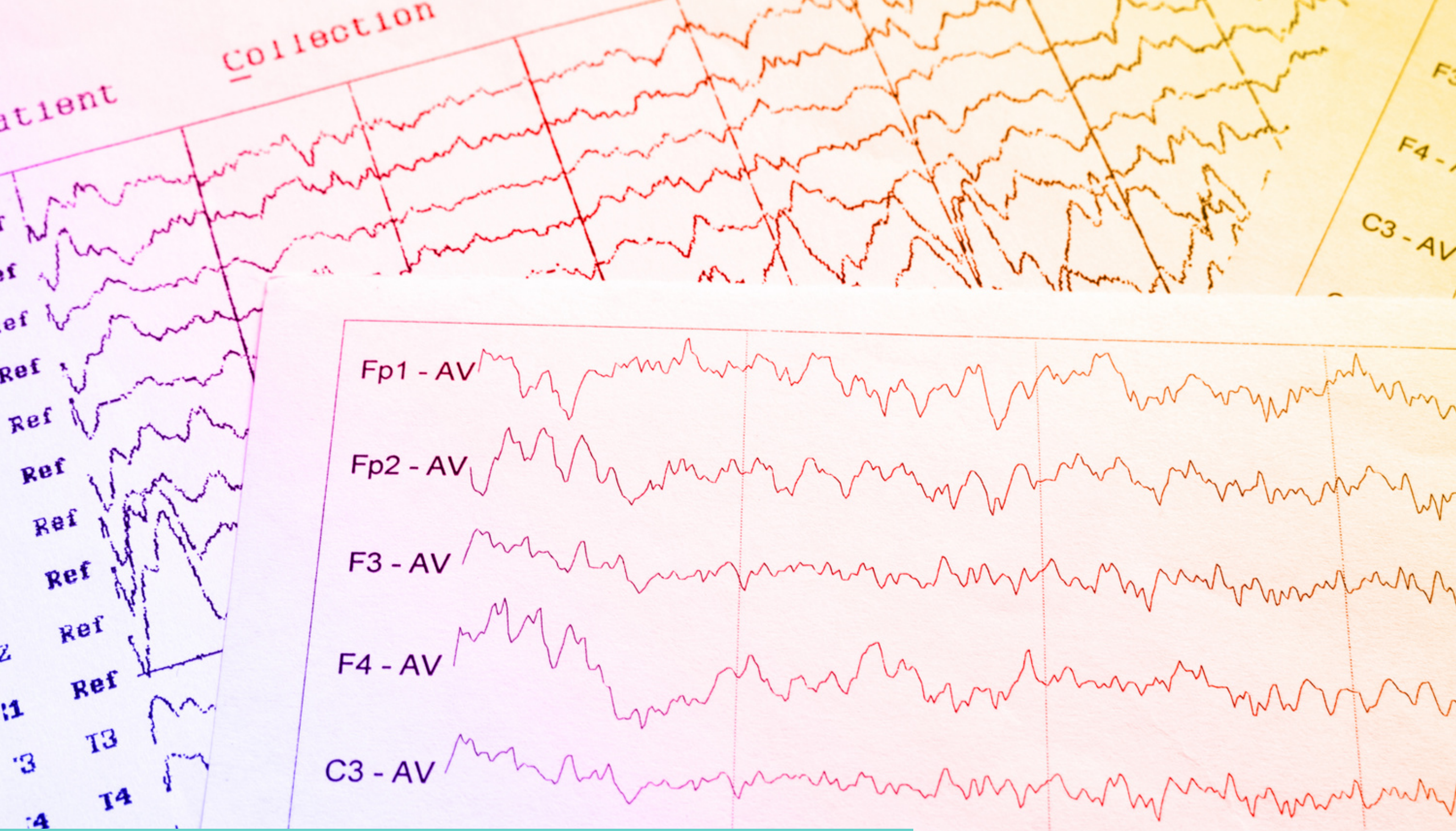
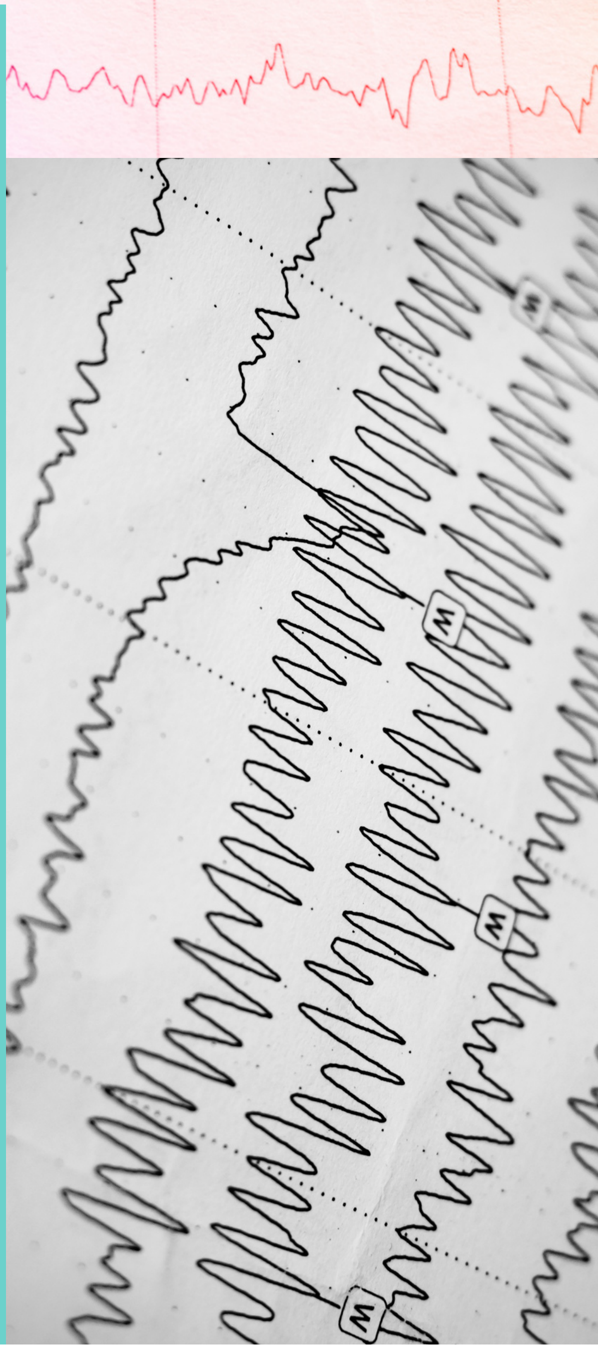


