

Article

Genetic Algorithm-Based Data Optimization for Efficient Transfer Learning in Convolutional Neural Networks: A Brain–Machine Interface Implementation

Goragod Pongthaisorn ¹  and Genci Capi ^{2,*}

¹ Graduate School of Science and Engineering, Hosei University, Tokyo 184-8584, Japan; goragod.pongthaisorn.3w@stu.hosei.ac.jp

² Department of Mechanical Engineering, Hosei University, Tokyo 184-8584, Japan

* Correspondence: capi@hosei.ac.jp

Abstract: In brain–machine interface (BMI) systems, the performance of trained Convolutional Neural Networks (CNNs) is significantly influenced by the quality of the training data. Another issue is the training time of CNNs. This paper introduces a novel approach by combining transfer learning and a Genetic Algorithm (GA) to optimize the training data of CNNs. Transfer learning is implemented across different subjects, and the data chosen by GA aim to improve CNN performance. In addition, the GA-selected data shed light on the similarity in brain activity between subjects. Two datasets are used: (1) the publicly available BCI Competition IV, in which the subjects performed motor imagery (MI) tasks, and (2) the dataset created by healthy subjects of our laboratory performing motor movement (MO) tasks. The experimental results indicate that the brain data selected by the GA improve the recognition accuracy of the target CNN (TCNN) using pre-trained base CNN (BCNN). The improvement in accuracy is 11% and 4% for the BCI Competition IV and our laboratory datasets, respectively. In addition, the GA-selected training data reduce the CNN training time. The performance of the trained CNN, utilizing transfer learning, is tested for real-time control of a robot manipulator.



Citation: Pongthaisorn, G.; Capi, G. Genetic Algorithm-Based Data Optimization for Efficient Transfer Learning in Convolutional Neural Networks: A Brain–Machine Interface Implementation. *Robotics* **2024**, *13*, 14. <https://doi.org/10.3390/robotics13010014>

Academic Editors: Kensuke Harada, Shuxiang Guo, Yunchao Tang, Mingjie Dong and Wei Feng

Received: 5 December 2023

Revised: 3 January 2024

Accepted: 11 January 2024

Published: 15 January 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: brain–machine interface; convolutional neural networks; genetic algorithm; data selection; transfer learning

1. Introduction

In the future, BMI systems will have a wide range of applications, especially for patients who partially or entirely lose their mobility. For example, BMIs are used in the rehabilitation of post-stroke survivors whose brain damage impacts their mobility. In such implementations, the system captures brain signals, recognizes the user’s intention, and directly gives the command to external devices, e.g., exoskeleton, robotic hand, or other types of prosthetic devices [1,2]. BMI systems are also applied to patients who have completely lost their mobility by using brain signals during motion imagination (MI tasks) [3,4]. There are also different MI applications, such as controlling the robot manipulator [5] or the lower-limb exoskeleton via brain signals [6]. Recently, there have been attempts to expand the application of BMI systems into more complex environments, such as controlling vehicles or assessing the current mental state of drivers [7].

Among the various recording paradigms for brain signals, electroencephalogram (EEG) is commonly employed. EEG is acquired by placing electrodes in multiple areas of the subject’s scalp [8]. However, EEGs are significantly influenced by various sources of noise, leading to a non-stationary and highly variable nature of signals. This issue is crucial for developing robust BMI systems.

For such implementations of BMI systems, the following issues must be considered:

1. Improving the recognition accuracy of BMI systems to decode brain signals with high accuracy. This will improve the implementation of BMI systems, especially in applications such as prostheses and gadgets.
2. Real-time implementation of BMI systems requires a short processing time of captured brain signals in addition to high accuracy. The processing time is strongly related to the number of data and processing methods.

To develop BMI systems for real-life applications, the accurate mapping of EEG data obtained from subjects into their intended movements is essential. Previous methods have applied k-NN (k-nearest neighbor) [9] and support vector machine (SVM) [10] for EEG classification. In [11,12], k-means and k-median core-set data selection methods are utilized, in which a representative subset of points is used to speed up the learning process. In [13], the authors show that many “unforgettable” examples that are rarely incorrectly classified once learned could be omitted without impacting generalization. In [14], the authors proposed a Selection via Proxy (SVP) method to improve the computational efficiency of active learning and core-set selection in deep learning. In this method, cheaper proxy model representation is substituted with an expensive model during data selection. In [15], the authors proposed a data selection strategy to train a neural model to obtain the most significant data information and improve algorithm performance during training, leading to less computations and a reduction in classification error.

More recently, CNNs have gained popularity for EEG classification due to automatic feature extraction capabilities [16–18]. While deep learning offers high accuracy, it comes with the drawbacks of long training time and a substantial amount of required training data. A typical approach for BMI applications is to reduce information redundancy through data optimization techniques such as channels optimization [19]. Nonetheless, this approach does not solve the issue of generalization but rather finds the optimal channels, which can be different among subjects. Transfer learning emerges as a solution to address these challenges [20,21]. In [22], the authors analyze if specific layers are general or specific to one model. In addition, similarity between tasks is quantified. The results show that the transferred weights of frozen and fine-tuned layers perform better than random weights, especially for good generalization.

