Information Sciences 547 (2021) 1025–1044

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Multi-subject data augmentation for target subject semantic decoding with deep multi-view adversarial learning

Dan Li [a,b], Changde Du [a,b,c], Shengpei Wang [a,b], Haibao Wang [a,b], Huiguang He [a,b,d,*]

[a] Research Center for Brain-inspired Intelligence and National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
[b] University of Chinese Academy of Sciences, Beijing 100190, China
[c] Huawei Cloud BU EI Innovation Lab, Beijing 100085, China
[d] Center for Excellence in Brain Science and Intelligence Technology, Chinese Academy of Sciences, Beijing 100190, China

ARTICLE INFO

Article history:
Received 21 March 2020
Received in revised form 1 August 2020
Accepted 11 September 2020
Available online 9 October 2020

Keywords:
Data augmentation
Semantic decoding
Multi-view adversarial learning
Sparse reconstruction relation

ABSTRACT

Functional magnetic resonance imaging (fMRI) is widely used in the field of brain semantic decoding. However, as fMRI data acquisition is time-consuming and expensive, the number of samples is usually small in the existing fMRI datasets. It is difficult to build an accurate brain decoding model for a subject with insufficient fMRI data. The majority of semantic decoding methods focus on designing predictive model with limited samples, while less attention is paid to fMRI data augmentation. Leveraging data from related but different subjects can be regarded as a new strategy to improve the performance of predictive model. There are two challenges when using information from different subjects: 1) feature mismatch; 2) distribution mismatch. In this paper, we propose a multi-subject fMRI data augmentation method to address the above two challenges, which can improve the decoding accuracy of the target subject. Specifically, the subject information can be translated from one to another by using multiple subject-specific encoders, decoders and discriminators. The encoder maps each subject to a shared latent space, solving the feature mismatch problem. The decoders and discriminators form multiple generative adversarial network architectures, which solves the distribution mismatch problem. Meanwhile, to ensure that the representation of the latent space preserves information of the input space, our method not only minimizes the local data reconstruction loss, but also preserves the sparse reconstruction (semantic) relation over the whole dataset of the input space. Extensive experiments on three fMRI datasets demonstrate the effectiveness of the proposed method.

1. Introduction

Semantic decoding, as a technology to decode stimulus label from human brain activity, has attracted great interests in recent years. Building an accurate decoding model is helpful for us to understand the brain mechanisms [1,2]. Among the many brain activity acquisition techniques, functional magnetic resonance imaging (fMRI) shows great advantages in

* Corresponding author at: Research Center for Brain-inspired Intelligence and National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China.
E-mail addresses: lidan2017@ia.ac.cn (D. Li), duchangde@gmail.com (C. Du), wangshengpei2014@ia.ac.cn (S. Wang), wanghaibao2017@ia.ac.cn (H. Wang), huiguang.he@ia.ac.cn (H. He).

https://doi.org/10.1016/j.ins.2020.09.012
0020-0255/

semantic decoding task due to its high spatial resolution [1,3,4]. The overall procedure of fMRI-based semantic decoding is illustrated in Fig. 1. The fMRI signals used for semantic classification are recorded when the subjects are viewing the stimuli. Then, the semantic decoding model is built with the processed fMRI data. Finally, the brain mechanism can be analyzed in accordance with the prediction results of the semantic decoding model.

In reality, learning an accurate decoding model from fMRI data is still challenging. The decoding model usually requires plenty of data to learn a large number of parameters, because the dimensions of fMRI data are very high. However, the number of samples of a single subject is usually small in the existing fMRI dataset, as fMRI data acquisition is very costly. To deal with this issue, we can add more samples to the training set, making it an enlarged dataset. An enlarged dataset allows us to train more complex models and to reduce the impact of over-fitting. Based on these motivations, Zhuang et al. proposed an fMRI data augmentation method which uses generative models [5,6] to generate new samples [7]. The inputs of generative models are multivariate gaussian noise, therefore the method cannot generate samples that contain further information about the data than that already contained in the dataset. Zhang et al. proposed a method to improve the prediction accuracy on a primary fMRI dataset by jointly learning a model using other secondary fMRI datasets [8]. It is useful when the primary dataset is small. However, the method is not applicable when no subjects are shared between datasets, which is common for many fMRI datasets.

Compared with the previous methods, this paper resorts to an explicit strategy from the view of multi-subject fMRI data augmentation. We hope to explore information from multiple source subjects to overcome the insufficient training data problem of the target subject. The individual differences present two main challenges when attempting to utilize fMRI data from multiple subjects: feature mismatch and distribution mismatch. Feature mismatch refers to the fact that even the fMRI data of distinct subjects are induced by the same stimulus, the features (voxels) that are actually recorded for each subject may vary. Distribution mismatch means that the distribution of the brain responses of each subject is inconsistent. This paper provides a natural solution to both of challenges.

Based on the above discussions, we propose a multi-subject fMRI data augmentation method to improve the semantic decoding performance of target subject. Fig. 2 shows the whole framework of the proposed method. This method utilizes fMRI data from multiple source subjects to effectively enlarge the data of target subject. To solve the feature mismatch problem, we map the fMRI data of all the subjects to a share latent space, which also enables the multiple translations to be learned efficiently and simultaneously. To ensure that the representation of the latent space preserves information of the input space, the local reconstruction loss and the sparse reconstruction relation are included in this method. We use multiple generative adversarial network (GAN) [5] architectures to translate the subject information from one to another. It ensures that the learned translation respects the distribution of the target subject. With the learned multi-subject data augmentation model, we can generate new training samples from multiple source subjects for the target subject. In this manner, the training set of a subject is the combination of the original training samples and the augmented samples. The augmented samples can directly borrow the label from the source subjects. During training, we use the new training set for semantic classifier training. Experiments on the Handwritten Characters [9], Haxby [10] and CMU2008 [11] datasets demonstrate that the proposed method achieves state-of-the-art performance.

In summary, our main contributions are featured in the following aspects:

- We provide a new insight into the problem of brain semantic decoding. That is, we introduce a multi-subject fMRI data augmentation method to improve the performance of the target subject.
- A latent space is introduced to solve the problem of feature mismatch and multiple GAN architectures are introduced to solve the problem of distribution mismatch between distinct subjects.
- The experimental results show that our method is better than the baseline methods, especially when the size of the training data is small.

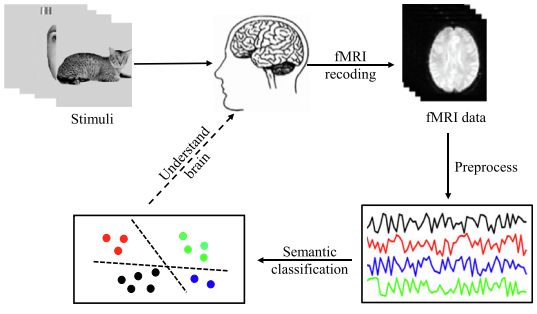

Fig. 1. Fundamental modules of fMRI semantic decoding. Firstly, the fMRI signals used for semantic classification are recorded when the subjects are viewing the stimuli. Then the decoding model is built with preprocess data to predict stimuli labels. Decoding models are also used to identify brain regions involved in the cognitive operations related to the observed stimuli. These models have become a standard tool in neuroimaging data analysis. In this paper, we focus on building an accurate semantic decoding model.

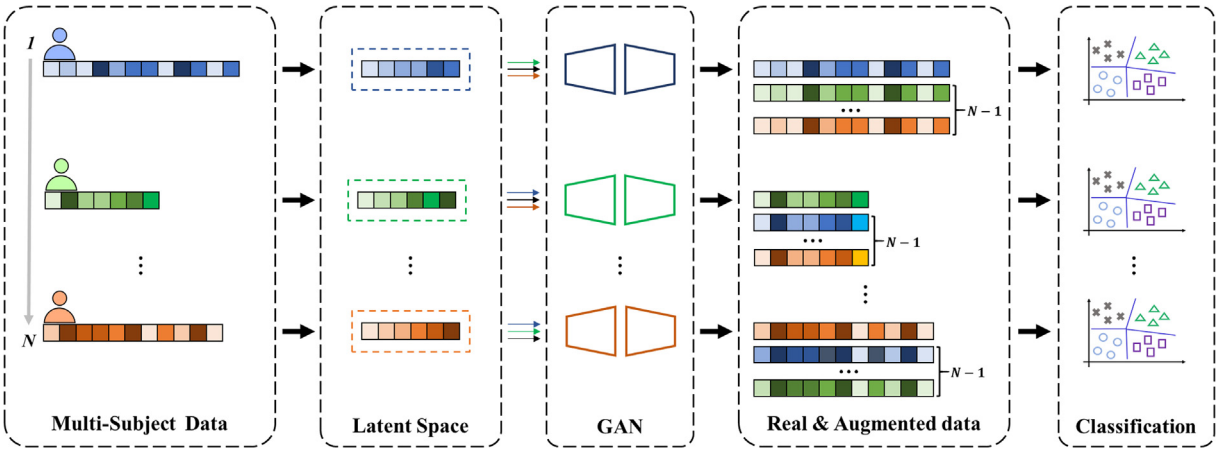

Fig. 2. An overview of the proposed multi-subject fMRI data augmentation semantic decoding framework. We first map fMRI data from original space to latent space to solve feature mismatch problem between the different subjects. To solve distribution mismatch problem, we introduce multiple GAN architectures. For each target subject, we input the information from other subjects in the latent space into the subject-specific GAN for data augmentation. In this manner, the training set of a subject is a combination of real fMRI training data and augmented data (in the fourth part, for each target subject, the first row is real training data and the other rows are augmented data). We further use the new training set for semantic classification.

2. Related works

2.1. Brain decoding

Brain decoding, including semantic decoding [4,12–14], image identification [15,16], image reconstruction [17,18], and etc., can be viewed as the study of finding the corresponding stimuli by the evoked brain responses. Semantic decoding is one of the most important parts of brain decoding, and previous studies have made significant progress in brain semantic decoding [4,12,13,19,20]. For example, Wen et al. first used the group sparse Bayesian logistic regression approach to select fMRI voxel, and then they constructed a classifier to classify the fMRI data [20]. Xu et al. proposed a fMRI data classification framework [13] based on neural network and support vector machine (SVM) [21], in which they first used hierarchical tensor training layer to extract latent representations for fMRI data, and then used SVM to classify these representations. The above models are built from the perspective of feature selection, while some researchers also study fMRI semantic decoding model from the perspective of classifier ensembles. Kuncheva et al. proposed a random subspace (RS) ensemble method to classify the single subject fMRI data, and the classification result is obtained by majority vote [4].

Most of the above methods are trained based on the limited samples of a single subject. If the built model is very complex, it will lead to overfitting. Different from them, we first use the information of multi-subject to enlarge the data of the target subject. Then, the new training set is used to build the predictive model for the target subject.

2.2. Generative adversarial network

Generative adversarial network (GAN) [5] has achieved great success in the field of image generation. Recently, many extensions of GAN have been applied to the data augmentation task [22–25].

The conditional GAN framework presented in [26] provides an algorithm for generating samples conditioned on discrete labels. Luo et al. [22] introduces a conditional GAN framework for electroencephalography (EEG) data augmentation to improve EEG-based emotion recognition task. As the input of conditional GAN is simple random noise, it can only generate samples that contain the information already contained in the target subject.

Cycle-GAN [23] is a framework for estimating cycle-consistent and reversible mappings between two domains. Zhong et al. used multiple Cycle-GAN models for image data augmentation to improve the performance of person re-identification task [27]. However, Cycle-GAN is not scalable to multiple domains because the number of mappings to be learned is $N(N-1)$ when we have N domains.