BCI COURSE

Term-Project

FBCNet: A Multi-view Convolutional Neural Network for Brain-Computer Interface Motor imagery-BCI

GROUP 4
311515023 趙品瑄
311554023 張瑀芯
311553009 許瑋芸
311551180 楊佳誠
1111950031 陳妍沂



Part 1. Introduction

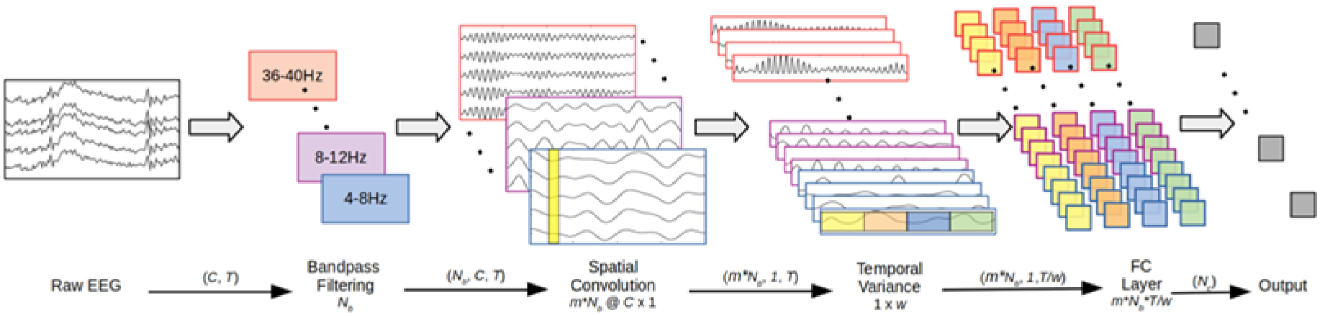
In this report, we will describe our implementation of the paper: *FBCNet: A Multi-view Convolutional Neural Network for Brain-Computer Interface.*

This paper proposed an end to end deep learning architecture for motor imagery classification tasks. The method for MI classification can be separated into 2 groups: traditional machine learning method and deep learning method.

