Enhancing machine vision using human cognition from EEG analysis

by

Alankrit Mishra

A thesis submitted in partial fulfillment of the requirements for the degree of
Master of Science
in
Computer Science
(Specialization in Artificial Intelligence)

to the

Department of Computer Science
and Faculty of Graduate studies

Lakehead University

Committee in charge:

Supervisor:	Dr. Garima Bajwa (Dept. of Computer Science, Lakehead University, Thunder Bay, ON)
Internal Examiner:	Dr. Thiago E. Alves de Oliveira (Dept. of Computer Science, Lakehead University, Thunder Bay, ON)
External Examiner:	Dr. Yimin Yang (Dept of Electrical and Computer Engineering, Western University, ON)

Enhancing machine vision using human cognition from EEG analysis

© Copyright 2022

by

Alankrit Mishra

Lakehead University, Thunder Bay, ON

Canada

Abstract

ENHANCING MACHINE VISION USING HUMAN COGNITION FROM EEG ANALYSIS

Visual classification is the perceptible/computational effort of arranging objects and visual contexts into distinct labels. Humans and machines have mastered this advanced problem in their own varied contexts. However, certain aspects inherent to the variability of the visual stimuli present need to be overcome. This thesis analyses the different dimensions of visual classification using a combination of human cognition and machine vision. Thus, it presents novel approaches to joint multimodal learning for machine-learnt visual features and features learnt using brain-visual embeddings via EEG.

First, the thesis proposes a pipeline structure of grayscale image-based encoding of brain-evoked EEG signals as a spatio-temporal feature for improved data convergence. This encoding results in a new benchmark performance of 70% accuracy in multiclass EEG-based classification (40 classes, a challenging benchmark EEG-ImageNet dataset) due to the inclusion of a stretched spatial space that accommodates all the responses of visual stimuli in a single visual sample. As a second contribution, it develops a new approach for cross-modal deep learning based on the concept of model concatenation. This unique model uses a mixed input of deep features from the image and brain-evoked EEG data encoded with a grayscale image encoding scheme. This strategy led to the high joint-learning performance in EEG-Image-based multimode fusion with an accuracy of 95%. Finally, this research found that the automated visual classifier (visual data represented by the corresponding brain-evoked EEG responses to stimuli) is enhanced when a stimulus is an actual object in a three-dimensional

space instead of an image of the same object in a two-dimensional space.

Thus, the thesis demonstrates that enhancing the distinction of visual stimuli features using a joint perception of humans and machines is the way forward to a reliable solution for visual classification.

Dedication

This thesis is dedicated to family and friendship. I have special gratitude for my loving parents, who have never failed to provide me with financial and moral support, for putting all my needs before their own and for teaching me the purposeful meaning of life. My baby brother, who always looks up to me for emotional security, has given me the source of my most profound and lasting gratification: his sweet affection. My girlfriend, who encouraged my pursuit of higher education and research and always stood by me like an enchanted light in my darkest moments.

Last but not least, I dedicate this work to my close friends for keeping me sane with all the jokes and reality checks and for helping to shape who I am today.

Contents

List of Figures

List of Tables

Acknowledgments

Words cannot express my gratitude to my supervisor, Dr. Garima Bajwa, for her invaluable patience and guidance throughout my journey of the thesis. I am deeply indebted to her continuous support, from advocating my research proposal and suggesting pragmatic approaches to meticulous editing and feedback on my writing. I would also like to thank Dr. Thiago E. Alves de Oliveira for providing me with the opportunity to gain first-hand experience in academic research paper writing and later present it in a conference, which greatly influenced the development of my thesis.

Many thanks to fellow graduate scholars for working with me as a team on different research projects through the course of my masters program. A special thanks to Nikhil Raj for spending countless hours brainstorming approaches and contributing to a pivotal idea that led to a breakthrough in this study. I am grateful to my colleagues at the Student Success Centre for providing me with a friendly, welcoming and supportive work-study environment, allowing me to complete my thesis efficiently.

Lastly, this endeavour would not have been possible without the generous financial support of Lakehead University's Faculty of Graduate Studies, Faculty of Science and Environmental Studies, and Department of Computer Science, who sponsored this research with the following grants:

Faculty of Science and Environmental Studies Award

International Graduate Student Research Award

Faculty Research Scholarship

Chapter 1

Introduction

1.1 Overview

The phrase "machine perception" refers to a computer system's capacity to understand data comparable to how people use their senses to relate to the world around them[144,118]. The purpose of machine perception, which is considered a type of artificial intelligence, is to provide the computer system with the appropriate hardware and software to identify pictures, sounds, and even touch in a way that improves the interactivity between human operators and machines. Machine perception advancements encompass online and offline applications, allowing robots to provide more assistance to operators.

Improved machine perception could be tremendously valuable in a variety of scenarios. For example, a doctor wants to get the patient's health history. It may extend beyond the records of their registered healthcare system to include any health-related data involving that person found in the old paper-based health information databases that the computer can digitally scan. As a result, an AI machine with advanced perception and computation can provide details if the patient was involved in a traffic accident several years ago or was

treated for a specific illness or injury while traveling, allowing the physician to evaluate the patient's current situation more competently.

Machine perception enables the computer to leverage sensory input and traditional computational techniques to acquire information more precisely and display it in a user-friendly manner. These include computer vision, machine hearing, machine touch, and machine smelling[118]. The following are the essential categories of machine perception that artificial intelligence or AI often draws upon several disciplines and is related to but distinct from general intelligent systems, natural language processing, and neural networks.

- **Machine vision.** Machine vision studies involve gathering, processing, analyzing, and comprehending high-dimensional input from the actual environment to generate numerical or symbolic information, such as judgments. Today, various uses of machine/-computer vision include facial recognition, geographic modeling, and even aesthetic evaluation[38].

- **Machine hearing.** Machine hearing is the capacity of a computer or machine to receive and analyze auditory data, such as music or speech[155]. It is also known as machine listening or computer audition. Some of the various applications of this subject include voice synthesis, speech recognition, and music recording and compression[100]. Furthermore, this technology allows the computer to emulate the human brain's ability to focus on a specific sound while filtering out background noise and competing stimuli. This particular talent is known as "Auditory Scene Analysis" which allows the system to partition many concurrent streams[154,155]. Machine hearing is used in many daily products such as telephones, voice translators, and vehicles.

- **Machine touch.** A machine or computer processes tactile information in the domain of machine perception known as "machine touch". Applications include dexterity and

tactile awareness of surface characteristics, where tactile information can promote quick reactions and environmental interaction[50].

This chapter focuses on the history and development of the visual perception of machines (or machine vision) and is organized as follows: The fundamental difference between human and machine perception is discussed in Section 1.2. To extract human perceptual awareness, we explore several strategies of the brain-computer interface in Section 1.3. Section 1.4 discussed the most popular EEG feature extraction and selection approaches, as these characteristics are later used for cognitive tasks such as classification and prediction. Section 1.5 then gives an overview of how humans and machines perceive visual information and also introduces the seed of the thesis by briefly reviewing the underlying studies carried out to improve automated visual classification using machine learning with temporal and spatial modalities. Finally, Section 1.6 structurally constructs all the research frameworks designed in this thesis.

1.2 Human vs. Machine Perception

Given that the dynamics of the human brain has evolved over millions of years, human perception is incredibly sophisticated and intuitive. It is intriguing to note that the human brain performs better when the task is perceptual rather than computational. Although artificial intelligence, or machine intelligence, has a far shorter evolutionary period than humans, it has already outperformed the human brain in computing through math and logic. Hans Moravec states a well-known AI paradox: "It is comparatively easy for computers to exhibit adult-level performance on intelligence tests or play checkers, but it is difficult or impossible to give them the skills of a one-year-old when it comes to perception and mobility"[114]. De-

spite technological advances in computational power that have yielded significant results in cognitive tasks that require extensive mental effort for human participants, such as advanced strategy games[174] or machine translation[177], machine perception continues to evolve at a far slower rate and struggles to perform perceptual tasks that appear intuitive to humans but may have ambiguous or ill-defined "ground truth," such as medical image interpretation[137].

Due to a lack of situational knowledge, vast volumes of irrelevant data are processed, resulting in the "curse of dimensionality" and computational explosions[131]. As a result, autonomous system designs are not robust, and machine learning approaches remain fragile. Phoha[131] discussed the technical oversight in the construct of current machine perception, which lacks context learning and cross-modality fusion.

- **Context Learning** Over the last decade, many research approaches that enable machines to derive the current operational context from input data have evolved in various disciplines. These include developments in image and scene processing, natural language processing, and cognitive neuroscience based on physics-based, environmentally adaptable sensing models. The context is typically fragmented and ambiguous between modalities and applications. For example, in image processing, it often takes the visual scene to be the context for object recognition or image classification; in human-machine interactions, context is generally the verbal semantics via which people communicate the present command to autonomous systems; and context is frequently modelled using attention and memory in cognitive sciences.

  Blasch et al.[18] focused on reducing the impact of context on the feature space in their study by using statistical detection and classification techniques that are context invariant. However, feature extraction algorithms frequently do not respond well to the extremely non-linear and non-stationary influences of the operating environment.

Formalising this strategy, Phoha et al.[132] proposed a mathematical characterisation of machine extractable context, applicable to all relevant sensing modalities for an application, enabling contextual decision-making in dynamic data-driven classification systems. Further research into merging data-driven and model-based approaches for context learning, discovering novel contexts that were not labelled during the training phase, and dynamic modelling of context drift remain promising research areas for increasing machine perception and learning.

- **Cross-modality Fusion** Intelligent machines rely on a sensing infrastructure for measurement, communication, and computing to observe the progression of physical dynamic processes in their operating environment[131]. Sensors need to interact with observed phenomena to provide time series data (temperature, pixel intensity, etc.) of the evolutionary processes generated by physical stimuli[18]. The multivariate information space formed by these time series is an amorphous computing environment with a high degree of redundancy. Furthermore, sensors of various modalities exhibit contextually variable performance in noisy settings. It is vital to fully leverage their heterogeneity to extract reliable data from multi-modal sources by combining complementary information from different modalities[131,53].

  The current literature on information fusion makes only a rudimentary use of heterogeneous modality[53]. Decision-level fusion algorithms often combine the probability distributions produced independently by each sensor into a single decision. For humans, this is comparable to incorporating the senses of a blind and a deaf person rather than coordinating a single person's visual and auditory sense perceptions. This approach destroys causal information on feature-level relationships. Machine perception methods are required to use the multi-modal data at the feature level properly. Auto-

mated algorithms that leverage cross-sensor interdependence are required, as humans can effectively coordinate their visual and aural input to distinguish the two scenes or audio[132]. These algorithms will use domain-specific non-linear, and non-stationary phenomena such as phase changes generated by physical stimuli. Addressing the scientific and engineering issues of generating actionable knowledge from multiple sources of electronic input with varying modalities and contexts is critical to modifying the behaviours of machine intelligence.

Scientific fields such as vision science have always relied heavily on methods and procedures of psychophysics, but there is now a growing appreciation for them by machine learning researchers, sparked by a growing overlap between biological and artificial perception[60,188,45,142,139]. Machine perception guided by behavioural measurements, as opposed to guidance restricted to arbitrarily assigned human labels, has significant potential to fuel further progress in artificial intelligence[44].

Towards better HCI

Perceptual user interfaces that offer a human-understandable representation of complicated data sources and improve human-computer interactions (HCI) are another technological challenge that will accelerate and support advances in machine perception and cognition of sensed information. In the next section, we will look at a form of HCI called the Brain-Computer Interface, which sets a new pathway for extracting temporal information from the human brain that can enable the development of machine perception at a human level.

1.3 Human perception using BCI

A promising way to provide essential communication abilities to a person affected by locked-in syndrome is the Brain-Computer Interface (BCI), also known as the Brain-Machine Interface (BMI). This technology allows humans to interact with their surroundings without involving peripheral nerves or muscles. This technology has been used primarily to create assist devices in the medical industry. Every BCI system comprises five components: brain activity measurement, pre-processing, feature extraction, classification, and translation into a command[108].

Branches of BCI Technology

BCI Frameworks can be separated on the basis of three schemes: recording technique, operation method, and dependability, which can be classified as dependent or independent. Dependent BCIs need some kind of motor control, such as gaze control[150] from the user or healthy subjects. Motor Imagery (MI)-based BCIs are an ideal example and have been extensively used, whereas independent BCIs do not require any form of motor control by the user and are suitable for stroke patients or severely impaired patients. A successful independent BCI system based on steady-state visual evoked potential (SSVEP) was proposed in 2016[158] to identify two different goals.

Using recording methods, BCI can be classified as invasive and non-invasive[5,136]. Invasive BCIs require microelectrode arrays to be implanted inside the skull, unlike those placed on the scalp in the case of non-invasive BCI. Two common invasive modalities are intracortical recording and electrocorticography (ECoG). Non-invasive modalities are EEG (Electroencephalogram), MEG (Magnetoencephalography), PET (positron emission tomography), fMRI (functional magnetic resonance imaging) and fNIRS (functional near-infrared

spectroscopy)[99,5]. Most BCI researchers prefer a non-invasive approach to avoid the risk of surgery. The selection of the measurement method depends on various parameters, for instance spatial resolution, temporal resolution, invasiveness, measured activity, cost and portability[136].

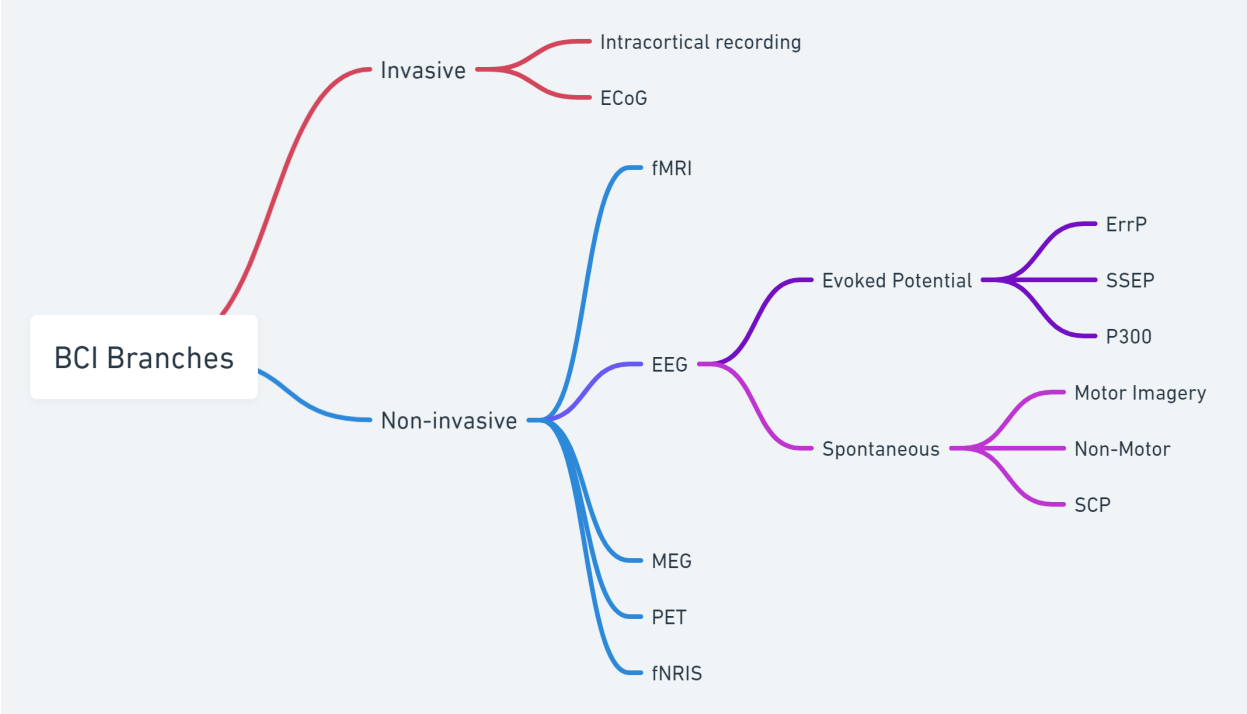


Figure 1.1: A flow-diagram to illustrate the BCI types according to recording

Due to its high temporal resolution, inexpensive cost and mobility, EEG is the most widely used non-invasive modality in BCI to elicit different control signals such as SCP, SSVEP, MI, ErrP, and P300[5]. The EEG measures voltage changes caused by the flow of ionic current in brain neurones during synaptic excitations[13]. To acquire brain impulses, electrodes are placed on the scalp. For various EEG headsets, the electrode number ranges from 1 to 256. The amplitude of the recorded EEG signal is the voltage differential between the active and reference electrodes over time. EEG amplitudes typically range from -100

to +100 microvolts. EEG signals can be divided into different bands, each of which has a unique biological meaning. Figure 1.1 shows the different types of BCI as their recording technique.

There can be two modes of operation in BCI systems, synchronous and asynchronous. In a synchronous system, the user-system interaction is carried out within a certain period of time. Asynchronous BCI, on the other hand, allows the patient to generate a mental task to interact with the program at any time. Synchronous BCIs are much easier to design, but not as user-friendly, compared to synchronous BCIs[164].

EEG control signals

BCI tries to identify the neurophysiological signals of a subject to connect a command to each of these signals. Some of these control signals are easy to recognise and much easier to control by the user. Some commonly used EEG control signals include SCP(slow cortical potential), P300, MI, MRCP (movement-related cortical potential), ErrP (Error-related potential), SSVEP, SSAEP (steady-state auditory evoked potential) and SSSEP (steady-state somatosensory evoked potential)[136]. When a person receives a periodic stimulus, such as a flashing visual or amplitude modulated sound, steady-state evoked potentials (SSEPs) occur[43,119]. The stimulation frequency or harmonics are equal to the frequencies of the EEG signal, which is an essential feature of SSEP. Each SSVEP-based BCI requires a set number of visual stimuli corresponding to specified BCI output instructions. BCI devices based on P300 are based on sequential flashing stimuli. These stimuli might be used in various BCI applications, including directing a robot arm, cursor, or mobile robot. P300 is created in the parietal regions (Pz) of the brain when the stimulus is given for 300 ms[138,98]. Even with less likely stimuli, the response's peak amplitude has been observed to be substantially big-

ger. Sensorimotor rhythms (SMR) are generated in the motor regions of the brain by motor imagery (MI)[89]. The left and right hands initiate an activity. MI is formed by the central parts of the brain (C3, C4), whereas Cz generates the image of foot movement. Due to the proximity of the relevant brain areas, left and right foot motions are nearly challenging to differentiate in EEG.

EEG-based real-world BCI applications

Human brain impulses can be recognised and converted into device commands to operate assistive devices using BCI technology. The scope of this technology has expanded beyond medical applications to include non-medical uses.

BCI Controlled Wheelchair A BCI wheelchair can improve the quality of life and autonomy of patients with motor neuron disorders (MND), such as amyotrophic lateral sclerosis (ALS). This advancement enables disabled people to manage the wheelchair using their brain activity, giving them autonomy as they travel through an experimental environment. Four types of EEG control signals are used to handle BCI wheelchairs, which are MI[89,172,163], P300[138,98], SSVEP[43,119] and hybrid[94,25]. The feature extraction methods are quite heterogeneous. However, common spatial pattern (CSP) is the most commonly used EEG feature method in BCI wheelchair applications[25].

BCI Spellers Farwell and Donchin[48] introduced the matrix speller, a P300 speller, in 1988. It was the first BCI speller, with a maximum accuracy of 95% and a speed of 12 bits per second. Blankertz et al.[159] also published a Hex-O-Spell that is based on imagined movement. The speller outperformed the traditional matrix speller in terms of performance. Oct-O-Spell, a novel MI-based speller, was introduced, featuring an octagon divided evenly

into eight sections. These sections comprised a total of 26 letters, characters, numerals, or symbols[26].

BCI Biometrics Biometrics, such as iris, face, and fingerprint identification, are commonly used to prevent security breaches. An EEG system-based biometric has been found to have the distinct benefit of being nearly difficult to replicate[4]. Ruiz Blondet et al.[140] investigated the stability of EEG brain waves in 15 human volunteers.

BCI Emotion Recognition Data collection from brain signals associated with human emotion is a critical step toward emotional intelligence. Detecting mood changes in EEG data has recently gained popularity among BCI researchers, who have conducted various investigation of emotion recognition during the past 20 years[133,178]. The average accuracy and information transfer rate (ITR) obtained were 91.1% and 85.80%, respectively.

BCI Virtual Reality and Gaming Existing BCI-based video game prototypes are based on three different BCI paradigms: steady-state evoked potential (SSVEP), P300 event-related potential (ERP), and mental imagery (MI). The authors found a 66% mean accuracy for Mind Game (specifically, this was the rate at which the correct target was selected out of 12 possible targets). Other P300-based BCI games have been proposed in the literature[6],[56]. An intriguing use of this type was shown at the Cybathlon 2016. Eleven people with tetraplegia fought each other in a virtual world where their avatars raced through an obstacle course. The results ranged greatly among the 11 individuals, as predicted.

BCI Robotic Arm Yang C. et al.[181] demonstrated a shared control system for mind control of a robot manipulator by merging a SSVEP-based BCI with visual servoing (VS)

technology. Duan et al.[42] presented a hybrid BCI system consisting of SSVEP and MI. The studies conducted by four participants yielded an average accuracy of 85.45%.

1.4 EEG-based feature extraction strategies

Most of the discrete and non-redundant information within the EEG is extracted after the noise removal phase using different feature extraction techniques. The most notable feature extraction using EEG-based BCIs are those that work in the time-domain, frequency-domain, and time-frequency domain.

Time Domain Autoregressive modelling (AR), a standard method for extracting features from time-domain data, involves regressing the current observation in a series linearly against one or more preceding observations. The AR model has been used as a feature extraction method in EEG-based BCI systems in numerous recent articles[83,189,28]. Combination techniques for feature extraction are also used, where each feature vector comprises of AR coefficients and rough entropies. Due to their high resolution, softer spectra, and flexibility to apply to brief data segments, researchers favour AR models. Higher model orders increase noise, whereas lower model orders do a poor job of representing the signal. Therefore, choosing the proper AR modelling order is still a challenge. A Bayesian information criterion, a final prediction error, or an Akaike information criterion (AIC) are commonly used to estimate the modelling.

Unlike any fixed modelling order, Atyabi et al.[10] proposed that a sufficient combination of AR features produced from several AR modelling orders is a representation of the underlying signal. The analysis of respiratory rate variability from EEG[61], adaptive Hermite decomposition[157], and RR time series[165] has been used to extract characteristics for the

identification of the state of sleepiness from EEG signals. The Higuchi technique[8,77] has been used to extract the feature from the fractal dimension of raw signals in the context of emotion recognition using an EEG signal. Aydin et al.[11] suggested using logarithmic energy entropy to extract EEG features, which may be used to determine how much randomness is contained in the signal. Zarei et al.[183] developed a hybrid feature extraction technique that combines PCA (principal component analysis) and the cross-covariance technique to extract discriminatory information from mental states of EEG.

Frequency Domain Several EEG-based BCIs use features from frequency domain analysis. The fast Fourier transform (FFT)[70,40,21,180], power spectral density (PSD)[32,27,116,130,15,97,121,29], band power[105,143,82], and spectral centroid[117] are examples of methods based on the frequency domain. In their work, Rashid et al.[136] observed that FFT and Welch's technique could be used to determine the PSD of a signal. In contrast to the FFT, Welch's approach minimizes PSD artifacts but results in sub-par frequency resolution. Local feature scale decomposition is another frequency-domain-based feature extraction method that can compute the PSD without using an FFT. This procedure divides the raw data into underlying chunks corresponding to the primary signal's characteristics. Local characteristic-scale decomposition[96] is another frequency domain-based feature extraction method that does not use FFT to calculate the PSD. This process separates the raw data into constituent parts representing the characteristics of the primary signal. The signal is broken down into different frequency components using Fourier analysis, and their relative intensities are calculated. Traditional spectrum analysis approaches are not appropriate for obtaining significant and crucial information because of the non-stationarity and non-Gaussianity aspects of the EEG signals. The most discriminative spectral characteristics are extracted from the PSD of the EEG signals by the Gursel Ozmen et al.[64] frequency domain-based feature extraction method. New

spectral estimators, the quantile periodogram and the lasso quantile periodogram, based on quantile regression and L1-norm regularisation, respectively, were reported by Meziani et al.[109].

Time-Frequency Domain Sometimes, the lack of temporal features renders the use of spectral characteristics for feature extraction ineffective. Similarly, time-domain interpretation can occasionally overlook spectral traits that the classifier might find crucial. Time-frequency analysis, which leverages both the time domain and the frequency domain, is thought to be able to overcome the limitations of a single domain that is either time domain or frequency domain. This strategy is better for the efficacy of EEG-based BCIs. In EEG-based BCIs, a variety of time-frequency-based feature extraction techniques have been used. The most popular techniques are wavelet packet decomposition (WPD)[19,39,175], continuous wavelet transform (CWT)[20,87,73], discrete wavelet transform (DWT)[63,14,40,75,95], and short-time Fourier transform (STFT)[151,31,65,162]. Deep learning techniques have been used to create spectral images produced by CWT[123,104] and STFT[35] that can be categorized. Mammone et al.[104] proposed an EEG-based motor planning exercise in which a time-frequency map created by beam forming and CWT is used as input to the CNN. Since there is important information contained in several EEG bands[79], decomposition techniques like DWT and WPD (wavelet packet decomposition) are effective because they can decompose the brain waves at multi-resolution and multi-scale[90]. Additionally, they can extract dynamic features, which is important for EEG signals because of their non-stationary and non-linear properties[79]. To get the highest level of accuracy, Kevric and Subasi[79] looked into three different decomposition approaches, namely WPD, EMD (Empirical Mode Decomposition), and DWT. The decomposed EEG sub-bands have been used to derive higher-order statistics (HOS) features. When compared to WPD, DWT coefficients have a lesser frequency resolution, but HOS can

make up for wavelet strategy shortcomings.

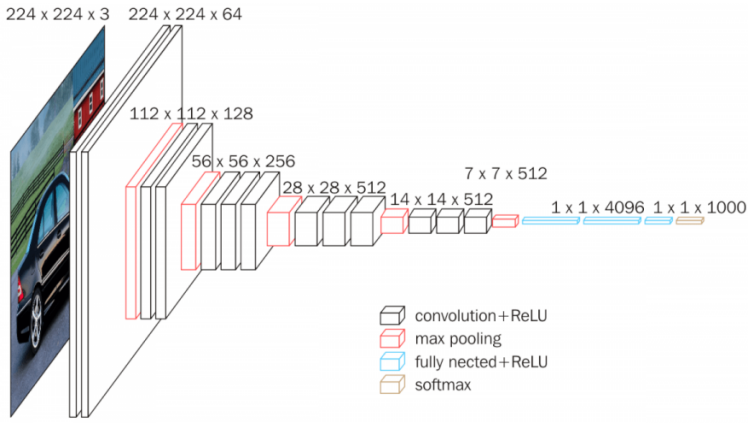
For the purpose of extracting features, Zhou et al.[191] coupled the use of DWT with the Hilbert transform (HT). The wavelet envelope of the decomposed sub-bands was generated using HT after the EEG data was decomposed using DWT. They used both time-series and envelope information, which helped them get the best accuracy possible. Wavelet packet analysis (WPA) was suggested by Göksu[59] as a method for identifying features in an EEG-SCP response, looking closely at the WPA sub-images using log energy entropy. Yang et al.[179] suggested Fisher wavelet packet decomposition (WPD)-CSP to extract characteristics. In this method, EEG channels are broken down by WPA, the average power of each sub-band is computed, and then CSP is applied to the sub-bands that have been chosen.