StarGAN [24] is a multi-domain translation method that is scalable to multiple domains by using a single generator. However, when the mapping functions between different pairs are significantly different, using a single network as mapping functions may limit its performance.

RadialGAN [25] is a method to effectively enlarge the target dataset by utilizing the information from multiple source datasets. It mainly studies the case that the data labels are continuous in the data augmentation process. Our work is motivated by RadialGAN. Compared with the RadialGAN, our method not only minimizes local data reconstruction loss, but also preserves the sparse reconstruction (semantic) relation over the whole dataset of the subject when solving the feature mismatch problem. The introduction of reconstruction relation matrix can also better deal with label missing scenarios than

RadialGAN method. Besides, we extend the improved method to the fMRI visual information semantic decoding field which data labels are discrete.

3. Proposed method

Suppose that we have N subjects corresponding N data spaces $\mathcal{X}^{(1)},\ldots,\mathcal{X}^{(N)}$ and a label space $\mathcal{Y}$, $\mathbf{X}^{(i)}$ denotes the fMRI activity patterns of the i-th subject taken from $\mathcal{X}^{(i)}$, $\mathbf{Y}$ denotes the semantic label matrix of the subject taken from $\mathcal{Y}$. Here, $\mathbf{X}^{(i)} \in \mathbb{R}^{n_i \times d_i}$, n_i denotes the size of the training set for the i-th subject and d_i denotes the dimension of fMRI data for the i-th subject. The training set consists of N paired datasets, which can be denoted by $\mathbf{S}_1 = \left(\mathbf{X}^{(1)}, \mathbf{Y}\right), \ldots, \mathbf{S}_N = \left(\mathbf{X}^{(N)}, \mathbf{Y}\right)$. Our goal is to utilize the datasets $\{S_j : j \neq i\}$ (as well as S_i) to learn the classifier C_i, where $C_i : \mathcal{X}^{(i)} \rightarrow \mathcal{Y}$ $(i = 1 \ldots N)$. The illustration of our method is shown in Fig. 3 and Fig. 4. Specifically, each subject contains the specific encoder E_i, decoder G_i, and discriminator D_i $(i = 1,\ldots,N)$. The encoder solves the feature mismatch problem between different subjects by mapping the fMRI data of a subject from original space to the latent space $\mathcal{Z}$. Meanwhile, we incorporate the sparse information $\mathbf{R}_i$ $(i = 1,\ldots,N)$ to

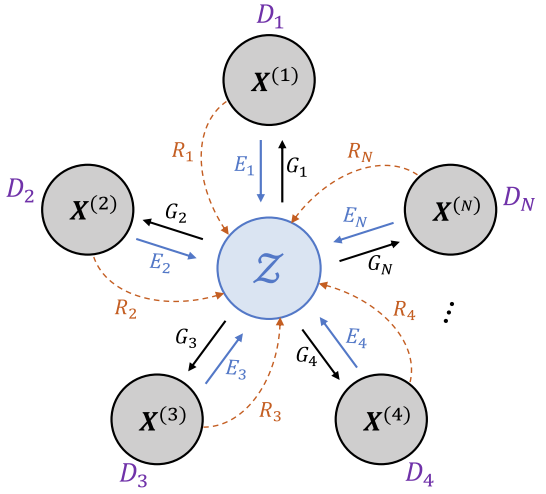

Fig. 3. The flowchart of our method. $\mathcal{Z}$ denotes latent space. $\mathbf{X}^{(i)}$ is the fMRI data of the i-th subject. E_i, G_i and D_i denote encoder, decoder and discriminator, respectively. $\mathbf{R}_i$ is the reconstruction (semantic) relation of i-th subject. The data of i-th subject is translated to the j-th subject via $\mathcal{Z}$ using E_i and G_j.

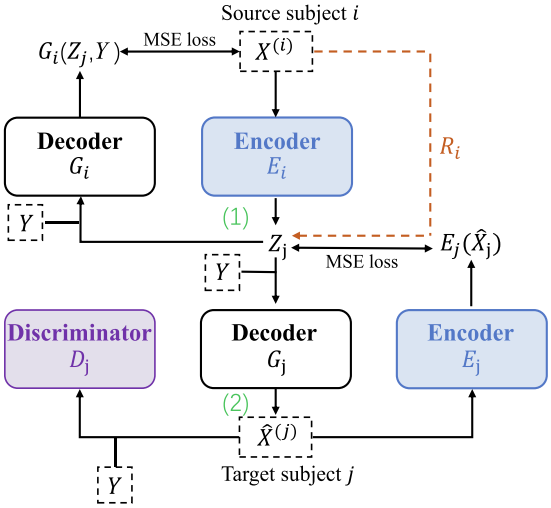

Fig. 4. The detailed flowchart of our method. (1) The encoder E_i maps $\mathbf{X}^{(i)}$ in the source subject i to the latent space. (2) The decoder G_j maps $\mathbf{Z}_j$ in the latent space to the target subject j.

enable the learned representation of the latent space to preserve the reconstruction relation over the whole dataset of the subject. The reconstruction relation $\mathbf{R}_i$ is captured from the original space of the i-th subject ($i = 1, \ldots, N$). The decoder and discriminator form a GAN architecture that translates the information from one subject to another. This adversarial framework ensures that the learned translation respects the distribution of the target subject. In the semantic decoding stage, the classifier can be trained under the guidance of both the augmented data and the real training data, and it therefore takes the advantages of multiple-source information. The overall objective function of our data augmentation method is defined as follows:

$$\mathcal{L} = \mathcal{L}_{adv} + \lambda \mathcal{L}_{cyc} + \lambda \mathcal{L}_{recon1} + \gamma \mathcal{L}_{recon2} \tag{1}$$

$\mathcal{L}_{adv}$ is the adversarial loss, and $\mathcal{L}_{cyc}$ is the cyc–consistency loss. $\mathcal{L}_{recon1}$ is the local reconstruction loss, and $\mathcal{L}_{recon2}$ is the reconstruction relation loss. λ and γ are the hyper–parameters. The details of the method are described in the following. For the sake of readability, we list the frequently used symbols and their definitions in Table 1.

3.1. Deep multi-view adversarial learning

We employ multiple GAN architectures to generate new training samples: different subjects are considered as different views (domains). Unlike the original GAN whose input is random noise, the input of our GAN architectures is the information of source subjects. Here, $E_i : \mathcal{X}^{(i)} \to \mathcal{Z}, G_i : \mathcal{Z} \times \mathcal{Y} \to \mathcal{X}^{(i)}$, and $D_i : \mathcal{X}^{(i)} \times \mathcal{Y} \to \mathbb{R}$ for $i = 1, \ldots, N$.

For the j-th source subject, $\mathbf{e}_j = E_j\left(\mathbf{X}^{(j)}\right)$ is a random variable taking values from the latent space $\mathcal{Z}$. And we define $\mathbf{Z}_i$ to be a mixture of the random variables $\{\mathbf{e}_j : j \neq i\}$ with $\mathbb{P}(\mathbf{Z}_i = \mathbf{e}_j) = \alpha_{ij}\left(\alpha_{ij} = \frac{n_j}{\sum_{k \neq i} n_k}\right)$. Sampling from $\mathbf{Z}_i$ therefore corresponds to sampling uniformly from $\bigcup_{j \neq i} \mathbf{S}_j$.

The discriminator D_i attempts to distinguish real samples of $\mathbf{X}^{(i)}|\mathbf{Y}$ from fake samples $\widehat{\mathbf{X}}^{(i)}|\mathbf{Y}$. Here, we train the E_i, G_i, and D_i for all i simultaneously.

The adversarial loss of the i-th target subject is defined as follows:

$$\begin{aligned}
\mathcal{L}^i_{adv} &= \mathbb{E}\left[logD_i\left(\mathbf{X}^{(i)}|\mathbf{Y}\right)\right] + \mathbb{E}\left[log\left(1 - D_i\left(\widehat{\mathbf{X}}^{(i)}|\mathbf{Y}\right)\right)\right] \\
&= \mathbb{E}\left[logD_i\left(\mathbf{X}^{(i)}|\mathbf{Y}\right)\right] + \mathbb{E}[log(1 - D_i(G_i(\mathbf{Z}_i, \mathbf{Y})|\mathbf{Y}))] \\
&= \mathbb{E}\left[logD_i\left(\mathbf{X}^{(i)}|\mathbf{Y}\right)\right] \\
&+ \sum_{j \neq i} \alpha_{ij} \mathbb{E}\left[log\left(1 - D_i\left(G_i\left(E_j\left(\mathbf{X}^{(j)}\right), \mathbf{Y}\right)|\mathbf{Y}\right)\right)\right].
\end{aligned} \tag{2}$$

Table 1
Definition of frequently used symbols.

Symbol	Definition
N	Number of subjects
$\mathcal{X}^{(i)}$	i-th data space
$\mathbf{X}^{(i)}$	fMRI data of the i-th subject taken from $\mathcal{X}^{(i)}$
$\mathcal{Y}$	Label space of all subjects
$\mathbf{Y}$	Semantic label matrix of the subject taken from $\mathcal{Y}$
$\mathbf{S}_i$	i-th paired dataset of the traning set
$\bigcup_{j \neq i} \mathbf{S}_j$	Union of j-subjects ($j \neq i$)
C_i	i-th classifier
E_i	Encoder of the i-th subject
G_i	Decoder of the i-th subject
D_i	Discriminator of the i-th subject
$\mathcal{Z}$	Latent space
$\mathbf{R}_i$	Reconstruction (semantic) relation
$\mathbf{e}_i$	A random variable taking values from the latent space $\mathcal{Z}$
$\mathbf{Z}_i$	A mixture of the random variables $\{\mathbf{e}_j : j \neq i\}$
$\widehat{\mathbf{X}}^{(i)}$	Generated samples of i-th subject
θ^i_E	Parameter of the encoder E_i
θ^i_G	Parameter of the decoder G_i
θ^i_D	Parameter of the discriminator D_i
n_i	Size of the training set for the i-th subject
d_i	Dimension of fMRI data for the i-th subject
λ, β, γ	Hyper-parameters
$\|\cdot\|_2$	ℓ_2-norm operator
$\mathbb{E}[\cdot]$	Expectation operator

Note that $\mathcal{L}^i_{adv}$ depends on D_i, G_i and $\{E_j : j \neq i\}$.

To stabilize the model training process, we apply WGAN [28,29] to our method by replacing $\mathcal{L}^i_{adv}$ with the following loss:

$$\begin{aligned}\mathcal{L}^i_{wgan} &= \mathbb{E}\left[D_i\left(\mathbf{X}^{(i)}|\mathbf{Y}\right)\right] - \mathbb{E}[D_i(G_i(\mathbf{Z}_i,\mathbf{Y})|\mathbf{Y}))] \\ &+\beta\mathbb{E}\left[\left(\|\bigtriangledown_x D_i(\overline{\mathbf{X}}^{(i)}|\mathbf{Y})\|_2 - 1\right)^2\right] \\ &= \mathbb{E}\left[D_i\left(\mathbf{X}^{(i)}|\mathbf{Y}\right)\right] \\ &-\sum_{j\neq i}\alpha_{ij}\mathbb{E}\left[\left(D_i\left(G_i\left(E_j\left(\mathbf{X}^{(j)}\right),\mathbf{Y}\right)|\mathbf{Y}\right)\right)\right] \\ &+\beta\mathbb{E}\left[\left(\|\bigtriangledown_x D_i(\overline{\mathbf{X}}^{(i)}|\mathbf{Y})\|_2 - 1\right)^2\right]\end{aligned} \tag{3}$$

where $\overline{\mathbf{X}}^{(i)}$ is given by sampling uniformly along the straight lines between the pairs of samples of $\mathbf{X}^{(i)}$ and $\hat{\mathbf{X}}^{(i)}$, β is a hyper-parameter.