Traditional machine learning methods such as filter bank common spatial pattern support vector machine (FBCSP-SVM) focused on the extraction of neuro-physiologically sound features from EEG data to achieve higher classification accuracies. However, they still suffer from the high susceptibility to intra-trial variance and have a high dependency on handcrafted features.

Deep learning architectures, particularly the ones based on Convolutional Neural Network (CNN) such as EEGNet, have gained popularity in the BCI domain due to their ability of effectively learning the local connectivity patterns from the given data. These architectures have outperformed the classical machine learning techniques, but improvements achieved are still marginal.

In this work, the author proposed FBCNet to further improve the accuracy by combining the idea of FBCSP and CNN. The following is the architecture of FBCNet:



Part 1. Introduction

The architecture can be summarized into 4 stages：

1. Multi-view data representation:
Spectrally filtering the raw EEG with multiple narrow-band filters, the idea was adopted from FBCSP.

2. Spatial transformation learning:
Spatial discriminative patterns learned using a Depthwise Convolution layer.

3. Temporal feature extraction:
Novel Variance layer is used to effectively extract the temporal information. The main function of this variance layer, similar to a pooling layer, is to reduce the number of features from ($m \times Nb \times T$) to ($m \times Nb \times T/w$)

Output of Variance layer:

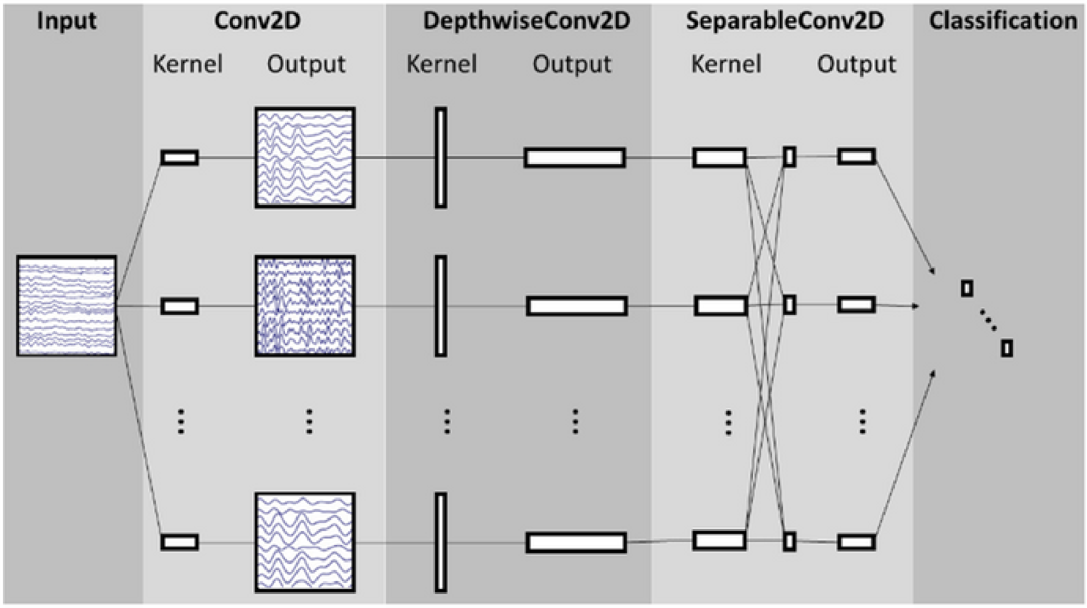
$$v = Var(g(t)) = \frac{1}{T}\sum_{t=0}^{T-1}(g(t)-\mu)^2$$

4. Classification:
Log activation to FC layer, then passed to softmax layer to get the output probabilities of each class.

We tried to implement FBCNet and compare the performance against other baseline models in various training schemes.

Part 2. Baselines

1. EEGNet



In EEGNet, it can be divided into four parts: Conv2D, DepthwiseConv2D, SeparableConv2D and Classification. The four major parts are as follows.

