Neurocomputing 408 (2020) 91–99

Contents lists available at ScienceDirect

Neurocomputing

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

Information-preserving feature filter for short-term EEG signals

Yue Yao[a], Josephine Plested[a], Tom Gedeon[a,*]

Research School of Computer Science, The Australian National University, Canberra, Australia

ARTICLE INFO

Article history:
Received 3 February 2019
Revised 30 October 2019
Accepted 2 November 2019
Available online 10 March 2020

Keywords:
Deep learning
EEG
Generative adversarial networks
Physiological signals
Image translation

ABSTRACT

The brain-computer interface (BCI) has become one of the most important biomedical research fields and has many useful applications. An important component of BCI, electroencephalography (EEG) is in general sensitive to noise and rich in all kinds of information from our brain. In this paper, we study the feature fusion problem in electroencephalography (EEG). We introduce (1) a discriminative feature extractor which can classify multi-labels from short-term EEG signals, and (2) a new strategy to filter out unwanted features from EEG signals based on our feature extractor. Filtering out signals relating to one property of the EEG signal while retaining another is similar to the way we can listen to just one voice during a party, which is known as the cocktail party problem in the machine learning area. Built based on the success of short-term EEG discriminative model, the feature filter is an end-to-end framework which is trained to map EEG signals with unwanted features directly to EEG signals without those features. Our experimental results on an alcoholism dataset show that our novel model can filter out over 90% of alcoholism information on average from EEG signals, with an average of only 4.2% useful feature accuracy lost, showing effectiveness for our proposed task.

© 2020 Elsevier B.V. All rights reserved.

1. Introduction

The human brain is a complex system. Research towards thought patterns and expanding the way people exchange information with the outside world has never stopped for cognitive neuroscience and neurorehabilitation. With the rapid development of cognitive science, neuroscience, computer science, and signal processing technology in recent years, the brain-computer interface (BCI) provides human beings with other ways to communicate with the world, and also allows us to get a better understanding of the physical mechanisms of human thought [1,2].

As an essential part of brain-computer interfaces (BCIs), electroencephalograph (EEG) signals, also known as brainwaves, have found a variety of exciting and useful applications for users and have become increasingly important in various areas. Gathered from the scalp, the EEG is a signal containing information about the electrical activity of the brain. Electrodes placed on the scalp are used to detect electrical information from the brain under the scalp, bone and other tissues. Since it is an overall measurement of human brain electrical activity, it contains a wealth of information. This is the reason why EEG can be applied to diverse areas like personal recognition [3], disease identification [4], sleep stage classification [5], visual image generation using brainwaves [6], and brain typing [7].

From the viewpoint of data analysis, automatic EEG analysis is challenging due to the inherent feature of bio-signals. One source of ambiguity is the fused nature of features, which is common for most bio-signal feature learning tasks. The fusion here means that any one experimental trial of signals contains both wanted features and unwanted features for the given tasks. Also, due to the lack of macroscopic knowledge of the mechanism of EEG activity, this fused feature problem in EEG is more serious than for many other physiological signals. Besides that, brain wave analysis is challenging in the following aspects:

- **Low signal to noise ratio**: For EEG signals, being full of information also means full of noise and interference, making it very hard to extract reliable features [8–11].
- **Data format varies**: Depending on the collection device, EEG signals have a different format [12]. Hence it becomes difficult to construct standard algorithms to extract features from EEG.
- **Limited training data**: Constructing a hand-labeled training corpus for fine-grained EEG analysis is labor-intensive. Since EEG data collections are often domain dependent, it is not practical to always collect new training data for new domains. Furthermore, due to the feature fusion problem, the privacy issue is also one important reason why current datasets do not involve large numbers of subjects, thereby making it practically impossible to build a huge dataset like ImageNet [13].

* Corresponding author.
E-mail addresses: yue.yao@anu.edu.au (Y. Yao), jo.plested@anu.edu.au (J. Plested), tom@cs.anu.edu.au (T. Gedeon).

https://doi.org/10.1016/j.neucom.2019.11.106
0925-2312/© 2020 Elsevier B.V. All rights reserved.

- **Large individual difference**: EEG signals have large individual differences, making it hard to learn robust features across subjects [14–16].

In this work, we first designed an end-to-end framework for short-term EEG signals classification. To address the above difficulties, deep learning approaches are utilized in this paper to achieve both learning and visualization. Autoencoder-based techniques are used for feature learning and dimensionality reduction for short-term EEG signals. In this paper this method is referred to as Image-wise autoencoders. The Image-wise autoencoders are designed based on Fast Fourier Transform (FFT) and convolutional neural networks. Using FFT, we can obtain the three EEG frequency bands, then we use these frequency bands to achieve an RGB-color visualization (an image) [10]. Then, a CNN based autoencoder is designed to extract features from these color images with both classification loss and reconstruction loss. Under this design, our models successfully overcomes the difficulties of consistent handling of EEG data.

Furthermore, in a real-world situation, customers not only require accuracy for the brain-computer interface but also require a competent level of privacy and information safety [17]. For example, if we would like to use EEG for a personal recognition task for a bank, the only information we would like to upload is personal identity-related information. But unfortunately, EEG is a fused feature data with a messy, vibrant symphony of personal information, including one's individuality, learning capacity and emotion information. That is, all brain activity related feature will be uploaded and available for legal or illegal uses. For the bank example, since there currently does not exist a suitable information filtering algorithm, both the bank and potential future hackers will also be able to get our other information like disease information, emotion information and so on. Current research has tried to specify several standards for operating on EEG data to protect users' privacy but that has not solved the problem fundamentally [18–20].

To address the above issue, for the first time, we propose a feature filter for short-term EEG signals. In practice, we do not use the idea of subtracting features to filter out properties as such properties are not well-defined. Instead, we choose to generate a new EEG trial without the unwanted features but maintaining the desired features of the original EEG trial signal. Thus, a generative adversarial network (GAN) based technique is utilized to create such an EEG signal. In this paper, we also introduce a feature filter, which is as an extension of our short-EEG discriminative model. As mentioned earlier, the feature filter of EEG is more like a style transformation. So we are inspired by the idea of Image-to-Image translation [21] introduced in the computer vision area. This approach is designed to map one image distribution to another image distribution in order to achieve a style transformation. In this section, such a translation mechanism is used for feature filtering.

Contributions are summarized as listed:

- We propose the EEG feature filter and feature extractor to support it based on our conference papers [22–24]. For the first time, we consider the situations that competent privacy information protection is generally required for customers. We transfer time series EEG signals to EEG images, thereby reducing the feature filter problem to an image translation problem.
- We conduct detailed experiments to validate the performance of the proposed network and contribution of each component. Experiment on UCI EEG datasets shows our competent level of information preservation and privacy protection.

2. Background and related work

Convolutional neural networks (CNNs) are feature extraction networks proposed by LeCun [25], based on the structure of the mammalian visual cortex, thus providing structural information about the data via the network topology. The difference between convolution neural networks and the traditional neural networks is the convolution layer. We consider the convolution layers as feature extractors. Then, the fully connected layer serves as a classifier trying to find decision boundaries between each class. From another point of view, the role of the fully connected layer is similar to the kernel method, warping the high-level feature space to make each class approximately linearly separable.