Different to previous works, we combine GAs and transfer learning to select training data, such as to improve the recognition rate and training time of BMI systems in real-time applications. In this work, a GA is employed to select the optimal data for training the BCNN. This ensures that after transfer learning, the TCNN achieves a superior recognition rate and shorter training time. The evaluation is conducted on two datasets: The first is the BCI Competition IV [23], an online dataset collected from nine subjects performing MI tasks. The second dataset is collected from subjects in our laboratory at Hosei University, involving several grasping motions of the right hand or motor movement (MO) tasks.

We investigate various aspects of transfer learning, including the choice of pre-trained models, the selection of transfer layers, and the fine-tuning process. We explore the trade-offs between model complexity and performance, aiming to strike a balance that facilitates accurate and efficient BMI operation. Our findings contribute to the development of more practical and data-efficient BMI systems, which hold great potential for improving the quality of life of individuals with disabilities and expanding the scope of brain-controlled applications. The proposed algorithm can reveal similarities between the data selected by the GA and the data used to train the TCNN through transfer learning. In addition, we utilize the trained CNNs to map brain signals in real time to the robot motion for human–robot interaction applications. Robot implementation shows good performance, with potential for a wide range of applications.

2. Method

The flowchart of the proposed method is shown in Figure 1. First, a single subject’s EEG data, called the target data, are separated from the EEG dataset of all subjects. The separated EEG data are used to train the TCNN. In our model, the GA selects the subjects

whose EEG data will be used to train the BCNN. Initially, an initial population of individuals is generated, each characterized by a set of genes serving as the binary data of each subject to be included or not in the training of the BCNN (Figure 1). Using the selected EEG data by the GA, we initially train the BCNN. Then, we implement transfer learning utilizing the pre-trained BCNN as the base model. The criterion for the selection of the dataset by the GA is to maximize the average classification accuracy of the TCNN. Based on this evaluation criteria, the best individuals are selected, and crossover and mutation operations are performed to generate the next generation. This iterative process is repeated for a predetermined number of generations, and the individual with the highest fitness is generated.

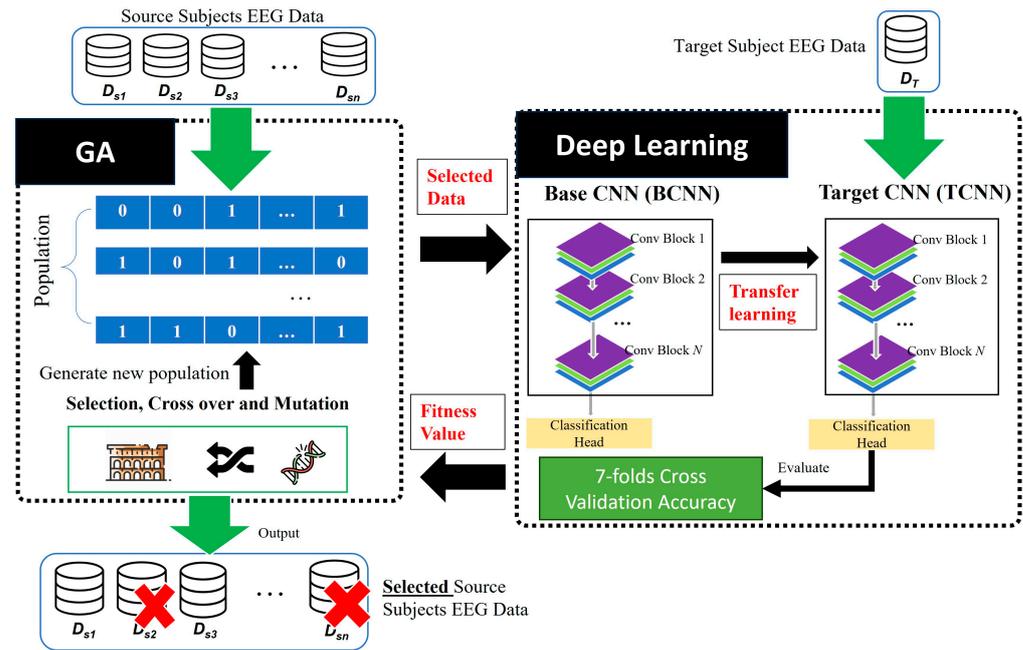


Figure 1. Flowchart of the proposed method.

2.1. Convolutional Neural Network (CNN)

DNNs are a category of Neural Networks that artificially replicate the human brain. DNNs, unlike CNNs with only a few intermediate layers, consist of many intermediate layers, enabling them to handle more complex problems.

On the other hand, a CNN is a type of DNN that incorporates repeatedly both convolutional and pooling layers to extract features from the training data. In this work, EEGNet [9] is utilized for both the BCNN and TCNN, as shown in Figure 2. The network comprises two primary convolution blocks for spatial–temporal fusion and a simple dense layer for classification. The first block includes two convolutional layers: a normal convolution layer for temporal feature extraction and a depth-wise convolutional layer for frequency-specific spatial information. The subsequent block employs a separable convolutional layer to fuse both spatial and temporal information. Additionally, this block utilizes the ReLU activation function, average pooling, and a dropout layer.