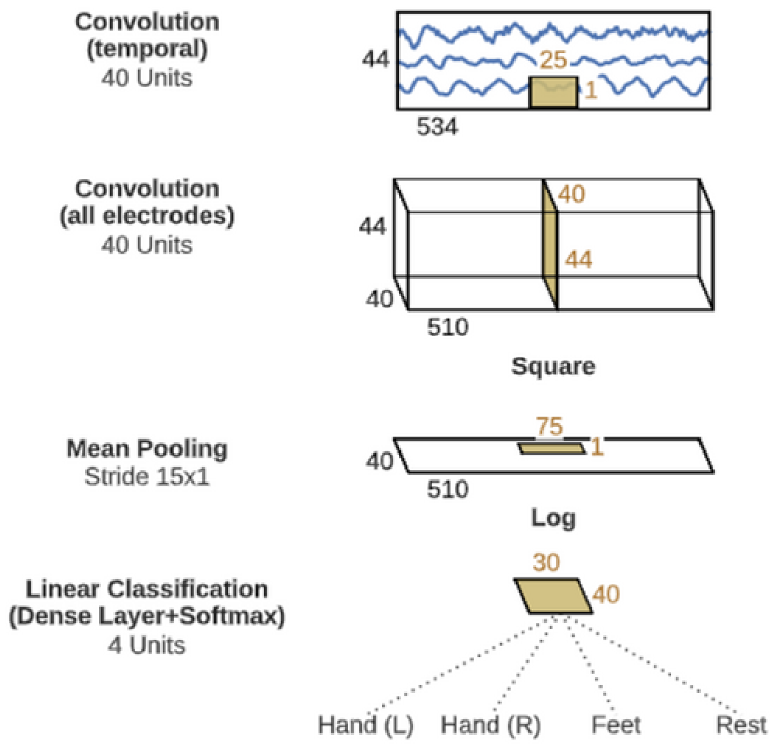
1. **Conv2D**: In this layer, the input EEG signal will be converted into a 2D feature map through 2D convolutional filters for subsequent operations.
2. **Depthwise Conv2D**: In this layer, a kernel with a size of (channel, 1) is mainly used to learn a spatial filter.
3. **SeparableConv2D**: This layer is mainly used to learn how to summarize each feature map individually and then optimally merging the outputs. Using this layer can not only effectively reduce the number of parameters, but also clearly understand the relationship between each feature map.
4. **Classification**: In this layer, the feature will be passed to softmax for classification of N categories. As mentioned in the paper, they omitted the use of dense layer for feature aggregation before the softmax to reduce the number of parameters in the model.

Pros
EEGNet uses depthwise separable convolutions, which makes it computationally efficient and reduces overfitting. It also includes a spatial attention mechanism that helps the model focus on important features in the EEG signals.

Cons
EEGNet may not be able to capture as much complex information in the EEG signals as SCCNet, and may not perform as well on more challenging classification tasks.

Part 2. Baselines

2. ShallowConvNet



The idea of ShallowConvNet is inspired by FBCSP, which is mainly designed to decode features related to frequency band power.

In the first two layers, EEG data will be passed through the temporal and spatial convolution, which is similar to the bandpass filtering in FBCSP and the spatial filtering of CSP.

Then, it will passed to an average pooling layer and softmax to achieve log-variance calculation, which is similar to FBCSP.

Since ShallowConvNet has multiple pooling regions between a trial, it can learn the temporal structure of the power variation of the frequency band in a trial, which can effectively improve the classification performance.

Pros

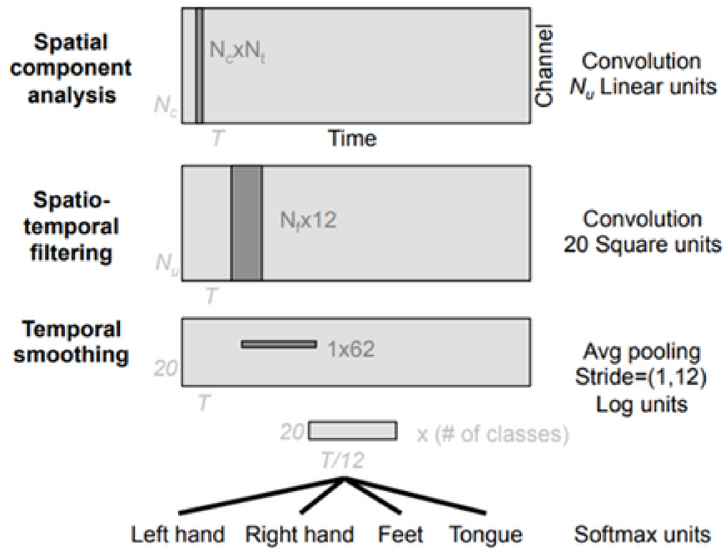
ShallowConvNet was proposed simultaneously with DeepConvNet for decoding motor-imagery EEG data, but ShallowConvNet has been more popular than DeepConvNet because of its smaller model size. Therefore, it can learn more complicated features using the less parameters in the training phase.

