12-21-2021

A Deep Generative and Discriminative Approach in Modelling Spatial-spectral Dynamics of Varying Cognitive Load from EEG Recordings

Felix Havugimana

A DEEP GENERATIVE AND DISCRIMINATIVE APPROACH IN MODELLING SPATIAL-SPECTRAL DYNAMICS OF VARYING COGNITIVE LOAD FROM EEG RECORDINGS

by

Felix Havugimana

A Thesis

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

Major: Computer Engineering

The University of Memphis

December 2021

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

DEDICATION

To Mom, my brothers and Sisters Josephine, Chantal, Felicien, Lambert, and Berthilde for your unconditional and endless love, prayers, and support.

ACKNOWLEDGMENTS

I would like to take this opportunity to express my gratitude to my major advisor Dr. Mohammed Yeasin for giving me the opportunity to be part of his lab and for his continued support and mentoring. Dr. Yeasin has been patient with me, knowing that I came with little knowledge in programming and Neuroscience ; he taught me how to learn by asking good research questions and identifying the research problem. Also, I would like to thank Dr. Pouya Bashivan for the permission to use his dataset. I want to thank lab mates at CVPIA lab Kazi Ashraf Moinuddin, Dr. Mohammad Bany, Dr. Sultan Mahmud, Dr. Faruk Ahmed, Dr. Rakib -Al-Fahad, Haitham Najdawi, Gaurav Singh, Sultana Akhter, and Fatin Ishrak for their advice and support. Also, I want to thank my brother Dr. Kubwimana J. Lambert for being a great mentor. And above all, I would like to thank the almighty God for giving me the strength to do this work.

ABSTRACT

Cognitive load refers to the amount of used working memory resources, which is limited in both capacity and duration. Predicting cognitive load from raw electroencephalogram (EEG) recordings remains a challenge because of the high degree of noise due to technical variations in the recording process and the multi-factorial nature of the mapping between the EEG data and cognitive load. We present parameter-optimized deep convolutional neural network (CNN) models to predict four levels of cognitive load. We use eigenspace-based bootstrap sampling and Generative Adversarial Network (GAN) to address the issue of noise and a small number of samples in EEG. We transform time-series signals into a spatial-spectral representation called Topomap, which maintains both spatial and spectral information embedded in EEG recordings. We use two different EEG data representations for cognitive load prediction. First, we use power spectral densities of three individual frequency bands (Theta, Alpha, Beta) to create the topomap. Second, we combine all three bands to develop a composite representation. We performed empirical evaluations to determine the role of individual frequency bands in predicting cognitive load. The prediction accuracy of CNN models built using Theta, Alpha, Beta bands, and composite representation are 89%, 89%, 91%, and 92%, respectively. The results suggest that the Beta band has the most predictive power and composite representation produces higher accuracy than the individual frequency bands.

CHAPTER 1. INTRODUCTION

Working memory (WM) provides temporary storage and processing of information necessary for performing cognitive tasks such as reasoning, decision-making, and language comprehension Baddeley (1992). The amount of used WM resources is known as a cognitive load (CL). The WM is limited in both capacity and duration. According to Hick's law, the reaction time to a stimulus increases logarithmically with the number of available choices Hick (1952). Excessive WM usage can lead to cognitive overload, which has adverse effects on performing cognitive tasks Paas et al. (2003).

According to Sweller Sweller et al. (1998), there are three types of cognitive load: intrinsic, extraneous, and germane cognitive load. Intrinsic CL arises from the inherent complexity of the task to be learned; extraneous cognitive load comes from how information is presented to a learner, and germane cognitive load refers to the effort put into creating permanent storage of knowledge (schema). Measuring and minimizing extraneous cognitive load is critical in applications such as instruction design, brain-computer interface(BCI) Ozkan and Kahya (2018); Roy et al. (2013), human-computer interaction(HCI) Kumar and Kumar (2016); Chen et al. (2011); Hollender et al. (2010), and learning institutions Paas et al. (2010); Niederhauser et al. (2000); Choi et al. (2014). Therefore, accurate prediction of cognitive load is critical to ensure effective WM usage while performing a task.

Neuro-imaging techniques such as electroencephalogram (EEG), functional magnetic resonance imaging (fMRI), and positron emission tomography (PET) are the primary means of collecting brain activities Puce and Hämäläinen (2017); Junghöfer et al. (2006); Nasrallah and Dubroff (2013). The EEG is the most widely used method due to its high temporal resolution and affordable cost. Time-frequency analyses, such as Fast Fourier

transform (FFT), Short term fast Fourier transform (STFFT), and wavelet transform, are commonly used methods to extract features from EEG signals Zarjam et al. (2013); Amin et al. (2015); Bashivan et al. (2015); Liang et al. (2005); Mazher et al. (2017).

Many reported literature used machine learning and spectral features to predict cognitive load or estimate working memory capacity. For example, in Bashivan et al. (2015), the wavelet entropy and band-specific powers from single-trial EEG data were used to classify cognitive load with 92% accuracy. Support vector machine(SVM) was used in Nuamah et al. (2017) to predict cognitive task using task engagement index (ratio of EEG power bands (beta/(alpha + theta)) features with an average accuracy of 93.33%. Recently, Convolutional neural network (CNN) was employed in classifying CL with 93% Liu and Liu (2017). Nevertheless, deep learning has not been extensively used for cognitive load prediction mainly due to the lack of adequate EEG data to train the models, which is essential in achieving a satisfactory model performance.

In this work, we propose a spatial-spectral representation of EEG recordings called topomap to capture spatial-spectral features and use deep CNN architecture to learn the representation (features) and predict (extraneous) cognitive load for individuals engaging in a WM auditory task. The topomap represents the distribution of electrical activity across the brain as measured using electrodes locations whose position is known. It captures both spectral and spatial information of EEG recordings. Moreover, topomaps allow us to use the CNN model to extract and learn both low-level and high-level features from the spatial-spectral representation of the signal.

The proposed approach addresses the issues related to excessive noise in EEG recordings and a small sample problem that inhibits the application of deep learning models on EEG data. To reduce the technical noise prevalent in the raw EEG data, we use event-related potential (ERP) averaged over a number of trials. We applied eigenspace-based bootstrap sampling (randomly sampled with replacement) for ERP calculation, followed by

Generative Adversarial Network (GAN)-based data generation to reduce noise and obtain adequate samples for training and testing CNN models. We built parameter-optimized CNN models to determine the role of the individual frequency bands (Theta, Alpha, Beta) in predicting four levels of cognitive load. A single frequency band may not contain all information embedded in the entire EEG signal. Thus, we combine three bands to create a composite representation to capture the full range of spatial-spectral information. We perform empirical analyses to understand the predictive power of individual frequency bands and compare their performance with composite representation.
The key contributions are:

1. Implementation of eigenspace-based bootstrap sampling and GAN to address the challenge of the limited amount of EEG data in training deep CNN model.

2. Transformation of the EEG recordings into spatial-spectral representation to capture the spatial relationship that is missing in time-series data.

3. Developing parameter optimized CNN models to predict four levels of (extraneous) cognitive load.

CHAPTER 2. REVIEW OF LITERATURE

Predicting cognitive load from EEG signals using machine learning involves modeling the relationship between features extracted from the signal and different mental load levels. Connectivity, entropy, and power spectral density features are the most reported EEG features used in predicting cognitive load Quatieri et al. (2016); Friedman et al. (2019); Amin et al. (2015); Zarjam et al. (2013); Trejo et al. (2015); Kumar and Kumar (2016). Among these features, power spectral density (PSD) features are the most admired due to the relationship between different frequency bands and WM activities at different cognitive load states, as reported in various literature Kumar and Kumar (2016); Dai et al. (2017); Clark et al. (2004); Tort et al. (2009). Classical machine learning models such as SVM, K-nearest neighbors (KNN), and Random forest have been extensively applied to PSD features for cognitive load prediction. However, due to the high degree of noise in EEG data, implementing these models involves a tedious feature selection. Also, to avoid the curse of dimensionality, the current state-of-the-art methods limit their WM analysis to band-specific featuresJohannesen et al. (2016); Trejo et al. (2015); Bashivan et al. (2015); Nuamah et al. (2017); Zarjam et al. (2013). Besides being tedious and band-specific, current WM analysis methods ignore spatial information available through EEG electrodes.

The above challenges show the need for a robust method to learn the spatial-spectral features from inherently noisy EEG data. One such method is deep learning. A few works have adopted deep learning for working memory analysis and cognitive load classification. Sahal Saha et al. (2018) used Stacked Denoising Autoencoder (SDAE) and multilayer perceptron to classify cognitive load from EEG signal. They attempted to reduce the noise

by using a denoising autoencoder with only data from four participants, which may not be enough to train deep learning models and achieve a high and reliable classification. An attempt to preserve the EEG signal's spatial-spectral structure was made in Bashivan et al. (2015). Similar to our work, they transformed EEG signals into 2D spatial-spectral images, which were used to classify cognitive load using deep recurrent CNN. While the work achieved a competitive performance, it did not provide a framework for EEG noise reduction. In addition, the data augmentation approach used in the paper by randomly adding noise to the images failed to improve classification performance.