2.2. Transfer Learning

In our method, transfer learning is implemented by transferring parts of the pre-trained BCNN model to train the TCNN. To improve accuracy, CNNs typically begin with randomly initialized parameters in the initial learning phase. However, due to the random selection of initial connection weights, a long training process is required. Therefore, transferring a set of previously trained connection weights can be advantageous for faster training with less data of the TCNN, as shown in Figure 3.



Figure 2. EEGNet architecture.

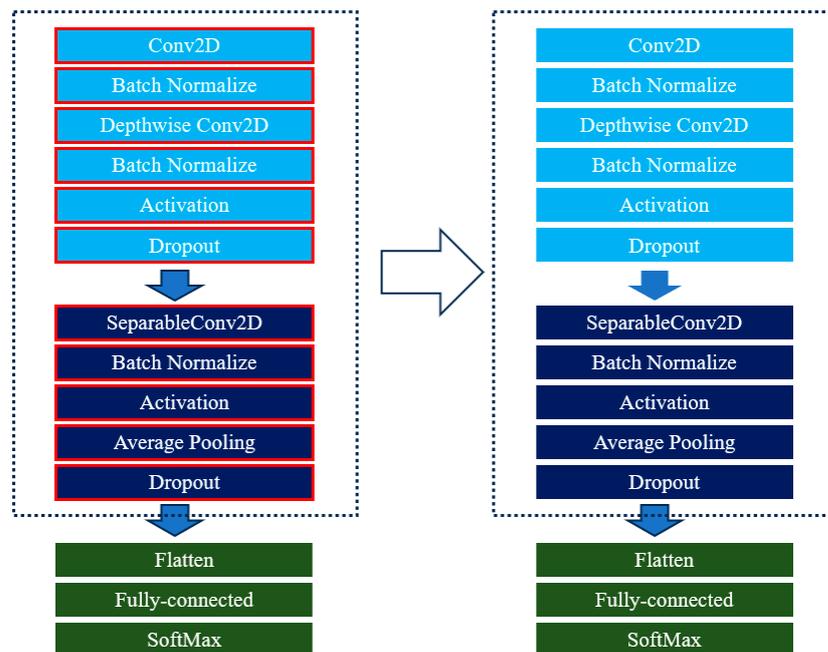


Figure 3. Transfer of convolutional layers from BCNN to TCNN.

There are two approaches to transfer learning: (1) fine-tuning, which involves updating the model for the new dataset, and (2) feature extraction, which involves preventing the update of the model during training, using only the model’s classification head. In our implementation, we use fine-tuning.

Given the need for the model to learn complex data and recognize a wide range of patterns, the BCNN model is initially trained with the EEG data of subjects selected by the GA. Next, the convolutional layers’ parameters of the BCNN model are transferred to the TCNN model, which is then fine-tuned with the target subject dataset. This approach results in rapid parameter adjustment and often achieves a better recognition rate compared to training the model from scratch.

3. Datasets

Due to the diverse tasks in BMI applications, to evaluate the performance of the proposed method, two different datasets—MCRP and MI tasks—are utilized. In MCRP tasks, EEG signals correspond to the movement of subjects' limbs. Conversely, in MI tasks, EEG signals are recorded during the imaginary movements of limbs. This section provides a detailed explanation of how the datasets are created to train and test the performance of the proposed method.

3.1. BCI Competition IV2a

The dataset, publicly available online, was collected by Graz University of Technology. The data are recorded from nine subjects who engaged in four motor imaginary tasks: right hand, left hand, tongue, and both feet. Data from each subject were gathered over 6 sessions, resulting in a total of 48 sessions (12 sessions for each motor imaginary class). Throughout the experiments, subjects were seated in front of a computer screen.

Each trial began with a cross appearing on the black screen at $t = 0$ s, followed by a warning sound, as depicted in Figure 4. After 2 s ($t = 2$ s), a cue in the form of an arrow pointing left, right, down, or up was shown on the monitor corresponding to one of the four motor classes. The arrow remained on the screen for 1.25 s. This instructed the subjects to perform the target motor imagery task until the cross vanished at $t = 6$ s. EEG recording utilized twenty-two Ag/AgCl electrodes spaced 3.5 cm apart, sampled at 250 Hz, and band-pass filtered between 0.5 Hz and 100 Hz.

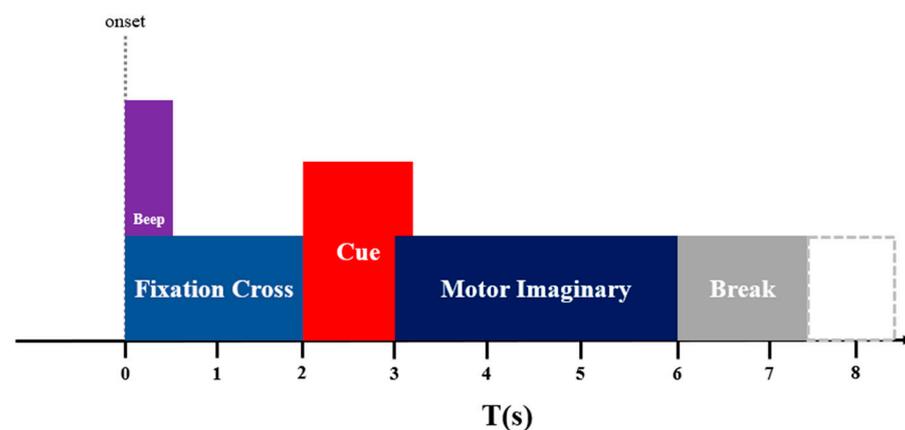


Figure 4. BCI Competition IV2a recording timeline.

