DISTINGUISHING FAKE AND REAL SMILES USING EEG SIGNALS AND DEEP LEARNING

by

Mostafa Mohamed Moussa

A Thesis presented to the Faculty of the
American University of Sharjah
College of Engineering
In Partial Fulfillment
of the Requirements
for the Degree of

Master of Science in
Biomedical Engineering

Sharjah, United Arab Emirates

April 2020

Declaration of Authorship

I declare that this thesis is my own work and, to the best of my knowledge and belief, it does not contain material published or written by a third party, except where permission has been obtained and/or appropriately cited through full and accurate referencing.

Signed…… Mostafa Mohamed Galaleldin Ahmad Moussa ….

Date……02/06/20……

The Author controls copyright for this report.

Material should not be reused without the consent of the author. Due acknowledgement should be made where appropriate.

© Year 2020

Mostafa Mohamed Galaleldin Ahmad Moussa

ALL RIGHTS RESERVED

Approval Signatures

We, the undersigned, approve the Master's Thesis of Mostafa Mohamed Moussa.

Thesis Title: Distinguishing Between Fake and Real Smiles Using EEG Signals and

Date of Defense: 30-Apr-2020

Name, Title and Affiliation	**Signature**
Dr. Usman Tariq Assistant Professor, Department of Electrical Engineering Thesis Advisor	
Dr. Hasan Al Nashash Professor, Department of Electrical Engineering Thesis Co-Advisor	
Dr. Salam Dhou Assistant Professor, Department of Computer Science and Engineering Thesis Committee Member	
Dr. Fares Yahya Visiting Postdoctoral Fellow, Department of Electrical Engineering Thesis Committee Member	
Dr. Hasan Al Nashash Director Biomedical Engineering Graduate Program	
Dr. Lotfi Romdhane Associate Dean for Graduate Studies and Research College of Engineering	
Dr. Sirin Tekinay Dean College of Engineering	
Dr. Mohamed El-Tarhuni Vice Provost for Graduate Studies Office of Graduate Studies	

Acknowledgement

I would like to thank my advisors Dr. Usman Tariq, and Dr. Hasan Al Nashash for providing knowledge, guidance, support, and motivation throughout my research stages. I'm profoundly grateful for their assistance, and their insightful discussions and suggestions.

I would also like to thank the professors of the Biomedical Engineering program who taught me the master level courses with an upright zeal. Their advice and motivation were important to push me forward and are greatly appreciated. Last but not least, I would like to thank my predecessor, Ms. Meera Alex, for I would not be able to come up with this work without her support, and the 28 subjects that participated in the experiment, which would take a lengthy amount of time to prepare them for. I would also like to thank the American University of Sharjah for granting me the graduate assistantship during my study.

Dedication

To my family…

Abstract

Genuineness of smiles is one aspect of the field of deception recognition, one that is prevalent in myriad social situations, and it is not easy to tell when a person's smile is genuine or not for the average person. Machine learning techniques, such as support vector machines or artificial neural networks, can allow better distinction between fake and real smiles by making use of electroencephalograms (EEG) from subjects with a simple experimental protocol, in which the subject's response is known by the experimenters. Machine learning techniques were previously used in affect recognition, though not for distinguishing real and fake smiles through EEG signals. The objective of this study is to distinguish between fake and real smiles using deep learning techniques, more specifically shallow neural networks, convolutional neural networks, and support vector machines (SVMs) as a baseline from EEG signals. The experimental approach involved presenting subjects with visual stimuli and recording their physical response and their EEG, which was used with the aforementioned algorithms. The SVM classifier used the radial basis function kernel, with optimized parameters, the simple neural network was a three-layer pattern recognition network with 150 hidden units using scaled conjugate gradient as the training function, the convolutional neural networks used stochastic gradient descent with a momentum of 0.95 for all the different architectures, and the optimal one was selected based on the results. The accuracies of the simple neural network, convolutional neural network, and SVM are 88.879 %, 90.446 %, and 48.387 % respectively for subject-dependent classification, and the convolutional neural network yielded 53.418 % for subject-independent classification.
Keywords: Electroencephalogram; machine learning; support vector machines; deep learning; artificial neural networks; convolutional neural networks; subject-dependent analysis; subject-independent analysis.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

CNN	Convolutional Neural Network
ECG	Electrocardiogram
ECOC	Error-Correcting Output Codes
EEG	Electroencephalogram
EMG	Electromyogram
EOG	Electrooculogram
fMRI	Functional Magnetic Resonance Imaging
fNIRS	Functional Near Infrared Spectroscopy
kNN	K^{th} Nearest Neighbour
LSTM	Long Short-Term Memory
NN	Neural Network
ReLU	Rectified Linear Units
RNN	Recurrent Neural Network
SVM	Support Vector Machine

Chapter 1. Introduction

Emotions vary in range and intensity among people. Automated methods for emotion recognition are important due to the significance of detecting affect in social situations, in addition to giving further insight into understanding the human brain, among other reasons. Narrowing down emotions represented by facial expressions by decomposing them into simple action units (groups of muscles that move together on faces) can make studying them simpler, to some extent. For example, simplifying anger into brow furrowing, or surprise into widening of the eyes, or sadness into frowning, makes experimentally discerning between discrete emotions easier, and by the same token, happiness can be simplified into smiling. However, some of those actions can be easily faked.

Smiling is one of the simplest acts a person could do to convey joy or satisfaction, among other feelings in social situations. Detecting fake smiles can be generally useful in behavioral studies involving social cues and can pave the way for better classification of discrete emotions. One side benefit of detecting fake smiles is encouraging genuine interaction in the workplace environment, as smiles perceived to be honest tend to instigate trust and subsequently entail better earning opportunities [1].

In this section, a brief overview of the problem is given, including the motivation behind the work and the means with which the work would be done, followed by the thesis contribution, then the general outline of the thesis is presented.

1.1. Overview

Affect deception recognition is generally a difficult task that requires personnel trained in psychology. When studying facial expression for affect recognition, a researcher must always account for the fact that emotions can easily be disguised, whether consciously or subconsciously. Machine learning techniques, such as support vector machines or artificial neural networks, can allow better distinction between faked emotions and real ones by making use of electroencephalograms (EEG) from subjects with a simple experimental protocol, in which the subject's response is known by the experimenters. The EEG contains features that can help identify patterns in the subjects' behaviors reflected via brain waves.

Emotions, in this case, the authenticity of a smile, can be quantified through somewhat subjective methods such as expert observation and analysis, or through more objective, quantitative methods.

Experiments for affect/emotion recognition often involve the use of facial expressions, electroencephalogram readings, and sometimes eye-tracking mechanisms (for visual stimuli, naturally) and functional magnetic resonance imaging. These techniques are used in a setup where the subjects are exposed to stimuli (visual, auditory, or both), and their responses are recorded by cameras, EEG, and eye-tracking software/fMRI if applicable [2, 3]. Machine learning can be used in addition to the techniques stated previously to utilize their results to give a definitive class to each emotion presented.

1.2. Problem Formulation

Many problems warrant the need for better means of deception recognition, such as depression, lie detection, and integrity of psychological studies.

Depression affects many around the world, and they usually handle that illness in silence, due to the stigma surrounding mental illness in many communities, or other individual factors. It may not be easy for most to seek necessary help with depression, so some external way to detect it can be a catalyst to getting them the help they may need.

Moreover, lie detection is more of an art than an exact science [4]. Polygraphs use physiological measures such as blood pressure, heart rate, and skin conductivity as an indicator of whether a subject is lying or not. These measures are variable to the extent that they can be unreliable sometimes, and can often be masked, albeit with some training. Not to mention the need for an expert who can read the results and give accurate conclusions based on them. Moreover, the method of reading them has no scientific basis [4].

Another issue arises in masking emotions in psychological studies, due to fear of embarrassment or change in moral self-image [5]. This, of course, requires a method to detect deception to main the integrity of the study, and even undertake reliable studies concerning deception in this field.

After analyzing the relevant works, and considering the problems associated with EEG in general, we concluded that distinguishing smiles was to be done in two "phases". The first phase comprised of 28 subjects performing an exercise in which we showed them stimuli that elicited a genuine response and they would press a button on the keyboard. The subjects would fake a smile and press a different button upon seeing the image of a book and press the third button if the displayed image elicited a neutral feeling or no response. The results of this experiment were to be taken mainly in the form of EEG readings. Eye-tracking data and some facial expression data are available, but they will not be utilized in our analysis. These results would be processed to clean the signals and for optimal manual feature extraction, and to exclude subjects whose data was not suitable (17 subjects remain), and put through k^{th} nearest neighbor (KNN) and support vector machines (SVMs) [6].

The second phase involved using deep learning techniques, primarily support vector machines, shallow neural networks, and convolution neural networks, with long-short-term-memory neural networks under consideration. For the neural networks, we propose four-fold cross-validation, where the data is divided into randomly selected folds, three for training and one for testing from the remaining events after minor pre-processing for each subject. Four-fold cross-validation also entails performing this split four times, randomly or manually changing the blocks. Deep neural networks allow the use of all 28 subjects, as they significantly simplify processing "by allowing automatic end-to-end learning of preprocessing, feature extraction and classification modules," while providing fairly accurate results [7].

1.3. Thesis Objectives

Due to the aforementioned need for a reliable means of classifying smiles, and the growing interest in deep learning in the field of affect recognition, we propose the utilization of artificial neural networks, as opposed to other machine learning techniques, such as kNN or SVM. The main objective of this thesis is to implement neural network architectures and concepts that have been used for other applications before with EEG data obtained from the dataset to be described with its complementary protocol, in order to distinguish between real and fake smiles. We will focus on using

simple neural networks and convolutional neural networks for this purpose, as we predict they would yield the most optimal results, from what we saw in [8-11].

1.4. Research Contribution

The contribution of this thesis is summarized below:

Contribution to the field of study:

- Propose neural network architectures and algorithms to distinguish between real and fake smiles in place of other machine learning algorithms to apply to the dataset obtained. This novel dataset includes 28 subjects, for which EEG data was obtained following the protocol described later.
- The proposed networks achieve better accuracy in contrast with machine learning algorithms, namely SVM when classifying smiles using EEG exclusively, and allow the omission of some steps in pre-processing.
- The proposed networks forego the need for feature engineering.
- The proposed networks are used for both subject-dependent and subject-independent analysis and are compared.
- The proposed networks can also use time-frequency information obtained via wavelet transform of the EEG data for classification in the future.

Contribution to society:

- Provide a more automated method of detecting depression.
- Encourage honest social and professional interaction.
- Provide a building block to pave the way for methods to detect deceit in lie detector tests and psychological studies.

1.5. Thesis Organization

The remainder of the thesis proceeds as follows: the next chapter, Chapter 2 provides background information and includes the literature review regarding the human brain, electroencephalography, and machine learning algorithms. Chapter 3 presents the experimental and machine learning setups used to obtain the data and Chapter 4 explains the methodology that is applied using these setups. Finally, Chapter 5 includes the discussion of the results, concludes the thesis, and describes future work.

Chapter 2. Background and Literature Review

This chapter gives the readers some needed background to follow the later material. Furthermore, we reviewed some works related to emotion recognition and their relationship with EEG.

2.1. Background

2.1.1. The human brain. In this work, our focus is on brain signals. Hence, it is a good idea to give readers some background in the human brain. The human brain is the main organ of the nervous system and makes up the central nervous system along with the spinal cord. The brain is split into three parts, namely the cerebrum, cerebellum, and the brainstem, all three of which contain different structures that have different functions that the body requires, such as temperature regulation and sensory relay by the brainstem. However, our focus in this research will be the cerebrum, as it includes the cerebral cortex, which in turn, is divided into the four main lobes: frontal, parietal, temporal, and occipital [12].

Each of these lobes is responsible for different base functions, the frontal and occipital lobes, for example, are responsible for cognitive skills and short-term memory, and vision, respectively. Whereas the temporal lobe is responsible for olfactory and auditory senses and complex visual stimuli, and the parietal lobe is responsible for integrating sensory information. Furthermore, one important feature of the cerebral cortex is its proximity to the cranium, and by extension the scalp, which is where EEG electrodes record electrical activations. The EEG reads these activations as micro-voltage, so a means of reducing the impedance between the scalp and the electrodes is generally always necessary. Hair, dust, and sweat can accumulate and increase the impedance, as they hinder the connection between the scalp and the electrodes. Therefore, gels are often used to simply complete the circuit from the scalp to the electrode and correctly measure brain activity. Some electrodes are placed nearby other brain structures, such as the upper area of the cerebellum as a reference (near the subjects' ears).

Figure 2.1 shows the four main lobes of the human brain and Figure 2.2 shows the structure of the brain and some functions of the cerebral cortex [13].

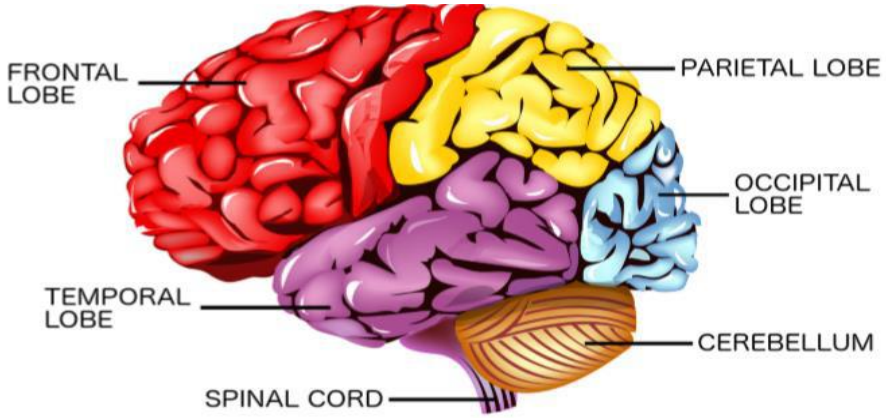


Figure 2.1: Main lobes of the cerebrum [14].

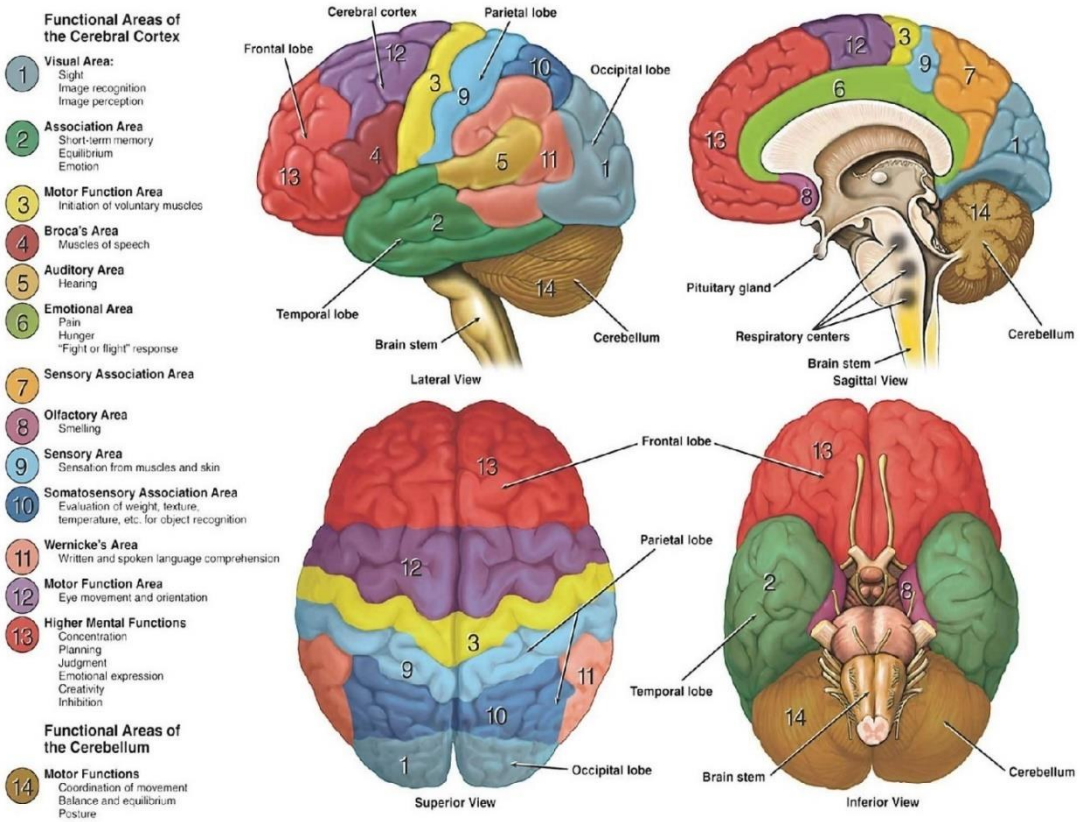


Figure 2.2: Anatomy and functions of the brain [13].

The signals observed from the scalp, or brain waves, are usually split into five different bands; delta (0.5-4 Hz), theta (4-8 Hz), alpha (8-12 Hz), beta (12 – 35 Hz),

and gamma (>35 Hz). Though the frequency ranges are not agreed upon in literature, the cut-offs and bandwidths are similar with only 1 Hz or so differences [15]. Each band is more active in certain activities. Delta waves are associated with deep sleep, theta waves are associated with sleep and dreaming, alpha waves are associated with relaxed brain states, beta waves are associated with normal activity and focused mental activity, and gamma waves are associated with brain hyperactivity [16]. Most of these activities are relevant to the experimental protocol, and as such, are used in the analysis, save for gamma waves. Figure 2.3 shows these five brain waves.