To contribute to the current effort in predicting cognitive load, we propose a data-driven approach (CNN) capable of learning spatial-spectral representations of EEG data and helping avoid tedious hand-crafted feature engineering and classification on a sub-optimal set of features. We also solve the challenges facing the application of machine learning and deep learning to EEG data by (*i*) reducing the noise and augmenting data samples through eigenspace-based bootstrap sampling for ERP calculation followed by GAN, (*ii*) transforming EEG signal into spectral topomaps to maintain spatial-structure of EEG data, and (*iii*) learning the representations with CNN models to predict cognitive load. To the best of our knowledge, the proposed method is the first deep learning-based approach to address the challenges of noise in EEG data and scarcity of data samples while preserving the EEG signal's spatial-spectral structure and achieving high and reliable CL prediction performance.

CHAPTER 3. METHODS AND PROCEDURES

3.1 Data

In our analysis, we used, with permission, auditory WM data utilized in Bashivan (2016). Experimental details and behavior results are given in Bashivan (2016). In brief, a total of 15 participants (female: 8 and male: 7) were engaged in the auditory WM experiment. The continuous EEG signals were recorded with 64 sintered Ag/AgCl electrodes placed around the scalp at standard 10-10 locations (Neuroscan, Quik-cap) with a sampling rate of 500Hz.

During the experiment, participants listened to a series of English characters (SET), each 300ms long with a 700 ms delay in between. After playing SET, there was a 3 seconds delay during which participants were instructed to memorize the characters. After the delay, a test character (TEST) was played, and participants were asked to press one of the two buttons to indicate whether the TEST was among SET or not. This process is summarized in Fig. 3.1. Per request, we received pre-processed auditory task EEG recordings from 11 participants.

Among the three stages of the WM experiment (encoding, maintenance, recall), we focus on the encoding stage, which covers the SET characters' presentation time. In each of the 60 experimental trials, the number of SET characters is 2, 4, 6, and 8. The size of SET reflects the level of cognitive load corresponding to the mental effort put into encoding SET in the memory. Throughout this paper, we label SET sizes 2, 4, 6, and 8 as cognitive load levels 1, 2, 3, and 4, respectively. Therefore, this work's classification task is to

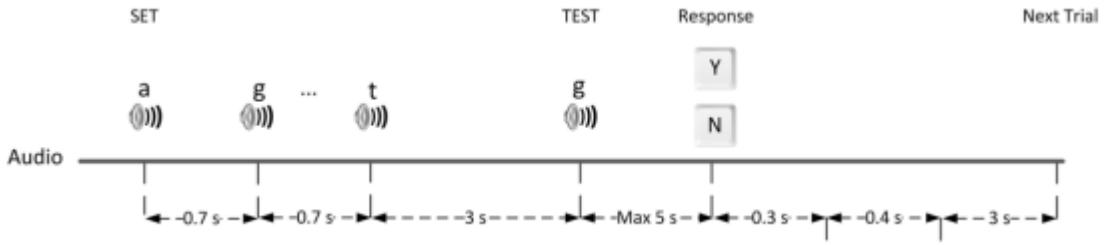


Figure 3.1 Audio task: Series of 300ms English characters were played with 700ms time delay between them. After listening to audio characters, listeners were given 300ms prior to listening to a TEST after Which they pressed a button to indicate whether a TEST character was among SET. Source:Bashivan (2016)

predict these four levels of cognitive load using spatial-spectral features extracted from their corresponding EEG recordings.

3.2 Eigenspace-based Bootstrap Sampling for Noise Reduction and Data Augmentation

Excessive noise in EEG recordings and insufficient samples are the common roadblocks that hinder the adoption of deep learning techniques for EEG data classification tasks. The common technique to reduce noise from single-trial data is by extracting Event-Related Potential(ERP). ERP is computed by averaging a participant's single-trial EEG signals, which removes random brain activities and produces low noise signal Coles and Rugg (1995). However, due to the small number of participants, computing ERP using all trials would not generate enough samples for training deep CNN models. To solve this problem, we obtained our ERP data through bootstrap sampling by averaging 20 single-trial signals selected randomly with replacement from the original 45 trials per task. We repeated this process 2000 times to generate 2000 samples for each participant per one cognitive load level.

However, sampling with replacement may generate redundant data as some trials can be repeatedly selected in multiple iterations. Our objective is to produce samples with significant noise reduction while having enough variance in the generated population. Therefore, we used principal components analysis (PCA) to create eigenspace for our original population (raw EEG data). After making eigenspace, we projected every bootstrap sample to this space and reconstructed it. We then used a distribution of reconstruction errors to pick samples that fall within the quartile range of 25% and 75% hence eliminating samples whose samples belong within the lowest and the highest 25% of reconstruction error distribution. This process dropped 50% of bootstrap samples which are not well represented in the original population's eigenspace. Samples with very low reconstructed as reconsidered redundant since they are very similar to original noisy data. Similarly, samples with very highest reconstruction error should be eliminated as they are far from direction of maximum variance of the original data.

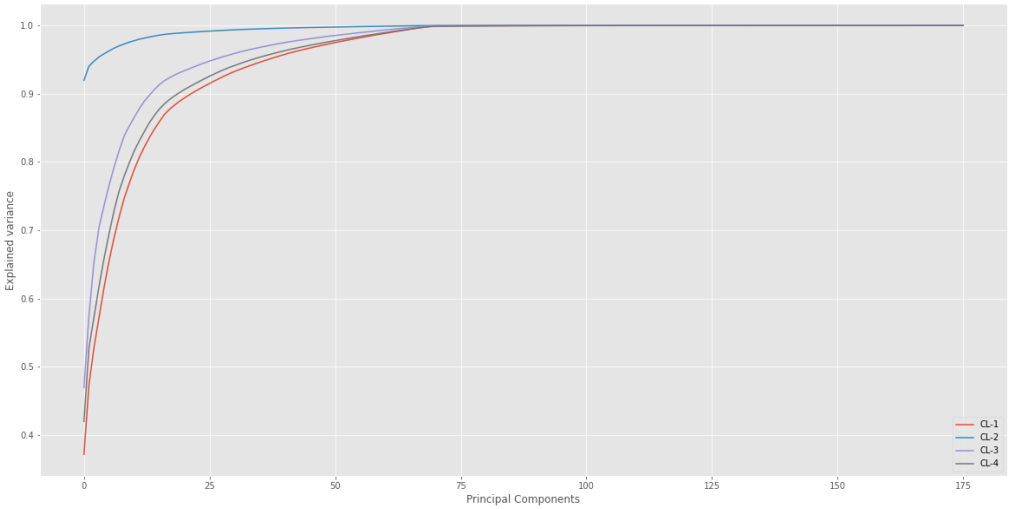


Figure 3.2 Cumulative explained variance ratio for raw EEG data for four cognitive load levels)

3.2.0.1 Principal Components Analysis (PCA) and Eigenspace Creation

Principal components of a data matrix can be obtained by calculating the eigenvectors of the covariance matrix of the data. PCA is widely used for dimension reduction. In our case, we used PCA to find eigenspace that well represents our raw EEG data. Our raw EEG time-series signals have 176 time points. To create the eigenspace, we first selected the number of principal components that capture the direction of maximum variance in the data and then used these components to create our eigenspace. To find the optimal number of principal components (K), we first fit PCA to our data and use the cumulative explained variance ratio to find the first components most of the data variance. Fig.3.2 shows the plot cumulative explained variance ratio against the number of principal components for four levels of cognitive load. The figure shows that the first 75 components cover nearly 100% of data variance. Hence we used $k = 75$ to create our eigenspace to which we projected our bootstrap samples. Example histogram of reconstruction errors is shown in Fig.3.3.

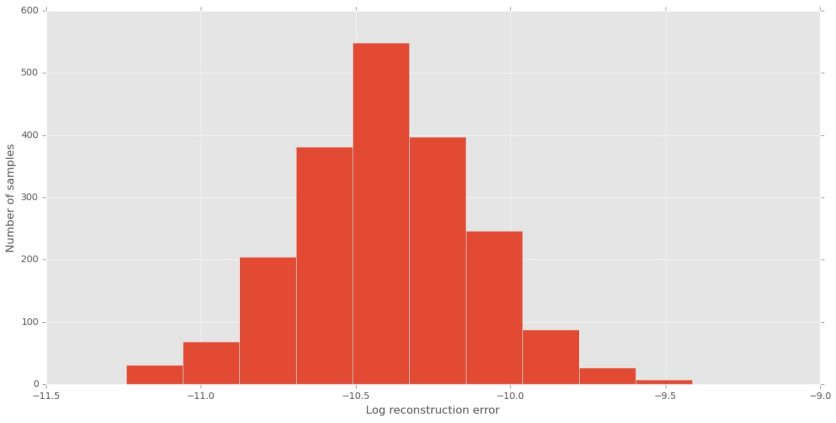


Figure 3.3 Histogram of log reconstruction errors. The reconstruction errors is obtained by subtracting the reconstructed signal from original signal.

After ERP calculation and samples selection using eigenspace, the size of the data reduces from 2000 to 1000 samples per subject per cognitive load level. So, since we have 11 subjects and four CL levels, the total size of the dataset is 44000 images at this point.

