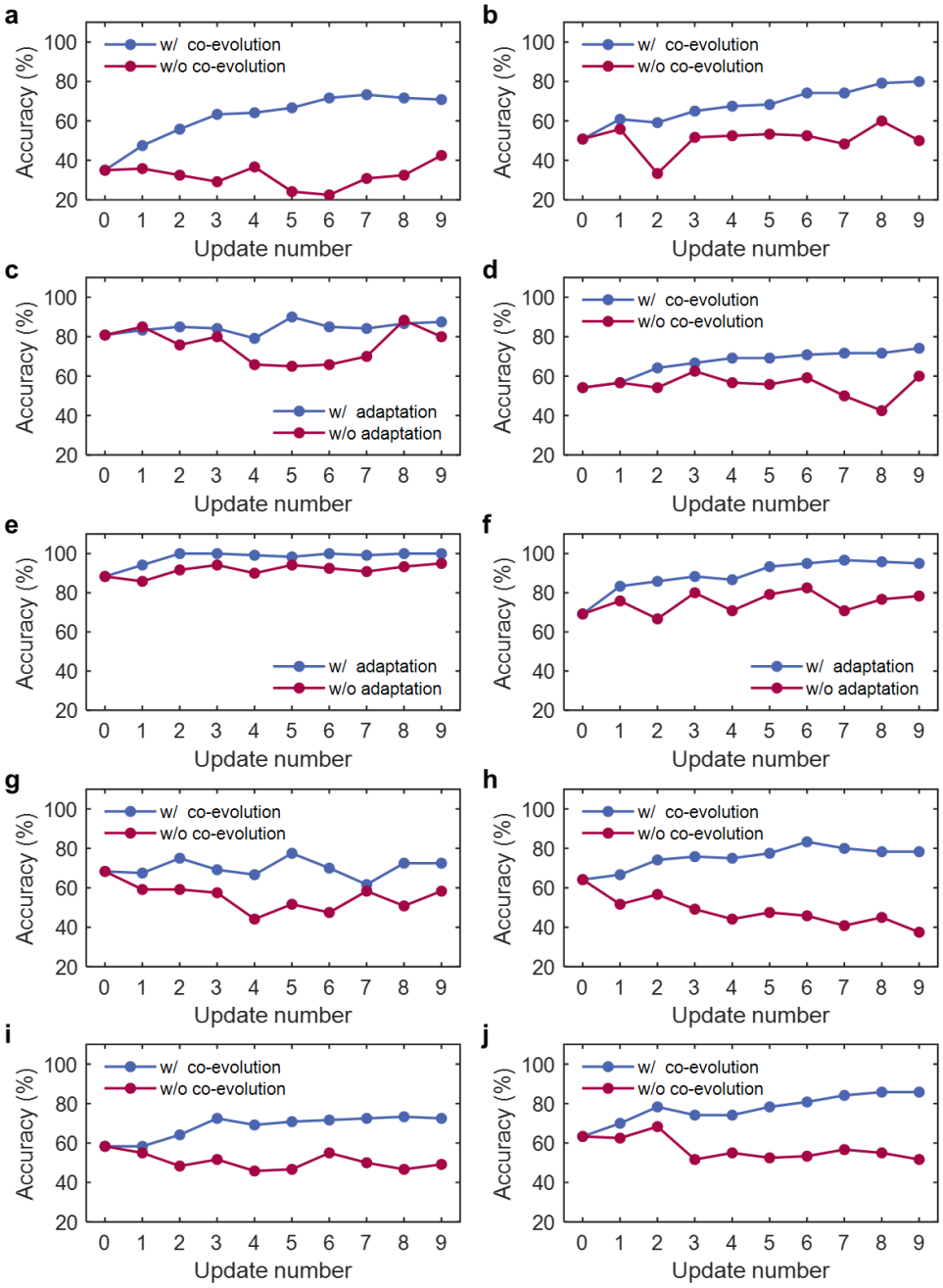


Extended Data Figure 3 | Results of SU experiments for brain-memristor decoder co-evolution. a, Decoding accuracies over 5 update cycles for the BCIs with (w/) and without (w/o) co-evolution. **b,** Adaptive memristor decoder map after each update cycle. **c,** Correlation coefficients between each decoder map and the final one. **d**, Correlation coefficients between the decoder maps for each channel.