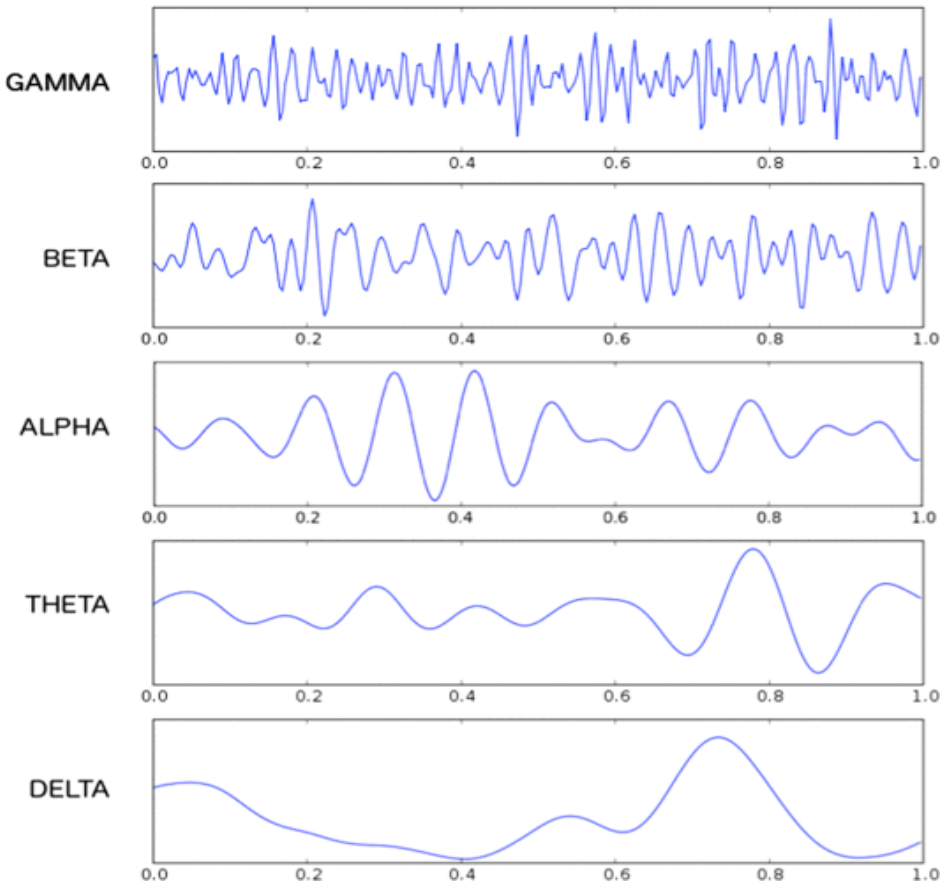


Figure 2.3: The five brain waves [14].

The brain is still vastly enigmatic, but we do understand that stimuli yield responses from a healthy brain. We also understand that, as stated earlier, smiling is a

measure of happiness. Emotional brain activity is normally associated with the prefrontal cortex [14], hence an emphasis on the electrodes in that general area.

2.1.2. Electroencephalography. Electroencephalography or EEG is a method of reading brain activity that relies on electrical sources in the brain to obtain data.

It belongs to a class of methods that use other electrical activations to read biometric information, such as heart rhythm (electrocardiography), eye movement (electrooculography), muscle contraction and relaxation (electromyography), and as such, involves placing electrodes (usually Ag/AgCl) on the subject's scalp, from which data is read. Consequently, the data is recorded as voltage waveforms, as shown in Figure 2.4.

EEG signals are more random, and have lower amplitude, in general, as opposed to ECG, EOG, or EMG which rely mostly on predictable or directable muscle movement. This entails the need for more extensive pre-processing than ECG, EOG, and to a lesser extent than those two, EMG, to obtain the desired information from the EEG.

Figure 2.4 also showcases an important feature of EEG, and electrical activation reading methods in general, and that is event triggers. The figure shows epochs of data that begin a certain amount of time before the event, and end a certain amount after, which makes analyzing the EEG easier and more meaningful.

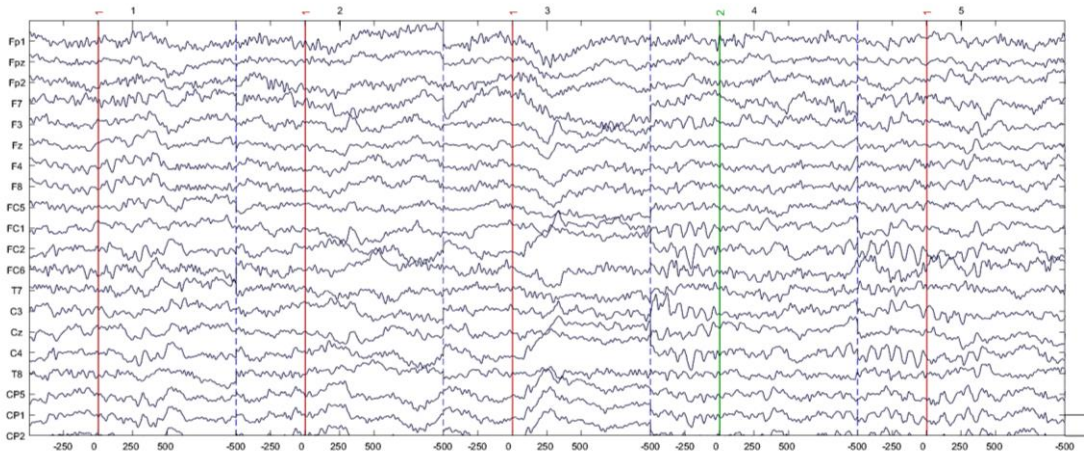


Figure 2.4: 19-channel EEG showing 5 epochs of data.

Since it retains information on brain activity, EEG data can be used to read how the brain acts when the person smiles genuinely or not and can thus be used to translate that into a somewhat definitive real/fake answer. An EEG cannot be faked as easily as a smile or another facial expression, if at all possible.

There are challenges with using EEG with machine learning, such as the possibility of low signal-to-noise ratio (SNR) if pre-processing is not done correctly [17], the non-stationary nature of EEG signals, and subject variability. The latter is more problematic in global, inter-subject schemes [7]. Some recently developed machine learning algorithms that make up deep learning have the potential to bypass these challenges, while also reducing the factor of human error, due to "automatic" feature learning and extraction.

2.1.3. Machine learning. We have a supervised machine learning problem at hand in this work. We hope to differentiate between acted and actual smiles using EEG signals. We use three machine learning algorithms in this work; support vector machines (SVMs), Artificial Neural Networks (ANNs) and Convolutional Neural Networks (CNNs). This work involves having an experimentally acquired dataset, in which the condition corresponding to a class is known, that is used as the training input and output. In this case, it is known at which trial or image the subjects' smile was real, fake, or if they did not smile at all.

This aids a machine learning algorithm in classifying; when a training set is available, it has a precedent to learn from. In this particular case, which EEG signal epoch belongs to which of these three classes. This work is concerned with three algorithms; SVMs, simple artificial neural networks, and convolutional neural networks.

A support vector machine, or a kernel machine, is a supervised learning, kernel-based, linear algorithm that can be applied to numerous classification and regression applications through the use of kernel functions [18]. A support vector machine is meant to find an optimal hyperplane that separates the different classes, making that hyperplane the decision boundary. The points closest to the hyperplane are called support vectors and are used to maximize the margin and build the model. The dimensionality of the hyperplane depends on the number of features available; if two

classes are available then it is a 2-D problem and the hyperplane is simply a line. Figure 2.5 shows an example of a two-class problem.

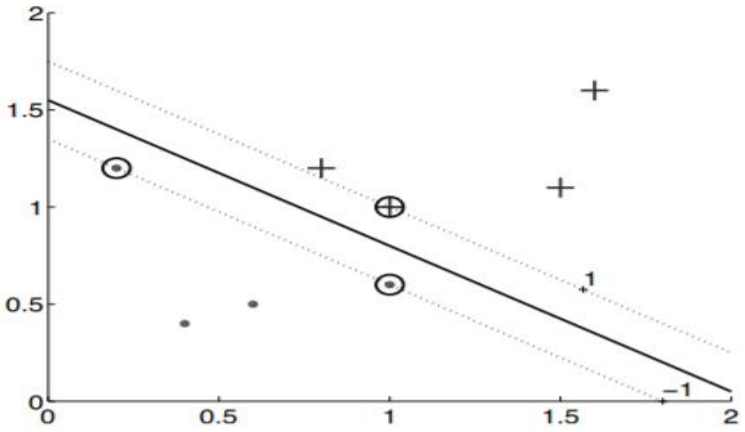


Figure 2.5: A two-class problem, where the classes are pluses and dots. Circled points are less than the margin (shown by dashed lines) away from the line, and are called support vectors [18].

In some cases, the data is not linearly separable, and consequently, a separating hyperplane is not possible. Hence, a hyperplane is selected such that the classification error is minimal. This means that a plus can be on the side where the dot class is. This is still a misclassification, but it helps build the model. Another way to solve nonlinear problems involves making them linear in another space; this is done by mapping them to a new space using a nonlinear transformation that depends on the basis function.

Support vector machines have various kernel functions that perform differently for different problems. The three most popular kernel functions are polynomial functions, Gaussian kernel/radial basis functions (RBF), and sigmoid functions [18].

Polynomial functions are represented by (1), RBF by (2), and sigmoid by (3), where K is the matrix of kernel values or Gram matrix, q is the polynomial order, D is some distance function, and s is a spread value [18].

$$K(x^t, x) = (x^T x^t + 1)^q \quad (1)$$

$$K(x^t, x) = \exp\left(-\frac{D(x^t, x)}{2s^2}\right) \quad (2)$$

$$K(x^t, x) = \tanh(2x^T x^t + 1) \quad (3)$$

The main kernel function we are concerned with is RBF, as it generally fits more situations than polynomial kernel and is more commonly used compared to sigmoid kernel. For optimal classification results, the hyperparameters of the algorithm need to be optimized. These hyperparameters include the kernel scale, also known as gamma or sigma, and box constraint for radial basis functions. The kernel scale is the scale factor that is multiplied by the distance function and the box constraint is a parameter that controls the penalty on data points that violate the margin.

SVMs are usually used for binary classification but can also be used for multi-class problems by dividing them into binary problems. Binary problems can be made in two ways; using a one-vs-one approach in which a model is fit for every two classes, or using a one-vs-all approach, in which a model is fit for every class against all of them. Both approaches require binary classification.

SVMs are simple to use and remain one of the most popular machine learning techniques to date. However, they present one drawback that can be time-consuming in many research areas, and that is feature extraction and engineering. Feature engineering is not a difficult process, but it can be tedious. This is because not all features could perform well or are as simple to extract and process, so it is also a matter of optimization.

Essentially, an artificial neural network functions in a manner similar to the human brain; it takes in an input, processes it through hidden “neuronal” units, and gives out an output.

However, it is not entirely known how the brain processes information; we know that through impulse transmissions between complex neuronal networks, the brain gives an output, such as sensory perception, for input like visual signals through the eyes or mechanical vibrations through the ear, and that is where it is similar to artificial neural networks. Artificial neural networks learn from the input by extracting features and minimizing error in the output, updating network parameters in the process.

Figure 2.6 shows how an artificial neural network generally looks like; it has an input layer, as many hidden layers as its designer desires, and an output layer whose size equals the number of classes.

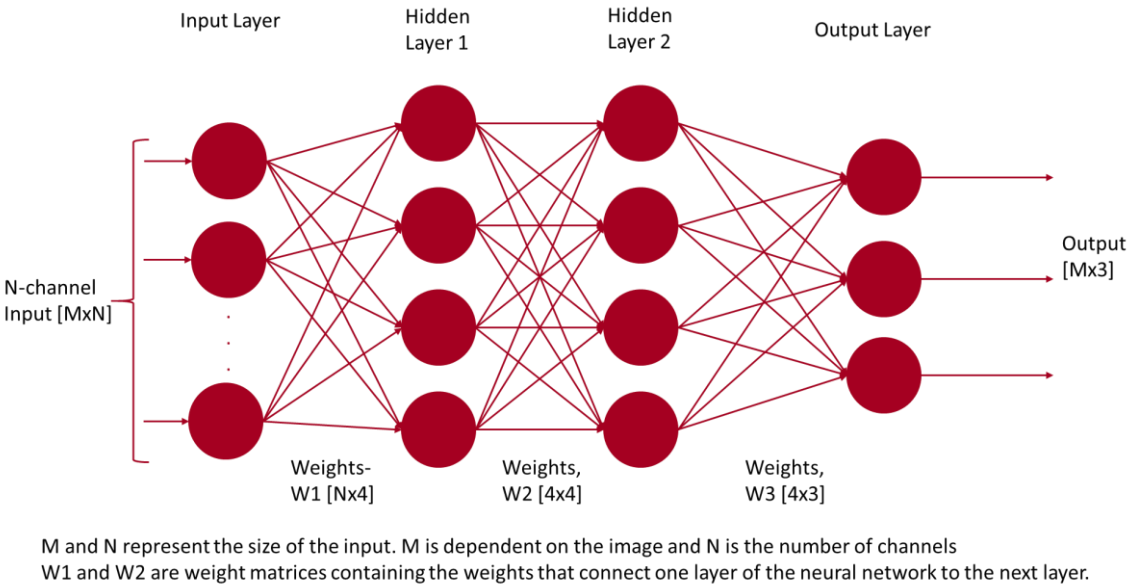


Figure 2.6: General architecture of an artificial neural network.

Artificial neural networks generally depend on the use of a backpropagation algorithm for learning. Backpropagation is simply a gradient descent algorithm that works to minimize a cost function by feeding forward, estimating the error, and propagating backward to reduce that error. The cost function is application dependent. Famous cost functions are mean squared error for regression and cross-entropy for classification.

Input goes through the input layer, which feeds into the hidden layers where the activation functions are applied, and the weights are constantly updated to minimize the cost function of the backpropagation algorithm. These optimal weights are the essence of the trained model. Finally, the result of classification or regression is output through the output layer.

Artificial neural networks differ from SVMs in hyperparameter optimization, as well. Since a neural network does most of the work, the user need not optimize

hyperparameters on their own, or automatically prior to training with the dataset. However, other parameters arise because of this, such as the activation functions, number of hidden layers, number of hidden units in each layer, mini-batch size for batchwise training, training function, and any parameters associated with the training function. Training functions include scaled conjugate gradient, stochastic gradient descent, and stochastic gradient descent with momentum (the momentum is an example of a training function parameter). These parameters apply to convolutional neural networks as well, in addition to filter number, filter size, and convolution strides.

These parameters are defined by the user, though most are usually set to a default value that generalizes well by software tools such as MATLAB. The activation function is applied to the weights, the number of which is defined by the number of hidden layers and hidden units. The mini-batch size represents the number of instances used for training and validation at a time.

CNNs are artificial neural networks that deal primarily with images. Many computer vision applications utilize them, CNNs can also be used with bio-electric measurements, such as EEG, ECG, EMG, EOG, etc.

Figure 2.7 shows the architecture of a typical convolutional neural network. Its fundamental layer is a convolutional layer, and as such it performs convolution. It takes in data like images or reshaped EEG data and convolves this data with the defined filter banks. This results in several "feature maps" which are the result of the convolution of the input with the defined filters. Afterward, pooling is done to reduce the size of the feature maps, in order to keep the computations in check and to learn more complex features. This can be done by max-pooling or average pooling. Pooling works to downsample feature maps by summarizing feature presence. The use of pooling, however, is not a requirement for a convolutional neural network, as we show later on. Moreover, a layer must be added for the activation function to reduce nonlinearity, which can be rectified linear units (ReLU), sigmoid, among a few other activation functions.

A batch normalization layer may also be added to normalize the input across a mini-batch by re-scaling and re-centering it. This speeds up CNN training, decreases sensitivity to network initialization, and has a regularization effect; it adds some noise

to the activations. Following that, a fully connected layer is added to combine information across several feature maps. One advantage deep learning has over machine learning is that the user does not need to extract or even know what features to use; the network learns and does that automatically. The latter three layers can be duplicated and stacked with some alterations to the layers; this is not necessary but can be helpful in some cases. The final fully connected layer must have as many units as the classes defined by the input, no more and no less. In code, a SoftMax layer and a classification layer are added to classify. A SoftMax layer applies a SoftMax function, which is an activation function that turns numbers into probabilities with a sum of one. Simply put, its output is a vector denoting the probability distribution of potential outcomes.

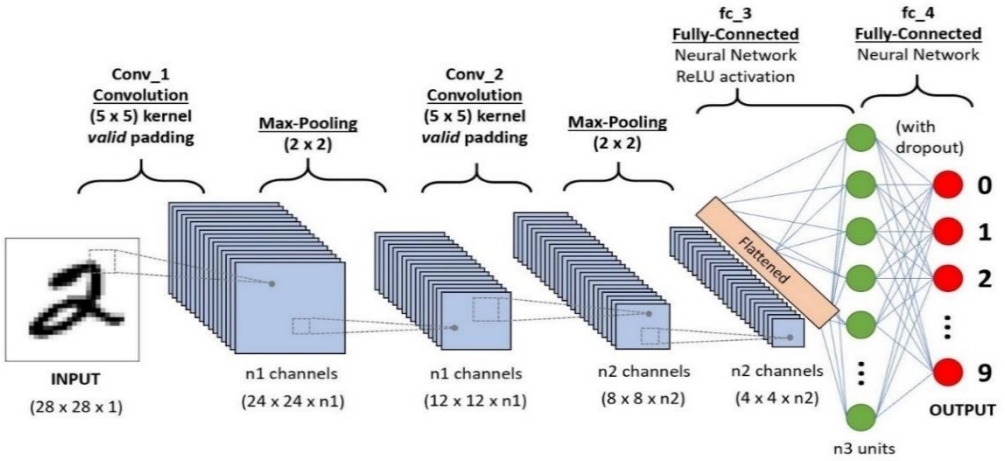


Figure 2.7: Typical convolutional neural network architecture [19].

