Brain-guided manifold transferring to improve the performance of spiking neural networks in image classification

Zahra Imani, Mehdi Ezoji, Timothée Masquelier

Brain-guided Manifold Transferring to Improve the Performance of Spiking Neural Networks in Image Classification

Zahra Imani[1], Mehdi Ezoji[1]* and Timothée Masquelier[2]

[1]Faculty of Electrical and Computer Engineering, Babol Noshirvani University of Technology, Babol, Iran

[2] CERCO UMR 5549, CNRS – Universit´e Toulouse 3, Toulouse, France

* m.ezoji@nit.ac.ir

Abstract: Spiking neural networks (SNNs), as the third generation of neural networks, are based on biological models of human brain neurons. In this work, a shallow SNN plays the role of an explicit image decoder in the image classification. An LSTM-based EEG encoder is used to construct the EEG-based feature space, which is a discriminative space in viewpoint of classification accuracy by SVM. Then, the visual feature vectors extracted from SNN is mapped to the EEG-based discriminative features space by manifold transferring based on mutual k-Nearest Neighbors (Mk-NN MT). This proposed "Brain-guided system" improves the separability of the SNN-based visual feature space. In the test phase, the spike patterns extracted by SNN from the input image is mapped to LSTM-based EEG feature space, and then classified without need for the EEG signals. The SNN-based image encoder is trained by the conversion method and the results are evaluated and compared with other training methods on the challenging small ImageNet-EEG dataset. Experimental results show that the proposed transferring the manifold of the SNN-based feature space to LSTM-based EEG feature space leads to 14.25% improvement at most in the accuracy of image classification. Thus, embedding SNN in the brain-guided system which is trained on a small set, improves its performance in image classification.

Keywords: Spiking Neural Networks, Manifold Transferring, Brain-guided system, Image Classification, LSTM, and EEG.

1. INTRODUCTION

Artificial Neural Networks (ANNs), especially deep neural networks, have shown good performance in machine learning such as pattern recognition. Although these methods are highly effective, they function differently from the human brain. In recent years, models inspired by the human brain function to mimic its ability in cognitive tasks have attracted the research community's attention [1]. Spiking Neural Networks (SNNs) are the third generation of neural networks with biologically plausibility [1]. Spiking neurons (the basic unit of SNN) simulate the process in which nerve cells take stimulation, generate action potentials and emit spikes, such that information transmission in neurons is done by propagating discrete spikes [2].

SNNs are challenging to train because spiking neurons have discrete, nonlinear, and non-differentiable transfer functions [2]. Spike-timing dependent Plasticity (STDP) is a physiological unsupervised learning mechanism used for training the convolutional and fully connected (FC) SNNs [3]. In STDP, synaptic weight change is based on the temporal correlation between pre- and post-synaptic spike time [4][1]. ANN-to-SNN conversion is the second method used to training SNNs. In this method, first, an ANN is trained and, then the weights are transferred to the equivalent/corresponding SNN [2][5][6][7][8]. Spike-based backpropagation algorithms with the integration of spike timing information are the third method used for SNN training [1][9][10][11]. The non-differentiability of the spiking neuron is handled in spike-based backpropagation methods by defining a surrogate gradient as a continuous approximation of the real gradient [12][13] [14].

The last method is the hybrid training technique which combines ANN-SNN conversion and spike-based back-propagation [15][16][17]. In this method, ANN-SNN conversion is used as an initialization step followed by spike-based back-propagation incremental training [15].

In this paper, SNN is trained with conversion method and compared with other training methods i.e. training from scratch and also hybrid training. Then, the trained SNN is applied as an image encoder in the image classification problem. To improve the separability of these encoded image manifolds, this encoder is embedded in a brain-guided system. Thus, simplicity and bio-inspiring learning algorithm of SNN is combined with the discriminative potential of the brain-guided system.

Experimental results on a subset of ImageNet dataset [18] show that this proposed method improves the performance of the SNN in image classification problem.

The rest of the paper is as follows: In the “related works” section, we present a literature review about the existing brain guided models. In the “proposed method” section, the details of SNN are explained in model architecture and learning rule and use SNN in our brain guided system. In the “experiments” section, we describe our experiments and the obtained results. Finally, remarks and future work is provided in Conclusion.

2. RELATED WORKS

EEG is a non-invasive method that measures the electrical activity of the brain through electrodes that are placed on the scalp. These signals are widely used in research fields such as clinical diagnosis of diseases as Alzheimer's [19][20], Multiple sclerosis (MS) [21], Parkinson's [22] and, brain-computer interface (BCI) applications [23][24][25].

Researchers in cognitive neuroscience figure out that visual stimuli creates discriminating patterns related to different categories in brain activities [26][27]. The human brain can recognize common patterns in stimuli quickly. Researches in this area will be grouped in two categories:

EEG based visual stimulus Classification. In [28], linear discriminant analysis (LDA) classifier was used to classify EEG signals evoked by visual stimuli. In this research, several experiments have been performed on the different numbers classes and time intervals of the EEG signals. In [29], Fares et al. proposed an EEG-based bi-directional deep learning framework for visual object classification. This framework consists of three steps: extracting information from the EEG signal, feature encoding, and classification. The first step is to extract region-level information from the raw signals. All EEG signal channels are divided into the right hemisphere, left hemisphere and, middle group based on the physical position of the channel electrode. By calculating each channel's difference in the left hemisphere group and its corresponding channel in the right hemisphere and combining the obtained difference with the middle group, region-level information is extracted. The feature encoding step is performed to extract the EEG description from the region-level information through a stacked bi-directional long short-term memory (LSTM) network. After independent component analysis (ICA) based feature selection, SoftMax and support vector machine (SVM) are examined as the classifier. The classification accuracy on the ImageNet-EEG dataset [30] with SVM at the best performance was 97.3%. In [31], the RA-BiLSTM network has been introduced to classify brain activity evoked by visual stimuli. The RA-BiLSTM architecture consists of a forward A-LSTM layer and a backward A-LSTM layer. The difference between A-LSTM and LSTM is a channel level

attention gate. The accuracy of the proposed method on the ImageNet-EEG dataset is 98.4%.

Human brain-guided system for image classification. Spampinato et al. [30] provide a new perspective on the "human-based computing" strategy for automated visual classifications. The idea of this method is that the model learns a discriminative space of brain signals from visual categories, and then the images are mapped to this manifold. LSTM is used as the EEG encoder in this method. Fine-tuned convolutional networks, such as AlexNet, GoogleNet, and VGG, are used as a feature extractor to encode the image. The feature vectors extracted from the images are mapped to the corresponding feature vectors extracted from the EEG signals by several regression methods, such as k-Nearest Neighbors (KNN), ridge, and Random Forest (RF). The output of this step is fed to the classifier and the image class label is assigned. The proposed method on the ImageNet-EEG database[30] achieves an accuracy of 83% for image classification. In [27], Kavasidis et al. proposed an approach to generate images using EEG visual evoked brain signals. The proposed method involves an encoder identifying a hidden feature space for classifying the brain signal and a decoder that converts the learned features into images using a deconvolution method. The encoder consists of an LSTM layer. Variational autoencoder (VAE) and generative adversarial network (GAN) are used for decoding to generate corresponding images. The ImageNet-EEG dataset was used to evaluate the proposed image generation method. A qualitative examination of the produced images shows that VAE performs better in displaying the object's structure and leads to more realistic but low resolution images. On the other hand, GAN has high-resolution but artificial images. The results confirm that both models can translate EEG features into meaningful images in corresponding classes.

Palazzo et al. [32] proposed a multi-model approach that uses deep encoders for images and EEG. The method presented in a Siamese configuration for joint manifold learning, and maximizing the degree of compatibility between visual features and brain representation. In this method, the EEG encoder consists of a convolutional network called EEG-ChannelNet. For image encoding, a pre-trained convolutional neural network (CNN) is used to extract visual features and apply them to a linear layer for mapping in shared embedding space. The ImageNet-EEG dataset is used to train the network and evaluate the proposed method. In the best case, using Inception-v3 in the image encoder, the EEG signals classification rate is 60.4% and for images is 94.4%.

Researchers in [33] have proposed integrating explicit and implicit learning into a deep learning framework to analyze the content of EEG signals and stimulus images. The image is processed via a CNN-based encoder in the explicit learning path as well as a stimulus to stimulate the brain. In the implicit learning path, raw EEG signals are processed via LSTM. Then, K nearest images to the presented image is retrieved in the KNN regressor layer. Then, the mean of the EEG features corresponding to these K images are considered as the EEG-based description of the image. In this method, SoftMax and multi-class SVM are used as classifiers. The proposed framework using the ImageNet-EEG dataset[30] has achieved an accuracy of 94.1% at best.