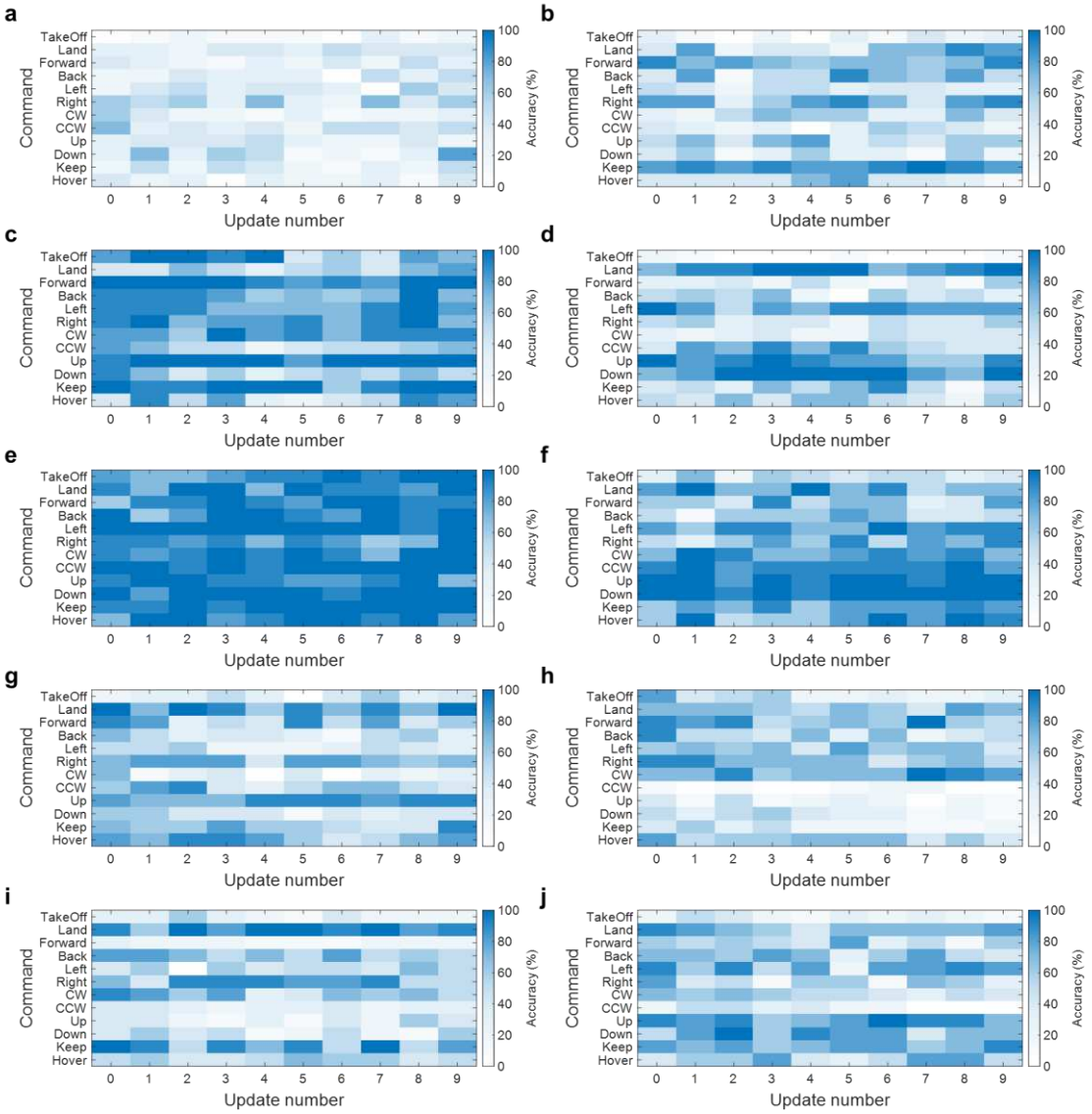


Extended Data Figure 4 | Overall performance of the ErrP detector. a, Average TPR, TNR and BACC as a function of the number of samples in the training set, and the testing set contains 360 samples. The samples in the training set and testing set are randomly selected from the data collected in the ErrP calibration experiment without overlap. **b**, Average TPR, TNR and BACC as a function of the classification threshold. If the feature value of the DCPM-based classifier exceeds a predefined threshold, it labels the classification of "target"; otherwise, it labels the classification of "non-target". These metrics are calculated from the leave-one-block-out cross-validation on the training

