Hemodynamic Analysis for Cognitive Load Assessment in Mathematical Problem Solving Task

A Thesis

Submitted in partial fulfillment of the requirement for the award of the Degree of

Master of Technology in Intelligent Automation and Robotics offered by

Department of Electronics and Telecommunication Engineering

By

Priyanka Paul

Registration No.: 160232 of 2021-2022

Examination Roll No.: M6IAR24014

Under the guidance of

Prof. Amit Konar

Department of Electronics and Telecommunication Engineering,

Jadavpur University, Kolkata-700032

India

May, 2024

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

CERTIFICATE

This is to certify that the dissertation entitled "**Hemodynamic Analysis for Cognitive Load Assessment in Mathematical Problem Solving Task**" has been carried out **Priyanka Paul** (University Registration No: **160232** of 2021-2022, Examination Roll No.: **M6IAR24014**) under my guidance and supervision and be accepted in partial fulfillment of the requirement for the Degree of Master of Technology in Intelligent Automation and Robotics offered by Department of Electronics & Telecommunication Engineering. The research results presented in the thesis have not been included in any other paper submitted for the award of any degree to any other University or Institute.

_____

Prof. Amit Konar
Supervisor,
Dept. of Electronics & Telecommunication Engineering,
Jadavpur University

_____

Prof. Manotosh Biswas
Head of the Department,
Dept. of Electronics & Telecommunication Engineering,
Jadavpur University

_____

Prof. Dipak Laha
Dean,
Faculty Council of Engineering and Technology,
Jadavpur University

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

CERTIFICATE OF APPROVAL*

The forgoing thesis is hereby approved as a creditable study of an engineering subject and presented in a manner satisfactory to warrant acceptance as prerequisite to the degree for which it has been submitted. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn there in but approve the thesis only for which it is submitted.

Committee on final examination for the evaluation of the thesis

_____

Signature of the Examiner

_____

Signature of the Supervisor

*Only in the case the thesis is approved

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC THESIS

I hereby declare that the thesis entitled "**Hemodynamic Analysis for Cognitive Load Assessment in Mathematical Problem Solving Task**" contains literature survey and original research work by the undersigned candidate, as part of her Degree of Technology in Intelligent Automation and Robotics offered by Department of Electronics and Telecommunication Engineering.

All information has been obtained and presented with academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Name: PRIYANKA PAUL

Class Roll No: 002110704016

Thesis Title: Hemodynamic Analysis for Cognitive Load Assessment in Mathematical Problem Solving Task

Signature of the Candidate

ACKNOWLEDGEMENT

First and foremost, I would like to express my earnest gratitude and heartfelt indebtedness to my advisor, Prof. Amit Konar, Department of Electronics and Telecommunication Engineering, for the privilege and the pleasure, of allowing me to work under him towards my Degree of Master of Technology in Intelligent Automation and Robotics under Electronics and Telecommunication Engineering department. This work would not have been materialized without his whole-hearted help and support. Working under him has been a great experience. I sincerely thank my supervisor, particularly for all the faith he had in me. I am thankful to Prof. Manotosh Biswas who has acted as Head of the Department of Electronics and Telecommunication Engineering during the tenure of my studentship. I would also like to show my special gratitude to Ms. Sayantani Ghosh, PhD Research Scholar under the Department of Electronics and Telecommunication Engineering (ETCE), for her constant guidance and valuable advice. I would also like to thank Mr. Rupayan Halder and Mr. Apurba Nandi, M.Tech student under the Department of ETCE for their help in my work throughout the entire time.

Lastly, I would like to thank my parents for their love, support and guidance through the course of the present work.

Date: _____

PRIYANKA PAUL
M.Tech in Intelligent Automation and Robotics
Dept. of Electronics & Telecommunication Engineering
Jadavpur University
Registration No.: 160232 of 2021-2022
Examination Roll No.:M6IAR24014

PREFACE

The field of Brain-Computer Interface (BCI) technology represents a significant frontier in the advancement of human-computer interaction. BCIs enable direct communication between the brain and external devices, providing remarkable opportunities for individuals with severe disabilities to interact with their environment without relying on traditional muscle-based actions. This work aims to explore the feasibility, effectiveness, and future potential of BCIs, with a particular focus on the measurement of cognitive load using functional Near-Infrared Spectroscopy (fNIRS).