Much CNN based research has been applied to EEG. Depending on the type of the kernel, CNN based work can be divided into normal CNN as well as frequency-based CNN. Normal CNN takes the raw EEG as the input while frequency-based CNN extracts frequency features from raw EEG. Examples of normal CNN approaches include Deep4net [26] and EEGNet [14]. The SyncNet [27] is an example of a frequency-based CNN for EEG. An interesting commonality is that one-dimensional convolutions are often applied among convolution procedures [28,29].

An **Autoencoder** is a kind of compression algorithm, or dimension reduction algorithm, which has similar properties to Principal Components Analysis (PCA). As compared with PCA, the autoencoder has no linear constraints. The autoencoder structure has been widely used for image compression, for example [30], which inspired us to try an autoencoder based learning algorithm. An autoencoder can be divided into two parts, an encoder and a decoder. The number of nodes in the hidden layer is generally less than the nodes in the input layer and the output layer. That is, the original input is compressed to a smaller feature vector. In Eq. (1) below, ϕ and ψ stand for encoder and decoder, respectively, and L means squared loss. The objective of the autoencoder is to minimize the difference between the input and the generated output. A CNN based autoencoder [31] uses convolution operations as the encoder and deconvolution operations for the decoder, making it better for operating on image data.

$$\phi, \psi = argmin_{\phi,\psi} L(X, (\phi \circ \psi)X), \tag{1}$$

Prior to our work, a number of autoencoder related methods have been applied to EEG signals. Stober [32] used convolutional autoencoders with custom constraints to learn features and improve generalization across subjects and trials. It achieved commendable results but it uses CNN directly on the time domain features from EEG signals but not frequency domain features like our methods. But Stober's work inspired us that it could be a general conclusion that the autoencoder based structure can increase the cross-subject accuracy, forming our basic inspiration to try autoencoder based structures.

The most similar work to our classification model is by Tabar and Halici [33]. They used EEG motor imagery signals and a combined CNN and fully connected stacked autoencoders (SAE) to find discriminative features. They used Short-time Fourier transform (STFT) to build an EEG motor imagery (MI) which is unlike our 3D electrode location mapping as used in our work (described in Section 3). Also, their autoencoder design is quite different from ours since they used a CNN followed by an 8-layer SAE. Nevertheless, they have demonstrated that autoencoders can help to learn robust features from EEG signals.

Generative adversarial networks (GANs) are systems of two neural networks contesting with each other in a minimax game framework [34]. The GAN approach has achieved great success in the image generation area [35–37]. GANs include two main parts, namely a generator and a discriminator. The generator is mainly used to learn the distribution of the real image and produce images in order to fool the discriminator, while the discriminator needs to accept real images while rejecting generated images. Throughout this process, the generator strives to make the generated image more realistic, while the discriminator strives to

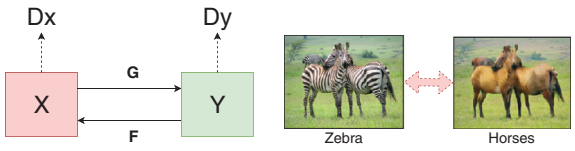


Fig. 1. CycleGAN structure and image translation example [37]. A CycleGAN learns a bidirectional mapping mapping from two domians (e.g. zebra images and horses images). It is composed of two generators (*G* and *F*) and two discriminators (Dx and Dy).

identify the real image. The key part of GAN is the adversarial loss. For the image generation task, the adversarial loss is very powerful for images in one domain transformed to the other domain since this domain cannot be discriminated by simple rules, but deep leaning models have achieved some success.

Image-to-Image translation is a kind of system that can learn the mapping between an input image distribution and an output image distribution using two separate image domains [21]. Shown in Fig. 1, given a source distribution *X*, we are aiming to use a generative model *G* to map our source distribution *X* to the target distribution *Y*. An example is shown in Fig. 1, though it is not perfect, the translation system has successfully transformed the most important features between zebra and horses like the hide color. In this translation system, we do not explicitly tell the neural network to change some features. Instead, we have the prior knowledge of two separate image distributions. As a result, it is possible for us to extract the stylistic differences between two image distributions and then directly translate them from one domain to the other domain.

The **cycle-consistent adversarial network** (CycleGAN) is a well-known image-to-image translation method for unpaired images [37]. It overcomes the difficulty of getting paired images, and forms an autoencoder-like structure to achieve image translation. In Fig. 1, *G* is such a generator that generates a domain *Y* image from domain *X*, while *F* is the generator that generates a domain *X* image from the domain *Y*. Dx and Dy are two discriminators that are used to determine whether the coming image really belongs to domain *X* or domain *Y*, respectively. The training procedure can be separated into two symmetric parts. One is $X \rightarrow G(X) \rightarrow F(G(X))$. In this autoencoder-like loop, the training loss comes from two parts, the first is the discriminator loss which comes from Dy to judge whether $G(X)$ is really from domain *Y* and the second is the reconstruction loss to judge whether $F(G(X))$ is the same as *X* or not. The other loop $Y \rightarrow F(Y) \rightarrow G(F(Y))$ is the same in principle.

All of these GAN methods are based on two hypotheses. One is that it is possible to build a strong classifier that can discriminate such features, and the second is the availability of a reliable generator that can filter out original features and rebuild target features. For the first hypothesis, if we cannot train a strong classifier in normal labeled training, it will be almost impossible for us get a strong discriminator in training, because adversarial training itself is not well designed to help train the discriminator. That is not an issue for many GAN based methods which have achieved great success in the CV area, since the most popular current datasets like MNIST [38] and CIFAR-10 [39] have already achieved more than 90% accuracy using different CNNs to serve as accurate discriminators. In contrast to CV, since the NLP area does not have a universally recognized text classification method for grammar checking, current GAN methods for NLP, like Seqgan [40] and its improved version Leakgan [41] do not have a strong discriminator to guide the generator. For our second hypothesis, we have to have a strong generator which can rebuild features. But building a strong generator is closely related to the given type of data. For the image translation area, convolution and deconvolution-based methods are often used.

Image-wise autoencoders as mentioned in the last section are the solution we use to meet the two hypotheses of building a GAN for EEG. An image-wise autoencoder is used to extract discriminative and robust features from EEG images. During the autoencoder training, it can reduce reconstruction loss to a very low level for the test set, making it possible to become a generator for the GAN structure. Furthermore, when we connect the features to a fully connected layer to work as a classifier, it achieves convincing results with more than 90% accuracy in the within-subject test discriminator.

3. Methodology

3.1. UCI EEG dataset

The dataset we use is from UCI, the EEG dataset from Neurodynamics Laboratory at the State University of New York. It has in total 122 participants with 45 control subjects and 77 subjects diagnosed with alcoholism [27,43]. Each subject has 120 separate trials. If a subject is labeled with alcoholism, all 120 trials belonging to that subject will be labeled as alcoholism. The stimuli used are several pictures from the Snodgrass and Vanderwart picture set. It is a sort time EEG where one trial of EEG signal is of one second length. Each trial is sampled at 256Hz using 64 electrodes. For the classification task, models are first evaluated using data within subjects, which is randomly split as 7:1:2 for training, validation and testing for one person [27]. The classification objective is to discover whether the subject has been diagnosed with alcoholism or not. Also, we note that this is not a balanced dataset. It is a two-task classification but alcoholism trials account for more than 70% of the data. For training the feature filter, we also use within-subject testing but just split the source distribution (alcoholism) within subjects, which is randomly split as 7:1:2 for training, validation and testing for each alcoholism subject. The target distribution is the whole data from control subjects. The usual challenges of handling EEG make it more difficult to apply deep learning methods compared with computer vision data or natural language processing data. The UCI EEG dataset is not an exception. First, a label is applied to one trial in this dataset. But as one trial contains 64 channels and 256 time series data, making it a 64×256 large matrix. In other words, a single EEG trial has 64×256 attributes, difficult for a neural network to find meaningful features if treated as 16,384 independent inputs. Second, EEG is a kind of time-series data but it lacks recognizable patterns in single time slices (1/sampling rate) compared with natural language processing, since each word in NLP often has a specific meaning. Third, as previous work has shown, if we consider raw EEG signals and directly use a convolution neural network for raw EEG data, there is always a serious problem to determine the size of the kernels to use at each stage [28,33]. That is, because the original features could be distributed with different time differences in a single trial depending on the scenario (different classification task for example), making it hard for convolution kernels to extract features.