data. **c**, Confusion matrix between the “target” class and the “non-target” class when the threshold is set as 0.5 in **b**. **d**, Grand averaged metrics on all blocks of online data. The error bars in **a** and **d**, and the shaded areas in **b** indicate the standard deviation. **e**, Correct and incorrect evoked signals from channel FCZ (green and red lines, respectively).



Extended Data Figure 5 | Accuracy evolution for BCIs with and without co-evolution for 10 subjects. a-j, Decoding accuracy results for subjects B1-B10, respectively.

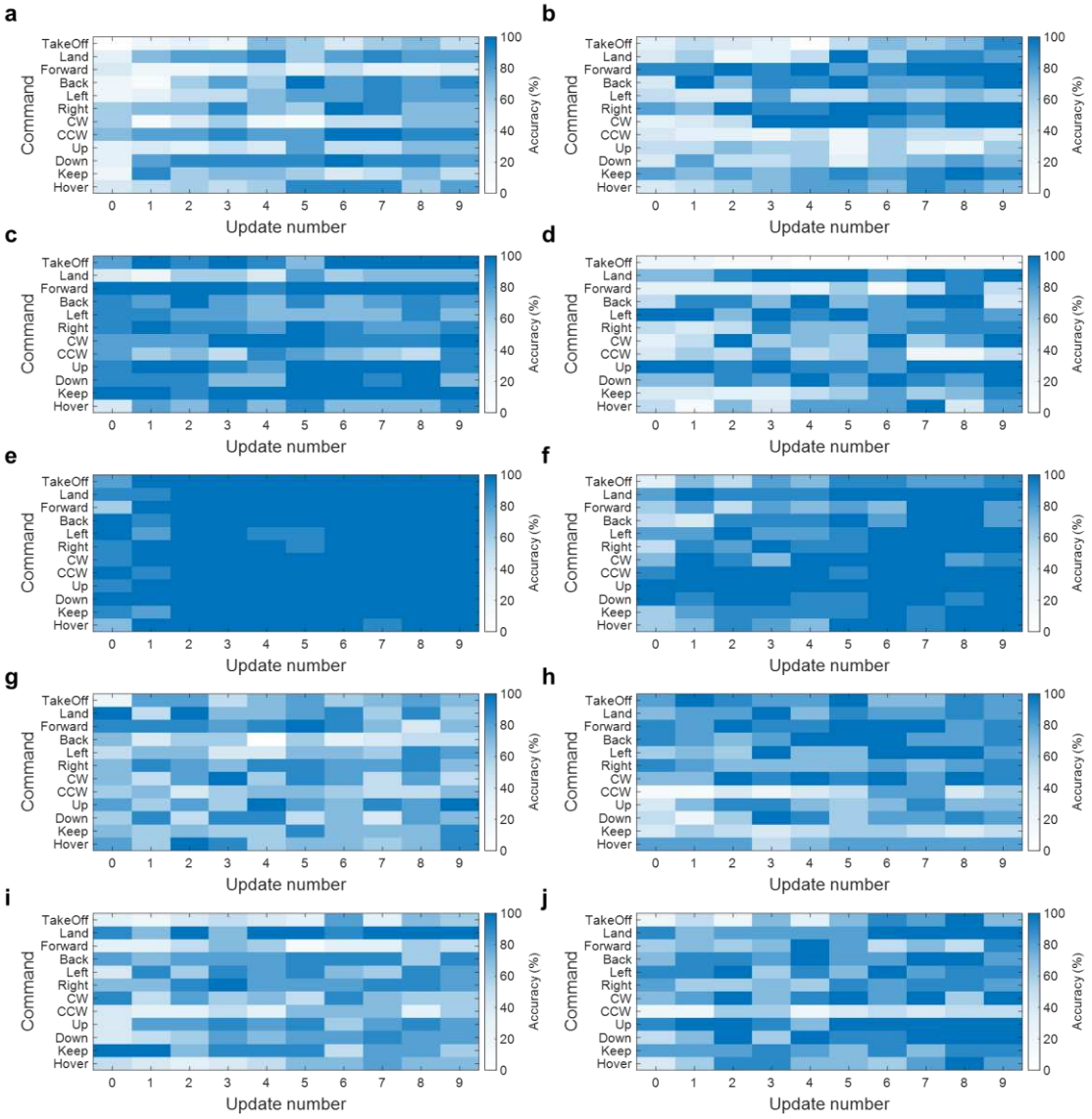


927

928 **Extended Data Figure 6 | Accuracy changes for each control command in the BCI**

929 **without co-evolution for 10 subjects. a-j,** Accuracies for subjects B1-B10, respectively.