The network in Figure 2.7, for example, has an input layer of size 28x28x1, two convolution layers with 5x5 convolutions, two 2x2 max-pooling layers, one ReLU layer, two fully connected layers, one with n3 units, as we can see underneath the green-colored units, and with 9 units for digit recognition, in addition to a flatten layer preceding it to collapse the dimensions of the input.

2.2. Literature Review

Generally, emotion recognition is a subjective field that requires extensive human involvement. Though people can innately and routinely perceive and analyze

emotions in social settings, observation requires more quantitative measures to produce reliable results.

Scientists collect data for human emotion recognition by labeling images and videos with the perceived emotion category. Furthermore, many times scientists record people watching stimuli that are designed to elicit an emotional response. The recorded data is then labeled with the emotion category using self-reports and/or expert labeling. There may be various modalities through which the felt emotion of a person may be detected. These may include facial expressions, speech, or even body gestures. Recently there has been some interest in measuring emotions using brain signals, such as EEG.

Before moving on to studies involving EEG, studies that focus on using facial expressions are investigated. Zhang *et al.* [20] investigated the use of involuntary facial expressions to detect deceit in a system that represents expressions as facial Action Units (AUs).

Smiles can be both socially-motivated, as Crivelli *et al*'s study involving observing judo competitors has concluded [21], or spontaneous as evidenced by one study by Kawakami *et al*. [22]. This study involved observing five infants for over 30 hours and simply counting the number of spontaneous smiles and laughs. Results show seven spontaneous smiles and one laugh in the time of observation, meaning smiles are not just social utensils but can occur spontaneously.

Affect in general, can be defined in terms of valence, arousal, and dominance, the latter of which is generally disregarded for affect recognition studies [8-10, 23, 24]. Valence is a measure of attractiveness or repulsion, and scales from 'sad' (1) to 'happy' (9), and arousal is a measure of activity and scales from 'bored' (1) to 'alert' (9) [25].

These ranges could include most of the six basic emotions described by Ekman and Friesen [26] (shown in Figure 2.8). One study explores how accurately people can identify emotions through ambiguous facial expressions using event-related brain potentials (ERPs) and showing the participants discrete facial expressions (angry, happy, fearful, and neutral) and blended ones (angry-happy, fearful-happy, neutral-happy, and so on) [2]. The study concluded that coupled with certain emotions, people

tend to be able to distinguish genuine smiles from an ingenuine one, though it was faster with angry eyes, slower with fearful or neutral eyes.

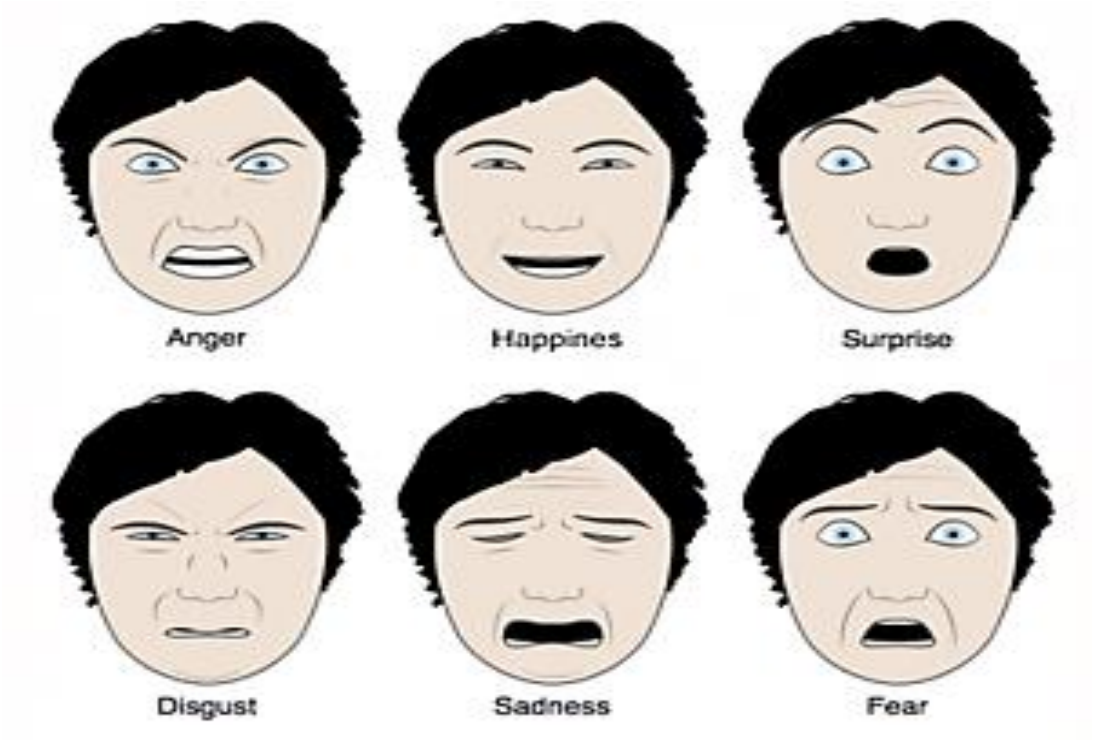


Figure 2.8: Six basic emotions [27].

This work does not directly study these emotions, valence, or arousal, but uses a simple measure; a person's smile. Unless trained in some way or prepared for an exercise, one cannot always distinguish between genuine and ingenuine smiles, though one part of a study involving three and four-year-old children identifying smiles has found that four-year-old children were able to correctly identify genuine smiles in 75% of the trials on average, which is better than chance (50%), whereas the three-year-old children only managed to do so in 51% of the trials, almost similar to chance [28]. The authors of [20] strove to detect deceit using involuntary facial expression combinations that pertain to certain emotions, obtaining great results for differentiating true emotions from acted ones, as stated in Section 1.1.

Moreover, there has been some interest in detecting deceit. For example, a study that uses facial expression was mentioned in 2.1 [20]. Their work is based on the fact

that not all AU combinations (such as AU 23 or lip tightener) can be faked by someone feigning a certain emotion. AUs are represented by Major Components (MCs), such as AU 23 as MC 16 or "red parts of lips narrowed", which support vector machines (SVMs) are trained for [20]. Using time-based features yielded better results than using distance-based features in testing. In the end, they managed to detect deceit with accuracies of 86.02%, 73.16%, 80.46%, and 90.15% for anger, enjoyment, fear, and sadness respectively [20]. An example of works that use both facial expressions and EEG with deep learning is Soleymani *et al*'s [29]. They use LSTM networks with the annotated videos and face tracking data and with the extracted power spectral density of EEG data and concluded that facial expression yields better results for emotion detection than EEG. The LSTM architecture with two hidden layers yielded a p-value of 0.49 ± 0.37 and a root mean square error (RMSE) of 0.043 ± 0.025 with 10 hidden units for facial expression data and yielded a p-value 0.26 ± 0.33 and an RMSE of 0.052 ± 0.029 with 32 hidden units for EEG data [29].

Despite having worse results than facial expression in the previously described experiment, EEG with deep learning remains a viable solution to detecting emotions, since the EEG still includes emotional information in addition to facial expression brain signals, and recording subjects may not always be possible [29].

Furthermore, smile detection is a more specific form of emotion detection. It would be interesting to detect if a person is faking a smile or is genuinely smiling, as this can yield interesting metrics that may be used as a proxy for happiness.

In a study, Hossain and Gedeon [30] used kNN, SVM, simple neural networks, and ensemble classifiers to classify smiles from videos in the MAHNOB [31] and AFEW [32] datasets by human observers. The gauged responses included the papillary response and Galvanic skin response. After pre-processing, the former yielded an accuracy of 97.8% and the latter 96.6% for the ensemble classifiers with leave-one-out cross-validation, which outperformed the other classifiers. However, they also compare this classifier to the observers' classification through self-reports, which "correctly judged smiles as real only 58.9% of the time (on average) to 68.4% (by voting)" [30].

Deep learning has been widely used in affect recognition as of late due to that potential, with many variations on similar networks for various purposes. Miranda and

Patras [23], for instance, use both convolutional and recurrent neural networks together to predict positive and negative affective behavior, in addition to the big five personality traits. For valence and arousal, they obtained average-F1 scores of 0.59, and 0.61 respectively using EEG while the subjects watched videos (audiovisual stimuli) [23]. Another study that is similar in objective was done by Xu and Plataniotis [33]. Their objective involved investigating semi-supervised learning, stacked denoising autoencoders, and deep belief networks for affective state classification, and has achieved higher average-F1 scores, at 86.60% for valence and 86.67% for arousal. Other studies about EEG classification include Williams' investigation on deep learning and transfer learning used multi-layer perceptrons, convolutional neural networks, LSTM, and LSTM-fully convolutional networks to classify positive/negative affect [11]. Williams' best technique, the LSTM + fully convolutional neural network, yielded a maximum accuracy of 64.36%. More studies that use the AMIGOS dataset [34] or DEAP dataset [35] using convolutional neural networks with three different setups that use 2,5, and 10 fully connected layers [36], LSTM [8], deep convolutional neural networks [9], and 3-D convolutional neural networks [10], yielding maximum accuracies of, respectively, 99.72%, 85.65% for arousal and 85.45% for valence, 76% for arousal and 75% for valence, and 88.49% for arousal and 87.44% for valence. Mehmood *et al.* use various algorithms for emotion recognition, but we are concerned with their multilayer perceptron algorithm [37]. They use Weka after feature extraction and made two tests, one with all features, and the other with optimal features, which yielded 50.1% and 73.6%, respectively [37]. Despite manual feature extraction, the best among these methods is evidently [36] at 99.72%, which uses 5 fully connected layers, and phase-locking values as features. Phase locking values "represent the phase synchronization between two time series by taking the absolute average of phase differences over temporal windows" [36].

Though valence and arousal can be used as indicators for certain emotions and to provide a holistic picture for smile recognition, it is simpler to use the EEG or facial expressions to directly classify emotions. Jang, Gunes, and Patras proposed a deep learning algorithm called SmileNet that detects smiling faces in real-time using fully convolutional neural networks with a testing accuracy of 95.76±0.56 % [38]. Their

work does not distinguish fake smiles from genuine ones, but it is a decent precedent of using convolutional neural networks to detect smiles with facial expression.

The aforementioned works mostly show promising results with new techniques. However, none of them are identical in scope to our work. Our work is concerned with detecting fake smiles from true ones, meaning our focus is a motor response that corresponds to a discrete emotion instead of valence and arousal, with the distinction between a genuine response and an acted one. As a motor response, it is reasonable to assume that a fake smile is identical to a true one in appearance or even in the brain signal. However, a true smile would appear more pronounced in brain activity and involuntary facial expression, as the stimulus elicited a genuine emotional reaction. Therefore, it is sound to hypothesize that a fake smile would look different from a genuine one in an EEG and observation, paving the way for machine learning to be used in distinction.

There are fewer works that are concerned with detecting fake emotions from real ones with deep learning than valence and arousal or singular expressions opposed to no expression. One such work is by Huynh and Kim, in which they combine mirror neural modeling and LSTM networks with parametric bias (PB) with facial expression [39]. Facial features are extracted and are split into either genuine or fake. Using just the LSTM-PB yielded a validation accuracy of 71 % for the six basic emotions and a testing accuracy of 66.7 % [39]. Another example would be Kumar *et al*'s work, in which they use a convolutional neural network with three convolutional layers followed by a fully connected layer to detect fake and true smiles from images of facial expression from the FERC-2013 dataset [40]. The first two convolutional layers had 64 5x5 filters and the third had 128 4x4 filters with 3x3 max pooling between them, and a rectified linear units (ReLU) layer before the first max-pooling layer. Their algorithm could detect a real smile from an image with 100 % accuracy and a fake smile with an accuracy of around 90 %, but they do not pit true smiles against fake smiles with their labels unknown [40]. Furthermore, Bahkt *et al*'s work has a similar objective to [40], but focuses on the zygomatic major and orbicularis oculi muscles in videos from the ChaLearn LAP dataset [41] and use SVMs for classification between true and fake

smiles. They obtain a maximum accuracy of 79.2 % focusing on those two muscles as their image features [42].

Deep learning and EEG can also be used together for other applications such as brain-computer interface (BCI) for purposes such as motor imagery. Though motor imagery is somewhat different from this work, it is similar in one fundamental aspect; brain activity causing a motor response. In one paper, Dose *et al.* propose a convolutional neural network with temporal convolution followed by spatial convolution using the Physionet EEG Motor Movement/MI Dataset [43] for motor imagery [44]. Their aim was stroke rehabilitation, and to that end, the protocol they had involved three tasks, the most notable is the 2-class distinction between left and right foot motor imagery. For this task, the authors obtained 80.01% accuracy for global classification and 86.13% accuracy for subject-dependent classification. This shows promise for the use of convolutional neural networks in particular for tasks that involve motor responses, such as smiling.

All of these works and our work do classification or regression offline, meaning that all of these algorithms are used after the data is collected, and not in real-time. It is best to have an online classification system, but an offline-developed algorithm that shows promising results can be used in online classification. Developing and adjusting models and algorithms online is difficult and bothersome to the subjects, hence the focus on offline algorithms.

After reviewing the aforementioned works, we note that few of them address the issue of deception recognition with EEG and machine learning. This demonstrates the need for more data that is concerned with deception recognition with EEG.

We apply deep learning algorithms on EEG data that was collected in our lab. This work demonstrates the strength of these algorithms on data as difficult as EEG data. We do three-way subject-dependent and subject interdependent classification for recognizing acted smiles, genuine smiles, or neutral expression. We achieve strong results, particularly in subject-dependent classification.

We will now review the experimental setup and methodology in the next chapter.

Chapter 3. Experimental Setup and Methodology

In this chapter, the subjects, the equipment and experimental protocol used to obtain the EEG data, and the algorithms used to analyze it are described. Eye movements are also available in the data obtained but will not be used in this work. The inclusion/selection criteria are mentioned, followed by the set up used to display the visual stimuli. This chapter mainly describes the protocol developed in [6]. Furthermore, we formulate the problem of using artificial neural networks in classifying smiles with EEG inputs. We also present the proposed networks for classification.

3.1. Subjects

We gathered a pool of 8 female subjects and 20 male subjects ranging between 18- 26 in age, with no known mental illnesses. Most subjects were from the Electrical Engineering bachelor's program. The subjects chose their preferred timing to perform the experiment, to ensure they were in a more relaxed state, and because it was not hypothesized that Circadian rhythm would affect the outcome of the experiment. The study was approved by the Institutional Review Board (IRB) of the American University of Sharjah. The subjects were informed of the experiment prior and their informed consent was obtained.

The entirety of the protocol was safe and non-invasive; the only associated risks were discomfort with the cap and gel, and possible fatigue, though that was unlikely. In addition to minimal risk, the rewards for participating in the experiment were minimal and included providing participants with snacks after the experiment, or in some cases bonus assignments for their coursework. Benefits to the subject include furthering their understanding of the technology used for the experiment.

3.2. Visual Stimuli and Delivery Protocol

The stimuli came from a set of 247 images, carefully selected from the Geneva Affective Picture Database (GAPED) image dataset [6, 45], expected to elicit a positive reaction (a genuine smile) or no reaction (neutral response), and images of a book randomly placed now and then where the subject was asked to fake a smile. The GAPED is a vast emotional stimulus database that is comprised of 730 pictures that are meant to induce positive, neutral, and negative emotions. Images meant to elicit positive

emotions include, but are not limited to baby humans and animals, whereas the neutral set includes inanimate objects. The negative set is not necessary for this particular work and includes images of human or animal rights violations in addition to images of snakes and spiders [45]. All of these pictures are divided based on their valence and arousal scores. The book is an additional image and can be seen in the instructions window onscreen in Figure 3.2.

Firstly, the subjects would learn where three keys are on the keyboard, each corresponding to a response; “P” when they genuinely smiled, “N” for neutral responses, and “Q” for fake smiles. To clarify, the subject was asked to fake a smile and press the key whenever they would see the book image. Images only remain for 2 seconds or until the subject pressed a key, followed by 1 second of the “focus” cross. The full description of the experimental protocol can be found in [46].

3.3. EEG Data Collection

The setup can be seen in Figure 3.1 and Figure 3.2 below and the task sequence is shown in Figure 3.3 following it. Figure 3.2 shows a subject wearing a 64-electrode EEG cap connected to a parallel port and computer, and it also shows the eye tracker set up to gauge eye motions. This device is not pertinent to this work, though its software was used for the protocol. The subjects are placed on a chair, put on the 10-20 64-electrode EEG cap, then the experimenters fill in the electrodes with Ag/AgCl gel to reduce the impedance between the electrodes and the scalp, and when ready, the EEG is recorded using ANTNeuro software with the help of the 64-channel parallel port shown in Figure 3.1.