3.2. CapiLab MCRP Dataset

In our laboratory, data are collected following the Ethics Policy on Research Involving Humans of Hosei University based on the Declaration of Helsinki.

We gathered EEG data from four subjects aged 20–22. They were seated in front of a laptop, as shown in Figure 5a. The MCRP tasks were grasping a ball, smartphone, and a pen. In addition, a no-motion task was added in which the subjects just rest without grasping any object. The data acquisition method is illustrated in Figure 5b. Initially, animated objects depicting a ball, a smartphone, and a pen were displayed on the upper part of the monitor to remind the subjects to pay attention to the upcoming task. On the lower part of the monitor was the animated hand. As the animated object moved down toward the animated hand, simultaneously, a real object matching the animated one reached the subject's right palm. Once the animated object stopped moving, the grasping motion of the fingers started on the monitor, which lasted 2 s. Simultaneously, the subjects grasped the real objects.

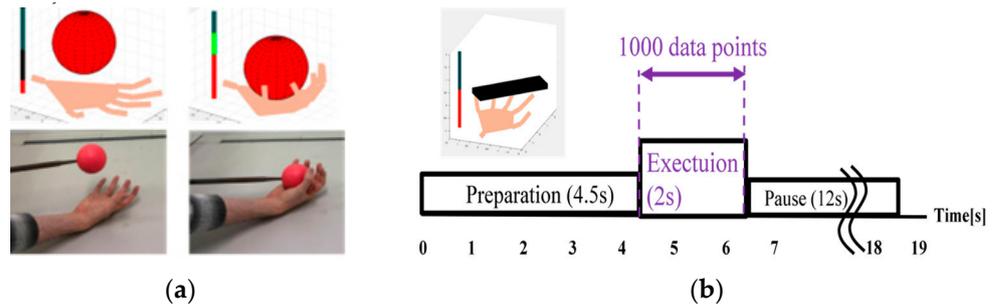


Figure 5. CapiLab MCRP dataset recording: (a) animated object displayed on the screen and (b) timeline of recording.

The electrode arrangement was according to the international 10–20 method. We employ a Mitsar-EEG-201 system to record the user’s brain activity with 19 recording channels. The device’s recording frequency was 500 Hz. Because each hand motion was recorded for two seconds, each trial generated a 1000×19 data point matrix. A low pass filter of 0.16 Hz and a high pass filter of 40 Hz were applied. To reduce electric line noise, a notch filter was installed at 50 ± 5 Hz.

4. Results and Discussion

In our implementation, we divided the collected data into training data (90% of the total data), and the remaining 10% of the data were used for validation. The GA and CNN parameters are shown in Tables 1 and 2, respectively. The process began by selecting one subject’s data for the TCNN. The GA then determined the dataset from the remaining subjects to train the BCNN. In our implementation, the average recognition accuracy of the TCNN after transfer learning is the fitness function of the GA. If the GA selected zero subjects, training was skipped, and the accuracy was set to -0.5 .

Table 1. GA parameters.

Parameter Name	BCI Competition IV2a	CapiLab MRCP
Crossover rate	0.3	0.3
Crossover method	2-points crossover	2-points crossover
Mutation rate	0.1	0.1
Selection algorithm	Tournament	Tournament
Generations	30	20
Individual	50	20

Table 2. CNN parameters.

Parameter Name	BCI Competition IV2a	CapiLab MRCP
Training size	90% of data	90% of data
Testing size	10% of data	10% of data
Optimizer	Adam	Adam
Learning rate	1×10^{-3}	1×10^{-3}
Batch size	64	64
Epochs	30	30

We evaluated similarity using correlation coefficients and Euclidean distances. Supposing that x and y are two different signals, cross-correlation is calculated based on the following equation:

$$Z_k = \sum_{i=0}^{||x||-1} X_i Y^*_{\{i-k+N1\}} \tag{1}$$

where * denotes a complex conjugate operation. Another similarity evaluation is Euclidian distance, which is calculated as follows:

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

4.1. Results ofr BCI Competition IV2a Dataset

Because the data of Subject 4 were damaged, we removed these data from the dataset. So, the data of eight subjects were used to evaluate the method for MI tasks. Subject 8 was selected as the target subject. Therefore, the data of Subject 8 were selected to train the TCNN using transfer learning. The GA selected the best data from the remaining seven subjects to train the BCNN.

The GA population size was 50 individuals, evolving through 30 generations. Figure 6 shows the average fitness value (recognition accuracy after transfer learning) and the standard deviation for each generation through the course of evolution. The GA converged after 20 generations with a consistent increase in the objective function for each generation.