In [26], a method for automatic visual classification that uses an ensemble LSTM-based deep network to encode EEG is introduced. This study proposed the LSTMS-B model to decode the activities of the human brain. This model is based on the standard Bagging algorithm, which suggests a voting strategy for decoding and classifying EEG signals. Separately, in the image encoding phase, the ResNet-18 network and KNN, RF, and support vector regression (SVR) are used as

regressors to map the image features to the EEG features. The classification accuracy of the proposed method on the ImageNet-EEG dataset is reported to be 90.16% at best.

A two-step method for combining Gabor features from EEG data and features extracted from images is proposed in [34]. In this method, the feature vectors extracted by Inception-v3 are combined with the EEG feature vectors, and the classification is done by the new feature. The results showed that combining EEG and image features improves the classification accuracy on an EEG-image dataset (with 14 electrodes and images of six different classes) from 91% to 97%.

3. PROPOSED METHOD

There is a significant relation between the brain activation map and what evokes the human visual system (HVS) [32][35]. In other words, it seems that visual stimulus is encoded into brain activation map by the HVS, which can be sensed by EEG, fMRI, etc. Due to the high ability of the human in object recognition, it can be assumed that the brain's encoding system provides high separability between different visual classes. One of the easiest ways to extract the distinguishing information of brain function is EEG. Therefore, we have taken such useful information as a guide to distinguish between different classes of SNN-encoded images by inspiration from the brain function.

In this work, SNN is chosen because of its higher compatibility with HVS [36] and its low achieved performance in classification problems [37]. It is expected that synergistic interactions between the brain-inspired coding system and the artificial networks can improve SNN performance and also HVS modeling by embedding the SNN in a brain-guided system.

As it can be seen from literature review, a human brain-guided system consists of three fundamental parts: (i) an implicit learning part or EEG-based encoder, (ii) an explicit learning part or image encoder and (iii) a manifold mapping from image-based feature space into EEG-based encoding space.

As shown in Figure 1, the proposed method for image classification consists of two stages: training and test stages. In the training stage, a LSTM-based network encodes the EEG signals corresponding to visual stimuli, and a classifier is employed to classify these extracted feature vectors. The SNN-based encoder is trained using these visual stimuli (images), simultaneously. Then, the manifold transferring block is trained to map the visual feature vector into the corresponding EEG-based feature vector, i.e. a mapping from explicit visual space to implicit EEG-based space.

In the test stage, the SNN-based visual feature vector fed to manifold transferring block to find its corresponding implicit feature vector in EEG-based space. Finally, the classifier assigns a label to this vector. Therefore, there is no need for EEG signals in the test stage.

In the following, the details of each component of the proposed method are discussed.

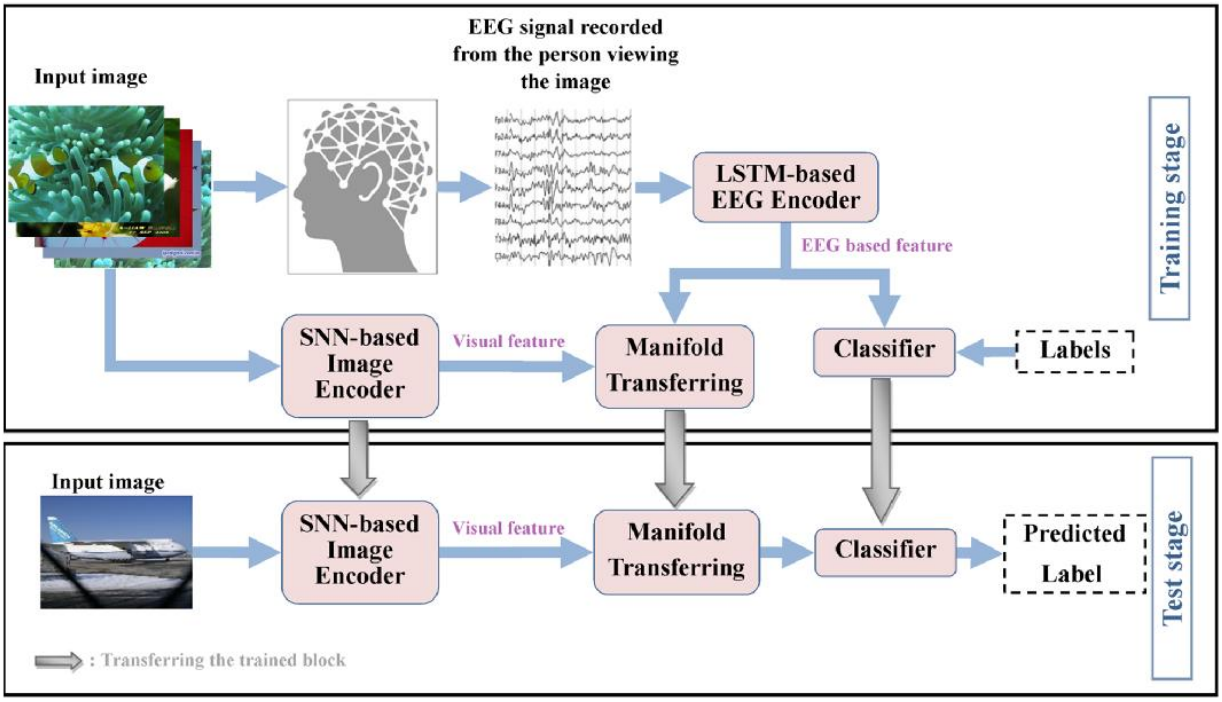


Figure 1. Graphical abstract of the proposed model for image classification

3.1. SNN-based Image Encoding

In this section, the spiking neuron model is described, and the architectural details of the SNN-based image encoder are explained.

3.2. Spiking neuron model

In spite of ANN, a SNN composed of spiking neurons, such as Leaky Integrate-and-Fire (LIF) neurons which are defined as follows[38]:

$$M[t] = U[t-1] + \frac{1}{\tau}(X[t] - (U[t-1] - U_{reset})) \quad (1)$$

$$S[t] = \Theta(M[t] - U_{th}) \quad (2)$$

$$U[t] = U[t](1 - S[t]) + U_{reset}\, S[t] \quad (3)$$

Here M[t] and U[t] denote the membrane potential after neural dynamics and after the trigger of a spike at time step t, respectively. Where τ represents the membrane time constant and X[t] is the input current at time step t. U_{th} and U_{reset} are the firing

threshold and the reset potential, respectively. The output spike at time step t, defined as S[t], which equals 1 if there is a spike and 0 otherwise.

The spike generative process is described with Eq. (2), where $\Theta(x)$ is the heaviside step function and is defined by $\Theta(x) = 1$ for $x \geq 0$ and $\Theta(x) = 0$ for $x < 0$. Eq. (3) illustrates resetting processes.

In surrogate gradient learning method, the linear or exponential approximation of the gradient respectively represented by Eq. (4) and Eq. (5), is used during error back-propagation [15]:

$$\Theta'(x) = \alpha \, max\{0, 1 - |x|\} \tag{4}$$

$$\Theta'(x) = \alpha e^{-\beta|x|} \tag{5}$$

Where α and β are constants, and x can be substituted by $M[t] - U_{th}$.

According to surveys conducted in [15], the term $\Delta t = t - t_s$ $(0 < \Delta t < T)$ is a good replacement for $M[t] - U_{th}$, where t_s is the last time step when the post-neuron generated a spike, and T represents the total number of time steps.

This replacement helps for faster access during training by be pre-computed and stored all possible values of gradients in a table.

Unlike STDP, surrogate gradient-based learning enables efficient end-to-end training of SNNs [14].

3.2.1. SNN-based Image Encoder

In this work, the above-mentioned idea (in Figure 1) is applied to the spiking neural network introduced in [15] which its architecture is tabulated in Table 1.

Table 1. SNN model architecture with VGG5 structure

	Layer	Input size	# kernel	Kernel size	pad	Stride	Output size
Input layer	Poisson Generator	3×64×64	-	-	-	-	3×64×64
Feature Extractor Layer	Conv-1	3×64×64	64	3×3	1	1	64×64×64
	AvgPool-1	64×64×64	-	3×3	0	3	64×21×21
	Conv-2	64×21×21	128	3×3	1	1	128×21×21
	Conv-3	128×21×21	128	3×3	1	1	128×21×21
	AvgPool-2	128×21×21	-	3×3	0	3	128×7×7
Classifier	FC1	128×7×7 = 6272	-	-	-	-	4096
	FC2	4096	-	-	-	-	4096
	FC3	4096	-	-	-	-	2

In the input layer, the images are converted into spike trains. The Poisson generator function generates a Poisson spike train whose pulses are proportional to the pixel's values at the input [15].