At the same time, to reduce the space of possible mapping functions, the learned mapping function should satisfy the constraint of cyc-consistency [23]: i.e. $E_i(G_i(z,y)) \approx z$.

The i-th cyc-consistency loss is defined as follows:

$$\begin{aligned}\mathcal{L}^i_{cyc} &= \mathbb{E}[\|\mathbf{Z}_i - E_i(G_i(\mathbf{Z}_i,\mathbf{Y})\|_2)] \\ &= \sum_{j\neq i}\alpha_{ij}\mathbb{E}\left[\|E_j\left(\mathbf{X}^{(j)}\right) - E_i\left(G_i\left(E_j\left(\mathbf{X}^{(j)}\right),\mathbf{Y}\right)\right)\|_2\right].\end{aligned} \tag{4}$$

Note that $\mathcal{L}^i_{cyc}$ depends on G_i and $\{E_j,\ j = 1,\ldots,N\}$. It ensures that the output after two mappings (from the latent space to target subject domain and from that domain back to the latent space) is similar to the initial input.

3.2. Constraints in latent space

Because of the mismatch of features between subjects, the data we directly used to enlarge the target subject is the latent space representation of source subjects rather than the original data. To utilize the information (e.g., semantic structure, neighbor information) of the source subjects, we must constrain the representation in the latent space. In the RadialGAN method, they first concatenate semantic labels to the source data and map new source data to the latent space with the encoder. Then, each data point actually performs as its supervisor (reconstruction loss) to learn a latent space representation. However, since the dimension of fMRI data is usually much higher than that of labels, it is difficult to keep the semantic structure information of latent space representation when fMRI data and labels are directly concatenate to the encoder. In addition, each data point as its supervisor is difficult to learn the structured globality (w.r.t. the entire dataset). The relation matrix of entire input dataset not only reflects the relationship between samples, but also reflects the semantic structure of the entire dataset (a sample is always represented by samples of the same category). Therefore, in addition to reconstruction loss, we also introduce the reconstruction relation loss for latent space representation learning.

The i-th reconstruction loss of original space and latent space is defined as follows:

$$\mathcal{L}^i_{recon1} = \mathbb{E}\left[\|\mathbf{X}^{(i)} - G_i\left(E_i\left(\mathbf{X}^{(i)}\right),\mathbf{Y}\right)\|_2\right] \tag{5}$$

Note that $\mathcal{L}^i_{recon1}$ depends on G_i and E_i. The loss $\mathcal{L}^i_{recon1}$ aims to minimize the loss between the input $\mathbf{X}^{(i)}$ and the reconstruction $G_i\left(E_i\left(\mathbf{X}^{(i)}\right)\right)$, and each data point acts as a supervisor to learn a compact representation of latent space. Obviously, $\mathcal{L}^i_{recon1}$ is designed to consider the local structure of each input sample in the representation learning [30,31].

Motivated by the papers [32–34], we also introduce the reconstruction relation loss $\mathcal{L}^i_{recon2}$ in the representation learning to preserve the reconstruction relation of whole input dataset:

$$\mathcal{L}^i_{recon2} = \mathbb{E}\left[\|E_i\left(\mathbf{X}^{(i)}\right) - \mathbf{R}_i E_i\left(\mathbf{X}^{(i)}\right)\|_2\right] \tag{6}$$

Note that $\mathcal{L}^i_{recon2}$ depends on E_i. The loss $\mathcal{L}^i_{recon2}$ is designed based on the so-called manifold assumption [35] which states that the reconstruction relation is invariant to different representation spaces. In this paper, we mainly consider the reconstruction relation in $\mathbf{R}_i (i = 1 \ldots N)$ which is obtained by solving the following problem:

$$\begin{aligned}&\min_{\mathbf{R}_i}\sum_{k=1}^{n_i}\|\mathbf{X}^{(i)}_k - (\mathbf{R}_i)_k\mathbf{X}^{(i)}\|_2^2 + \alpha\|\mathbf{R}_i\|_1 \\ &s.t.(\mathbf{R}_i)_{kk} = 0\end{aligned} \tag{7}$$

where $\|\cdot\|_1$ and $\|\cdot\|_2$ denote ℓ_1-norm and ℓ_2-norm, respectively. $\mathbf{R}_i$ is the sparse reconstruction relation over the whole dataset (i-th subject) and $(\mathbf{R}_i)_k$ is the k-th row vector of matrix $\mathbf{R}_i$. $(\mathbf{R}_i)_{kk}$ denotes the k-th element of the vector $(\mathbf{R}_i)_k$, and the constraint is used to avoid degenerated solutions [30,31].

3.3. Training details and prediction

In the above multi-view deep generative model, E_i, G_i and D_i are implemented via fully-connected networks and we denote their parameters by θ_E^i, θ_G^i and θ_D^i, respectively.

The overall objective function of our method is as the following minimax problem:

$$\min_{\mathbf{G},\mathbf{E}}\max_{\mathbf{D}}\left(\sum_{i=1}^{N}\mathcal{L}_{wgan}^{i}(D_i,G_i,\ \{E_j:j\neq i\})+\lambda\sum_{i=1}^{N}\mathcal{L}_{cyc}^{i}(E_i,G_i)+\lambda\sum_{i=1}^{N}\mathcal{L}_{recon1}^{i}(E_i)+\gamma\sum_{i=1}^{N}\mathcal{L}_{recon2}^{i}(G_i,\ \{E_j:j\neq i\})\right) \tag{8}$$

where $\mathbf{G}=(G_1,\ldots,G_N),\mathbf{E}=(E_1,\ldots,E_N)$ and $\mathbf{D}=(D_1,\ldots,D_N)$. Here, λ,γ are hyper-parameters.

The pseudo-code for our method is shown in Algorithm 1. As shown in Algorithm 1, we solve this minimax problem iteratively. First, we fixed $\mathbf{G}$ and $\mathbf{E}$ to train $\mathbf{D}$ with a mini-batch k_D and then fixed $\mathbf{D}$ to train $\mathbf{G}$ and $\mathbf{E}$ with a mini-batch k_G.

Once the translation functions $E_1,\ldots,E_N$ and $G_1,\ldots,G_N$ were trained, we can simultaneously obtain N augmented dataset $\mathbf{S}_1',\ldots,\mathbf{S}_N'$ for N subjects, where $\mathbf{S}_i'=\mathbf{S}_i\cup\bigcup_{j\neq i}G_i(E_j(\mathbf{S}_j))$. Then, the classifiers $C_1,\ldots,C_N$ are learned on these augmented datasets.

Algorithm 1. Pseudo-code for our method

Input: Paired datasets $\left(\mathbf{X}^{(1)},\mathbf{Y}\right),\ldots,\left(\mathbf{X}^{(N)},\mathbf{Y}\right)$.
(training set of each subject)
//Intitialization:
$\theta_G^1,\ldots,\theta_G^N,\ \theta_E^1,\ldots,\theta_E^N,\ \theta_D^1,\ldots,\theta_D^N$
// Compute the reconstruction relation:
Obtain the relation matrix $\mathbf{R}_i\in\mathbb{R}^{n_i\times n_i}$ of $\mathbf{X}^{(i)}$ by solving Eq. (7) (i=1,..., N).
// Optimizations:
while not converge **do**
(1) Update D with fixed G, E
for i = 1,..., N **do**
Draw k_D samples from $\mathbf{S}_i$, $\left\{\left(\mathbf{x}_k^{(i)},\mathbf{y}_k\right)\right\}_{k=1}^{k_D}$
Draw k_D samples from $\bigcup_{j\neq i}\mathbf{S}_j$, $\left\{\left(\mathbf{x}_k^{(j_k)},\mathbf{y}_k\right)\right\}_{k=1}^{k_D}$
for k = 1,..., N **do**
$\widehat{\mathbf{x}}_k^{(i)}\leftarrow G_i\left(E_{j_k}\left(\mathbf{x}_k^{(j_k)}\right),\mathbf{y}_k\right)$
end for
Update θ_D^i using stochastic gradient decent (SGD)
$\nabla_{\theta_D^i}-\left(\sum_{k=1}^{k_D}logD_i\left(\mathbf{x}_k^{(i)}|\mathbf{y}_k\right)\right.$
$\left.+\sum_{k=1}^{k_D}log\left(1-D_i\left(\widehat{\mathbf{x}}_k^{(i)}|\mathbf{y}_k\right)\right)\right)$
end for
(2) Update G, E with fixed D
for i = 1,..., N **do**
Draw k_G samples from $\mathbf{S}_i$, $\left\{\left(\mathbf{x}_k^{(i)},\mathbf{y}_k\right)\right\}_{k=1}^{k_G}$
Draw k_G samples from $\bigcup_{j\neq i}\mathbf{S}_j$, $\left\{\left(\mathbf{x}_k^{(j_k)},\mathbf{y}_k\right)\right\}_{k=1}^{k_G}$
Draw $k_G\times n_i$ submatrix $(\mathbf{R}_i)_{k_G}$ from the $\mathbf{R}_i$
end for
Update $\theta_{\mathbf{G},\mathbf{E}}=\left(\theta_G^1,\ldots,\theta_G^N,\theta_E^1,\ldots,\theta_E^N\right)$ using SGD
$\nabla_{\theta_{G,E}}\sum_{i=1}^{N}\sum_{k=1}^{k_G}log\left(1-D_i\left(G_i\left(E_{j_k}\left(\mathbf{x}_k^{(j_k)}\right),\mathbf{y}_k\right)|\mathbf{y}_k\right)\right)$
$+\lambda\sum_{i=1}^{N}\sum_{k=1}^{k_G}\|\mathbf{x}_k^{(i)}-G_i\left(E_i\left(\mathbf{x}_k^{(i)}\right),\mathbf{y}_k\right)\|_2$

(continued on next page)

Algorthim 1. (*continued*)

$+\lambda\sum_{i=1}^{N}\sum_{k=1}^{k_G}\|E_{j_k}\left(\mathbf{x}_k^{(j_k)}\right)-E_i\left(G_i\left(E_{j_k}\left(\mathbf{x}_k^{j_k}\right),\mathbf{y}_k\right)\right)\|_2$
$+\gamma\sum_{i=1}^{N}\sum_{k=1}^{k_G}\|E_i\left(\mathbf{x}_k^{(i)}\right)-(\mathbf{R}_i)_{k_G}(k,:)E_i\left(\mathbf{X}^{(i)}\right)\|_2$
end while
Output: The augmented data for the i-th target subject $\cup_{j\neq i}G_i(E_j(\mathbf{S}_j))(i=1,\ldots,N,\ j=1,\ldots N)$.

3.4. Discussion

For the latent representation learning, if γ in (8) is set as 0, the proposed method without the loss $\mathcal{L}_{recon2}^{i}$ reduces to the Auto-Encoder (AE). In this sense, our method augments AE by considering the valuable relations among different samples (i.e., reconstruction relation). The introduction of reconstruction relation matrix makes our method has good scalability. As we know, label missing is a very common and tricky problem in the field of semantic classification. In this case, RadialGAN [25] may discard the label missing data in order to expand the dataset smoothly. Because in RadialGAN, in addition to encoder, decoder and discriminator also need label as input. For our method, no label information is needed to compute the relation matrix. Therefore, in the case of labels missing: 1) this will not affect the latent space features learning of our method; 2) we can use the reconstruction relation matrix and the existing labeled data to predict pseudo labels for the label missing data, so it can be applied to the process of data augmentation.