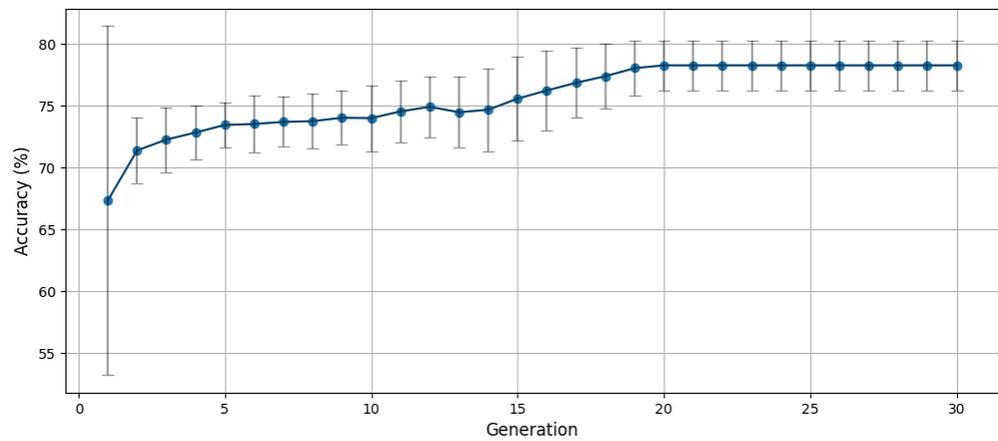


Figure 6. Objective function for each generation using BCI Competition IV2a dataset.

Table 3 shows the values of correlation coefficient and Euclidian distance to measure the similarity between the data of the subjects selected by the GA (Subjects 2, 3, and 7) and the target subject (Subject 8).

Table 3. Similarity ranking result of BCI Competition IV2a via correlation coefficient and Euclidian distance of target subject compared with each subject.

Subject	Similarity to Subject 8	
	Correlation Coefficient	Euclidian Distance
Subject 1	0.50023	4.94123
Subject 2	0.49920	4.83422
Subject 3	0.50332	4.76755
Subject 4	0.50444	5.84189
Subject 5	0.50214	6.80476
Subject 6	0.50301	4.76024
Subject 7	0.50149	4.91620

4.2. Results of CapiLab Dataset

In these experiments, Subject 4 is selected as the target subject. The GA selects the best subjects to train the BCNN from the remaining three subjects. There are 600 data for

each subject (150 for each class, 19 channels \times 1000 data points), resulting in a total of 2400 data/subjects. We reduced the GA population size to 20 individuals, and the termination criteria to 20 generations. The reason for having a smaller number of generations and individuals is the small number of subjects the GA searches through.

Figure 7 shows the average fitness for each generation through the course of evolution. The objective function increased fast during the first generations and converged by the tenth generation. This is because the search space of the GA is small (three subjects).

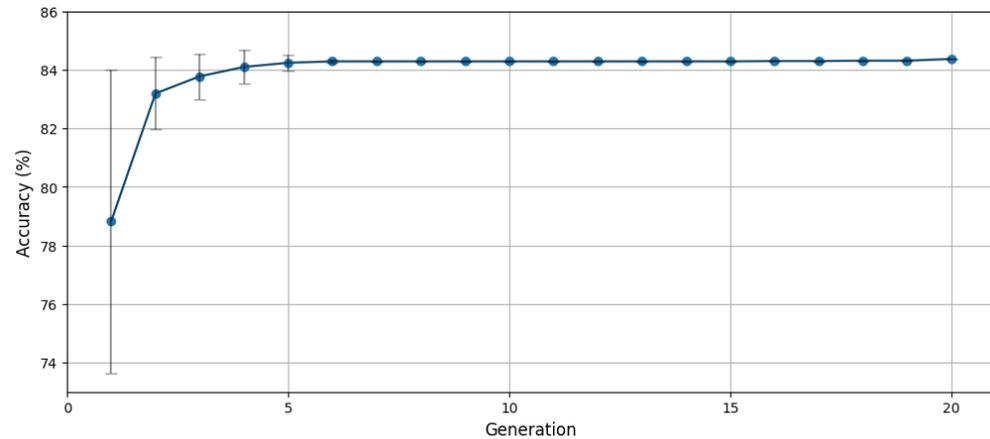


Figure 7. Objective function for each generation using CapiLab dataset.

Table 4 presents the similarity between the subjects selected by the GA and those trained by the TCNN. Table 4 demonstrates that the subjects chosen by the GA exhibited the highest similarity to Subject 4, both in terms of the correlation coefficient and Euclidean distance.

Table 4. Similarity ranking result for CapiLab dataset.

Subject	Similarity to Subject 4	
	Correlation Coefficient	Euclidian Distance
Subject 1	0.5000511	409.5032
Subject 2	0.5020856	500.9085
Subject 3	0.5033397	502.7810

However, there is an issue requiring further investigation. As shown in Tables 3 and 4, only two out of seven results agreed on the similarity criteria for the BCI competition dataset, and one out of three agreed for the CapiLab dataset. We attribute this to the convolutional operation of EEGNet, which employs depth-wise convolution for spatial-temporal information fusion, combining temporal and spatial similarities. Another issue is the filtering technique that is applied during data preprocessing. Some of the spatial information was lost due to the applied filters.