The computer used in this protocol delivery had an Intel Core i5 processor using a Windows XP operating system, whereas the computer used for data collection had an Intel Core i5 processor using a Windows 7 operating system. Stimulus delivery and response was performed and recorded using the eye tracker software on the former computer, but can also be done on other software, such as MATLAB, by simply writing code that emulates this protocol. The code displays the instruction page until any key is pressed, then shows the focus cross for a set amount of time, followed by an image for a set period or until a key is pressed, then another focus cross, and so on until the image set is displayed to the subject. Simultaneously, the code sends the trigger to the

parallel port each time a key is pressed, and that is shown on the EEG. Using MATLAB instead of the eye-tracking software eliminates the need for the eye-tracking kit if eye-tracking data is not used in the analysis, thus eliminating a possibly unnecessary cost to the experiment.

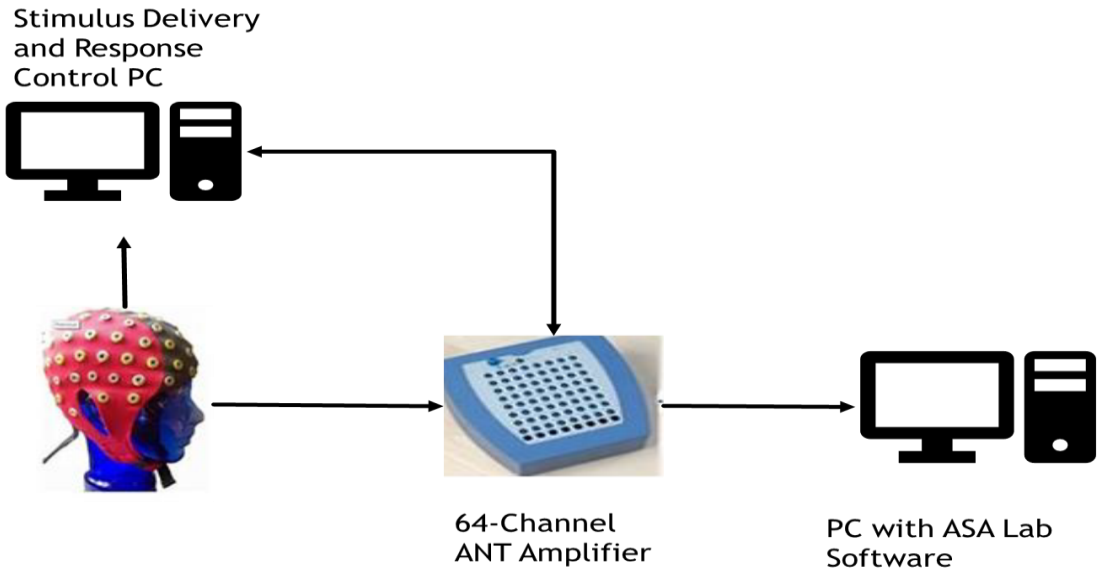


Figure 3.1: Experimental setup [6].

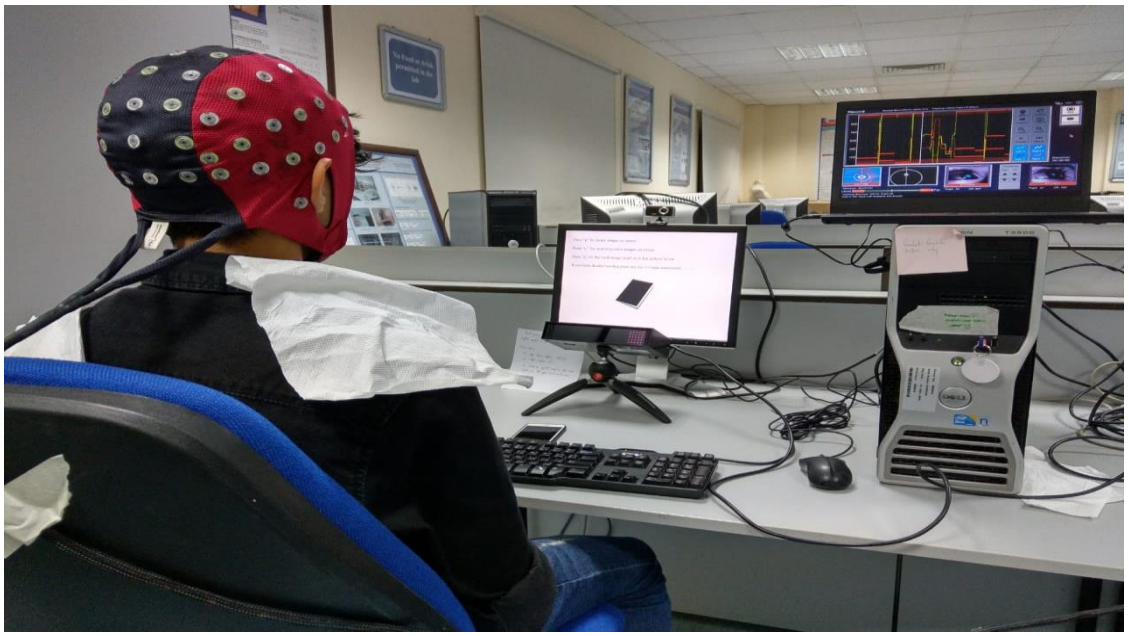


Figure 3.2: Hardware setup for the experimental protocol [6].

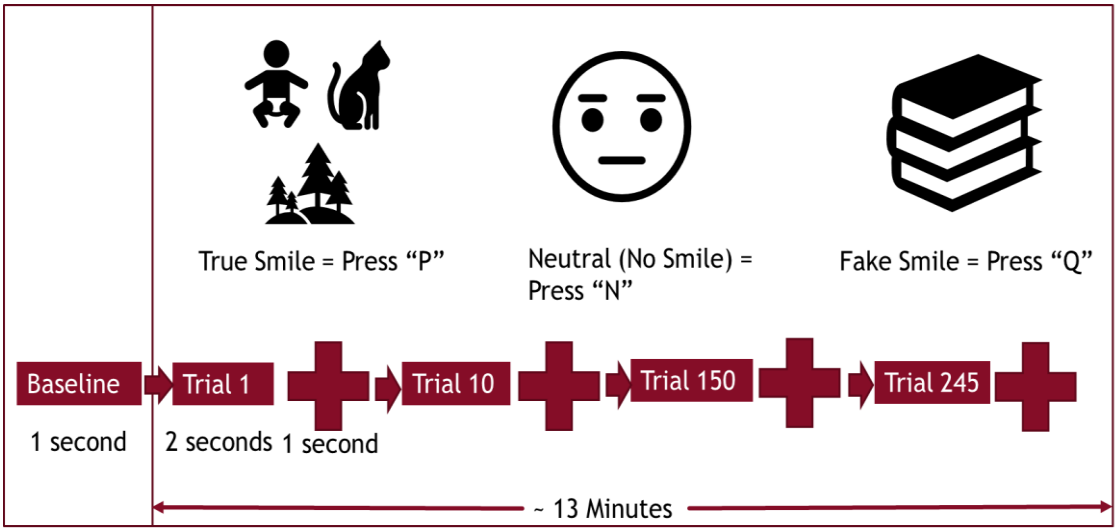


Figure 3.3: Task presentation sequence [47].

3.4. EEG Pre-processing

Though pre-processing is minimally required (manual artifact removal and filtering) with artificial neural networks, some steps are undertaken to maintain similarity with Alex's experiment [6]. Pre-processing goes as follows: To begin with, channels M1 and M2 – the channels corresponding to the two electrodes just above the ears- are removed leaving 62 channels. This would change the electrode scheme from the default 64-electrode scheme to Figure 3.4. M1 and M2 are the reference channels, and are located right beneath the subjects' ears, and, as such, show as a constant 0 in the EEG signals. Therefore, keeping them would skew the data especially when performing independent component analysis (ICA). Afterward, the EEG data is chopped, to remove time segments where motion artifacts were too dominant or distortive to the signal. What remained was filtered using finite impulse response (FIR) filtering in the EEGLAB [48] MATLAB plugin, with the lower cut-off frequency being 0.5 Hz and the upper cut-off frequency being 40 Hz.

FIR filters were the main filtering technique applied throughout this work, primarily due to their simplicity of implementation on EEGLAB in contrast with infinite impulse response (IIR) filters. The default order of the filter is 3000 ($3 * \frac{Sampling\ Frequency}{Low\ Cut-off\ Frequency}$), where the sampling frequency is 500 Hz, and the low cut-off is 0.5 Hz), which is relatively large. However, that is not a deterrent to using FIR, as the

filtering does not occur in real-time. After filtering, the data is re-referenced to the average of all electrodes. The frequency response of this filter is shown in Figure 3.5.

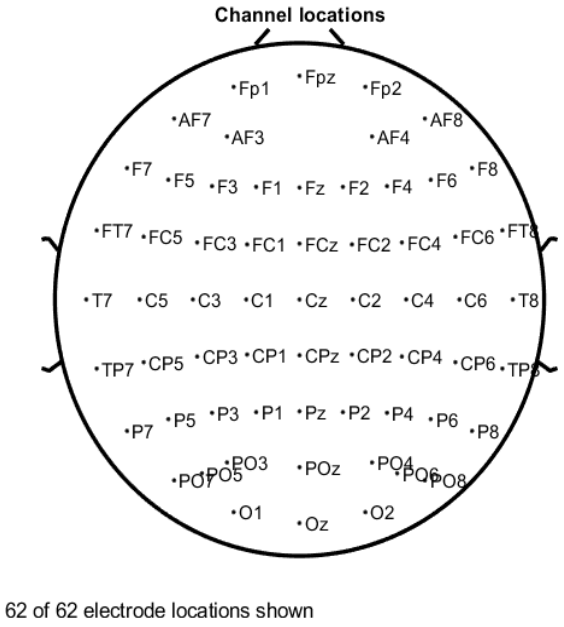


Figure 3.4: Electrode scheme after M1 and M2 were removed.

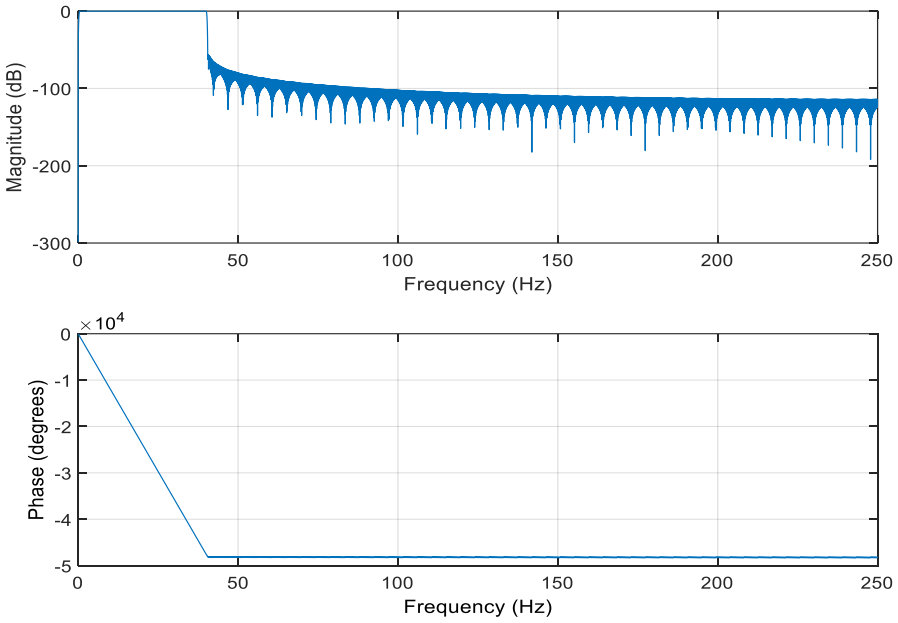


Figure 3.5: Bandpass filter response, Fc1 = 0.5 Hz, Fc2 = 40 Hz.

After filtering, the data is almost ready to be used as input to either network, only the eye blink artifacts must be removed. This is done using independent component analysis (ICA). Each channel is decomposed into 62 independent components and the first 3 components were removed, as we could tell by inspection that they were the root cause behind blinks. The inspection involved looking at the components that were extremely distorted in addition to trial and error while looking at the components and the resulting signal after removal. This is also done with EEGLAB. Finally, the epochs are extracted to make classification easier; that way we have as many epochs as the number of events remaining after chopping the signal, each of which corresponds to one of the three classes.

In addition to using the entirety of each dataset as the input, different frequency bands are input into the simple neural network and the selected architecture of the convolutional neural network. Each band corresponds to the four brain wave bands: beta between 12 and 35 Hz, alpha between 8 and 12 Hz, theta between 4 and 8 Hz, and delta between 0.5 and 4 Hz. The frequency responses are shown in Figure 3.6 through Figure 3.9.

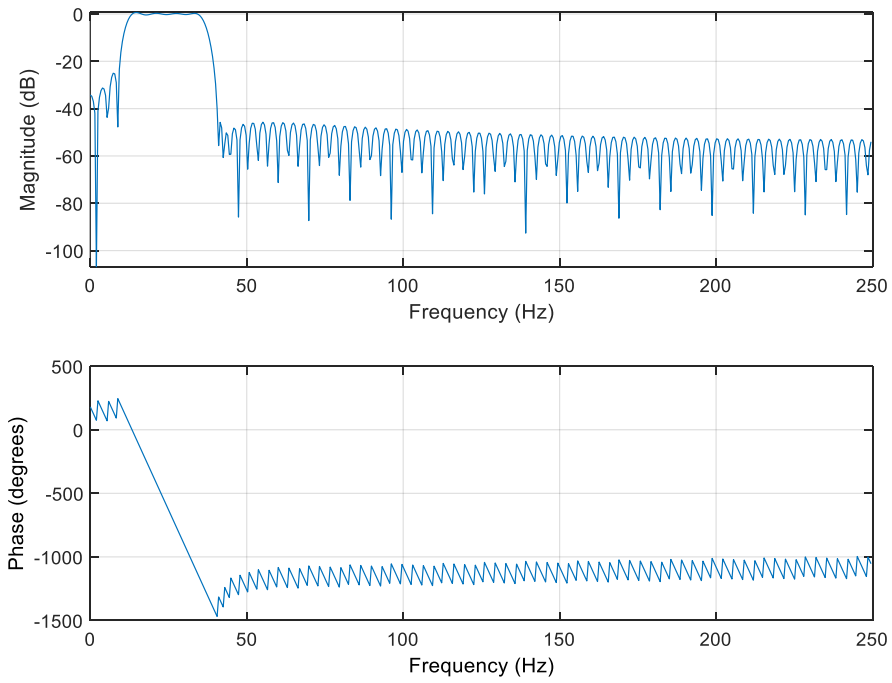


Figure 3.6: Beta (12-35Hz, order = 150) bandpass filter response.

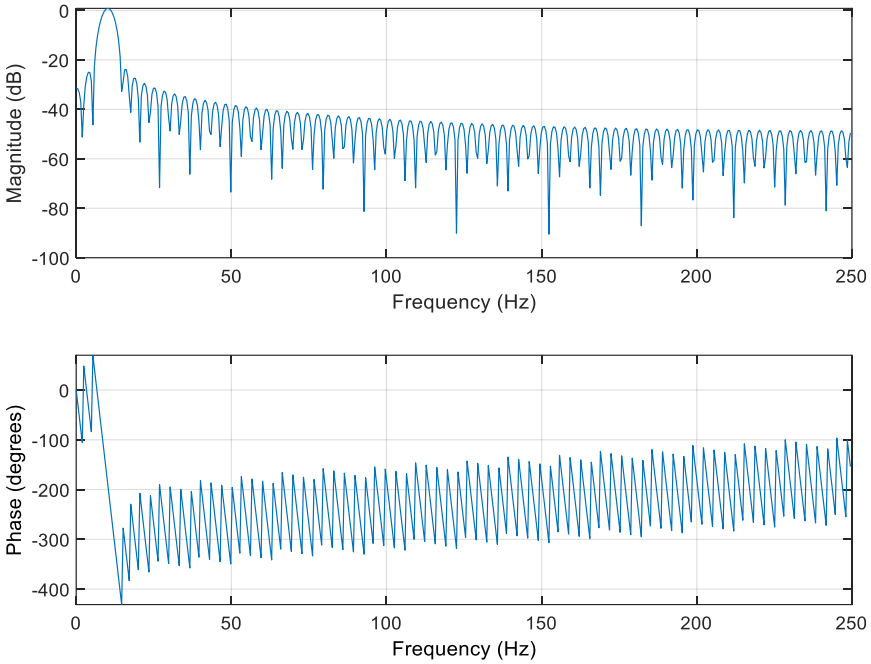


Figure 3.7: Alpha (8-12Hz, order = 150) bandpass filter response.