1.5 Visual perception in humans and machines

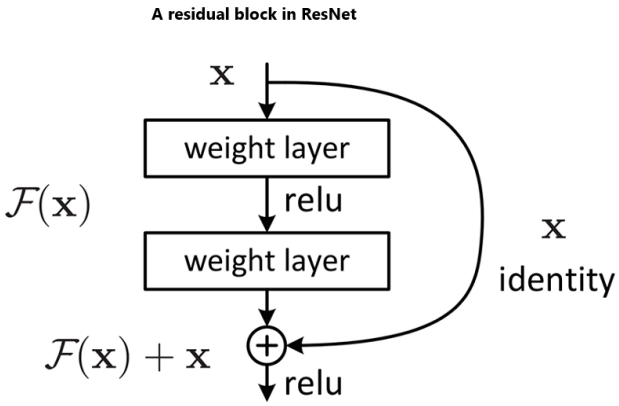
Human visual perception is both beautiful and complicated. It started with microscopic creatures evolving a mutation that made them light-sensitive. There is an abundance of species on the planet, many of which have incredibly similar visual systems. They consist of the eyes to capture light, receptors in the brain to access it, and the visual cortex to analyze it. This visual system has been genetically developed and balanced to help humans do something as basic as appreciate a sunrise. However, in the last three decades, scientists have made efforts toward extending visual context to machines. Around 1816, the first form of photographic camera was constructed, consisting of a small box containing a piece of silver chloride-coated paper[58]. The silver chloride darkened in the light-exposed portions when the shutter was opened. Two hundred years later, we have advanced devices that can take photographs in digital form[51], allowing machines to precisely mimic how the human eye perceives light and color, but it is considerably more challenging to understand the context

of the picture. The fact that the human brain recognizes visual stimuli just by looking at them is by virtue of a million years' worth of evolutionary context to help us immediately understand this. However, a computer does not have that same advantage. A machine algorithm sees the image as a vast array of integer values representing intensities across the color spectrum, but without context, only a tremendous amount of data. It turns out that context is the key to enabling computers to grasp visual information, similar to human cognition.

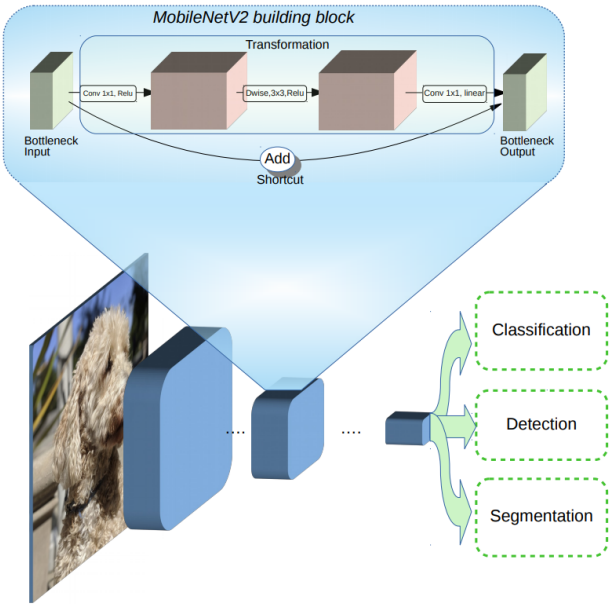
Furthermore, machine learning methods enable artificial intelligence to properly teach the context of a visual, allowing an algorithm to comprehend all the numbers in a particular pattern. A convolutional neural network (CNN)[124] is a specific form of artificial neural network that functions by dividing an image into smaller groupings of pixels known as filters. Each filter is a pixel matrix, and the network performs a series of computations on these pixels, comparing them to pixels in a pattern the network seeks. It can recognize high-level patterns, such as rough edges and curves, in the first layer of a CNN. As the network conducts many convolutions, it can recognize individual things such as faces and animals. This is achieved by using a large amount of labeled training data. All filter settings, like weights and biases, are randomized when CNN starts. As a result, its early projections are illogical. When CNN produces a prediction against labelled data, it compares how close its forecast was to the image's actual label using an error function. The CNN modifies its filter settings and restarts the operation based on this error or loss function. Ideally, each repetition improves accuracy marginally. Following CNN's invention, many prominent deep learning architectures such as VGG16[145], ResNet[67], MobileNet[71], and EfficientNet[153] were built, with CNN serving as the backbone layer of their design. These models evolved to perform different computer vision tasks such as object identification, recognition, depth estimation, and most notably compared performance on digital image classification tasks on ImageNet[37],



(a) VGG16[145] architecture



(b) ResNet[67] architecture



(c) MobileNetv2[71] Architecture

Figure 1.2: Architecture designs of popular state-of-the-art CNN model.

a large-scale picture dataset. Figure 1.2 displays some popular CNN architectures.

Although machine vision has achieved better accuracy in visual perspective tasks, is it robust enough to compete with the human visual system? Prior studies claim that machines' visual learning methods are different from how humans infer visual cues[57,33,101,49]. Some academics argue that deep neural networks, such as CNN, cannot achieve human-like intelligence

because machine vision algorithms cannot bridge the recognition gap of human sensitivity to exact feature configurations[169]. Another study by Funke et al.[54], on the contrary, states that human bias can inhibit the interpretation of visual features such as contours, making machine intelligence more appropriate for the task. Makino et al.[103] also tested human and machine visual perception for detecting soft lesions in breast cancer medical imaging and observed that both radiologists (human perception) and CNN recognized distinct regions of interest on low pass filtering. This establishes the practice of independently considering human and machine visual perception because both contribute relevant, yet isolated, information.

The work in this thesis builds upon prior groundwork, looking for ways to incorporate human perception into machine learning to optimize machine perception's robustness. We can now extract temporal data from the human brain using the BCI outlined in the preceding section, making it a human prospective-evoked modality input to machine learning models. We have chosen the visual classification problem as our case study. Through the course of subsequent chapters in this thesis, experiments involve temporal modality as a form of EEG signal data and spatial modality as a form of image data to improve the performance of automated visual classification. In the end, a proposal for a joint representation of cross-modal fusion (of data from both temporal and spatial domains) is evaluated.

The practical implications of this study are to take a significant leap forward in the collaboration of neuroscience and artificial intelligence. We aim to explore a new and direct form of human-computer interaction (a new vision of the "human-based computation" strategy) for automated vision tasks.

Integrating the BCI evoked human visual perception will greatly improve the performance of BCI-based applications and enable a new form of brain-based image automated annotation[81] compared to the current state where a lot of manual effort is required to annotate or label a visual stimulus.

Second, joint representational learning will especially contribute to imaging systems (e.g. diagnosis, surveillance, object tracking) as the AI vision system will learn directly and integrate with human observative decisions. It can provide vital suggestions like a new region of interestMakino et al. [103] that was not perceived by the human eye due to biasFunke et al. [54], and later can learn to filter out noninteresting regions using temporal and spatial joint learning.

1.6 Structure of the thesis

There are five chapters in this thesis. The first chapter provides an overview of the problem statement.

Chapter 2 presents all the classical approaches to classification performed on a two-class visual dataset. We also study a machine-computed analysis of the spectral findings of the EEG claimed by Marini et al. [107] here.

Chapter 3 discusses pipeline-based deep learning algorithms for visual classification using EEG-ImageNet [149,127], a 39-class visual dataset. We investigate cross-modal feature extraction strategies for EEG data to enhance the feature space and improve classification.

Chapter 4 proposes a novel method to perform automatic visual classification using image and EEG data as input for the same visual stimulus. The deep learning frameworks established in this work are used to reevaluate the claims of the spectral discoveries of EEG made by Marini et al. [107].

Chapter 5 concludes our thesis, summarising the contributions of our approaches and the future scope of this research.

All references used are presented at the end of this thesis.

The implementation of code, the design of the model, the abbreviations, and the resources are provided in the Appendix.

Chapter 2

Visual classification using classical Machine Learning classifiers

2.1 Introduction

Automation through machines plays a vital role in the era we live in. We continually seek technology innovation to advance machine perception for real-time decision-making in numerous applications[131]. Since machines use binary logic, designing machines to behave and make decisions like humans is hard. With the invention of many state-of-the-art paradigms like Human-Computer Interaction (HCI), Brain-Computer Interfaces (BCI), and Artificial Intelligence (Computer Vision, Natural Language Processing, and Machine Learning), we now have black box thinking machines. Although there is expanding research to understand and interpret machine perception and learning[113], there is still a lot to uncover and tune additional learning parameters other than the weights and biases employed so far.

On the other hand, humans excel in comprehending complex tasks such as classification, detection, and intuition, which machines cannot match. Although the recent discovery of

Deep Learning has resulted in significant improvements in classification performance, their generalization capabilities are not human. They learn a discriminative feature space that depends on the training dataset rather than generic principles. Previous studies have also demonstrated that incorporating human crowdsourcing as part of the training process improves machine performance[173]. We are interested in one significant component of machine perception, i.e., visual perception. By extracting characteristics from single or multiple pictures or a sequence of images, vision algorithms assist machines in comprehending the context of visual input in areas such as object detection, motion tracking, and gesture recognition (in a video). As a result, they would provide a meaningful interpretation of the world and enable actionable decisions.

In the last ten years, deep learning models like Convolution Neural Network (CNN)[124] have taken a giant leap in performance, and they have achieved nearly perfect accuracy with state-of-the-art models. However, the model is generally confined to supervised learning and a specified training dataset, making it bulkier and more reliant on a large number of data collections. This constraint can be addressed using self-supervised learning by transferring human cognitive ability to generalize tasks. Recent advances in image analysis have effectively used transfer learning in applications such as robot training using simulations[141]. This research explored one of the methods for establishing transfer learning called the Brain-Computer Interface (BCI). These systems allow communication between the brain and various machines[176]. They operate in three stages: collecting brainwave signals, processing them with algorithms, and extracting valuable features from the given model based on the brain signal received. Based on previous development history, the BCI can be segregated into three types:

- Non-invasive – No exposed brain, only scalp. E.g., Electroencephalogram (EEG)

- Semi-invasive – Electrodes exposed to the brain partially.
- Invasive – This procedure includes direct implant of the electrode to the cortex to measure neuron activity.

As per the review from Chapter 1 we established that an electroencephalogram (EEG) signal is the safest way to collect brainwave data. We propose using various cognition-based computations with human brain wave input (EEG signals) to improve visual classification. However, it is vital to understand that, unlike machine vision algorithms, the visual stimuli of matter for the human brain are not limited to just two-dimensional images but can be objects in the real world itself. Hence, the visual stimulus can be classified into two forms:

- Real Objects - stimulus as tangible solids that can be interacted with in the real world.
- 2D images - flat displays with inconsistent depth indications. 2D graphics can be printed, though they are often shown via a monitor or projection screen[146]. They can vary in iconicity or how closely the image matches the real object[55], from line drawings or clipart to copies of actual photographs.

The core objective of this work is to extract features from EEG signals acquired from the human brain when the user observes the item in three dimensions (real world) and then in two dimensions through images. The features can then be processed in various ways, including visual mapping to time and frequency and other heuristic techniques that will be discussed further in later sections. The information acquired from EEG signals will subsequently be compared with spatial visual features generated by machine vision extractors for image classification generalization.

2.2 Related Work

When it comes to machine learning for visual recognition, the top-performing models are built on Convolutional Neural Networks (CNN) like the LeNet[86], and its potential has been proved on well-known datasets like the MNIST[85], and ImageNet[37]. Development in this field has stalled for over a decade since these models are data-driven and require much computational power. The ImageNet classification benchmark was saturated after Tan and Le[152] introduced Efficient-Net.

The first successful non-invasive BCI work was introduced in 1988[22], and since then, EEG data analyses have contributed to many task-based classifications, and pattern recognition in neuroscience, such as seizure detection[110]. EEG data became popular in machine learning applications like emotion identification[80], where features extracted from raw EEG time-series signals such as average band power and power spectral density were used with classifiers like SVM[185].

Türk and Özerdem[167] developed an intriguing classification method which demonstrated how an EEG signal can be categorized using a scaleogram image of a signal wave as a feature and wavelet processing. We attempted to replicate this strategy in our experiment by producing a scaleogram of an EEG signal from our dataset.

Visual information is rich in stimulation and can be captured/monitored via EEG signals. Visual stimuli studied through EEG signals revealed fascinating results such as image reconstruction[78] and image tagging[81]. Following on with this idea,[149] used a Recurrent Neural Networks (RNN) technique for learning visual stimuli induced EEG data and determining a more compact and intelligible representation of such data. They proposed a CNN-based approach for regressing images into the learned EEG representation, allowing for automated visual classification in a "brain-based visual object manifold."

Another study by Fares et al.[46] proposed an EEG-based image classification architecture by merging region-level information with stacked bi-directional LSTMs[68] as a solution. The dynamic correlations concealed in EEG data are captured using stacked bi-directional LSTMs. Both[149,46] discovered that the gamma band signal is essential to achieve strong performance in object classification and plays a crucial role in emotion classification.

According to a neuroscience study[107], EEG signals demonstrated a transient early occipital negative for actual items. It could be due to 3-D stereoscopic differences, as well as a late persistent parietal amplitude modulation consistent with an 'old-new' memory advantage for real things over photographs. Moreover, a neuroimaging study found different neural representations for real, tangible objects versus similar images during hand actions, mainly when 3D cues conveyed important information to grasp[52]. These findings show that real-world items elicit more powerful and long-lasting action-related brain responses than pictures[147]. We aimed to explore whether this approach could be applied to improve visual classification of machines. The same data set by Marini et al.[106] was used for our experiments.

2.3 Dataset

The dataset used for our experiments was a subset extracted from data provided by Marini et al.[106]. The data was collected by capturing the EEG signals of subjects while viewing 192 trial mixes of kitchen and garage items. The actual objects were displayed in 96 trials, whereas photographs of the same objects were shown in the other 96 trials as depicted in figure 2.2. Twenty-four subjects participated in this experiment. Data were recorded with a 128 noninvasive electrode system, as shown in figure 2.1. The sampling rate of 512Hz gave 1434 timepoints of 2800ms (-800 to 2000ms) in total. The stimulus response was recorded from 0 to 800 ms, and the next 800 ms (800 to 1600 ms) were closed eyes before switching

to the next test. Table 2.1 describes the detailed structure of the dataset[106] parameters. The undesirable artifacts were already removed out of the box, and the processed EEG data were used for feature extraction. This dataset was initially used by Marini et al.[107] for their neuroscience study to distinguish EEG signals recorded for a real object from its image.

Table 2.1: Parametric values of the Marini et al.[106] dataset taken for this study

Datasets parameters	**Values**		
Stimulus type	Real/Physical object	Image of the object	Both Image and Real
Total number of trials	2112	2112	4224
Number of classes	2 (Kitchen and Garage)		
Number of subjects	22		
Stimuli per subject	96	96	192
Stimuli per class	48	48	96
EEG recording time for each stimulus	800ms/1600ms*		
Sampling rate of EEG recording	512 Hz		

*The stimuli was shown to subject in first 800ms and
then subject's vision was blocked in next 800ms.

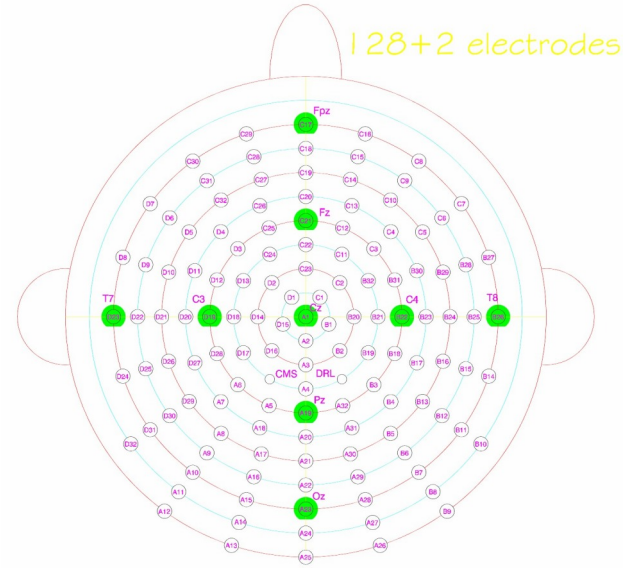


Figure 2.1: The electrodes highlighted in green are chosen for this study.

The dataset also contained images (scripts/stimuli folder) of the objects for which the EEG data were recorded, which we have used to extract features of the image.

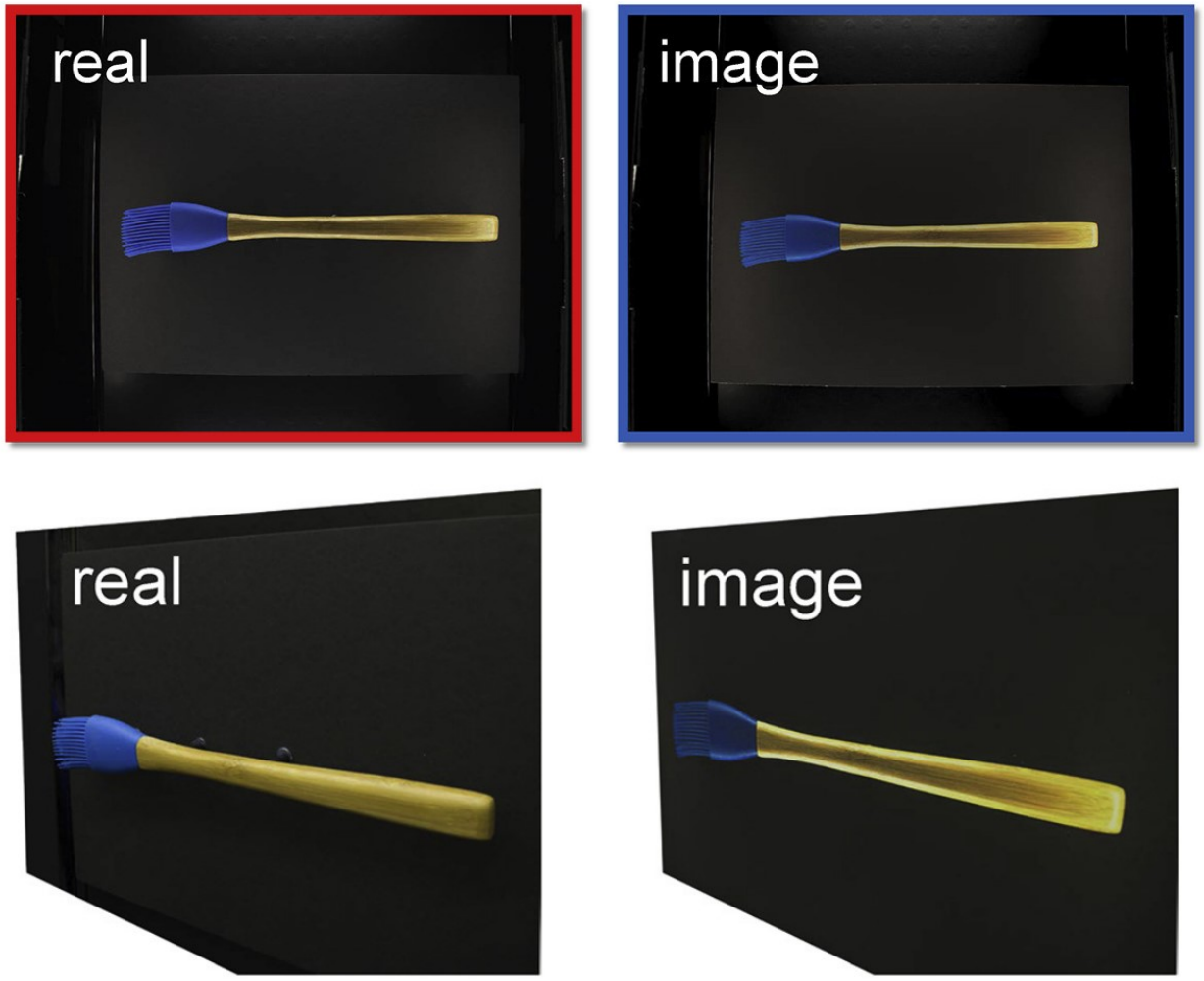


Figure 2.2: An example of real versus. image stimuli as shown in the study by Marini et al. [107]

The raw EEG data had some missing information for two subjects (subjects 2 and 7). Hence, we took data for 22 of the 24 subjects for our experiments. The EEG signals went through a number of preprocessing batches including normalizing using Z-score and then

baseline correction in pre-stimulus period (-200 to 0 ms) to obtain zero-centred values with a unitary standard deviation.

2.4 Methodology

The EEG-based visual classification approach is divided into three components: feature extraction from EEG signals (recorded for image and real objects) and images corresponding to the same visual stimuli; second, encoding and processing of feature vectors to reduce dimensionality; and last, feeding the features along with classes to different classifiers to measure accuracy. The following sections will go through each of these in-depth.

Feature Extraction

The feature extraction for digital images of stimuli involve binarization and applying the HOG filter to compare various results. HOG, or Histogram of Oriented Gradients, is a feature descriptor that reinforces a structure for an image since it computes the features using both the gradient's magnitude and angle. It creates histograms for the image areas based on the magnitude and orientation of the gradient[36]. The binarization process was performed locally by comparing each pixel value with the corresponding threshold[160]. A pixel value more than threshold was marked by 1, while a value less than the threshold was represented by zero. The threshold for each color image was determined by taking the mean of the highest and lowest grey values inside the chosen local window. The contrast was the distinction between the highest and lowest grey values. Finally, the binarized feature vector was extracted by comparing the contrast value to the threshold. The Otsu's thresholding methos[125] was used for binarization.

We used various feature extraction techniques to obtain temporal features from EEG data of stimuli. Once the raw signal was obtained from the selected electrodes (in accordance with the experiments mentioned in Section 2.5), we extracted the following characteristics with the help of EEGLIB library[24]:

- Power Spectral Density (PSD)
- Petrosian Fractal Dimension (PFD)
- Detrended Fluctuation Analysis (DFA)
- Higuchi Fractal Dimension (HFD)
- Band power average (including alpha, theta, gamma, beta)
- Synchronisation Likelihood

We segmented the entire EEG signal recording of each subject into data samples based on the stimulus onset epoch as the total recording period was 2800 ms. After eliminating artifacts and sequencing delays, the actual stimulus-response length ranged from 0 to 1600 ms. As a result, those time points were trimmed to produce a new batch of samples for the feature set. It is also essential to consider the data loss due to low-frequency resolution in small-length signals[171]. To manage this issue, we used multi-taper and periodogram spectral analysis[12]. We created custom band power averages for all bands to create a separate feature set for analysis. We isolated the features for the alpha and beta bands exclusively, as the alpha and beta frequency ranges show high power impedance for motor imagery reflexes[107].

Data Encoding

Since we are working with raw EEG data, it is likely that the feature space is not evenly distributed or that certain features are more relevant than others. We did some encodings on the data to improve the feature space. The dataset stated in section 2.3 is treated as a feature set using the encoding methods described.

Label Encoding: The values in the label set are in a string format. The Label Encoding method encodes the labels from strings to respective integers. It is easier to feed integer arrays to classifiers like SVM as they have high time complexity.

PCA encoding: Principal Component Analysis (PCA figure 2.3) technique is used to see the data spread according to their principal component. It helps to understand the data variance which can be used for training the model. Using this approach, we can decrease the sample data's dimensions, making the model more efficient to train. It also help to tackle any imbalance in the data. We created encoded train and test feature sets to be used with various models.

Feature Selection: Sequential feature selection methods are a greedy search technique to reduce an initial d-dimensional feature space to a k-dimensional feature subspace, where (k<d)[1]. The goal of feature selection algorithms is to automatically choose a subset of features that are most relevant to the problem. We used the Sequential Feature Selection methods from mlxtend library[135] with the wrapper method approach. Two ML classifiers - Decision Tree and Gaussian Naïve Bayes were used. The first method returned nearly 20 important features, and the Naïve Bayes wrapper gave 50. The important features length vary for different experiments and are discussed in section 2.5.

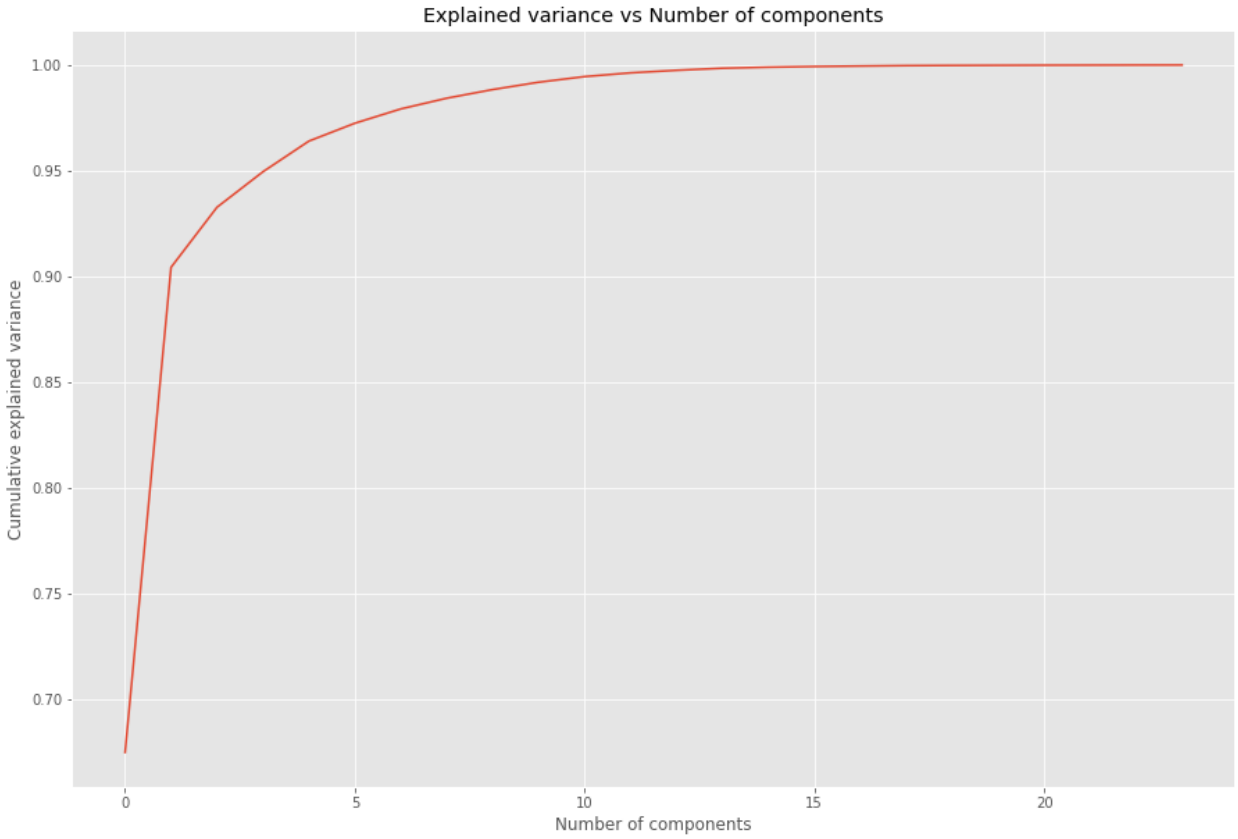


Figure 2.3: Explained Variance VS Number of Components

Classifiers

Our experiment used seven classifiers to classify EEG-based data with raw features, pca encoded features, and SFS encoded features. The classifiers that we used were Decision Tree, Random Forest, K-nearest Neighbour, Support Vector Machine, Artificial Neural Network and Logistic Regression from Sklearn[129]. Furthermore, for image classification, we used CNN as the classifier.

The details of each classifier are as follows:

K Nearest Neighbours Classifier: KNN is a supervised learning model that applies learn-

ing based on each query point's k nearest neighbors, where k is an integer value that the user specifies. Instead of attempting to build a broad internal model, it merely stores instances of the training data. The classification is determined by the simple majority vote of each point's nearest neighbor[187]. A query point is assigned to the data class that has the most representatives among that point's closest neighbors. We took the values of k as five and performed our experiment. The time complexity for the classifier was average.

SVM: Support Vector Machines are supervised learning models of classification. It works on linearly separable data and uses the maximum margin to find decision boundaries to separate them. SVM is also used in time-series modeling to achieve good performance[7]. We used hyper-parameters values as: kernel = 'rbf', C = 1000, gamma = 0.0001 to get the best performance.