More details of this spiking neural network is presented in [15].

3.3. LSTM-based EEG Encoder

The structure of the EEG encoder is shown in Figure 2. This network consists of a layer with n LSTM cells that followed by two fully connected layers. In the LSTM layer, the dimensionality of the output is $m \times n$, which the last state (n in size) is fed to the first fully connected layer. The LSTM-based network can extract information from a long sequence of EEG signals because it can learn long-term dependencies [26]. The embedding layer has n neurons. The number of neurons in the classifier layer, N, is equal to the number of classes. After training this network, the EEG-based feature vector is extracted from the output of embedding layer.

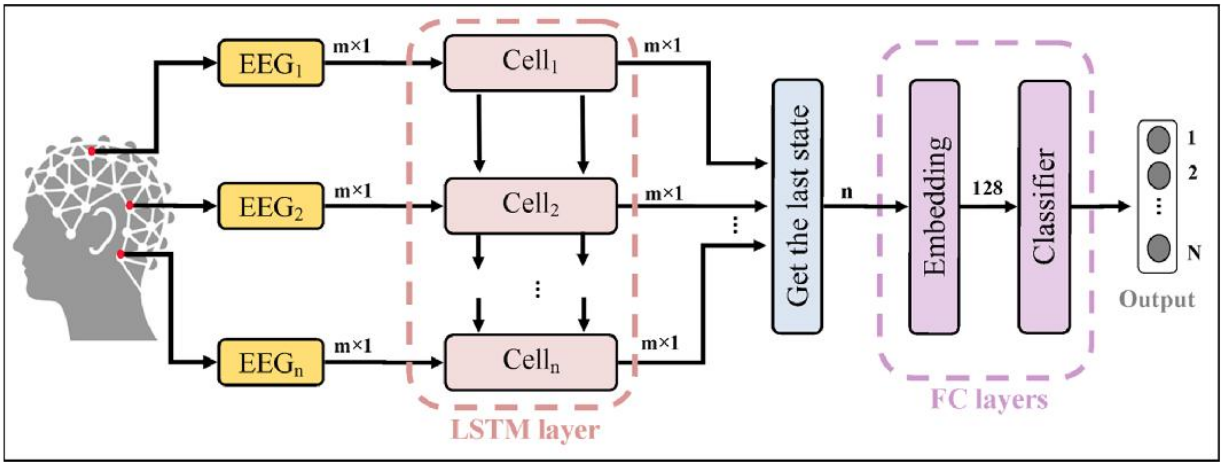


Figure 2. Architecture of the applied LSTM-based EEG encoder, where (n) is the number of EEG signal channels.

3.4. Manifold transferring

It is worth noting that in the training set there is an EEG-based feature vector Y_i corresponding to each image-based feature vector X_i. Obviously, Y_is lie on a manifold embedded in **Y**-space and also for **X**-space. Therefore, the manifold **X** is corresponding to manifold **Y**. Manifold Transferring (MT) refers to a set of methods (e.g. KNN, Random Forest, Ridge regressor, etc) to find the corresponding instance in the manifold **X** for each test observation in manifold **Y**. The core of these methods is to find the best examples in the manifold **Y** which represent the test observation. Then, the vectors of **X** corresponding to these best examples are similarly combined to map the test observation into the manifold **X**. For example, in KNN manifold transferring, *k* nearest neighbors of the test visual feature vector O are considered from the training set. The neighbors are weighted by the similarity to the test observation which is usually the reciprocal of their distances to it which encoded in W^*. The corresponding instances in EEG-based manifold corresponding to these neighbors combined with the mentioned weights to represent the test visual vector O in the EEG-based manifold. In general form,

for the observation O in the feature space **Y**, W^* is the weights of the training examples of **Y** that minimizes fidelity penalty term of F subject to the restriction of R. Included in the cost function J.

$$J(O, w) = \underset{W}{argmin}\, F(Y, O, W) + \alpha R(W) \quad (6)$$

Where α is the regularization parameter.

In this work, the mutual k-Nearest Neighbors manifold transferring (Mk-NN MT) is introduced and compared with different MT methods. In this view, if two samples are in k-neighborhood of each other, they are neighbors [39].

In this approach, the visual feature vector of the test image is transferred to the weighted average of the EEG-based feature vectors corresponding to its k' mutual neighbors in the training visual feature space. If there isn't any mutual neighborhood ($k' = 0$) for a test visual feature, the KNN manifold transferring is executed. The algorithm of Mk-NN MT is presented in Figure 3.

Mutual k-NN Manifold Transferring Algorithm
Input: k (# Neighbors), **Y** (Training Samples of visual feature space), **X** (Training samples of EEG-based feature space) , O (an observation or test sample in visual feature space) **Output**: X_o as the presentation of O in EEG-based feature space
Step 1: Find and select k nearest neighbors of O in **Y** space named Y_1, Y_2, …, Y_k sorted in order of similarity. **Step 2**: Find and select all mutual neighbors of O between Y_i s. $k' \leq k$ denotes the number of these neighbors and update the set of selected Y_is. **Step 3**: Calculate the weight w_i of each Y_i based on predefined similarity measurement. **Step 4**: Find X_i in the EEG-based feature space corresponding to the selected Y_is. **Step 5**: Calculate X_o as the weighted average of X_is

Figure 3. Mutual k-NN Manifold Transferring Algorithm

4. EXPERIMENTAL RESULTS

To evaluate the proposed SNN-based brain-guided system, several experiments are conducted on the ImageNet-EEG dataset [24]. All the experiments are implemented in PyTorch, on the CPU of Intel Core i3-3600k- 3.70 GHz, with NVIDIA GeForce GTX 1060 with 48 GB RAM.

4.1. ImageNet-EEG Dataset

The ImageNet-EEG dataset presented in [30] has been used to train and evaluate the proposed method. This dataset is a subset of ImageNet [18], including 40 classes of images. There are 50 images in each class; each image is displayed to six subjects (five male and one female) on the monitor screen for 500 milliseconds. These six subjects were homogeneous in terms of education level, age, and cultural background [30]. Pause time for displaying between classes was 10 seconds. Each person's EEG signals are recorded/ received by a cap with 128 electrodes (Brain products actiCAP128Ch). During the acquisition process, Brainvision signal amplifiers were used [26].

The first 40 samples and the last 20 samples of each EEG signal were removed to reduce the effect of sequential images. Thus, the range (40ms-480ms) was used in this study similar to [26]. So, the size of input EEG signals is 440×128 (time × channels) .This dataset is divided into six different splits of training, validation, and test sets, with ratios of 80% (1600 images), 10% (200), 10% (200), respectively.

5. Experiments and Evaluations

The aim of this work is to improve the performance of the SNN in a two and three-class image classification problem based on transferring its feature manifold to

more separable space via a brain-guided system. SNN plays a role of image encoder in explicit path of a brain guided system.

Some classes were selected from the data set and trained for two-class classification problems. These classes are ‘Airliner’, ‘Pizza’, and ‘Fish’. "Airplane" and "Fish" classes are similar in terms of object structure in the image, and "Pizza" class is different from them in this aspect. Figure 4 shows some examples of images in these classes. Details of the number of images and EEG signals in these classes are given in Table 2.

Figure 4. Some examples of the selected classes

Table 2. Details of selected classes from ImageNet-EEG dataset in split # 0

Class	# Train images	# Validation images	# Test images	# Train EEG signal
Airliner	32	7	11	192
Fish	33	12	5	198
Pizza	30	11	9	180

As can be seen, each class contains a variety of images with variation in type, pose, light, background, etc. And also, the main body of objects are similar in different classes e.g. between Airline and Fish. In addition, the number of training images in each class is very small. It is obvious that SNN faces a challenging problem.

In the following, several experiments are conducted to evaluate the proposed method.

Experiment I: In this experiment, we evaluate the EEG encoder in the implicit path of the proposed brain-guided network. The proposed LSTM-based EEG network was trained for two and three class EEG classification by setting the parameters as listed in Table 3. In this EEG encoder, EEG features are extracted penultimate FC layer. Therefore, the size of the EEG feature vector is 128. The results are shown in Table 4.