930

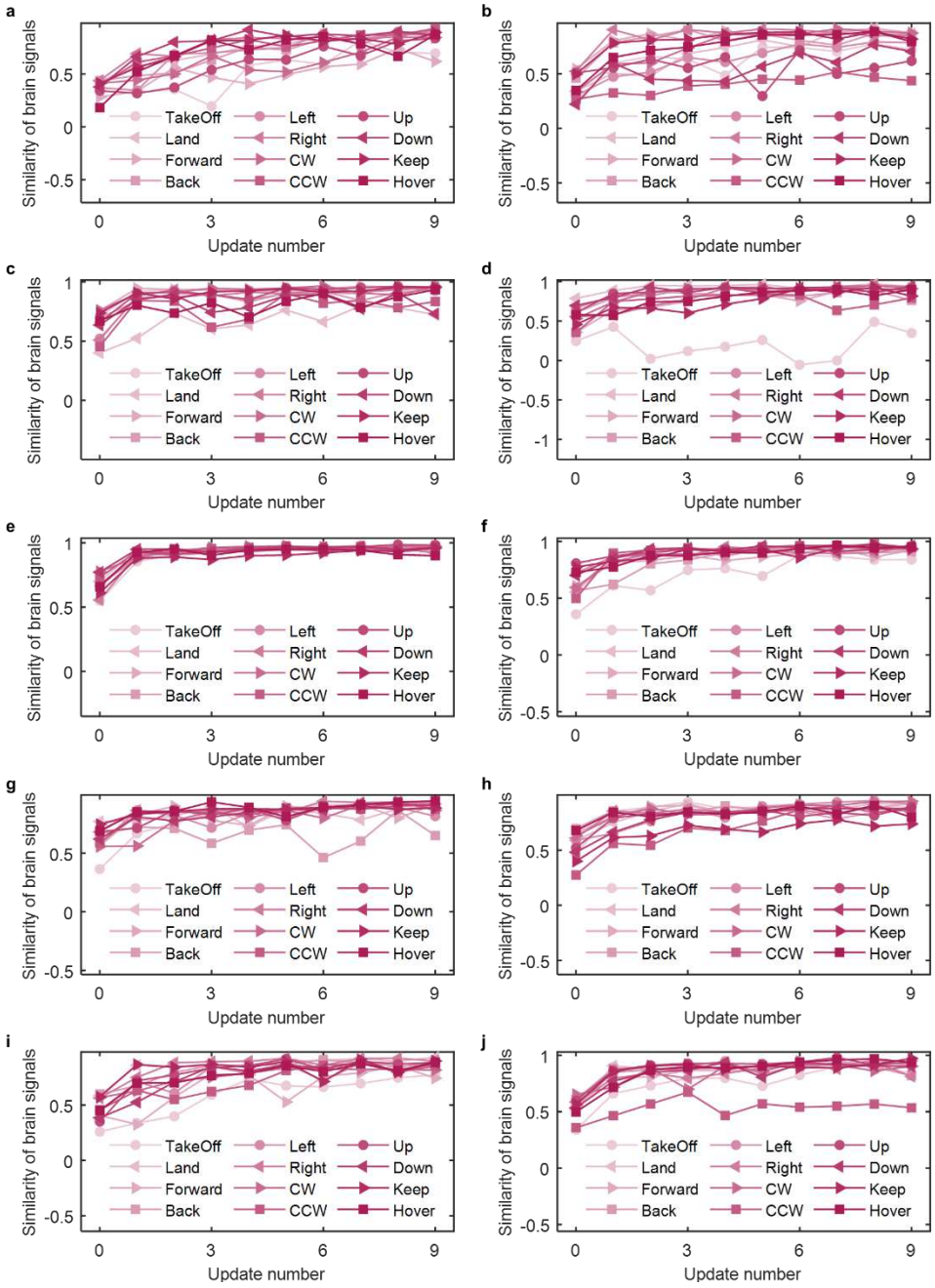


931

932 **Extended Data Figure 7 | Accuracy changes for each control command in the BCI**

933 **with co-evolution for 10 subjects. a-j,** Accuracies for subjects B1-B10, respectively.