Decision Tree Classifier: Decision tree is one of the fastest supervised classifiers with significantly less time complexity, albeit lower accuracy when dealing with large datasets[88]. We have used hyper-parameter, max_depth = 7, to obtain the best performance.

Random Forest Classifier: A random forest is a meta estimator that employs averaging to increase predicted accuracy and control over-fitting by fitting a number of decision tree classifiers on different sub-samples of the dataset[23]. It is a better classifier than Decision Tree but takes more time to compute because of its model complexity.

Gaussian NB: Naïve bayes classifiers are simple classifiers that use probability for classification[30]. It is one of the fastest classifiers for many applications.

Multilayer perceptron Classifier: MLP Classfier used Artificial Neural Network for classification. It uses the concept of multilayer perceptron backpropagation to update weights and learns by training[186]. We used hyper-parameter, max_iter = 1000, for best model performance.

Logistic Regression: Logistic Regression is a classification model rather than a regression

model. A logistic function is used in this classifier to describe the probability, defining the probable outcomes of a single experiment[182]. The model has higher time complexity than other ML classifiers. We have used hyper-parameter, max_iter = 2000, for best model performance.

2.5 Experiments and results

We segregated our experiments into three categories to bring our research hypothesis into action. First, we took images to determine the baseline accuracy of classification using only spatial image features fed to the traditional ML classifiers. Second, we extracted time and frequency domain features from raw EEG data and fed them to several ML classifiers to measure classification accuracy. Furthermore, we created time-frequency map images of the EEG signals and used image-based classification to see whether the accuracy could be improved.

Experiment 1: Image classification using classical feature extraction and ML classification

Figure 2.4 shows the architecture diagram of Experiment 1. Here, we extract spatial features from the images with two different filters, i.e. binarization of images and application of histogram of ordered gradient (HOG) filer to the images. Once the image is processed, it is fed to the different classifiers to observe the classification performance. The classifiers used in this experiment are Decision Tree, Random Forest, K-Nearest Neighbour, Support Vector Machine, Multilayer Perceptron, Gaussian Naive Bayes, and Logistic Regression.

Table 2.2 illustrates the performance comparison of the best data processing, encoding,

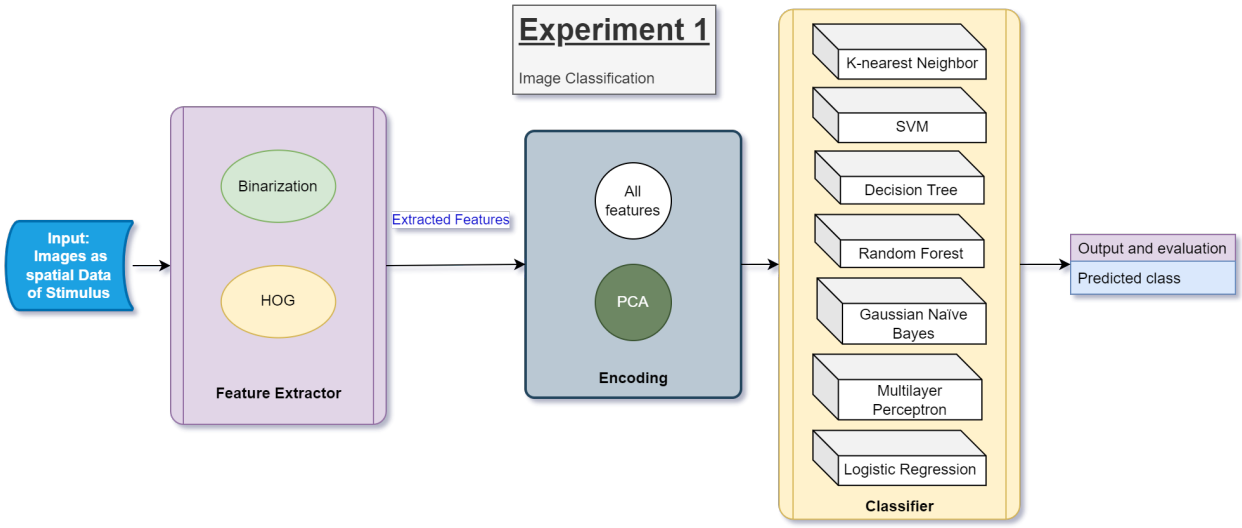


Figure 2.4: Architecture diagram of Experiment 1.

Table 2.2: Performance comparison of image stimuli data with different classifiers.

Feature Extractor	Feature Encoder	Best-Accuracy	Best Classifier Setup
Binarization	No Encoding	0.46	SVM: Kernel as 'RBF"
Binarization	PCA	0.41	SVM: Kernel as 'RBF"
HOG	**No Encoding**	**0.67**	**Gaussian Naïve Bayes**
HOG	PCA	0.65	Logistic Regression

and classifier implementation on the image stimuli data. The image stimulus data was divided into 80% training and 20% testing using a stratified 5-fold cross-validation split to determine the accuracy (please refer to appendix A.2 for details). We observed that the accuracy was higher when the feature space had no encoding than the reduced feature space using PCA. Interestingly, the image's HOG features perform better in classification compared to binarized features. The highest accuracy achieved was 67% when the HOG features are fed to the Gaussian Naive Bayes classifier model without any encoding.

Experiment 2: EEG classification using classical feature extraction and ML classifiers

In this experiment, we used the EEG signal data from Marini et al.[106] dataset to analyze the visual classification performance of various conventional ML classifiers when human-brain evoked temporal data is fed as a visual feature. We also investigated if the machine vision algorithms can predict the claims of neuroscience experiments derived from human perception in Marini et al.[107]. The claims are as follows:

- Claim 1: The event-related potentials (ERP) showed more positive values when occipital and central cluster electrodes were chosen.
- Claim 2: A stronger and more sustained neural signature is invoked of motor preparation contralateral to the dominant hand of subjects.
- Claim 3: The real object showed better ERP than its 2D image when subjects were shown a visual stimulus.

Figure 2.5 shows the architecture diagram of Experiment 2. We encoded the raw EEG signals with many preprocessing steps. As the data provided by Marini et al.[106] was a raw EEG signal, we used different feature extraction methods to find meaningful features. The objective of this experiment was to classify the two classes, i.e., kitchen and garage, by EEG signal features. We used EEGLIB to create a feature set for the whole length of the signal with Power Spectral Density (PSD), Petrosian Fractal Dimension (PFD), Detrended Fluctuation Analysis (DFA), Higuchi Fractal Dimension (HFD), Band power average (including alpha, theta, gamma, and beta) and Synchronisation Likelihood (see section 2.4). We also created a feature set by trimming only the 1600 ms stimulus-response with the bandpower

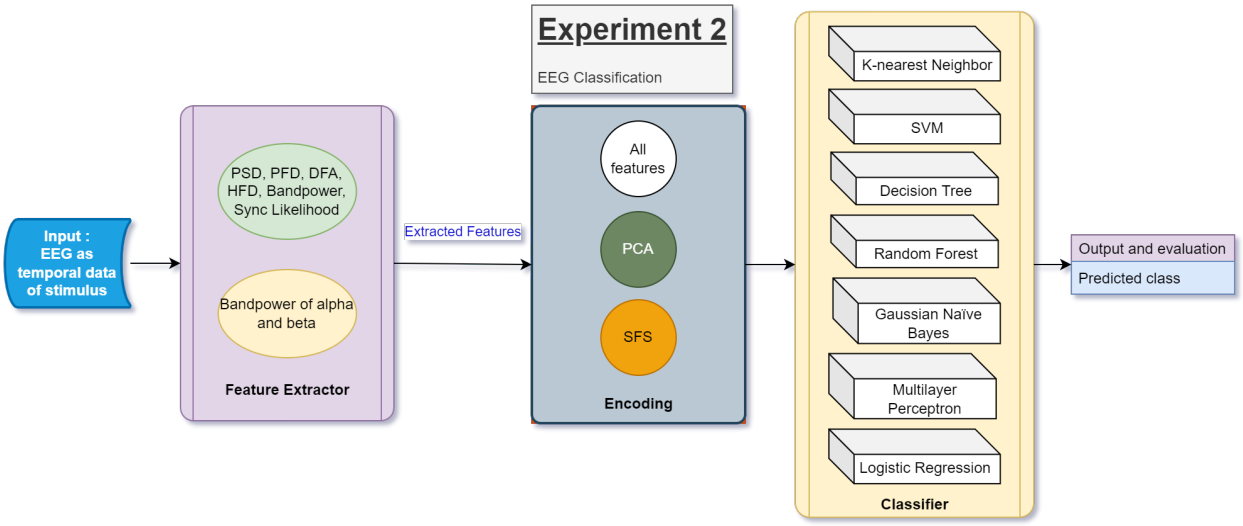
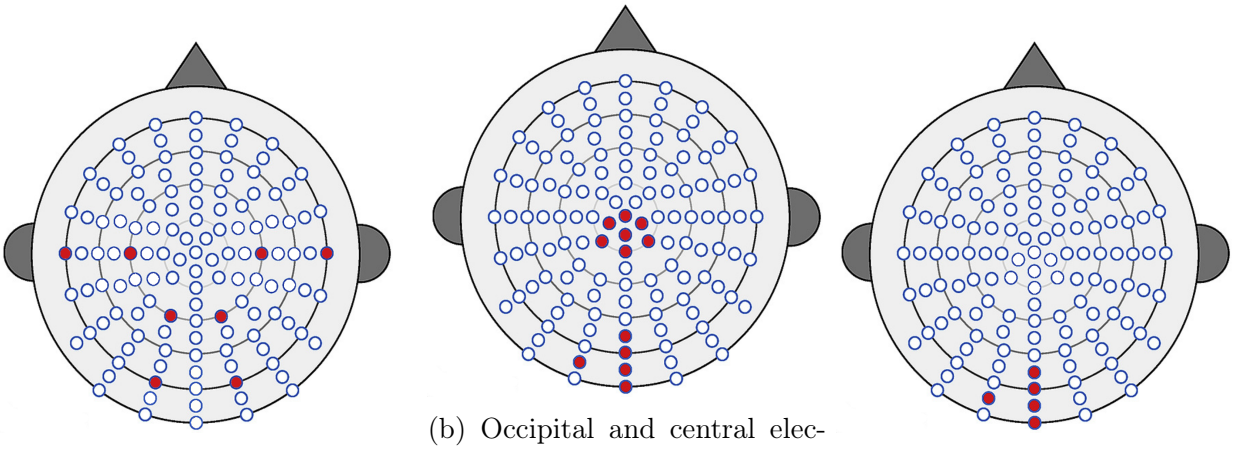


Figure 2.5: Architecture diagram of Experiment 2.

average of the signal using multitaper and periodogram spectrum. Once the feature sets were ready, they were fed to different classifiers to compare the performance of classification. The classifiers used in this experiment were Decision Tree, Random Forest, K-nearest Neighbour, Support Vector Machine, Multilayer Perceptron, Gaussian Naive Bayes, and Logistic Regression. The EEG feature vectors were divided into a subject-wise split of 80% training and 20% testing set uisng a 5-fold stratified group cross-validation (please refer to Appendix A.3 for details). By subject-wise spilt, we mean visual stimuli data was stratified for each subject for both the training and testing set.

Experiment 2a: EEG classification based on electrode selection

Based on the claim 1 findings of the Marini et al.[107] research, the event-related potentials showed more positive values when occipital (A22, A23, A24, A25, A14) and central (A1, A2, A3, B1, B2, D15, D16) cluster electrodes were chosen and they also showed more significant



(a) 8 electrode system

(b) Occipital and central electrode clusters

(c) Occipital electrode clusters

Figure 2.6: Different electrode system chosen for Experiment 2a and 3.

differences for visual classification. Hence, in Experiment 2a, we investigated the performance of our machine classifiers based on different electrode groups. At first, we took eight electrodes (A5, A15, A28, A32, B22, B26, D19, and D23) from the occipital and parietal lobes, which we consider eminent for visual stimulation. The position mapping of electrodes is displayed in figure 2.6a. Secondly, we considered the occipital and central electrode clusters as taken by Marini et al.[107], shown in figure 2.6b. In the end, we ran the same set of classification experiments considering all 128 electrodes.

The performance comparison of all our experiments using different groups of electrode systems is displayed in table 2.3 and 2.4. We observed that, for an 8-electrode system, the accuracy of the classifier does not seem to improve when we extract all the time and frequency domain features. Furthermore, using sequential feature selection, we found that the band power average features revealed to have the highest contributing factor to classification. Consequently, there was an improvement in accuracy when we took only band power average as features using the periodogram spectrum, and the SFS encoder that returned alpha and

Table 2.3: Performance comparison of 8 electrode EEG data with different classifiers.

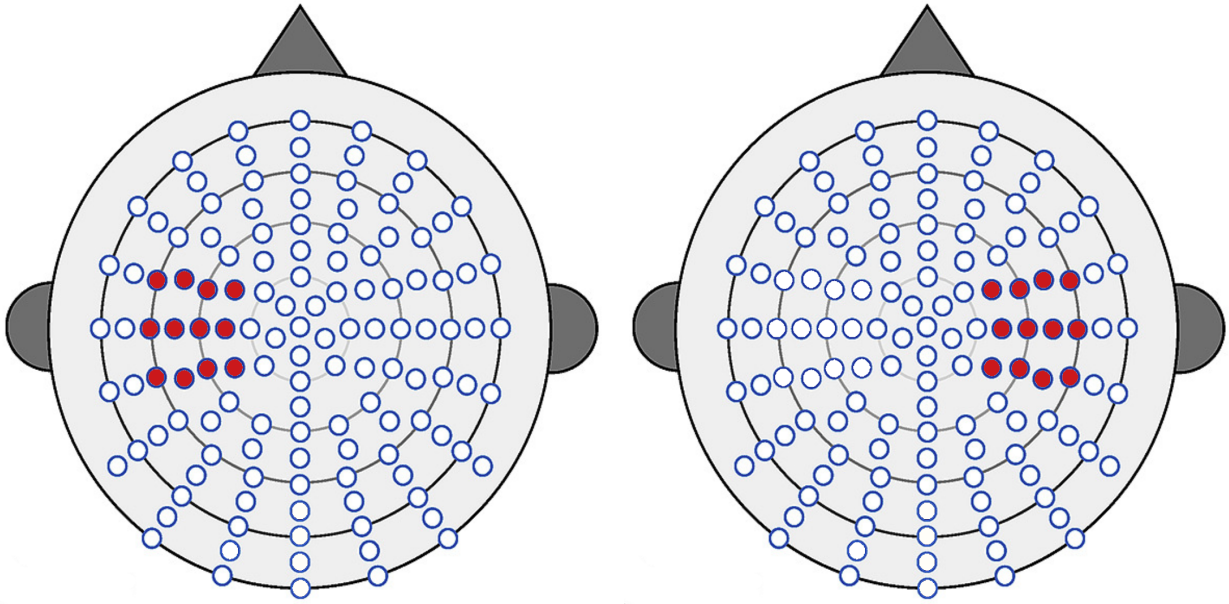
Feature Extractor	Feature Encoder	Best-Accuracy	Best Classifier Setup
Power Spectral Density (PSD) Petrosian Fractal Dimension (PFD) Detrended Fluctuation Analysis (DFA) Higuchi Fractal Dimension (HFD) Band power average (including alpha, theta, gamma, beta) Synchronization Likelihood	No encoding	0.51	Random Forest
	SFS encoding	0.5	Multilayer Perceptron
	PCA encoding	0.5	Multilayer Perceptron
Band power average (including alpha, theta, gamma, beta) using Multitaper Spectrum	No encoding	0.5	Multilayer Perceptron
	SFS	**0.52**	**K-nearest Neighbor**
	PCA	0.51	Multilayer Perceptron
Band power average (including alpha, theta, gamma, beta) using Periodogram Spectrum	No encoding	0.5	Multilayer Perceptron
	SFS	**0.53**	**K-nearest Neighbor**
	PCA	0.52	Multilayer Perceptron

Table 2.4: Performance comparison of occipital, central and all electrode EEG data with different classifiers.

Number of electrodes taken	Feature Extractor	Feature Encoder	Best-Accuracy	Best Classifier Setup
Occipital and central electrode clusters	Band power average of alpha and beta band using Periodogram Spectrum	No encoding	0.51	Multilayer Perceptron
		SFS	0.5	Multilayer Perceptron
		PCA	0.51	Random Forest
All 128 electrodes	Band power average of alpha and beta band using Periodogram Spectrum	**No encoding**	**0.53**	**Logistic Regression**
		SFS	0.51	K-nearest Neighbor
		PCA	0.51	Multilayer Perceptron

beta frequency bands as the most potent features. Following this result, we decided to use only the band power average of alpha and beta band frequencies using the periodogram spectrum for all further experiments.

Later, we compared the accuracy of our ML classifiers with a set of occipital-central electrode clusters and another set considering all electrodes, as shown in table 2.4. Through the analysis of results, we discovered that machine algorithms perform better when data from all electrodes was used with no encoding.



(a) Electrode clusters of left hemisphere (b) Electrode clusters of right hemisphere

Figure 2.7: Hemispherical electrode system chosen for Experiment 2b.

Experiment 2b: EEG classification based on hemispherical brain region

This experiment was carried out to measure the performance of machine visual classification based on claim 2 of Marini et al.[107] neuroscience study. According to the authors[107], a higher ERP in the motor-cortex hemispherical area was contralateral to the dominant hand. All the subjects in their study were right handed and hence, they reported a stronger ERP difference in the left hemispherical regions of the brain than in the right.

To replicate this experiment in machine-learned visual classification, we selected a group of 12 electrodes around left (C3) and right (C4) motor cortex electrodes as hemispherical regions, as shown in figure 2.7. Table 2.5 lists the results of all classifiers used in this experiment. The classification only showed any improvement in the left motor cortex region compared to the right for the dataset[106], making our conclusion unclear.

Table 2.5: Performance comparison of EEG data based on the hemispherical regions of the brain.

Feature Extractor	Classifier	Acc (Left-hem-cluster)	Acc (Right-hem-cluster)
Band power average of alpha and beta band using Periodogram Spectrum	K-nearest Neighbor	0.49	0.52
	SVM: Kernel as 'RBF"	0.5	0.48
	Decision Tree	0.51	0.51
	Random Forest	0.49	0.51
	Gaussian Naïve Bayes	0.53	0.5
	Multilayer Perceptron	0.5	0.49
	Logistic Regression	0.48	0.53

Experiment 2c: EEG classification based on real object versus image stimuli

The dataset used for this study had two different types of EEG recording trials for each visual stimulus; one when subjects observed the physical object of the real world and the other when they viewed the planar 2D images of the same object (see section 2.3, table 2.1). The goal of this experiment was to estimate claim 3 from the study Marini et al. [107], which stated that visual classification could improve when visual stimuli for humans are real objects rather than images of the same stimuli. The visual stimulus feature set was divided into 96 real-object EEG recording trials and 96 2D image EEG recording trials instead of a single feature set of 192 trials. We performed classification using all combinations of electrode system pipelines and ML classifiers as shown in table2.6.

At first, it appears that there is no significant difference in classification performance of the traditional machine learning approach when comparing real-object stimuli with image stimuli. However, when we compared the results using color coding (as shown in the figure 2.6), we discovered that classifiers performed marginally better with the feature set of real-object stimuli.

Table 2.6: Performance comparison of EEG data based on real-object and planar image stimuli.

Number of electrodes taken	Classifier	Acc (Image stimuli)	Acc (Real Stimuli)
All 128 electrodes	K-nearest Neighbor	0.49	0.52
	SVM: Kernel as 'RBF"	0.5	0.52
	Decision Tree	0.49	0.5
	Random Forest	0.51	0.54
	Gaussian Naïve Bayes	0.5	0.5
	Multilayer Perceptron	0.51	0.51
	Logistic Regression	0.51	0.52
Occipital and central electrode clusters	K-nearest Neighbor	0.49	0.53
	SVM: Kernel as 'RBF"	0.49	0.5
	Decision Tree	0.5	0.5
	Random Forest	0.52	0.53
	Gaussian Naïve Bayes	0.49	0.5
	Multilayer Perceptron	0.49	0.49
	Logistic Regression	0.49	0.52
Left hemisphere electrode clusters	K-nearest Neighbor	0.5	0.51
	SVM: Kernel as 'RBF"	0.5	0.5
	Decision Tree	0.49	0.5
	Random Forest	0.46	0.48
	Gaussian Naïve Bayes	0.52	0.53
	Multilayer Perceptron	0.48	0.5
	Logistic Regression	0.5	0.47
Right hemisphere electrode clusters	K-nearest Neighbor	0.52	0.48
	SVM: Kernel as 'RBF"	0.5	0.5
	Decision Tree	0.51	0.49
	Random Forest	0.53	0.54
	Gaussian Naïve Bayes	0.5	0.53
	Multilayer Perceptron	0.51	0.5
	Logistic Regression	0.53	0.52
Mean accuracy		**0.5**	**0.51**

Experiment 3:EEG Classification using Scaleogram

Figure 2.8 shows the architecture diagram of Experiment 3. In this approach, we generated the time-frequency map images from raw EEG data. Scaleogram images were created using wavelet transformation, as these images could provide distinctive features for classifying time-series signals[167]. Initially, the signal from 0 to 800 ms was extracted from the original signal to capture the subject's response to the presented stimuli. We took the electrode signal

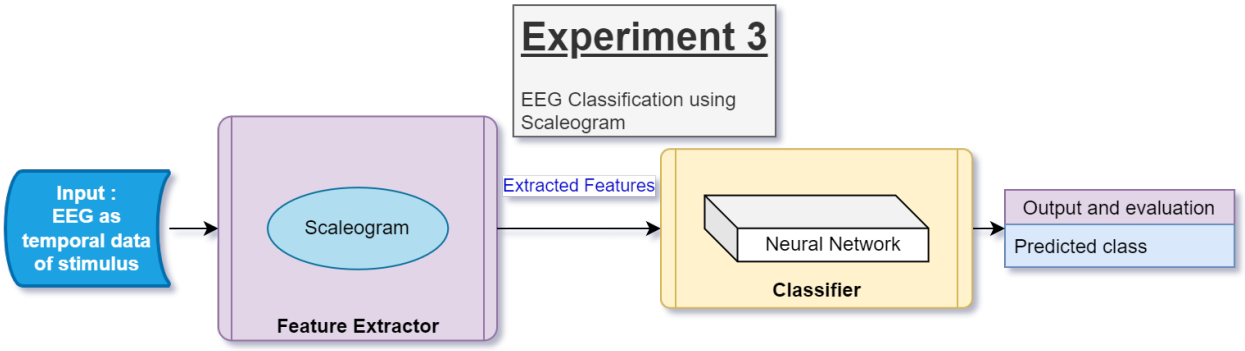


Figure 2.8: Architecture diagram of Experiment 3.

"O1" and the clusters of occipital electrodes (A22, A23, A24, A25, A14: see figure2.6c) for this experiment. The parameters used to form the scaleogram were specific. We took the Morlet wavelet for transformation with a scale of 255 and used the continuous wavelet transformation (CWT) from the Pywavelet libraries.

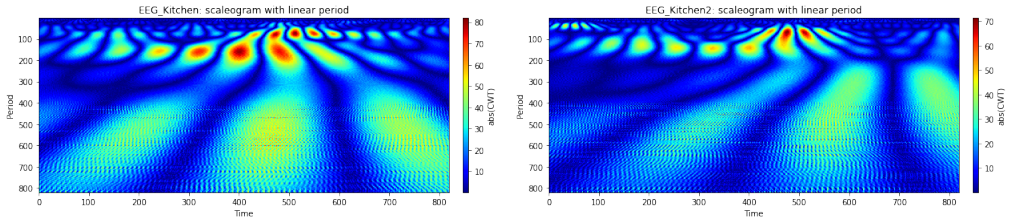


Figure 2.9: An example of scaleogram images from EEG data for the kitchen category.

The scaleogram patterns of class 'Garage' seemed to differ from those of class 'Kitchen' based on the initial observation taken from the recording of the Subject 1, 'O1' channel. Figures 2.9 and 2.10 show a few example scaleograms of the two classes. The EEG scaleograms as images were fed as input to a neural network classifier (see table 2.7 for network design) to obtain a baseline classification using only one channel (O1) as a feature.

The 96 trials for the 'O1' channel EEG image stimuli were insufficient to obtain a reason-

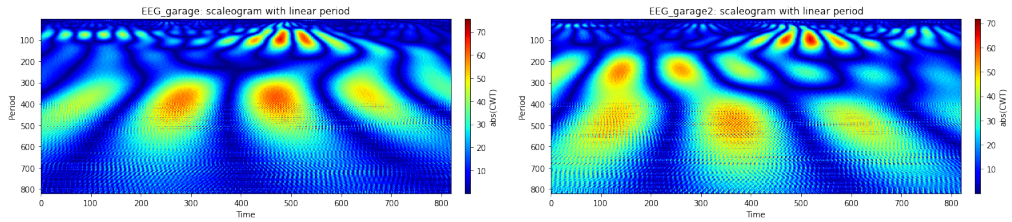


Figure 2.10: An example of scaleogram images from EEG data for the garage category.

Table 2.7: Performance comparison of EEG data based on the Scaleogram image extraction

Electrodes taken	**Feature Extractor**	**Classifier**	**Accuracy**
O1	CWT (scaleogram)	Neutral Network*	0.37
Occipital cluster			0.41
*Neutral Network setup: Input layer, Flattening Layer, Dense Neuron Layer(No. of neurons: 300), Dense Neuron Layer(No. of neurons: 100), Dense Neuron Layer(No. of neurons: 2, Activation: Softmax)			

able accuracy. We then took the averaged value of the EEG signals in the occipital cluster for all subjects as input to the scaleogram encoding and provided the accuracy results in Table 2.7. The accuracy returned from the classifier was not up to the mark for a binary classification (less than 50%).

2.6 Discussion

In this work, we used classic machine learning classifiers to assess the effectiveness of binary classification for two types of visual stimulus feature sets. These feature sets include spatial data of digital images and temporal data of human brain-evoked EEG signals while viewing the real object and its 2D life-sized photographs. We also used a time-frequency domain

encoder called a scaleogram to convert EEG data into a spectral image.

After a batch of experiments, we found the following findings. Binary classification requires a large amount of image data to categorize the two classes only on the basis of spatial features. Enhancing the training set and using edge feature extractors such as HOG may help improve performance to some extent if the dataset is small. Features such as band power average, PSD, and time-frequency map do not aid in improving EEG-based classification if the signal is recorded at less than the minimum required frequency resolution of the desired band. In our case, the total length of the signal was 2.8 s, of which only 1600 ms can be counted for the actual stimulus response. Even though the difference is visible at the minute level of observation, the machine could not train itself with such a low resolution. The conventional ML classifiers used in this study could not adequately simulate the differential findings claimed by neuroimaging analysis Marini et al. [107] - the occipital, central and one hemispherical side (contralateral to the dominant hand) of the motor cortex regions contribute and perceive more to visual classification. However, the visual classification of a visual stimulus as a tangible object versus a 2D image showed some, although marginal, similarity to the original finding [107].

Summary of the key contributions in this chapter:

The following points describe the snapshot of this chapter when we used traditional machine learning approaches for visual classification of Marini et al. [106] dataset:

- The best visual classification performance for Marini et al. [107] dataset is 67%, which was obtained by spatial data (setup: HOG features from images when fed to Gaussian NB classifier).

- EEG alpha and beta band averages are the best features for classifying temporal EEG data.
- EEG data classification is best performed when all 128 electrodes are selected with no encoding (like PCA) with an accuracy of 53%.
- Hemispherical region Region-based classification of EEG data did not show any observable difference with ML classifiers
- Visual stimulus as real objects showed a marginal improvement over 2D planner images.

In the next chapter 3, we introduce deep learning approaches and propose to improve visual feature vectors. Later, we evaluated visual classification performance through various pipeline approaches from the deep learning architecture for a 40-class benchmark EEG dataset called EEG-ImageNet. Moreover, we re-evaluated all the claims by Marini et al. [107] in chapter 4 with classical and deep learning approaches performed in this chapter and in chapter 3. We also propose a cross-modal fusion approach to achieve state-of-the-art performance in automated visual classification.