Table 3. Architecture and parameter setting of the proposed LSTM-based EEG encoder

Parameter	Value
LSTM layer	1
Number of LSTM cell	128
LSTM layer input size	440×128
LSTM layer output size	440×128
Number of FC layer	2
Batch size	16
Optimizer	Adam
Learning rate	0.001
Weight decay	0.5
Epoch number	50
Loss function	Cross entropy

Table 4. Classification accuracy of the proposed EEG encoder

classes	Train accuracy (%)	Val accuracy (%)	Test accuracy (%)	Training time(s)
Airliner/Fish	98.96	100	96.8	28
Airliner/Pizza	99.18	100	96.43	27
Fish/Pizza	98.91	100	100	27
Airliner/Fish/Pizza	99.82	100	97.22	37
Average	99.21	100	97.61	29.75

As it can be seen from Table 4, this proposed encoder can encode EEG signals effectively.

Experiment II: As mentioned before, the SNN introduced in Table 1 is alone employed as the image encoder for image classification. In this step, images were resized to 64×64. The images are fed to SNN as input after a normalization and augmentation step.

SNN is trained with three different methods:

- **Back-propagation from the Scratch**: SNN is trained from the scratch using spike-based back-propagation with surrogate gradient-based optimization method. Here, linear approximation of the gradient is used for training.
- **Conversion ANN-SNN**: First an ANN is trained and then its weights are converted using threshold balancing method [15] and assigned to the corresponding SNN.
- **Hybrid Training:** First an ANN is trained and then its weights are converted as initial weights to SNN. Then, this SNN is trained with surrogate gradient-based STDB (Spike timing dependent back propagation) method [15]. In this method, exponential approximation of the gradient is used for training.

The parameters used for training the SNN-based image encoder are listed in Table 5.

Table 5. Architecture and parameter setting of the proposed SNN-based image encoder

Parameter	Value
Convolution layer	3
Average pooling layer	2
FC layer	2
Batch size	8
Optimizer	Adam
Learning rate	0.0001
Weight decay	0.0005
α	0.3
β	0.01
Epoch number	100
Loss function	Cross entropy

Figure 5 shows the results of these different methods of training the SNN to classify images for time steps.

According to the results, the conversion method seems more appropriate for SNN training in the studied dataset.

It can be seen that at small values of time steps, e.g. T = 15, the classification accuracy for training data is low. In the other hand, with increasing time steps, there is no significant change in the accuracy of data classification. Thus, given the low computational complexity and acceptable accuracy, we set the time steps to 50, empirically.

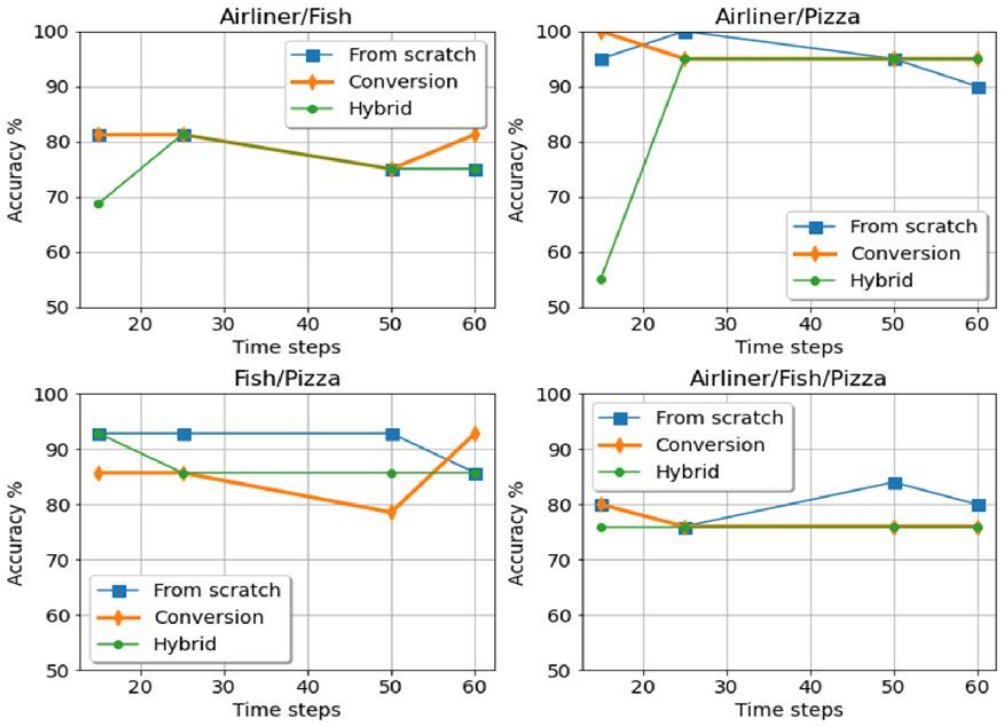


Figure 5. Accuracy of SNN in image classification with different time steps for the different training methods

Experiment III: Here, we evaluate the manifold transferring which is conducted from image-based visual feature space to EEG-based feature space in some aspects. After training SNN with conversion method, this network can be used as an image encoder in the brain-guided system. For each input image, the output of the AvgPool-2 of SNN is extracted as visual features vector with the size of 6272 (128×7×7).

At the same time, each EEG signal (corresponding to the each subject that observed each image) is encoded into a feature vector of size 128. In this way, six EEG-based feature vectors are extracted from the EEG encoder. The average of these feature vectors is considered as the EEG-based feature vector corresponding to that image.

M*k*-NN (k=5), KNN (k = 1, 3, or 5), Random Forest (RF) (number of estimators =100), and Ridge ($\alpha = 1$) regressions were used as manifold transferring method to transfer the visual feature space to EEG-based feature space.

After manifold transferring, a classifier that is trained on EEG-based feature space can classify the transferred feature vector. SVM, Soft-max, and k-NN (k=5) are first trained with EEG-based feature vectors extracted from EEG encoder, and then are examined as classifiers in the proposed brain-guided system.

The average of the accuracy of SVM classifier for different methods of MT, is plotted at time steps in Figure 6.

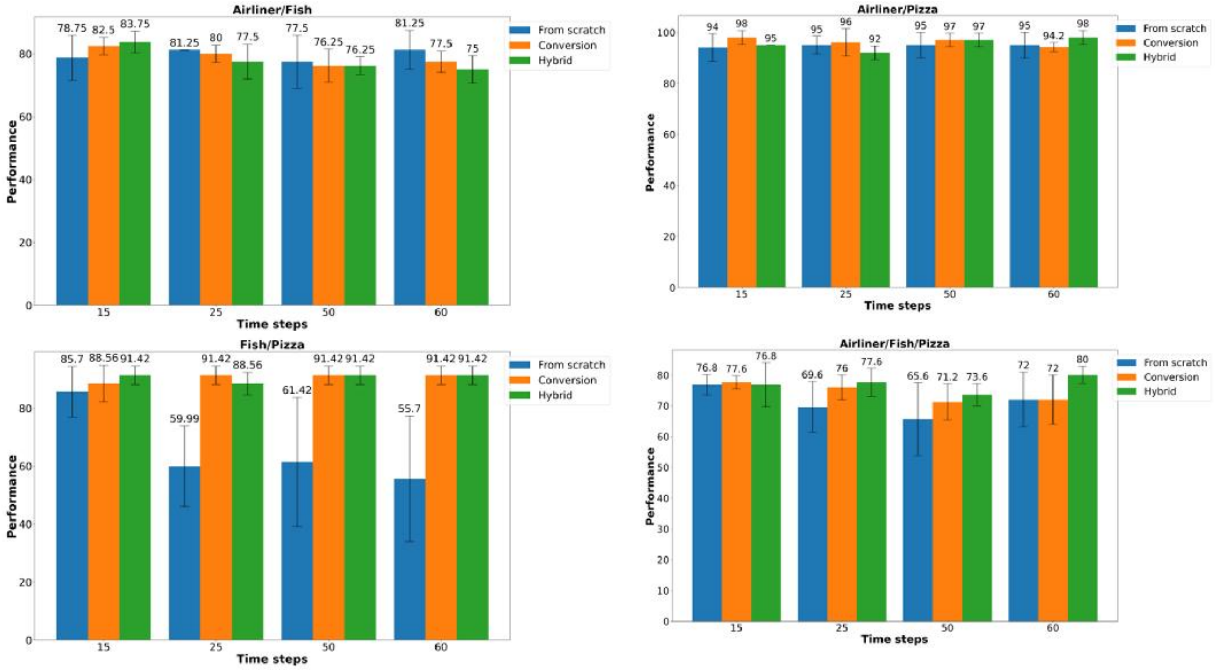


Figure 6. Brain-guided network average performance chart for different MT methods on various time steps, for classification on Airliner/Fish, Airliner/Pizza, Fish/Pizza, and Airliner/Fish/Pizza

The obtained classification results of three classifiers corresponding to different SNN training methods, with time steps equal to 50, are shown in Table 6.