The objective of this work is to contribute to the development of reliable methods for cognitive load measurement, which is essential for optimizing instructional design, enhancing user experience, and advancing human-computer interaction. By integrating fNIRS with other physiological and behavioral measures, this research aims to provide a comprehensive understanding of cognitive load dynamics and propose standardized protocols to facilitate broader adoption and comparability of results across studies.

The first chapter introduces Brain-Computer Interfaces (BCIs), discussing their historical evolution, the biological basis of brain signals, and the development of electroencephalography (EEG) as a precursor to modern BCIs. It also covers BCI principles, focusing on improving communication and control for individuals with severe disabilities. The second chapter explores the principles and technical aspects of fNIRS, detailing instrumentation, optode setup, and addressing challenges like motion artifacts and signal quality.The third chapter delves into signal processing, while the fourth focuses on classification techniques. The fifth chapter analyzes cognitive load, followed by the sixth chapter, which presents our experimental approach and results. The seventh chapter provides conclusions, and the final chapter discusses future research directions. An appendix is included to guide executing the code developed during this study.

Table of Contents

Chapter No.		Page No.
1. Introduction		1
1.1.	Brain Computer Interface	2
1.2.	History	3
1.3.	Biological Background of Brain	6
1.4.	Types of BCI	
1.5.	Challenges and Future Directions of BCIs	25
1.6.	Literature Review	26
1.7.	Scope of Work	28
1.8.	References	30
2. Functional Near-Infrared Spectroscopy		33
2.1.	Introduction	34
2.2.	History	
2.3.	Physical Principle of fNIRS	35
2.4.	fNIRS Cap	38
2.5.	Spectroscopic Techniques	40
2.6.	Steps Involved in fNIRS Device	41
2.7.	Noises in fNIRS Device	46
2.8.	Data Analysis Software	47
2.9.	Advantages of fNIRS	49
2.10.	Disadvantages of fNIRS	50
2.11.	Applications	51
2.12.	References	53
3. Signal Processing		56
3.1	Introduction	57
3.2	Categories	57
3.3	How Signal Processing involve in fNIRS	59
3.4	Feature Extraction	62
3.5	References	69
4. Classification		71
4.1	Introduction to Machine Learning	72

4.1.1 How Does Machine Learning Work 73
4.1.2 Classification with Machine Learning 75
4.2.3 List of Machine Learning Classification Algorithms 76

4.2 Deep Learning 76
4.2.1 Introduction to Deep Learning 76
4.2.2 Types of Algorithms used in Deep Learning 80
4.2.3 Convolutional Neural Networks (CNNs) 81
4.3 Reference 93

5. Cognitive Load Analysis 96

5.1 Introduction 97
5.2 History 97
5.3 What is Cognitive Load? 98
5.4 Types of Cognitive Load 99
5.5 Human Cognitive Architecture 100
5.6 Measurement of Cognitive Load 103
5.7 Effects of Heavy Cognitive Load 105
5.8 Sub-Population Studies 107
5.9 Advantages of Cognitive Load Analysis 109
5.10 Challenges in Cognitive Load Analysis 110
5.11 Applications 111
5.12 References 113

6. Cognitive Load Measurement using fNIRS 115

6.1 Abstract 116
6.2 Introduction 116
6.3 System Overview 117
6.4 Principle and Methodologies 119
6.5 Experiment and Result 125
6.6 References 133

7. Conclusion 136
8. Future Scope of Work 139

Appendix 144

List of Figures

Figure No. **Page No.**

Figure 1(a)	Three main parts of the brain	4
Figure 1(b)	Lobes of the human brain	4
Figure 1.(c)	An image of an EEG device set-up	7
Figure 1.(d)	Gold coated EEG electrodes	8
Figure 1.(e)	An image of a conventional EEG cap	8
Figure 1.(f)	The international 10-20 electrode placement system	9
Figure 1.(g)	The international 10-10 electrode placement system	10
Figure 1.(h)	Different brain waves	13
Figure 1.(i)	An image of a MEG scanner with patient	18
Figure 1.(j)	A functional Magnetic resonance imaging device	20
Figure 1.(k)	A whole brain fNIRs device	21
Figure 1.(l)	Positron Emission Tomography scanner device	22
Figure 1.(m)	Intracortical Acquistion Technique	23
Figure 1.(n)	The microelectrode array implanted into the cortex	23
Figure 1.(o)	The grid placement in ECoG	24
Figure 2.(a)	Hitachi ETG-4000	35
Figure 2.(b)	Functional near-infrared spectroscopy	36
Figure 2.(c)	Schematic channel location and measurement principle of fNIRS device	37
Figure 2.(d)	Absorption spectra for Hb and HbO	38
Figure 2.(e)	The international 10-10 electrode placement system	39
Figure 3.(a)	Visualization of the signal processing approaches used	61
Figure 3.(b)	fNIRs signal processing pipelines	62
Figure 3.(c)	Visual representation of feature extraction	64
Figure 3.(d)	Feature engineering lifecycle	67
Figure 4.(a)	Types of machine learning	73