One of the challenges of the proposed multi-subject method is how to deal with the situation that the number of the subjects in the fMRI dataset is very large. In view of this situation, for our method, we can divide a large number of subjects into several subgroups in accordance with their similarity (correlation). Then data augmentation is performed on subjects of each subgroups. The number of augmented data for each target subject (the number of subgroups) can be adjusted according to the experimental effects. In addition, unlike other data augmentation methods [23,26] which need many manual steps in this case, our method performs data augmentation for each target subject simultaneously.

Another challenge is that there is a poor correlation between one subject and other subjects. Our paper also focuses on solving the problem of individual differences. In view of this situation, we can try our method first. In addition, we can use some selection methods [36,37] to select the subjects and discard the subject that is less relevant to most other subjects before data augmentation.

4. Experiments

In this section, we report the performance of the proposed method compared with other methods.

4.1. Experimental setup

4.1.1. Datasets

We evaluated the proposed method on three public fMRI datasets containing different numbers of subjects. We briefly introduce them in the following.

- *Handwritten Characters*[1] [9]: The data acquisition experiments had 3 subjects viewing grey-scale handwritten character images from six semantic categories. In this paper, we study the 6-class task on Handwritten Characters fMRI dataset with 3 subjects.
- *Haxby*[2] [10]: This is a block-design fMRI dataset from a study on the face and object representation in the human ventral temporal cortex. The data acquisition experiments had 6 subjects viewing grey-scale images stimuli from 8 semantic categories, with 12 runs per subject. The data for the 9th run of the 5th subject was corrupted and therefore should not be used for analyses. In this paper, we study the challenging 8-class task on the Haxby fMRI dataset with 5 subjects.
- *CMU2008*[3] [11]: The CMU Science 2008 is a challenging multi-class fMRI dataset. The data acquisition experiments had 9 subjects viewing 60 different word-picture stimuli from 12 semantic categories, with 5 exemplars per category and 6 runs per subject. In this paper, we study the challenging 8-class task on CMU2008 fMRI dataset with 9 subjects. The fMRI data of CMU2008 is from the whole brain and has a high dimension, in which a lot of voxels may not respond to the visual stimuli. Therefore, we use the mutual information [12] to remove the unrelated voxels before the decoding experiments.

[1] Data are available at http://sciencesanne.com/research/.
[2] Data are available at http://www.pymvpa.org/datadb/haxby2001.html.
[3] Data are available at http://www.cs.cmu.edu/afs/cs/project/theo-73/www/science2008/data.html.

4.1.2. Compared methods

The following methods are used as compared methods.

- *Baseline:* The baseline method refers to without data augmentation for each subject.
- *Simple-Combination (S-Combination)* [25]: In the S-Combination method, we combine all subjects of the fMRI dataset by defining the feature space as the union of all feature spaces and setting missing dimension to zero.
- *CGAN* [26]: The CGAN provides an algorithm for learning to generate samples conditioned on a discrete label. The input of CGAN is simple random noise, which limits its performance.
- *Cycle-GAN* [23]: The Cycle-GAN translates information from the source domain to the target domain. It requires that the source domain and the target domain have the same feature dimension. Hence in the training process, we set the missing dimension of the subject to zero.
- *StarGAN* [24]: The StarGAN provides a framework for multi-domain translations by using only a single model. However, this method also requires multiple domains to have the same feature dimension. In the training process, we set the missing dimension of the subject to zero.
- *RadialGAN*[25]: The RadialGAN is a multi-source data augmentation method. The comparison results between our method and *RadialGAN* method are shown in the ablation study section.

4.1.3. Parameter setting

In this paper, we have three hyper-parameters λ, β and γ. We conduct the cross-validation on the training sets to choose the best λ, β among {1, 2, 5, 10} and also select the γ from {0.001, 0.01, 0.5, 1}. We also select the dimension of latent space from $\left\{max_i(d_i), max_i\left(\frac{d_i}{2}\right), max_i\left(\frac{d_i}{4}\right)\right\}(i = 1, \ldots, N)$. For Cycle-GAN, we set the hyper-parameters α, β and γ to $1, 10$ and 5, respectively. For starGAN, we set the hyper-parameters λ_1 and λ_2 to 1 and 10, respectively.

In addition, we consider multiple fully-connected networks as the type of the encoders, decoders and discriminators. The fully-connected networks are adopt due to following reasons: 1) The size of fMRI data is usually small, which is easy to cause over-fitting. Many brain encoding and decoding papers always built regression models for fMRI data [38–41]. The regression model can be regarded as a single-layer fully-connected network without activation function; 2) We experimentally found that even with such a simple neural network, our method could remarkably outperform many popular methods; 3) It is relatively easy to tune parameters for the fully-connected networks comparing with other neural networks. The depth of all fully-connected networks is set to 3. For fair comparison, we use the same network architecture in all compared methods. For all netwroks, the number of hidden nodes in the first, second and third layers are $m, m/2$, and m (where m is the dimension of the input), respectively. The tanh function is used as the activation function for each layer except for the output layer where we use the sigmoid function. The batch size of our method and StarGAN is set to 64, and the batch size of other compared methods is set to 32. The Adam optimizer with a learning rate of 0.0001 is utilized for training all models.

4.1.4. Metrics

As SVM (Support Vector Machine) has especial advantages in solving high-dimensional pattern recognition problems [42], most semantic decoding models usually use SVM as the classifier [4,13,19]. In this paper, the prediction performance of SVM is used to evaluate the effectiveness of our data augmentation algorithm. The evaluation criteria of prediction are Accuracy and F1-Score.

4.2. Ablation study

We conduct ablation study with detailed results in Table 2 to evaluate the effectiveness of each component in our algorithm. **Non-rr** means that there is no reconstruction relation in the latent space (RadialGAN). **Non-f** refers to there is no feature mismatch component in this method. For **Non-f** method, we set the missing dimension of the fMRI data to zero before inputting it to the decoder. **Non-d** denotes that there is no distribution mismatch component in the method, that is, there are no multiple discriminators in the training process.

Table 2 shows the average classification accuracy of subjects (cross validation is performed for each subject) on three datasets. For each dataset, the results are averaged across all subjects. The best performance is shown in boldface. By comparing Non-f with Non-d, it can be seen that the method without feature mismatch component can drastically reduce the performance. For the Handwritten dataset, the role of the distribution mismatch component is more important than the feature mismatch component. By comparing Non-rr with Ours, it can be seen that sparse reconstruction (semantic) relation can further improve the experimental results. The above results show that each component of the method is of great significance to improve the experimental performance.

4.3. Comparison with state-of-the-art

In this section, we compare our method with some state-of-the-art approaches on the Handwritten Characters, Haxby, and CMU2008 datasets.

Table 2
Prediction performance comparison in Ablation Study on three datasets.

Method	Handwritten	Haxby	CMU2008
Non-rr	0.502 ± 0.047	0.580 ± 0.080	0.411 ± 0.230
Non-f	0.460 ± 0.043	0.518 ± 0.047	0.341 ± 0.166
Non-d	0.442 ± 0.070	0.585 ± 0.081	0.428 ± 0.225
Ours	**0.516** ± 0.049	**0.609** ± 0.090	**0.435** ± 0.186

Because our method can simultaneously enlarge the data of each target subject, we report the classification accuracy of each subject on three datasets. For each subject in the Handwritten Characters dataset, the number of training samples and test samples is 300 and 60, respectively. For each subject in the CMU2008 dataset, the number of training samples and test samples is 200 and 40, respectively. For each subject in the Haxby dataset, we use 288 samples (4 runs) as training data and 576 samples (8 runs) as test data. It is important to note that here, we did not use any test data of source (target) subjects in the process of data augmentation. The test data of each subject is only used in the test stage of classification. To avoid the over-fitting problem, we select different training sets and test sets in the data augmentation process for each subject. We perform 6-fold cross validation on each subject in the Handwritten Characters and CMU2008 datasets (6 suns), and the average classification accuracy of each subject is shown in Table 3 and Table 5. For the Haxby dataset, we randomly divide the data of the 12 runs into three parts (each part contains data of 4 runs). Each takes one part as training set, and the other as test set for each subject. The average classification accuracy of each subject on the Haxby dataset is reported in Table 4. We also conducted experiments on the Haxby dataset with different convolutional neural network architectures, the results are shown in Table 9 of appendix section.

From Table 3 and Table 4, it can be seen that our method achieves the best classification performance. On the Handwritten Characters dataset, the performance of our method is at least $3.70\%, 2.80\%$ and 4.50% higher than the second-best results regarding these three subjects. On the Haxby dataset, the performance of our method is at least $6.40\%, 5.90\%, 3.30\%, 6.50\%$ and 4.30% higher than the second-best results regarding these five subjects. From Table 5, we can see our method achieves the best classification performance on most subjects. On the CMU2008 dataset, the performance of our method is at least $4.20\%, 1.70\%, 3.80\%, 2.60\%, 1.30\%$, and 1.70% higher than the second-best results regarding these six subjects.

The average results of all subjects on each dataset are shown in Fig. 5. From Fig. 5, we find that our method consistently outperforms other methods. For these three datasets, the performance of our method is at least $3.70\%, 5.30\%$, and 1.60% higher than the second-best results. Furthermore, the performance of Cycle-GAN method is greatly limited which may be partly due to the fact that no semantic information is used in the domain translation process and each pair of mappings only utilizes two corresponding subjects information. The Simple-combination method has low performance because it does not consider the distribution mismatch problem and feature mismatch problem between subjects. The mapping functions of StarGAN are a single network, which limits its performance. Although the CGAN only learns the information of the target subject in the data augmentation process, it uses semantic information in the training process and the target subject does not need to set the missing dimension to zero. Therefore, its performance is moderate.

We also present the average classification accuracy of each category on three datasets before and after data augmentation. The results are shown in Fig. 6. As can be seen from Fig. 6, if the average accuracy of a category does not improve after data augmentation, the point corresponding to this category is on the diagonal of this figure. For the Handwritten dataset, the classification accuracy of two-thirds of categories is improved after data augmentation. For the Haxby dataset, the classification accuracy of all categories is improved after data augmentation. For the CMU2008 dataset, the classification accuracy of three-quarters of categories is improved after data augmentation. These demonstrate the superiority of the proposed method again.

4.4. Augmented data evaluation

To evaluate the quality of augmented data generated from source subjects (training data), we use the augmented data as training set and the target subject training data as test set. The average classification results of subjects on the three datasets are shown in Table 6. **WODA** means that without data augmentation (domain translation), the source subjects (training) data is directly used as training set and the target subject training data is used as test set. **WDA** means that the source subjects (training) data is used as training set after data augmentation, the target subject training data is still directly used as test set. From Table 6, we can see the source subjects data after domain translation capture the information of the target subject training data well.

At the same time, we also evaluate the quality of the augmented data on the test data of target subject. The average classification results of subjects on three datasets are shown in Table 7. In the **WODA** stage, there are two types of training set: real training data of target subject and other subjects (training) data. In the **WDA** stage, training sets are real training data of target subject and augmented data (generated from other subjects (training data)), respectively. The test set of all stages is the real test data of target subject. By comparing the results of the **WODA** stage and the **WDA** stage, it can be noticed that the quality of the augmented data is superior. By comparing the results of real training data and augmented data, we can see the

Table 3
Prediction performance comparison with other methods on the Handwritten Characters dataset.

Subject	Accuracy					
index	Baseline	S-Combination [25]	CGAN [26]	Cycle-GAN [23]	StarGAN [24]	Ours
1	0.491 ± 0.042	0.441 ± 0.059	0.466 ± 0.055	0.371 ± 0.060	0.466 ± 0.040	**0.528** ± 0.045
2	0.433 ± 0.030	0.383 ± 0.069	0.411 ± 0.057	0.338 ± 0.076	0.402 ± 0.038	**0.461** ± 0.035
3	0.513 ± 0.034	0.474 ± 0.060	0.455 ± 0.061	0.455 ± 0.041	0.494 ± 0.048	**0.558** ± 0.062
Average	0.479 ± 0.041	0.432 ± 0.046	0.444 ± 0.029	0.388 ± 0.060	0.454 ± 0.047	**0.516** ± 0.049

Table 4
Prediction performance comparison with other methods on the Haxby dataset.