As can be seen from Table 6, in the two-class classifications, the M*k*-NN MT method achieved better accuracy in most cases. In the three-class (Airliner/Fish/Pizza) problem, M*k*-NN, while not the winner, came second in classification accuracy in most cases. In addition, the accuracy of the brain-guided model on three-class classification with different MT methods tabulated in Table 7. As can be seen, the M*k*-NN MT method achieved better performance than KNN (k = 1, 3 and 5) in most cases.

Table 6. Accuracy of Brain-guided network in image classification.
In each classifier, the best accuracy is reported along with its MT method.

Classes	SNN training method	SNN Acc %	Brain guided model Acc %		
			SVM	SoftMax	KNN-5
Airliner/Fish	Back-propagation From the scratch	75	81.25 KNN-3, KNN-1, RF, Ridge	87.5 Ridge	81.25 KNN-3, KNN-1, RF, Ridge
	ANN-SNN Conversion	75	81.25 RF, Ridge, M*k*-NN	87.5 RF	81.25 RF, Ridge, M*k*-NN
	Hybrid Training	75	81.25 Ridge, M*k*-NN	81.25 Ridge	81.25 Ridge, M*k*-NN
Airliner/Pizza	Back-propagation From scratch	95	100 RF, Ridge	95 kNN-1, , M*k*-NN	100 RF, Ridge
	ANN-SNN Conversion	95	100 KNN-5, kNN-3, M*k*-NN	95 KNN-1, KNN-3,KNN-5, M*k*-NN	100 KNN-5, KNN-3, Mk-NN
	Hybrid Training	95	100 KNN-5, KNN-3, M*k*-NN	100 M*k*-NN	100 KNN-5, KNN-3, M*k*-NN
Fish/Pizza	Back-propagation From scratch	92.85	85.71 RF, Ridge	42.85 Ridge, kNN-1, M*k*-NN	85.71 RF, Ridge
	ANN-SNN Conversion	78.57	92.85 KNN-3, KNN-5 RF, Ridge, M*k*-NN	85.71 KNN-1	92.85 KNN-3, KNN-5 RF, M*k*-NN
	Hybrid Training	85.71	92.85 KNN-3, KNN-5 RF, Ridge, M*k*-NN	85.71 KNN-1	92.85 KNN-3, KNN-5 RF, Ridge, M*k*-NN
Airliner/Fish/Pizza	Back-propagation From scratch	84	80 Ridge	76 RF	84 Ridge
	ANN-SNN Conversion	76	80 RF	80 RF	84 Ridge
	Hybrid Training	76	76 Ridge, RF, KNN-3	76 kNN-5	76 Ridge, RF

1

2

Table 7. Accuracy of Brain-guided network in Airliner/Fish/Pizza image classification.

SNN training method	MT method	Brain guided model Acc % Classifier		
		SVM	SoftMax	KNN-5
Back-propagation From scratch	M*k*-NN	64	68	64
	KNN-1	60	64	60
	KNN-3	52	60	52
	KNN-5	60	60	56
	RF	76	76	76
	Ridge	80	64	84
ANN-SNN Conversion	M*k*-NN	76	76	76
	KNN-1	68	68	68
	KNN-3	72	72	72
	KNN-5	64	64	64
	RF	80	80	80
	Ridge	72	76	84
Hybrid Training	M*k*-NN	72	72	72
	KNN-1	72	72	72
	KNN-3	76	72	72
	KNN-5	68	76	68
	RF	76	76	76
	Ridge	76	68	76

According to Table 8, the brain-guided model achieves more improvements in image classification accuracy, when M*k*-NN and KNN-5 are used to manifold transferring and classifier, respectively.

Table 8. The improvement of classification accuracy in brain-guided network compared to the SNN

SNN training method	Classes	Improvement % MT method - Classifier						
		RF - SVM	RF - KNN-5	Ridge-SVM	KNN-3 - SVM	Ridge-KNN-5	M*k*-NN-SVM	M*k*-NN-kNN-5
From scratch	Airliner/Fish	+6.25	+6.25	+6.25	+6.25	+6.25	0	0
	Airliner/Pizza	+5	+5	+5	0	+5	0	0
	Fish/Pizza	-7.14	-7.14	-7.14	-10	-7.14	-12	-12
	Airliner/Fish/Pizza	-8	-8	-4	-8	0	-20	-20
Conversion	Airliner/Fish	+6.25	+6.25	+6.25	0	+6.25	+6.25	+6.25
	Airliner/Pizza	0	0	0	+5	0	+5	+5
	Fish/Pizza	+14.28	+14.28	+14.28	+14.28	+7.14	+14.28	+14.28
	Airliner/Fish/Pizza	+4	+4	0	0	+8	-2	0
Hybrid	Airliner/Fish	0	0	+6.25	0	+6.25	+6.25	+6.25
	Airliner/Pizza	0	0	0	+5	0	+5	+5
	Fish/Pizza	+7.14	+7.14	+7.14	+7.14	+7.14	+7.14	+7.14
	Airliner/Fish/Pizza	0	0	0	0	0	-4	-4

In general, the brain-guided model increases the accuracy of SNN-based image classification at most about 14.25%. When the SNN is trained in the conversion method, it leads to more improvement in the accuracy of the brain-guided network than the others.

Figure 7 shows the t-SNE plots of features extracted from the image encoder and EEG encoder in the brain-guided network. As can be seen, when SNN is trained in conversion or hybrid method, the visual features are more discriminative.

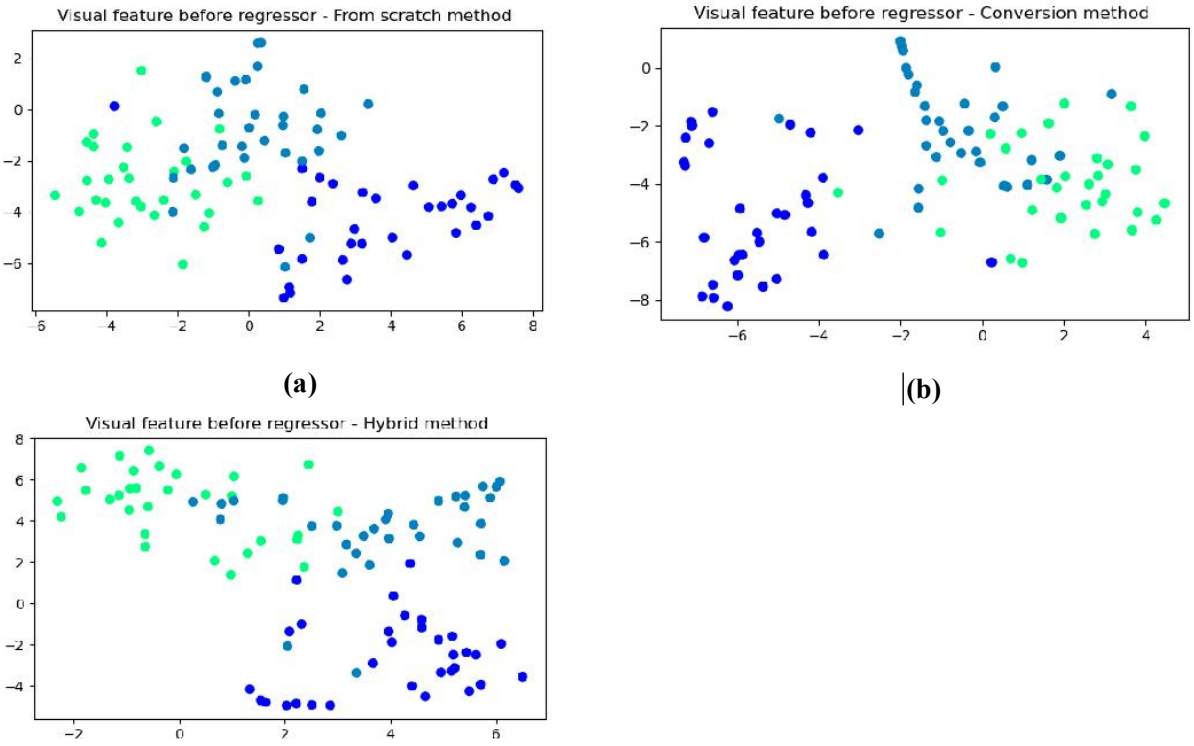


Figure 7. t-SNE illustration of training visual features extracted by SNN-based image encoder in Brain-guided model for Airliner / Fish / Pizza, SNN is trained by (a) From scratch method, (b) Conversion method and (c) Hybrid method

Experiment IV: In this section, an analysis is performed on the features extracted from the SNN-based image encoder. For this purpose, the Fisher criterion is used.