Chapter 3

Deep learning approaches for visual classification

This chapter is based on the accepted conference paper: The International Conference on Intelligent Data Science Technologies and Applications (IEEE - IDSTA 2022), San Antonio, TX, US.

- Mishra , A., Raj, N., & Bajwa, G. (2022). EEG-based Image Feature Extraction for Visual Classification using Deep Learning (Mishra et al. [112]).

While capable of segregating visual data, humans take time to examine a single piece, let alone thousands or millions of samples. The deep learning models efficiently process sizeable information with the help of modern-day computing. However, their questionable decision-making process has raised considerable concerns. Recent studies have identified a new approach to extract image features from EEG signals and combine them with standard image features. These approaches make deep learning models more interpretable and also enables faster converging of models with fewer samples. Inspired by recent studies, we developed an efficient way of encoding EEG signals as images to facilitate a more subtle understanding of brain signals with deep learning models. Using two variations in such encoding methods, we classified the encoded EEG signals corresponding to 39 image classes with a benchmark accuracy of 70% on the layered dataset of six subjects, which is significantly higher than the existing work.

3.1 Introduction

Nowadays, digital data consists mainly of visual content such as images or videos. Visual classification is advancing our civilization through applications ranging from facial recognition to improved product discoverability. Humans have evolved to be natural and accurate classifiers, but our ability to categorize objects or create new categories is occasionally limited. We classify a scene using intuition and experience, but only if the distinctive patterns are visible[102]. Machine perception can capture critical classifications and detect small patterns that the human mind ignores. Although deep learning models such as CNN provide good performance, they lack clear explanations due to a black-box decision-making approach[168], time consuming and computationally expensive for prediction and classification improvement[41].

Recently, Kaneshiro et al. identified a new approach to visual classification with machine learning using EEG signals from the human brain[76]. It attempts to map human perception to picture data collected from machine classifiers for visual classification tasks.

Previous studies have also used EEG signals as images, encoding them in the space-time domain[184] and the time-frequency domain[166]. In this way, we can leverage the EEG cognitive features to aid in further classification of images by using techniques discussed in our methodology section. Representing these EEG signals in multidimensional encoded image space via a single sample provides rich data for classification. As deep learning models require a large amount of data to learn and extract features efficiently, these encodings enable us to do the same.

In the following sections, we briefly describe the previous work, our initiatives, and the results with comparisons of various methodologies. The main focused approaches are as follows:

1. Visual classification using only images

2. Visual classification using only EEG data

3. Visual classification using two-dimensional grayscale EEG encoded image data

3.2 Contributions

We found that EEG-ImageNet[127] is one of the challenging benchmark datasets with 40 classes, which is a high number with EEG classification. Thus, it can be used to improve and achieve a robust classification of EEG and image data. Our specific contribution was to employ this dataset with an approach to encode EEG data[184] in an 8-bit grayscale image with 128 channels per trial. Using it with CNN + SVM pipeline-based transfer learning, we outperformed state-of-the-art models for EEG-ImageNet dataset classification.

3.3 Related Work

We reviewed previous studies and approaches for EEG classification and visual classification that use an EEG dataset.

EEG data consists of multiple channels of time-series signals per sample or trial. Over the years, many studies and state-of-the-art approaches have contributed to improve EEG data classifications. SyncNet[93] and EEGNet[84], used for benchmarking classes in the EEG datasets, are notable mentions of deep learning models of high performance.

Li et al.[93] built the SyncNet that used structured 1D convolution layers to extract power from both time and frequency domains and classified the data based on joint modeling of 1D CNNs. Lawhern et al.[84] used 2D CNNs along different dimensions of EEG data to create EEGNet. The first set learned frequency information via temporal convolution and then learned spatial features of specific frequency using a depth-wise set of CNNs.

One of the necessities in the standard EEG processing pipeline is feature engineering[92]. Traditional feature extraction provides only certain aspects of EEG, such as frequency or temporal domain content. A time-frequency resolution of the EEG data can achieve a two-dimensional representation of the EEG. Therefore, signals can be converted to a spectrogram image using STFT (short-time Fourier transform)[166,161] or to a scaleogram image using CWT (continuous wavelet transform)[167]. Thus, it can leverage the performance of the pre-trained deep learning models using transfer learning. The earlier efforts by Raghu et al.[134] have shown success in EEG classification using spectrogram encoded images instead of raw EEG signals. Hence, we explored efficient ways to use the image-transformed features from EEG-ImageNet data in one of our classification experiments using CNN-based deep learning models without losing any channel or frequency information.

Zhang et al.[184] followed a unique classification approach based on an EEG dataset[17]. They used 8-bit heatmap scaling to convert the raw EEG signals into images. Later, they used pre-trained MobileNet to extract deep features from these images. In the end, they used an SVM classifier and obtained good classification performance.

Multi-modal fusion of diverse data has been emerging research to automate visual classification problems. Spampinato et al.[149] presented the first automated visual classification method driven by human brain signals using a CNN-based regression on the EEG manifold. Visual image stimuli evoked EEG data were learned with an RNN and then used to classify images into a learned EEG representation. Their promising results paved the way for human brain processes involved in effectively decoding visual recognition for further inclusion in automated methods.

Li et al.[91] claimed that the results reported by Spampinato et al.[149] depended on a block design, and a rapid-event design process cannot replicate the results. The block design and training/test set splits were such that every trial in each test set came from a block with many

Table 3.1: Performance comparison of previous approaches on EEG-ImageNet[127] dataset.

Model performances with correctly filtered EEG-ImageNet		
Classifier models	**Accuracy on ([14-70] Hz)**	**Accuracy on ([5-95] Hz)**
Stacked LSTMs[149,127]	NA	0.22
SyncNet[93]	0.24	0.27
EEGNet[84]	0.34	0.32
EEG-ChannelNet[127]	0.41	0.36
GRUGate Transformer[156]	0.48	0.46

attempts in the corresponding training set. Li et al.[91] also claimed that the wrong block design approach led to a high classification accuracy of long-term brain activity associated with a block rather than the perception of class stimuli.

Palazzo et al.[128] defended their previous research[149] by counter analyzing the claims made by Li et al.[91] while admitting the faults in data pre-processing. As a result, the classification performance was lower than the previously claimed average accuracy of around 83%[149]. According to their latest work[127], the reduced accuracy was attributable to EEG drift because the earlier work mistakenly used unfiltered EEG data. The authors[149] achieved nearly 20% accuracy with correctly filtered data (high-frequency gamma-band); EEGNet[84] reported about 30% accuracy, and EEG-Channel Net[127] obtained approximately 50% accuracy. However, in their experimental finding, the temporal correlation in[149]'s data was nominal, and the block design was suitable for classification studies after pre-processing. Consequently, they corrected for the publicly available data with proper filtering.

Following this revelation, the performance of the models developed by Fares et al.[47], Mukherjee et al.[115], Kavasidis et al.[78] and Zheng et al.[190] cannot be compared as they are based on the unfiltered EEG data from Spampinato et al.[149] and the filtered dataset was published in 2020[127].

Tao et al.[156] used the filtered data provided by Palazzo et al.[127] with different frequency sets of [55-95] Hz, [14-70] Hz, and [5-95] Hz to compare all state-of-the-art models. Their proposed model for EEG classification was based on the GRUGate Transformer. They achieved 61% accuracy on the high gamma band filtered data but reached only 49% with all band frequency data ([5-95] Hz). Table 3.1 compared the performance of all previous studies that used the correctly filtered EEG ImageNet dataset.

3.4 Dataset

For this study, we have used the updated filtered dataset published in 2020[149][127]. It is the first EEG dataset for ImageNet visual classification's multi-class subset. For future convenience, we will refer to this dataset as EEG-ImageNet.

EEG signals were recorded from six subjects viewing a subset of the ImageNet dataset with 40 classes, each containing 50 images. The EEG sequence was collected from 128 electrodes with a sampling rate of 1000 Hz and 500 ms in duration. According to Kaneshiro et al.[76], the first 500 ms of single-trial EEG responses are informative for the categories and characteristics of visual objects in this investigation. They also found that as little as 80 ms of response from a single electrode is enough to classify EEG signals.

The number of trials found in the dataset is 11,964, after removing 36 low-quality samples from 12000 recordings. It is also worth mentioning that we discovered 11 missing trials for one class (mushrooms, labeled 33 in the dataset) for subject 1. We deleted all data with label 33 from both the Image and EEG datasets, *resulting in a 39-class dataset with 11,682 samples.* Table 3.2 shows the details of the parameter of the dataset we discussed.

Many versions of the dataset were constructed with different bandpass filters, ranging from [5-95] Hz to [14-70] Hz for various experiments. For our research, we used both forms

Table 3.2: Parametric values of EEG-ImageNet[127] dataset taken for this study

Datasets parameters	Values
Total number of trials	11,682
Stimulus type	Image
Number of classes	39
Number of subjects	6
Stimuli per subject	1947
Stimuli per class	50*
EEG recording time for each stimulus	500ms
Sampling rate	1000 Hz

*There are approximately 50 images for each class.

of filtered data to utilize various brain signal bands (theta, alpha, beta, and gamma) information captured during visual stimulation. Data were also adjusted using a z-score per channel to provide zero-centered values with a unitary standard deviation[127].

3.5 Methodology

LSTM-based EEG Model

The EEG data is a time series signal, so LSTM models are a reasonable choice to extract features for this application[84,47]. They can successfully learn on data with long-range temporal dependencies considering the time lag between inputs and their corresponding outputs.

We used a mix of common stacked Bi-LSTMs and LSTMs to measure the baseline accuracy of our EEG data for image classification. Previous studies also showed comparable performance using stacked LSTMs[149] and stacked bi-directional LSTMs[47] for EEG-ImageNet data. Figure 3.1 shows the design of our LSTM-based EEG model, and we explain the parameters in our experiments (see Section 3.6)

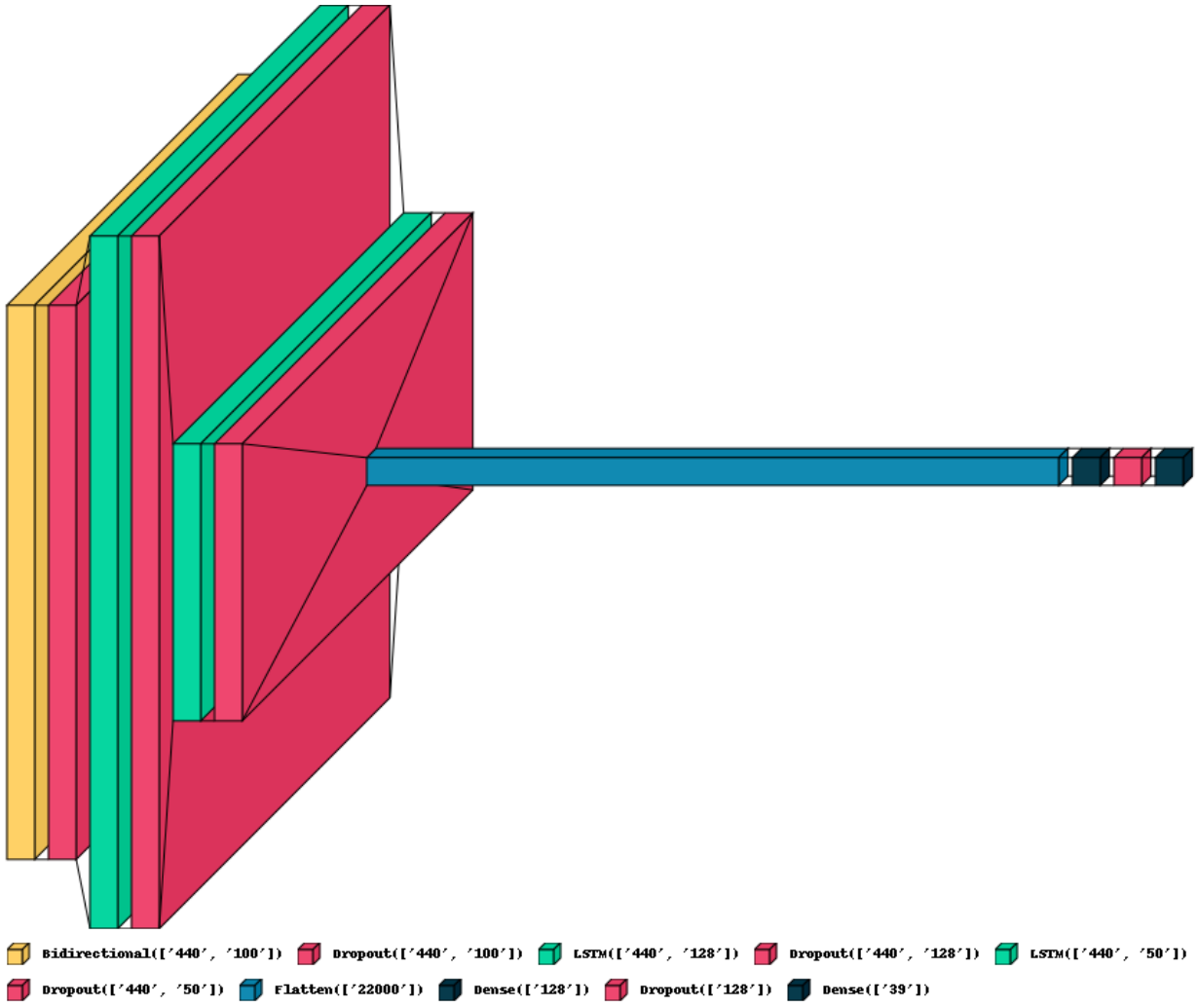


Figure 3.1: Design architecture of LSTM-based EEG Model (LEM)

CNN-based Image Model

Convolutional neural networks are the most effective deep learning models to extract detailed features from images. With the support of ImageNet dataset [37] and deep learning models such as AlexNet, VGG [145], ResidualNet [67], MobileNet [71], and EfficientNet [153], image classification has improved immensely and almost achieved its peak performance. However, the depth and parameters of these models necessitate a considerable resource for training with the ImageNet

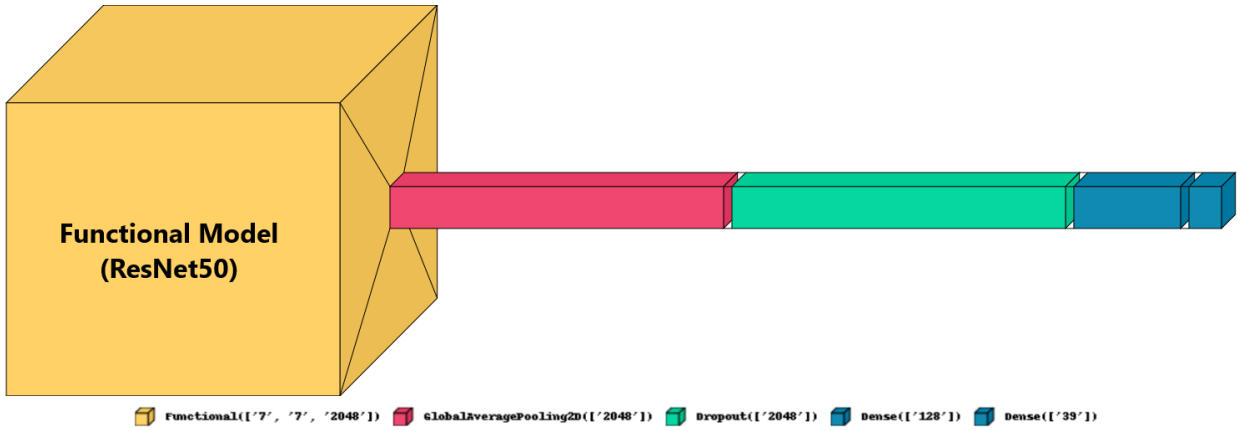


Figure 3.2: Design architecture of CNN-based Image Model (CIM)

dataset. We can utilize these models on the go because they are already pre-trained using ImageNet weights. We took these pre-trained models and added a fully connected layer to fine-tune the model concerning our dataset, as our image data are a subset of ImageNet. We conducted experiments to extract spatial features from images using these models; for example, we classified the image data from the EEG-ImageNet dataset using the CNN-based Image Model with ResNet50 as the functional model (see architecture in figure 3.2). The parameters of these models were explained in our experiments (see section 3.6).

EEG-to-Image-based model

Although deep learning models such as LSTM and CNN can extract features from raw time series data directly, it is crucial to account for noise and volatility in stochastic signals such as EEG. Additionally, before training the model with a sample, a pipeline technique should be determined to treat the raw EEG data as a feature consistent with the model's design.

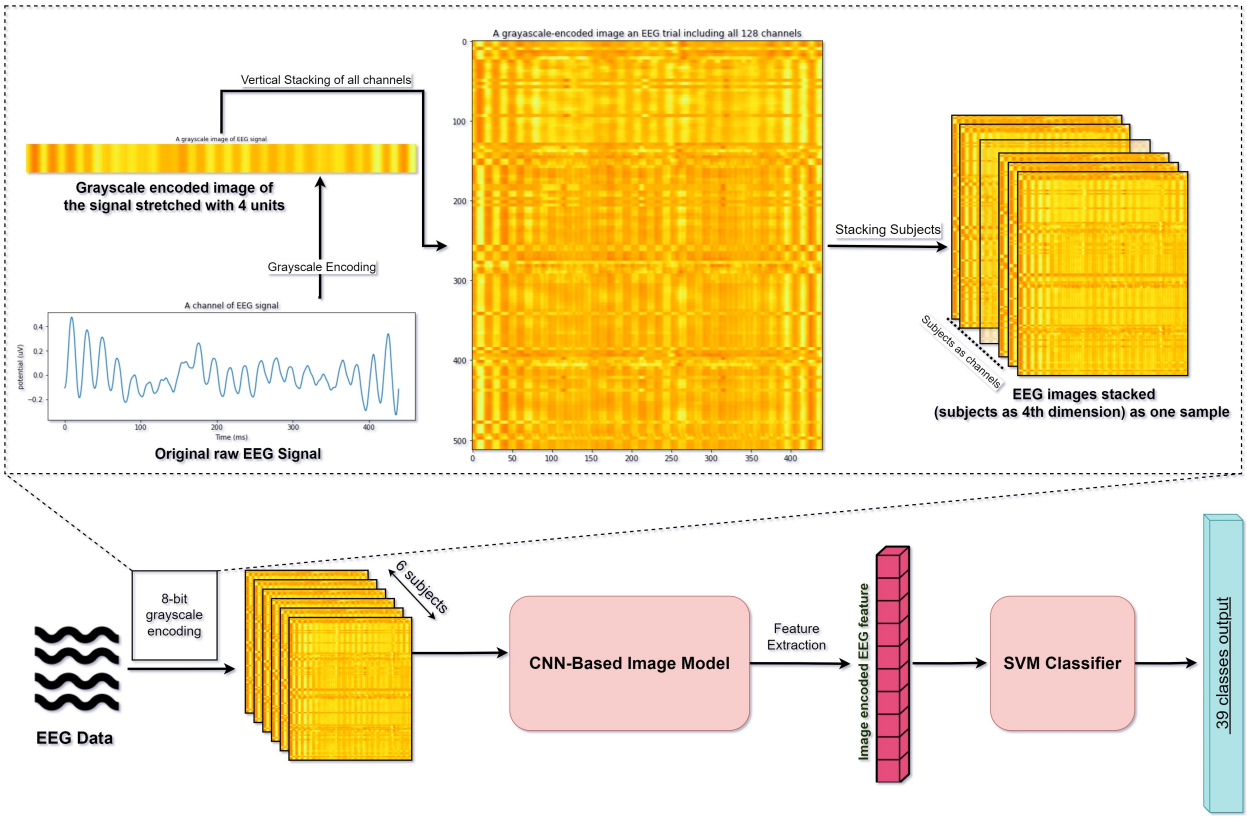


Figure 3.3: The process of encoding EEG trials to images for EEG-to-Image-based models.

Grayscale Image encoding - A unique approach to encode EEG signals data to 2D spatial Grayscale Image sample

We designed a feature extractor method to transform the EEG signals into an 8-bit grayscale heatmap image[184]. We applied this encoding in 2 ways.

Creating a 3 channel grayscale Image-encoded EEG data sample to replicate RGB channels of Image data. In the first method, we created a grayscale heat map of the EEG signals for each subject for each test (40 trials / images per class). The process involved normalizing the signals with a min-max scalar to transform values in the range of

(0,1). The normalized signals were then converted to 8-bit grayscale heatmap images using an encoding scheme described by Zhang et al.[184](see figure 3.3). However, we used a factor of four instead of 32 to increase pixel values to incorporate data from all 128 electrodes of a single trial. After this, we vertically layered each electrode's (4, 440) grayscale image to create an image of size (512, 440) corresponding to all 128 channels. In the end, each EEG trial's grayscale image was cloned three times (512x440x3) and resized to (224x224x3) to match the input shape of the pre-trained models such as MobileNet, Resnet, and EfficientNet.

Creating a six channel grayscale Image-encoded EEG data sample that includes trials of the six subjects viewing the same image stimulus. We designed to group all subjects' EEG signals corresponding to the same image stimulus in the second method. As a result, the dimension of our EEG encoded image changed from (512x440x3) having 11,682 trials to (512x440x6) with 1947 trials. Instead of replicating the grayscale image data three times, we used the trials of six subjects for the same image as channels. The inference is that this will improve the efficiency of data input processing. The coded algorithm for this encoding method is shown in appendix A.1.

Having the flexibility to adjust the size of the signal image, we combined the EEG representations from all the 128 electrodes of a single trial. Thus, we created a unique signature for each sample without exhausting computational resources. This strategy allows integration of structural and textural analysis methods such as pixel variance, morphological gradient calculations, normalization, and enhancement algorithms to improve classification accuracy. It also allows the application of feature extraction methods that characterize the many forms, textures, and structures of each image, such as the Gray Level Co-occurrence Matrix (GLCM)[66], Hu's Moments[72], and Local Binary Patterns[122]. Figure 3.3 illustrates the extraction of features through 8-bit grayscale image encoding with a stretch of 4 for the

128 channels of a trial and our classification model to assist the grayscale image encoder.

3.6 Experiments and Results

The length of an EEG sequence in the filtered EEG-ImageNet dataset was 500 ms. We selected 440 time points (20 - 460 ms) from 500 ms data, since the precise duration of each signal can vary[127]. Therefore, we excluded the beginning and final 20 samples (20 ms) to prevent interference in adjacent signal recordings.

We used only 1947 out of the 1996 image samples in the ImageNet subset after removing the label 33 (mushrooms) associated with missing trials. Hence, the new size of the EEG-ImageNet dataset had 11,682 recordings (1947x6) with 39 class labels. We split the data into 70% train, 15% validation, and 15% test sets. The reason for choosing this [70/15/15] split instead of [80/10/10] (which was standard split opted by previous studies[156,149,127] using this dataset.) was based on observing consistent accuracy in the cross validation scheme across all the runs. We stratified the data by image samples and labels, implying that each data split included trials from all subjects with the same visual stimulus in the same group. This type of stratification eliminated any bias in the split and avoided overfitting during training. The code implementation of the stratified group split is described in Appendix A.2.

Furthermore, we processed our EEG-ImageNet data based on the following experiments using the models explained in Section 3.5.

Deep learning approaches for visual classification of images

The EEG-ImageNet dataset contains visual stimuli as a subset of the ImageNet dataset (1947 images from 39 classes). To measure the benchmark classification performance of this subset of ImageNet, we performed image classification experiments using different CNN-based pre-

trained models such as AlexNet, VGG16, Resnet50, and MobileNet. We did not choose any model with high depth or parameters, as it would increase the complexity of the model. Unlike Palazzo et al.[127], we did not perform any image augmentation because we chose these models as image feature extractors for our future multi-modal implementations.

Parameters: This model architecture consisted of an input layer of shape (224x224x3), followed by a functional model layer that fits CNN-based pre-trained models, a dense layer of 128 neurons and a softmax class layer as classifiers. We used a stochastic gradient descent optimizer to train the data. The code implementation for this model is illustrated in the Appendix A.5.

Results: We found that the ResNet50 model performed better than other CNN-based models with a test accuracy of 84% because it converges faster for the small number of samples per class[190].

Deep learning approaches for visual classification of EEG data

In this section, we describe our classification experiments performed with EEG-ImageNet data.

Raw EEG data with LSTM-based EEG Model

We directly used raw time-series EEG signals from all subjects in this experiment. The shape of each input EEG data sample is (440x128), where 440 is the number of time points and 128 is the number of channels for each trial.

Parameters: The LSTM-based EEG Model was built with an input layer with the same shape as each EEG sample. It was connected to 50 stacked bidirectional LSTMs, followed by two stacks (128 and 50) of common LSTMs, and finally, a dense layer of 128 neurons. We used

the adam optimizer to train the model with a softmax classifier. The code implementation for this model is shown in appendix A.4.

Results: We observed that the performance of our LSTM-based EEG Model followed a similar trend (Table 3.4) as previous state-of-the-art models[93,84,127,156] using only the raw EEG signals. The beta-gamma-filtered data [14-70] Hz showed somewhat better performance than all data from the frequency band [5-95] Hz.

EEG data encoded as 2D vectors with EEG-to-Image-based model

We performed two types of processing in the EEG-ImageNet dataset for **EEG signal-to-image models** with the help of **Grayscale Image encoding** described previously (see Section 3.5).

Parameters: We leveraged a pipeline framework for transfer learning to train our models with EEG-encoded image data. We extracted the deep features from 8-bit grayscale images of each EEG trial using CNN-based image models such as MobileNet, ResNet, and EfficientNet. These deep features were then input into various machine learning classifiers, including SVM (RBF kernel), K Nearest Neighbor, Random Forest, Decision Tree, and Logistic Regression, to assess our classification performance.

Table 3.3: Classification accuracy of different CNN (3 channels) + ML classifier models on grayscale EEG encoded image data for [14-70] Hz data.

CNN Extractor + classifier	**Image size (512x440)**	**Image resized (224x224)**
MobileNet + SVM(rbf)	0.42	0.36
MobileNet + kNN	0.41	0.36
ResNet + SVM(rbf)	0.5	0.43
ResNet + kNN	0.49	0.41
EfficientNet + SVM(rbf)	**0.51**	**0.41**
EfficientNet + kNN	0.5	0.41

Results:

Table 3.4: Performance comparison of classification models with varying cut-off frequencies in bandpass filters on the EEG-ImageNet data.

EEG data Encoding (with data split)	**Classifier models**	**Accuracy on beta-gamma filtered data ([14-70] Hz)**	**Accuracy on All freq. data ([5-95] Hz)**
Raw EEG data*	Stacked LSTMs[149 127]	NA	0.22
Raw EEG data*	SyncNet[93]	0.24	0.27
Raw EEG data*	EEGNet[84]	0.34	0.32
Raw EEG data*	EEG-ChannelNet[127]	0.41	0.36
Raw EEG data*	GRUGate Transformer[156]	0.48	0.46
Raw EEG data**	LSTM based Model (3.5,3.6)	0.28	0.26
Grayscale image encoded EEG data** (3.5,3.6)	**EfficientNet + SVM(rbf)**	**0.51**	**0.64**
Grayscale image encoded EEG data with subjects as channels (six)** (3.5,3.6)	**EfficientNet + SVM(rbf)**	**0.68**	**0.70**

*Previous study models mentioned in this tabel have used (80% train, 10% validation and 10% test) data spit for EEG-ImageNet dataset.
**The models designed by our study have used (70% train, 15% validation and 15% test) group - stratified spilt for EEG-ImageNet dataset.

We tested the accuracy of all top pipeline combinations using [14-70] Hz EEG-ImageNet data encoded as grayscale images, shown in Table 3.3. We noticed information loss from the encoded images when the image size reduced from (512x440) to (224x224). We also found that the EfficientNet feature extractor with SVM classifier outperformed other model combinations. Hence, we choose to run this model setup for all available frequency data (i.e., [5-95] Hz EEG-ImageNet data).