3.2. Image-wise autoencoders

The image-wise autoencoders take images as input while using a CNN to extract features. The whole procedure is shown in Fig. 2, and below is some further explanation.

(**A) EEG to Image:**

The method is derived from Bashivan's work [42]. As shown in Fig. 3, it is a method that combines the time-series information and spatial channel locations information over the scalp in a trial of EEG signals. An FFT is performed on the time series to estimate the power spectrum of the signal for each trial (64×256). From

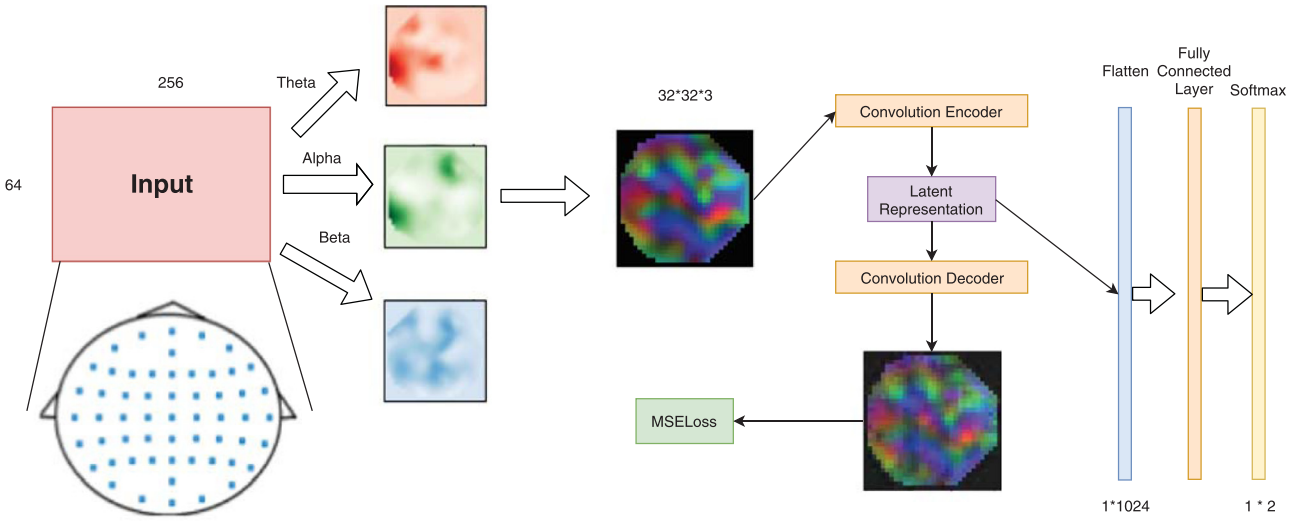


Fig. 2. Structure of Image-wise autoencoder. We extract three frequency band to constract EEG images [42]. The classification model takes EEG images as input and perform a image reconstruction and classification. The joint loss (cross entropy and MSEloss) is used for model training.

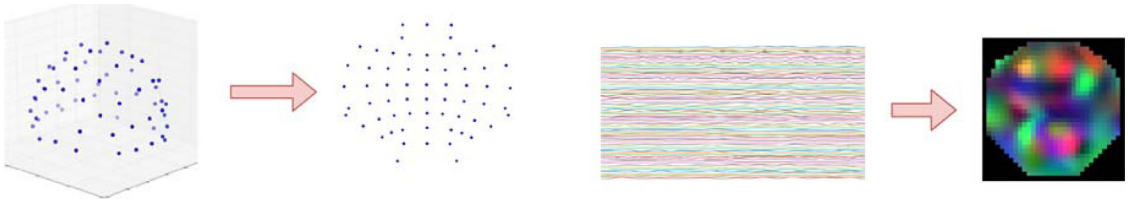


Fig. 3. Left: Polar Projection transform 3D coordinate to 2D coordinate [42]. Right: EEG signal to image example. It maps the multi-channel sequence like data into grid like data, which is more convenient to operate.

the background, we have seen theta (4–7Hz), alpha (8–13Hz), and beta (13–30Hz) wave are most representative for EEG signals when people are awake [44]. Thus, these three frequency bands are extracted from the original EEG, and the sum of squared absolute values in these frequency bands are used, forming a 64 × 3 map. To form an RGB EEG image, the theta frequency will be the red channel, alpha the green channel and beta the blue channel. For each frequency band (64 × 1), shown in Fig. 3, Azimuthal Equidistant Projection (AEP) also known as Polar Projection is used to map the three-dimensional 64 channel position into two-dimensional positions on a flat surface. That is, all EEG electrodes positions are mapped into a consistent 2D space because the original EEG electrodes are distributed over the scalp in a three-dimensional fashion. In this way, each 64 × 1 frequency band can be mapped to a 32 × 32 mesh, forming 32 × 32 × 3 data. The CloughTocher scheme is used for estimating the values in-between the electrodes over the 32 × 32 mesh. Thus, a trial of 64 × 256 EEG signals is transformed to 32 × 32 × 3 color pictures.

The motivation for this is straight-forward. For the EEG2Img method, theoretically, we can adjust the size of the output EEG image as needed. For on one trial of EEG signal, we can directly transfer it to one EEG image with 32 × 32 × 3 format which is a very typical format in computer vision area and there exist many mature and successful approaches and models for such form. As a result, by utilizing such method, it is possible for us to test those models for EEG images.

(B) Autoencoder design:

The design of this CNN based autoencoder is inspired by the CNN applications for CIFAR-10 dataset [39]. The CIFAR-10 dataset consists of 60,000 32 × 32 color images in 10 classes, with 6000 images per class, with the same input dimension as our generated EEG pictures. Our encoder and decoder are described in Table 1. The design of the autoencoder follows Zeiler and Fergus' ideas for convolution and deconvolution [45]. The Rectied Linear Unit (ReLU) is used for activation layers to speed up the training process while dropout is performed after every activation layer to make the model more robust, since it forces all the layers before the dropout to extract redundant representations. Adam optimizer is used with 1e-4 learning rate and the batch size is set to 64. Xavier normal initialization is used for convolution kernels.

Table 1
Image-wise autoencoder structure.