4.3. Comparison of CNNs Recognition Rates Trained with All and GA-Selected Subject Data

Table 5 shows the recognition accuracy in the case of (1) all the subjects’ data being used to train the BCNN; (2) data of the subjects selected based on the highest correlation coefficients (Subjects 3, 4, and 6 for BCI Competition IV and Subject 3 for CapiLab); (3) data of the subjects selected based on the lowest Euclidian distances (Subjects 2, 3, and 6 for BCI Competition IV and Subject 1 for CapiLab); and when only the data of the subjects selected by the GA were used. The results show that employing the GA to optimize the training data resulted in an improvement in recognition accuracy of approximately 11% and about 4% for the BCI Competition IV and Capilab dataset, respectively. The reduction in the size

of the training dataset significantly reduces training time, demonstrating the feasibility of the proposed method in selecting the best subjects' data.

Table 5. Improvement comparison between TCNN trained with all data, CC-based, ED-based similarity, and proposed system with GA.

Datasets	Accuracy (%)			
	All Data	Correlation Coefficient	Euclidian Distance	GA
BCI Competition IV2a	71.43 ± 5.11	66.01 ± 6.75	77.92 ± 5.95	78.81 ± 5.92
CapiLab MCRP	81.1 ± 7.92	74.54 ± 6.71	77.54 ± 6.71	84.4 ± 5.81

4.4. Real-Time Robot Control Using Brain Signals

In the case of the Capi-lab dataset, we verified the performance of the TCNN in real-time control of the robotic arm. The experimental setup is shown in Figure 8. The trained TCNN (Subject 4) recognizes the object grasped by the subject using brain signals, and the robot manipulator moves to the target object placed on the table. The subject was wearing a gel-based EEG cap, which was connected to the EEG amplifier. In our implementation, we utilized a KINOVA Gen 3 lightweight robot manipulator. The response time of the robot is 2.075 s, which is 2 s for brain data collection and 0.075 s for the TCNN to recognize the grasping object. For the experiment, we limited the speed of robot motion for safety reasons. So, the robot reached the target object in around 2.4 s. Figure 9 shows the video capture of three different trials. The recognition rate decreases to nearly 60% due to slight changes in electrode positions. Therefore, collecting more training data is necessary to improve the robustness of the developed BMI system. Such implementation increases the applicability of BMI systems for human–robot interaction (HRI) tasks.

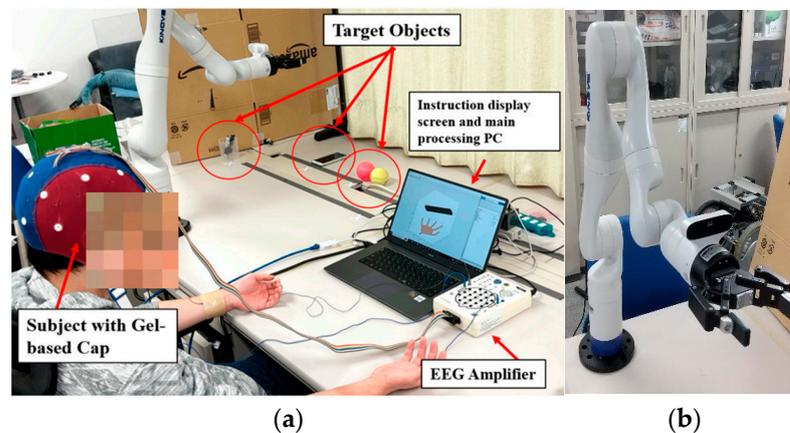


Figure 8. HRI experiment setup: (a) subject with EEG cap for recording EEG signals and (b) robot manipulator.

4.5. Discussion

The proposed method outperformed other data selection methods and makes accurate recognitions with a reduced amount of data. The experimental results demonstrated significant improvements in the recognition accuracy of the TCNN when the BCNN was trained with the GA-selected data. Moreover, beyond accuracy gains, the GA-selected training data contributed to a notable reduction in CNN training time. This efficiency improvement is particularly crucial for real-time robotic applications where response time is essential. The findings from this work not only underscore the effectiveness of the proposed approach but also highlight its potential for practical implementations. By reducing the number of training data, the proposed method minimizes the computational burden of CNNs. This work can be further extended for the exploration and refinement of

hybrid techniques in BMI systems, contributing to advancements in neurotechnology and human–machine interfaces.