Subject	Accuracy					
index	Baseline	S-Combination [25]	CGAN [26]	Cycle-GAN [23]	StarGAN [24]	Ours
1	0.656 ± 0.016	0.548 ± 0.035	0.672 ± 0.009	0.432 ± 0.017	0.612 ± 0.035	**0.736** ± 0.015
2	0.550 ± 0.036	0.479 ± 0.053	0.553 ± 0.035	0.375 ± 0.108	0.522 ± 0.030	**0.612** ± 0.028
3	0.530 ± 0.010	0.492 ± 0.002	0.530 ± 0.087	0.365 ± 0.026	0.523 ± 0.010	**0.563** ± 0.011
4	0.415 ± 0.009	0.404 ± 0.025	0.430 ± 0.013	0.300 ± 0.024	0.403 ± 0.026	**0.495** ± 0.033
5	0.600 ± 0.035	0.521 ± 0.037	0.593 ± 0.028	0.377 ± 0.075	0.563 ± 0.059	**0.643** ± 0.047
Average	0.550 ± 0.089	0.489 ± 0.054	0.556 ± 0.089	0.369 ± 0.047	0.525 ± 0.077	**0.609** ± **0.090**

Table 5
Prediction performance comparison with other methods on the CMU2008 dataset.

Subject	Accuracy					
index	Baseline	S-Combination [25]	CGAN [26]	Cycle-GAN [23]	StarGAN [24]	Ours
1	0.725 ± 0.094	0.608 ± 0.043	0.712 ± 0.075	0.529 ± 0.043	0.695 ± 0.076	**0.767** ± 0.087
2	0.512 ± 0.130	0.362 ± 0.086	0.525 ± 0.136	0.316 ± 0.043	0.525 ± 0.144	**0.542** ± 0.099
3	0.454 ± 0.070	0.404 ± 0.106	0.458 ± 0.098	0.329 ± 0.087	0.441 ± 0.090	**0.496** ± 0.090
4	0.608 ± 0.040	0.462 ± 0.081	0.612 ± 0.044	0.370 ± 0.082	0.575 ± 0.063	**0.638** ± 0.046
5	0.241 ± 0.054	**0.275** ± **0.052**	0.262 ± 0.072	0.195 ± 0.048	0.241 ± 0.086	0.254 ± 0.069
6	0.216 ± 0.034	0.208 ± 0.046	0.225 ± 0.047	0.220 ± 0.024	0.204 ± 0.040	**0.238** ± 0.026
7	0.341 ± 0.070	0.308 ± 0.060	0.358 ± 0.070	0.283 ± 0.071	0.320 ± 0.079	**0.375** ± 0.052
8	0.270 ± 0.087	0.266 ± 0.090	**0.291** ± **0.093**	0.237 ± 0.046	0.241 ± 0.086	0.288 ± 0.098
9	0.291 ± 0.049	0.245 ± 0.045	**0.329** ± **0.048**	0.225 ± 0.035	0.279 ± 0.029	0.321 ± 0.065
Average	0.406 ± 0.178	0.349 ± 0.126	0.419 ± 0.168	0.300 ± 0.103	0.391 ± 0.174	**0.435**± 0.186

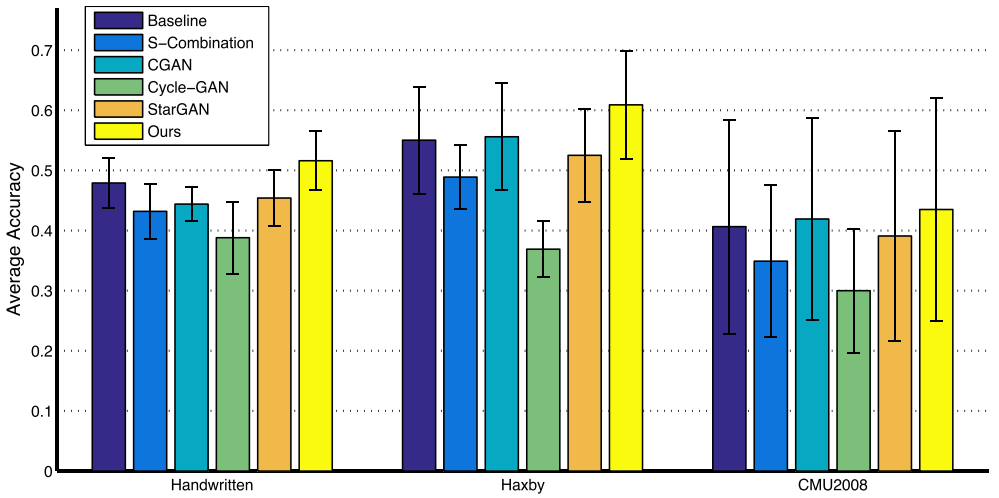

Fig. 5. Average accuracy comparisons between our method and the other methods on the three datasets. For each dataset, the results are averaged across all subjects. Error bars represent standard error.

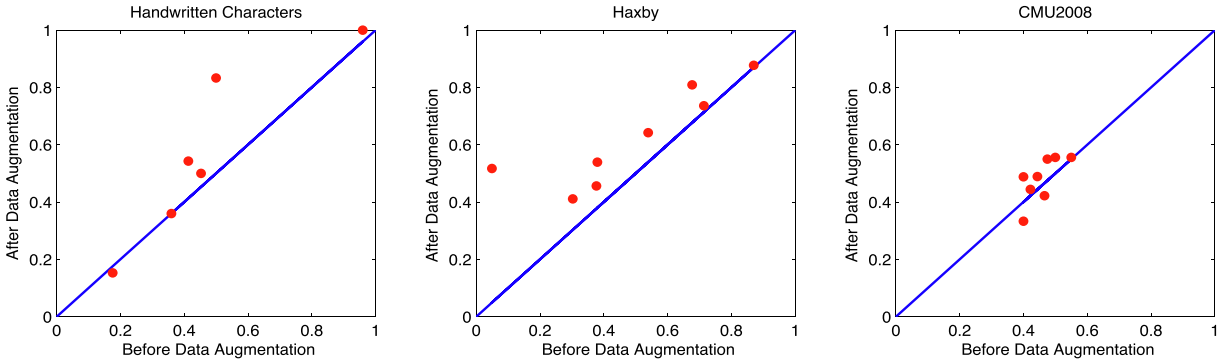

Fig. 6. The average accuracy of each category on three datasets. The horizontal axis represents the average classification accuracy of each category before data augmentation. The vertical axis represents the average classification accuracy of each category after data augmentation.

Table 6
Augmented Data Evaluation on the training sets of three datasets.

Dataset	Accuracy	
	WODA	WDA
Handwritten	0.197 ± 0.014	**0.864** ± 0.013
Haxby	0.251 ± 0.027	**0.950** ± 0.030
CMU2008	0.186 ± 0.041	**0.722** ± 0.155

Table 7
Augmented Data Evaluation on the test sets of three datasets.

Dataset	Accuracy			
	WODA		WDA	
	Real training data	Other subjects data	Real training data	Augmented data
Handwritten	0.479 ± 0.041	0.188 ± 0.025	0.479 ± 0.041	**0.505** ± 0.083
Haxby	0.550 ± 0.080	0.233 ± 0.024	0.550 ± 0.080	**0.582** ± 0.081
CMU2008	0.406 ± 0.178	0.216 ± 0.0363	0.406 ± 0.178	**0.422** ± 0.153

results of the latter are better than the former on the three datasets. This shows that there is no over-fitting in data augmentation process. It also demonstrates that the augmented data translated from other subjects contains new information that is different from the real training data of target subject.

Fig. 7 and Fig. 8 provide a visualized evidence for the superior performance of our data augmentation method (different colors represent different categories).

4.5. Comparison on multiple augmented samples

In this section, we compare our method with these benchmarks when we add different number of augmented samples to target subject. Table 8 shows the average results of all subjects on the Haxby dataset. Accuracy and F1-score are used as evaluation criteria. The F1-score calculated from each category reflects the average predicted results for all categories. The number of the augmented samples is denoted with the multiples of the number of a source subject samples. The number of training samples for a source subject is 288. $1 * 288$ represents that the number of augmented samples for each target subject is 288. The average accuracy and F1-score of the baseline method (no augmented samples $0 * 288$) are 0.545 and 0.549, respectively.

From Table 8, we can see that for CGAN and our method, as the number of augmented samples increases, the performance gain increases. This is due to the large number of samples that SVM can utilize. The Cycle-GAN and StarGAN require that the source subject and the target subject have the same feature dimension and the missing dimension of the subject must be set to zero. It therefore requires two methods to learn a large sparse matrix to solve the problem. Thus, even though the number of augmented data increases, the improvement of prediction performance is limited due to the high dimensional data. CGAN is a method that only uses target subject information in the data augmentation process. As can be seen from Fig. 9, with the number of augmented samples increases, our method achieves higher accuracy than the CGAN.

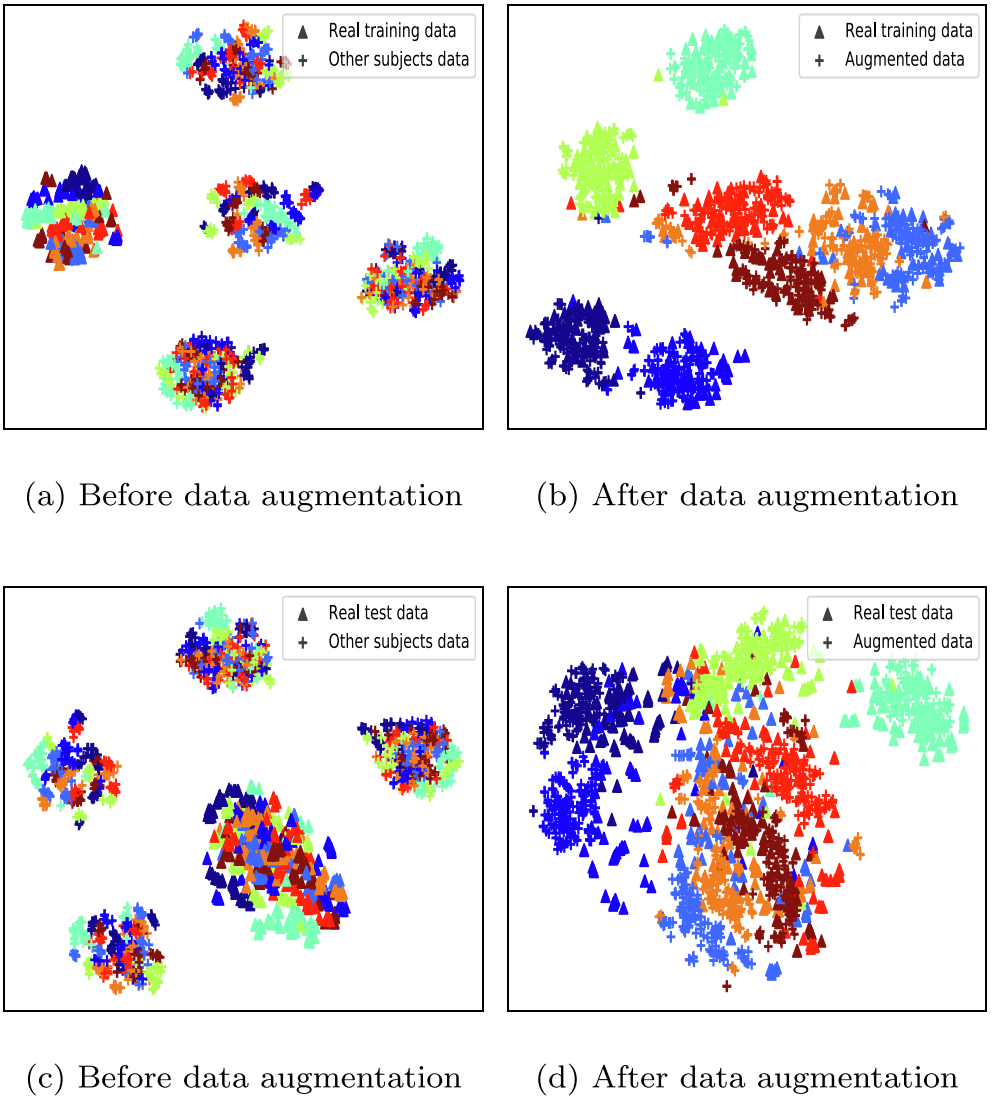

Fig. 7. Visualization on the Haxby dataset. (a) (c) Real training (test) data of a target subject and (training) data of other subjects before data augmentation. (b) (d) Real training (test) data of a target subject and augmented data generated from other subjects (training data).