934



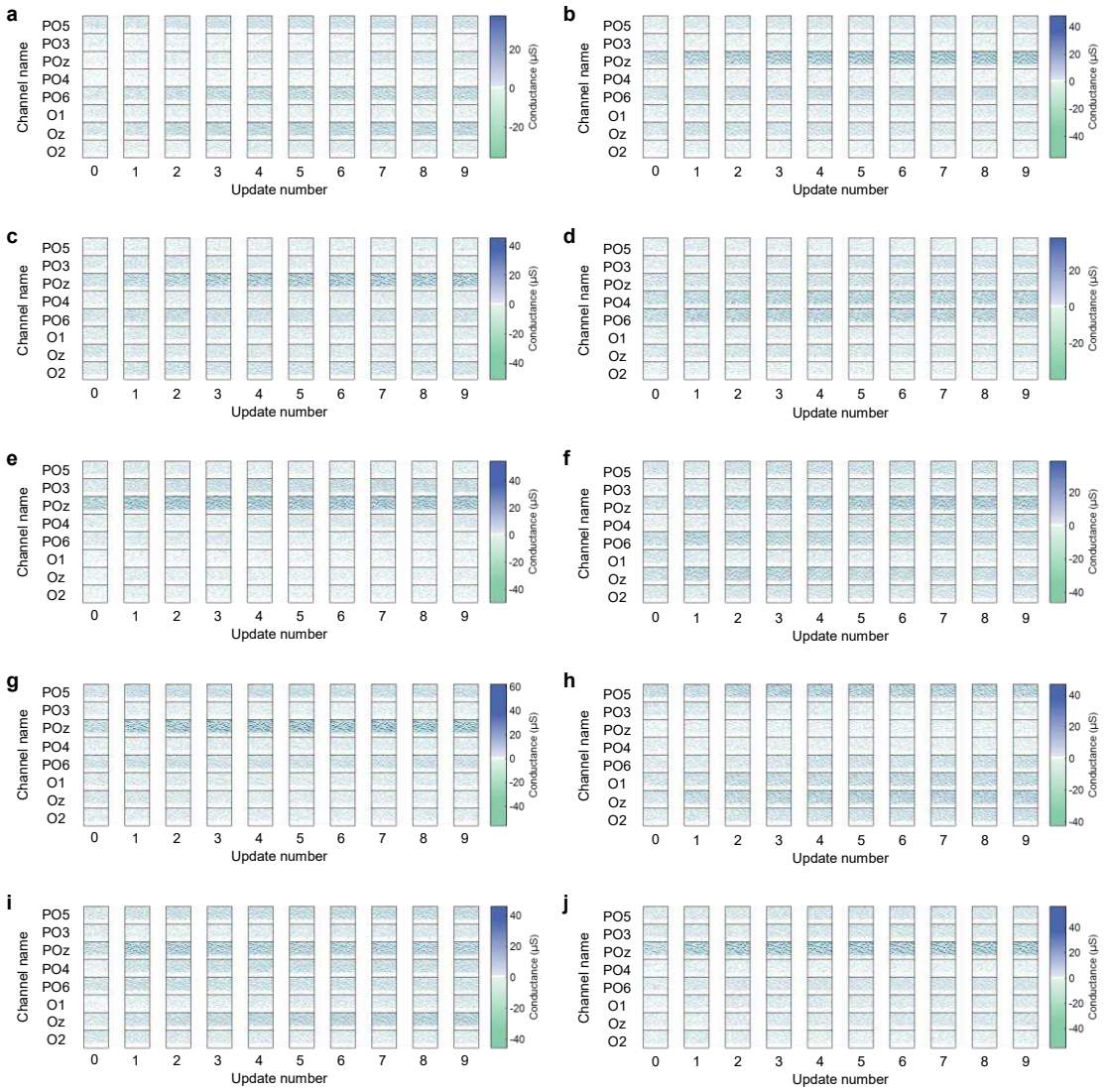
935

936 **Extended Data Figure 8 | Similarity evolution of brain signals during consecutive**