Figure 4.(b)	Steps for machine learning algorithm	75
Figure 4.(c)	Convolutional neural network	82
Figure 4.(d)	Max. pooling operation	84
Figure 4.(e)	Some activation functions and their properties	85
Figure 4.(f)	Characteristics Comparison of Mish and Swish with other activation functions	86
Figure 4.(g)	Convolutional neural network	86
Figure 4.(h)	Visual representation of Convolutional neural network	88
Figure 5.(a)	Cognitive load theory	100
Figure 5.(b)	Standard model of cognitive architecture	102
Figure 5.(c)	Human brain inspired distributed cognitive architecture	102
Figure 5.(d)	Experimental Set up to measure cognitive load	105
Figure 5.(e)	Analysis of cognitive load	108
Figure 6.(a)	Block diagram of cognitive load analysis	118
Figure 6.(b)	Block diagram of proposed CNN architecture	121
Figure 6.(c)	Montage	125
Figure 6.(d)	Optode arrangement's topo layout	125
Figure 6.(e)	Experimental set-up : subject participating in the experiment	126
Figure 6.(f)	Stimuli diagram	127
Figure 6.(g)	Output graph of wavelet transform from trial	128

List of Tables

Table No.		**Page No.**
Table 6.1 :	Classification Accuracy of Various Algorithms for Cognitive Load Analysis	130
Table 6.2 :	Classification Accuracy with Activation Function	131

CHAPTER 1

Introduction

1.1 Brain Computer Interface

After the invention of computers, there has been significant interest in exploring the interaction between humans and computers. This has led to the evolution of computer interfaces from basic methods like perforated cards, keyboards, and mice to more advanced technologies such as touch screens and speech recognition. Human-Computer Interface (HCI) research aims to create user-friendly, efficient, and ergonomic systems for faster and more comfortable communication between humans and computers. Interfaces based on gesture recognition, voice recognition, and physical movement have gained considerable attention and have become commercially available.

Brain-Computer Interface (BCI) [1] technology has emerged as an interdisciplinary research area that enables direct interaction between the brain and computers using signals generated by the brain itself. BCIs measure neural activity in the brain, process this activity, and convert it into outputs that can be used by computers. The primary goal of BCI technology is to provide alternative means of communication for individuals with severe disabilities, allowing them to communicate and control devices without relying on muscle control. BCIs can improve motor and sensory functions in people with physical impairments, enabling them to achieve greater sensory efficiency.

BCIs utilize brain imaging technology to detect brain activity in real-time. The electrical signals generated by the brain are digitized and interpreted as digital information, which can then be used to control devices such as prosthetic limbs or spelling machines. The process of BCI operation involves three main steps: gathering brain signals, interpreting them, and generating appropriate outputs for connected devices.

BCI technology [2] is based on the workings of the brain, which consists of neurons, dendrites, and axons interconnected to transmit electrical signals. These signals are generated when a person thinks, moves, feels, or remembers something, and they travel between neurons at high speeds. Although most signals are contained within the neurons' myelin sheaths, some escape and can be detected and interpreted to direct devices effectively.

1.2 History

The history of brain-computer interfaces (BCIs) began with Hans Berger's discovery of human brain electrical activity and the development of electroencephalography (EEG) in 1924. Berger's rudimentary EEG recordings using silver wires evolved into more sophisticated techniques, leading to new insights into brain diseases.

Jacques Vidal coined the term "BCI" and published the first peer-reviewed papers on the topic. His 1973 paper outlined the "BCI challenge" of controlling external objects using EEG signals, and in 1977, he demonstrated noninvasive EEG control of a graphical object on a computer screen.

In 1988, a report described noninvasive EEG control of a robot, while in 1990, a closed-loop BCI controlled a computer buzzer using anticipatory brain potentials. Recent studies have explored neural stimulation's potential to restore functional connectivity and behaviors. Since 2013, DARPA has funded BCI technology through the BRAIN initiative, supporting research at various institutions.