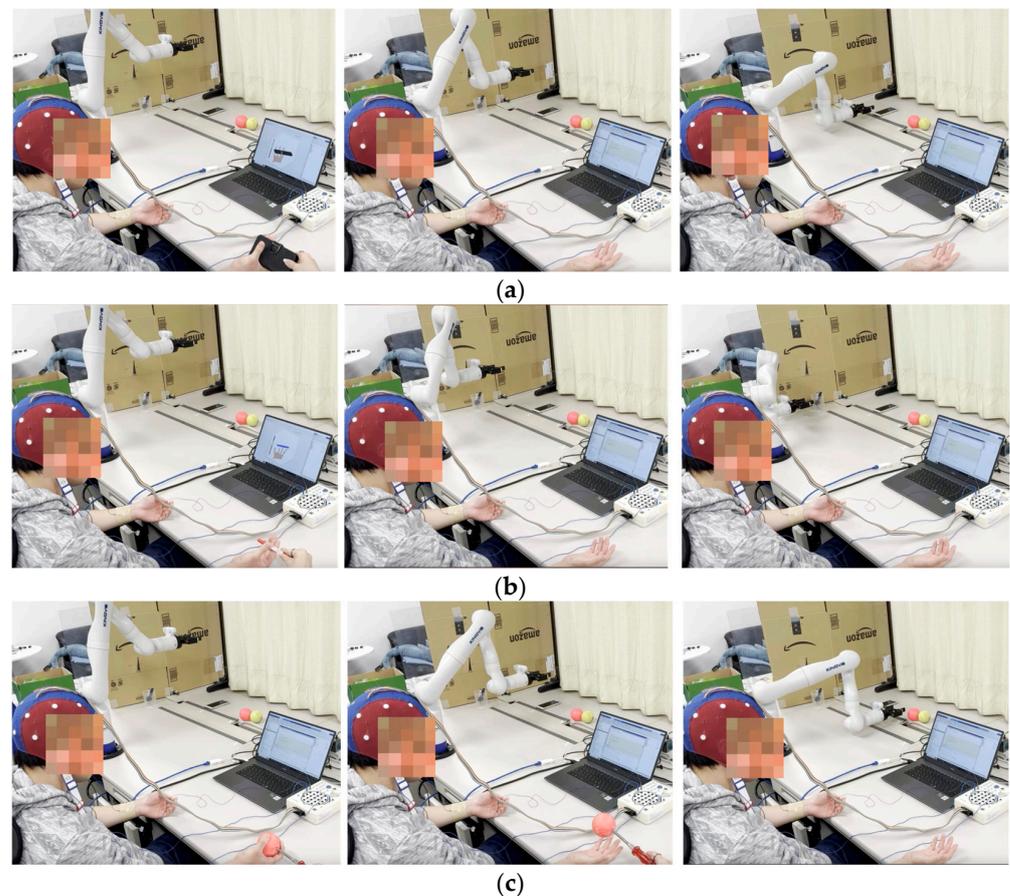


Figure 9. Video capture of robot motion using brain signals. (a) smartphone; (b) pen; (c) ball.

5. Conclusions

This paper presented a new approach to improve the performance of CNNs by addressing two critical challenges: the quality of training data and the training time of CNNs. By synergistically integrating transfer learning and the GA, our methodology optimizes the training data for CNNs, demonstrating notable improvements in recognition accuracy and reduced training time. The utilization of transfer learning across different subjects proves to be a key factor in our approach, generating a more generalized and robust model. The innovative use of the GA for data selection improved CNN performance. In addition, the results provide valuable insights into the similarities in brain activity across subjects. This dual optimization strategy refined the neural network’s ability to recognize patterns and contributed to a deeper understanding of brain functions. The empirical results underscore the effectiveness of our proposed approach. Additionally, the reduction in training time is significant for practical implementations of BMI systems. The trained CNNs also showed good performance and effective real-time control of robot manipulators using brain signals for human–robot interaction applications.

Author Contributions: G.P. is responsible for the brain activity data collection, implementation, and evaluation of the CNNs, genetic algorithms, and transfer learning. In addition, he implemented CNNs for real-time robot control. G.C. contributed to initiating the research topic and evaluated the CNNs, GA, and robot implementation results. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Grant-in-Aid for Scientific Research (KAKENHI) Grant Number 21K03970.

Data Availability Statement: The BCI competition IV2a dataset is open data and can be obtained from <https://www.bbci.de/competition/iv/> (2 February 2023). The CapiLab MRCP dataset may not be publicly uploaded due to privacy and agreement reasons. However, for scientific research requests, please contact the authors.