Encoder	Decoder
Input 32 × 32 × 3 Color Image	Input 16 × 8 × 8 Matrix
3 × 3 conv, 2 × 2 max-pooling ReLU, 0.25 dropout 3 × 3 conv, 2 × 2 max-pooling ReLU, 0.25 dropout 3 × 3 conv,ReLU	3 × 3 deconv, 2 × 2 max-un-pooling ReLU, 0.25 dropout 3 × 3 deconv, 2 × 2 max-un-pooling ReLU, 0.25 dropout 3 × 3 deconv

(C) Classification task:

The features extracted from image-wise autoencoders will be flattened into a long vector, composed of 16 hidden unit representations × 64 autoencoders in the channel-wise case or 16 × 8 × 8 matrix in the image-wise case. Then we use a feedforward network with three hidden layers. During training of these three fully connected layers using 4e−5 learning rate, the encoder of both the channel-wise and image-wise autoencoders will also be fine-tuned by the classification loss using a much smaller learning rate (1e−7).

3.3. Feature filter for EEG

For the feature filter, we consider the problem of supervised domain transformation, where we are given source domain

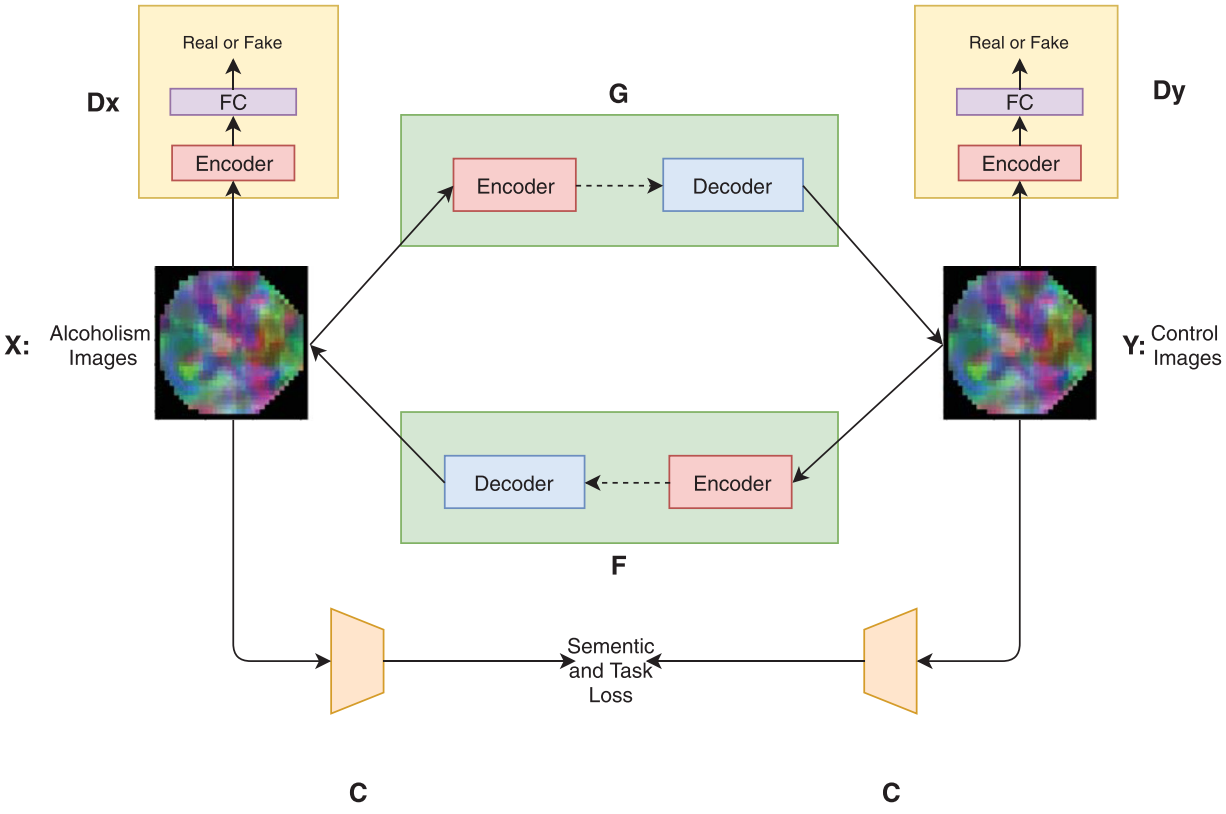


Fig. 4. Structure of feature filter. It is trained to map EEG images with unwanted features (i.e. alcoholism information) directly to EEG signals without those features (i.e. control images). Modified from CycleGAN strucuture, it has an additional classfier C which provide sementic and task loss to keep uesful feature maintained in the process of feature filtering.

distribution X with both wanted and unwanted features, labels Z for wanted features, target domain distribution Y with wanted features only. The given source domain distribution X is not paired with target domain distribution Y.

Shown in Fig. 4, for the UCI EEG dataset, the task of a feature filter map EEG images with the alcoholism condition to an EEG image with the control condition. The objective of the feature filter is to directly learn a mapping from domain X to domain Y. So given an EEG image from X domain, the mapping representation in domain Y is our filter result. For this CycleGAN based Structure, the specific loss formulations are shown as follows.

3.3.1. Loss formulation

The objective of the feature filter is composed of three parts: adversarial loss, autoencoder loss and sentiment and classification loss. They can be expressed as:

(A) Adversarial loss:

The adversarial loss is the key part for the mapping from one distribution to another. For achieving this, the adversarial discriminator is used to judge whether the image is real or fake. For the loop $X \rightarrow G(X) \rightarrow F(G(X))$,

The ability of judging whether an image belongs to a certain distribution is given by the adversarial loss. For loop $X \rightarrow G(X) \rightarrow F(G(X))$, it is defined as:

$$L_{GAN}(G, D_Y, X, Y) = E_{x \sim pdata(x)}[log[(1 - D_Y(G(x))]] + E_{y \sim pdata(y)}[logD_Y(y)]$$

This is generally the standard format of GAN loss and used to make sure the generated samples are convincing. The adversarial loss for the loop $Y \rightarrow F(Y) \rightarrow G(F(Y))$ is in a similar format.

However, in practice, the training of a GAN is quite unstable. Though the adversarial loss will force the generated image to look similar to real images, there is no guarantee for the direction of changes. To further make sure the feature filter meets our requirements, an autoencoder loss and sentiment loss are introduced as regularization terms.

(B) Autoencoder loss:

The autoencoder loss is also called reconstruction loss or cycle-consistency. It is basically an L1 loss which is used to keep $X \approx F(G(X))$, that is the generator will be forced to maintain features from the original image to have enough information to reconstruct the image during the backward loop. As a result, for loop $X \rightarrow G(X) \rightarrow F(G(X))$, it refers to:

$$L_{AL}(G, F) = E_{x \sim pdata(x)}[||F(G(x)) - x||_1]$$

Loop $Y \rightarrow F(Y) \rightarrow G(F(Y))$ has a similar autoencoder loss to ensure G(F(Y)) is similar to Y.

(C) Sentiment and task loss:

The sentiment and task loss originates from Hoffman's CYCADA model on domain adaptation [46]. Hoffman's solution is to train a cycleGAN model with sentiment and task loss to generate fake target data $fake_Y$ from source data X_S, thereby forming ($fake_Y$, Z_S) data label pairs to advance the current state of the art domain adaptation model.

Though the objective for domain adaptation is not related to our feature filter task, their proposed sentiment and task loss is useful for building a feature filter. In their proposed CYCADA model, the goal for using sentiment and task loss is to maintain labeled information when generating ($fake_{XT}$, Z_S) data label pairs. Such an idea satisfies the property that the wanted features are maintained in our feature filter design.