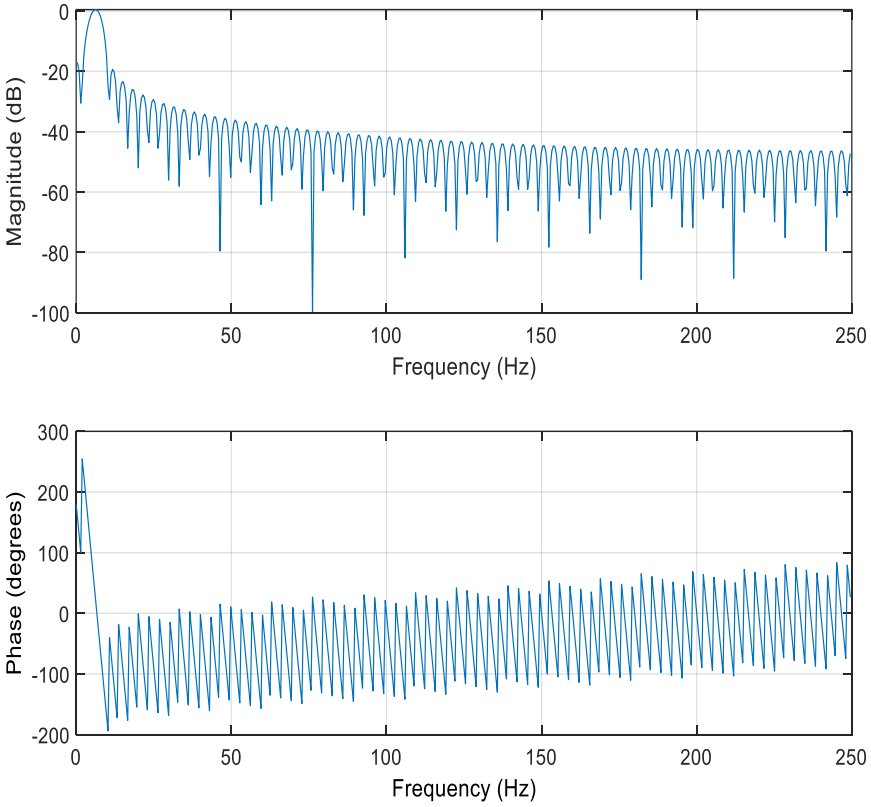


Figure 3.8: Theta (4-8Hz, order = 150) bandpass filter response.

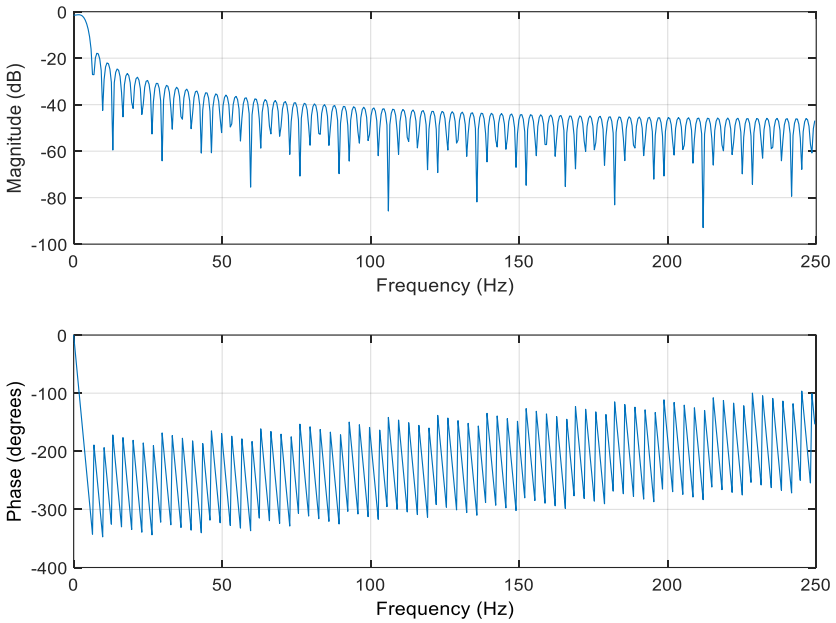


Figure 3.9: Delta (0.5-4Hz, order = 150) bandpass filter response.

3.5. Setup of Machine Learning Algorithms

In our research, three main machine learning algorithms are used. These algorithms include support vector machines (SVMs), simple neural networks, and convolutional neural networks.

As stated in Chapter 2, SVMs are used for binary classification, but multiple SVM classifiers can be combined for multi-class problems. This is done by splitting the n-class problem, for instance, into n binary classification problems in a one-vs-one manner or a one-vs-all manner. The outcomes of each classifier can be combined using the output probabilities.

Simple neural networks involve using three-layer artificial neural networks, where the data is input after pre-processing to the input layer and fed forward to the hidden layer and then the output layer.

Convolutional neural networks differ primarily to the simple network in one aspect; the inclusion of a convolutional layer after the input layer. The architecture can be more complex if desired; more hidden layers can be added, in addition to activation

operations, and normalization. All of their parameters should be optimized to give the best possible result.

3.5.1. Support vector machines. Before moving on to the main focus of the thesis, we establish a baseline as both a basis for comparison with the neural networks and comparison with previous work. This baseline simply involves support vector machines. SVMs are binary classifiers, but, as stated earlier, we can combine multiple binary classifiers to give us three-class classification. We achieve this using built-in functions in MATLAB. We use the radial basis function (RBF) kernel and tune its parameters on part of the training set. Hyperparameter tuning was done using two methods, grid search, and trial-and-error with a part of the training set. The latter yielded a better result and was less time-consuming.

3.5.2. Simple neural network. Following SVM, a simple three-layer network is trained using the data of each subject individually using scaled conjugate gradient as the training function, with 150 neurons in the hidden/middle layer. The network architecture is shown below in Figure 3.10.

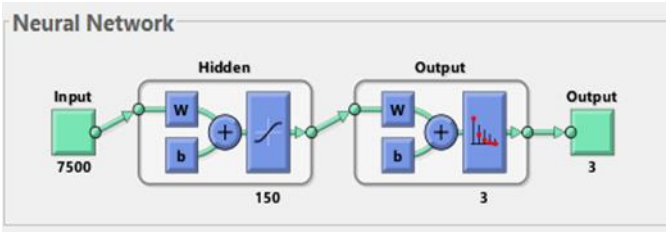


Figure 3.10: Three-layer network visualization.

3.5.3. Convolutional neural network architectures. The architecture and layers of the proposed convolutional neural network are shown in Figure 3.11 through Figure 3.15 below. It can be noted that there are no max or average pooling layers, instead, the convolutional layer(s) were in strides of 3 to 4, depending on the architecture.

The size of the input for the first subject was 62 × 750 × 1 × 180. It was reshaped from 62 × 750 × 180 (channel number x time samples from extracted epochs from -0.5 to 1-second × training events, 75% of the total 240 events for this subject) for 2-D convolution to work in MATLAB. The solver used was stochastic gradient descent with momentum (0.95) or SGDM, the batch size was 128, and the initial learn

rate was 0.01. Also, L2-regularization was used with a factor of 0.0005. All five architectures are described in the next paragraph.

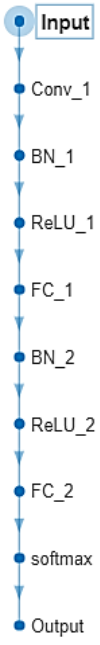
The architectures are fairly simple with only 10 to 13 layers and minor differences between them; the first architecture includes a single convolutional layer with 128 11×11 filters and strides of 4×4, followed by a batch normalization layer, a rectified linear units (ReLU) layer, a fully connected layer with 150 hidden units, a second batch normalization layer, a second ReLU layer, and a final fully connected layer with as many hidden units as our classes (3), followed by a softmax layer. The second and third architectures differ from this one in the convolutional layer; the second has 70 3×3 filters, and the third has 90 5×5 filters, also with 4x4 strides. The fourth and fifth architectures, however, have an additional convolutional layer, a batch normalization layer, and a ReLU layer after the first ReLU layer. The convolutional layers in the fourth architecture have 128 5×5 filters and 90 11×11 filters respectively, both with strides of 4×4, and the convolutional layers in the final architecture have 70 2×2 filters and 40 3×3 filters respectively, but with strides of 3x3 instead.

	Name	Type	Activations	Learnables
1	Input 62×750×1 images with 'zerocenter' normalization	Image Input	62×750×1	-
2	Conv_1 128 11×11×1 convolutions with stride [4 4] and padding [0 0 0 0]	Convolution	13×185×128	Weights 11×11×1×128 Bias 1×1×128
3	BN_1 Batch Normalization with 128 channels	Batch Normalization	13×185×128	Offset 1×1×128 Scale 1×1×128
4	ReLU_1	ReLU	13×185× 128	-
5	FC_1 150 fully connected layer	Fully Connected	1×1× 150	Weights 150×307840 Bias 150×1
6	BN_2 Batch Normalization with 150 channels	Batch Normalization	1×1× 150	Offset 1×1×150 Scale 1×1×150
7	ReLU_2	ReLU	1×1× 150	-
8	FC_2 3 fully connected layer	Fully Connected	1×1×3	Weights 3×150 Bias 3×1
9	softmax	Softmax	1×1×3	-
10	Output crossentropyex	Classification Output	-	-

Figure 3.11: CNN architecture 1.

	Name	Type	Activations	Learnables
1	Input 62×750×1 images with 'zerocenter' normalization	Image Input	62×750×1	-
2	Conv_1 70 3×3×1 convolutions with stride [4 4] and padding [0 0 0 0]	Convolution	15×187×70	Weights 11×11×1×70 Bias 1×1×70
3	BN_1 Batch Normalization with 128 channels	Batch Normalization	15×187×70	Offset 1×1×70 Scale 1×1×70
4	ReLU_1	ReLU	15×187×70	-
5	FC_1 150 fully connected layer	Fully Connected	1×1× 150	Weights 150×196350 Bias 150×1
6	BN_2 Batch Normalization with 150 channels	Batch Normalization	1×1× 150	Offset 1×1×150 Scale 1×1×150
7	ReLU_2	ReLU	1×1× 150	-
8	FC_2 3 fully connected layer	Fully Connected	1×1×3	Weights 3×150 Bias 3×1
9	softmax	Softmax	1×1×3	-
10	Output crossentropyex	Classification Output	-	-

Figure 3.12: CNN architecture 2.



	Name	Type	Activations	Learnables
1	Input 62×750×1 images with 'zerocenter' normalization	Image Input	62×750×1	-
2	Conv_1 90 5×5×1 convolutions with stride [4 4] and padding [0 0 0 0]	Convolution	15×187×90	Weights 5×5×1×90 Bias 1×1×90
3	BN_1 - Batch Normalization with 128 channels	Batch Normalization	15×187×90	Offset 1×1×90 Scale 1×1×90
4	ReLU_1	ReLU	15×187×90	-
5	FC_1 150 fully connected layer	Fully Connected	1×1× 150	Weights 150×252450 Bias 150×1
6	BN_2 Batch Normalization with 150 channels	Batch Normalization	1×1× 150	Offset 1×1×150 Scale 1×1×150
7	ReLU_2	ReLU	1×1× 150	-
8	FC_2 3 fully connected layer	Fully Connected	1×1×3	Weights 3×150 Bias 3×1
9	softmax	Softmax	1×1×3	-
10	Output - crossentropyex	Classification Output	-	-

Figure 3.13: CNN architecture 3.

	Name	Type	Activations	Learnables
1	Input 62×750×1 images with 'zerocenter' normalization	Image Input	62×750×1	-
2	Conv_1 128 5×5×1 convolutions with stride [4 4] and padding [0 0 0 0]	Convolution	15×187×128	Weights 5×5×1×128 Bias 1×1×128
3	BN_1 Batch Normalization with 128 channels	Batch Normalization	15×187×128	Offset 1×1×128 Scale 1×1×128
4	ReLU_1	ReLU	15×187×128	-
5	Conv_2 90 11×11×1 convolutions with stride [4 4] and padding [0 0 0 0]	Convolution	2×45× 90	Weights 11×11×128×90 Bias 1×1×90
6	BN_2 Batch Normalization with 90 channels	Batch Normalization	2×45× 90	Offset 1×1×90 Scale 1×1×90
7	ReLU_2	ReLU	2×45× 90	-
8	FC_2 150 fully connected layer	Fully Connected	1×1× 150	Weights 150×8100 Bias 150×1
9	BN_3 Batch Normalization with 150 channels	Batch Normalization	1×1× 150	Offset 1×1×150 Scale 1×1×150
10	ReLU_3	ReLU	1×1× 150	-
11	FC_3 3 fully connected layer	Fully Connected	1×1×3	Weights 3×150 Bias 3×1
12	softmax	Softmax	1×1×3	-
13	Output crossentropyex	Classification Output	-	-

Figure 3.14: CNN architecture 4.

	Name	Type	Activations	Learnables
1	Input 62×750×1 images with 'zerocenter' normalization	Image Input	62×750×1	-
2	Conv_1 70 2×2×1 convolutions with stride [3 3] and padding [0 0 0 0]	Convolution	21×250×70	Weights 2×2×1×70 Bias 1×1×70
3	BN_1 Batch Normalization with 128 channels	Batch Normalization	21×250×70	Offset 1×1×70 Scale 1×1×70
4	ReLU_1	ReLU	21×250×70	-
5	Conv_2 40 3×3×1 convolutions with stride [3 3] and padding [0 0 0 0]	Convolution	7×83×40	Weights 3×3×70×40 Bias 1×1×40
6	BN_2 Batch Normalization with 90 channels	Batch Normalization	7×83×40	Offset 1×1×40 Scale 1×1×40
7	ReLU_2	ReLU	7×83×40	-
8	FC_2 150 fully connected layer	Fully Connected	1×1× 150	Weights 150×23240 Bias 150×1
9	BN_3 Batch Normalization with 150 channels	Batch Normalization	1×1× 150	Offset 1×1×150 Scale 1×1×150
10	ReLU_3	ReLU	1×1× 150	-
11	FC_3 3 fully connected layer	Fully Connected	1×1×3	Weights 3×150 Bias 3×1
12	softmax	Softmax	1×1×3	-
13	Output crossentropyex	Classification Output	-	-

Figure 3.15: CNN architecture 5.

3.6. Training and Testing Sets

For all aforementioned architectures and algorithms, 4-fold cross-validation was used, meaning three folds were reserved for training and the remaining fold was used for testing.

Taking inter-subject variability into consideration, we preliminarily focused on subject-dependent classification. Later we extended our experimentation to be subject independent analysis as well. In the subject-dependent training and testing scenario, we trained on the data of the same subject and tested on the data from the same subject. While in the subject-independent setting, training and testing data came from different subjects.

After pre-processing was done for all subjects, (see Section 3.4), some of the initial 247 events are removed, and the remaining events range from 229 to 244 for different subjects. The rejected events were removed mainly due to motion artifact during the time of the event, which differs between subjects. Some subjects moved quite a lot, so more events had to be rejected due to motion artifact, such is the case with the subject whose EEG included 229 events. Others did not move as much, so fewer events had to be rejected, as is the case with the subject whose EEG included 244 events.

In the subject-dependent analysis, these events- ranging from 229 to 244- refer to the subject's key presses, and by extension, tells us their response at the instant of the event in the EEG. For each subject, the events were divided into four folds, so, for instance, the subject with 244 events would have 183 event epochs in the training folds and 61 event epochs in the testing fold, all of which are shuffled following each iteration.

The experimental protocol contained 247 images for all subjects instead of being time-limited and thus there were variations in completion time between subjects, and that was coupled with differences in motion artifact (and subsequently events to be removed). One way of doing subject-independent analysis involves splitting the data with the number of events selected to be 229 for all subjects to make a $62\times750\times229\times28$ matrix with all the data, that was reduced with PCA to $41\times750\times229\times28$. However, this caused the additional removal of some events for some subjects. Hence, for the subject-independent approach, the data of all subjects was concatenated into a

62×750×6673 matrix and the same method of cross-validation was applied. Note that the number of channels remains 62, the number of time samples per epoch remains 750, but the number of epochs becomes 6673, which is the sum of the number of epochs for all subjects. This means that the 3 training folds would include the data of 21 subjects (3/4 of 28), and the testing fold will have the data of 7 subjects. LeaveOneOut cross-validation- a special case of K-fold cross-validation with K = 28- was also a possibility for subject-independent analysis, but it significantly increased training time with little improvement to the overall result, thus we used 4-fold cross-validation for it, as well.

The metrics used for analysis include accuracy, sensitivity, specificity, precision, and F-score or F1-measure. The accuracy was obtained by calculating the number of correctly classified instances and dividing that by the total number of instances, sensitivity was obtained by calculating the true positive rate for each class and taking the mean, and specificity was similar but used the true negative rate instead. Precision was computed by finding the positive predictive value for each class and taking the mean, and F-score was obtained using this formula $F-score = \frac{2(Sensitivity)(Precision)}{Sensitivity+Precision}$.

Chapter 4. Results and Discussion

In this chapter, we present the results obtained from classification with the EEG data using the various classifiers described earlier. We then compare them with each other for subject-dependent analysis.

4.1. Classification Results

Hence, we moved on to develop our classification paradigm with SVMs and neural networks for our data. The results with SVM with an RBF kernel were not impressive on our EEG data, showing an accuracy of 48.387% ± 7.57%, which, without proper feature engineering, makes them completely uncompetitive in the scope of this work.

With 4-fold cross-validation, three folds were used to train the convolutional neural network and the remaining fold was used for testing. Figure 4.1 shows the training progress of the selected convolutional neural network architecture for one of the training folds for one subject as an example of the training progress plot. For the other folds, this training progress plot would be identical to this one and continue until all the training folds are through. K-1 folds are used for training and one is used for testing.