Acknowledgments: The authors would like to acknowledge the financial support from the Japanese Government, Grant-in-Aid for Scientific Research (KAKENHI) Grant Number 21K03970, as well as the three reviewers whose insightful comments helped improve the paper. The authors would like to thank the students of the BMI research group for their support, especially Shuichi Suguro.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Baniqued, P.D.E.; Stanyer, E.C.; Awais, M.; Alazmani, A.; Jackson, A.E.; Mon-Williams, M.A.; Mushtaq, F.; Holt, R.J. Brain-computer interface robotics for hand rehabilitation after stroke: A systematic review. *J. Neuroeng. Rehabil.* **2021**, *18*, 15. [[CrossRef](#)] [[PubMed](#)]
2. Lo, K.; Stephenson, M.; Lockwood, C. Effectiveness of robotic assisted rehabilitation for mobility and functional ability in adult stroke patients: A systematic review. *JBI Evid. Synth.* **2017**, *15*, 3049–3091.
3. Ferrero, L.; Quiles, V.; Ortiz, M.; Iáñez, E.; Azorín, J.M. A BMI based on motor imagery and attention for commanding a lower-limb robotic exoskeleton: A case study. *Appl. Sci.* **2021**, *11*, 4106. [[CrossRef](#)]
4. Chu, Y.; Zhao, X.; Zou, Y.; Xu, W.; Song, G.; Han, J.; Zhao, Y. Decoding multiclass motor imagery EEG from the same upper limb by combining Riemannian geometry features and partial least squares regression. *J. Neural Eng.* **2020**, *17*, 046029. [[CrossRef](#)] [[PubMed](#)]
5. Penaloza, C.I.; Nishio, S. BMI control of a third arm for multitasking. *Sci. Robot.* **2018**, *3*, eaat1228. [[CrossRef](#)] [[PubMed](#)]
6. Ferrero, L.; Quiles, V.; Ortiz, M.; Iáñez, E.; Gil-Agudo, Á.; Azorín, J.M. Brain-computer interface enhanced by virtual reality training for controlling a lower limb exoskeleton. *iScience* **2023**, *26*, 106675. [[CrossRef](#)] [[PubMed](#)]
7. Hekmatmanesh, A.; Nardelli, P.H.; Handroos, H. Review of the state-of-the-art of brain-controlled vehicles. *IEEE Access* **2021**, *9*, 110173–110193. [[CrossRef](#)]
8. Garg, P.; Davenport, E.; Murugesan, G.; Wagner, B.; Whitlow, C.; Maldjian, J.; Montillo, A. Automatic 1D convolutional neural network-based detection of artifacts in MEG acquired without electrooculography or electrocardiography. In Proceedings of the 2017 International Workshop on Pattern Recognition in Neuroimaging (PRNI), Toronto, ON, Canada, 21–23 June 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 1–4.
9. Bhattacharyya, S.; Khasnobish, A.; Chatterjee, S.; Konar, A.; Tibarewala, D.N. Performance analysis of LDA, QDA and KNN algorithms in left-right limb movement classification from EEG data. In Proceedings of the 2010 International Conference on Systems in Medicine and Biology, Kharagpur, India, 16–18 December 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 126–131.
10. Ines, H.; Slim, Y.; Noureddine, E. EEG classification using support vector machine. In Proceedings of the 10th International Multi-Conferences on Systems, Signals & Devices 2013 (SSD13), Hammamet, Tunisia, 18–21 March 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 1–4.
11. Har-Peled, S.; Kushal, A. Smaller coresets for k-median and k-means clustering. *Discret. Comput. Geom.* **2007**, *37*, 3–19. [[CrossRef](#)]
12. Ivor, W.T.; James, T.K.; Pak-Ming, C. Core vector machines: Fast SVM training on very large data sets. *J. Mach. Learn. Res.* **2005**, *6*, 363–392.
13. Toneva, M.; Sordani, A.; Combes, R.T.D.; Trischler, A.; Bengio, Y.; Gordon, G.J. An empirical study of example forgetting during deep neural network learning. *arXiv* **2018**, arXiv:1812.05159.
14. Coleman, C.; Yeh, C.; Musmann, S.; Mirzasoleiman, B.; Bailis, P.; Liang, P.; Leskovec, J.; Zaharia, M. Selection via proxy: Efficient data selection for deep learning. *arXiv* **2019**, arXiv:1906.11829.
15. Ferreira, J.O.; Mendonça, M.O.; Diniz, P.S. Data selection in neural networks. *IEEE Open J. Signal Process.* **2021**, *2*, 522–534. [[CrossRef](#)]
16. Lawhern, V.J.; Solon, A.J.; Waytowich, N.R.; Gordon, S.M.; Hung, C.P.; Lance, B.J. EEGNet: A compact convolutional neural network for EEG-based brain-computer interfaces. *J. Neural Eng.* **2018**, *15*, 056013. [[CrossRef](#)] [[PubMed](#)]
17. Mattioli, F.; Porcaro, C.; Baldassarre, G. A 1D CNN for high accuracy classification and transfer learning in motor imagery EEG-based brain-computer interface. *J. Neural Eng.* **2022**, *18*, 066053. [[CrossRef](#)] [[PubMed](#)]
18. Raghu, S.; Sriraam, N.; Temel, Y.; Rao, S.V.; Kubben, P.L. EEG based multi-class seizure type classification using convolutional neural network and transfer learning. *Neural Netw.* **2020**, *124*, 202–212. [[CrossRef](#)] [[PubMed](#)]
19. Sugiyama, S.; Pongthansorn, G.; Aya, S.; Capi, G. EEG Channel Optimization for Wireless BMI-based Robot Interaction for Internet of Robotic Things. In Proceedings of the 2023 6th Conference on Cloud and Internet of Things (CIoT), Lisbon, Portugal, 20–22 March 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 131–135.
20. Ju, C.; Gao, D.; Mane, R.; Tan, B.; Liu, Y.; Guan, C. Federated transfer learning for EEG signal classification. In Proceedings of the 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Montreal, QC, Canada, 20–24 July 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 3040–3045.

21. Pogthaisorn, G.; Takahashi, R.; Capi, G. Learning Time and Recognition Rate Improvement of CNNs through Transfer Learning for BMI Systems. In Proceedings of the Conference on Biomimetic and Biohybrid Systems, Genoa, Italy, 10–13 July 2023; Springer Nature Switzerland: Cham, Switzerland, 2023; pp. 63–76.
22. Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How transferable are features in deep neural networks? In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), Montreal, QC, Canada, 8–11 December 2014; pp. 3320–3328.
23. Brunner, C.; Leeb, R.; Müller-Putz, G.; Schlögl, A.; Pfurtscheller, G. *BCI Competition 2008–Graz Data Set A*; Institute for Knowledge Discovery (Laboratory of Brain-Computer Interfaces), Graz University of Technology: Styria, Austria, 2008; Volume 16, pp. 1–6.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.