1.3 Biological Background of Brain

The brain is a complex and vital organ responsible for controlling and coordinating various bodily functions and processes. It is composed of billions of neurons, specialized cells that transmit electrochemical signals, and glial cells, which support and nourish neurons. The neurons form intricate networks through which information is processed and transmitted.

The human brain consists of over 1 trillion cells, with approximately 1 billion being nerve cells interconnected in a neural network. Similar to the AND-OR circuitry in computers, neurons in the brain can be connected in various configurations, including one-to-one, one-to-many, and many-to-many. Different regions of the brain form a network that shares information and handles various mental tasks. The brain is divided into three main parts: the cerebrum (or cortex), the cerebellum (or little brain), and the brainstem.

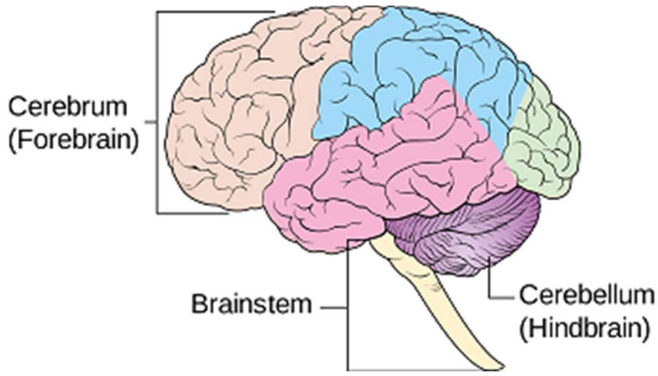


Figure 1(a). Three main parts of the brain

The cerebral cortex, the brain's outer layer of neuronal tissue, is approximately 2 to 4 millimeters thick and is located directly behind the skull. It is responsible for tasks such as thought, language, memory, attention, and higher-level mental processes. The cerebral cortex is subdivided into several areas, including the motor cortex, visual cortex, and somatosensory cortex. Additionally, it can be classified into four lobes based on topographical conventions: the temporal lobe, frontal lobe, parietal lobe, and occipital lobe [14].

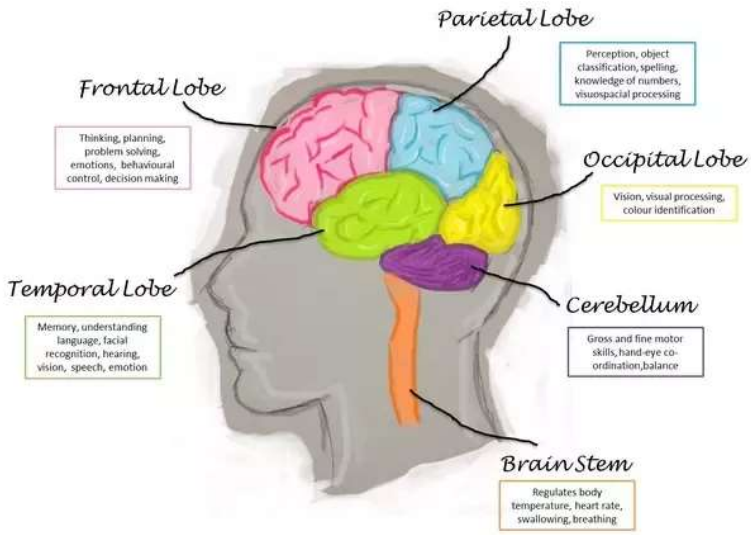


Figure 1(b). Lobes of the human brain

The human brain is divided into four main lobes, each with distinct functions and responsibilities:

I. Frontal Lobe

Location: Located at the front of the brain, behind the forehead.

Functions: The frontal lobe is involved in a wide range of higher cognitive functions, including motor control, problem-solving, decision-making, planning, reasoning, impulse control, personality expression, and social behavior. It also plays a crucial role in speech production and language comprehension.

II. Parietal Lobe

Location: Positioned at the top of the brain, behind the frontal lobe and above the temporal lobe.

Functions: The parietal lobe is primarily responsible for processing sensory information from the body, including touch, temperature, pain, pressure, and proprioception (awareness of body position and movement). It also plays a role in spatial orientation, attention, perception of numerical quantities, and object manipulation.

III. Temporal Lobe

Location: Located on the sides of the brain, beneath the lateral fissure.