Cons

Compared with SCCNet and EEGNet, ShallowConvNet has larger model size because it is a deep neural network. As a result, on the easy classification task, the model complexity of ShallowConvNet is too large to learn well.

Part 2. Baselines

3. SCCNet



The main part of SCCNet consists of convolutional kernels, which are used to extract spatial and temporal features from EEG data. The model is designed to use spatial filter on EEG data for feature selection and suppress noise at the same time.

In spatial component analysis, convolution is used to extract the features of EEG, trying to decompose the original EEG data from channel domain to component domain.

In spatio-temporal filtering, convolution will be applied to temporal and spatial domains simultaneously, trying to perform spectral filtering, correlation between components, and other spatial-temporal related analysis on the spatial information of EEG.

In temporal smoothing, the average pooling layer is used to perform smoothing in the temporal domain and reduce the dimension. Finally, the prediction result is obtained through softmax.

Pros

SCCNet is also a shallow neural network that has the smaller model size. The design of SCCNet focuses on leveraging the benefits from applying spatial kernel to multi-channel EEG data in the first convolutional layer for purposes such as feature extraction and noise reduction.

Cons

Because the feature of small model size, SCCNet can only be adapted to relatively small EEG datasets, it may get the worse result on the large dataset classification task.

Part 3. Implementation

learning rate： 0.001

loss function : Negative log likelihood loss

optimizer： Adam

Training process：

Stage 1

- The model was trained using only the training set.
- Stopped if no increase in the validation set accuracy for consecutive 200 epochs.
- Network parameters with the best validation set accuracy were restored to use in the next stage.

Stage 2

- Training procedure was continued in the second stage wherein the model was trained with the complete training data (train + validation set).
- Stopped when the validation set loss reduced below the stage 1 training set loss.
- To avoid infinite training in the situation of non-convergence, the maximum number of training epochs were limited to 1500 and 600 for training stage 1 and 2 respectively.

Three training schemes：

1. 10-fold cross validation: The 10-fold cross validation (CV) analysis was conducted in a 10 fold setting, with the 9 folds being used for training and 1 fold for testing. Training data was further divided into a training set and a validation set. The test folder was never used in any of the training.
2. Hold-out analysis: Same as the Individual training scheme. The complete data from session 1 for the given subject was used for the training purpose, and the resulting model was tested on the session 2 data.
3. Subject independent: Use all the other subject's data for training purpose, and the resulting model was tested on the specific subject's session 2 data.

Part 3. Implementation

Experiment

Dataset

- **Original paper:** 4 dataset (BCIC-IV-2A, OpenMBI Data, Stroke Data A, Stroke Data B)
- **Ours:** used BCIC-IV-2A only

Baseline model comparison

- **Original paper:** compared FBCNet against FBCSP-SVM, EEGNet, DeepConvNet
- **Ours:** compared FBCNet against EEGNet, ShallowConvNet, SCCNet

Training scheme

- **Original paper:** 10-fold cross validation, HO(individual)
- **Ours:** 10-fold cross validation, HO(individual), subject independent for 9 subjects

Results

The upper table is the accuracy results from the original paper, the bottom table is the results for our implementation. In 10-fold cross validation scheme, the results for both EEGNet and FBCNet is close between original work and ours (EEGNet: 73.13 vs 74.79, FBCNet: 79.03 vs 79.09). The hold out test results slightly deviate from original work. This may be due to larger variability in smaller datasets.

Test Configuration	FBCSP-SVM	Deep ConvNet	EEGNet-8,2	FBCNet
10-fold cross validation	75.89	72.20	73.13	**79.03**
Hold out test set	68.06*	72.22	73.15	**76.20**

	EEGNet	ShallowConvNet	SCCNet	FBCNet
10-fold cross validation	74.79	56.01	71.68	**79.09**
Hold Out	70.80	52.58	67.44	**75.42**
subject independent	**61.96**	58.02	57.87	45.94