On the other hand, the CGAN shows a tendency to worsen SVM model on most subjects, especially when a larger number of augmented data are added to target subject. The reason for this phenomenon is that the samples generated by the CGAN can only contain the information already contained in the target subject. Furthermore, due to the limited size of the target subject, when a large number of samples are generated, the resulting augmented samples may be much more noisy than the original target subject. In contrast, our method can utilize the information contained in multiple subjects to obtain the less noisy augmented samples.

4.6. Comparison on multiple training samples

In this section, we compare our method with these benchmarks when we vary the number of training samples of each subject. The average accuracy of all subjects is shown in Fig. 10. As can be seen from Fig. 10, as the number of training samples increases, the performance increases due to the larger number of samples that these methods can utilize. For the different ratio of training data and test data, our method is always superior to other methods. Moreover, when the number of training samples is small, our method shows excellent performance compared with other methods. This is of great significance for the field of the brain decoding. Because fMRI data acquisition is very expensive and time-consuming, it is usually difficult to learn an accurate decoding model with a small number of data.

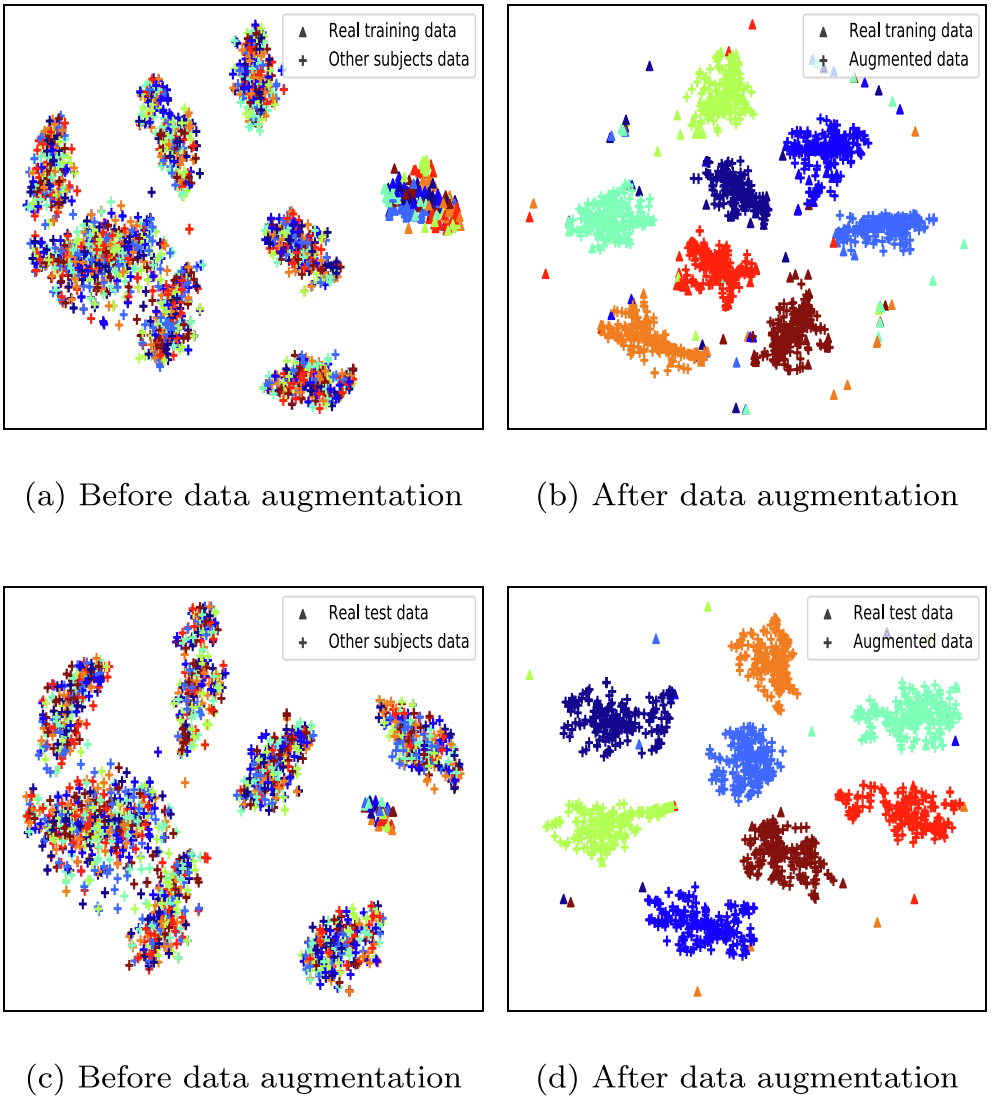

Fig. 8. Visualization on the CMU2008 dataset. (a) (c) Real training (test) data of a target subject and (training) data of other subjects before data augmentation. (b) (d) Real training (test) data of a target subject and augmented data generated from other subjects (training data).

Table 8
Prediction performance comparison with other methods on the Haxby dateset when using multiple augmented samples.

Method	1*288		3*288	
	Accuracy	F1-Score	Accuracy	F1-Score
CGAN [26]	0.549 ± 0.083	0.550 ± 0.078	0.550 ± 0.078	0.555 ± 0.074
Cycle-GAN [23]	0.550 ± 0.080	0.233 ± 0.024	0.550 ± 0.080	0.582 ± 0.081
StarGAN [24]	0.516 ± 0.077	0.516 ± 0.073	0.511 ± 0.082	0.508 ± 0.079
Ours	**0.592** ± 0.070	**0.598** ± 0.065	**0.607** ± 0.083	**0.611** ± 0.079

4.7. Reconstruction relation matrix analysis

Reconstruction relation matrix R_i in this paper is regarded as self-expression on the whole data of i-subject, the i-th row in the R_i denotes the representation coefficients of the i-th samples $X_i^{(i)}$ in the data matrix $X^{(i)}$. To better understand the reconstruction relation matrix, we visualize the matrix R. The visualization results are shown in Fig. 11. The visualization matrix in Fig. 11(a) (b) is obtained by sorting the learned reconstruction relationship matrix according to the corresponding categories

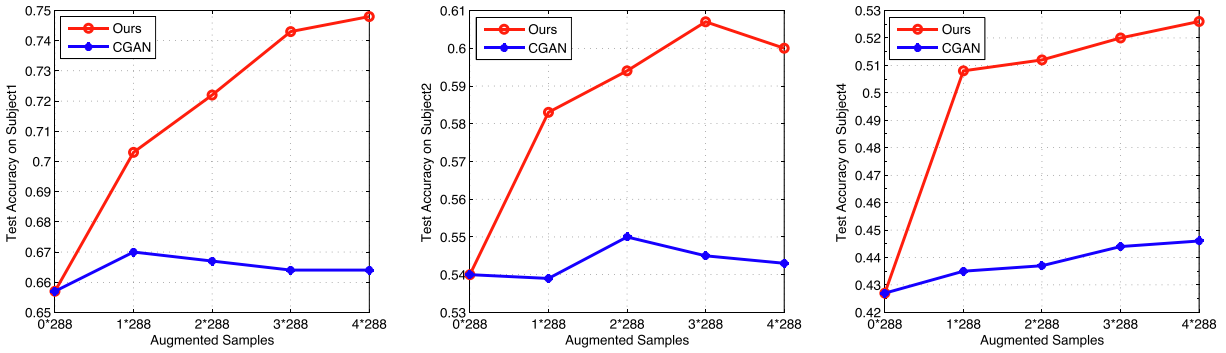

Fig. 9. Comparison between our method and CGAN method when using different number of augmented samples.

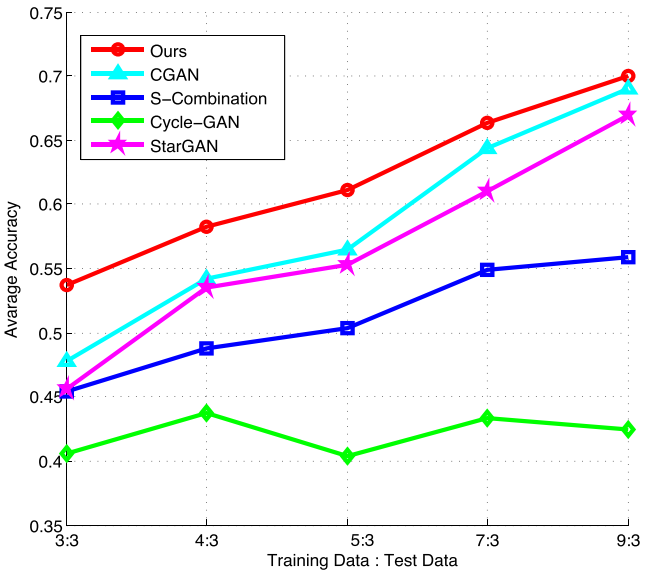

Fig. 10. Evaluation with different ratio of training data and test data. The average accuracy of baseline method is {0.478, 0.548, 0.563, 0.643, 0.685}.

of samples. From the Fig. 11, we observe that the elements with large values are mostly located on the diagonal of the matrix, and they show a block-diagonal structure (the number of blocks is equal to the number of categories in this dataset). This indicates that the semantic relation structure has been successfully learned from the whole dataset with a small number of data. We also visualize the latent space representation constrained by the reconstruction relation matrix, and the representation learned by RadialGAN and the original data. The visualization results are shown in Fig. 12. As can be seen from Fig. 12, compared with the RadialGAN method, the representation learned by our method is more consistent with the original data and more interpretable. The latent representations learned by our method are clustered, which keeps the semantic structure and neighbor information of the original data well. The latent representations learned by RadialGAN method are scattered, which loses a lot of original information and lacks interpretability.