3.3 EEG Spatial-spectral Features

Various frequency bands of EEG signals have been linked to the brain's WM processes. Standard five EEG frequencies are Delta (0.5 to 3Hz), Theta (3 to 7.5Hz), Alpha (7.5 to 12.5Hz), Beta (12.5 to 30Hz), and Gamma (> 30Hz). Frequency bands that are reported to be mostly correlated with the increase or decrease of cognitive load are Theta, Alpha, and Beta Dai et al. (2017); Clark et al. (2004); Tort et al. (2009). The traditional representation of EEG spectral features uses feature vectors aggregating mean PSD values from all 64 electrodes. However, this representation ignores the spatial structure of the EEG signal. In this work, we use a spatial-spectral representation of EEG data by transforming the signal into 2D images (topomap), preserving both the spatial and spectral structure of the data. Our objective is to use spatial-spectral features and learning by the convolutional neural network in predicting four cognitive load levels. First, we investigate the predictive power of individual frequency bands. Band specific average PSD were extracted from ERP signal of encoding period by applying FFT to the time series signal using welch method provided by MNE software's time-frequency analysis function (mne.time_frequency.psd_welch()) Gramfort et al. (2014). Since we used 64 channel EEG data, we generated 64 average PSD values for each FFT calculation. Second, we hypothesize that a single frequency band may not contain all WM information necessary to achieve a high and stable CL classification performance. Thus, beyond band-specific topomap, we develop a composite spatial-spectral representation by stacking

together topomap images from Theta, Alpha, and Beta bands. This feature fusion gives a three-channel image with each channel corresponding to an individual frequency band.

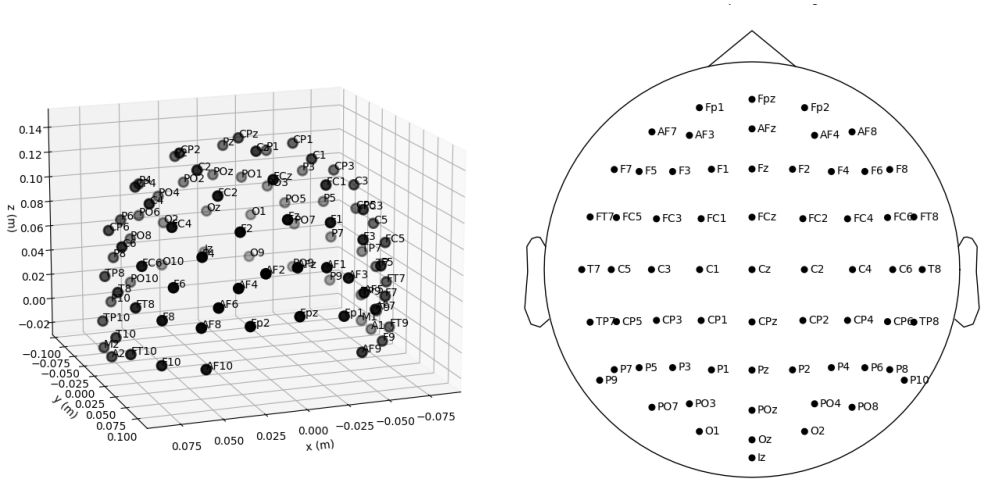
3.4 Topomap Generation

The spatial information available from EEG electrode locations is usually left out in most current state-of-the-art techniques for CL prediction. Our goal is to preserve the data's spatial and spectral structure by projecting PSD values to a 2D montage. A montage is a representation of EEG channel positions in 2D or 3D space. We used MNE software Gramfort et al. (2014) to obtain 2D channel positions by projecting their corresponding positions in 3D space to 2D montage using a sphere as a reference. Fig.3.4 shows 3D (left) and 2D (right) space of 64 EEG channels.

To generate PSD topomap images, we interpolate PSD values over the brain scalp surface montage approximating the values between channel positions. For individual frequency bands, topomaps are obtained by projecting 64 average PSD values for Theta, Alpha, and Beta bands over the scalp surface (MNE function:mne.viz.plot_topomap()). We obtain composite topomap by stacking together topomaps from individual frequency bands. Fig.3.5 summarises the above transformation of ERP signal into topomap images.

3.5 Data Augmentation with GAN

GAN stands for Generative Adversarial Networks, and it was first proposed in 2014 by Goodfellow et al. (2014). GANs belong to the type of unsupervised generative machine learning models such as AutoEncoders (AE)Ng et al. (2011), and Variational Autoencoder (VAE) Pu et al. (2016), and Convolutional Autoencoder (CAE) Chen et al. (2017). Generative models have shown a surprisingly big success in data generation, data encoding, information retrieval, and image super-resolution applications.



(a) 3D channel positions

(b) 2D channel positions

Figure 3.4 64 EEG channel positions (a),(b) in 3D and 2D space respectively(Gramfort et al. (2014))

Autoencoder-based generative models for image generation uses an encoder network to down-sample input image to a smaller dimension latent feature space which a decoder network will learn to regenerate the original image. On the other hand, GAN samples from a complex low dimension noise and learns the transformation from sampled latent space to original distribution. GAN models are composed of two deep neural networks, generator and discriminator, trained simultaneously in a two-player zero-sum game way. The generator network takes a fixed length random latent space vector drawn from a random noise such as Gaussian distribution and generates samples through a generative process. After training, the points in this latent will be mapped to their corresponding points in the real domain (real sample space). The discriminator network takes samples from real data distribution and generator and learns to classify them as real or fake. Therefore, during the training, the discriminator learns to accurately discriminate between actual samples from fake samples generated by the generator. So, the generator tries to fool the discriminator by generating samples as close as possible to real samples. The

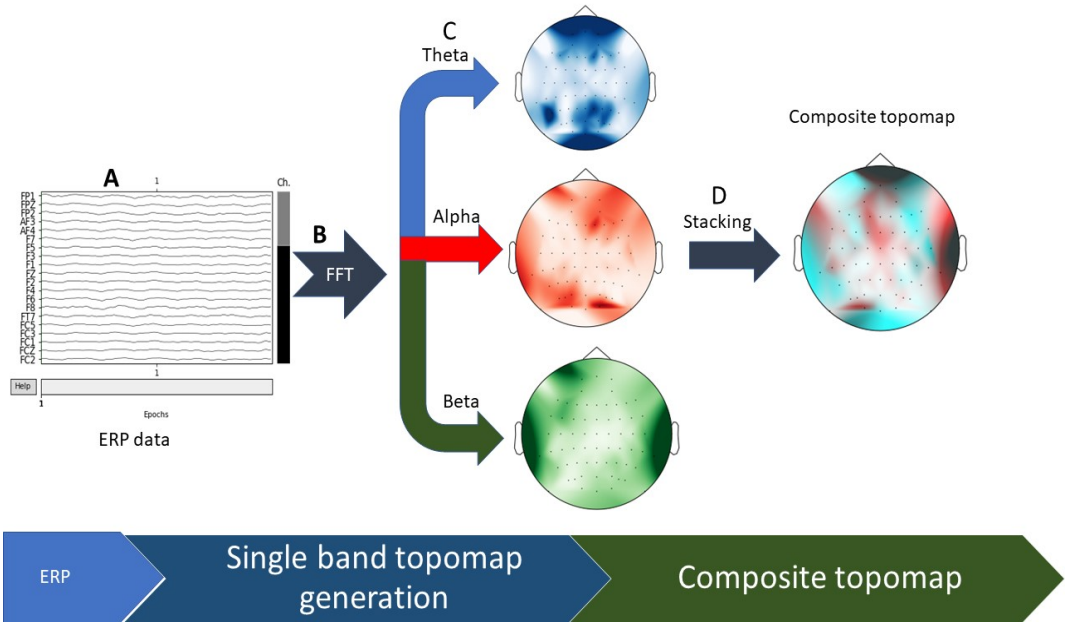


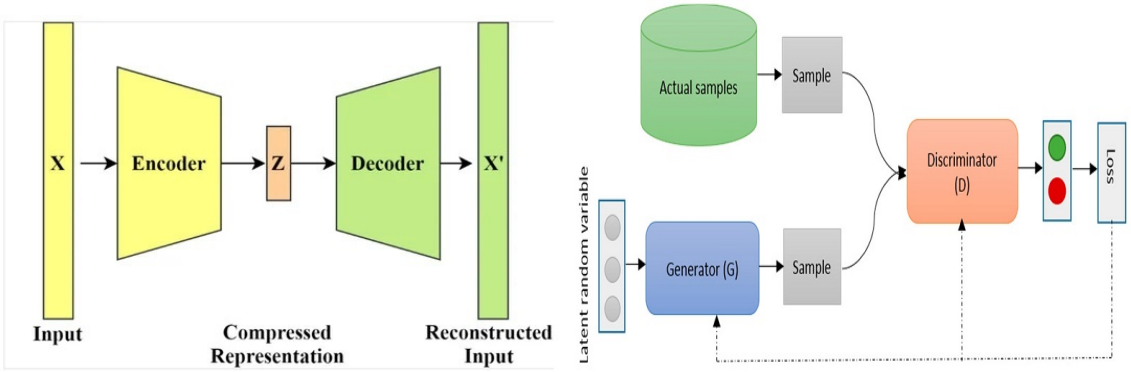
Figure 3.5 Overview of EEG signal to spatial-spectral representation: A) We extract ERP signal from raw EEG data. B) We apply FFT to ERP signal to get average PSD density from three frequency bands. C) We project PSD values over scalp surface to obtain spectral topography maps.D) Single-frequency topomaps are stacked horizontally to form a composite topomap

training continues till when the discriminator can longer distinguish fake from real samples. At the end of the training, we remove the discriminator network and use the generator to generate new samples.