The sentiment and task loss is given by an additional classifier C which gives labeled information. For the definition of task loss,

Table 2
The Conv-Deconv generator structure.

Encoder	Decoder
Input 32 × 32 × 3 Color Image	Input 128 × 8 × 8 Matrix
4 × 4 conv, Leaky ReLU, 4 × 4 conv, Leaky ReLU, 3 × 3 conv, Leaky ReLU, 3 × 3 conv, Leaky ReLU,	4 × 4 Deconv, Leaky ReLU, 4 × 4 Deconv, Leaky ReLU, Tanh

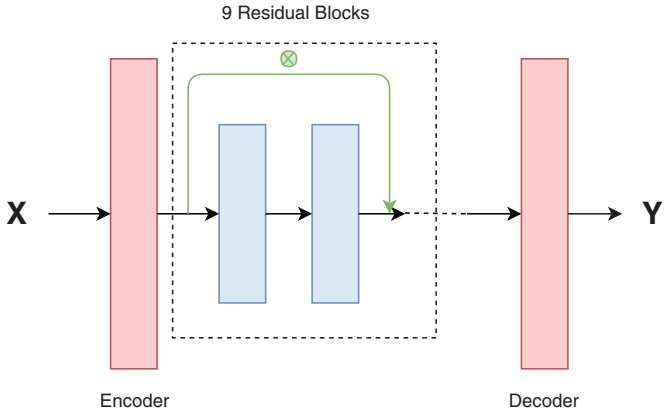


Fig. 5. Resnet-9 generator structure.

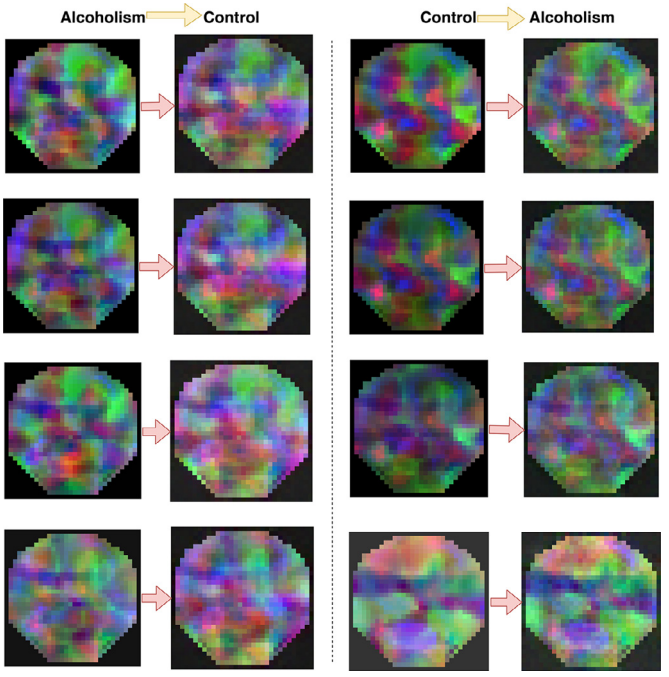


Fig. 6. Visulizations of feature filter output. Feature filter filter out features by influencing the EEG images style. These style changes cannot be interpreted but it is reflected by the change of the result of the classifier.

it is basically a simple cross-entropy loss:

$$L_{task}(C,X,Z) = -E_{(x,z)\sim(X,Z)} \sum_{k=1}^{K} \mathbb{1}_{[k=z]} log(\sigma(C^{(k)}(x)))$$

where σ means the softmax function. In practice, the classifier will be trained on source domain X and wanted label Z. As a result, loss $L_{task}(C, X, Z)$ will be used to show that the target feature label is retained.

So the classifier C works as a constraint by giving a semantic consistent loss. The semantic consistent loss will not take any explicit labeled information but focuses on the label consistency. That is, the two generators will not change the labeled information when performing image translation. If we define $p(C,X) = argmax(C(X))$, the semantic consistency loss is as follows:

$$L_{sem}(G,F,X,Y,C) = L_{task}(C,F(Y),p(C,Y)) + L_{task}(C,G(X),p(C,X))$$

As a conclusion, using the full loss functions mentioned above, we add those loss functions, and we have the final objective:

$$L_{total} = L_{task}(C,X,Z) + L_{GAN}(G,D_Y,X,Y) + L_{GAN}(G,D_X,Y,X) + L_{AL}(G,F) + L_{AL}(F,G) + L_{sem}(G,F,X,Y,C)$$

3.3.2. Network architecture

We first use a modified version of Image-wise Autoencoder as our generator (shown in Table 2), and our discriminator is the combination of Image-wise Autoencoder and one fully connected layer.

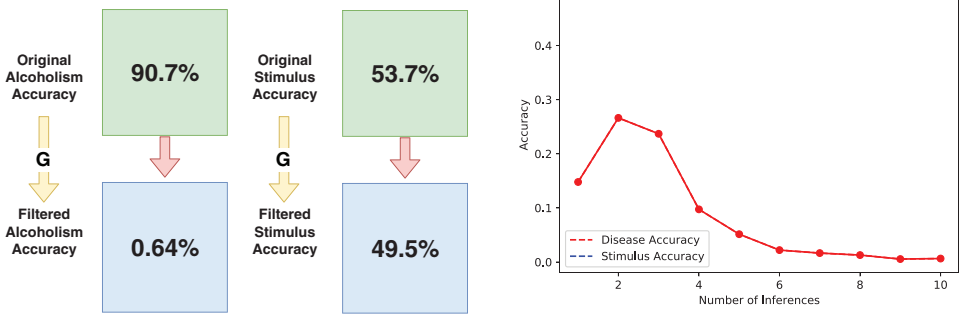
For improving performance, we tried the ResNet-9 generator and patchGAN combination for training. The combination of ResNet generator and patchGAN achieves the best performance in many image translation applications. Shown in Fig. 5, the residual-based generator is based on Johnson's ResNet model on super-resolution [47]. Similar to their work, our network is composed of one encoding blocks, nine residual blocks, and one decoding blocks. The encoding or decoding block use the convolution/deconvolution-InstanceNormReLU structure, and each residual block follows the residual connection structure which contains convolution-InstanceNorm-ReLU-convolution-InstanceNorm. The advantage of using ResNet-9 is because it is capable of handling deep neural networks [48], thereby making it easier for the generator to learn the mapping from the source distribution to the target distribution [49].

The patchGAN discriminator is derived from pix2pix [21], which is a paired image translation framework. The ordinary discriminator determines whether an image is real or fake from the entire image while the PatchGAN discriminator uses local patches. For loop $X \rightarrow G(X) \rightarrow F(G(X))$, The discriminator D_y takes in two images, the real image Y and the generated image G(X), passes them through 5 downsampling convolutional-BatchNorm-LeakyReLU layers, and outputs a matrix for further classification. That is, each element in the matrix corresponds to the classification of one patch. The advantage of using patchGAN is to avoid conflict with the autoencoder loss. Since we are using the final matrix to classify the image as real or fake, the patchGAN structure is used primarily to model high-frequency structure, whereas the autoencoder loss already provides low-frequency information [21].