Functions: The temporal lobe is involved in auditory processing, language comprehension, memory formation (especially declarative memory), emotion regulation, and visual perception. Specific areas within the temporal lobe are responsible for recognizing faces, understanding spoken language, and forming new memories.

IV. Occipital Lobe

Location: Positioned at the back of the brain, behind the parietal and temporal lobes.

Functions: The occipital lobe is primarily dedicated to processing visual information received from the eyes. It plays a crucial role in visual perception, including the detection of shapes, colors, motion, depth, and spatial relationships. The occipital lobe is responsible for forming mental images, recognizing objects, and interpreting visual stimuli.

1.4 Types of BCI

BCI can be classified into two distinct approaches, which are:

1.4.1 Non-invasive BCIs

A non-invasive approach involves placing sensors on the skin, typically on the scalp, without the need for surgical implantation. These sensors allow for the recording and reception of neural activity without penetrating the patient's skull. Patients can simply wear the device on their head, eliminating the need for surgery to capture brain signals. Non-invasive recording methods in BCI (Brain-Computer Interface) technology allow for the acquisition of brain signals without the need for surgical intervention. Here's a detailed overview of some common non-invasive techniques:

1.4.1.1 Electroencephalography (EEG)

EEG,[3] short for Electroencephalography, is a non-invasive neuroimaging technique that measures the electrical activity of the brain. It involves placing electrodes on the scalp to detect the voltage fluctuations resulting from the synchronized activity of neurons. EEG recordings provide insight into the brain's electrical patterns, allowing researchers and clinicians to study brain function, diagnose neurological conditions such as epilepsy, and interface with computers for various applications, including brain-controlled interfaces and neurofeedback. EEG is known for its high temporal resolution, capturing changes in brain activity with millisecond precision, making it suitable for real-time monitoring and analysis.



Figure 1(c). An image of an EEG device set-up

Electroencephalography (EEG) [6] is a neuroimaging technique that measures the electrical activity generated by the brain. Here are more details about EEG:

1) Principle

EEG relies on the detection of electrical potentials generated by the activity of neurons in the brain. Neurons communicate with each other through electrical impulses, and when millions of neurons fire together, they produce electrical fields that can be detected on the scalp.

2) Setup

EEG involves placing electrodes on the scalp using a special cap or headset. These electrodes are typically made of conductive materials and are positioned according to standardized electrode placement systems, such as the International 10-20 system. The electrodes are connected to an amplifier, which amplifies the signals detected by the electrodes.



Figure 1(d). Gold coated EEG electrodes



Figure 1(e). An image of a conventional EEG cap

3) EEG device cap structure and electrode position

The EEG cap at least have three electrodes i.e., a ground, a recording electrode and a reference electrode to record the EEG signals. Figure: along with an amplified, an analog to digital converter, a computing device such as a computer is also needed. The electrodes are consisting of silver, silver chloride or gold and can be considered as wet, which requires an adhesive paste to be placed between the electrodes and the scalp. The reference electrode is considered to be the common electrode for

measurement of all electrodes. For the comparisons of potential difference, the active and the reference electrodes serve as a signal receptor. From the active and the reference signal, the insignificant data are diminished with the help of the ground electrode which serve as the baseline of brainwave signals. In order to maintain the placement of EEG electrodes reliably and precisely, it is necessary to ensure that the electrodes are placed properly in the cortical areas to the human subject. For half a century the 10-20 system shown in Figure is considered as the standard system for positioning the electrodes on the scalp and also ensures the spacing between the electrodes with 21 number of electrodes. Further, the 10-20 system (for the effectiveness of 10/20-derived systems, see Jurcak et al. 2007) has been extended to achieve a higher density of electrode settings under this 10-10 or the 10-5 systems that shown in Figure to allow more than 500 electrode position The area of the scalp is divided into six portions under this 10-20 system, in which from nation to anion with an interval rates of 10%,20%,20%,20%,20% and 10%(Fp: prefrontal, F: frontal, C: central, P: parietal, O: occipital, respectively).The electrode placement location on the scalp has a letter to identify the lobe or area of the brain by giving its name from parietal (P), temporal (T), prefrontal (Fp),occipital (O), frontal (F), central (C). A zero (Z) reference electrode is also present on the midline of the sagittal plane of the skull (Fpz, Fz, Cz, Oz). On the right hemisphere of the brain, the electrodes with even numbers are placed with odd numbers placed on the left hemisphere of the brain. The skull circumference is measured from just above the ears (T3, T4), from the nose bridge (at Fpz) and from the occipital point (at Oz).