Fig.3.6 shows a general overview and architectural difference between autoencoders (left). and GAN (right).

GAN uses Minimax loss, an optimization used during simultaneous training of generator and discriminator networks. Mathematically, the discriminator tries to maximize the log probabilities assigned to real images $logD(x)$ and log of the inverted probability of generated fake images $log(1 - D(G(Z))$. On the other hand, the generator tries to minimize the log of inverted probabilities assigned fake images $log(1 - D(G(Z))$. Minimax loss optimization for GAN models is summarized below:

- Discriminator: Maximize $\{logD(X) + log(1 - D(G(Z))\}$
- Generator : Minimize $log(1 - D(G(Z))$

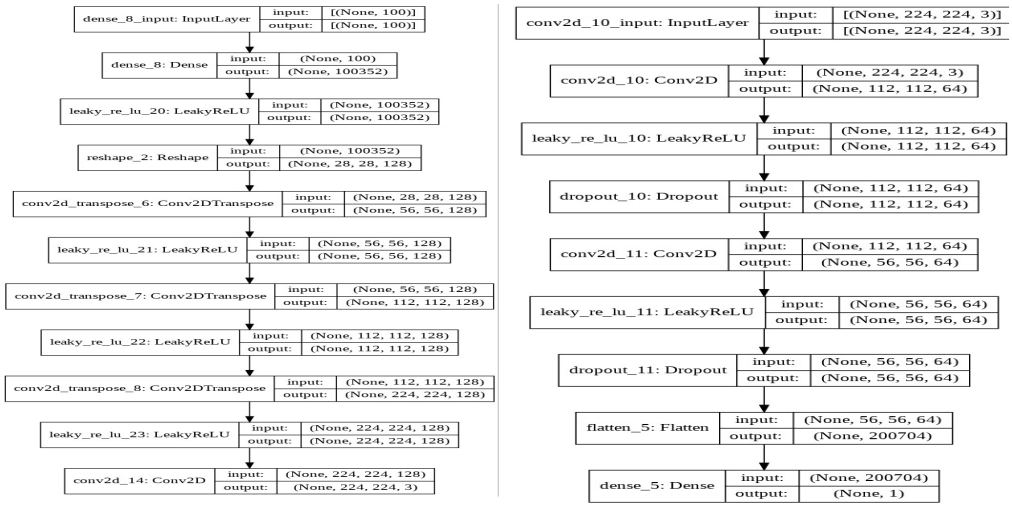


(a) Autoencoder (b) GAN

Figure 3.6 Autoencoder and GAN network overview: (a) Autoencoder down-samples input sample X to produce a compressed representation Z which is then learned by decoder network to reconstruct X'. (b) GAN samples from random latent variable to produce fake examples which will be fed to a discriminator network together with real samples for classification. The error is back-propagated through both networks with generator trying to minimize it (fooling the discriminator)

In this work, we used GAN to increase the number of spatial-spectral topomaps for cognitive load classification. GAN will allow us to further reduce the redundancy in data generated through bootstrap sampling and arguably improve model generalization. To achieve this goal, we randomly selected (without replacement) $22,000(50\%)$ topomap images from 44000 images generated eigenspace-based bootstrap sampling to train GAN. Due to a small number of training samples, we designed a shallow GAN network with a total of $10,941,187$ trainable parameters. The Generator network starts with a 100 long latent vector sampled from a standard normal distribution (mean (μ)=0) and standard deviation (σ) =1) followed by a dense layer, three successive deconvolution layers (upsampling) and one convolution layer with output shape of $224 \times 224 \times 3$ which is the

dimension of real images. The discriminator network consists of two successive convolution and dropout layers, followed by one dense layer and a sigmoid layer. Fig.3.7 Shows the architectures generator and discriminator networks of our GAN model.



(a) Generator network (b) Discriminator network

Figure 3.7 Generator and Discriminator architectures:(a) Generator network takes a 100 latent vector as input and generate a $224 \times 224 \times 3$ image. (b) Discriminator takes a $224 \times 224 \times 3$ images from real data set or Generator images and returns probability of the image being real

We train the GAN model for 100 epochs with a batch size of 32 images. Fig.3.8 shows training loss curves for the GAN model trained on composite topomaps. From the figure, we can see that as training continues, discriminator loss starts to increase, whereas generator loss decreases until both losses become close to each other.

Fig. 3.9 shows example of real image used to train GAN and image generated using the trained GAN. From the figure, we can see that GAN is capable of producing that look like real images.

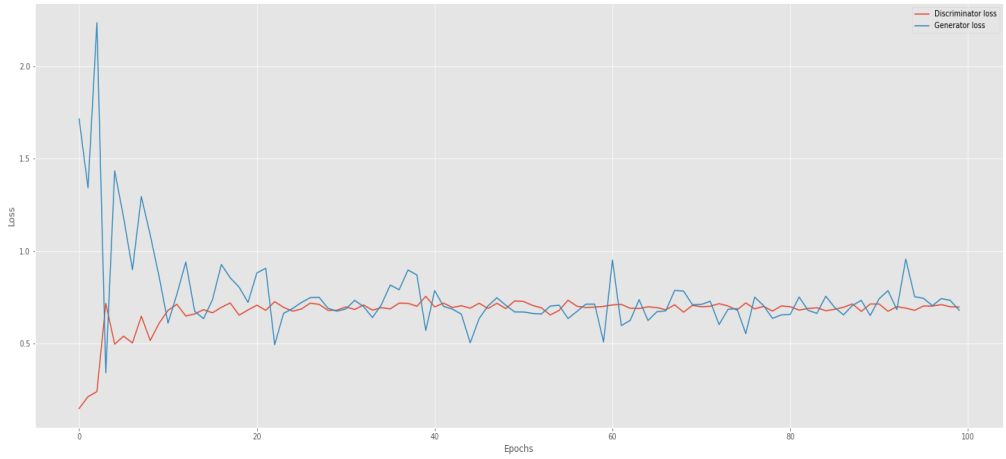
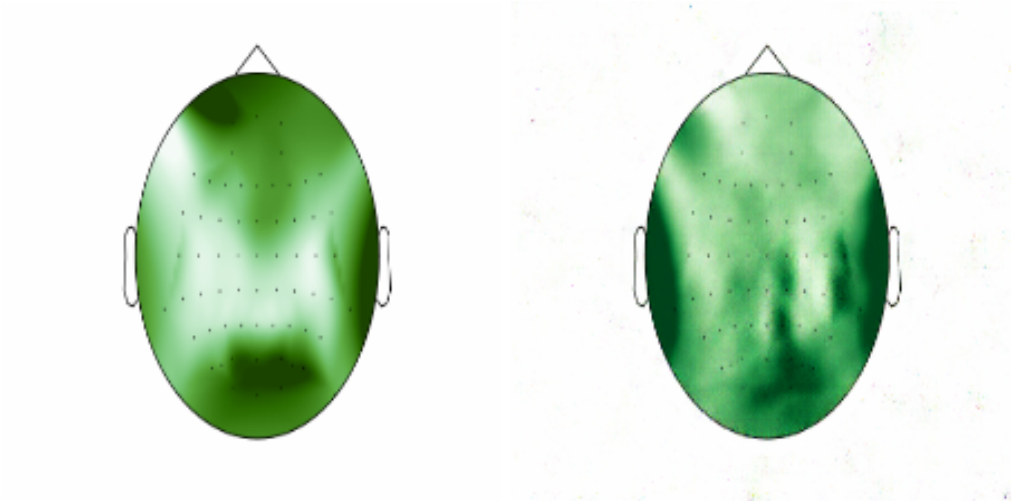


Figure 3.8 Discriminator and Generator Losses. The figures shows how discriminator generator networks losses change for the GAN model trained on stack topomaps (CL-3) for 100

3.6 Cognitive Load Classification with Convolution Neural Networks

The above sections described the data generation workflow, including data augmentation through Eigenspace bootstrap sampling and GAN, and data transformation from time series to spatial-spectral representation. The next step is to use the generated dataset and CNN for cognitive load classification. We used Convolutional Neural Networks (CNN) as our classifier due to its ability to learn spatial features from EEG topomaps. To evaluate the efficacy of our data generation technique performed empirical evaluation of the CNN model performance on data set generated though bootstrapping, and GAN. In both cases, we first test the role of the individual frequency band in predicting cognitive load, and second we compare the performance of individual band and composite topomaps.