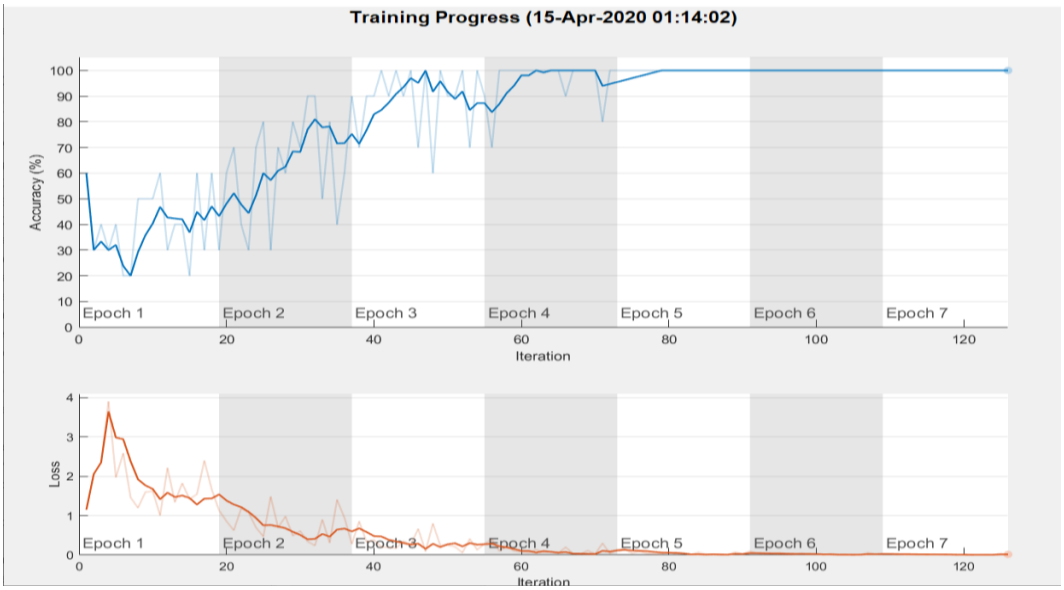


Figure 4.1: Progress of one training fold with a convolutional neural network.

With the simple neural network and the CNN architecture described earlier in 3.5, the testing outcomes are shown in Table 4.1 through Table 4.4.

Table 4.1: Comparison of results of all algorithms for subject-dependent classification.

Algorithm Used		**CNN 1**	**CNN 2**	**CNN 3**	**CNN 4**	**CNN 5**	**Simple NN**	**SVM**
Number of Layers	Convolutional	1	1	1	2	2	N/A	N/A
	Fully Connected	2	2	2	2	2	1	
	Total Number of layers	10	10	10	13	13	3	
Number of filters in convolutional layer(s)		128	70	90	128 & 90	70 & 40	N/A	N/A
Number of hidden units in fully connected layers		150 & 3	150 & 3	150 & 3	150 & 3	150 & 3	150	N/A
Filter size(s) / Box Constraint (SVM)		[11×11]	[3x3]	[5x5]	[5x5] & [11x11]	[2x2] & [3x3]	N/A	1
Stride(s) / Kernel Scale (SVM)		[4x4]	[4x4]	[4x4]	[4x4]	[3x3]	N/A	Auto
Mean Accuracy (%)		90.446 ± 3.11	89.376 ± 2.81	89.513 ± 2.34	89.279 ± 3.15	89.629 ± 3.14	88.879 ± 2.78	48.387 ± 7.57
Mean Sensitivity (%)		89.352 ± 5.54	87.618 ± 6.25	88.012 ± 6.45	89.766 ± 7.14	90.106 ± 7.27	88.084 ± 2.33	N/A
Mean Specificity (%)		92.868 ± 4.22	92.007 ± 4.26	91.559 ± 3.84	89.489 ± 6.01	89.991 ± 5.50	93.806 ± 1.22	N/A
Mean Precision (%)		90.978 ± 4.39	88.799 ± 4.38	88.311 ± 4.46	86.717 ± 6.30	87.907 ± 5.39	88.313 ± 2.56	N/A
Mean F-score (%)		90.028 ± 3.76	88.036 ± 4.09	88.005 ± 4.42	87.918 ± 4.61	88.748 ± 4.63	88.197 ± 2.41	N/A

Table 4.2: Results of the simple network for all subjects.

Subject	**Accuracy (%)**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**	**F-score (%)**
1	88.8	89.0	93.9	90.3	89.6
2	84.7	84.0	91.9	84.9	84.5
3	86.9	87.1	93.4	87.1	87.1
4	91.8	92.3	95.3	92.3	92.3
5	87.9	88.6	93.2	88.5	88.5
6	87.7	88.0	93.8	87.7	87.8
7	87.5	87.6	93.6	87.2	87.4
8	93.8	93.9	96.7	94.7	94.3
9	83.2	83.0	90.8	82.2	82.6
10	87.7	88.1	93.6	88.7	88.4
11	88.1	89.2	94.1	89.2	89.2
12	89.8	89.0	94.2	89.7	89.3
13	85.0	85.3	92.0	84.8	85.1
14	91.0	87.6	94.3	87.5	87.6
15	87.7	87.8	93.1	88.1	87.9
16	94.3	90.1	96.1	91.1	90.6
17	87.3	87.6	93.5	87.2	87.4
18	91.2	88.1	94.5	87.1	87.6
19	93.8	91.4	95.3	93.6	92.5
20	91.8	87.3	94.3	87.5	87.4
21	90.3	86.1	93.2	87.0	86.5
22	91.4	90.7	94.8	90.1	90.4
23	90.1	88.4	94.3	89.2	88.8
24	84.4	84.4	92.2	86.0	85.2
25	87.3	86.6	93.1	87.4	87.0
26	89.4	89.8	94.3	89.3	89.6
27	87.7	88.2	93.5	87.6	87.9
28	88.1	87.4	93.4	86.9	87.2
Mean	88.879	88.085	93.806	88.313	88.197
Standard Deviation	2.781	2.329	1.221	2.558	2.414

Table 4.3: Results of CNN1 for all Subjects.

Architecture 1					
Subject	**Accuracy (%)**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**	**F-score (%)**
1	87.5	85.3	90.8	88.6	86.9
2	87.7	90.2	89.6	86.8	88.5
3	90.7	90.7	92.1	89.8	90.2
4	93.1	93.6	94.9	93.6	93.6
5	90.1	90.5	98.8	96.6	93.4
6	92.3	88.3	95.6	94.2	91.2
7	88.5	83.5	95.1	89.2	86.3
8	88.8	90.5	91.1	88.8	89.6
9	86.1	86.6	90.2	88.2	87.4
10	89.0	94.3	88.5	83.0	88.3
11	89.4	88.0	93.5	91.3	89.6
12	91.1	74.1	99.4	97.7	84.3
13	83.8	81.0	90.4	81.9	81.4
14	96.7	88.4	100.0	100.0	93.8
15	86.9	84.3	92.3	88.7	86.4
16	95.1	98.0	92.9	95.4	96.7
17	89.8	95.1	89.6	86.6	90.7
18	91.3	94.6	86.6	92.7	93.6
19	95.5	97.4	92.1	95.5	96.5
20	94.7	95.7	93.3	95.0	95.3
21	88.3	93.9	80.4	88.5	91.1
22	95.1	88.9	98.9	96.0	92.3
23	88.5	82.7	96.2	86.0	84.3
24	86.9	86.8	91.3	88.5	87.6
25	91.1	95.5	91.2	92.7	94.1
26	89.8	90.5	91.3	86.9	88.7
27	92.7	92.3	95.8	91.1	91.7
28	92.2	81.4	98.4	94.1	87.3
Mean	90.446	89.352	92.868	90.978	90.028
Standard Deviation	3.105	5.539	4.215	4.385	3.762

Table 4.4: Average results of the subject-independent analysis with CNN1.

Accuracy (%)	Sensitivity (%)	Specificity (%)	Precision (%)	F-score (%)
53.4	56.7	62.9	52.7	54.6

The confusion matrices of one iteration of testing of subject-dependent analysis with the simple neural network and CNN1 and subject-independent analysis with CNN1 are shown in Figure 4.2 through Figure 4.4. The confusion matrix shows whether or not the predicted class matched target class, and what other classes it was confused for. Class 1 corresponds to a true smile, 2 corresponds to a fake smile, and 3 corresponds to neutral. These matrices are discussed in more detail in 4.2.

Accuracy, specificity, sensitivity, precision, and F-score can all be acquired through the confusion matrices, as described in 3.6.

Figure 4.2: Confusion matrix of the simple neural network.

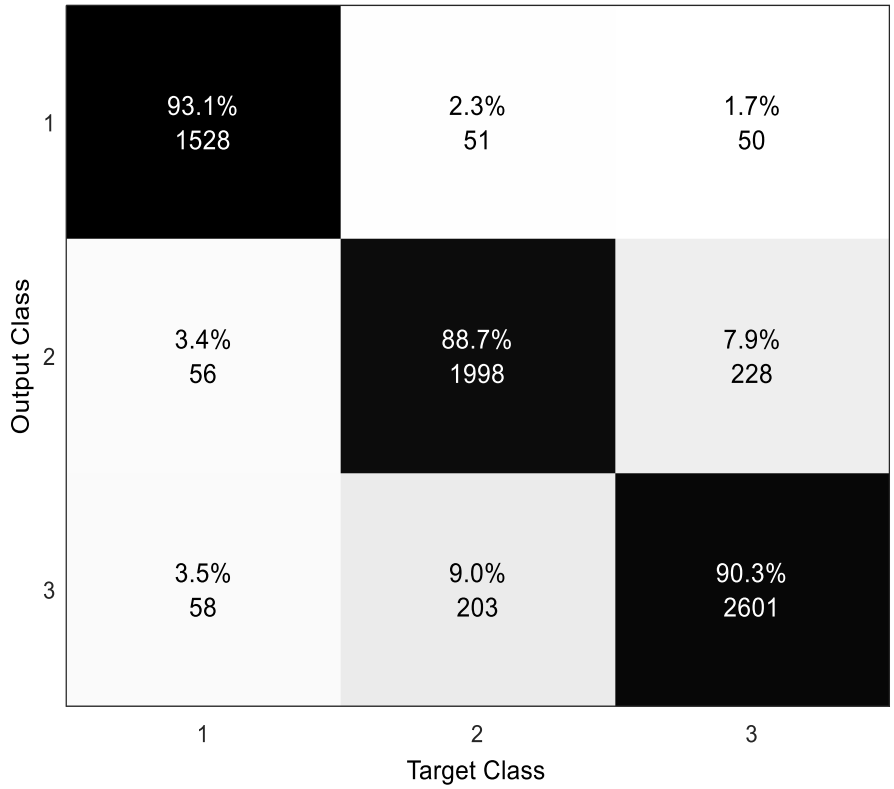


Figure 4.3: Confusion matrix of CNN1.

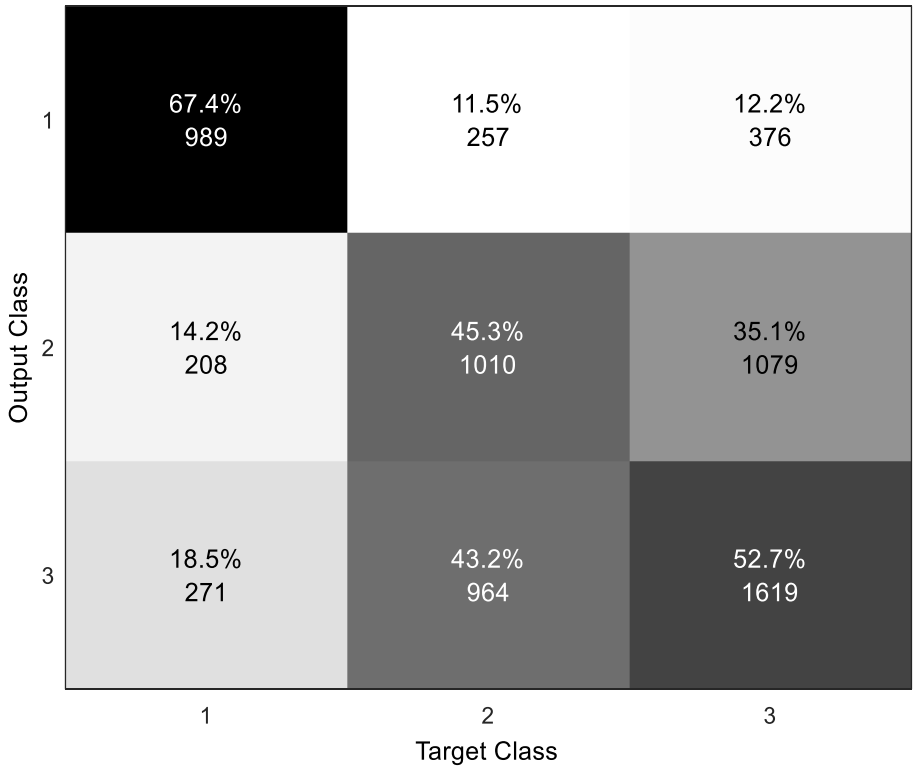


Figure 4.4: Confusion matrix of subject-independent analysis using CNN1.

CNN1 was selected over the other architectures due to its superior performance. This is due to the large number of filters and filter size in its convolutional layer, though our testing shows filter size has a larger effect on performance than the number of filters. For instance, reducing the number of filters from 128 to 70 while keeping the size 11×11 reduced the accuracy from 90.45 % to 90.21 %, whereas reducing the filter size from 11×11 to 3×3 reduced the accuracy from 90.45 % to 89.15 %. The effect of adding an extra convolutional layer (and subsequent batch normalization and ReLU layers) can be seen in Table 4.1 as we go from CNN1 to CNN4. It can be noted that the number of filters decreased as the network went deeper and that the filter size increased, meaning that the second convolutional layer has fewer filters with a larger size than the first. From CNN4 and CNN5, we can see that adding more convolutional and subsequent layers decreases performance. However, reducing the stride size could mitigate this decrease at the expense of significantly increased training time.

The first three architectures are shallower than the latter two. These architectures were tested first because it is faster to train shallow networks than deeper ones. Furthermore, decreasing the strides in convolutional layers, as seen in CNN5, increases training time even further.

To evaluate the performance of different clinical bands of the EEG data, we extracted alpha, beta, theta, and delta bands and used them for four-fold cross-validation with CNN1 architecture. Filtering was performed on the full spectrum after motion and eye blink artifacts are rejected for the four bands. The range of the full spectrum was 0.5 to 40 Hz as mentioned earlier, meaning it includes the four investigated bands in addition to some gamma wave components.

As explained earlier, beta waves include the frequencies ranging from 12-35 Hz, alpha waves from 8-12 Hz, theta from 4-8 Hz, and delta from 0.5-4 Hz and are obtained using the filters described in 3.4.

Gamma waves are not tested on their own in our work, but their results can be inferred when the results of the four tested bands are compared with those of the full spectrum. This is because the remainder of the full spectrum, excluding these four bands includes some frequencies of gamma waves. The results are shown in Table 4.5 through Table 4.9.

Table 4.5: Results of Beta (12-35Hz) and Alpha (8-12Hz) Bands Using the Simple Neural Network.

Band	**Beta**					**Alpha**				
Subject	**Accuracy (%)**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**	**F-Score (%)**	**Accuracy (%)**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**	**F-Score (%)**
1	84.2	83.0	91.3	82.7	82.8	78.8	78.3	89.0	78.6	78.4
2	82.6	82.3	91.0	82.4	82.3	82.2	81.6	91.1	82.1	81.9
3	81.6	81.3	90.5	81.7	81.5	86.9	86.9	93.4	87.1	87.0
4	85.7	84.9	92.3	85.1	85.0	86.5	87.4	92.9	86.7	87.1
5	84.5	83.9	92.0	84.8	84.3	78.0	78.4	88.8	77.8	78.1
6	86.9	86.4	93.1	86.2	86.3	80.0	79.7	89.2	80.7	80.2
7	82.9	82.9	91.3	82.6	82.7	78.8	78.4	89.3	78.9	78.6
8	86.3	86.5	93.1	89.7	88.1	82.9	82.8	91.1	83.1	82.9
9	83.2	81.9	90.6	82.0	81.9	85.3	84.5	92.3	85.0	84.8
10	87.3	86.8	93.3	87.3	87.0	79.7	80.2	89.6	80.0	80.1
11	84.8	85.6	92.2	84.7	85.2	84.0	84.7	91.9	84.0	84.3
12	85.6	83.4	91.7	83.1	83.3	84.8	84.3	92.5	83.8	84.1
13	84.6	84.2	91.8	84.1	84.1	84.2	84.0	91.9	84.8	84.4
14	88.1	84.0	92.5	83.8	83.9	82.4	78.9	89.3	81.1	80.0
15	84.0	83.5	91.7	83.5	83.5	87.7	87.9	93.8	88.8	88.4
16	89.8	84.3	92.3	83.9	84.1	88.6	83.2	91.6	83.6	83.4
17	84.8	83.9	91.9	83.8	83.8	83.7	83.6	91.6	83.2	83.4
18	88.2	81.6	90.2	80.1	80.8	86.4	81.3	91.2	82.1	81.7
19	92.5	88.7	93.8	94.3	91.4	87.9	84.9	92.2	85.7	85.3
20	89.3	85.3	92.7	87.4	86.3	91.4	87.5	94.0	89.8	88.7
21	91.5	84.6	92.6	85.0	84.8	86.0	79.9	89.5	81.9	80.9
22	88.5	85.1	92.6	86.1	85.6	84.0	83.4	91.4	82.0	82.7
23	88.4	84.9	92.6	85.3	85.1	81.9	80.5	90.2	81.4	80.9
24	85.7	85.0	92.7	84.9	84.9	84.8	83.7	91.9	84.1	83.9
25	86.1	83.4	91.1	86.7	85.0	82.4	81.9	90.9	82.3	82.1
26	84.8	84.4	92.0	85.5	85.0	78.4	79.1	89.3	79.4	79.3
27	85.7	83.9	92.0	84.1	84.0	80.0	80.6	90.1	79.9	80.2
28	85.7	83.7	92.0	84.3	84.0	82.0	80.7	89.5	80.0	80.4
Mean (%)	86.180	84.263	92.038	84.824	84.537	83.560	82.443	91.046	82.782	82.610
Standard Deviation (%)	2.574	1.614	0.856	2.661	2.075	3.365	2.870	1.513	2.977	2.891

Table 4.6: Results of Theta (4-8Hz) and Delta (0.5-4Hz) Bands Using Simple Neural Network.