Ultimately, we compared the performance of state-of-the-art EEG classifiers with the classifier we designed in Table 3.4. The comparison shows the classification accuracy for both data types available for the EEG-ImageNet dataset. *Our grayscale EEG encoded image approach trained with EfficientNet + SVM (RBF kernel) classifier achieved approximately 21% higher accuracy than other approaches using the all frequency dataset [5-95] Hz.* It is worth mentioning that our other approaches also performed well with the filtered data.

3.7 Discussion

LSTM[84] and CNN-based 1D models[127] generally work well with time series data, including many EEG datasets[17,170]. However, this typical strategy of using LSTM with EEGs did not yield a high classification accuracy with the EEG-ImageNet dataset. The low performance of the EEG-ImageNet dataset can be attributed to its complexity, as it is one of the most extensive EEG datasets available in terms of containing an unusually high number of classes $(39)^2$, thus a harder EEG classification problem.

2D CNNs can extract deep features from the data very effectively compared to LSTMs and 1D CNNs. Similarly, EEG signals can be encoded into 2D time-frequency image representations (spectrograms/scaleograms) with methods like STFT and CWT to leverage the deep feature extraction capability of pre-trained CNNs ([67 153]). However, more resources are required for computation as we need a separate time-frequency map of each of the 128 channels, increasing sample sizes and adding enormous complexity to the data processing. In addition to the complexity, the STFT and CWT methods lose feature information when resizing the images produced from the encoding. Therefore, previous studies[161 166] chose a selected number of channels for each trial instead of all EEG electrodes.

On the other hand, our method of encoding the EEG signals to grayscaled image vectors outperforms the current state-of-the-art methods significantly (21%), as seen in Table 3.4. The reason is that we accommodate the two-dimensional feature information from all the 128 channels in a single image by stretching the feature space of each channel instead of compressing it.

In a different approach, we consider the six subjects as six separate channels of an image, because CNN can accommodate more than three channels. This strategy reduces the redundancy of the six different readings from the subjects but preserves the essential visual

stimulus information (all the subjects are watching the same image).

We also observe that the EEG data consisting of all frequencies, i.e., from 5 Hz to 95 Hz, performs better than the dataset filtered with beta and gamma bands, i.e., 14 Hz to 70 Hz when encoded with two-dimensional image representations. This higher performance shows that EEG data has definitive classifying information in alpha frequency and can be helpful when we encode the dataset into deeper dimensions.

Based on the performance of all visual classification techniques described in our study, we conclude that the strategically encoding of EEG data into a two-dimensional feature space provides more exploratory information than raw EEG signals. Furthermore, we learn that the EEG data, presented as an input image to deep learning models, the data from low-frequency EEG bands such as alpha are more accessible and contribute significantly to visual classification.

Thus, our attempt to classify the EEG data using all trials of the subjects as channels improved the convergence and efficiency of the model.

Summary of the key contributions in this chapter:

The following points describe the snapshot of this chapter when we used Deep learning approaches for visual classification of EEG-ImageNet [127] dataset:

- The best visual classification performance for the EEG-ImageNet dataset was 85%, which was obtained using spatial data (setup: CNN-based Image Model).
- For EEG (temporal) data, a Grayscale EEG encoded image approach trained with EfficientNet + SVM (RBF kernel) classifier achieved 70% accuracy, which is approximately 21% higher than current state-of-the-art approaches using the all frequency dataset [5-95] Hz.

- Grayscale image encoding of EEG data is efficient as it accommodates the two-dimensional feature information from all the 128 channels in a single image. It also uses the subjects as a 4th-dimensional channel.

- EEG data consisting of all frequencies [5-95] Hz performs better than [14-70] Hz data when encoded with 2D image representation.

In the next chapter 4, we will explore additional techniques to efficiently encode EEG data for classification and enhance the multi-modal fusion approaches of EEG and Image data.

Chapter 4

Multimode fusion approaches for visual classification

This chapter is based on the accepted conference paper: 17th International Symposium on Visual Computing (Springer - ISVC 2022), San Diego, CA, US.

- Mishra , A. & Bajwa, G. (2022). A New Approach to Visual Classification Using Concatenated Deep Learning for Multimode Fusion of EEG and Image Data (Mishra and Bajwa[111])

In this work, we explore various approaches for automated visual classification of multimodal inputs such as EEG and Image data for the same item, focusing on finding an optimal solution. Our new technique examines the fusion of EEG and Image data using a concatenation of deep learning models for classification, where the EEG feature space is encoded with 8-bit-grayscale images. This concatenated-based model achieves a 95% accuracy for the 39 class EEG-ImageNet dataset, setting a new benchmark and surpassing all prior work. Furthermore, we show that it is computationally effective in multimodal classification when human subjects are presented with visual stimuli of objects in three-dimensional real-world space rather than images of the same. This discovery will improve machines' visual perception and bring it closer to the learned human vision.

4.1 Introduction

When we think about classification problems, the first thing that comes to mind is the search for similar patterns. The human mind learns to classify things on the go in a semi-supervised approach. It is fascinating to observe that one half of the human brain searches for similar patterns, while the other labels are based on intuition. On the contrary, machines employ binary logic to discover patterns in a supervised environment[131]. They lack the intuition that the human mind possesses.

The same applies in the case of visual classification, where humans can easily classify whether the object's high-level pattern is visible to the naked human eye[102] and the prediction is almost accurate most of the time. However, the distinctive micro-pattern can sometimes make a classification task difficult for humans, whereas machines outperform in finding that pattern.

As a result, given that humans and machines perceive visual cues in different ways (which we also discussed previously in chapter 1.5), our aim is to find the best approach to combine human cognition with machine perception for better visual classification.

Recent research[34,16] has revealed that our thoughts can be decoded using brain wave signals. Methods such as fMRI, MEG, and EEG were used to capture brain signals. The activity of every neuron in the brain must be carefully monitored to replicate the human-level neural representations that can adequately capture a visual process. The neuroscience community uses EEG recordings for their portability and ease of use.

Marini et al.[107] showed through an event-related potential analysis of EEG data recorded from the human brain that human perception improves when seeing a visual object in real life compared to its image.

Our study represents a new approach based on model concatenation to improve visual

classification tasks with the help of human protective brain response data collected through EEG recording and spatial features extracted from machines for visual stimuli.

This merger can be achieved through multimodal deep learning. The concept of multimodal deep learning was introduced by Ngiam et al.[120], Sohn et al.[148] and can be defined as a technique to relate similar information from multiple input sources, called modalities. These methods allow the deep learning model to learn the common contextual patterns of different input sources jointly and to create a shared representational meaning of the data. Multimodal learning can be categorised in two ways, intra-modal learning, where the modalities have the same feature representation (e.g. images of the same bird with two different angles), and cross-modal learning, where modalities have different feature representations (e.g. image and sound of the same bird). We are interested in cross-modal learning for visual classification.

In the next section, we will discuss relevant studies proposed to improve and automate visual classification using EEG brainwave data and multimodal learning.

4.2 Related work

The visual classification task using EEG data was first performed by Kaneshiro et al.[76] in 2015, who proposed a representational similarity-based linear discriminant analysis framework to classify 12 different object categories and obtained an accuracy of 28.87% in their proposed data set, known as the object category-EEG data set.

Zhang et al.[184] also proposed a unique approach to visual classification using an EEG dataset[17]. They used 8-bit heatmap scaling to convert raw EEG signals into images. Later, pre-trained MobileNet was used to extract deep features from these images. In the end, they used an SVM classifier and obtained a classification performance of 95.33%.

In their research, Marini et al.[107] found that EEG signals demonstrated a transient string ERP for actual items, possibly due to 3-D stereoscopic differences, in addition to a late persistent parietal amplitude modulation consistent with an 'old-new' memory advantage for actual objects over images. They also discovered that the regional motor cortex side has proportionally higher event-related desynchronisation compared to the contralateral dominant hand. Marini et al.[107] provided the EEG dataset used in their experiments.

Ilievski et al.[74], and Guillaumin et al.[62] showed robust performance for visual classification using multimodal learning with text and image as cross-modal input. Similarly, Owens et al.[126] and Arandjelovic et al.[9] performed visual classification using shared visual and auditory space modalities.

Spampinato et al.[149] introduced multimodal visual classification using EEG and Image data. They used the feature regression method in which visual image stimuli evoked EEG data were learnt with the help of stacked LSTMs and then used to classify images into a learnt EEG representation. The classification performance showed outstanding results until later when it was revealed that the EEG data were not correctly filtered, which added bias to the data. This revelation voided the results of this approach and all other derived works that have used unfiltered data.

Palazzo et al.[127] corrected the dataset used by[149] and later published the filtered EEG-ImageNet dataset which we have used in this study. The joint learning approach of the Siamese network for multimodal visual classification achieved an accuracy of 90.5% with a pre-trained ResNet classifier for Images and 1D CNNs architecture for EEG data.

In the following sections, we discuss all the settings, methods, and experiment results of our approach to multimodal visual classification on the datasets provided by Marini et al.[106]. and Palazzo et al.[127].

4.3 Datasets

Visual classification with multimodal image and EEG data learning has been an emerging study since 2017[149]. As a result, there are only a limited number of publicly available databases, so collecting additional data was not a priority of our research. This research focused on two existing multimodal datasets for visual classification tests, as detailed in the following.

Table 4.1: Parameters of the two publicly available datasets.

Datasets	Trials	Stimulus	Classes	Subjects	Stimuli	Stimuli per class	Rec. per stimulus	Sampling rate
EEG-ImageNet[127]	11,682	Image	39	6	1947	50*	440ms	1000 Hz
Marini et al.[106]	4,224	Image	2	22	96	48	800ms/1600ms	512 Hz
		Real	2	22	96	48	800ms/1600ms	512 Hz

*There are approximately 50 images for each class.

EEG-ImageNet

EEG-ImageNet dataset was published by Spampinato et al.[149] and later updated[127] due to filtering issues and signal bias caused by EEG drift, which we have discussed in chapter 3(section 3.3). We used the recently updated dataset, commonly known as EEG-ImageNet. It was created by recording EEG signals from six subjects using a 128-channel actiCAP electrode system. The recordings included each subject viewing 2000 images (50 images per class with 40 classes from a subset of ImageNet dataset[37]. The signals were recorded for 500 ms for each trial at a sampling rate of 1000 Hz. The total number of trials was 12,000 for 40 classes; however, due to low-quality samples and some missing trials in the dataset, we used 11,682 trials for 39 classes, approximately 50 images for each class. All data from class, mushrooms (labelled 33 in the dataset) were excluded, and some classes did not have all 50 images and the corresponding EEG recordings tagged.

The original authors[127] already normalised the data with a z-score and a corrected baseline per channel. Two filtered formats were available in the band pass ranges [5-95]Hz and [14-70]Hz. We followed the processing by Pallazzo et al.[127] and trimmed the first and last 20 ms from each trial to make all signal lengths 440 ms. We used the variant [5-95] Hz of the data set for this study as it performed comparatively better than [14-70] Hz for the classification of EEG data[127,128]. All parameters used for the EEG-ImageNet dataset are shown in Table 4.1.

Visual stimuli EEG dataset: real-world 3D objects and corresponding 2D image stimuli

Marini et al.[106] introduced an EEG dataset with two distinct but similar visual stimuli. It consisted of 24 subjects viewing three-dimensional real-world kitchen and garage objects and their corresponding images while recording EEG signals. Each subject's data had 192 trials, 96 of which were real-world objects, and the other 96 were exact-size photographs of the same items. A 128-electrode set-up was used to record the signal data at a 512 Hz sampling rate. The entire length of each raw signal was 2800 ms (-800 to 2000ms), of which 800 ms (0 to 800 ms) was the actual response of subjects observing the stimulus, and the next 800 ms (800 to 1600 ms) were with their eyes closed before switching to the subsequent trial. Images (found in the scripts/stimuli) are also provided in the dataset.

We discovered that this dataset is a unique seed in multimodal visual classification research, allowing us to evaluate the performance of state-of-the-art machine learning classifiers and multimodal deep learning architectures based on EEG such as EEG-ChannelNet[127] and our proposed model discussed in Section 4.5. As discussed previously, neuroscience research[106] determined that when a real-world object is used as a stimulus, event-related

desynchronisation is more dominant than simulated pictures of the same thing used. In this work, our objective was to examine whether we could use these findings to improve classification accuracy.

The unwanted artifacts of the original data were removed. However, the raw EEG signals were not processed for ERP analysis. We used various processing techniques, including normalising the data using z-score and then baseline correcting the signal in the prestimulus period (-200 to 0 ms) to give zero-centred values with a unitary standard deviation.

$$Z = \frac{x \quad \mu}{\sigma} \tag{4.1}$$

Equation (4.1) represents the z-score of a signal, where 'x' is the original signal, 'μ' and 'σ' are the mean and standard deviation of the signal, respectively.

We used data from 22 participants, since two of them (two and seven) had fragmentary data. For optimisation, we clipped the signal data to 800 ms (0 to 800 ms) for deep learning models and 1600 ms (0 to 1600 ms) for conventional machine classifiers.

4.4 Data Encoding and Processing

It is vital to process and encode the data as input relevant to the model configuration for optimal performance. We used various feature extraction and data encoding techniques to optimise the feature space and process the data.

Classical Feature Extraction for EEG data

The EEG visual stimuli datasets usually have more categorical information in the alpha, beta, and gamma frequency bands of the signal, as observed by previous studies[149,47,156].

We performed a periodogram spectral analysis to use the relative band power average of all signals as a feature in each trial. This analysis is best suited for low-frequency resolution in small-length signals in the datasets we used[3]. These feature sets are later fed to the machine learning classifiers discussed in Section 4.5 with different mixed PCA pipelines and feature selection encoding.

Histogram of Ordered Gradient(HOG) for Image Feature Extraction

HOG, or Histogram of Oriented Gradients, is a feature descriptor applied as a feature extractor for various computer vision applications - notably, object recognition and classification. The HOG descriptor reinforces a structure for an image since it computes the features using both the gradient's magnitude and angle. Histograms are created for the areas of the image based on the magnitude and orientation of the gradient[36]. We used the HOG filter on the image data to produce a one-dimensional feature vector that can be fed into classifiers to assess the baseline accuracy.

Principal Component Analysis (PCA) Encoding

PCA is a statistical encoder for converting high- to low-dimensional data by picking the main components that capture the most relevant data about the dataset. The features are chosen on the basis of the variance they produce in the output. It is vital to note that the primary components do not have a relationship with each other.

We used PCA encoders to compress the feature dimension of both images and EEG data to evaluate the differences in classification results using only the main components with 99 percent variance.

Feature Selection Encoding

We followed the Sequential Feature Selection (SFS) encoder to identify the best features from the relative power average of the alpha, beta and gamma band feature sets. We noticed that most of the essential features selected were from the averages of the alpha and beta bands, with only a few potent features from the gamma band. Ironically, this finding differs from previous findings by Spampinato et al.[149], Fares et al.[47] who had used unfiltered EEE-ImageNet data for classification, implying that the results of these studies cannot be considered. We used alpha- and beta-band power averages as features in our baseline classification investigations.

Grayscale-image Encoding for EEG data

We used the grayscale-image encoder from chapter 3 that was designed as a feature extractor to convert the values of an EEG signal to an 8-bit grayscale heatmap image. This encoding method was first introduced by Zhang et al.[184] for EEG classification. This strategy allows integration of structural and textural analysis methods, such as pixel variance, morphological gradient calculations, normalisation, and enhancement algorithms, to improve classification accuracy.

As shown in table 4.4 grayscale encoding showed significant performance in the classification of both Marini et al.[106] and EEG-ImageNet[127] data. Consequently, we used this encoder for further experiments in this study.

We then stretched the pixels to a factor of 4 and incorporated the data from all 128 electrodes in a single test. After this, we vertically layered each electrode's (4, 440) grayscale image to create an image of size (512, 440) corresponding to all 128 channels. The data were processed using two strategies with the grayscale image encoder. In the first method, the

grayscale image of each EEG trial was cloned three times (512x440x3) to match the input shape of the CNN-based models.

The shape of the input data with our first method for EEG-ImageNet was (11682x512x440x3) and the Marini et al. dataset was (4224x512x440x3).

In the second method, we stacked all the subjects' EEG signals corresponding to the same stimulus trial as represented in figure 3.3 in Chapter 3. Thus, unlike the first method of replicating the same image three times, the images for all subjects will be stacked as channels. This processing approach turned out to be more efficient than the first one, as it uses different subjects as separate dimensions.

The shape of the input data with our second method for EEG-ImageNet was (1947x512x440x6), and the Marini et al. dataset was (192x512x440x22), given six subjects and 22 subjects in the respective datasets.

4.5 Methods and Model Implementation

Conventional Machine Learning Classifiers

We employed Decision Tree, Random Forest, K-nearest Neighbor, Support Vector Machine (SVM), Multilayer Perceptron, and Logistic Regression as traditional machine learning classifiers in our evaluation. These model setups are default configurations from the sklearn library[129] and were used in the chapter 2. For our experiments, we modified the SVM kernel to RBF. The accuracy of the baseline classification using the one-dimensional feature vectors collected in Section 4.4 was determined using these classifiers.

LSTM-based EEG Model (LEM)

The LSTM-based EEG Model, as used in chapter 3 has an input layer with the same shape as each sample of raw EEG data. It was first linked to 50 stacked bidirectional LSTMs[69], then to two stacks of common LSTMs (128 and 50), and lastly to a dense layer of 128 neurons. To train the model with a softmax classifier, we employed the Adam optimiser. Each input EEG data sample has the shape (ts, ch), where "ts" represents the number of time points, and "ch" represents the number of channels for each trial.

CNN-based Image Model (CIM)

This model also follows the CIM architecture mentioned in chapter 3, including an input layer representing the shape of the image data that would be supplied to the model, a functional model layer that fits CNN-based pre-trained models, a 128-neuron dense layer, and a softmax classification layer. For training, we applied a stochastic gradient descent optimizer.

Pre-trained models such as ResNet[67], VGG16[145], MobileNet[71], and EfficientNet[153] were used as functional models for several experiments mentioned in Section 4.6.

Grayscale-image Encoded EEG Model(GEM)

The architecture of the GEM consists of a pipeline framework where, at first, the raw EEG data signal data is converted to an image feature set using the Greyscale image encoder mentioned in section 4.4. This image feature set is fed to a CIM where the functional model layer is EfficientNet for classification. EfficientNet has previously shown the best performance for grayscale image-encoded EEG data[184]. Figure 4.1 illustrates the design of the GEM model.

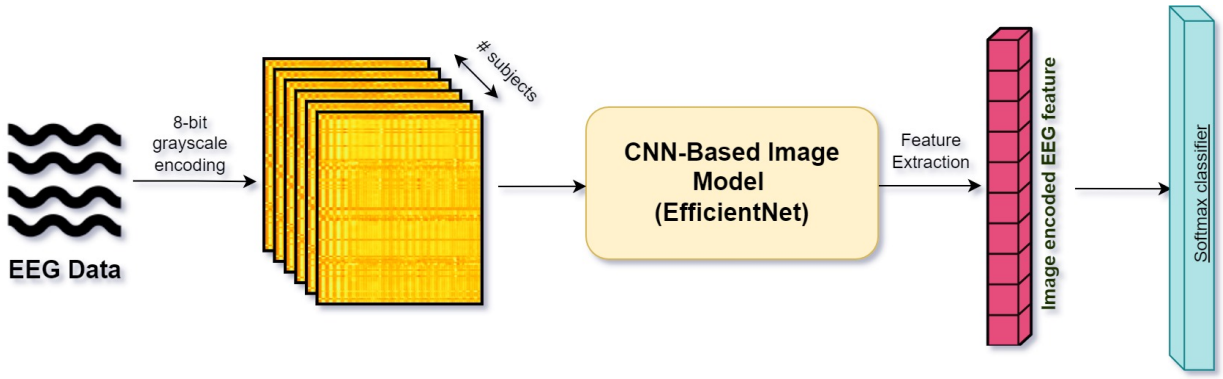


Figure 4.1: Design of the Grayscale-image Encoded EEG Model (GEM)

Regression-based model[149]

The regression-based approach, similar to Spampinato et al. [149,128], consisted of a bi-directional LSTM to extract features from EEG signals and a CNN to extract features of the image for training. As a regressor, we employed a fully connected single-layer model to regress the image features derived from a pre-trained CNN model with EEG features acquired by the BiLSTM model. For simplicity, we set the feature dimension of both the image and EEG features at 128. To match the number of EEG samples with the image samples (1947), we averaged the EEG signals between the six subjects for each stimulus of the image. For the test dataset, the regressor predicted or mapped the image features to the EEG features associated with the test images before using them in the classification model. In other words, the regressed image features were learnt through EEG, and these predicted features were then used to classify the images of the test set. The approach of the regression-based model is shown in figure 4.2.

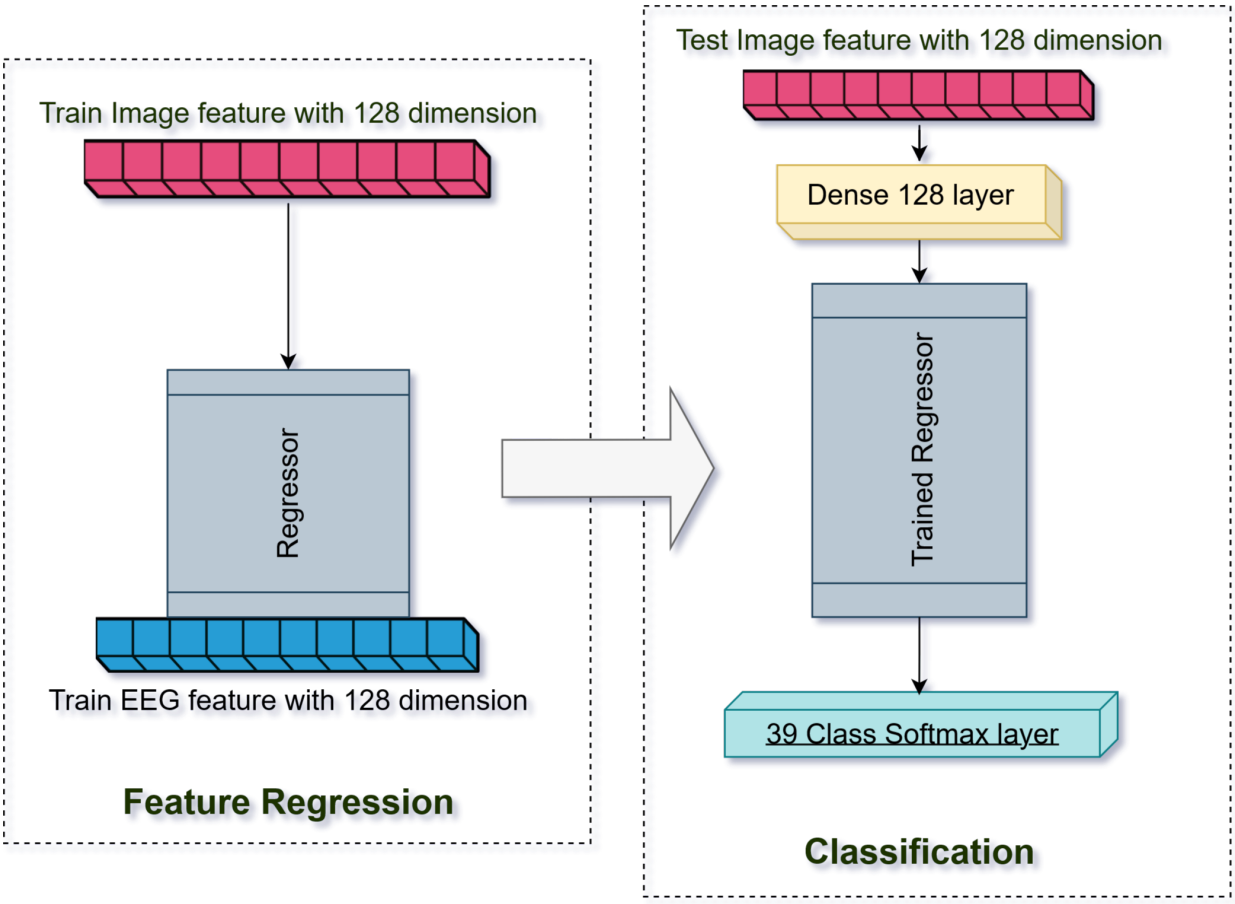


Figure 4.2: The process used to build a Regression-based model.

Vertical Stacking model

In this strategy, we construct a new expanded feature dataset by vertically stacking the features extracted from the EEG and image datasets and then assessing the general classification accuracy using different classifier models (Figure 4.3). The combined features contain the deep features retrieved from the baseline models (LSTM-based EEG and CNN-based Image models). We could append this data due to the dimensionality match in the EEG and image deep features (128). The goal is to augment the number of features from diverse data inputs with comparable properties.

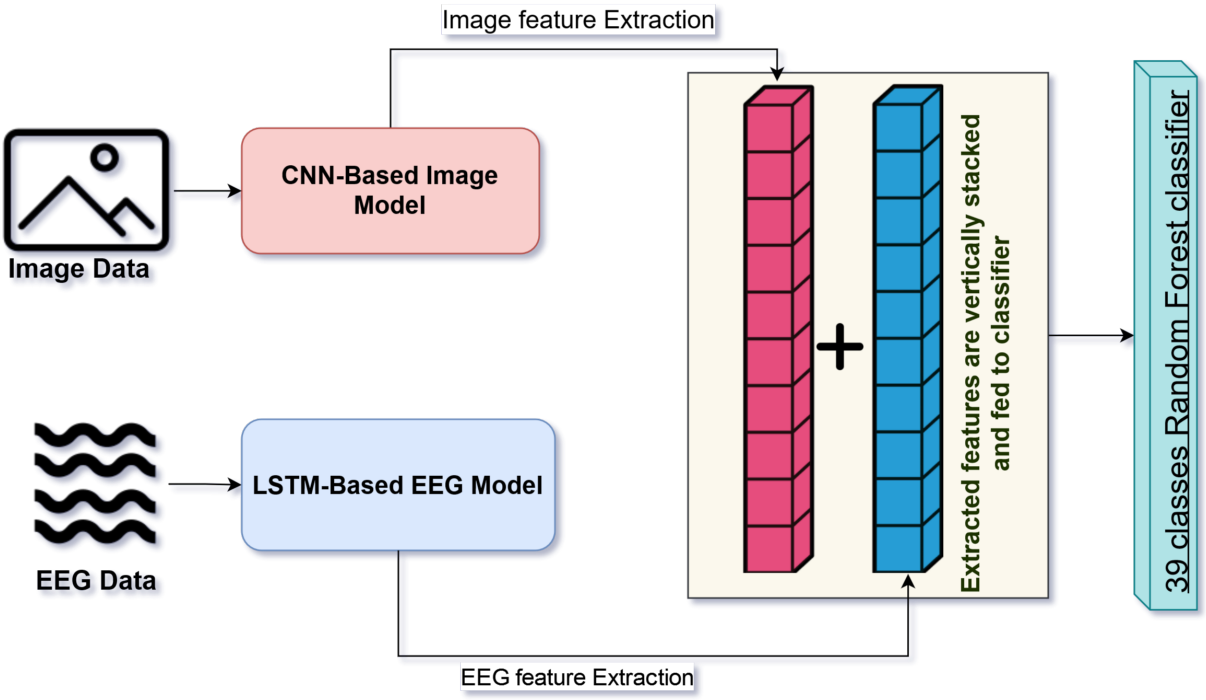
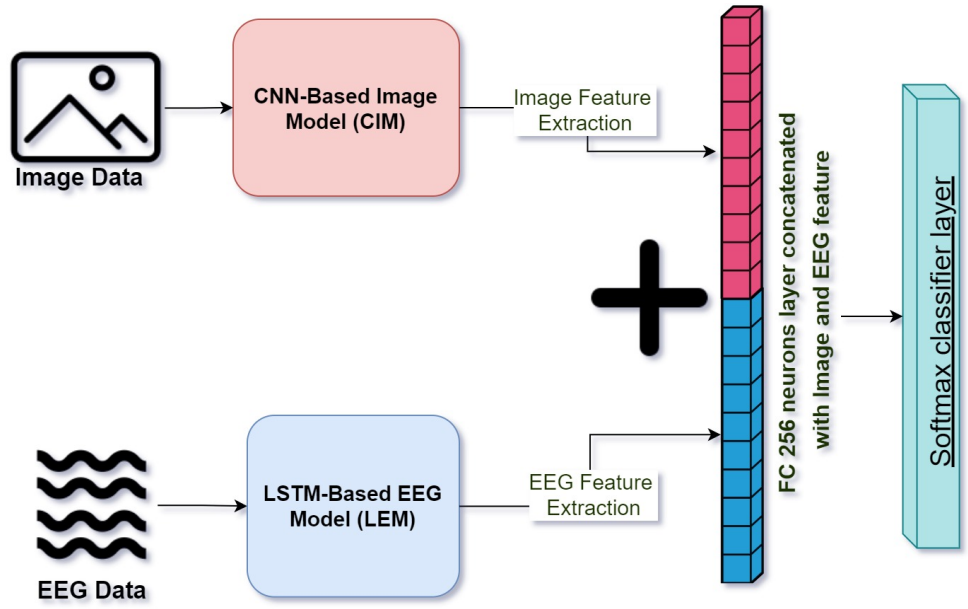


Figure 4.3: The Vertical Stacking model obtained with stacked features from baseline models.