4.8. Sensitivity analysis

In this section, we conduct experiments to analyze the sensitivity of hyper-parameters and the sensitivity of the dimensions of latent space. In our method, three hyper-parameters including λ, β, and γ need to be set properly. For the sake of studying the sensitivity of our method with respect to different values of these hyper-parameters, we plot the average classification results on the Haxby dataset by tuning the values of λ, β, and γ. The results are shown in Fig. 13. For different λ and γ, the performance of our method fluctuates slightly. The results verify the importance of the diverse regularization terms. From another perspective, our method also performs stably in a fixed range of β. For the sake of studying the sensitivity of our method with respect to various dimensions of latent space, we plot the average classification results on the Haxby dataset by tuning the dimensions of the latent space. The results are shown in Fig. 14. From Fig. 14, we can observe that when the dimensions of latent space changes from max{dim}/8 to max{dim}, the average accuracy of our method consistently outper-

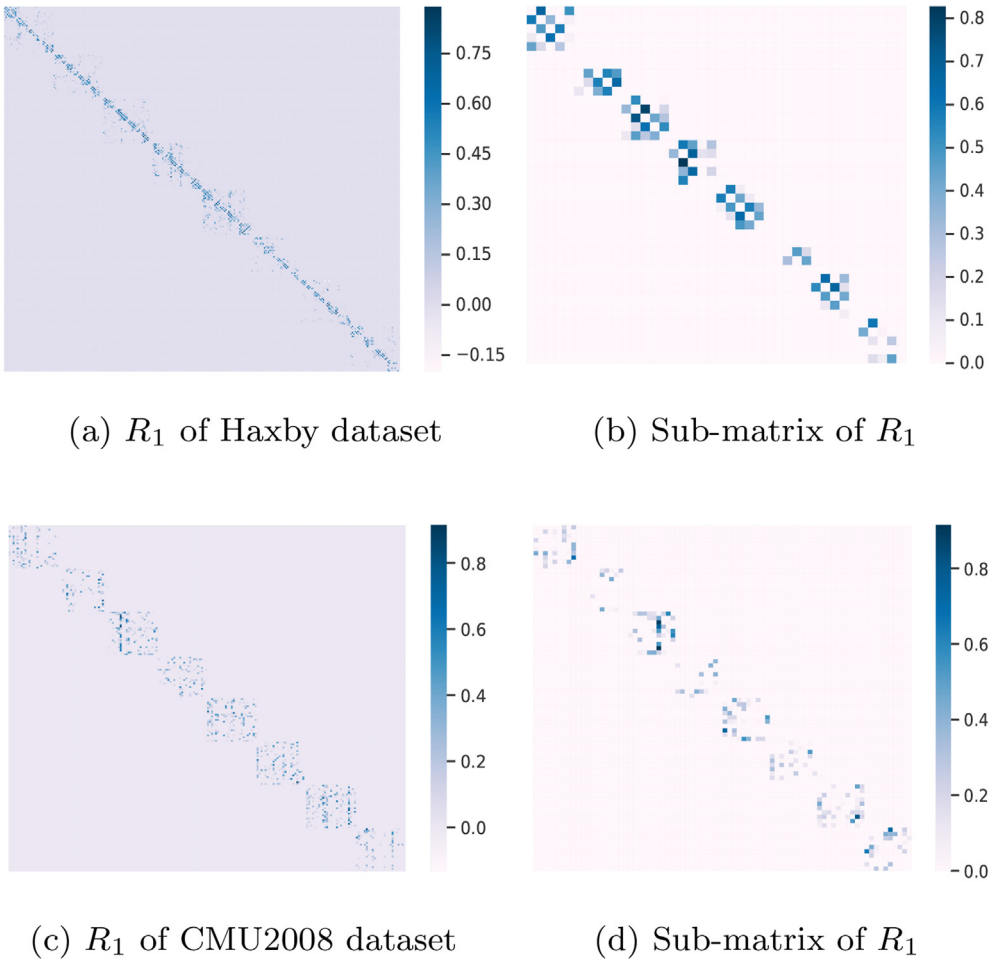

Fig. 11. Visualization on the reconstruction relation matrix.

forms the baseline method. The average accuracy of our method is changed from 0.602 to 0.616. This also demonstrates that our method performs stably in a fixed range of the dimensions of latent space.

5. Conclusion and discussion

In this paper, we introduce a multi-subject fMRI data augmentation method to improve the performance of single subject semantic decoding. This method maps the fMRI data of all subjects to the latent space to solve the problem of feature mismatch. In addition, by adding the MSE loss and reconstruction (semantic) relation information, the representation of the latent space preserves the local and global structure of the input. It uses multiple GAN architectures to ensure that the distribution of generated samples matches the distribution of the target subject. Then, real samples and augmented samples form a new training set for training semantic classifiers. Experiments on three fMRI datasets demonstrate that the proposed method outperforms state-of-the-art approaches.

This fMRI data augmentation framework is of great significance in the field of brain semantic decoding: (1) this method can well deal with the situation that a small number of labeled samples are available; (2) the enlarged dataset enables us to train more complex models and effectively reduce the impact of over-fitting, thus improving the performance of the predictive model. In the future, we will apply our data augmentation framework (as a data preprocessing tool) to some typical brain decoding models, such as other fMRI classification models and fMRI visual image reconstruction models.

CRediT authorship contribution statement

Dan Li: Methodology, Software, Validation, Writing - original draft. **Changde Du:** Methodology, Writing - review & editing. **Shengpei Wang:** Writing - review & editing. **Haibao Wang:** Writing - review & editing. **Huiguang He:** Writing - review & editing, Funding acquisition, Supervision.

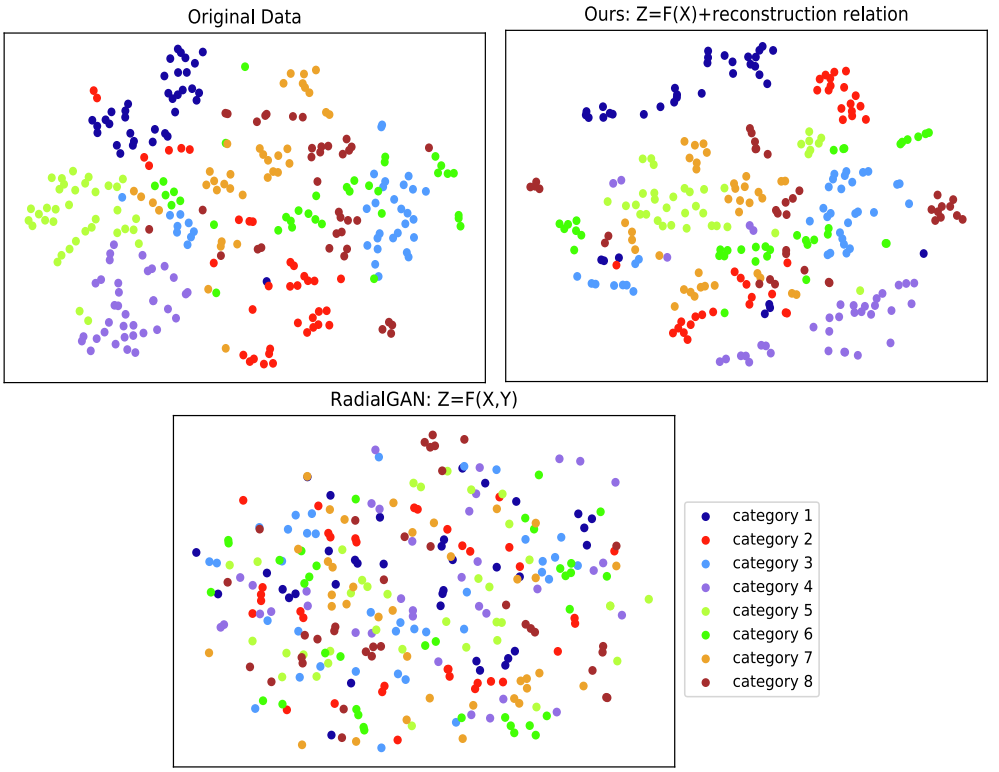

Fig. 12. Visualization on the Haxby dataset (subject 1). From left to right and from top to bottom are the original data, the latent representation learned by our method and the latent representation learned by RadialGAN method, respectively. These two kinds of latent representations are obtained under the same hyper-parameters. Different colors represent different categories.

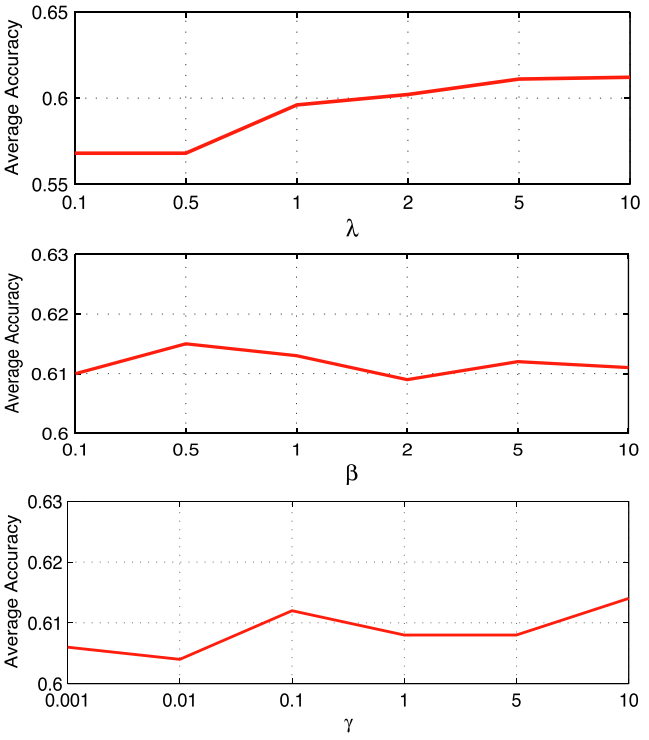

Fig. 13. A sensitivity analysis of hyper-parameters. The average accuracy of baseline method is 0.566.

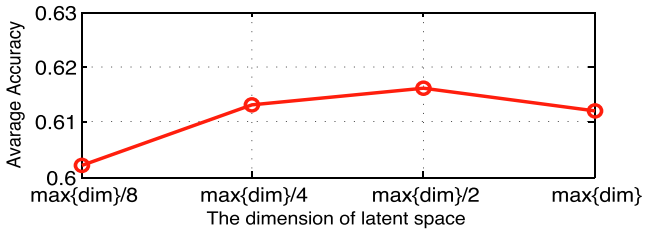

Fig. 14. A sensitivity analysis of the dimensions of latent space. max{*dim*} denotes the maximum dimension value of all subjects in the Haxby dataset. The average accuracy of baseline method is 0.566.

Declaration of Competing 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.

Acknowledgment

This work was supported in part by the National Natural Science Foundationof China (61976209, 62020106015), CAS International Collaboration Key Project (173211KYSB20190024), and Strategic Priority Research Program of CAS (XDB32040000).

Appendix A. Appendix

Table 9

Table 9
Prediction performance comparison with different network architectures on the Haxby dataset. For these experiments, we have to reshape the fMRI data $X \in R^{N\times M}$ (preprocessed data, N denotes the size of samples, M denotes the dimension of samples) into $X \in R^{N\times 1\times M\times 1}$. The **first** network architecture is as follows: (1, 1, 1, 1) → (1, M/2, 1, 1) → fully-connected layer. Different from the **first** network, the **second** network architecture is (1, M/2, 1, 1) → (1, M/4, 1, 1) → fully-connected layer. Four element array is the size of convolution kernel.