Band	**Theta**					**Delta**				
Subject	**Accuracy (%)**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**	**F-Score (%)**	**Accuracy (%)**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**	**F-Score (%)**
1	88.8	88.8	94.1	89.6	89.2	85.0	84.3	91.9	84.4	84.4
2	86.0	85.6	92.6	87.3	86.4	84.3	84.3	91.9	84.9	84.6
3	87.3	87.7	93.7	87.9	87.8	89.0	88.2	94.6	88.2	88.2
4	88.1	88.7	93.5	88.6	88.6	85.7	85.8	92.4	85.9	85.9
5	82.3	82.4	90.8	82.0	82.2	84.9	84.9	91.6	85.4	85.2
6	85.3	85.1	91.7	86.2	85.7	86.1	86.2	92.5	85.1	85.6
7	83.0	83.6	91.3	82.5	83.1	84.2	84.3	92.3	83.3	83.8
8	88.3	87.7	93.9	88.3	88.0	88.8	88.5	94.4	88.9	88.7
9	82.0	81.7	90.9	83.4	82.5	81.6	81.8	90.8	81.4	81.6
10	88.1	88.8	94.0	88.2	88.5	83.1	83.5	91.2	84.2	83.8
11	85.3	84.9	92.4	84.7	84.8	87.7	88.4	93.6	88.4	88.4
12	79.3	77.6	88.7	78.3	77.9	86.1	85.3	92.7	83.8	84.6
13	87.5	88.5	93.3	88.2	88.3	81.7	80.9	90.3	81.3	81.1
14	86.1	83.6	92.2	82.7	83.2	85.7	83.1	91.7	84.5	83.8
15	87.7	86.7	93.5	88.1	87.4	88.5	88.8	93.5	89.1	88.9
16	89.4	84.5	92.4	85.8	85.1	91.8	88.5	95.4	88.6	88.6
17	87.7	88.6	93.7	87.6	88.1	84.5	84.9	91.8	87.0	86.0
18	89.0	86.1	92.9	85.4	85.8	91.7	89.8	95.4	89.3	89.5
19	93.8	89.1	95.1	88.5	88.8	89.6	85.8	93.2	87.9	86.8
20	91.4	89.4	94.5	88.3	88.8	91.8	89.0	94.6	88.5	88.8
21	89.8	86.6	93.6	87.3	86.9	86.9	84.5	92.1	84.3	84.4
22	89.8	89.0	94.1	89.5	89.3	93.9	92.4	95.9	91.6	92.0
23	84.9	84.0	91.9	83.5	83.8	85.4	84.8	92.4	84.9	84.8
24	86.1	85.3	92.9	85.1	85.2	85.7	86.1	92.7	85.0	85.6
25	78.8	79.5	89.0	77.5	78.5	83.3	82.3	90.8	82.8	82.5
26	88.1	87.7	93.9	88.6	88.2	86.1	86.5	92.7	86.8	86.7
27	90.2	89.5	94.8	90.2	89.8	86.5	86.6	92.8	87.5	87.0
28	88.1	86.4	92.5	87.8	87.1	87.7	86.2	92.4	87.9	87.0
Mean (%)	86.864	85.969	92.779	86.113	86.038	86.681	85.921	92.771	86.108	86.012
Standard Deviation (%)	3.350	3.002	1.542	3.196	3.065	3.027	2.559	1.430	2.492	2.486

Table 4.7: Results of Beta (12-35Hz) and Alpha (8-12Hz) Bands Using CNN1.

Band	**Beta**					**Alpha**				
Subject	**Accuracy (%)**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**	**F-Score (%)**	**Accuracy (%)**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**	**F-Score (%)**
1	82.5	85.3	85.5	83.0	84.2	87.1	88.1	88.5	86.5	87.3
2	86.4	86.3	88.8	85.4	85.9	87.3	92.2	87.3	84.7	88.3
3	88.2	84.1	92.8	90.0	87.0	91.5	92.5	95.7	94.3	93.4
4	87.8	89.0	89.0	86.6	87.8	86.9	87.2	91.2	88.8	88.0
5	81.5	74.6	94.1	82.5	78.3	81.0	74.6	95.9	87.0	80.3
6	89.4	90.1	94.8	93.5	91.7	92.7	91.9	94.8	93.6	92.7
7	88.9	87.3	94.5	88.5	87.9	86.8	78.5	93.9	86.1	82.1
8	87.9	92.4	89.6	87.4	89.8	85.0	88.6	84.4	81.6	84.9
9	85.3	86.6	88.7	86.6	86.6	86.5	86.6	91.0	89.0	87.8
10	87.7	92.0	88.5	82.7	87.1	87.3	92.0	88.5	82.7	87.1
11	89.0	88.0	93.5	91.3	89.6	89.8	90.7	93.5	91.6	91.2
12	88.6	72.4	96.6	87.5	79.2	91.1	77.6	97.8	91.8	84.1
13	87.1	83.3	94.2	88.6	85.9	84.6	85.7	92.3	85.7	85.7
14	93.1	86.0	97.5	88.1	87.1	91.5	88.4	97.5	88.4	88.4
15	82.9	83.3	90.2	85.9	84.6	85.7	85.3	90.2	86.1	85.7
16	90.7	94.6	88.8	92.7	93.6	91.9	97.3	84.7	90.6	93.8
17	88.2	88.2	92.4	89.1	88.7	88.2	92.2	87.5	83.9	87.9
18	88.2	93.9	85.4	92.0	92.9	94.8	98.6	87.8	93.5	96.0
19	91.4	96.1	87.6	93.1	94.6	90.9	95.5	85.4	91.9	93.6
20	91.0	93.5	87.6	90.9	92.2	91.0	96.4	84.8	89.3	92.7
21	87.0	92.5	79.3	87.7	90.1	89.5	95.9	80.4	88.7	92.2
22	88.9	81.5	96.3	86.3	83.8	90.6	83.3	95.8	84.9	84.1
23	84.3	78.8	92.9	75.9	77.4	88.9	84.6	95.6	84.6	84.6
24	86.9	85.8	92.8	90.1	87.9	84.0	84.0	88.4	84.8	84.4
25	88.2	91.0	88.5	90.3	90.6	89.0	91.0	92.0	93.1	92.0
26	83.7	83.2	88.0	81.4	82.3	87.8	89.5	91.3	86.7	88.1
27	86.1	83.3	91.6	82.3	82.8	84.5	83.3	88.6	77.4	80.2
28	85.2	78.0	97.3	90.2	83.6	86.9	78.0	95.1	83.6	80.7
Mean (%)	87.4	86.5	91.0	87.5	86.9	88.3	88.2	90.7	87.5	87.8
Standard Deviation (%)	2.7	5.8	4.1	4.0	4.4	3.0	6.1	4.4	4.0	4.3

Table 4.8: Results of Theta (4-8Hz) and Delta (0.5-4Hz) Bands Using CNN1.

Band	Theta					Delta				
Subject	Accuracy (%)	Sensitivity (%)	Specificity (%)	Precision (%)	F-Score (%)	Accuracy (%)	Sensitivity (%)	Specificity (%)	Precision (%)	F-Score (%)
1	87.1	87.2	90.8	88.8	88.0	87.5	85.3	90.8	88.6	86.9
2	88.6	91.2	88.8	86.1	88.6	88.6	91.2	89.6	86.9	89.0
3	90.7	92.5	91.4	89.2	90.8	91.5	91.6	92.8	90.7	91.2
4	89.0	86.2	92.6	90.4	88.3	91.4	90.8	92.6	90.8	90.8
5	85.3	79.4	97.0	90.9	84.7	87.5	88.9	98.2	94.9	91.8
6	91.1	91.9	91.9	90.3	91.1	89.8	89.2	91.9	90.0	89.6
7	88.9	83.5	92.7	84.6	84.1	88.9	84.8	95.1	89.3	87.0
8	89.2	90.5	91.9	89.6	90.0	86.3	86.7	88.1	85.0	85.8
9	84.5	87.5	85.7	83.8	85.6	86.5	85.7	89.5	87.3	86.5
10	88.1	94.3	87.2	81.4	87.4	91.1	93.2	91.9	87.2	90.1
11	91.1	90.7	94.2	92.5	91.6	90.7	90.7	93.5	91.6	91.2
12	86.5	72.4	98.3	93.3	81.6	90.3	74.1	97.2	89.6	81.1
13	90.8	85.7	94.2	88.9	87.3	87.1	86.9	91.0	83.9	85.4
14	92.7	88.4	98.0	90.5	89.4	95.5	88.4	99.0	95.0	91.6
15	87.8	89.2	88.8	85.0	87.1	88.2	86.3	93.0	89.8	88.0
16	91.9	95.9	86.7	91.6	93.7	95.9	98.0	94.9	96.7	97.3
17	91.5	95.1	91.7	89.0	91.9	89.8	93.1	93.1	90.5	91.8
18	91.7	95.2	85.4	92.1	93.6	93.9	97.3	90.2	94.7	96.0
19	93.4	98.1	88.8	93.8	95.9	93.4	96.1	88.8	93.7	94.9
20	91.8	97.1	84.8	89.4	93.1	93.4	93.5	93.3	94.9	94.2
21	91.6	97.3	83.7	90.5	93.8	86.2	92.5	78.3	87.2	89.8
22	95.1	87.0	97.9	92.2	89.5	93.4	88.9	97.4	90.6	89.7
23	86.8	78.8	95.1	82.0	80.4	88.9	84.6	95.6	84.6	84.6
24	86.5	90.6	89.1	86.5	88.5	86.1	83.0	92.8	89.8	86.3
25	90.2	94.0	88.5	90.6	92.3	89.8	91.0	91.2	92.4	91.7
26	90.2	91.6	92.7	88.8	90.2	87.3	90.5	89.3	84.3	87.3
27	89.4	84.6	94.0	86.8	85.7	90.6	92.3	96.4	92.3	92.3
28	90.2	79.7	96.2	87.0	83.2	90.2	86.4	95.7	86.4	86.4
Mean (%)	89.7	89.1	91.4	88.8	88.8	90.0	89.3	92.5	90.0	89.6
Standard Deviation (%)	2.5	6.2	4.1	3.2	3.8	2.8	4.8	4.0	3.5	3.6

Table 4.9: Results of the Four Clinical Bands for Subject-Independent Classification Using CNN1.

Beta					Alpha				
Accuracy	**Sensitivity**	**Specificity**	**Precision**	**F-score**	**Accuracy**	**Sensitivity**	**Specificity**	**Precision**	**F-score**
48.7	53.6	60.5	49.7	51.6	46.6	55.4	58.4	49.3	52.2
Theta					**Delta**				
Accuracy	**Sensitivity**	**Specificity**	**Precision**	**F-score**	**Accuracy**	**Sensitivity**	**Specificity**	**Precision**	**F-score**
49.8	52.8	62.4	50.5	51.6	50.6	56.8	61.0	51.5	54.0

Figure 4.5 shows a summary of Table 4.2 to Table 4.8 in the form of a bar chart. The mean accuracy and standard deviation of subject-dependent analysis and the accuracy of four-fold cross-validation with all events and subject-independent analysis for all clinical bands and the full spectrum are included, with the exclusion of SVM in this case. The performance of the four brain waves was not measured for SVM due to its comparatively inferior performance with the full spectrum.

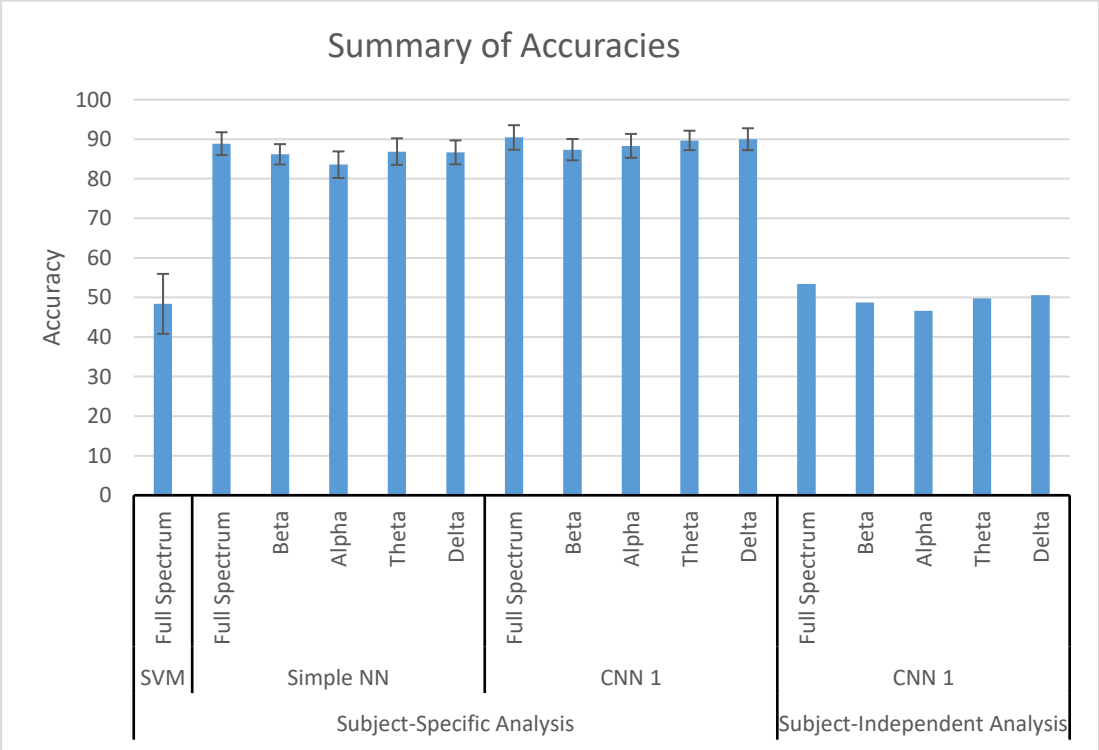


Figure 4.5: Final summary of results.

4.2. Discussion

In the following, we discuss our results. Note that, only the average accuracy of the SVMs was recorded for all 28 subjects, but the metrics used to evaluate simple neural network and convolutional network performance also included the average sensitivity, specificity, precision, and F-score. The results are collectively shown in Table 4.1. Firstly, ECOC SVM and simple neural networks were verified with the Fashion-MNIST dataset [49]. The results achieved on these datasets were comparable to the other published works. As stated earlier, five architectures of convolutional neural networks are tested. The architectures were mostly similar with the differences being the convolutional layer; the first three had the same layers but with different convolutional layers, and the last two had two convolutional layers instead of one like the other three. Optimal architecture is selected based on cross-validation accuracy and its standard deviation, and architecture complexity/computation time, both factors were equally important. It is important to note that the results shown for SVM are those of the one-vs-one coded classifier, as that set of results was better than one-vs-all using the RBF kernel, as the linear and polynomial kernels both yielded lower accuracies (47.43± 8.58 % and 43.5233 ± 7.92 % respectively). Note that there is only one work on the discrimination of genuine vs acted smiles with EEG data by Alex [6, 46]. However, [6] only does two-way classification and extracts a multitude of features (power spectral density, Pearson correlation coefficient, phase-locking value, and phase lag index). Since [6] follows a different method of analysis, the results cannot be compared. We do three-way classification, which is a harder problem. Moreover, we extract features from a dataset that are learned algorithmically.

It is evident from Table 4.1 that convolutional neural networks, in general, do perform better than, though similarly to the simple neural network. However, both are vastly superior to SVM, given the same pre-processing steps for this three-class problem. This is due to the lack of feature extraction and engineering for all algorithms, and due to the fact that SVMs perform better for binary problems. It is also evident that architecture 1 yields better results in terms of accuracy, specificity, precision, and F-score than the other architectures. Hence, we selected CNN1 as the optimal architecture, due to these metrics, and due to the less complex nature of the architecture, hence shorter computation time and fewer required resources. The results of all subjects for

this architecture are shown in Table 4.3 and the results of all subjects with the simple neural network are shown in Table 4.2 to provide a better picture on how these networks perform overall, and more importantly, to showcase subject variability and how it translates to classification differences.

Despite having lower overall performance than CNN1, the simple network performs well enough and takes less time to train than the convolutional networks, and on MATLAB, it provides a wider variety of options, such as various training functions as opposed to just SGDM, ADAM, and RMSProp, and the ease of changing all network parameters, not to mention the fact that it does not require high-end hardware to train and obtain results. This makes it a better choice than the other algorithms discussed if hardware resources present an issue.

As we can see from Table 4.4, the accuracy of the subject-independent analysis is lower than the average accuracy for subject-dependent analysis. This is mainly attributed to subject variability. Subject variability is concerned with physiological differences between the subjects. Meaning different brains could react differently to the same stimuli. Subject variability is also concerned with differences in subject motion and blinking during the experiment; each subject moves differently from the others and they blink at different times.