If the x_{nk} is the *n*-th instance of the *k*-th class. In this case, the between-scatter matrix and within scatter matrix are respectively defined as [40]:

$$S_B = \sum_{k=1}^{K} (m_k - m) N_k (m_k - m)^T \tag{7}$$

$$S_w = \sum_{k=1}^{K} \sum_{n=1}^{N_k} (x_{nk} - m_k)(x_{nk} - m_k)^T \tag{8}$$

where *K* is the number of classes and N_k is the number of samples in the *k*-th class. m and m_k are the total mean and average in *k*-th class, respectively.

The Fisher criterion is defined as follows:

$$Fisher\ criterion = \frac{tr(S_B S_B{}')}{tr(S_w S_w{}')} \tag{9}$$

Comparison of Fisher discrimination criterion for visual features extracted from SNNs with different training methods is performed in Figure 8.

As can be seen, for Airliner/Pizza and Airliner/Fish/Pizza categories this criterion in the Hybrid method is higher than the other two methods. For all classes, the Conversion method is higher than from the scratch method. The geometric mean of Fisher ratio shows that the relative performance of the Hybrid method is about 1.078 higher than Conversion in the four/ (three two-class) studied classification problems.

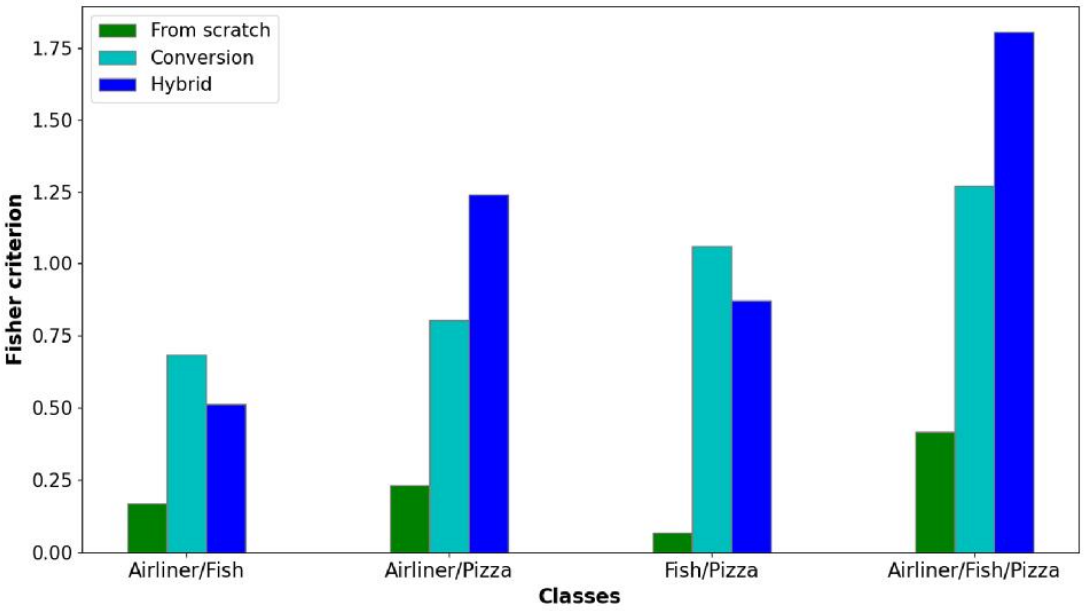
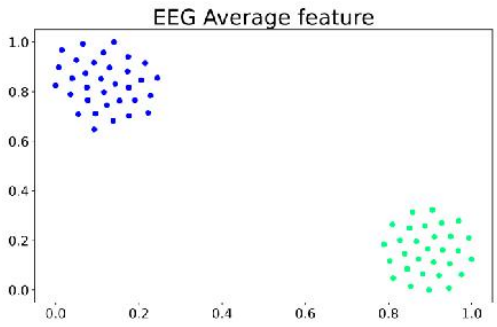


Figure 8. Fisher discrimination criterion for visual features extracted from SNNs with the different training methods

Experiment V: Here, a study has been done on the effect of manifold transferring on the separability of visual features. We show the improvement of separability of the SNN-based visual feature space in the proposed method in three ways:

- **t-SNE plot**: Figure 9 shows the t-SNE plot for features extracted from encoders (EEG features and Visual features). As can be seen, the visual features have a higher discriminability after mapping to the EEG feature space by manifold transferring.

(a)



(b)

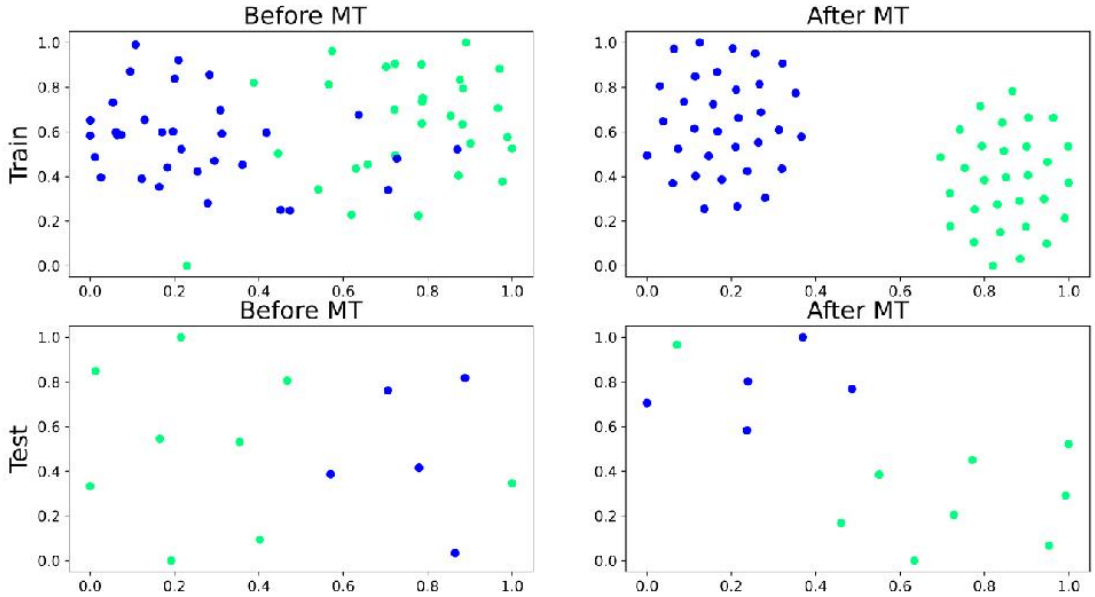


Figure 9. t-SNE illustration of features extracted by encoders in Brain-guided model for Fish / Pizza, (a) EEG average features, (b) Visual features before and after manifold transferring for train and test data.

- **Fisher criterion**: Equation 9 in the expresses Fisher's criterion as the ratio of extra-class variance to intra-class variance. Qualitatively, if the intra-class variance is low and the extra-class variance is high, the distinction becomes more straightforward. In Figures 10 and 11, Fisher's criterion is calculated as the degree of separability of visual features before and after manifold transferring. Figure 10 shows that the separability of the visual features is increased by MT, the separability in the choosed training data is higher than in the test data.