(a) Real topomap (b) Fake Topomap

Figure 3.9 Example image generated with GAN:(a) example image from real dataset. (b) Example image generated using GAN

3.6.1 Model Selection and Hyper-parameter Optimization

All CNN models were designed using Keras API, which uses Tensorflow as a backend. To ensure maximum performance of our CNN models, we have implemented hyper-parameter optimization using Hyperopt. Hyperopt is an open-source python library for Bayesian hyper-parameter optimization Bergstra et al. (2013). Traditional non-bayesian optimization techniques such as random search or grid search choose hyper-parameters for the model by exhaustive search through the entire search space. Optimization methods such as random search are not suitable for learning models with many hyper-parameters or hyper-parameters with a continuous range. Unlike random search, Hyperopt utilizes the search history from previous trials to suggest the best hyper-parameters for the next trial using Tree-structured Parzen Estimator (TPE) Bergstra et al. (2011). TPE starts optimization by finding pairs of hyper-parameters x and corresponding losses y from previous trials; then it splits the probability $p(x/y)$ into two distributions $l(x)$ and $g(x)$ where $l(x)$ is associated with a subset of hyper-parameters with smallest loss values and

$g(x)$ for the rest. TPE aims at maximizing expected improvement EI given by the equation:

$$EI_y^* = \int_{-\infty}^{+\infty} (y^* - y) p(x/y)\, dy$$

which leads to

$$EI_y^* = \int_{-\infty}^{+\infty} (y^* - y) \frac{p(x/y)}{p(x)}\, dy$$

, where y^* is expected performance. The authors prove from the above equation that EI is proportional to $(\frac{g(x)}{l(x)})^{-1}$. Therefore, throughout successive optimization trials, TPE finds set of hyper-parameters values x that minimize the ratio $\frac{g(x)}{l(x)}$ or $x^* = argmin_x \frac{g(x)}{l(x)}$

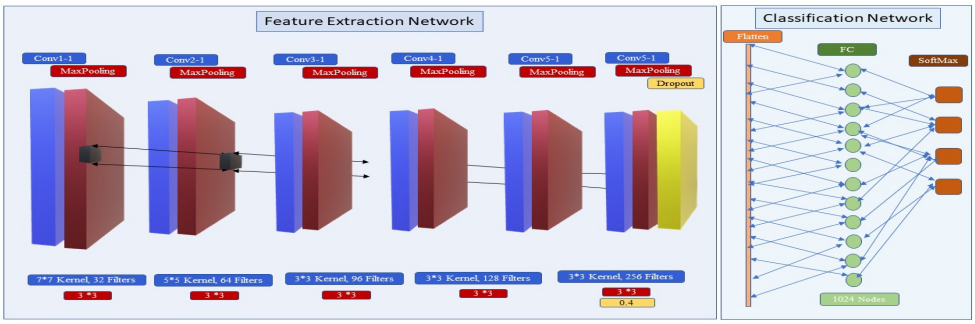


Figure 3.10 CNN architecture for Eigenspace-based stacked band dataset. Feature extraction network consists of six Convolution and MaxPooling sets followed by a dropout layer , and classification network consists of Flatten layer, a single Dense layer with 1024 nodes and a SoftMax layer with 4 nodes

We used Hyperport to find both the best hyper-parameters and the architecture of our models. The search space for the models includes activation function, convolution layers dropout probability, convolution hidden layers probability, convolution kernel size, fully connected units multiplier, optimizer, pooling type, and kernel size for the residual network (if any).

Fig.3.10 shows the architecture of a model built using hyperopt. The architecture of the CNN model shown in Fig.3.10 consists of six Convolution layers as a feature extractor, and

a classification network made of one dense layer and softmax layer. Each Convolution layer is followed by the batch normalization, Relu, and MaxPooling layer. The kernel size of all Convolution layers is 3×3 except for the first layer (7×7) and the second (5×5). The number of filters in the Convolution layer started at 32, doubled the number of filters for each successive layer, and stopped at 256 for the last two Convolution layers. For the classification network, we used a single fully connected layer with 1024 nodes and a SoftMax dense layer to learn the decision surface to separate different levels of topomap images. The Softmax layer has an output shape of four, representing the four cognitive load classes.

We applied an L2-norm weight regularization penalty of 0.01 to the last three Convolution layers to account for over-fitting. Also, we used a single dropout layer with a dropout ratio of 0.4 before the Soft-Max layer.

All CNN models were trained to minimize a categorical cross-entropy loss function using a decreasing learning rate criterion starting from 0.0001 with a decreasing factor of 0.2 for 100 epochs. To train the CNN architectures, we have randomly split the available images using a 70% - 30% split ratio for training and testing, respectively.

CHAPTER 4. RESULTS

In Chapter 3, we described our approach for noise reduction, data augmentation, transformation of the signal into spatial-spectral representations, and modeling of a CNN architecture capable of learning these representations. The main objective of going through the above process was to find a data-drive approach robust against the noise and capable of learning both spatial and spectral information of EEG data to reach a reliable prediction of extraneous cognitive load. In this section, we show the performance of our deep CNN models trained on spatial-spectral features from individual frequency bands and composite topomaps.

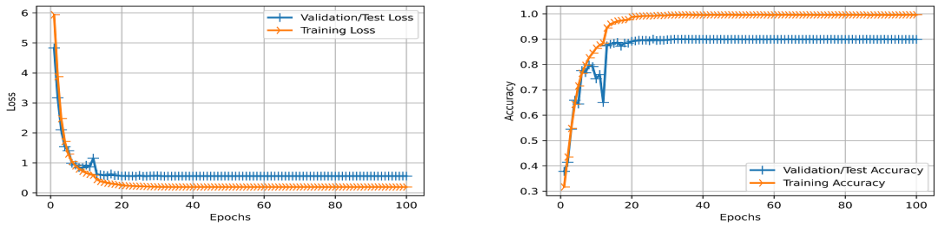
The performance metrics used in evaluating the models are accuracy, precision, recall, weighted average F1-score, and and Area Under the Receiver Operating Characteristic Curve (ROC AUC). Accuracy is the most used evaluation metric for supervised machine learning classifiers. Precision, recall, and F1-score are also useful in determining the effect of false positive and false negative results on model performance, especially when these two quantities are very high or there is an uneven number of samples among classes. Our results show how the above metrics values change from one frequency band to another, reflecting their respective predictive power. We used data generated through two methodologies, namely bootstrap sampling and bootstrap sampling combined with GAN. Sections.4.1 and 4.2 summarize cognitive load classification performance on dataset from methods

4.1 CNN Models' CL classification Performance on Eigenspace-based Bootstrap Sampling Dataset

First, we trained the deep CNN model on topomap images from three frequency bands (Theta, Alpha, Beta) obtained through a transformation from ERP time series to spatial-spectral representation. ERP data were obtained through the bootstrap sampling process described in the section. 3.2. Second, we trained the model on composite topomap images resulting from the fusion of individual band topomaps. For each band and composite representation, we have 44000 images from four different cognitive load levels with 11000 images(samples) each. To train the CNN architecture, we randomly split the available images using a 60% - 25%-15% split ratio for training, testing, and validation.The validation set is important because our images come from bootstrap sampling, which may in rare cases generate duplicate images in both train and test sets, which can lead to a biased model performance. We trained the models for 100 epochs with different batch sizes. Fig.4.2 shows the performance of our CNN models in predicting four levels of cognitive load.

Fig.4.1 shows accuracy and loss curves (learning curves) for the CNN model trained on composite topomap. The learning curves are very useful in evaluating the learning process of the model from one epoch to another. The curves help us to see whether the model is underfitting or overfitting the data. From our learning curves, we can see a good fit of the model to dataset since training and validation loss curves gradually and smoothly decrease and reach a stable point with a minimal gap between them (generalization gap). Similarly, we see a smooth and gradual increase of training and validation curves reaching stable and steady optimal points after 60 epochs. We trained the model for 100 epochs to ensure that the training process does not end immaturely. Here we show the learning curves for

composite since it gave the best model performance. Training curves for Theta, Alpha, and Beta bands are shown in APPENDIX.



(a) Loss curve for stack band topomap (b) Accuracy curve for stack band representation

Figure 4.1 Training and validation curves for CNN model trained on Stacked band topomaps. The model was trained over 100 epoch with a batch size of 32 images

Fig.4.2 shows cognitive load prediction performance in terms of accuracy, precision, recall, F1-score, and Area Under the Receiver Operating Characteristic Curve (ROC AUC). From the figure, we can deduce that the Beta band carries more cognitive load predictive power than Theta and Alpha since it outperformed the other two bands in predicting four cognitive levels with an accuracy of 88%. Further, we did not find a significant difference between alpha and theta bands' predictive power as the performance of CNN models trained on their corresponding topomaps is approximately the same with the accuracy of 86% and 85% respectively. Though we achieved a good classification performance from individual frequency bands, combining the individual bands improved the performance significantly up to the accuracy of 90%.

In Fig.4.3, we show the confusion matrix of the CNN model trained on Theta, Alpha, Beta, and composite representation. Looking at the confusion matrices, we can see that for all bands, our model has more misclassifications at the upper intermediate cognitive load level (CL-3), which could result from high signal variations during the transition from low to high cognitive loads. On the other hand, the CNN models easily discriminated