Nowadays one of the most popular BCI devices is EEG because it is non-invasive, inexpensive, and portable to use. These features of the EEG device make it one of the trendiest devices in current research. However, mathematically it is quite difficult to measure exactly the amount of current flowing inside the brain that produces the signals.

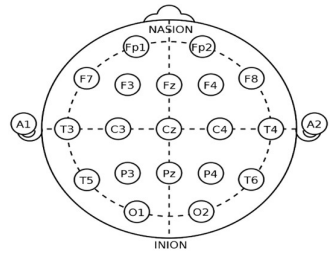


Figure 1(f). The international 10–20 electrode placement system

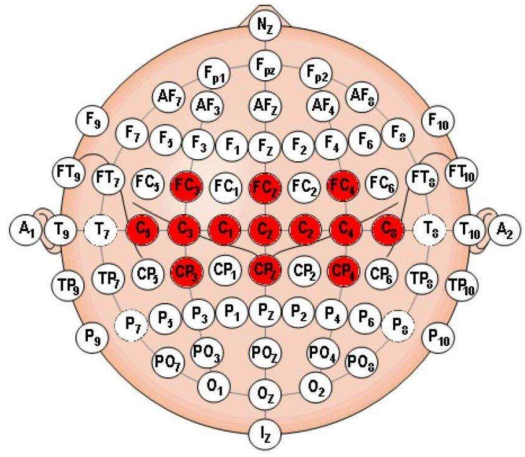


Figure 1(g). The international 10-10 electrode placement system.

4) Recording

EEG recordings are obtained by measuring the voltage differences between pairs of electrodes. The electrodes detect the electrical activity from the underlying brain regions, and the resulting signals are amplified, filtered, and digitized for analysis.

5) Frequency Bands of EEG signal

EEG signals can be decomposed into different frequency bands, each associated with different brain states and functions. The main frequency bands include:

a. Delta (0.5-4 Hz)

- **Characteristics**

Delta waves are the slowest EEG waves and are typically observed during deep sleep stages (NREM sleep), especially during stages 3 and 4 of non-rapid eye movement (NREM) sleep. They are also present in infants and are associated with brain development.

- **Functions**

Delta waves are believed to be involved in processes such as restorative sleep, memory consolidation, and growth hormone release. They may also play a role in regulating bodily functions such as heart rate and hormone secretion.

b. Theta (4-8 Hz)

- **Characteristics**

Theta waves are prominent during drowsiness, light sleep stages (NREM stage 1 and 2), and REM (rapid eye movement) sleep. They are also observed during states of relaxation, meditation, and creative visualization.

- **Functions**

Theta waves are associated with processes such as dreaming, episodic memory retrieval, emotional processing, and spatial navigation. They are believed to be involved in integrating information from different brain regions and facilitating learning and memory consolidation.

c. Alpha (8-12 Hz)

- **Characteristics**

Alpha waves are most prominent during wakeful relaxation with closed eyes and are attenuated by attention-demanding tasks or sensory stimulation. They are commonly observed over the occipital and parietal regions of the brain.

- **Functions**

Alpha waves are associated with a relaxed wakeful state, passive attention, and inhibition of irrelevant sensory information. They may also play a role in coordinating neural activity within and between brain regions, as well as regulating arousal levels and visual processing.

d. Beta (12-30 Hz)

- **Characteristics**

Beta waves are characteristic of active wakefulness and are typically observed during states of focused attention, concentration, and cognitive processing. They are most prominent over the frontal and central regions of the brain.

- **Functions**

Beta waves are associated with alertness, active engagement in tasks, and motor planning and execution. They are believed to reflect cortical activation and synchronization of neural activity involved in higher cognitive functions, such as problem-solving, decision-making, and motor control.

e. Gamma (>30 Hz)

- **Characteristics**

Gamma waves are the fastest EEG waves and are typically observed during cognitive tasks, perception, and consciousness. They are often localized to cortical areas involved in sensory processing, memory, and attention.

- **Functions**

Gamma waves are associated with processes such as sensory integration, feature binding, attentional selection, and conscious awareness. They are believed to reflect the synchronization of neural ensembles involved in complex cognitive functions and the coordination of information processing across distributed brain networks.