The confusion matrices in Figure 4.2, Figure 4.3, and Figure 4.4 show what each of the algorithms confuses the target classes with. From

Figure 4.2, which shows the confusion matrix of the simple neural network, we can see that the algorithm could generally correctly classify the events in that instance, as it has an accuracy of 88.879 %. We also note that the algorithm confuses fake smiles with the neutral class and the neutral class with fake smiles more than the other class pairs. The confusion matrices shown in Figure 4.3 and Figure 4.4 also show a higher rate of confusion between the neutral class and fake smiles than other class pairs. However, CNN1 yields higher accuracy than the simple neural network, and consequently a lower confusion rate even among fake and neutral classes than the simple network. That is not the case for subject-independent analysis with CNN1. Although the confusion rate among fake and neutral classes is higher than the rest, this algorithm shows a large discrepancy in accuracy compared to the previous subject-

dependent algorithms. This is because it confuses fake for neutral and neutral for fake a lot more frequently than the other algorithms. As stated before, the reason for this discrepancy is subject variability. However, we conjecture the reason for higher confusion rates between fake and neutral is the similarity of fake and neutral class stimuli. The stimulus presented for the fake smile prompt is simply an image of a book and this image can generally be thought of as neutral, as it would not generally elicit a genuine response based on its valence and arousal scores. Hence, a fake smile would appear on the EEG as a neutral response coupled with the brain's signal to the body to move the required facial muscles to smile. This makes it slightly more difficult to distinguish those two classes than others in subject-dependent analysis and makes it tremendously difficult in subject-independent analysis, where it confuses fake for neutral and neutral for fake almost as much as it correctly classifies neutral and fake.

For each frequency band (beta, alpha, theta, and delta) we can see that the average performance is, in fact, slightly subpar when compared with using the entire pre-processed spectrum (0.5-40 Hz). This entails the loss of some features or patterns as some of those frequencies are filtered out. It is noted that despite all four bands showing a lower accuracy than the entire spectrum for all algorithms, theta and delta perform almost equivalently for the simple architecture, and delta performs better than the other three for the selected CNN architecture for subject-dependent and subject-independent analysis. We note the pattern of delta performing well throughout the three algorithms, despite theta coming close with the simple network. Delta waves are usually associated with deep sleep, yet show the best, or close to the best, performance, excluding the full spectrum, for all algorithms. We speculate this is due to the low frequency of delta waves, meaning they do not get affected by noise as much as the higher frequency bands, which translates to a lower error or higher accuracy. However, the superior performance obtained with the full spectrum could mean smiles are related to gamma waves more closely than beta, alpha, theta, and delta waves. Gamma waves are associated with brain hyperactivity, which the experimental protocol could fall under in terms of brain stimulation. This could also mean that smile expression is not frequency-specific.

Chapter 5. Conclusion

In conclusion, the purpose of this research was finding a method to detect fake smile using EEG data with machine learning and deep learning techniques. To that end, we developed an experiment with which to obtain the EEG data of 28 healthy subjects between the ages of 18 and 26 with normal or corrected to normal vision, where they would respond to a stimulus with a fake smile, a true smile, or neutrally. After the data is collected, it was pre-processed to ensure the significant parts of the EEG remain. The signal was first inspected manually, and distorted regions were rejected, then the signal was bandpass filtered from 0.5 to 40 Hz, then it was pruned with independent component analysis (ICA) to remove eye blink artifacts, and finally the epochs were extracted. Three machine learning algorithms were used for classification, SVMs, simple NNs, and CNNs. SVMs perform worse than both deep learning techniques for this dataset, showing an accuracy of 48.387 % compared to simple NN's 88.879 % and CNN's 90.446 % for subject-independent analysis.

From Chapter 4, CNNs show an improvement over the simple network, showing almost a 1.6% increase in accuracy for the selected convolutional network architecture. Though both the first CNN architecture and the simple network do not take long to train, and their computation time is comparable, we recommend the use of the simple neural network over the convolutional one, especially for lower-end machines where a GPU is not available.

To sum up, convolutional neural networks are slightly better than simple neural networks, but both proved to be vastly superior to SVMs for this three-class problem, and with this particular dataset.

Future work includes subject-independent analysis with the same methods stated earlier with the simple neural network, and eliminating some of the pre-processing steps, such as eye blink removal, which can be time-consuming. Furthermore, the use of time-frequency data (obtained via wavelet transform) might give an improvement in results, particularly using convolutional neural networks. Finally, the use of LSTM networks remains a prospect worth looking into.

References

[1] S. Centorrino, E. Djemai, A. Hopfensitz, M. Milinski, and P. Seabright, "Honest signaling in trust interactions: smiles rated as genuine induce trust and signal higher earning opportunities," *Evolution and Human Behavior*, vol. 36, no. 1, pp. 8-16, 2015, doi: https://doi.org/10.1016/j.evolhumbehav.2014.08.001.

[2] M. G. Calvo, H. Marrero, and D. Beltrán, "When does the brain distinguish between genuine and ambiguous smiles? An ERP study," *Brain and Cognition*, vol. 81, no. 2, pp. 237-246, 2013, doi: https://doi.org/10.1016/j.bandc.2012.10.009.

[3] M. G. Calvo and D. Beltrán, "Brain lateralization of holistic versus analytic processing of emotional facial expressions," *NeuroImage*, vol. 92, pp. 237-247, 2014, doi: https://doi.org/10.1016/j.neuroimage.2014.01.048.

[4] W. G. Iacono, "Accuracy of polygraph techniques: Problems using confessions to determine ground truth," *Physiology & Behavior*, vol. 95, no. 1, pp. 24-26, 2008, doi: https://doi.org/10.1016/j.physbeh.2008.06.001.

[5] I. Thielmann and B. E. Hilbig, "No gain without pain: The psychological costs of dishonesty," *Journal of Economic Psychology*, vol. 71, pp. 126-137, 2019, doi: https://doi.org/10.1016/j.joep.2018.06.001.

[6] M. Alex, "Discrimination Between Genuine and Acted Expressions Using EEG Signals and Machine Learning " Master Thesis, American University of Sharjah, 2019 .

[7] Y. Roy, H. Banville, I. Albuquerque, A. Gramfort, T. Falk and J. Faubert, "Deep learning-based electroencephalography analysis: a systematic review", *Journal of Neural Engineering*, vol. 16, no. 5, pp. 1-37, 2019. doi: 10.1088/1741-2552/ab260c.

[8] S. Alhagry, A. Aly and R. E., "Emotion Recognition based on EEG using LSTM Recurrent Neural Network", *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 10, 2017. doi: 10.14569/ijacsa.2017.081046.

[9] L. Santamaria-Granados, M. Munoz-Organero, G. Ramirez-González, E. Abdulhay, and N. Arunkumar, "Using Deep Convolutional Neural Network for

Emotion Detection on a Physiological Signals Dataset (AMIGOS)," *IEEE Access*, vol. 7, pp. 57-67, 2019, doi: 10.1109/ACCESS.2018.2883213.

[10] E. Salama, R. El-Khoribi, M. E.Shoman and M. A.Wahby, "EEG-Based Emotion Recognition using 3D Convolutional Neural Networks", *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 8, pp. 329, 2018. doi: 10.14569/ijacsa.2018.090843.

[11] J. Williams, "Deep Learning and Transfer Learning in the Classification of EEG Signals", Master Thesis, University of Nebraska, 2017.

[12] R. Guilherme Carvalhal, "The cerebral sulci and gyri," *Neurosurgical Focus FOC*, vol. 28, no. 2, p. E2, 2010, doi: 10.3171/2009.11.FOCUS09245.

[13] K. Sukel. "Neuroanatomy: The Basics," Dana Foundation. https://www.dana.org/article/neuroanatomy-the-basics/ (accessed April 2020).

[14] S. M. Alarcão and M. J. Fonseca, "Emotions Recognition Using EEG Signals: A Survey," *IEEE Transactions on Affective Computing*, vol. 10, no. 3, pp. 374-393, 2019, doi: 10.1109/TAFFC.2017.2714671.

[15] D. Schomer and F. Lopes da Silva, *Niedermeyer's Electroencephalography: Basic Principles, Clinical Applications, and Related Fields.*, 7th ed. Oxford: Oxford University Press, 2011.

[16] D. Plass-Oude Bos, "EEG-based Emotion Recognition," *The Influence of Visual and Auditory Stimuli*, vol. 56, no. 3, pp. 1-17, 2006.

[17] N. Bigdely-Shamlo, T. Mullen, C. Kothe, K.-M. Su, and K. A. Robbins, "The PREP pipeline: standardized preprocessing for large-scale EEG analysis," *Frontiers in Neuroinformatics*, vol. 9, pp. 16-16, 2015, doi: 10.3389/fninf.2015.00016.

[18] E. Alpaydin, *Introduction to machine learning* (Adaptive computation and machine learning). Cambridge, Mass: MIT Press, 2010, p. 537.

[19] S. Saha. "A Comprehensive Guide to Convolutional Neural Networks—the ELI5 way." Towards Data Science. https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53 (accessed April 15, 2019).

[20] Z. Zhang, V. Singh, T. E. Slowe, S. Tulyakov, and V. Govindaraju, "Real-time Automatic Deceit Detection from Involuntary Facial Expressions," in *2007*

IEEE Conference on Computer Vision and Pattern Recognition, 17-22 June 2007, pp. 1-6, doi: 10.1109/CVPR.2007.383383 .

[21] C. Crivelli, P. Carrera, and J.-M. Fernández-Dols, "Are smiles a sign of happiness? Spontaneous expressions of judo winners," *Evolution and Human Behavior*, vol. 36, no. 1, pp. 52-58, 2015, doi: https://doi.org/10.1016/j.evolhumbehav.2014.08.009.

[22] F. Kawakami, K. Kawakami, M. Tomonaga, and K. Takai-Kawakami, "Can we observe spontaneous smiles in 1-year-olds?," *Infant Behavior and Development*, vol. 32, no. 4, pp. 416-421, 2009, doi: https://doi.org/10.1016/j.infbeh.2009.07.005 (accessed March 2020).

[23] J. A. Miranda-Correa and I. Patras, "A Multi-Task Cascaded Network for Prediction of Affect, Personality, Mood and Social Context Using EEG Signals," in *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, 15-19 May 2018, pp. 373-380, doi: 10.1109/FG.2018.00060 .

[24] W. Liu, W. Zheng and B. Lu, "Emotion Recognition Using Multimodal Deep Learning", in *International Conference on Neural Information Processing*, pp. 521-529, 2016. doi: 10.1007/978-3-319-46672-9_58.

[25] A. Frydenlund and F. Rudzicz, "Emotional Affect Estimation Using Video and EEG Data in Deep Neural Networks", in *Canadian Conference on Artificial Intelligence*, pp. 273-280, 2015. doi: 10.1007/978-3-319-18356-5_24.

[26] P. Ekman and W. V. Friesen, "Constants across cultures in the face and emotion," *Journal of Personality and Social Psychology*, vol. 17, no. 2, pp. 124-129, 1971, doi: 10.1037/h0030377.

[27] "Six Basic Emotions," Management Mania. https://managementmania.com/en/six-basic-emotions (accessed March 25, 2019).

[28] R. Song, H. Over, and M. Carpenter, "Young children discriminate genuine from fake smiles and expect people displaying genuine smiles to be more prosocial," *Evolution and Human Behavior*, vol. 37, no. 6, pp. 490-501, 2016, doi: https://doi.org/10.1016/j.evolhumbehav.2016.05.002 (accessed April 2019).

[29] M. Soleymani, S. Asghari-Esfeden, Y. Fu, and M. Pantic, "Analysis of EEG Signals and Facial Expressions for Continuous Emotion Detection," *IEEE Transactions on Affective Computing*, vol. 7, no. 1, pp. 17-28, 2016, doi: 10.1109/TAFFC.2015.2436926.

[30] M. Z. Hossain and T. Gedeon, "Observers' physiological measures in response to videos can be used to detect genuine smiles," *International Journal of Human-Computer Studies*, vol. 122, pp. 232-241, 2019, doi: https://doi.org/10.1016/j.ijhcs.2018.10.003 (accessed April 2020).

[31] S. Petridis, B. Martinez, and M. Pantic, "The MAHNOB Laughter database," *Image and Vision Computing*, vol. 31, no. 2, pp. 186-202, 2013, doi: https://doi.org/10.1016/j.imavis.2012.08.014 (accessed February 2020).

[32] A. Dhall, R. Goecke, S. Lucey, and T. Gedeon, "Acted Facial Expressions In The Wild Database," *Technical Report TR-CS-11, 2*, pp. 1, 2011.

[33] H. Xu and K .N. Plataniotis, "Affective states classification using EEG and semi-supervised deep learning approaches," in *2016 IEEE 18th International Workshop on Multimedia Signal Processing (MMSP)*, 21-23 Sept. 2016, pp. 1-6, doi: 10.1109/MMSP.2016.7813351 .

[34] J. A. M. Correa, M. K. Abadi, N. Sebe, and I. Patras, "AMIGOS: A Dataset for Affect, Personality and Mood Research on Individuals and Groups," *IEEE Transactions on Affective Computing* (Early Access), pp. 1-1, 2018, doi: 10.1109/TAFFC.2018.2884461.

[35] S. Koelstra *et al.*, "DEAP: A Database for Emotion Analysis; Using Physiological Signals," *IEEE Transactions on Affective Computing*, vol. 3, no. 1, pp. 18-31, 2012, doi: 10.1109/T-AFFC.2011.15.

[36] S. Moon, S. Jang, and J. Lee, "Convolutional Neural Network Approach for EEG-Based Emotion Recognition Using Brain Connectivity and its Spatial Information," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 15-20 April 2018, pp. 2556-2560, doi: 10.1109/ICASSP.2018.8461315 .

[37] R. M. Mehmood, R. Du, and H. J. Lee, "Optimal Feature Selection and Deep Learning Ensembles Method for Emotion Recognition From Human Brain EEG

Sensors," *IEEE Access*, vol. 5, pp. 14797-14806, 2017, doi: 10.1109/ACCESS.2017.2724555.

[38] Y. Jang, H. Gunes, and I. Patras, "SmileNet: Registration-Free Smiling Face Detection In The Wild," in *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, 22-29 Oct. 2017, pp. 1581-1589, doi: 10.1109/ICCVW.2017.186 .

[39] Y. Kim and X. Huynh, "Discrimination Between Genuine Versus Fake Emotion Using Long-Short Term Memory with Parametric Bias and Facial Landmarks," in *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, 22-29 Oct. 2017, pp. 3065-3072, doi: 10.1109/ICCVW.2017.362 .

[40] G. A. R. Kumar, R. K. Kumar, and G. Sanyal, "Discriminating real from fake smile using convolution neural network," in *2017 International Conference on Computational Intelligence in Data Science(ICCIDS)*, 2-3 June 2017, pp. 1-6, doi: 10.1109/ICCIDS.2017.8272651 .

[41] S. Escalera, X. Baró, H. J. Escalante, and I. Guyon, "ChaLearn looking at people: A review of events and resources," in *2017 International Joint Conference on Neural Networks (IJCNN)*, 14-19 May 2017, pp. 1594-1601, doi: 10.1109/IJCNN.2017.7966041 .

[42] N. Bhakt, P. Joshi, and P. Dhyani, "A Novel Framework for Real and Fake Smile Detection from Videos," in *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, 29-31 March 2018, pp. 1327-1330, doi: 10.1109/ICECA.2018.8474594.

[43] A. Goldberger *et al.*, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals," *Circulation,* vol. 101, pp. E215-20, 2000, doi: 01/10.1161.CIR.101.23.e215.

[44] H. Dose, J. S. Møller, S. Puthusserypady, and H. K. Iversen, "A Deep Learning MI - EEG Classification Model for BCIs," in *2018 26th European Signal Processing Conference (EUSIPCO)*, 3-7 Sept. 2018, pp. 1676-1679, doi: 10.23919/EUSIPCO.2018.8553332 .

[45] E. S. Dan-Glauser and K. R. Scherer, "The Geneva affective picture database (GAPED): a new 730-picture database focusing on valence and normative

significance," (in eng), *Behav Res Methods*, vol. 43, no. 2, pp. 468-7, 7 June 2011, doi: 10.3758/s13428-011-0064-1.

[46] F. Al-Shargie, U. Tariq, M. Alex, H. Mir, and H. Al-Nashash, "Emotion Recognition Based on Fusion of Local Cortical Activations and Dynamic Functional Networks Connectivity: An EEG Study," *IEEE Access*, vol .7, pp. 143550-143562, 2019, doi: 10.1109/ACCESS.2019.2944008.

[47] M. Moussa, U. Tariq, H. Al-Nashash, and F. Al-Shargie, "Discerning Genuine and Acted Smiles Using Neural Networks," in *2019 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, 10-12 Dec. 2019, pp. 1-4, doi: 10.1109/ISSPIT47144.2019.9001848 .

[48] A. Delorme and S. Makeig, "EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis," *Journal of Neuroscience Methods*, vol. 134, no. 1, pp. 9-21, 2004, doi: https://doi.org/10.1016/j.jneumeth.2003.10.009 (accessed September 2019).

[49] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms," 2017, *arXiv preprint arXiv:1708.07747*.

Vita

Mostafa Mohamed Moussa was born in 1997, in Cairo, Egypt. He received his primary and secondary education in Abu Dhabi, UAE. He received his B.Sc. degree in Electrical Engineering from the American University of Sharjah in 2018. During his bachelor’s study, he co-authored one paper which was presented in an international conference.

In September 2018, he joined the Biomedical Engineering master's program at the American University of Sharjah as a graduate research assistant. His research interests are in human augmentation, human anatomy and physiology, machine/deep learning, brain research, and genetic engineering.