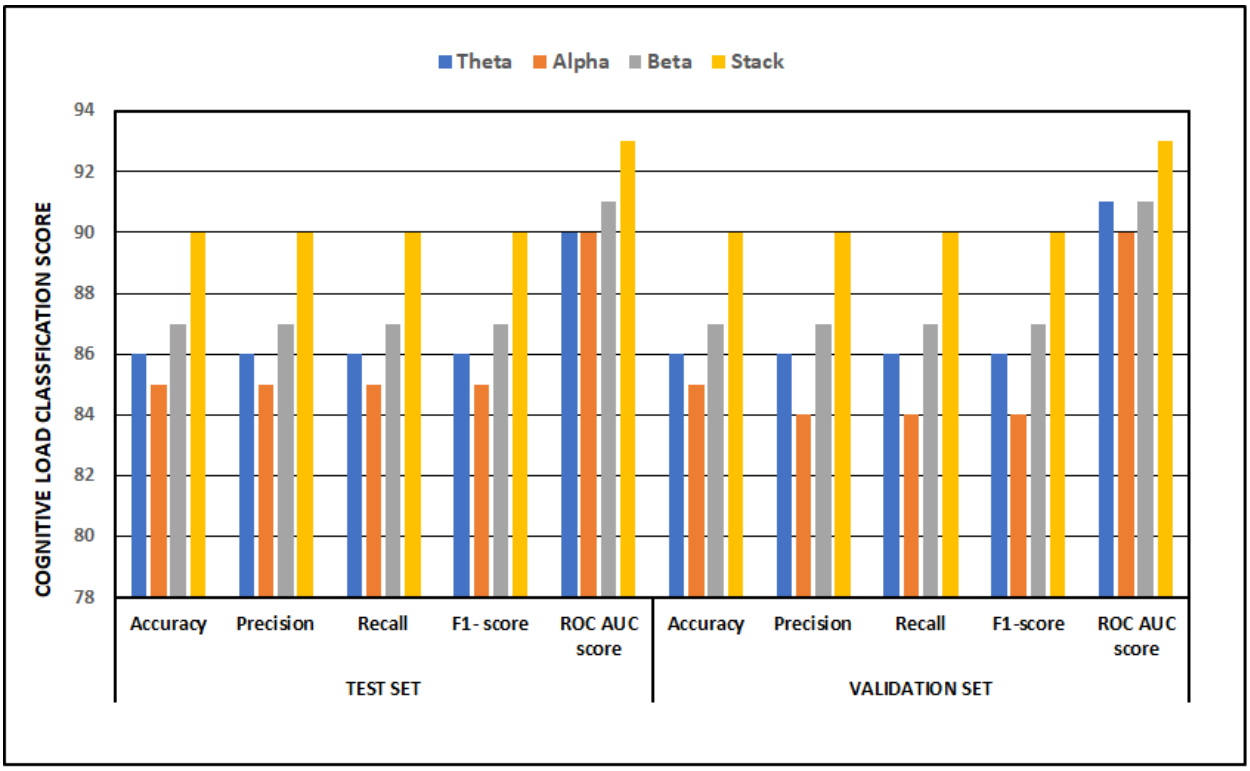
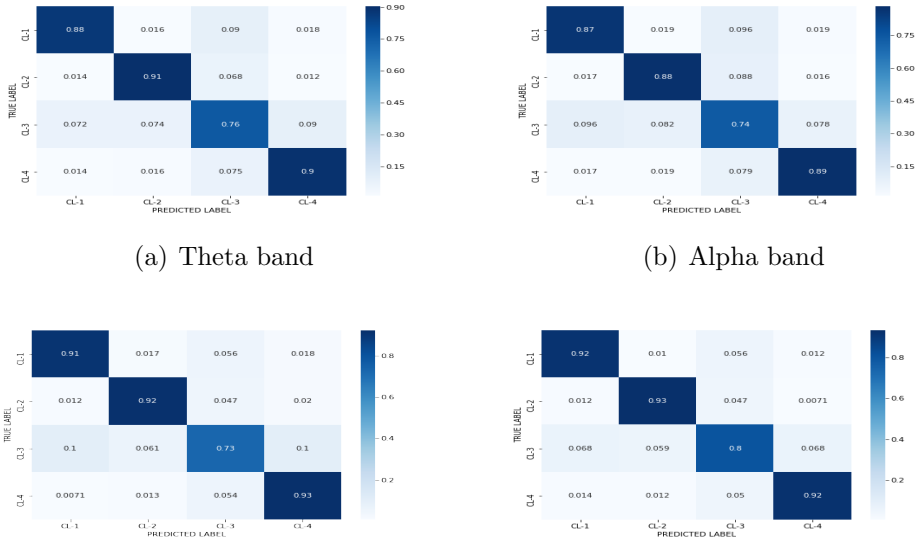


Figure 4.2 Cognitive load classification results with CNN model: Accuracy, precision, recall, and F-1 score were used to evaluate the model model on spatial-spectral topomap from theta, alpha, beta bands, and composite topomap in predicting four levels of cognitive load

other loads: load-1(SET size =2), load-2(SET size =4), and load-4(SET size = 8), which shows the steadiness of EEG signals at low and high memory loads.

4.2 CNN models' CL classification Performance on Combined Bootstrap Sampling and GAN Dataset

In this section, we report cognitive load classification performance from CNN models trained on topomap data from a combination of samples from the bootstrap process and GAN, see section .3.5 for details. A half of data used here came from bootstrap sampling, and another half was generated using GAN. Similar to Section. 4.1, we trained the deep



(a) Theta band

(b) Alpha band

(c) Beta band

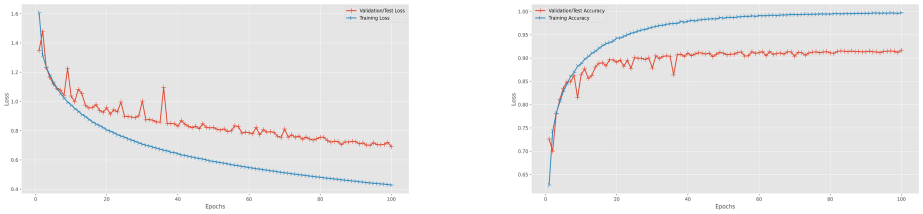
(d) Composite topomap

Figure 4.3 Confusion matrices of CNN models trained on spectral topomap from theta (a), alpha (b), beta (c) bands, and composite topomap (d)

CNN model on topomap images from Theta, Alpha, Beta frequency bands, and composite representation. The goal here is to test whether adding GAN to our data generation framework significantly improves classification performance and generalization. We followed the same training procedure described in Section.4.1.

The models were trained for 100 epochs to minimize categorical cross entropy. Fig. 4.4 shows training and validation curves (loss and accuracy) for the CNN model trained on stacked bands representation. From the figure, we see fluctuations in the curves for the first few epochs followed by a more smooth training after 60 epochs. The rough training at the beginning can be caused by the slight difference between real and GAN generated images which the model overcomes as training continues.

Fig.4.5 summarizes the classification performance of CNN models trained on real data combined with GAN generated topomaps. The figure shows performance improvement of 3%,4%, 2%, and 2% for theta, alpha, beta, and stack topomaps respectively. These results

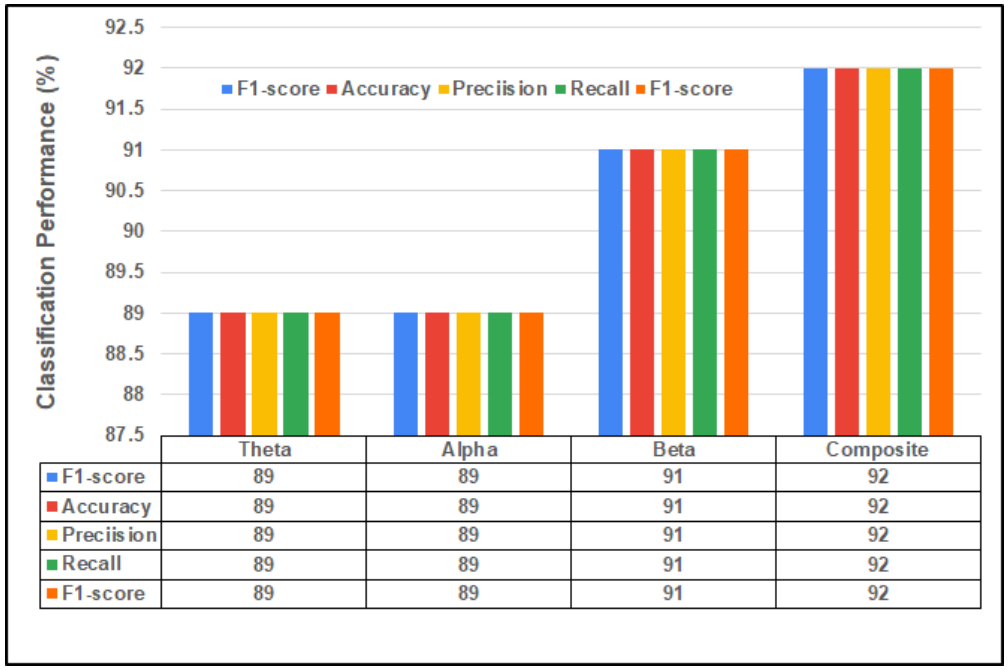


(a) Loss curve for stack band topomap (b) Accuracy curve for stack band representation

Figure 4.4 Training and validation curves for CNN model trained on Stacked band topomaps from real and GAN topomaps. The model was trained over 100 epoch with a batch size of 32 images

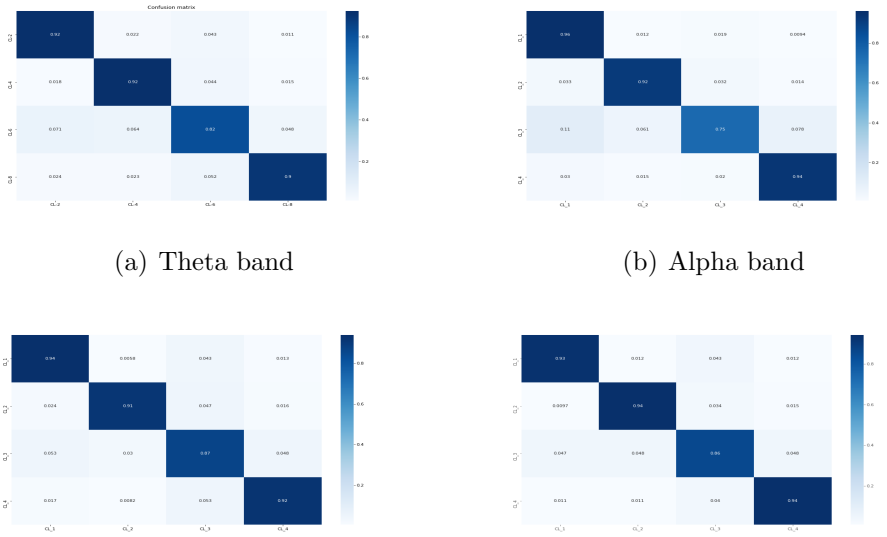
proves the robustness of GAN in generating reliable data for image classification. Looking at confusion matrices in Fig. 4.6 we observe similar trend as in Fig. 4.3 where models have difficulty in discriminating the third cognitive load level (CL-3) from other levels. In terms of role of frequency bands in predicting cognitive load, Beta band has the highest performance compared to other bands and composite representation shows the highest performance.