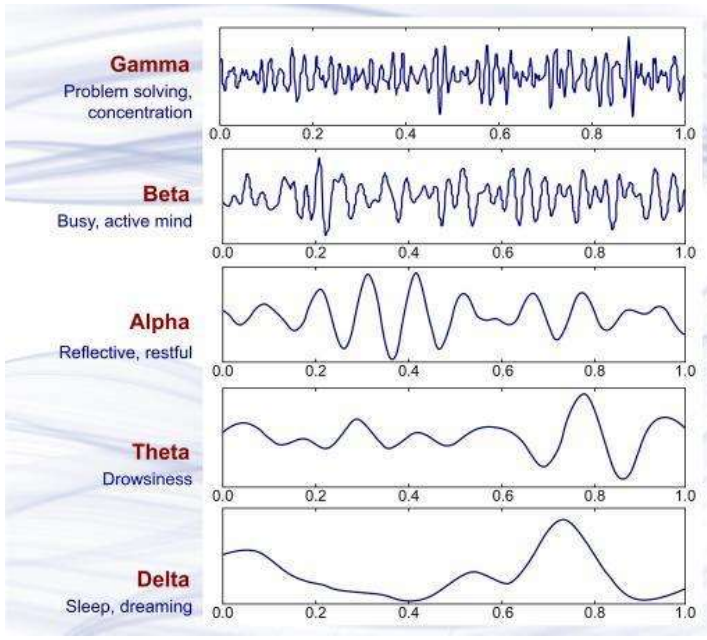


Figure 1(h). Different brain waves belonging to (a)Gamma, (b)Beta, (c)Alpha, (d)Theta, (e) Delta

6) Noises of EEG device

All the changes that occur in EEG recording are not directly affected by the electrical potential of the brain, called the artifacts. These artifacts are divided into sources like biological and artificial artifacts.

Below shows some of the most common artifacts of the EEG.

I. Biological artifacts

Biological artifacts are common occurrences in EEG recordings and can distort the accurate measurement of brain activity. Here's a detailed paraphrase of the types of biological artifacts and their mitigation techniques:

a) Lid and Eye Artifacts

- These artifacts stem from external noises surrounding the brain and are often caused by blinking.
- To minimize lid and eye artifacts, subjects may be instructed to refrain from blinking during EEG tests, although this can be challenging and tiring.
- Electrooculogram (EOG) measurements can be used to specifically detect eye movements by placing electrodes near or on the eyebrows. Samples with potential changes in these electrodes can then be excluded from analysis.

b) Pulse Artifacts

- Pulse artifacts arise when an electrode is positioned over a pulsating blood vessel, resulting in noise within a single electrode.
- These artifacts typically exhibit a uniform pattern, making them distinguishable from EEG signals in recordings.
- Pulse artifacts can be caused by dry or defective electrodes and may require electrode maintenance or replacement.

c) ECG Artifacts

- ECG artifacts are interference caused by the heartbeat and can significantly affect EEG measurements, particularly at reference points.
- These artifacts are challenging to avoid and can be up to ten times stronger in magnitude than EEG brain waves.
- Digital signal processing techniques, such as digital removal, can help mitigate ECG artifacts by isolating and filtering out the interference from EEG recordings.

d) Muscle Artifacts

- Muscle artifacts result from movements in facial and scalp muscles, often occurring close to the scalp.
- These artifacts are most commonly recorded in the temporal and frontal areas of the scalp.

- Intervention methods for muscle artifacts include the use of low-pass filters to remove high-frequency noise and grand averaging to eliminate random fluctuations in time and amplitude.

II. Technical Artifacts

Technical artifacts in EEG recordings can arise from issues with electrodes and cables, often stemming from electrode movement or cable damage.

a) Cable and Electrode Artifacts

- These artifacts occur due to faulty electrodes or movements of the reference electrode during EEG recording sessions.
- Unwanted subject movements can lead to positive or negative potential changes, resulting in electrode artifacts. However, these changes do not follow any specific pattern.
- The appearance of electrode and cable artifacts varies depending on the measurement mode employed. For instance, large electrode count systems may not always detect short circuits in electrodes.
- Cable damage is another potential source of artifacts, especially with prolonged use of cables during EEG recordings.

7) Advantages

- **Non-invasive:** EEG does not require surgery or penetration of the skull, making it safe and well-tolerated.

- **Portable:** EEG systems are lightweight and portable, allowing for recordings in various settings, including clinics, research laboratories, and homes.