Subject	Accuracy	
index	convolutional layers + fully-connected layer (**First**)	convolutional layers + fully-connected layer (**Second**)
1	0.674 ± 0.012	0.702 ± 0.024
2	0.559 ± 0.022	0.580 ± 0.016
3	0.545 ± 0.032	0.556 ± 0.034
4	0.440 ± 0.024	0.483 ± 0.035
5	0.642 ± 0.025	0.614 ± 0.059
Average	0.572 ± 0.091	0.587 ± 0.080

Fig. 15

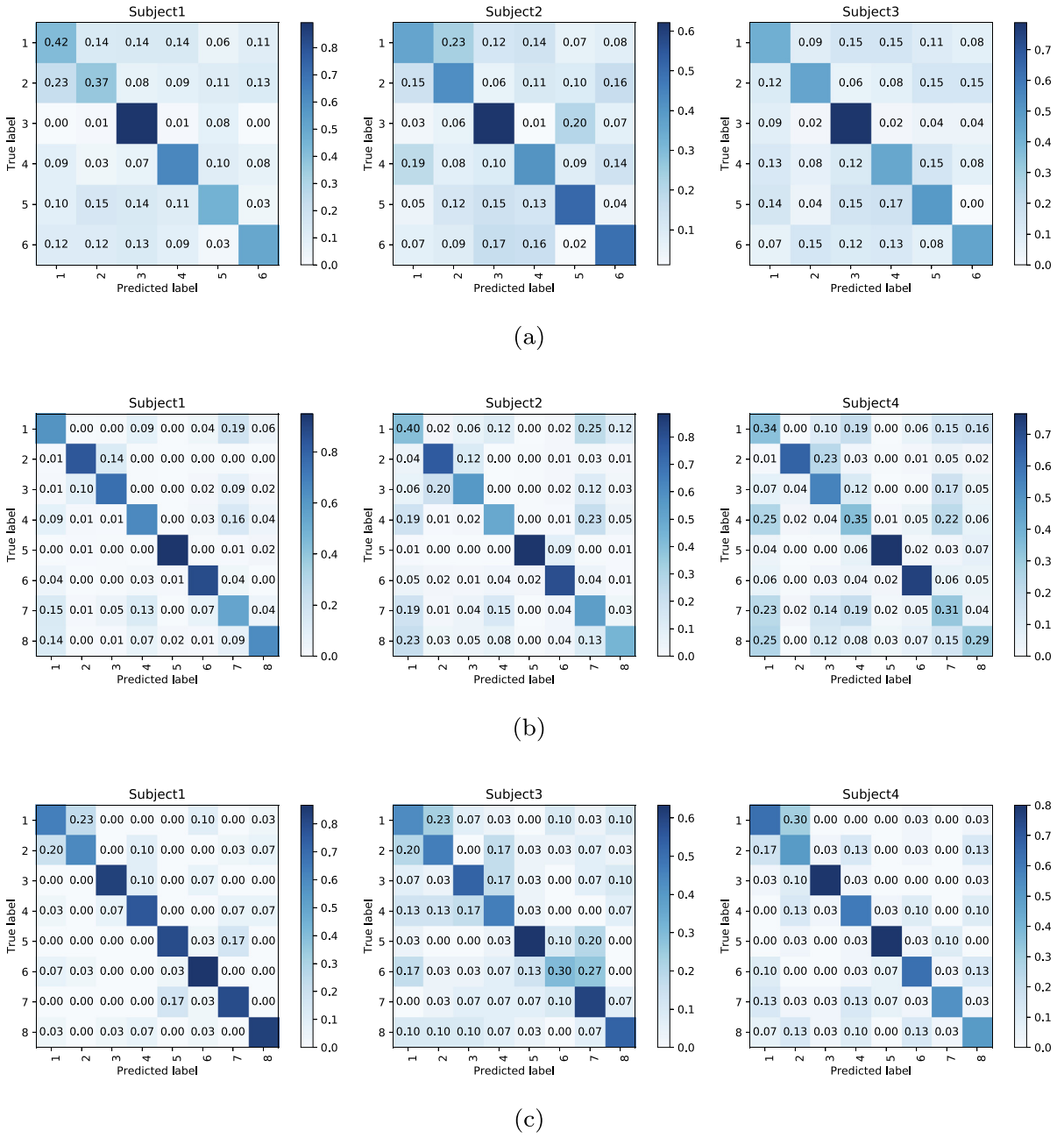

Fig. 15. Experimental results of confusion matrices on the three datasets. (a) the confusion matrices on the Handwritten dataset; (b) the partial confusion matrices on Haxby dataset; (c) the partial confusion matrices on CMU2008 dataset.

References

[1] Y. Gao, B. Zhou, Y. Zhou, L. Shi, Y. Tao, J. Zhang, Transfer learning-based behavioural task decoding from brain activity, in: The International Conference on Healthcare Science and Engineering, Springer, 2018, pp. 71–81.
[2] A.G. Huth, T. Lee, S. Nishimoto, N.Y. Bilenko, A.T. Vu, J.L. Gallant, Decoding the semantic content of natural movies from human brain activity, Front. Syst. Neurosci. 10 (2016) 81.
[3] K. Vodrahalli, P.-H. Chen, Y. Liang, C. Baldassano, J. Chen, E. Yong, C. Honey, U. Hasson, P. Ramadge, K.A. Norman, et al, Mapping between fMRI responses to movies and their natural language annotations, Neuroimage 180 (2018) 223–231.
[4] L.I. Kuncheva, J.J. Rodríguez, C.O. Plumpton, D.E. Linden, S.J. Johnston, Random subspace ensembles for fMRI classification, IEEE Trans. Med. Imag. 29 (2) (2010) 531–542.

[5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in Neural Information Processing Systems, 2014, pp. 2672–2680..
[6] D.P. Kingma, M. Welling, Auto-encoding variational bayes, arXiv preprint arXiv:1312.6114..
[7] P. Zhuang, A.G. Schwing, O. Koyejo, fMRI data augmentation via synthesis, in: 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019), IEEE, 2019, pp. 1783–1787.
[8] H. Zhang, P.-H. Chen, P. Ramadge, Transfer learning on fMRI datasets, International Conference on Artificial Intelligence and Statistics (2018) 595–603.
[9] S. Schoenmakers, M. Barth, T. Heskes, M. van Gerven, Linear reconstruction of perceived images from human brain activity, NeuroImage 83 (2013) 951–961.
[10] J.V. Haxby, M.I. Gobbini, M.L. Furey, A. Ishai, J.L. Schouten, P. Pietrini, Distributed and overlapping representations of faces and objects in ventral temporal cortex, Science 293 (5539) (2001) 2425–2430.
[11] T.M. Mitchell, S.V. Shinkareva, A. Carlson, K.-M. Chang, V.L. Malave, R.A. Mason, M.A. Just, Predicting human brain activity associated with the meanings of nouns, Science 320 (5880) (2008) 1191–1195.
[12] C.-A. Chou, K. Kampa, S.H. Mehta, R.F. Tungaraza, W.A. Chaovalitwongse, T.J. Grabowski, Voxel selection framework in multi-voxel pattern analysis of fMRI data for prediction of neural response to visual stimuli, IEEE Trans. Med. Imag. 33 (4) (2014) 925–934.
[13] X. Xu, Q. Wu, S. Wang, J. Liu, J. Sun, A. Cichocki, Whole brain fMRI pattern analysis based on tensor neural network, IEEE Access 6 (2018) 29297–29305.
[14] A. Hoyos-Idrobo, G. Varoquaux, Y. Schwartz, B. Thirion, Frem–scalable and stable decoding with fast regularized ensemble of models, NeuroImage 180 (2018) 160–172.
[15] K.N. Kay, T. Naselaris, R.J. Prenger, J.L. Gallant, Identifying natural images from human brain activity, Nature 452 (7185) (2008) 352.
[16] T. Horikawa, Y. Kamitani, Generic decoding of seen and imagined objects using hierarchical visual features, Nat. Commun. 8 (2017) 15037.
[17] C. Du, C. Du, L. Huang, H. He, Reconstructing perceived images from human brain activities with bayesian deep multiview learning, IEEE Trans. Neural Networks Learn. Syst..
[18] A.S. Cowen, M.M. Chun, B.A. Kuhl, Neural portraits of perception: reconstructing face images from evoked brain activity, Neuroimage 94 (2014) 12–22.
[19] X. Song, L. Meng, Q. Shi, H. Lu, Learning tensor-based features for whole-brain fMRI classification, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2015, pp. 613–620.
[20] Z. Wen, T. Yu, Z. Yu, Y. Li, Grouped sparse bayesian learning for voxel selection in multivoxel pattern analysis of fMRI data, NeuroImage 184 (2019) 417–430.
[21] M.A. Hearst, S.T. Dumais, E. Osuna, J. Platt, B. Scholkopf, Support vector machines, IEEE Intell. Syst. Appl. 13 (4) (1998) 18–28.
[22] Y. Luo, B.-L. Lu, EEG data augmentation for emotion recognition using a conditional wasserstein GAN, in: 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), IEEE, 2018, pp. 2535–2538..
[23] J.-Y. Zhu, T. Park, P. Isola, A.A. Efros, Unpaired image-to-image translation using cycle-consistent adversarial networks, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2223–2232.
[24] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, J. Choo, StarGAN, Unified generative adversarial networks for multi-domain image-to-image translation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8789–8797.
[25] J. Yoon, J. Jordon, M. Schaar, RadialGAN: Leveraging multiple datasets to improve target-specific predictive models using generative adversarial networks, in: International Conference on Machine Learning, 2018, pp. 5685–5693..
[26] M. Mirza, S. Osindero, Conditional generative adversarial nets, arXiv preprint arXiv:1411.1784..
[27] Z. Zhong, L. Zheng, Z. Zheng, S. Li, Y. Yang, Camstyle: a novel data augmentation method for person re-identification, IEEE Trans. Image Process. 28 (3) (2018) 1176–1190.
[28] M. Arjovsky, S. Chintala, L. Bottou, Wasserstein generative adversarial networks, in: International Conference on Machine Learning, 2017, pp. 214–223.
[29] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, A.C. Courville, Improved training of wasserstein GANs, in: Advances in neural information processing systems, 2017, pp. 5767–5777..
[30] X. Peng, J. Feng, S. Xiao, W.-Y. Yau, J.T. Zhou, S. Yang, Structured autoencoders for subspace clustering, IEEE Trans. Image Process. 27 (10) (2018) 5076–5086.
[31] X. Peng, S. Xiao, J. Feng, W.-Y. Yau, Z. Yi, Deep subspace clustering with sparsity prior, IJCAI (2016) 1925–1931.
[32] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, Y. Ma, Robust recovery of subspace structures by low-rank representation, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2012) 171–184.
[33] X. Peng, C. Lu, Z. Yi, H. Tang, Connections between nuclear-norm and frobenius-norm-based representations, IEEE Trans. Neural Networks Learn. Syst. 29 (1) (2016) 218–224.
[34] E. Elhamifar, R. Vidal, Sparse subspace clustering: algorithm, theory, and applications, IEEE Trans. Pattern Anal. Mach. Intell. 35 (11) (2013) 2765–2781.
[35] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326.
[36] E.W. Xiang, S.J. Pan, W. Pan, J. Su, Q. Yang, Source-selection-free transfer learning, in: Twenty-Second International Joint Conference on Artificial Intelligence, 2011..
[37] Z. Lu, Y. Zhu, S.J. Pan, E.W. Xiang, Y. Wang, Q. Yang, Source free transfer learning for text classification, in: Twenty-Eighth AAAI Conference on Artificial Intelligence, 2014..
[38] S. Wang, J. Zhang, H. Wang, N. Lin, C. Zong, Fine-grained neural decoding with distributed word representations, Inf. Sci. 507 (2020) 256–272.
[39] K. Seeliger, U. Güçlü, L. Ambrogioni, Y. Güçlütürk, M.A. van Gerven, Generative adversarial networks for reconstructing natural images from brain activity, NeuroImage 181 (2018) 775–785.
[40] C. Du, C. Du, L. Huang, H. He, Conditional generative neural decoding with structured CNN feature prediction, AAAI (2020) 2629–2636.
[41] H. Wang, L. Huang, C. Du, H. He, Learning what and where: An interpretable neural encoding model, in: 2019 International Joint Conference on Neural Networks (IJCNN), IEEE, 2019, pp. 1–8..
[42] V. Jakkula, Tutorial on support vector machine (svm), School of EECS, Washington State University, 37..