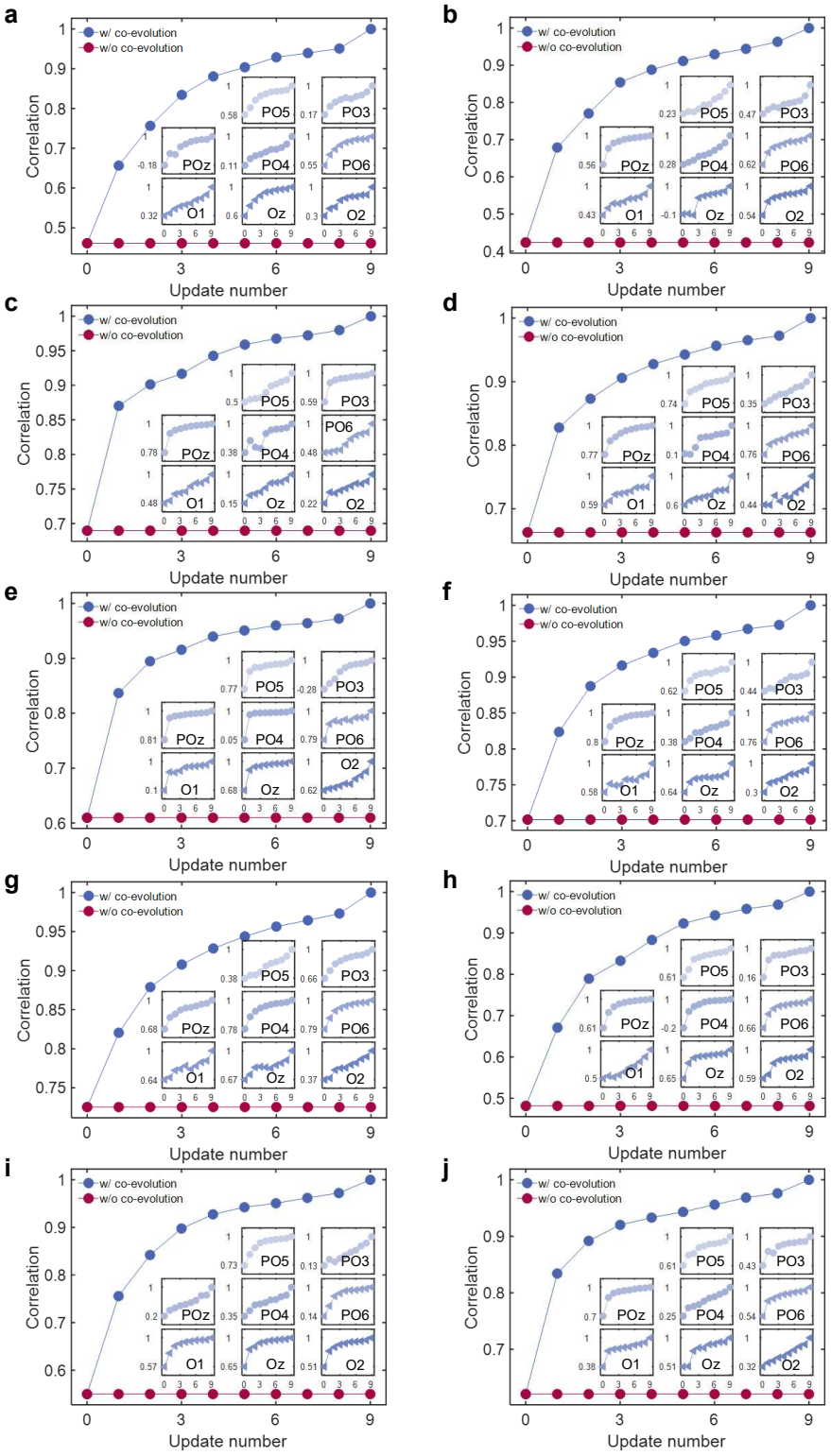
937 **decoder updates. a-j,** Similarity evolution in the brain signals for each control command

938 for subjects B1-B10, respectively.