Part 3. Implementation

Results: accuracy for individual subjects

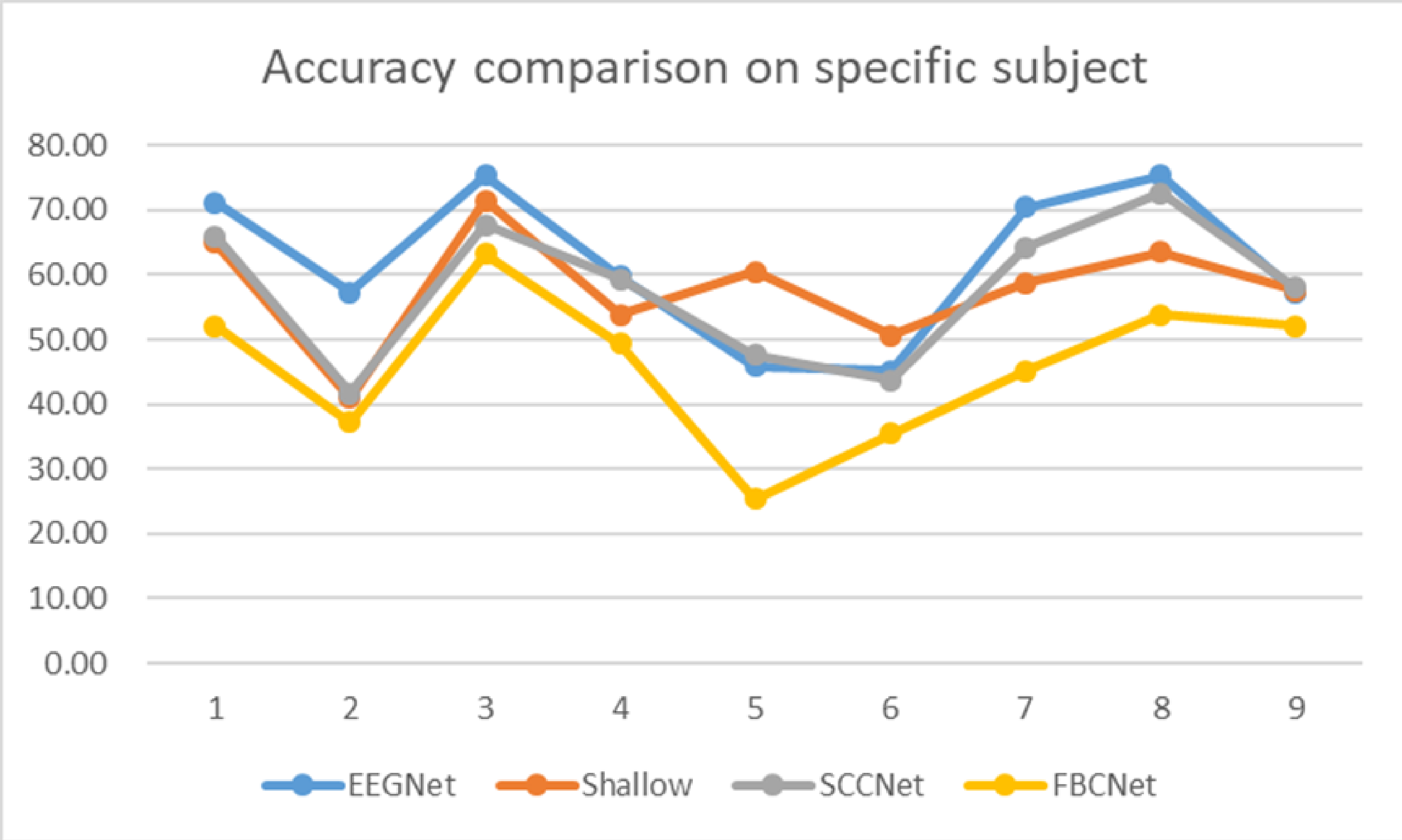
Training schemes	10-fold cross validation			
Subject	EEGNet	Shallow	SCCNet	FBCNet
1	77.32	55.22	70.98	83.92
2	57.32	48.30	59.29	62.50
3	82.81	66.87	90.09	93.83
4	69.46	43.92	53.13	68.48
5	77.41	49.41	59.91	81.47
6	66.74	43.08	63.17	59.33
7	73.62	67.09	82.90	93.34
8	81.61	61.11	82.95	87.58
9	86.83	69.06	82.68	81.38
average	74.79	56.01	71.68	79.09

Training schemes	Hold Out			
Subject	EEGNet	Shallow	SCCNet	FBCNet
1	76.56	59.37	71.88	84.72
2	59.72	37.84	51.04	57.29
3	86.63	71.52	89.24	89.58
4	68.40	40.27	57.29	78.81
5	70.49	45.13	62.15	70.48
6	57.99	47.22	51.74	54.86
7	73.00	51.73	71.88	81.94
8	72.83	57.63	79.17	81.25
9	71.61	62.50	72.57	79.86
average	70.80	52.58	67.44	75.42