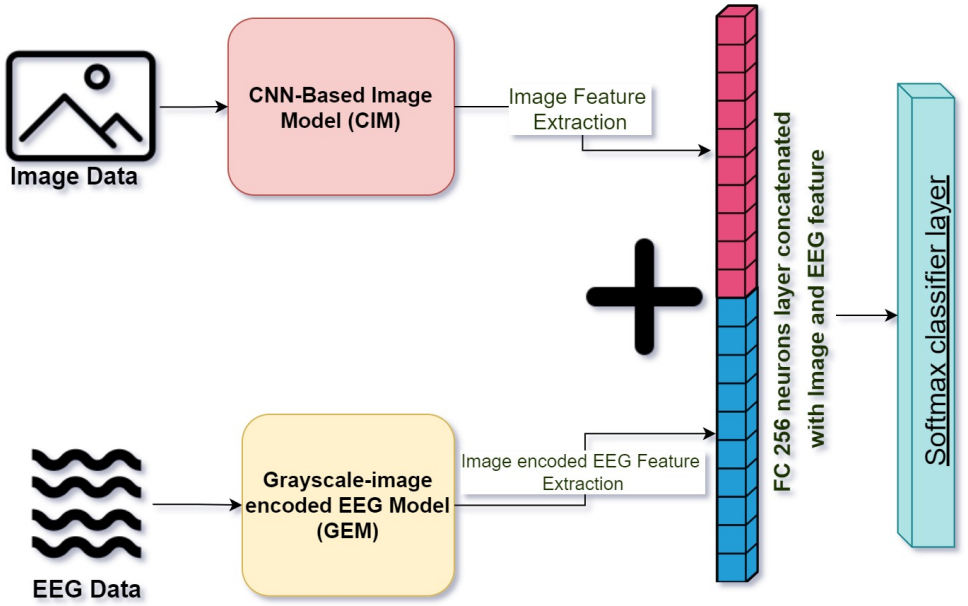
Concatenation-based Models

A concatenation-based technique often combines the data obtained from two or more machine learning models and then labels those features. To predict the different classes in our datasets, we integrate the models' penultimate levels, i.e. fully connected layers, immediately before the classification layer and then form a softmax layer. Concatenated models are popular multimodal deep learning models due to their fast convergence and generalisation, since different modalities do not lose any feature value during joint learning, which aids final classification.

We have concatenated LEM and GEM (which take EEG data as input) with the CIM Model (which takes stimuli images as input) to perform multimodal joint learning visual classification experiments, as shown in Figures 4.4a and 5.1c.



(a) LEM concatenated with CIM



(b) GEM concatenated with CIM

Figure 4.4: Concatenation model design used for multimodal deep learning visual classification

4.6 Experiments and Results

In this section, we investigated the performance of the EEG-ImageNet dataset[127] and Marini et al.[106] using various combinations of encoding mentioned in Section 4.4 and classification models mentioned in section 4.5. Please note that we have used the same stratified 5-fold cross-validation split for Image data and stratified group 5-fold validation split for EEG data as described in chapter 2 for Marini et al.[106] dataset and in chapter 3 for the EEG-ImageNet[127] dataset (please refer to Appendices A.2 and A.3 for details).

Baseline Visual Classification for EEG and Image data

In the first set of experiments, we obtained the baseline performance of the two data, the EEG data, and the corresponding image stimulus data. The goal was to find the extent of distinct visual information present in the classical features of the data.

The classical features of the image stimuli data were extracted using the HOG filter and then fed to the traditional machine learning classifiers mentioned in Section 4.5. We also ran the same test by encoding the HOG-extracted features with PCA for efficiency comparison. Table 4.2a shows the baseline accuracy performance of the best data processing, encoding, and classifier implementation on the Image data from both the EEG-ImageNet and Marini et al. datasets. We observed a slight drop in accuracy when the feature space was reduced using PCA.

To evaluate the baseline accuracy of the EEG data, we used 1600 ms(0 to 1600 ms) of the EEG sequence for the Marini et al. dataset and 440 ms for EEG-ImageNet. The raw EEG signal is processed to extract the power average of the alpha and beta band as a set of features using the periodogram method mentioned in Section 4.4. Table 4.2b shows the best baseline performance for each dataset.

Table 4.2: Baseline performance for EEG and Image data

(a) Baseline accuracy performance for Image stimuli in datasets

Image Dataset	# of classes	Accuracy	Best Classifier Setup
Marini et al.	2	0.67	HOG - Gaussian Naïve Bayes
Marini et al.	2	0.65	HOG+PCA - Logistic Regression
EEG-ImageNet	39	0.05	HOG - SVM
EEG-ImageNet	39	0.04	HOG+PCA - Gaussian Naïve Bayes

(b) Baseline accuracy performance for EEG data in datasets

EEG Dataset	# of classes	Accuracy	Best Classifier
EEG-ImageNet	39	0.15	Multilayer perceptron
Marini et al.	2	0.53	Logistic Regression

Visual Classification using Deep Learning Models

The last experiment tested visual and brain features using traditional machine learning techniques. In this section, we continue to evaluate the depth of classification individually for EEG and Image stimuli features using various state-of-the-art deep learning classifier models. We evaluated the performance of our CIM for Image Stimuli data, LEM and GEM models for EEG data with our chosen datasets.

Table 4.3: CIM performance on Image stimuli data

DL Classifier Model	EEG-ImageNet Acc	Marini et al. Acc
ResNet	**0.85**	**0.81**
VGG 16	0.63	0.72
MobileNet	0.33	0.63
AlexNet	0.2	0.54

To classify images, we have used different CNN-based Image Models mentioned in Section 4.5. Images from the stimuli data sample were first resized (224x224x3) to be correctly fed to CIM models. Table 4.3 shows both datasets' classification results for the Image stimulus data. We discovered that the ResNet model provided the best overall accuracy.

Table 4.4: Performance comparison of EEG data on our deep learning classification model with other SOTA models.

Marini et al. dataset		
EEG data Encoding	**Classifier models**	**Marini et al.[106] Acc**
Raw EEG data	LSTM based Model	0.5
Grayscale image encoded EEG data	EfficientNet + SVM(rbf)	0.52
Grayscale image encoded EEG data with all 22 subjects as channel	**EfficientNet + SVM(rbf)**	**0.73**
EEG-ImageNet (with dataset split comparision))		
EEG data Encoding	**Classifier models**	**EEG-ImageNet -Acc**
Raw EEG data*	Stacked LSTMs[149]	0.22
Raw EEG data*	SyncNet[93]	0.27
Raw EEG data*	EEGNet[84]	0.32
Raw EEG data*	EEG-ChannelNet[127]	0.36
Raw EEG data*	GRUGate Transformer[156]	0.46
Raw EEG data**	LSTM based Model (LEM)	0.26
Grayscale image encoded EEG data**	EfficientNet + SVM(rbf)	0.64
Grayscale image encoded EEG data with all 6 subjects as channel**	**EfficientNet + SVM(rbf)**	**0.70**

*Previous study models mentioned in this tabel have used (80% train, 10% validation and 10% test) data spit for EEG-ImageNet dataset.
**The models designed by our study have used (70% train, 15% validation and 15% test) group - stratified spilt for EEG-ImageNet dataset.

For the LEM classifier, the EEG data were used as is from the dataset. However, for the GEM classifier, we applied image-encoded EEG data, as described in Section 4.4. The classification performance of the EEG signal data in each dataset is represented in Table 4.4. Marini et al. and the EEG ImageNet datasets obtained better classification when the EEG signals were grayscale encoded, while GEM performs best when all subjects of the visual stimulus are stacked as a distinct channel dimension.

Hemispherical Brain Region Classification Comparison

In this experiment, our objective was to estimate the categorisation potency of EEG signal data based on the left and right hemispherical regions of the brain with various traditional and deep learning classifiers used in our experiments 4.2 and 4.6. Marini et al.[107] and Fares et al.[47] claimed that the left hemisphere of the brain processes the classification task better than the right hemisphere and showed robust findings in their experiments.

Table 4.5: Visual classification performance of EEG data based on the hemispherical regions of the brain

Exp.	Implementation approach	Classifier Model Used	Dataset	Acc (Left-hem)	Acc (Right-hem)
1	Alpha and beta band average as features	Decision Tree	Marini et al.	0.51	0.51
2	Alpha and beta band average as features	Gaussian Naïve Bayes	Marini et al.	**0.53**	0.5
3	Grayscale image-encoded EEG data Model (GEM)	EfficientNet	Marini et al.	**0.52**	0.51
4	Alpha and beta band average as features	Random Forest	EEG-ImageNet	0.05	0.05
5	Alpha and beta band average as features	Multilayer Perceptron	EEG-ImageNet	0.06	**0.07**
6	Grayscale image-encoded EEG data Model (GEM)	EfficientNet	EEG-ImageNet	0.13	**0.28**

We selected a cluster of 12 electrodes around left (C3) and right (C4) motor-cortex electrodes as the hemispherical regions. Table 4.5 lists the results of the best traditional and deep learning classifiers for both datasets. The classification was marginally improved in the left motor cortex region than in the right for the Marini et al. dataset. However, it is interesting to note that the right hemispherical region provided better visual classification accuracy for the EEG-ImageNet dataset.

Visual Classification using Multimodal Deep Learning

The previous experiments were carried out to separately evaluate the classification performance of EEG and Image stimuli data. The isolation of these modality inputs on different classifiers gave us an understanding of the accuracy of the benchmark.

In this section, we tested various joint learning experiments. The model architecture accepts multimodal input, EEG, and image feature data and predicts the label of the visual stimulus. As discussed in Section 4.2, multimodal deep learning architectures have provided state-of-the-art performance when mixed data input is fed.

We proposed the concatenation-based mentioned and vertical stacking models in Section 4.5 for multimodal visual classification using deep learning. For our analysis, we evaluated the concatenation-based approach; an LEM model concatenated with the CIM model and

Table 4.6: Performance of the multimodal deep learning classification approach for EEG-ImageNet.

Exp.	Implementation approach	Model Used	Dataset[127]	Accuracy
1	LSTM-based EEG Model (LEM)**	Stacked (BiLSTM + LSTMs) and 128 FC	EEG	0.28
2	**Grayscale image-encoded EEG data Model (GEM)****	**EfficientNet + SVM(rbf)**	**EEG**	**0.70**
3	CNN-based Image Model (CIM)**	ResNet pretrained with FC 128	Image	0.84
4	Regression-based Model[149]*	LEM feature regressed with CIM	Image + EEG	0.03
5	Siamese network[127]*	Joint learning with 1D CNN and ResNet	Image + EEG	0.91
6	Vertical Stacking**	ResNet pretrained and LEM (end to end)	Image + EEG	0.70
7	LEM - based Concatenation Model**	LEM concatenated with CIM	Image + EEG	0.82
8	**GEM - based Concatenation Model****	**GEM concatenated with CIM**	**Image + EEG**	**0.95**

*Previous study models mentioned in this tabel have used (80% train, 10% validation and 10% test) data spit for EEG-ImageNet dataset.
**The models designed by our study have used (70% train, 15% validation and 15% test) group - stratified spilt for EEG-ImageNet dataset.

Table 4.7: Comparison of visual classification based on real object and planner image stimuli (Marini et al. dataset).

Exp.	Implementation approach	Model Used	Dataset Marini et al.	Acc (Image stimuli)	Acc (Real Stimuli)
1	Baseline classification Model	ML classifiers	EEG	0.48	0.5
2	LSTM-based EEG Model (LEM)	Stacked (BiLSTM + LSTMs) and 128 FC	EEG	0.51	0.51
3	Grayscale image-encoded EEG data Model (GEM)	EfficientNet	EEG	0.49	0.52
4	**GEM - based Concatenation Model**	**GEM concatenated with CIM**	**Image + EEG**	**0.72**	**0.78**

another GEM model concatenated with CIM. Moreover, in another experiment, we vertically stacked deep features extracted from LEM and CIM models and added a Random Forest classifier for output.

Table 4.6 illustrates the performance comparison of all state-of-the-art multimodal (Image and EEG data as input) visual classification approaches with our implementation. The results indicate that our GEM-based concatenation model outperformed the other architectures and reached 95% accuracy for the EEG-ImageNet data. The vertical feature stacking method achieved modest performance with an accuracy of 70%.

Classification Performance for Real Object versus Image Stimuli

As discussed in Section 4.1, the Marini et al. dataset had two different kinds of EEG recording trials for each visual stimulus data; one when the subject observed the real-world object and the other when they viewed the planar images of the same object.

The objective of this experiment was to observe if machine-learning classification improves when visual stimuli are real objects instead of images. We merely found any significant difference in classification performance using the traditional machine learning approach when comparing real stimuli with image stimuli. In this analysis, we applied all the best deep learning approaches proposed for visual classification to dive into our investigation.

Table 4.7 shows the performance results of our best-proposed classifiers. The GEM-based concatenation classifier provided a 6% increase in accuracy when visual stimuli were real compared to planar image stimuli.

4.7 Discussion

The datasets used in this work are resourceful but challenging. With 39 classes (a significantly high number in EEG studies), the EEG-ImageNet dataset is one of the benchmark datasets for the overall EEG classification problem. The Marini et al. dataset, however, only has two classes, and it is worth noting that there are only 192 total visual stimuli trials, of which 96 are for real-world object stimuli, and the remaining 96 are for image stimuli. This makes it harder to classify using deep learning models as they require a large number of samples to train perfectly from scratch. We chose these two datasets to evaluate the optimal performance of our proposed visual classification models.

The baseline classification experiments provided the seed results to compare the stretch of

improvement that we achieved while designing more complex classifier architectures. While experimenting with many deep learning architectures for visual classification, we found that the Grayscale image-encoded EEG Model (GEM) was best suited for visual classification of challenging datasets like EEG-ImageNet[127] and Marini et al.[107]. It performed better, as we accommodated the two-dimensional feature information from all 128 channels in a single image by stretching rather than compressing each channel's feature space.

We obtained mixed and inconclusive results for the experiments based on the hemisphere mentioned in table 4.5. The Marini et al. dataset showed almost no improvement in accuracy when the classification task was evaluated in the left hemispherical motor-cortex region compared to the right one. The results for the EEG-ImageNet dataset provided better classification accuracy results in the right hemisphere region compared to the left, which contradicts Fares et al.[47]. As stated by Marini et al.[107] the stronger ERP in the hemispherical region of the motor cortex is contralateral to the dominant hand, and all subjects in the Marini et al. data set are right handed. Unfortunately, we do not have information on the dominant hands of subjects who participated in the EEG-ImageNet dataset.

We also compared different multimodal deep learning approaches to our datasets. Unlike other modalities tagged with images such as text and audio, it is harder for machines to classify patterns from EEG signals as they are more volatile and louder[127]. The concatenation-based approach with grayscale encoding of the EEG data allowed us to accommodate all the data (such as electrodes and the entire set of features for all modalities), unlike other approaches where we had to select the best electrodes[166,161,134], to reduce complexity or select partial information for each modality[149,127,78]. The GEM-based concatenation model also helped discover that machine perception can be enhanced if we use real-world objects as stimuli instead of images.

Summary of the key contributions in this chapter:

The following points describe the snapshot of this chapter when we used multimode fusion learning approaches for visual classification of the EEG-ImageNet [127] and Marini et al. [106] dataset:

- The best visual classification performance for the EEG-ImageNet dataset was 95%, which was obtained by multimodal fusion of temporal (EEG) and spatial (image) data (setup: GEM concatenated with CIM).
- Grayscale image-encoded EEG Model (GEM) was best suited for visual classification of challenging EEG datasets such as the EEG-ImageNet [127] and Marini et al. [106].
- The neuroscience claim of a better visual classification based on hemispherical regions of EEG data contralateral to the dominant hand is inconclusive, as no information was provided on the dominant hands of subjects who participated in the EEG-ImageNet dataset.
- The GEM-based concatenation model also helped discover that machine vision can be enhanced by using real-world objects as stimuli instead of images.

Chapter 5

Conclusion and Future scope

To conclude this thesis, we initially introduced machine perception and how the fundamental construct of machine perception differs from that of humans. We then navigated through the factors where machine and human perceptual systems work in different ways and how to computationally integrate human brain-evoked temporal information into machine-readable data via brain-computer interfaces using EEG. Furthermore, we narrowed our research to visual perception, where we compared machine vision with human visual perception and learnt that humans and machines extract essential visual context and have their own strengths and weaknesses[57,33,101,49,54,169,103]. Consequently, we developed the seed of our study by leveraging the generalisation and sensitivity of human visual perception (using EEG recordings) and also keeping the high computational efficiency and robustness of machine algorithms to reduce the space and time complexity. Therefore, we chose the visual classification task as the case study in this thesis. We designed an automated visual system that learns from a joint/shared representation for both human brain-evoked temporal data and machine learning algorithms-evoked spatial data.

Various approaches to visual classification were applied, referring to several feature

modalities of human and machine perception. The first strategy demonstrated in chapter 2 was based on a typical classification paradigm in which visual representational characteristics were retrieved from spatial (images) and temporal (EEG) data using classical methods and given to conventional classifiers for classification. Performance was trivial due to the rudimentary feature vectors and the versatile nature of the data. This analysis is critical for our study, as it determined the efficacy of the features through the progression of machine learning and perception in each isolated modality. As a result, a baseline performance was established for the data used in this work.

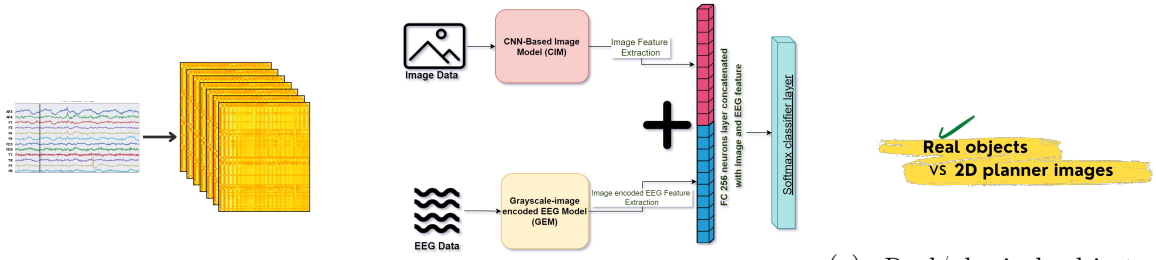
This study progressed to deep learning methodologies to assess the visual classification of both types of visual data as examined in Chapter 3. A pipeline architecture based on transfer learning was used to extract deep features from the raw data as input (both the picture and the EEG separately) and then fed to multiple machine classifiers. A state-of-the-art method was also shown to extract deep features from EEG data as Grayscale Image encoded data. Consequently, the highest performing model had a 70% accuracy in training after grayscale encoding. Deep learning approaches demonstrated a considerable increase over baseline performance in general.

The last chapter of this thesis (Chapter 4) explored the joint representation of visual features of images and EEG data by multimodal fusion of deep learning models implemented in Chapter 3.

The key takeaways from this work are as follows:

(a) A new state-of-the-art approach called Grayscale image encoding of EEG data is efficient as it accommodates the two-dimensional feature information from all the 128 channels in a single image. It also allows for the flexibility of using subjects as a fourth-dimensional channel, increasing the efficiency of the dataset.

(b) We used the above encoding to design "Grayscale image-encoded EEG Model (GEM)"



(a) Grayscale image encoding of EEG data

(b) GEM-based concatenated model.

(c) Real/physical objects are better visual stimuli than images.

Figure 5.1: The key take ways of this thesis work (shown in the order of points laid out.)

for visual classification. This model provided a new benchmark accuracy performance of 70% in EEG- based visual classification with a challenging 39-class dataset such as EEG-ImageNet and later obtained an accuracy of 95% using concatenation-based multimodal deep learning classification tasks when features of both modalities (EEG and image data) were used as cross-modal input.

(c) We also discovered that automated visual classification could be improved for multimodal inputs of EEG and image data when the visual stimulus shown to subjects while recording EEG is a real-world object instead of an image.

The motivation for this study was to encourage greater collaboration between artificial intelligence and neuroscience. It is crucial to improve the efficiency and precision with which machine perception processes contextual information while learning more about visual representational data patterns and matching human interpretation standards. This research will be expanded in the future to seek better alternative multimodal feature fusion algorithms to improve automatic visual classification and to produce a new multimodal EEG-image dataset with real-world items from the picture given as visual stimulus.

These and other machine perception breakthroughs are critical for unlocking the promise of a dynamic data-rich environment via multi-sensor, multi-level, data-to-decision techniques.

Traditional applications such as surveillance, object categorisation, target tracking, pattern discovery, machine learning, and data mining will benefit from new degrees of trustworthy autonomy. Furthermore, they will enable new developments in cyber-physical systems that will improve our quality of life in remote health care, emergency response, traffic flow management, power generation and delivery, condition monitoring and diagnostics of machinery, geospatial analysis, social networks, and other areas.

Bibliography

[1] Aha, D. W. and Bankert, R. L. 1995. A comparative evaluation of sequential feature selection algorithms, *Pre-proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics*, PMLR, pp. 1–7.

[2] Ahmed, H., Wilbur, R. B., Bharadwaj, H. M. and Siskind, J. M. 2021. Object classification from randomized eeg trials, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3845–3854.

[3] Akin, M. and Kiymik, M. K. 2000. Application of periodogram and ar spectral analysis to eeg signals, *Journal of Medical Systems* **24**(4): 247–256.

[4] Alariki, A. A., Ibrahimi, A. W., Wardak, M. and Wall, J. 2018. A review study of brain activity-based biometric authentication, *Journal of Computer Science* **14**(2): 173–181. **URL:** *https://thescipub.com/abstract/jcssp.2018.173.181*

[5] Altaheri, H., Muhammad, G., Alsulaiman, M., Amin, S. U., Altuwaijri, G. A., Abdul, W., Bencherif, M. A. and Faisal, M. 2021. Deep learning techniques for classification of electroencephalogram (eeg) motor imagery (mi) signals: a review, *Neural Computing and Applications* pp. 1–42.

[6] Angeloni, C., Salter, D., Corbit, V., Lorence, T., Yu, Y.-C. and Gabel, L. A. 2012. P300-based brain-computer interface memory game to improve motivation and performance, *2012 38th Annual Northeast Bioengineering Conference (NEBEC)*, pp. 35–36.

[7] Anguita, D., Ghio, A., Oneto, L., Parra Perez, X. and Reyes Ortiz, J. L. 2013. A public domain dataset for human activity recognition using smartphones, *Proceedings of the 21th international European symposium on artificial neural networks, computational intelligence and machine learning*, pp. 437–442.

[8] Anh, V. H., Van, M. N., Ha, B. B. and Quyet, T. H. 2012. A real-time model based support vector machine for emotion recognition through eeg, *2012 International conference on control, automation and information sciences (ICCAIS)*, IEEE, pp. 191–196.

[9] Arandjelovic, R. and Zisserman, A. 2017. Look, listen and learn, *Proceedings of the IEEE International Conference on Computer Vision*, pp. 609–617.

[10] Atyabi, A., Shic, F. and Naples, A. 2016. Mixture of autoregressive modeling orders and its implication on single trial eeg classification, *Expert systems with applications* **65**: 164–180.

[11] Aydın, S., Saraoğlu, H. M. and Kara, S. 2009. Log energy entropy-based eeg classification with multilayer neural networks in seizure, *Annals of biomedical engineering* **37**(12): 2626–2630.

[12] Babadi, B. and Brown, E. N. 2014. A review of multitaper spectral analysis, *IEEE Transactions on Biomedical Engineering* **61**(5): 1555–1564.

[13] Baillet, S., Mosher, J. and Leahy, R. 2001. Electromagnetic brain mapping, *IEEE Signal Processing Magazine* **18**(6): 14–30.

[14] Bajwa, G. and Dantu, R. 2016. Neurokey: Towards a new paradigm of cancelable biometrics-based key generation using electroencephalograms, *Comput. Secur.* **62**: 95–113.

[15] Bascil, M. S., Tesneli, A. Y. and Temurtas, F. 2016. Spectral feature extraction of eeg signals and pattern recognition during mental tasks of 2-d cursor movements for bci using svm and ann, *Australasian physical & engineering sciences in medicine* **39**(3): 665–676.

[16] Bastos, A. M., Usrey, W. M., Adams, R. A., Mangun, G. R., Fries, P. and Friston, K. J. 2012. Canonical microcircuits for predictive coding, *Neuron* **76**(4): 695–711.

[17] Begleiter, H. 1999. Eeg database.
URL: *https://kdd.ics.uci.edu/databases/eeg/eeg.data.html*

[18] Blasch, E., Herrero, J. G., Snidaro, L., Llinas, J., Seetharaman, G. and Palaniappan, K. 2013. Overview of contextual tracking approaches in information fusion, *Geospatial InfoFusion III*, Vol. 8747, SPIE, pp. 77–87.

[19] Bong, S. Z., Wan, K., Murugappan, M., Ibrahim, N. M., Rajamanickam, Y. and Mohamad, K. 2017. Implementation of wavelet packet transform and non linear analysis for emotion classification in stroke patient using brain signals, *Biomed. Signal Process. Control* **36**: 102–112.

[20] Borisoff, J., Mason, S., Bashashati, A. and Birch, G. 2004. Brain-computer interface design for asynchronous control applications: improvements to the lf-asd asynchronous brain switch, *IEEE Transactions on Biomedical Engineering* **51**(6): 985–992.

[21] Bousseta, R., El Ouakouak, I., Gharbi, M. and Regragui, F. 2018. Eeg based brain computer interface for controlling a robot arm movement through thought, *Irbm* **39**(2): 129–135.

[22] Bozinovski, S., Sestakov, M. and Bozinovska, L. 1988. Using eeg alpha rhythm to control a mobile robot, *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 1515–1516 vol.3.

[23] Breiman, L. 2001. Random forests, *Machine learning* **45**(1): 5–32.

[24] Cabañero-Gomez, L., Hervas, R., Gonzalez, I. and Rodriguez-Benitez, L. 2021. eeglib: A python module for eeg feature extraction, *SoftwareX* **15**: 100745.

[25] Cao, L., Li, J., Ji, H. and Jiang, C. 2014. A hybrid brain computer interface system based on the neurophysiological protocol and brain-actuated switch for wheelchair control, *Journal of Neuroscience Methods* **229**: 33–43.
URL: *https://www.sciencedirect.com/science/article/pii/S0165027014001058*

[26] Cao, L., Xia, B., Maysam, O., Li, J., Xie, H. and Birbaumer, N. 2017. A synchronous motor imagery based neural physiological paradigm for brain computer interface speller, *Frontiers in human neuroscience* **11**: 274.

[27] Carlson, T. and Millan, J. d. R. 2013. Brain-controlled wheelchairs: a robotic architecture, *IEEE Robotics & Automation Magazine* **20**(1): 65–73.