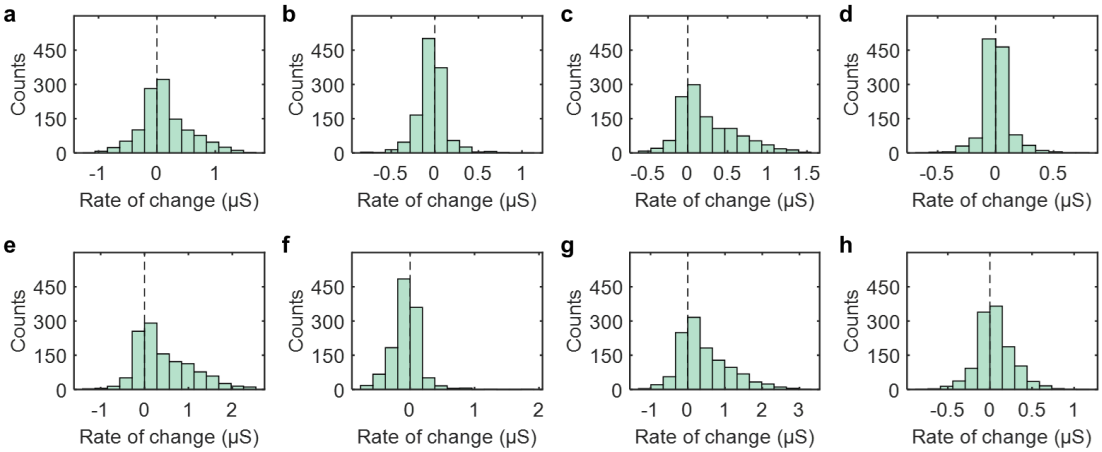
939



Extended Data Figure 9 | Adaptive memristor decoder maps under different update cycles for 10 subjects. a-j, Memristor decoders for subjects B1-B10, respectively.



Extended Data Figure 10 | Correlation of adaptive decoder maps for 10 subjects. a-j, Correlation evolution for subjects B1-B10, respectively.



947

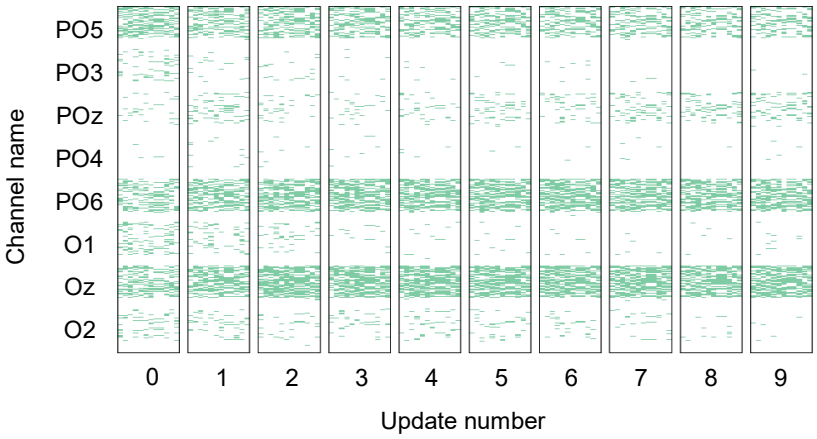
948 **Extended Data Figure 11 | The fitted slopes of all eight channels for subject B1. a-h,**

949 Distribution of slopes for channels PO5, PO3, POz, PO4, PO6, O1, Oz, and O2 respectively.

950 The slope represents the change of decoder parameters per update during the co-

951 evolution.