4. Results and discussion

4.1. Evaluation method

The evaluation method for GAN is a difficult problem which needs to take many factors into account [50]. For a long time after the original GAN paper was published, the generated results from GANs still needed to be judged by manual selection in the CV area. After the critical work from Google brain, the FrÃ©chet Inception Distance (FID) and F1 scores [50] were introduced to judge the generation quality of a GAN. Both the FID and F1 score require a strong pretrained classifier in CV, making it impossible to directly use in the bio-signal area.



(a) Performance with Classification Results.

(b) Performance with the Number of Inferences.

Fig. 7. (a) Feature filter performance shown on additional classifiers. There is a significant decrease on unwanted feature (alcoholism) accuracy and mild drop on wanted feature (stimulus) accuracy. (b) Performance using multiple inference. Multiple inference has mild influence on stimulus accuracy but decrease alcoholism accuracy by a large margin.

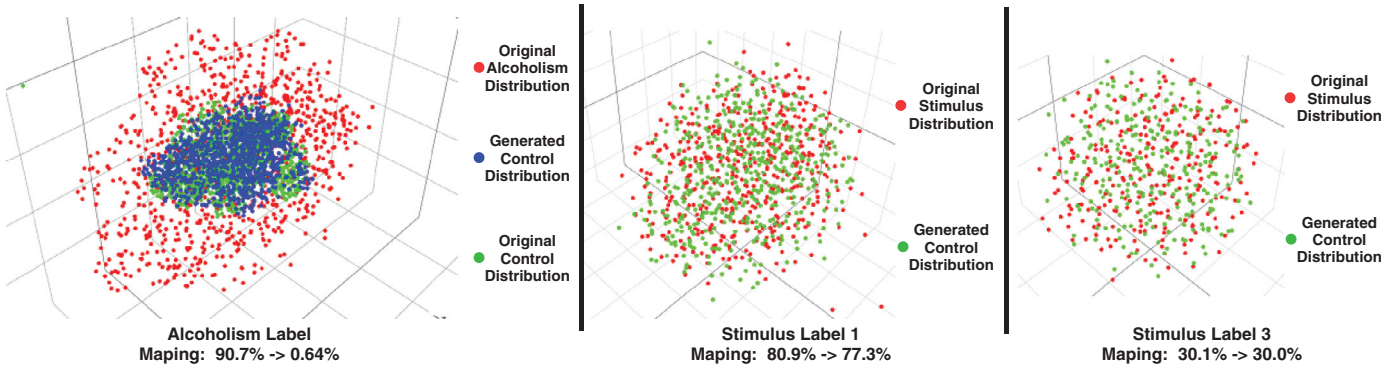


Fig. 8. Performance with t-SNE visualization. There exists a clear gap between alcoholism images and control images. The generated distribution is similar to original control distribution. Two figures on the right further show the clear mapping result for each stimulus label.

Thus, we learn from the idea of using FID and Inception Score (IS) but simply use the idea of training an additional classifier to judge the classification accuracy changes. The classifier we take is still the Image-wise autoencoder with fully connected layer (FC) which is trained separately from adversarial training. In this work, we are trying to filter out alcoholism information while keeping stimulus information. So, the desired best result should be that we get a large alcoholism accuracy reduction while keeping reasonable stimulus accuracy (low stimulus accuracy reduction) through the GAN based autoencoder.

4.2. Experiment results

Fig. 6 shows a visual example of the result of the feature filter. The left two columns map disease EEG images to control EEG images, the right two columns map control EEG images to the disease EEG images. From each direction, it can be seen that our feature filter has made a slight style transformation to images. However, those style changes are not interpretable since features from the original EEG images are not interpretable. But from Fig. 7 left part, initially, 90.7% of the original images are correctly classified as alcoholism. After our feature filter, only 0.6% of the images are classified as alcoholism. That is nearly all images have their alcoholism information filtered out. At the same time, stimulus accuracy has only been reduced by 4.2%, and the remaining accuracy is still well above chance since it is a 5-class classification problem.

Furthermore, one testing technique is to go through the feature filter multiple times. This idea is inspired by Ge's work for grammar error correction [51]: in their work, they observed that some sentence with multiple grammatical errors cannot be corrected by the Seq2Seq [52] inference using a single round of inference. So they involve multiple rounds of inference in both training and testing. In our work, we have not involved multiple inferences in training but merely used our trained feature filter to make multiple inferences on validation and test data. The result shown in Fig. 7 indicates that result is stable after six round of inference. The accuracy increases in the first 3 rounds, we think that is because our feature filter removes unstable factors rather than filtering out the unwanted information in the first three rounds.

Table 3
Comparison Models and Loss Functions.

Method	Alcoholism Acc % (low is aim)	Stimulus Acc % (high is aim)
G:Conv-Deconv D:Conv ($L_{GAN} + L_{AL}$)	18.2	47.7
G:Resnet D:PatchGAN ($L_{GAN} + L_{AL}$)	0.643	48.9
G:Resnet D:PatchGAN (L_{total})	**0.642**	**49.5**

The performance difference between models and loss functions are shown in Table 3. The results show that the best performance after multiple times of inference on the test set. We can see that Resnet and patchGAN combination contribute most to the performance boost. The sentiment loss and task loss contributes to keeping stimulus information but does not achieve significant improvement on the drop of alcoholism accuracy. One hypothesis we have is that the stimulus classifier is currently far from a strong classifier. Our initial stimulus classifier at 53.7% is reasonable where chance is 20%, but cannot really be called a strong classifier. Thus,

we think that could be one factor why adding sentiment and task loss has not achieved a larger improvement.

4.3. Working mechanism investigation

Since style changes from EEG images are not interpretable we turn our attention to visualize distance between distributions. Inspired by the FID score, we can first get our original EEG images and generated EEG images embedded into a feature space given by some convolution layers since these feature space can be a competent representation of the original distribution. So we choose to put our original EEG images and generated EEG images into the pre-trained Image-wise Autoencoder again (without the final fully connected layer) to get feature representation and then get t-SNE visualization applied. Fig. 8 shows our t-SNE visualization results. The most left part shows the mapping from the original alcoholism distribution to generated control distribution through our feature filter. We can see that the generated control image distribution is close to the original control distribution. Also, they have clear distances from the original alcoholism image distribution which matches our significant accuracy drop on alcoholism. The middle part and right part of Fig. 8 shows the mapping from original stimulus distribution to generated stimulus distribution for stimulus label 1 and stimulus label 3 respectively. From the middle part, originally 80.9% of data is classified correctly for stimulus data with label 1 and we still get 77.3% of data classified correctly after the feature filter is applied. From the right part, we can see that is 30.1% to 30.0% accuracy changes for label 3. So we can see that no matter the original classification result, the feature filter has no serious influence on the accuracy drop and t-SNE visualizations further shows that the two distributions are nearly the same though feature filters.

4.4. Limitation and future work

The first limitation is that our method is based on EEG2Img and image translation techniques, which means that it is only suitable for short-term EEG signals. The design of a feature filter for long-term EEG signals remains to be solved. The second limitation is future work for the generator; the U-net structure is also applicable for the generator since it is also the current state of the art method for several image translation tasks. The third limitation is in our model: we simply stack error functions but do not really optimize the training procedure. To further reduce the loss of wanted features, we can begin with the modification of the training procedure for our GANs.