[28] Chai, R., Naik, G. R., Nguyen, T. N., Ling, S. H., Tran, Y., Craig, A. and Nguyen, H. T. 2016. Driver fatigue classification with independent component by entropy rate bound minimization analysis in an eeg-based system, *IEEE journal of biomedical and health informatics* **21**(3): 715–724.

[29] Chakladar, D. D. and Chakraborty, S. 2018. Multi-target way of cursor movement in brain computer interface using unsupervised learning, *Biologically Inspired Cognitive Architectures* **25**: 88–100.

[30] Chan, T. F., Golub, G. H. and LeVeque, R. J. 1982. Updating formulae and a pairwise algorithm for computing sample variances, *COMPSTAT 1982 5th Symposium held at Toulouse 1982*, Springer, pp. 30–41.

[31] Chaudhary, S., Taran, S., Bajaj, V. and Sengur, A. 2019. Convolutional neural network based approach towards motor imagery tasks eeg signals classification, *IEEE Sensors Journal* **19**(12): 4494–4500.

[32] Chiappa, S. and Bengio, S. 2003. Hmm and iohmm modeling of eeg rhythms for asynchronous bci systems, *Technical report*, IDIAP.

[33] Chollet, F. 2019. On the measure of intelligence, *arXiv preprint arXiv:1911.01547* .

[34] Clark, A. 2013. Whatever next? predictive brains, situated agents, and the future of cognitive science, *Behavioral and brain sciences* **36**(3): 181–204.

[35] Dai, M., Zheng, D., Na, R., Wang, S. and Zhang, S. 2019. EEG classification of motor imagery using a novel deep learning framework, *Sensors (Basel)* **19**(3).

[36] Dalal, N. and Triggs, B. 2005. Histograms of oriented gradients for human detection, *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, Vol. 1, Ieee, pp. 886–893.

[37] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database, *2009 IEEE conference on computer vision and pattern recognition*, Ieee, pp. 248–255.

[38] Dhar, S., Ordonez, V. and Berg, T. L. 2011. High level describable attributes for predicting aesthetics and interestingness, *CVPR 2011*, IEEE, pp. 1657–1664.

[39] Dhiman, R., Priyanka, N. A. and Saini, J. S. 2018. Motor imagery classification from human EEG signatures, *Int. J. Biomed. Eng. Technol.* **26**(1): 101.

[40] Djamal, E. C., Abdullah, M. Y. and Renaldi, F. 2017. Brain computer interface game controlling using fast fourier transform and learning vector quantization, *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)* **9**(2-5): 71–74.

[41] Doan, T.-N., Do, T.-N. and Poulet, F. 2013. Large scale visual classification with many classes, *International Workshop on Machine Learning and Data Mining in Pattern Recognition*, Springer, pp. 629–643.

[42] Duan, F., Lin, D., Li, W. and Zhang, Z. 2015. Design of a multimodal eeg-based hybrid bci system with visual servo module, *IEEE Transactions on Autonomous Mental Development* **7**(4): 332–341.

[43] Duan, J., Li, Z., Yang, C. and Xu, P. 2014. Shared control of a brain-actuated intelligent wheelchair, *Proceeding of the 11th World Congress on Intelligent Control and Automation*, pp. 341–346.

[44] Dulay, J., Poltoratski, S., Hartmann, T. S., Anthony, S. E. and Scheirer, W. J. 2022. Guiding machine perception with psychophysics, *arXiv preprint arXiv:2207.02241* .

[45] Escalera, S., Baró, X., Gonzalez, J., Bautista, M. A., Madadi, M., Reyes, M., Ponce-López, V., Escalante, H. J., Shotton, J. and Guyon, I. 2014. Chalearn looking at people challenge 2014: Dataset and results, *European conference on computer vision*, Springer, pp. 459–473.

[46] Fares, A., Zhong, S.-h. and Jiang, J. 2019. Eeg-based image classification via a region-level stacked bi-directional deep learning framework, *BMC Medical Informatics and Decision Making* **19**(6): 1–11.

[47] Fares, A., Zhong, S. and Jiang, J. 2018. Region level bi-directional deep learning framework for eeg-based image classification, *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, IEEE, pp. 368–373.

[48] Farwell, L. A. and Donchin, E. 1988. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials, *Electroencephalography and clinical Neurophysiology* **70**(6): 510–523.

[49] Firestone, C. 2020. Performance vs. competence in human–machine comparisons, *Proceedings of the National Academy of Sciences* **117**(43): 26562–26571.

[50] Fleer, S., Moringen, A., Klatzky, R. L. and Ritter, H. 2020. Learning efficient haptic shape exploration with a rigid tactile sensor array, *PloS one* **15**(1): e0226880.

[51] Fossum, E. R. and Hondongwa, D. B. 2014. A review of the pinned photodiode for ccd and cmos image sensors, *IEEE Journal of the electron devices society* .

[52] Freud, E., Macdonald, S. N., Chen, J., Quinlan, D. J., Goodale, M. A. and Culham, J. C. 2018. Getting a grip on reality: Grasping movements directed to real objects and images rely on dissociable neural representations, *Cortex* **98**: 34–48.

[53] Frigui, H., Zhang, L. and Gader, P. D. 2010. Context-dependent multisensor fusion and its application to land mine detection, *IEEE Transactions on Geoscience and Remote Sensing* **48**(6): 2528–2543.

[54] Funke, C. M., Borowski, J., Stosio, K., Brendel, W., Wallis, T. S. and Bethge, M. 2021. Five points to check when comparing visual perception in humans and machines, *Journal of Vision* **21**(3): 16–16.

[55] Ganea, P. A., Allen, M. L., Butler, L., Carey, S. and DeLoache, J. S. 2009. Toddlers' referential understanding of pictures, *Journal of experimental child psychology* **104**(3): 283–295.

[56] Ganin, I. P., Shishkin, S. L. and Kaplan, A. Y. 2013. A p300-based brain-computer interface with stimuli on moving objects: Four-session single-trial and triple-trial tests with a game-like task design, *PLOS ONE* **8**(10): null.
URL: *https://doi.org/10.1371/journal.pone.0077755*

[57] Geirhos, R., Meding, K. and Wichmann, F. A. 2020. Beyond accuracy: quantifying trial-by-trial behaviour of cnns and humans by measuring error consistency, *Advances in Neural Information Processing Systems* **33**: 13890–13902.

[58] Gernsheim, H. 1986. *A concise history of photography*, Courier Corporation. Paper no. 10.

[59] Göksu, H. 2018. BCI oriented EEG analysis using log energy entropy of wavelet packets, *Biomed. Signal Process. Control* **44**: 101–109.

[60] Grieggs, S., Shen, B., Rauch, G., Li, P., Ma, J., Chiang, D., Price, B. and Scheirer, W. 2021. Measuring human perception to improve handwritten document transcription, *IEEE Transactions on Pattern Analysis and Machine Intelligence* .

[61] Guede-Fernandez, F., Fernandez-Chimeno, M., Ramos-Castro, J. and Garcia-Gonzalez,

M. A. 2019. Driver drowsiness detection based on respiratory signal analysis, *IEEE access* **7**: 81826–81838.

[62] Guillaumin, M., Verbeek, J. and Schmid, C. 2010. Multimodal semi-supervised learning for image classification, *2010 IEEE Computer society conference on computer vision and pattern recognition*, IEEE, pp. 902–909.

[63] Guo, S., Lin, S. and Huang, Z. 2015. Feature extraction of p300s in eeg signal with discrete wavelet transform and fisher criterion, *2015 8th International Conference on Biomedical Engineering and Informatics (BMEI)*, pp. 200–204.

[64] Gursel Ozmen, N., Gumusel, L. and Yang, Y. 2018. A biologically inspired approach to frequency domain feature extraction for eeg classification, *Computational and Mathematical Methods in Medicine* **2018**.

[65] Ha, K.-W. and Jeong, J.-W. 2019. Motor imagery EEG classification using capsule networks, *Sensors (Basel)* **19**(13): 2854.

[66] Haralick, R. M., Shanmugam, K. and Dinstein, I. 1973. Textural features for image classification, *IEEE Transactions on Systems, Man, and Cybernetics* **SMC-3**(6): 610–621.

[67] He, K., Zhang, X., Ren, S. and Sun, J. 2015. Deep residual learning for image recognition, *CoRR* **abs/1512.03385**.
URL: *http://arxiv.org/abs/1512.03385*

[68] Hochreiter, S. and Schmidhuber, J. 1997a. Long short-term memory, *Neural computation* **9**(8): 1735–1780.

[69] Hochreiter, S. and Schmidhuber, J. 1997b. Long short-term memory, *Neural computation* **9**(8): 1735–1780.

[70] Hortal, E., Planelles, D., Costa, A., Iánez, E., Úbeda, A., Azorín, J. M. and Fernández, E. 2015. Svm-based brain–machine interface for controlling a robot arm through four mental tasks, *Neurocomputing* **151**: 116–121.

[71] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M. and Adam, H. 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications, *CoRR* **abs/1704.04861**.
URL: *http://arxiv.org/abs/1704.04861*

[72] Hu, M.-K. 1962. Visual pattern recognition by moment invariants, *IRE transactions on information theory* **8**(2): 179–187.

[73] Ieracitano, C., Mammone, N., Hussain, A. and Morabito, F. C. 2020. A novel multimodal machine learning based approach for automatic classification of EEG recordings in dementia, *Neural Netw.* **123**: 176–190.

[74] Ilievski, I. and Feng, J. 2017. Multimodal learning and reasoning for visual question answering, *Advances in neural information processing systems* **30**.

[75] Ji, N., Ma, L., Dong, H. and Zhang, X. 2019. EEG signals feature extraction based on DWT and EMD combined with approximate entropy, *Brain Sci.* **9**(8): 201.

[76] Kaneshiro, B., Perreau Guimaraes, M., Kim, H.-S., Norcia, A. M. and Suppes, P. 2015. A representational similarity analysis of the dynamics of object processing using single-trial eeg classification, *Plos one* **10**(8): e0135697.

[77] Kaur, B., Singh, D. and Roy, P. P. 2018. Eeg based emotion classification mechanism in bci, *Procedia computer science* **132**: 752–758.

[78] Kavasidis, I., Palazzo, S., Spampinato, C., Giordano, D. and Shah, M. 2017. Brain2image: Converting brain signals into images, *Proceedings of the 25th ACM international conference on Multimedia*, pp. 1809–1817.

[79] Kevric, J. and Subasi, A. 2017. Comparison of signal decomposition methods in classification of EEG signals for motor-imagery BCI system, **31**: 398–406.

[80] Kim, M.-K., Kim, M., Oh, E. and Kim, S.-P. 2013. A review on the computational methods for emotional state estimation from the human eeg, *Computational and mathematical methods in medicine* **2013**.

[81] Koelstra, S., Mühl, C. and Patras, I. 2009. Eeg analysis for implicit tagging of video data, *2009 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops*, pp. 1–6.

[82] Kreilinger, A., Hiebel, H. and Müller-Putz, G. R. 2016. Single versus multiple events error potential detection in a bci-controlled car game with continuous and discrete feedback, *IEEE Transactions on Biomedical Engineering* **63**(3): 519–529.

[83] Lawhern, V., Hairston, W. D., McDowell, K., Westerfield, M. and Robbins, K. 2012. Detection and classification of subject-generated artifacts in eeg signals using autoregressive models, *Journal of neuroscience methods* **208**(2): 181–189.

[84] Lawhern, V. J., Solon, A. J., Waytowich, N. R., Gordon, S. M., Hung, C. P. and Lance, B. J. 2018. Eegnet: a compact convolutional neural network for eeg-based brain–computer interfaces, *Journal of neural engineering* **15**(5): 056013.

[85] LeCun, Y. 1998. The mnist database of handwritten digits, *http://yann. lecun. com/exdb/mnist/* .

[86] LeCun, Y. et al. 2015. Lenet-5, convolutional neural networks, *URL: http://yann. lecun. com/exdb/lenet* **20**(5): 14.

[87] Lee, H. K. and Choi, Y.-S. 2019. Application of continuous wavelet transform and convolutional neural network in decoding motor imagery brain-computer interface, *Entropy (Basel)* **21**(12): 1199.

[88] Li, H., Guo, Y.-j., Wu, M., Li, P. and Xiang, Y. 2010. Combine multi-valued attribute decomposition with multi-label learning, *Expert Systems with Applications* **37**(12): 8721–8728.

[89] LI, J., LIANG, J., ZHAO, Q., LI, J., HONG, K. and ZHANG, L. 2013. Design of assistive wheelchair system directly steered by human thoughts, *International Journal of Neural Systems* **23**(03): 1350013. PMID: 23627660.
URL: *https://doi.org/10.1142/S0129065713500135*

[90] Li, M., Luo, X., Yang, J. and Sun, Y. 2016. Applying a locally linear embedding algorithm for feature extraction and visualization of MI-EEG, *J. Sens.* **2016**: 1–9.

[91] Li, R., Johansen, J. S., Ahmed, H., Ilyevsky, T. V., Wilbur, R. B., Bharadwaj, H. M. and Siskind, J. M. 2020. The perils and pitfalls of block design for eeg classification experiments, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **43**(1): 316–333.

[92] Li, X., Zhang, P., Song, D., Yu, G., Hou, Y. and Hu, B. 2015. Eeg based emotion identification using unsupervised deep feature learning.

[93] Li, Y., Dzirasa, K., Carin, L., Carlson, D. E. et al. 2017. Targeting eeg/lfp synchrony with neural nets, *Advances in Neural Information Processing Systems* **30**.

[94] Li, Y., Pan, J., Wang, F. and Yu, Z. 2013. A hybrid bci system combining p300 and ssvep and its application to wheelchair control, *IEEE Transactions on Biomedical Engineering* **60**(11): 3156–3166.

[95] Lin, J.-S. and She, B.-H. 2020. A BCI system with motor imagery based on bidirectional long-short term memory, *IOP Conf. Ser. Mater. Sci. Eng.* **719**(1): 012026.

[96] Liu, A., Chen, K., Liu, Q., Ai, Q., Xie, Y. and Chen, A. 2017. Feature selection for motor imagery eeg classification based on firefly algorithm and learning automata. **URL:** *http://dx.doi.org/10.3390/s17112576*

[97] Liu, Y.-J., Yu, M., Zhao, G., Song, J., Ge, Y. and Shi, Y. 2017. Real-time movie-induced discrete emotion recognition from eeg signals, *IEEE Transactions on Affective Computing* **9**(4): 550–562.

[98] Lopes, A. C., Pires, G. and Nunes, U. 2013. Assisted navigation for a brain-actuated intelligent wheelchair, *Robotics and Autonomous Systems* **61**(3): 245–258. **URL:** *https://www.sciencedirect.com/science/article/pii/S0921889012002072*

[99] Lotte, F., Bougrain, L., Cichocki, A., Clerc, M., Congedo, M., Rakotomamonjy, A. and Yger, F. 2018. A review of classification algorithms for eeg-based brain–computer interfaces: a 10 year update, *Journal of neural engineering* **15**(3): 031005.

[100] Lyon, R. F. 2010. Machine hearing: An emerging field [exploratory dsp], *IEEE signal processing magazine* **27**(5): 131–139.

[101] Ma, W. J. and Peters, B. 2020. A neural network walks into a lab: towards using deep nets as models for human behavior, *arXiv preprint arXiv:2005.02181* .

[102] MacInnes, J., Santosa, S. and Wright, W. 2010. Visual classification: Expert knowledge guides machine learning, *IEEE Computer Graphics and Applications* **30**(1): 8–14.

[103] Makino, T., Jastrzębski, S., Oleszkiewicz, W., Chacko, C., Ehrenpreis, R., Samreen, N., Chhor, C., Kim, E., Lee, J., Pysarenko, K. et al. 2022. Differences between human and machine perception in medical diagnosis, *Scientific reports* **12**(1): 1–13.

[104] Mammone, N., Ieracitano, C. and Morabito, F. C. 2020. A deep CNN approach to decode motor preparation of upper limbs from time–frequency maps of EEG signals at source level, *Neural Netw.* **124**: 357–372.

[105] Mandel, C., Lüth, T., Laue, T., Röfer, T., Gräser, A. and Krieg-Brückner, B. 2009. Navigating a smart wheelchair with a brain-computer interface interpreting steady-state visual evoked potentials, *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, pp. 1118–1125.

[106] Marini, F., Breeding, K. A. and Snow, J. C. 2019a. Dataset of 24-subject eeg recordings during viewing of real-world objects and planar images of the same items, *Data in brief* **24**: 103857.

[107] Marini, F., Breeding, K. A. and Snow, J. C. 2019b. Distinct visuo-motor brain dynamics for real-world objects versus planar images, *Neuroimage* **195**: 232–242.

[108] Mason, S. and Birch, G. 2003. A general framework for brain-computer interface design, *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **11**(1): 70–85.

[109] Meziani, A., Djouani, K., Medkour, T. and Chibani, A. 2019. A lasso quantile periodogram based feature extraction for eeg-based motor imagery.
URL: *http://dx.doi.org/10.1016/j.jneumeth.2019.108434*

[110] Mirowski, P., Madhavan, D., LeCun, Y. and Kuzniecky, R. 2009. Classification of patterns of eeg synchronization for seizure prediction, *Clinical neurophysiology* **120**(11): 1927–1940.

[111] Mishra, A. and Bajwa, G. 2022. A new approach to visual classification using concatenated deep learning for multimode fusion of eeg and image data, *2022 17th International Symposium on Visual Computing (ISVC) In Press*, Springer, pp. –.
URL: *http://www.isvc.net/wp-content/uploads/2022/09/ISVC22_Final_Program.pdf*

[112] Mishra, A., Raj, N. and Bajwa, G. 2022. Eeg-based image feature extraction for visual classification using deep learning, *2022 Third International Conference on Intelligent Data Science Technologies and Applications (IDSTA) In Press*, IEEE, pp. –.
URL: *https://intelligenttech.org/IDSTA2022/IDSTApackingList/26_DTL2022_RC_8931.pdf*

[113] Molnar, C. 2020. *Interpretable machine learning*, Lulu. com.

[114] Moravec, H. 1988. *Mind children: The future of robot and human intelligence*, Harvard University Press.

[115] Mukherjee, P., Das, A., Bhunia, A. K. and Roy, P. P. 2019. Cogni-net: Cognitive feature learning through deep visual perception, *2019 IEEE International Conference on Image Processing (ICIP)*, IEEE, pp. 4539–4543.

[116] Müller, S. M. T., Bastos, T. F. et al. 2013. Proposal of a ssvep-bci to command a

robotic wheelchair, *Journal of Control, Automation and Electrical Systems* **24**(1): 97–105.

[117] Murugappan, M., Murugappan, S., Balaganapathy and Gerard, C. 2014. Wireless eeg signals based neuromarketing system using fast fourier transform (fft), *2014 IEEE 10th International Colloquium on Signal Processing and its Applications*, pp. 25–30.

[118] Nevatia, R. 1982. Machine perception., *PRENTICE-HALL, INC., ENGLEWOOD CLIFFS, NJ 07632, 1982, 209* .

[119] Ng, D. W.-K., Soh, Y.-W. and Goh, S.-Y. 2014. Development of an autonomous bci wheelchair, *2014 IEEE Symposium on Computational Intelligence in Brain Computer Interfaces (CIBCI)*, pp. 1–4.

[120] Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H. and Ng, A. Y. 2011. Multimodal deep learning, *ICML*.

[121] Nguyen, D., Tran, D., Sharma, D. and Ma, W. 2017. On the study of eeg-based cryptographic key generation, *Procedia computer science* **112**: 936–945.

[122] Ojala, T., Pietikainen, M. and Maenpaa, T. 2002. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, *IEEE Transactions on pattern analysis and machine intelligence* **24**(7): 971–987.

[123] Ortiz-Echeverri, C. J., Salazar-Colores, S., Rodríguez-Reséndiz, J. and Gómez-Loenzo, R. A. 2019. A new approach for motor imagery classification based on sorted blind source separation, continuous wavelet transform, and convolutional neural network, *Sensors (Basel)* **19**(20): 4541.

[124] O'Shea, K. and Nash, R. 2015. An introduction to convolutional neural networks, *arXiv preprint arXiv:1511.08458* .

[125] Otsu, N. 1979. A threshold selection method from gray-level histograms, *IEEE transactions on systems, man, and cybernetics* **9**(1): 62–66.

[126] Owens, A., Wu, J., McDermott, J. H., Freeman, W. T. and Torralba, A. 2016. Ambient sound provides supervision for visual learning, *European conference on computer vision*, Springer, pp. 801–816.

[127] Palazzo, S., Spampinato, C., Kavasidis, I., Giordano, D., Schmidt, J. and Shah, M. 2020. Decoding brain representations by multimodal learning of neural activity and visual features, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **43**(11): 3833–3849.

[128] Palazzo, S., Spampinato, C., Schmidt, J., Kavasidis, I., Giordano, D. and Shah, M. 2020. Correct block-design experiments mitigate temporal correlation bias in eeg classification, *arXiv preprint arXiv:2012.03849* .

[129] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. and Duchesnay, E. 2011. Scikit-learn: Machine learning in Python, *Journal of Machine Learning Research* **12**: 2825–2830.

[130] Pham, T., Ma, W., Tran, D., Nguyen, P. and Phung, D. 2013. Eeg-based user authentication in multilevel security systems, *International conference on advanced data mining and applications*, Springer, pp. 513–523.

[131] Phoha, S. 2014. Machine perception and learning grand challenge: situational intelligence using cross-sensory fusion, *Frontiers in Robotics and AI* **1**: 7.

[132] Phoha, S., Virani, N., Chattopadhyay, P., Sarkar, S., Smith, B. and Ray, A. 2014. Context-aware dynamic data-driven pattern classification, *Procedia Computer Science* **29**: 1324–1333.

[133] Picard, R. W. 2003. Affective computing: challenges, *International Journal of Human-Computer Studies* **59**(1-2): 55–64.

[134] Raghu, S., Sriraam, N., Temel, Y., Rao, S. V. and Kubben, P. L. 2020. Eeg based multi-class seizure type classification using convolutional neural network and transfer learning, *Neural Networks* **124**: 202–212.
URL: *https://www.sciencedirect.com/science/article/pii/S0893608020300198*

[135] Raschka, S. 2018. Mlxtend: Providing machine learning and data science utilities and extensions to python's scientific computing stack, *The Journal of Open Source Software* **3**(24).
URL: *http://joss.theoj.org/papers/10.21105/joss.00638*

[136] Rashid, M., Sulaiman, N., PP Abdul Majeed, A., Musa, R. M., Bari, B. S., Khatun, S. et al. 2020. Current status, challenges, and possible solutions of eeg-based brain-computer interface: a comprehensive review, *Frontiers in neurorobotics* p. 25.

[137] Razzak, M. I., Naz, S. and Zaib, A. 2018. Deep learning for medical image processing: Overview, challenges and the future, *Classification in BioApps* pp. 323–350.

[138] Rebsamen, B., Burdet, E., Guan, C., Teo, C. L., Zeng, Q., Ang, M. and Laugier, C.

2007. Controlling a wheelchair using a bci with low information transfer rate, *2007 IEEE 10th International Conference on Rehabilitation Robotics*, pp. 1003–1008.

[139] Rojas Q, M., Masip, D., Todorov, A. and Vitria, J. 2011. Automatic prediction of facial trait judgments: Appearance vs. structural models, *PloS one* **6**(8): e23323.

[140] Ruiz Blondet, M. V., Laszlo, S. and Jin, Z. 2015. Assessment of permanence of non-volitional eeg brainwaves as a biometric, *IEEE International Conference on Identity, Security and Behavior Analysis (ISBA 2015)*, pp. 1–6.

[141] Rusu, A. A., Večerík, M., Rothörl, T., Heess, N., Pascanu, R. and Hadsell, R. 2017. Sim-to-real robot learning from pixels with progressive nets, *Conference on Robot Learning*, PMLR, pp. 262–270.

[142] Scheirer, W. J., Anthony, S. E., Nakayama, K. and Cox, D. D. 2014. Perceptual annotation: Measuring human vision to improve computer vision, *IEEE transactions on pattern analysis and machine intelligence* **36**(8): 1679–1686.

[143] Serdar Bascil, M., Tesneli, A. Y. and Temurtas, F. 2015. Multi-channel eeg signal feature extraction and pattern recognition on horizontal mental imagination task of 1-d cursor movement for brain computer interface, *Australasian physical & engineering sciences in medicine* **38**(2): 229–239.

[144] Serov, A. 2013. Subjective reality and strong artificial intelligence, *arXiv preprint arXiv:1301.6359* .

[145] Simonyan, K. and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition, *arXiv preprint arXiv:1409.1556* .

[146] Snow, J. C. and Culham, J. C. 2021. The treachery of images: how realism influences brain and behavior, *Trends in Cognitive Sciences* **25**(6): 506–519.

[147] Snow, J. C., Skiba, R. M., Coleman, T. L. and Berryhill, M. E. 2014. Real-world objects are more memorable than photographs of objects, *Frontiers in human neuroscience* **8**: 837.

[148] Sohn, K., Shang, W. and Lee, H. 2014. Improved multimodal deep learning with variation of information, *Advances in neural information processing systems* **27**.

[149] Spampinato, C., Palazzo, S., Kavasidis, I., Giordano, D., Souly, N. and Shah, M. 2017. Deep learning human mind for automated visual classification, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6809–6817.

[150] *Steady-State VEP-Based Brain-Computer Interface Control in an Immersive 3D Gaming Environment | EURASIP Journal on Advances in Signal Processing | Full Text* n.d.. `https://asp-eurasipjournals.springeropen.com/articles/10.1155/ASP.2005.3156`.

[151] Tabar, Y. R. and Halici, U. 2017. A novel deep learning approach for classification of EEG motor imagery signals, *J. Neural Eng.* **14**(1): 016003.

[152] Tan, M. and Le, Q. 2019a. Efficientnet: Rethinking model scaling for convolutional neural networks, *International conference on machine learning*, PMLR, pp. 6105–6114.

[153] Tan, M. and Le, Q. V. 2019b. Efficientnet: Rethinking model scaling for convolutional neural networks, *CoRR* **abs/1905.11946**.
URL: *http://arxiv.org/abs/1905.11946*

[154] Tangian, A. 2001. How do we think: Modeling interactions of memory and thinking, *Cognitive Processing* **2**(1): 117–151.

[155] Tanguiane, A. S. 1993. *Artificial perception and music recognition*, Springer.

[156] Tao, Y., Sun, T., Muhamed, A., Genc, S., Jackson, D., Arsanjani, A., Yaddanapudi, S., Li, L. and Kumar, P. 2021. Gated transformer for decoding human brain eeg signals, *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, IEEE, pp. 125–130.

[157] Taran, S. and Bajaj, V. 2018. Drowsiness detection using adaptive hermite decomposition and extreme learning machine for electroencephalogram signals, *IEEE Sensors Journal* **18**(21): 8855–8862.

[158] Tello, R. M., Müller, S. M., Hasan, M. A., Ferreira, A., Krishnan, S. and Bastos, T. F. 2016. An independent-bci based on ssvep using figure-ground perception (fgp), *Biomedical Signal Processing and Control* **26**: 69–79.
URL: *https://www.sciencedirect.com/science/article/pii/S1746809415002086*

[159] *The Berlin Brain-Computer Interface presents the novel mental typewriter Hex-o-Spell. - MURAL - Maynooth University Research Archive Library* n.d.. `https://mural.maynoothuniversity.ie/1786/`.

[160] Thepade, S., Das, R. and Ghosh, S. 2014. A novel feature extraction technique using binarization of bit planes for content based image classification, *Journal of Engineering* **2014**.

[161] Thodoroff, P., Pineau, J. and Lim, A. 2016. Learning robust features using deep

learning for automatic seizure detection, *Machine learning for healthcare conference*, PMLR, pp. 178–190.

[162] Tian, G. and Liu, Y. 2019. Simple convolutional neural network for left-right hands motor imagery EEG signals classification, *Int. J. Cogn. Inform. Nat. Intell.* **13**(3): 36–49.