Overall, our results show that without tedious hand-crafted feature selection, and while preserving the spatial-spectral structure of EEG data, the CNN model can learn the mapping between EEG signal and different levels of cognitive load with high accuracy. Moreover, we can reach higher performance beyond individual frequency bands by training the CNN model on the composite band representation. Further, we can solve the small sample problem in EEG through bootstrap sampling and GAN, enabling us to model deep neural network models and improve model generalization.



	Theta	Alpha	Beta	Composite
F1-score	89	89	91	92
Accuracy	89	89	91	92
Preciision	89	89	91	92
Recall	89	89	91	92
F1-score	89	89	91	92

Figure 4.5 Cognitive load classification results with CNN model: Accuracy, precision, recall, and F-1 score were used to evaluate the model model on spatial-spectral topomap images generated through GAN and bootstrap sampling in predicting four levels of cognitive load



(a) Theta band

(b) Alpha band

(c) Beta band

(d) Composite topomap

Figure 4.6 Confusion matrices of CNN models trained on spectral topomap from theta (a), alpha (b), beta (c) bands, and composite topomap (d) generated through GAN and bootstrap sampling

CHAPTER 5. SUMMARY AND DISCUSSION

This work presented a data-driven approach to learn the spatial-spectral representation of EEG signals recorded from participants performing an auditory working memory experiment. CNN models were used to learn and map the EEG representations to four levels of CL corresponding to the complexity introduced into WM tasks. Data augmentation using eigenspace-based bootstrap sampling in computing the ERP to reduce the noise followed by GAN allowed us to obtain enough and low noise data to find bias-variance trade-off in building CNN models. Also, the transformation of ERP data into spectral topomap images allowed us to preserve the spatial-spectral structure of EEG signals. Further, we used deep CNN architecture to alleviate the tedious feature extraction and selection commonly used in classical machine learning.

We conducted empirical analyses to compare the predictive power of individual frequency bands (theta, alpha, beta) and composite representation in classifying cognitive load. Our results show that the Beta band(12.5 Hz - 30 Hz) has more predictive power than Theta (3 - 7.5 Hz) and Alpha (7.5 - 12.5 Hz) in classifying cognitive load with accuracy $> 91\%$. The classification accuracy is $> 92\%$ for combinations of 3-frequency bands. Our results also suggest that generative models such as GAN show potential success in solving the small sample problem in EEG analysis.

Our results, however, still rely on the average of trials (ERP) for cognitive load classification. ERP calculation can eliminate useful information from a single trial signal due to averaging. In order to achieve a high classification performance, future work should consider other denoising and data techniques that work on single trials.

Overall, our results show that the combination of data transformation, data augmentation, and use of CNN models are highly accurate ($> 92\%$) in predicting CL and robust against typical noise present in the EEG recordings.

BIBLIOGRAPHY

Amin, H. U., Malik, A. S., Ahmad, R. F., Badruddin, N., Kamel, N., Hussain, M., and Chooi, W.-T. (2015). Feature extraction and classification for eeg signals using wavelet transform and machine learning techniques. *Australasian physical & engineering sciences in medicine*, 38(1):139–149.

Baddeley, A. (1992). Working memory. *Science*, 255(5044):556–559.

Bashivan, P. (2016). *Commonality and singularity in working memory network predicting performance and individual differences*. The University of Memphis.

Bashivan, P., Rish, I., Yeasin, M., and Codella, N. (2015). Learning representations from eeg with deep recurrent-convolutional neural networks. *arXiv preprint arXiv:1511.06448*.

Bashivan, P., Yeasin, M., and Bidelman, G. M. (2015). Single trial prediction of normal and excessive cognitive load through eeg feature fusion. In *2015 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, pages 1–5.

Bergstra, J., Bardenet, R., Bengio, Y., and Kégl, B. (2011). Algorithms for hyper-parameter optimization. In *25th annual conference on neural information processing systems (NIPS 2011)*, volume 24. Neural Information Processing Systems Foundation.

Bergstra, J., Yamins, D., and Cox, D. (2013). Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *International conference on machine learning*, pages 115–123. PMLR.

Chen, M., Shi, X., Zhang, Y., Wu, D., and Guizani, M. (2017). Deep features learning for medical image analysis with convolutional autoencoder neural network. *IEEE Transactions on Big Data*.

Chen, S., Epps, J., Ruiz, N., and Chen, F. (2011). Eye activity as a measure of human mental effort in hci. In *Proceedings of the 16th international conference on Intelligent user interfaces*, pages 315–318.

Choi, H.-H., Van Merriënboer, J. J., and Paas, F. (2014). Effects of the physical environment on cognitive load and learning: towards a new model of cognitive load. *Educational Psychology Review*, 26(2):225–244.

Clark, C. R., Veltmeyer, M. D., Hamilton, R. J., Simms, E., Paul, R., Hermens, D., and Gordon, E. (2004). Spontaneous alpha peak frequency predicts working memory performance across the age span. *International Journal of Psychophysiology*, 53(1):1–9.

Coles, M. G. and Rugg, M. D. (1995). *Event-related brain potentials: An introduction.* Oxford University Press.

Dai, Z., De Souza, J., Lim, J., Ho, P. M., Chen, Y., Li, J., Thakor, N., Bezerianos, A., and Sun, Y. (2017). Eeg cortical connectivity analysis of working memory reveals topological reorganization in theta and alpha bands. *Frontiers in human neuroscience*, 11:237.

Friedman, N., Fekete, T., Gal, Y. K., and Shriki, O. (2019). Eeg-based prediction of cognitive load in intelligence tests. *Frontiers in human neuroscience*, 13:191.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, 27.

Gramfort, A., Luessi, M., Larson, E., Engemann, D. A., Strohmeier, D., Brodbeck, C., Parkkonen, L., and Hämäläinen, M. S. (2014). Mne software for processing meg and eeg data. *Neuroimage*, 86:446–460.

Hick, W. E. (1952). On the rate of gain of information. *Quarterly Journal of experimental psychology*, 4(1):11–26.

Hollender, N., Hofmann, C., Deneke, M., and Schmitz, B. (2010). Integrating cognitive load theory and concepts of human–computer interaction. *Computers in human behavior*, 26(6):1278–1288.

Johannesen, J. K., Bi, J., Jiang, R., Kenney, J. G., and Chen, C.-M. A. (2016). Machine learning identification of eeg features predicting working memory performance in schizophrenia and healthy adults. *Neuropsychiatric electrophysiology*, 2(1):3.

Junghöfer, M., Peyk, P., Flaisch, T., and Schupp, H. T. (2006). Neuroimaging methods in affective neuroscience: Selected methodological issues. *Progress in brain research*, 156:123–143.

Kumar, N. and Kumar, J. (2016). Measurement of cognitive load in hci systems using eeg power spectrum: an experimental study. *Procedia Computer Science*, 84:70–78.

Liang, S. F., Lin, C. T., Wu, R. C., Chen, Y. C., Huang, T. Y., and Jung, T. P. (2005). Monitoring driver's alertness based on the driving performance estimation and the eeg

power spectrum analysis. In *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, pages 5738–5741.

Liu, Y. and Liu, Q. (2017). Convolutional neural networks with large-margin softmax loss function for cognitive load recognition. In *2017 36th Chinese control conference (CCC)*, pages 4045–4049. IEEE.

Mazher, M., Abd Aziz, A., Malik, A. S., and Ullah Amin, H. (2017). An eeg-based cognitive load assessment in multimedia learning using feature extraction and partial directed coherence. *IEEE Access*, 5:14819–14829.

Nasrallah, I. and Dubroff, J. (2013). An overview of pet neuroimaging. In *Seminars in nuclear medicine*, volume 43, pages 449–461. Elsevier.

Ng, A. et al. (2011). Sparse autoencoder. *CS294A Lecture notes*, 72(2011):1–19.

Niederhauser, D. S., Reynolds, R. E., Salmen, D. J., and Skolmoski, P. (2000). The influence of cognitive load on learning from hypertext. *Journal of educational computing research*, 23(3):237–255.

Nuamah, J. K., Seong, Y., and Yi, S. (2017). Electroencephalography (eeg) classification of cognitive tasks based on task engagement index. In *2017 IEEE Conference on Cognitive and Computational Aspects of Situation Management (CogSIMA)*, pages 1–6. IEEE.