Training schemes	Subject independent			
Subject	EEGNet	Shallow	SCCNet	FBCNet
1	71.18	64.93	65.97	52.08
2	57.29	40.97	41.67	37.15
3	75.35	71.53	67.71	63.19
4	59.73	53.82	59.38	49.30
5	45.83	60.42	47.57	25.34
6	45.14	50.69	43.75	35.41
7	70.49	58.68	64.24	45.13
8	75.35	63.54	72.57	53.81
9	57.29	57.64	57.99	52.08
average	61.96	58.02	57.87	45.94

Part 3. Implementation

Results: subject indendendent scheme



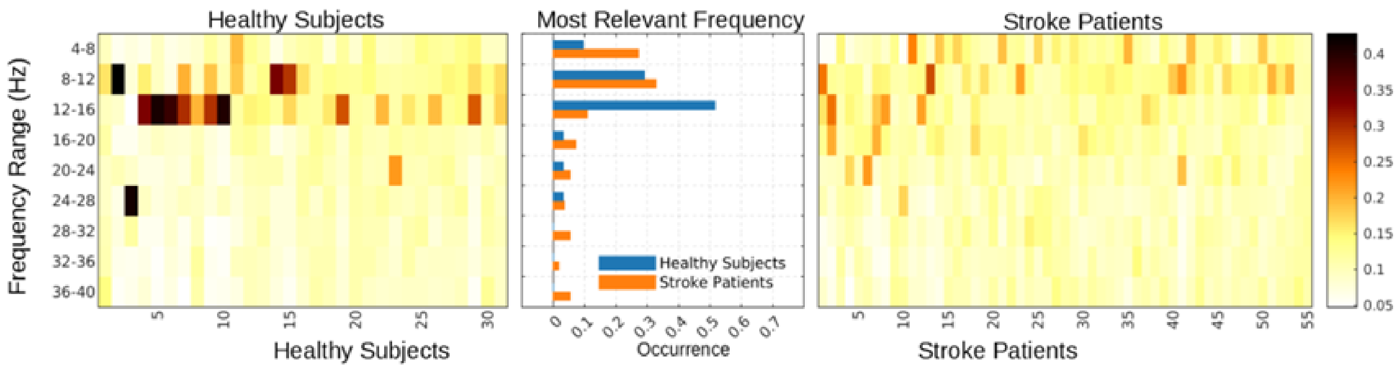
Part 4. Discussion

It is evident that there is significant variability across subjects, particularly when examining subject 5. In this case, the accuracy of FBCNet is essentially comparable to random guessing. Even if we exclude subject 5 as an outlier, FBCNet consistently performs the worst among the architectures considered.

One possible explanation for FBCNet's poor performance compared to other architectures is the the emphasis on features in specific frequency bands. The filter bank and the variance layer, which aims to represent the spectral power in the time series, may efficiently extract individual spectral characteristics. However, the differences in individual spectral characteristics could also contribute to the lower accuracy of FBCNet.

In the bottom figure, taken from the original paper, the researchers compare the importance of spectral features between healthy subjects and stroke patients. However, even among healthy subjects, there are substantial differences in spectral characteristics. We believe this disparity in spectral characteristics could explain FBCNet's performance in a subject-independent scheme, particularly in cases where the spectral characteristics are significantly different.

Overall, the observed variability across subjects and the divergence in individual spectral characteristics shed light on FBCNet's comparatively poorer performance.



Part 5. Conclusion

In our implementation, we assessed the performance of FBCNet in comparison to several baseline models across different training schemes. Consistent with the findings in the original paper, FBCNet demonstrated strong performance, possibly due to its ability to learn subject-specific spectral power-related features.

However, when considering a subject-independent scheme, it became apparent that other general-purpose architectures such as EEGNet, ShallowConvNet, and SCCNet consistently outperformed FBCNet. Despite its success in subject dependent and individual schemes, poor performance in subject independent scheme may limit applications of FBCNet to specific use compare to alternative architectures.

In conclusion, while FBCNet showed promise in learning subject-specific spectral power-related features, its performance fell short when tested in a subject-independent setting. The other architectures proved to be more effective in terms of overall performance and generalizability.