952

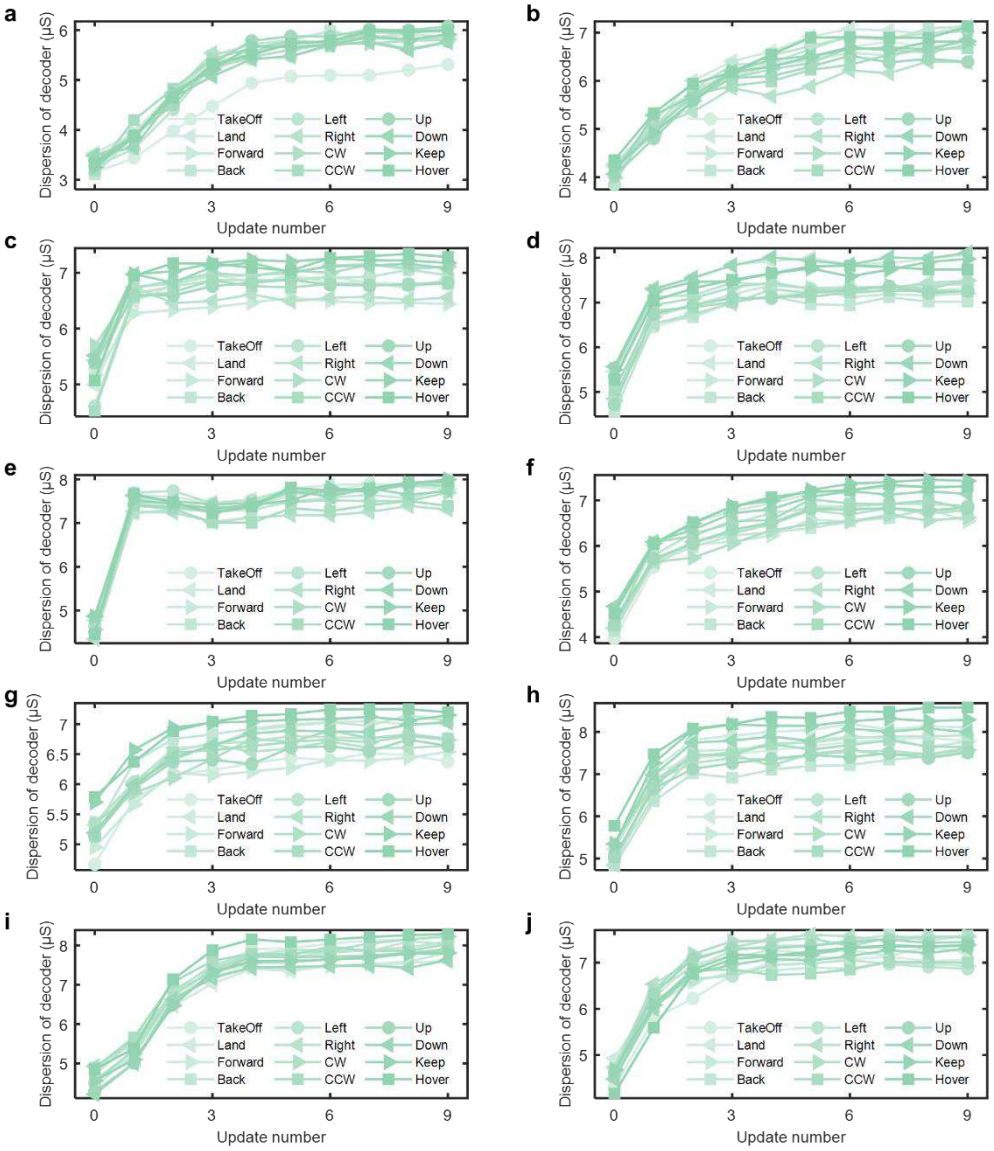


953

954 **Extended Data Figure 12 | The evolution of the positions of top 15% parameters of**

955 **the adaptive decoders for subject B1.**

956



957

958 **Extended Data Figure 13 | Distribution evolution of decoder parameter magnitudes**

959 **for 10 subjects. a-j,** Evolution of the dispersion (standard derivation) of the decoder

960 parameters for subjects B1-B10, respectively.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- SupplementaryvideoMemristorenabledneuromorphicBCI.mp4
- ExtendedDataTables.pdf