Ozkan, N. F. and Kahya, E. (2018). Classification of bci users based on cognition. *Computational Intelligence and Neuroscience*, 2018.

Paas, F., Tuovinen, J. E., Tabbers, H., and Van Gerven, P. W. (2003). Cognitive load measurement as a means to advance cognitive load theory. *Educational psychologist*, 38(1):63–71.

Paas, F., Van Gog, T., and Sweller, J. (2010). Cognitive load theory: New conceptualizations, specifications, and integrated research perspectives. *Educational psychology review*, 22(2):115–121.

Pu, Y., Gan, Z., Henao, R., Yuan, X., Li, C., Stevens, A., and Carin, L. (2016). Variational autoencoder for deep learning of images, labels and captions. *Advances in neural information processing systems*, 29:2352–2360.

Puce, A. and Hämäläinen, M. S. (2017). A review of issues related to data acquisition and analysis in eeg/meg studies. *Brain sciences*, 7(6):58.

Quatieri, T. F., Williamson, J. R., Smalt, C. J., Perricone, J., Helfer, B. J., Nolan, M. A., Eddy, M., and Moran, J. (2016). Using eeg to discriminate cognitive workload and performance based on neural activation and connectivity. Technical report, MIT Lincoln Laboratory Lexington United States.

Roy, R. N., Bonnet, S., Charbonnier, S., and Campagne, A. (2013). Mental fatigue and working memory load estimation: interaction and implications for eeg-based passive bci. In *2013 35th annual international conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 6607–6610. IEEE.

Saha, A., Minz, V., Bonela, S., Sreeja, S., Chowdhury, R., and Samanta, D. (2018). Classification of eeg signals for cognitive load estimation using deep learning architectures. In *International Conference on Intelligent Human Computer Interaction*, pages 59–68. Springer.

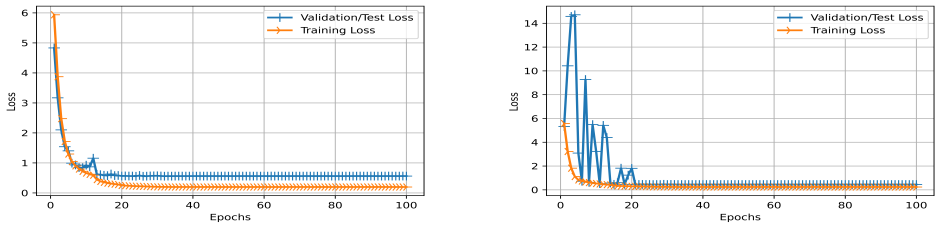
Sweller, J., Van Merrienboer, J. J., and Paas, F. G. (1998). Cognitive architecture and instructional design. *Educational psychology review*, 10(3):251–296.

Tort, A. B., Komorowski, R. W., Manns, J. R., Kopell, N. J., and Eichenbaum, H. (2009). Theta–gamma coupling increases during the learning of item–context associations. *Proceedings of the National Academy of Sciences*, 106(49):20942–20947.

Trejo, L. J., Kubitz, K., Rosipal, R., Kochavi, R. L., Montgomery, L. D., et al. (2015). Eeg-based estimation and classification of mental fatigue. *Psychology*, 6(05):572.

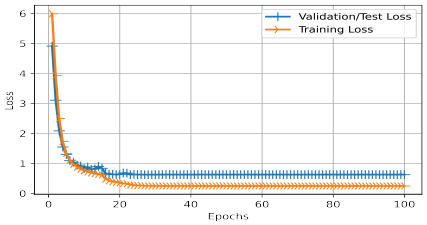
Zarjam, P., Epps, J., Chen, F., and Lovell, N. H. (2013). Estimating cognitive workload using wavelet entropy-based features during an arithmetic task. *Computers in biology and medicine*, 43(12):2186–2195.

APPENDIX A. Accuracy and Loss Curves

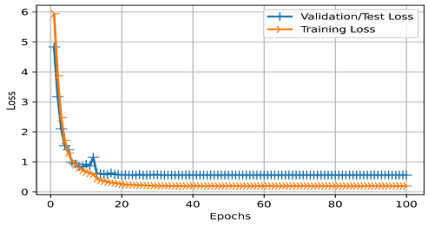


(a) Theta band

(b) Theta band

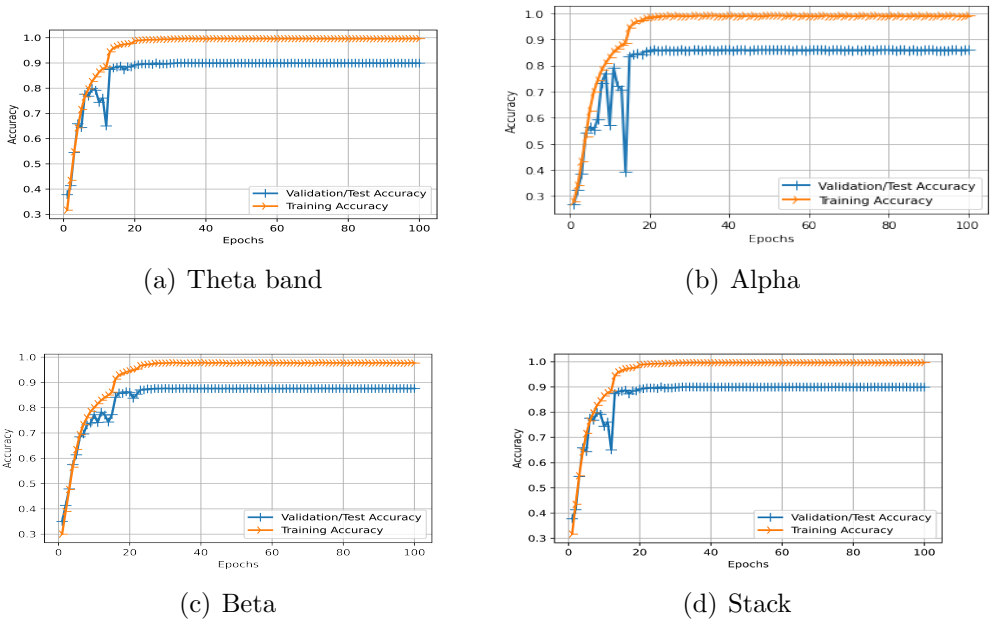


(c) Theta band



(d) Theta band

Figure A.1 Training and validation loss curves for CNN models trained on Theta, Alpha, Beta, Stack spatial-spectral representations of EEG for datasets obtained through Eigenspace-based bootstrap Sampling. The models was trained over 100 epochs.



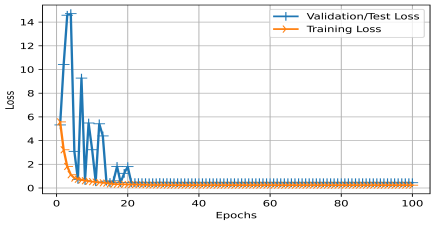
(a) Theta band

(b) Alpha

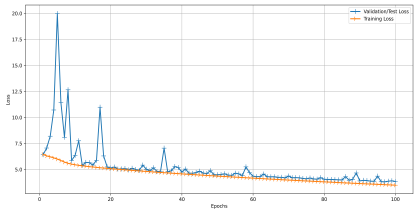
(c) Beta

(d) Stack

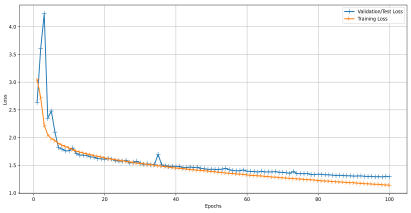
Figure A.2 Training and validation accuracy curves for CNN models trained on Theta, Alpha, Beta, Stack spatial-spectral representations for datasets obtained through Eigenspace-based bootstrap Sampling. The models was trained over 100 epochs



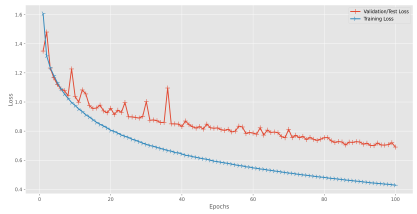
(a) Theta band



(b) Alpha

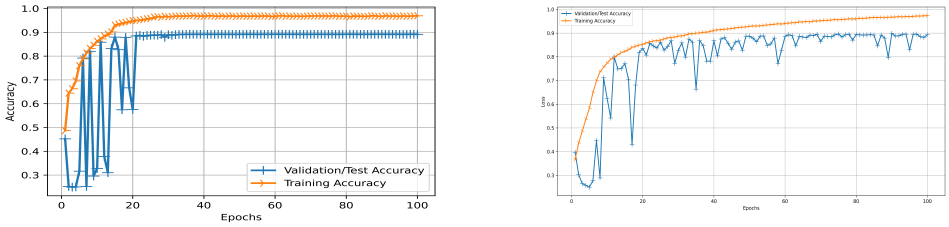


(c) Beta



(d) Stack

Figure A.3 Training and validation loss curves for CNN models trained on Theta, Alpha, Beta, Stack spatial-spectral representations for datasets obtained through bootstrap Sampling and GAN The models was trained over 100 epochs



(a) Theta band

(b) Alpha

(c) Beta

(d) Stack

Figure A.4 Training and validation accuracy curves for CNN models trained on Theta, Alpha, Beta, Stack spatial-spectral representations for datasets obtained through bootstrap Sampling and GAN The models was trained over 100 epochs