5. Conclusion

Removing or filtering features out of EEG signals is difficult. However, building a feature filter will have a significant improvement on people's privacy protection. This approach can lead to many useful applications, such as privacy protection. An example could be where a hospital stores only the medical condition related EEG signal, but the bank stores only personal identification part of an EEG (assuming a future ATM collects EEG for greater security). This paper proposes an information-preserving feature filter, which converts the feature filtering task to an image translation task. The experiment results using accuracy drops show that our proposed feature filter can filter out nearly 90% of unwanted features and keep most of the desired features.

Appendix A

Training procedure.

Algorithm 1 Feature Filter Training Pseudocode.

for number of training iterations **do**
  Draw a minibatch of samples $x^{(1)},\ldots, x^{(m)}$ from domain X and their labels Z
  Draw a minibatch of samples $y^{(1)},\ldots, y^{(n)}$ from domain Y
  Compute the discriminator loss on real images:

$$L^D_{real} = \frac{1}{m}\sum_{i=1}^{m} logD_x(x^{(i)}) + \frac{1}{n}\sum_{i=1}^{n} logD_y(y^{(i)})$$

  Compute the discriminator loss on fake images:

$$L^D_{fake} = \frac{1}{m}\sum_{i=1}^{m} log(1 - D_y(G(x^{(i)}))) + \frac{1}{n}\sum_{i=1}^{n} log(1 - D_x(F(y^{(i)})))$$

  Update the discriminators D_x and D_y by $\frac{1}{2}(L^D_{real} + L^D_{fake})$
  Compute the classification loss for classifier C:

$$L^C_{task} = -\frac{1}{m}\sum_{i=1}^{m}\sum_{k=1}^{K} \mathbb{1}_{[k=z]} log(\sigma(C^{(k)}(x^{(i)})))$$

  Update the classifier C by L^C_{task}
  Compute the $X \rightarrow Y$ loss for generator G:

$$L^G = \frac{1}{m}\sum_{i=1}^{m}(log(1 - D_y(G(x^{(i)}))) + L_{AL}(G, F) + L_{sem}(G, F, X, Y, C))$$

  Compute the $Y \rightarrow X$ loss for generator F:

$$L^F = \frac{1}{n}\sum_{i=1}^{n}(log(1 - D_x(F(y^{(i)}))) + L_{AL}(F, G) + L_{sem}(G, F, X, Y, C))$$

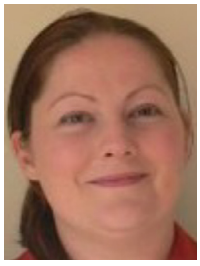
  Update the generators G and F by L^G and L^F respectively
end for

References

[1] J.R. Wolpaw, N. Birbaumer, D.J. McFarland, G. Pfurtscheller, T.M. Vaughan, Brain–computer interfaces for communication and control, Clin. Neurophysiol. 113 (6) (2002) 767–791.
[2] M.A. Lebedev, M.A. Nicolelis, Brain–machine interfaces: past, present and future, Trends Neurosci. 29 (9) (2006) 536–546.
[3] F. Su, L. Xia, A. Cai, Y. Wu, J. Ma, Eeg-based personal identification: from proof-of-concept to a practical system, in: Proceedings of the 2010 Twentieth International Conference on Pattern Recognition (ICPR), IEEE, 2010, pp. 3728–3731.
[4] N.D. Truong, A.D. Nguyen, L. Kuhlmann, M.R. Bonyadi, J. Yang, O. Kavehei, A generalised seizure prediction with convolutional neural networks for intracranial and scalp electroencephalogram data analysis, arXiv preprint arXiv:1707.01976, (2017).
[5] F. Ebrahimi, M. Mikaeili, E. Estrada, H. Nazeran, Automatic sleep stage classification based on eeg signals by using neural networks and wavelet packet coefficients, in: Proceedings of the Thirtieth Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2008. EMBS 2008., IEEE, 2008, pp. 1151–1154.
[6] S. Palazzo, C. Spampinato, I. Kavasidis, D. Giordano, M. Shah, Generative adversarial networks conditioned by brain signals, PDF available on ucf. edu (2017).
[7] U. Orhan, K.E. Hild, D.E. II, B. Roark, B. Oken, M. Fried-Oken, Rsvp keyboard: an eeg based typing interface, in: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing/sponsored by the Institute of Electrical and Electronics Engineers Signal Processing Society. ICASSP, NIH Public Access, 2012.
[8] S. Gandhi, T. Oates, T. Mohsenin, D. Hairston, Denoising time series data using asymmetric generative adversarial networks, in: Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer, 2018, pp. 285–296.
[9] J.A. Urigüen, B. Garcia-Zapirain, Eeg artifact removalstate-of-the-art and guidelines, J. Neural Eng. 12 (3) (2015) 031001.
[10] S.-Y. Shao, K.-Q. Shen, C.J. Ong, E.P. Wilder-Smith, X.-P. Li, Automatic eeg artifact removal: a weighted support vector machine approach with error correction, IEEE Trans. Biomed. Eng. 56 (2) (2009) 336–344.
[11] Y. Liu, Y. Yao, W. Zhengjie, J. Plested, T. Gedeon, Generalized alignment for multimodal physiological signal learning, in: Proceedings of the International Joint Conference on Neural Networks (IJCNN), IEEE, 2019, pp. 1–8.