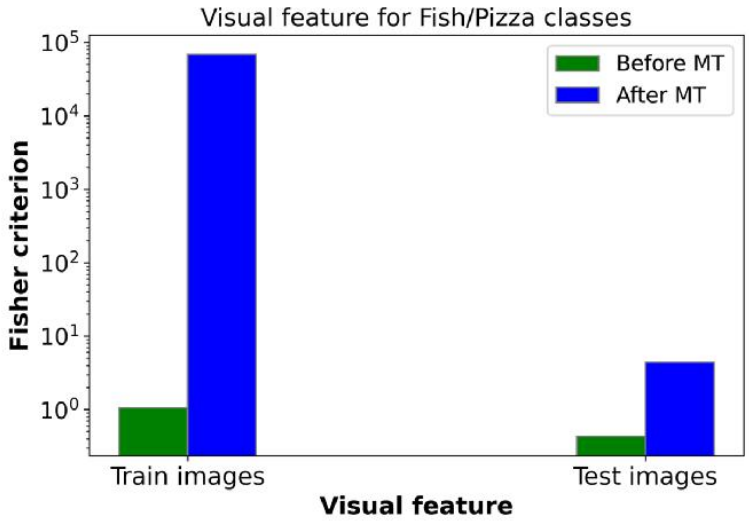


Figure 10. Comparison of Fisher discrimination criterion for visual features before and after MT in test and train images for Fish/Pizza classification

Figure 11 shows that the separability of the train visual features is increased by MT for different classification problems in this work.

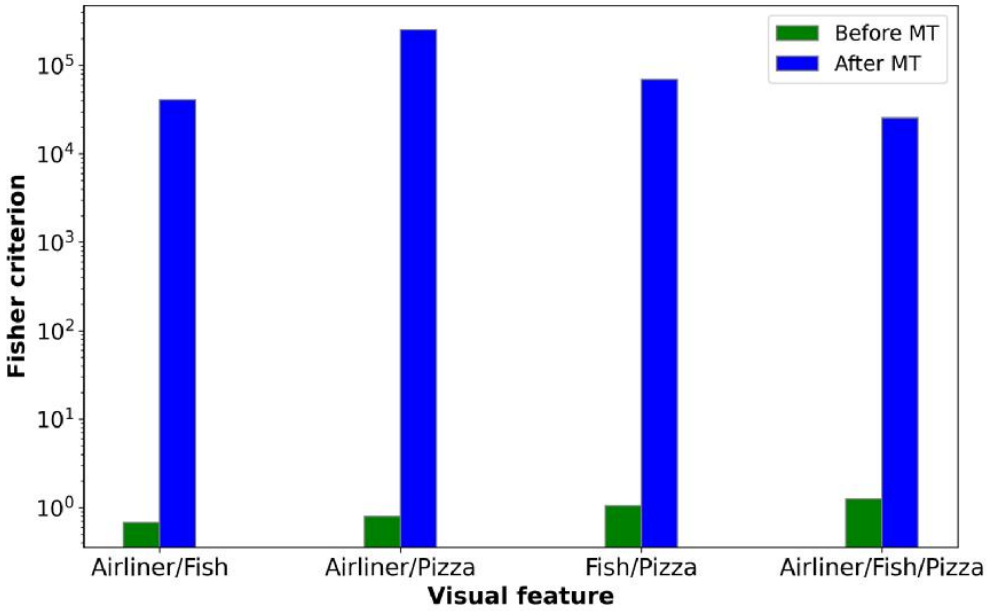


Figure 11. Comparison of Fisher discrimination criterion for train visual features before and after MT

- **SVM margin:** The SVM margin across the training set corresponds to generalization properties in linear models [41]. In Figures 12, SVM margin show as the separability of the train visual features increases after MT for all classification problems of this research.

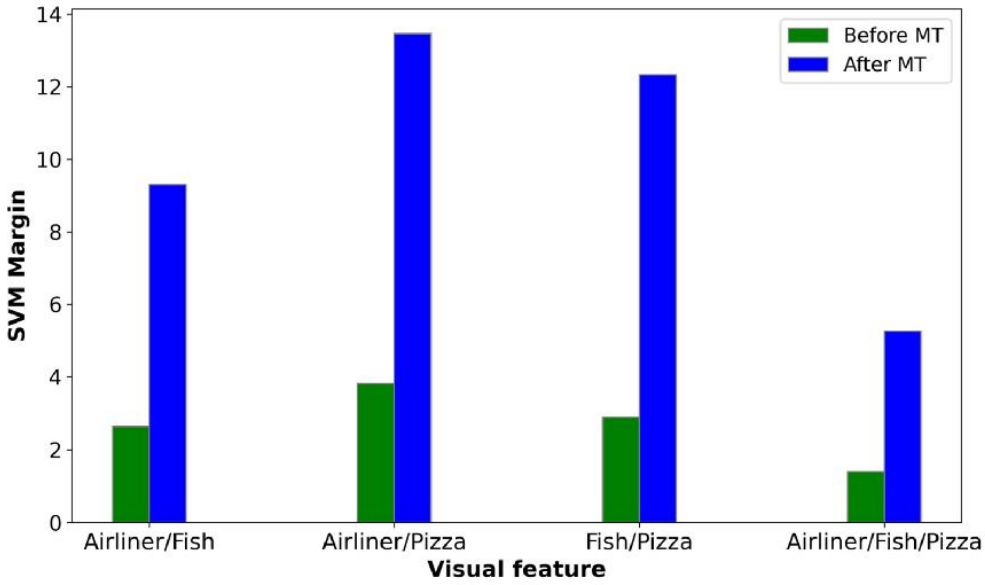


Figure 12. Comparison of SVM margin for train visual features before and after MT.

Experiment VI: In this part, we employed the spiking LSTM presented in [42] to encode the EEG signals ,with architecture same as Figure 2, to classify the image-evoked EEG signals. Therefore, the number of weights in the LSTM-based and spiking-LSTM-based encoders is the same. The training results of this EEG encoder are shown in Table 9. As can be seen, in the 3-classes classification problem, the accuracies have decreased compared to Table 4. Also, the Spiking LSTM-based encoder outperforms are lower than in the LSTM-based encoder in view of average accuracy.

Table 9. EEG classification accuracy with spiking LSTM substituted for LSTM in Figure 2.

Classes	Train accuracy	Validation accuracy	Test accuracy
Airliner/Fish	99.74	98.21	93.75
Airliner/Pizza	99.73	100	98.21
Fish/Pizza	99.73	100	98.75
Airliner/Fish/Pizza	98.75	96.59	93.75
Average	99.48	98.7	96.11

6. CONCLUSION

In this work, we attempt to employ the simplicity of SNN with EEG-based discriminative feature space. To this end, we assembled an SNN-based brain-guided system of LSTM-based EEG encoder, visual image decoder based on a SNN which is trained with conversion method, manifold transferring based on mutual k-NN, and SVM-based classifier. In the proposed model, the EEG signal is not needed after training. The test image is converted into spike patterns and mapped to LSTM-based EEG feature space. An SVM-based classifier assigns a class label to the resulted feature vector and corresponding image. Different aspects of the proposed system are investigated on the challenging small-sample ImageNet-EEG dataset.

Various analyzes have been performed on the results obtained from SNN-based image encoders and compared with other conventional methods for training SNN. The results show that the training method in SNN can affect the degree of discrimination of the features extracted from the images. According to the obtained results, it can be said that one can sort these methods with respect to their appropriation for training SNN-based image encoder in the brain-guided model, first the conversion and then hybrid, finally the training from the scratch method.

Also, the experimental results show that the separability of the SNN-based visual feature space is improved in the proposed method. In addition, this proposed SNN-embedding in brain guided system improves the accuracy of SNN in image classification up to 14.25% on the challenging small-sample ImageNet-EEG dataset without any overfitting phenomenon.

Declarations

Author Contribution: Zahra Imani: Conceptualization, Methodology, Software, Formal analysis, Validation, Visualization, Writing - original draft. Mehdi Ezoji: Supervision, Conceptualization, Methodology, Investigation, Validation, Writing - review & editing. Timothee Masquelier: Supervision, Conceptualization, Methodology, Investigation, Validation, Writing - review & editing.

Conflict of Interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Availability of Data and Materials: The custom code for data analysis is available upon request from the corresponding author.

Funding: No funding was obtained for this study.

REFERENCES

[1] G. Srinivasan, C. Lee, A. Sengupta, P. Panda, S. S. Sarwar, and K. Roy, “Training Deep Spiking Neural Networks for Energy-Efficient Neuromorphic Computing,” in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 8549–8553.

[2] Q. Fu and H. Dong, “An ensemble unsupervised spiking neural network for objective recognition,” *Neurocomputing*, vol. 419, pp. 47–58, 2021.

[3] T. Masquelier, “Spike-based computing and learning in brains, machines, and visual systems in particular (HDR Report).” Ph. D. Dissertation. https://doi. org/10.13140/RG. 2.2. 30232.49922, 2017.

[4] C. Lee, P. Panda, G. Srinivasan, and K. Roy, “Training deep spiking convolutional neural networks with stdp-based unsupervised pre-training followed by supervised fine-tuning,” *Front. Neurosci.*, vol. 12, p. 435, 2018.