[163] *Towards BCI-actuated smart wheelchair system | BioMedical Engineering OnLine | Full Text* n.d.. `https://biomedical-engineering-online.biomedcentral.com/articles/10.1186/s12938-018-0545-x`. (Accessed on 08/08/2022).

[164] *Towards Development of a 3-State Self-Paced Brain-Computer Interface* n.d.. `https://www.hindawi.com/journals/cin/2007/084386/`.

[165] Tripathy, R. and Acharya, U. R. 2018. Use of features from rr-time series and eeg signals for automated classification of sleep stages in deep neural network framework, *Biocybernetics and Biomedical Engineering* **38**(4): 890–902.

[166] Tsinalis, O., Matthews, P. M., Guo, Y. and Zafeiriou, S. 2016. Automatic sleep stage scoring with single-channel eeg using convolutional neural networks, *arXiv preprint arXiv:1610.01683* .

[167] Türk, Ö. and Özerdem, M. S. 2019. Epilepsy detection by using scalogram based convolutional neural network from eeg signals, *Brain sciences* **9**(5): 115.

[168] ul Hassan, M., Mulhem, P., Pellerin, D. and Quénot, G. 2019. Explaining visual classification using attributes, *2019 International Conference on Content-Based Multimedia Indexing (CBMI)*, IEEE, pp. 1–6.

[169] Ullman, S., Assif, L., Fetaya, E. and Harari, D. 2016. Atoms of recognition in human and computer vision, *Proceedings of the National Academy of Sciences* **113**(10): 2744–2749.

[170] Vaineau, E., Barachant, A., Andreev, A., Rodrigues, P. C., Cattan, G. and Congedo, M. 2019. Brain invaders adaptive versus non-adaptive P300 brain-computer interface dataset, *CoRR* **abs/1904.09111**.
URL: *http://arxiv.org/abs/1904.09111*

[171] Vallat, R. 2018.
URL: *https://raphaelvallat.com/bandpower.html*

[172] Varona-Moya, S., Velasco-Álvarez, F., Sancha-Ros, S., Fernández-Rodríguez, Á., Blanca, M. J. and Ron-Angevin, R. 2015. Wheelchair navigation with an audio-cued, two-class motor imagery-based brain-computer interface system, *2015 7th International IEEE/EMBS Conference on Neural Engineering (NER)*, IEEE, pp. 174–177.

[173] Veit, A., Wilber, M. J., Vaish, R., Belongie, S. J., Davis, J., Anand, V., Aviral, A., Chakrabarty, P., Chandak, Y., Chaturvedi, S., Devaraj, C., Dhall, A., Dwivedi, U., Gupte, S., Sridhar, S. N., Paga, K., Pahuja, A., Raisinghani, A., Sharma, A., Sharma, S., Sinha, D., Thakkar, N., Vignesh, K. B., Verma, U., Abhishek, K., Agrawal, A., Aishwarya, A., Bhattacharjee, A., Dhanasekar, S., Gullapalli, V. K., Gupta, S., G, C., Jain, K., Kapur, S., Kasula, M., Kumar, S., Kundaliya, P., Mathur, U., Mishra, A., Mudgal, A., Nadimpalli, A., Nihit, M. S., Periwal, A., Sagar, A., Shah, A., Sharma, V., Sharma, Y., Siddiqui, F., Singh, V., S., A., Tambwekar, P., Taskin, R., Tripathi, A. and Yadav, A. D. 2015. On optimizing human-machine task assignments, *CoRR*

abs/1509.07543.

URL: *http://arxiv.org/abs/1509.07543*

[174] Vinyals, O., Babuschkin, I., Czarnecki, W. M., Mathieu, M., Dudzik, A., Chung, J., Choi, D. H., Powell, R., Ewalds, T., Georgiev, P. et al. 2019. Grandmaster level in starcraft ii using multi-agent reinforcement learning, *Nature* **575**(7782): 350–354.

[175] Wang, L., Lan, Z., Wang, Q., Yang, R. and Li, H. 2019. ELM_kernel and wavelet packet decomposition based EEG classification algorithm, *Autom. Contr. Comput. Sci.* **53**(5): 452–460.

[176] Wolpaw, J. R., Birbaumer, N., Heetderks, W. J., McFarland, D. J., Peckham, P. H., Schalk, G., Donchin, E., Quatrano, L. A., Robinson, C. J., Vaughan, T. M. et al. 2000. Brain-computer interface technology: a review of the first international meeting, *IEEE transactions on rehabilitation engineering* **8**(2): 164–173.

[177] Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K. et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation, *arXiv preprint arXiv:1609.08144* .

[178] Xu, T., Zhou, Y., Wang, Z. and Peng, Y. 2018. Learning emotions eeg-based recognition and brain activity: A survey study on bci for intelligent tutoring system, *Procedia computer science* **130**: 376–382.

[179] Yang, B., Li, H., Wang, Q. and Zhang, Y. 2016. Subject-based feature extraction by using fisher WPD-CSP in brain-computer interfaces, *Comput. Methods Programs Biomed.* **129**: 21–28.

[180] Yang, C., Wu, H., Li, Z., He, W., Wang, N. and Su, C.-Y. 2017. Mind control of a robotic arm with visual fusion technology, *IEEE Transactions on Industrial Informatics* **14**(9): 3822–3830.

[181] Yang, C., Wu, H., Li, Z., He, W., Wang, N. and Su, C.-Y. 2018. Mind control of a robotic arm with visual fusion technology, *IEEE Transactions on Industrial Informatics* **14**(9): 3822–3830.

[182] Yu, H.-F., Huang, F.-L. and Lin, C.-J. 2011. Dual coordinate descent methods for logistic regression and maximum entropy models, *Machine Learning* **85**(1): 41–75.

[183] Zarei, R., He, J., Siuly, S. and Zhang, Y. 2017. A pca aided cross-covariance scheme for discriminative feature extraction from eeg signals, *Computer methods and programs in biomedicine* **146**: 47–57.

[184] Zhang, H., Silva, F. H. S., Ohata, E. F., Medeiros, A. G. and Rebouças Filho, P. P. 2020. Bi-dimensional approach based on transfer learning for alcoholism pre-disposition classification via eeg signals, *Frontiers in Human Neuroscience* **14**.
URL: *https://www.frontiersin.org/article/10.3389/fnhum.2020.00365*

[185] Zhang, J., Yin, Z. and Wang, R. 2016. Pattern classification of instantaneous cognitive task-load through gmm clustering, laplacian eigenmap, and ensemble svms, *IEEE/ACM transactions on computational biology and bioinformatics* **14**(4): 947–965.

[186] Zhang, M.-L. and Zhou, Z.-H. 2006. Multilabel neural networks with applications to functional genomics and text categorization, *IEEE transactions on Knowledge and Data Engineering* **18**(10): 1338–1351.

[187] Zhang, M.-L. and Zhou, Z.-H. 2007. Ml-knn: A lazy learning approach to multi-label learning, *Pattern recognition* **40**(7): 2038–2048.

[188] Zhang, R., Liu, Z., Zhang, L., Whritner, J. A., Muller, K. S., Hayhoe, M. M. and Ballard, D. H. 2018. Agil: Learning attention from human for visuomotor tasks, *Proceedings of the european conference on computer vision (eccv)*, pp. 663–679.

[189] Zhang, Y., Ji, X. and Zhang, Y. 2015. Classification of eeg signals based on ar model and approximate entropy, *2015 International Joint Conference on Neural Networks (IJCNN)*, IEEE, pp. 1–6.

[190] Zheng, X., Chen, W., You, Y., Jiang, Y., Li, M. and Zhang, T. 2020. Ensemble deep learning for automated visual classification using eeg signals, *Pattern Recognition* **102**: 107147.

[191] Zhou, J., Meng, M., Gao, Y., Ma, Y. and Zhang, Q. 2018. Classification of motor imagery eeg using wavelet envelope analysis and LSTM networks, *2018 Chinese Control And Decision Conference (CCDC)*, IEEE.

List of codes

Appendix A

Code Implementation

The Python programming language was used to implement all the experiments in this thesis. Libraries like scikit-learn, numpy, pandas, and scipy were mostly used for data processing. The deep learning models were built using the TensorFlow framework. The complete codebase can be found in the repository referenced in Appendix C.

A.1 Key algorithms

Code A.1: Code implementation of Grayscale Image-encoding of EEG data

```
#method to convert signals to gray scale
def convert_to_grayscale(eeg_signal):
    """
    This function converts the EEG signal to grayscale image
    arguments :
    eeg_signal : a numpy array of one EEG signal
    returns :
    grayscale_image : a numpy array of 8 bit grayscale image strecthed in 4 units
    """
    #min_max scalar normalization
    x = (x - np.min(x)) / (np.max(x) - np.min(x))
```

```
12      canvas = []
13      #stretching the gray scale to 4 units
14      for i in range(4):
15          canvas.append(x)
16      canvas = np.array(canvas)
17      #return the gray scale image with 8 bit (0-255) pixel values
18      return np.uint8(canvas*255)
19
20  #method to stack signals vertically
21  def stack_signals(set_of_signals):
22      """
23      This function accect a EEG data with shape (length, channels) and returns a stacked
            image with shape (4*cannels, length)
24      arguments :
25      set_of_signals : a numpy array of EEG signals with shape (length, channels)
26      returns :
27      stacked_image : a numpy array of stacked grayscale image with shape (4*cannels, length)
28      """
29      #swap the axes to get the shape (channels, length)
30      set_of_signals = np.swapaxes(set_of_signals, 0, 1)
31
32      i = 0
33      #iterate over the channels and convert each signal to grayscale image, then stack them
            vertically
34      for signal in set_of_signals:
35          signal = convert_to_grayscale(signal)
36          if i == 0:
37              canvas = signal
38          else:
39              canvas = np.vstack((canvas, signal))
40          i += 1
41      return canvas
42
43  #print shape of original eeg data and then grayscale image-encoded data
44  print("Shape of original EEG data: ", EEG_data[0].shape)
45  print("Shape of grayscale image-encoded EEG data: ", stack_signals(EEG_data[0]).shape)
46  #sample output :
```

```
#Shape of original EEG data:  (420, 128)
#Shape of grayscale image-encoded EEG data:  (512, 420)
```

Note: The Python code below shows the 5-fold stratified cross-validation split of image data and the 5-fold stratified group cross-validation split of EEG data. The grouping was based on stimulus ID to ensure that there would be no bias in the training and testing set because each visual stimulus has one image feature but multiple EEG features (according to the number of subjects viewing the stimulus). For traditional ML classifiers, all five splits were run simultaneously. However, for deep learning classifiers, they were run separately to account for limited memory resources.

Code A.2: Code implementation of 5-fold cross validation split for EEG and image data

```
#cross validation modules
from sklearn.model_selection import StratifiedGroupKFold, StratifiedKFold

#methof to split eeg data into train and test sets with 5 stratified groups
def group_kfold_cv(feature, label, groups, k=5):
    """
    This function accepts the input data and labels and performs a stratifed group k-fold
        cross validation
    arugments :
    feature : input data
    label : input labels
    groups : groups of data with same visual stimulus
    k : number of folds
    returns :
    list of cross validation splits for the data
    """
    #define group kfold
    group_kfold = StratifiedGroupKFold(
        n_splits=k, shuffle=True, random_state=130)
    #split data into train and test
    split_grp = []
    for train_index, test_index in group_kfold.split(feature, label, groups):
```

```
22          train_data = [X[train_index], y[train_index]]
23          test_data = [X[test_index], y[test_index]]
24          split_grp.append([train_data, test_data])
25      return split_grp
26
27
28 #split data into train and test sets for eeg data with cross validation groups
29 split_grp_img = group_kfold_cv(X_eeg, y_eeg, stimulus_id)
30
31 print("Shape of features and labels for eeg data:", X_eeg.shape, y_eeg.shape)
32
33 for i in range(len(split_grp_img)):
34     print('Fold :', i+1)
35     print('Train set :', split_grp_img[i][0][0].shape, split_grp_img[i][0][1].shape)
36     print('Test set :', split_grp_img[i][1][0].shape, split_grp_img[i][1][1].shape)
37
38 #ouput looks like this
39 # Shape of features and labels for eeg data: (4224, 830, 128) (4224,)
40 # Fold : 1
41 # Train set : (3383, 830, 128) (3383,)
42 # Test set : (841, 830, 128) (841,)
43 # Fold : 2
44 # Train set : (3383, 830, 128) (3383,)
45 # Test set : (841, 830, 128) (841,).....
46
47 #method to split image data into train and test sets with stratified 5-fold cv splits
48 def kfold_cv(feature, label, k=5):
49     """
50     This function accepts the input data and labels and performs a stratifed k-fold cross
           validation
51     arugments :
52     feature : input data
53     label : input labels
54     k : number of folds
55     returns :
56     list of cross validation splits for the data
57     """
```

```
    #define kfold
    kfold = StratifiedKFold(n_splits=k, shuffle=True, random_state=130)
    #split data into train and test
    split_img = []
    for train_index, test_index in kfold.split(feature, label):
        train_data = [X[train_index], y[train_index]]
        test_data = [X[test_index], y[test_index]]
        split_img.append([train_data, test_data])
    return split_img

#split data into train and test sets for image data
split_img = kfold_cv(X_img, y_img)

print("Shape of features and labels for image data:", X_img.shape, y_img.shape)

for i in range(len(split_img)):
    print('Fold :', i+1)
    print('Train set :', split_img[i][0][0].shape, split_img[i][0][1].shape)
    print('Test set :', split_img[i][1][0].shape, split_img[i][1][1].shape)

#ouput looks like this
# Shape of features and labels for image data: (96, 224, 224, 3) (96,)
# Fold : 1
# Train set : (76, 224, 224, 3) (76,)
# Test set : (20, 224, 224, 3) (20,)
# Fold : 2
# Train set : (76, 224, 224, 3) (76,)
# Test set : (20, 224, 224, 3) (20,).....
```

Code A.3: Code implementation of subject-wise split of EEG data

```
#convert train eeg data to numpy array
def convert_to_numpy(eeg_d):
    """
    this converts list a tensor to numpy array
    """
    train_eeg_data_np = []
    for tensor in eeg_d:
```

```
8           tensor_np = tensor.numpy()
9           #swap axes
10          tensor_np = np.swapaxes(tensor_np, 0, 1)
11          train_eeg_data_np.append(tensor_np)
12      train_eeg_data_np = np.array(train_eeg_data_np)
13      return train_eeg_data_np
14
15  # funtion to append all subjects to  4D array of
16  # (samples, time, channels, subjects)
17  def make_2d_eeg_data(eeg_data):
18      """
19       this function Converts list of eeg data with n samples to numpy array
20      """
21      eeg_data_2d = []
22      for sub in eeg_data:
23          sun_np = convert_to_numpy(sub)
24          eeg_data_2d.append(sun_np)
25      eeg_data_2d = np.array(eeg_data_2d)
26      #move axis to the end
27      return np.moveaxis(eeg_data_2d, 0, -1)
28
29  #function to spilt the data according to the subjects
30  def split_subjects(eeg_dict):
31      """
32      this function splits the data according to the subjects
33      arguments:
34      eeg_dict : a list of dictionaries with keys 'eeg_tensor', 'image', 'label', 'subject'
35      returns:
36      eeg_data : a list of eeg features(4D) list per subjest with labels
37      """
38      #sort data according to subjects
39      data_dict = sorted(eeg_dict, key=itemgetter('subject'))
40      #group data according to subjects and convert to list
41      feature_list = []
42      label_list = []
43      for sub_id, eeg_data in groupby(data_dict, key=itemgetter('subject')):
44          sub_list = [[data['eeg'][:, 20:460], data['label']] for data in eeg_data]
```

```
45          feature_list.append([dta[0] for dta in sub_list])
46          label_list.append([dta[1] for dta in sub_list])
47      return make_2d_eeg_data(feature_list), label_list[0]
48
49
50 #print 1st element of the test dictionary and length of the dictionary
51 print("Length of the egg sample:", len(eeg_dict))
52 print("keys of each egg sample:", eeg_dict[0].keys())
53 #subject wise split
54 subwise_eeg_data, subwise_label = split_subjects(eeg_dict)
55 #prinn shape of the subject wise data
56 print("Shape of the subject wise data", subwise_eeg_data.shape)
57
58 #ouput:
59 # Length of the egg sample: 11682
60 # keys of each egg sample: dict_keys(['eeg', 'image', 'label', 'subject'])
61 # Shape of the subject wise data and label: (1947, 440, 128, 6)
```

A.2 Model design

This section contains the Python code implementation for the deep learning models described in Chapters 3 and 4. This is an excerpt of code; the complete code can be found in the repository referenced in Appendix C.

Code A.4: Code implementation of LSTM-based EEG Model (LEM)

```
# make model function
def build_model(n_timesteps=440, n_features=128, n_classes=39, lr=0.001):
    """This function accepts the input shape of EEG data and number of classes and returns a
        compiled model
    arugments :
    n_timesteps : number of timesteps in the input data
    n_features : number of electrode channels in the input data
    n_classes : number of classes in the output data
    lr : intial learning rate for the model
    returns :
    model : a compiled LSTM model
    """
    # define model as sequential
    model = Sequential()
    # LSTM and dropout layers stack
    model.add(Bidirectional(LSTM(units=50, return_sequences=True),input_shape=(n_timesteps,
        n_features)))
    model.add(Dropout(0.2))
    model.add(LSTM(units=128, return_sequences=True,))
    model.add(Dropout(0.2))
    model.add(LSTM(units=50, return_sequences=True))
    model.add(Dropout(0.2))
    # flatten layer
    model.add(Flatten())
    # dense layer with 128 neurons and relu activation
    model.add(Dense(128, activation='relu'))
    model.add(Dropout(0.5))
    # output layer with softmax activation
    model.add(Dense(n_classes, activation='softmax'))
```

```
    # build and compile model with adam optimizer and categorical crossentropy loss
    model.build(input_shape=(n_timesteps, n_features))
    model.compile(optimizer='adam',loss='categorical_crossentropy', metrics=['accuracy'])
    model.summary()
    return model

# set hyperparameters for model training (reduced learning rate and early stopping)
relr = tf.keras.callbacks.ReduceLROnPlateau(monitor='val_accuracy', factor=0.1, patience=2,
    min_lr=1e-5)
earlystop = tf.keras.callbacks.EarlyStopping(monitor='val_accuracy', patience=5,
    restore_best_weights= True )
callbacks = [relr, earlystop]

# instantiating model
LSTM_model = build_model(n_timesteps=440, n_features=128, n_classes=2, lr=0.001)
```

Code A.5: Code implementation of CNN-based Image Model (CIM)

```
#make model function
def build_model(input_shape, num_classes, base_lr=0.001):
    """
    This function accepts the input shape of image data and number of classes, returns a
        compiled model
    arguments:
    input_shape: shape of input image
    num_classes: number of classes
    base_lr: initial learning rate
    returns:
    model: compiled CNN model
    """
    #setting base model as ResNet with imagenet weights
    base_model = tf.keras.applications.ResNet50(weights='imagenet', include_top=False)
    base_model.trainable = True
    #setting input layer with input shape
    image_imput = tf.keras.Input(shape=input_shape, name='image_input')
    #passing the input layer through base model
    x = base_model(image_imput)
    #global average pooling layer to reduce the number of parameters
```

```
    x = tf.keras.layers.GlobalAveragePooling2D()(x)
    #dropout layer to reduce overfitting
    x = tf.keras.layers.Dropout(0.2)(x)
    #dense layer with 128 neurons and relu activation function
    x = tf.keras.layers.Dense(128, activation="relu", name="deep_feature")(x)
    x = tf.keras.layers.Dropout(0.2)(x)
    #softmax activation layer with number of classes as output
    output = tf.keras.layers.Dense(num_classes, activation='softmax', name='output')(x)
    #creating model
    model = tf.keras.Model(inputs=image_imput, outputs=output)
    #settign up the model optimizer
    sgd = tf.keras.optimizers.SGD(lr=base_lr, momentum=0.9, nesterov=True)
    #compile model with loss function, optimizer and metrics
    model.compile(optimizer=sgd, loss='categorical_crossentropy',metrics=['accuracy'])
    model.summary()
    #return model
    return model

# set hyperparameters for model training (reduced learning rate and early stopping)
reduce_lr = tf.keras.callbacks.ReduceLROnPlateau(monitor='val_loss', factor=0.1, patience=3,
     min_lr=0.00001)
early_stopping = tf.keras.callbacks.EarlyStopping(monitor='val_loss', patience=10,
    restore_best_weights=True)
callbacks = [reduce_lr, early_stopping]

# instantiating model
CNN_model = build_model(img_input_shape, num_classes)
```

Code A.6: Code implementation of GEM-based Concatenated model

```
#make model function
def build_model(eeg_input_shape, img_input_shape, num_classes, base_lr=0.001):
    """
    This function accepts the input shape of stimuli data (both EEG and Image) and the
        number of classes and returns a concatenated model.
    arguments:
    eeg_input_shape: shape of EEG data
    img_input_shape: shape of Image data
```

```
    num_classes: number of classes
    base_lr: initial learning rate
    returns:
    model: compiled concatenated model
    """
    #eeg base model with 2d convolutions take Grayscale image-encoded EEG data as input
    eeg_base_model = tf.keras.applications.EfficientNetB2(weights='imagenet', include_top=
        False)
    eeg_base_model.trainable = True

    #image base model with 2d convolutions take RGB image data as input
    img_base_model = tf.keras.applications.vgg16.VGG16(include_top=False, weights='imagenet'
        , input_shape=(224, 224, 3), classes=num_classes)
    img_base_model.trainable = False

    #create input layers for EEG and Image data
    eeg_input = tf.keras.Input(shape=eeg_input_shape, name='eeg_input')
    img_input = tf.keras.Input(shape=img_input_shape, name='img_input')

    #image model layers
    img_res = (224, 224)
    #resize image data to 224x224 for VGG16
    imgl = tf.keras.layers.Lambda(lambda x: tf.image.resize(x, img_res))
    imgl = img_base_model(imgl(img_input))
    imgl = tf.keras.layers.GlobalAveragePooling2D()(imgl)
    imgl = tf.keras.layers.Dropout(0.4)(imgl)
    imgl = tf.keras.layers.Dense(512, activation='relu')(imgl)
    imgl = tf.keras.layers.Dropout(0.4)(imgl)
    #dense layer to extract spatial features from image data
    imgl = tf.keras.layers.Dense(128, activation='relu', name="Img_Deep_feature")(imgl)

    #eeg model layers
    #adjusting the input layer to take subject data in rgb channels
    dense_filter = tf.keras.layers.Conv2D(3, 3, padding="same")(eeg_input)
    eegl = eeg_base_model(dense_filter)
    eegl = tf.keras.layers.GlobalAveragePooling2D()(eegl)
    eegl = tf.keras.layers.Dropout(0.2)(eegl)
```

```
43      #dense layer to extract temporal features from EEG data
44      eegl = tf.keras.layers.Dense(128, activation="relu", name="eeg_deep_feature")(eegl)
45
46      #concatenate the two feature modalities
47      x = tf.keras.layers.concatenate([eegl, imgl])
48      # #softmax activation layer model output
49      model_output = tf.keras.layers.Dense(num_classes, activation='softmax', name='output')(x
            )
50
51      #create model
52      model = tf.keras.Model(inputs=[eeg_input, img_input], outputs=model_output)
53      #settign up the model optimizer, loss function and metrics
54      adam = tf.keras.optimizers.Adam(lr=base_lr, beta_1=0.9, beta_2=0.999, epsilon=None,
            decay=0.0, amsgrad=False)
55      losses = {'output': 'categorical_crossentropy'}
56      metrics = {'output': 'accuracy'}
57      #compile model
58      model.compile(optimizer=adam, loss=losses, metrics=metrics)
59      model.summary()
60      return model
61
62  # set hyperparameters for model training (reduced learning rate and early stopping)
63  reduce_lr = tf.keras.callbacks.ReduceLROnPlateau(monitor='val_loss', factor=0.1, patience=3,
        min_lr=0.00001)
64  early_stopping = tf.keras.callbacks.EarlyStopping(monitor='val_loss', patience=10,
        restore_best_weights=True)
65  callbacks = [reduce_lr, early_stopping]
66
67  # instantiating the concatenated model from both EEG and Image data
68  Concat_model = build_model(eeg_input_shape, img_input_shape, num_classes)
```

Appendix B

Abbreviations

Abbreviation	**Description**
actiCAP	actiCAP an instrument to measure EEG
AIC	Akaike Information Criterion
ALS	Amyotrophic Lateral Sclerosis
AI	Artificial Intelligence
AR	Autoregressive
Bi-LSTM	Bidirectional Long-short term memory
BCI	Brain–Computer Interface
Cz (like C3 or C4)	Central Electrodes
CIM	CNN (convolutional neural network) based Image model
CSP	Common Spatial Pattern
CWT	Continuous wavelet transform
CNN	Convolutional Neural Network

DFA	Detrended Fluctuation Analysis
DWT	Discrete Wavelet Transform
ECoG	Electrocorticography
EEG	Electroencephalogram
EMD	Empirical Mode Decomposition
ErrP	Error Related Potential
ERP	Event Related Potential
FFT	Fast Fourier Transform
fMRI	functional Magnetic Resonance Imaging
fNIRS	Functional near-infrared spectroscopy
GRU	Gated Recurrent Unit
GLCM	Gray-Level Co-occurrence Matrix
GEM	Grayscale-image Encoded EEG (Electroencephalogram) model
Hz	Hertz
HOS	Higher-Order Statistic
HFD	Higuchi Fractal Dimension
HT	Hilbert transform
HOG	Histogram of Gradients
HCI	Human-Computer Interactions
ITR	Information Transfer Rate
KNN	K Nearest Neighbours
LSTM	Long-short term memory

LEM	LSTM (long-short term memory) based EEG (Electroencephalogram) model
ML	Machine Learning
MEG	Magnetoencephalography
ms	millisecond
MNIST	Modified National Institute of Standards and Technology
MI	Motor Imagery
MND	Motor Neuron Disorders
MRCP	Movement-Related Cortical Potential
MLP	Multilayer Perceptron
1D	one dimensional
PFD	Petrosian Fractal Dimension
PET	Positron Emission Tomography
PSD	Power Spectral Density
PCA	Principal component analysis
RBF	Radial basis function
RNN	Recurrent Neural Network
ResNET	Residual Neural Network
SMR	Sensorimotor rhythms
SFS	Sequential Feature Selection
STFT	Short-time Fourier transforms
SCP	Slow Cortical Potential

SSVEP	Steady state visually evoked potential
SSAEP	Steady-State Auditory Evoked Potential
SSEPs	Steady-State Evoked Potentials
SSSEP	Steady-State Somatosensory Evoked Potential
P300	Stimulus is given for 300 ms in parietal regions
SVM	Support Vector Machines
RR time	The time elapsed between two successive R-waves of the QRS signal on the electrocardiogram
3D/3-D	Three dimensional
2D	Two dimensional
VGG	Variable Geometry Group
WPA	Wavelet packet analysis
WPD	Wavelet Packet Decomposition

Appendix C

Resources

Description	**Link**
A library for Python that provides tools to analyse electroencephalography (EEG) signals. This library is mainly a feature extraction tool that includes many frequently used algorithms in EEG processing using a sliding window approach	`https://github.com/Xiul109/eeglib`
A python based tutorial to compute the average power of a signal in a specific frequency range, using spectral estimation methods such as periodogram, Welch and multitaper.	`https://raphaelvallat.com/bandpower.html`

A Github repository of all the code implementations for experiments performed in Chapters 2,3 and 4. Please note: the repository is private until the work presented in the thesis is published.	Github link: `https://github.com/alankritmishra/Enhanced_Computervision_via_EEG`
A clone of Github repository is provided through the Google Drive link for now and is available upon request.	Gdrive link: `http://tiny.cc/Alankrit_CV_thesis_repo`
The flow diagrams and model configurations used throughout this thesis are designed using a web tool called drawio.	`https://github.com/jgraph/drawio`
The high GPU memory model training for this work was performed on the Python notebook known as "Google colab."	`https://colab.research.google.com`