[12] A. Schlögl, An overview on data formats for biomedical signals, in: Proceedings of the World Congress on Medical Physics and Biomedical Engineering, September 7–12, 2009, Munich, Germany, Springer, 2009, pp. 1557–1560.
[13] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.
[14] V. Lawhern, A. Solon, N. Waytowich, S. Gordon, C. Hung, B. Lance, Eegnet: a compact convolutional neural network for eeg-based brain-computer interfaces, J. Neural Eng. 15 (5) (2018). 056013–056013
[15] M. Schröder, T.N. Lal, T. Hinterberger, M. Bogdan, N.J. Hill, N. Birbaumer, W. Rosenstiel, B. Schölkopf, Robust eeg channel selection across subjects for brain-computer interfaces, EURASIP J. Appl. Signal Process. 2005 (2005) 3103–3112.
[16] Z. Wang, R.M. Hope, Z. Wang, Q. Ji, W.D. Gray, Cross-subject workload classification with a hierarchical Bayes model, Neuroimage 59 (1) (2012) 64–69.
[17] K. Sundararajan, Privacy and security issues in Brain Computer Interfaces, Ph.D. thesis, Auckland University of Technology, 2018.
[18] J. Kulynych, Legal and ethical issues in neuroimaging research: human subjects protection, medical privacy, and the public communication of research results, Brain Cogn. 50 (3) (2002) 345–357.
[19] V. Robinson, E.B. Varghese, A novel approach for ensuring the privacy of eeg signals using application-specific feature extraction and AES algorithm, in: Proceedings of the International Conference on Inventive Computation Technologies (ICICT), 2, IEEE, 2016, pp. 1–6.
[20] S. Al-Janabi, I. Al-Shourbaji, M. Shojafar, S. Shamshirband, Survey of main challenges (security and privacy) in wireless body area networks for healthcare applications, Egyp. Inform. J. 18 (2) (2017) 113–122.
[21] P. Isola, J.-Y. Zhu, T. Zhou, A.A. Efros, Image-to-image translation with conditional adversarial networks, in: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2017, pp. 5967–5976.
[22] Y. Yao, J. Plested, T. Gedeon, Deep feature learning and visualization for eeg recording using autoencoders, in: Proceedings of the International Conference on Neural Information Processing (ICONIP) 2018, 2018, p. 13.
[23] Y. Yao, J. Plested, T. Gedeon, A feature filter for eeg using GAN-based autoencoder, in: Proceedings of the International Conference on Neural Information Processing (ICONIP) 2018, 2018, p. 9.
[24] Y. Yao, Y. Liu, W. Zhengjie, J. Plested, T. Gedeon, Improved techniques for building eeg featurefilters, in: Proceedings of the International Joint Conference on Neural Networks (IJCNN), IEEE, 2019, pp. 1–8.
[25] Y. LeCun, Y. Bengio, et al., Convolutional networks for images, speech, and time series, in: The Handbook of Brain Theory and Neural Networks, 3361, 1995, p. 1995.
[26] R.T. Schirrmeister, J.T. Springenberg, L.D.J. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard, T. Ball, Deep learning with convolutional neural networks for eeg decoding and visualization, Hum. Brain Mapp. 38 (11) (2017) 5391–5420.
[27] Y. Li, K. Dzirasa, L. Carin, D.E. Carlson, et al., Targeting EEG/LFP synchrony with neural nets, in: Advances in Neural Information Processing Systems, 2017, pp. 4620–4630.
[28] S. Min, B. Lee, S. Yoon, Deep learning in bioinformatics, Brief Bioinform. 18 (5) (2017) 851–869.
[29] M. Hajinoroozi, Z. Mao, T.-P. Jung, C.-T. Lin, Y. Huang, Eeg-based prediction of driver's cognitive performance by deep convolutional neural network, Signal Process. Image Commun. 47 (2016) 549–555.
[30] T. Gedeon, J. Catalan, J. Jin, Image compression using shared weights and bidirectional networks, in: Proceedings of the Second International ICSC Symposium on Soft Computing (SOCO'97), 1997, pp. 374–381.
[31] J. Masci, U. Meier, D. Cireşan, J. Schmidhuber, Stacked convolutional auto-encoders for hierarchical feature extraction, in: Proceedings of the International Conference on Artificial Neural Networks, Springer, 2011, pp. 52–59.
[32] S. Stober, A. Sternin, A.M. Owen, J.A. Grahn, Deep feature learning for eeg recordings, arXiv preprint arXiv:1511.04306, (2015).
[33] Y.R. Tabar, U. Halici, A novel deep learning approach for classification of eeg motor imagery signals, J. Neural Eng. 14 (1) (2016) 016003.
[34] 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.
[35] A. Radford, L. Metz, S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, arXiv preprint arXiv:1511.06434, (2015).
[36] M.-Y. Liu, T. Breuel, J. Kautz, Unsupervised image-to-image translation networks, in: Advances in Neural Information Processing Systems, 2017, pp. 700–708.
[37] J.-Y. Zhu, T. Park, P. Isola, A.A. Efros, Unpaired image-to-image translation using cycle-consistent adversarial networks, in: Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), IEEE, 2017, pp. 2242–2251.
[38] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.
[39] A. Krizhevsky, G. Hinton, Learning multiple layers of features from tiny images, Technical Report, Technical report, University of Toronto, 2009.
[40] L. Yu, W. Zhang, J. Wang, Y. Yu, Seqgan: Ssequence generative adversarial nets with policy gradient., in: AAAI, 2017, pp. 2852–2858.
[41] J. Guo, S. Lu, H. Cai, W. Zhang, Y. Yu, J. Wang, Long text generation via adversarial training with leaked information, in: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, 2018.
[42] P. Bashivan, I. Rish, M. Yeasin, N. Codella, Learning representations from eeg with deep recurrent-convolutional neural networks, arXiv preprint arXiv:1511.06448, (2015).
[43] P. Sykacek, S.J. Roberts, Adaptive classification by variational Kalman filtering, in: Advances in Neural Information Processing Systems, 2003, pp. 753–760.
[44] P.A. Abhang, B.W. Gawali, Correlation of eeg images and speech signals for emotion analysis, Br. J. Appl. Sci. Technol. 10 (5) (2015) 1–13.
[45] M.D. Zeiler, R. Fergus, Visualizing and understanding convolutional networks, in: Proceedings of the European Conference on Computer Vision, Springer, 2014, pp. 818–833.
[46] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. Efros, T. Darrell, Cycada: cycle-consistent adversarial domain adaptation, in: Proceedings of the International Conference on Machine Learning, 2018, pp. 1994–2003.
[47] J. Johnson, A. Alahi, L. Fei-Fei, Perceptual losses for real-time style transfer and super-resolution, in: Proceedings of the European Conference on Computer Vision, Springer, 2016, pp. 694–711.
[48] C. Szegedy, S. Ioffe, V. Vanhoucke, A.A. Alemi, Inception-v4, inception-resnet and the impact of residual connections on learning., in: AAAI, 4, 2017, p. 12.
[49] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[50] M. Lucic, K. Kurach, M. Michalski, S. Gelly, O. Bousquet, Are GANs created equal? A large-scale study, in: Advances in Neural Information Processing Systems, 2018, pp. 700–709.
[51] T. Ge, F. Wei, M. Zhou, Reaching human-level performance in automatic grammatical error correction: an empirical study, 2018 arXiv preprint arXiv:1807.01270.
[52] J. Gehring, M. Auli, D. Grangier, D. Yarats, Y.N. Dauphin, Convolutional sequence to sequence learning, in: Proceedings of the Thirty-forth International Conference on Machine Learning-Volume 70, JMLR. org, 2017, pp. 1243–1252.

Yue Yao is a Ph.D. student in the Research School of Computer Science at the Australian National University. He has completed the degree of Bachelor of Advanced Computing at the ANU, with his Honours project supervised by Jo Plested and Prof Tom Gedeon. He is supervised for his Ph.D. by Prof Tom Gedeon. His paper in ICONIP 2018 was nominated as best student paper. Yue's research area is in the use of Generative Adversarial Networks on physiological data (primarily EEG) for advanced data manipulation and inference. For data manipulation, the objective can be to modify some signal bearing components of the EEG while not modifying other signals, as related to the task undertaken. For inference, the objective can be to determine human internal states such as choice, emotion discrimination or creativity.



Josephine Plested is a Ph.D. student in the Research School of Computer Science at the Australian National University. She has completed degrees of Bachelor of Actuarial Studies, Bachelor of Economics and Master of Computing at the ANU. She is supervised in her Ph.D. by Prof Tom Gedeon. Jo's research area is in developing best practices for the application of unsupervised transfer learning neural network techniques to various practical applications in the human centered computing research area including the understanding of videos. She has a keen interest in applying this work to improve the automated diagnosis of depression as she has a background in volunteering work with people with depression and would like to see the burden of this illness reduced through better diagnosis.

Tam's (Tom) Gedeon Tom Gedeon is Chair Professor of Computer Science at the Australian National University. He is formerly Deputy Dean and Head of Computer Science at ANU. His BSc and Ph.D. are from the University of Western Australia, and Grad Dip from UNSW. He is twice a former President of the Asia-Pacific Neural Network Assembly, and former President of the Computing Research and Education Association of Australasia. Tom's research focuses on bio-inspired computing (mainly neural, deep learning, fuzzy and evolutionary) and human centred computing (mainly eye gaze, wearable physiological signals, fNIRS, thermal, EEG) to construct truly responsive computer systems (biometrics and affective computing) and humanly useful information resources (hierarchical and time series knowledge), industrial (mining, defence) and social good (medical, educational) applications.