[5] B. Rueckauer, I.-A. Lungu, Y. Hu, M. Pfeiffer, and S.-C. Liu, “Conversion of continuous-valued deep networks to efficient event-driven networks for image classification,” *Front. Neurosci.*, vol. 11, p. 682, 2017.

[6] P. Panda, S. A. Aketi, and K. Roy, “Toward scalable, efficient, and accurate deep spiking neural networks with backward residual connections, stochastic softmax, and hybridization,” *Front. Neurosci.*, vol. 14, 2020.

[7] A. Sengupta, Y. Ye, R. Wang, C. Liu, and K. Roy, “Going deeper in spiking neural networks: VGG and residual architectures,” *Front. Neurosci.*, vol. 13, 2019.

[8] J. Wu *et al.*, “Progressive tandem learning for pattern recognition with deep spiking neural networks,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 7824–7840, 2021.

[9] E. O. Neftci, H. Mostafa, and F. Zenke, “Surrogate gradient learning in spiking neural networks,” *IEEE Signal Process. Mag.*, vol. 36, pp. 61–63, 2019.

[10] S. B. Shrestha and G. Orchard, “Slayer: Spike layer error reassignment in time,” *Adv. Neural Inf. Process. Syst.*, vol. 31, pp. 1412–1421, 2018.

[11] M. Zhang *et al.*, “Rectified linear postsynaptic potential function for backpropagation in deep spiking neural networks,” *IEEE Trans. Neural Networks*

Learn. Syst., vol. 33, no. 5, pp. 1947–1958, 2021.

[12] Y. Wu, L. Deng, G. Li, J. Zhu, and L. Shi, "Spatio-temporal backpropagation for training high-performance spiking neural networks," *Front. Neurosci.*, vol. 12, p. 331, 2018.

[13] G. Bellec, D. Salaj, A. Subramoney, R. Legenstein, and W. Maass, "Long short-term memory and learning-to-learn in networks of spiking neurons," *Adv. neural Inf. Process. Syst.*, vol. 31, 2018.

[14] E. O. Neftci, H. Mostafa, and F. Zenke, "Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks," *IEEE Signal Process. Mag.*, vol. 36, no. 6, pp. 51–63, 2019.

[15] N. Rathi, G. Srinivasan, P. Panda, and K. Roy, "Enabling deep spiking neural networks with hybrid conversion and spike timing dependent backpropagation," *arXiv Prepr. arXiv2005.01807*, 2020.

[16] I. Garg, S. S. Chowdhury, and K. Roy, "DCT-SNN: Using DCT To Distribute Spatial Information Over Time for Low-Latency Spiking Neural Networks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 4671–4680.

[17] S. Kundu, M. Pedram, and P. A. Beerel, "Hire-snn: Harnessing the inherent robustness of energy-efficient deep spiking neural networks by training with crafted input noise," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5209–5218.

[18] O. Russakovsky *et al.*, "Imagenet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[19] E. Lee, J.-S. Choi, M. Kim, H.-I. Suk, and A. D. N. Initiative, "Toward an interpretable Alzheimer's disease diagnostic model with regional abnormality

representation via deep learning," *Neuroimage*, vol. 202, p. 116113, 2019.

[20] R. Ju, C. Hu, and Q. Li, "Early diagnosis of Alzheimer's disease based on resting-state brain networks and deep learning," *IEEE/ACM Trans. Comput. Biol. Bioinforma.*, vol. 16, no. 1, pp. 244–257, 2017.

[21] Y. Yoo *et al.*, "Deep learning of joint myelin and T1w MRI features in normal-appearing brain tissue to distinguish between multiple sclerosis patients and healthy controls," *NeuroImage Clin.*, vol. 17, pp. 169–178, 2018.

[22] S. L. Oh *et al.*, "A deep learning approach for Parkinson's disease diagnosis from EEG signals," *Neural Comput. Appl.*, pp. 1–7, 2018.

[23] T. Song, W. Zheng, P. Song, and Z. Cui, "EEG emotion recognition using dynamical graph convolutional neural networks," *IEEE Trans. Affect. Comput.*, vol. 11, no. 3, pp. 532–541, 2018.

[24] S. Taheri, M. Ezoji, and S. M. Sakhaei, "Convolutional neural network based features for motor imagery EEG signals classification in brain–computer interface system," *SN Appl. Sci.*, vol. 2, no. 4, pp. 1–12, 2020.

[25] H. Jebelli, M. M. Khalili, and S. Lee, "Mobile EEG-based workers' stress recognition by applying deep neural network," in *Advances in Informatics and Computing in Civil and Construction Engineering*, Springer, 2019, pp. 173–180.

[26] X. Zheng, W. Chen, Y. You, Y. Jiang, M. Li, and T. Zhang, "Ensemble deep learning for automated visual classification using EEG signals," *Pattern Recognit.*, vol. 102, p. 107147, 2020.

[27] I. Kavasidis, S. Palazzo, C. Spampinato, D. Giordano, and M. Shah, "Brain2image: Converting brain signals into images," in *Proceedings of the 25th ACM international conference on Multimedia*, 2017, pp. 1809–1817.

[28] B. Kaneshiro, M. P. Guimaraes, H.-S. Kim, A. M. Norcia, and P. Suppes, "A

representational similarity analysis of the dynamics of object processing using single-trial EEG classification," *PLoS One*, vol. 10, no. 8, p. e0135697, 2015.

[29] A. Fares, S. Zhong, and J. Jiang, "EEG-based image classification via a region-level stacked bi-directional deep learning framework," *BMC Med. Inform. Decis. Mak.*, vol. 19, no. 6, p. 268, 2019.

[30] C. Spampinato, S. Palazzo, I. Kavasidis, D. Giordano, N. Souly, and M. Shah, "Deep learning human mind for automated visual classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6809–6817.

[31] S. Zhong, A. Fares, and J. Jiang, "An Attentional-LSTM for Improved Classification of Brain Activities Evoked by Images," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 1295–1303.

[32] S. Palazzo, C. Spampinato, I. Kavasidis, D. Giordano, J. Schmidt, and M. Shah, "Decoding brain representations by multimodal learning of neural activity and visual features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 11, pp. 3833–3849, 2020.

[33] J. Jiang, A. Fares, and S.-H. Zhong, "A Context-Supported Deep Learning Framework for Multimodal Brain Imaging Classification," *IEEE Trans. Human-Machine Syst.*, vol. 49, no. 6, pp. 611–622, 2019.

[34] N. Cudlenco, N. Popescu, and M. Leordeanu, "Reading into the mind's eye: boosting automatic visual recognition with EEG signals," *Neurocomputing*, vol. 386, pp. 281–292, 2020.

[35] A. Taherkhani, A. Belatreche, Y. Li, G. Cosma, L. P. Maguire, and T. M. McGinnity, "A review of learning in biologically plausible spiking neural networks," *Neural Networks*, vol. 122, pp. 253–272, 2020.

[36] Reza Hojjaty Saeedy, “Biologically Inspired Computer Vision/ Applications of Computational Models of Primate Visual Systems in Computer Vision and Image Processing,” University of New Hampshire, 2021.

[37] L. Deng *et al.*, “Rethinking the performance comparison between SNNs and ANNs,” *Neural networks*, vol. 121, pp. 294–307, 2020.

[38] W. Fang, Z. Yu, Y. Chen, T. Huang, T. Masquelier, and Y. Tian, “Deep Residual Learning in Spiking Neural Networks,” *Adv. Neural Inf. Process. Syst.*, vol. 34, pp. 21056–21069, 2021.

[39] R. Hajizadeh, A. Aghagolzadeh, and M. Ezoji, “Mutual neighborhood and modified majority voting based KNN classifier for multi-categories classification,” *Pattern Anal. Appl.*, vol. 25, no. 4, pp. 773–793, 2022.

[40] B. Ghojogh, F. Karray, and M. Crowley, “Fisher and kernel Fisher discriminant analysis: Tutorial,” *arXiv Prepr. arXiv1906.09436*, 2019.

[41] Y. Jiang, D. Krishnan, H. Mobahi, and S. Bengio, “Predicting the generalization gap in deep networks with margin distributions,” *arXiv Prepr. arXiv1810.00113*, 2018.

[42] S. V. Ali Lotfi Rezaabad, “Long short-term memory spiking networks and their applications,” in *International Conference on Neuromorphic Systems*, 2020, pp. 1–9.