- **High Temporal Resolution:** EEG provides real-time monitoring of brain activity with millisecond-level temporal resolution, making it suitable for capturing rapid changes in neural dynamics.

8) Challenges

- **Limited Spatial Resolution:** EEG has relatively poor spatial resolution compared to other neuroimaging techniques, making it challenging to localize the precise sources of brain activity.

- **Susceptibility to Artifacts:** EEG recordings can be contaminated by various artifacts, including muscle activity, eye movements, and environmental noise, which may affect data quality and interpretation.

- **Signal Interpretation:** Interpreting EEG signals requires expertise in signal processing and neuroscience, as EEG data can be complex and influenced by multiple factors.

9) Applications

- **Clinical Diagnosis:** EEG is widely used in clinical settings to diagnose and monitor various neurological conditions, including epilepsy, sleep disorders, and brain injuries.

- **Research:** EEG is a valuable tool in neuroscience research for studying brain function, cognitive processes, and neurological disorders.

- **Assistive Communication:** BCIs are used as assistive communication devices for individuals with severe motor disabilities, such as locked-in syndrome or advanced ALS.BCI-based communication systems allow users to communicate with others by translating their brain signals into text, speech, or commands to operate communication devices or computer interfaces.

- **Neurofeedback:** EEG-based neurofeedback involves training individuals to regulate their brain activity for therapeutic purposes, such as managing attention deficits or reducing symptoms of anxiety and depression.

- **Emotion Recognition and Mood Monitoring:** BCIs have the potential to recognize and monitor emotions and mood states based on patterns of brain activity associated with emotional responses. BCI-based emotion recognition systems can have applications in mental health monitoring, affective computing, and personalized interventions for mood disorders and stress management.

Overall, EEG is a versatile and widely used technique for studying brain function and behavior, with applications ranging from clinical diagnosis to neuroscience research and brain-computer interfacing. Ongoing advancements in EEG technology and signal processing techniques continue to enhance its utility and capabilities.

1.4.1.2 Magnetoencephalography (MEG)

Magnetoencephalography (MEG) is a non-invasive neuroimaging technique used to measure the magnetic fields produced by neuronal activity in the brain. It provides real-time information about the brain's dynamic activity with high temporal resolution (on the order of milliseconds) and good spatial resolution.

MEG works based on the principle of the magnetoencephalogram, which is the magnetic field generated by electrical currents flowing within the neurons of the brain. When neurons fire, they produce tiny electrical currents. These currents generate small magnetic fields that can be detected outside the head by MEG sensors. By analyzing these magnetic fields, researchers can infer the underlying neural activity.

MEG is often used in cognitive neuroscience research to study various aspects of brain function, including sensory perception, motor control, language processing, memory, and

consciousness. It's also used clinically to localize epileptic foci before surgery and to map brain function before neurosurgical procedures.

One of the advantages of MEG is its ability to provide direct measurements of neuronal activity with high temporal resolution, allowing researchers to track the timing of neural processes in real time. However, MEG has limitations in spatial resolution compared to other imaging techniques like fMRI (functional magnetic resonance imaging), which can provide more detailed information about brain anatomy.



Figure 1(i). An image of a MEG scanner with patient

1.4.1.3 FMRI (Functional Magnetic Resonance Imaging)

Functional Magnetic Resonance Imaging, or fMRI [18], is a powerful neuroimaging technique used to measure brain activity by detecting changes in blood flow. It's based on the principle that when a particular brain area is more active, it requires more oxygen, and thus there is an increase in blood flow to that region.

Here's a basic rundown of how fMRI works:

- **Blood Oxygenation Level Dependent (BOLD) Contrast**: The most common method used in fMRI is based on BOLD contrast. This relies on the fact that deoxygenated

hemoglobin in blood is magnetic, while oxygenated hemoglobin is not. When neurons become active and require more oxygen, blood flow increases to that region, resulting in changes in the ratio of oxygenated to deoxygenated hemoglobin.

- **Magnetic Field and Radio Waves:** The fMRI scanner uses powerful magnets and radio waves to create detailed images of the brain. When a person undergoes an fMRI scan, they lie inside a large, tube-like machine with a strong magnetic field.

- **Task or Resting State:** During an fMRI scan, the participant may be asked to perform specific tasks (such as looking at images, solving problems, or responding to stimuli) or may simply rest with their eyes closed. The brain activity associated with these tasks or the resting state is then measured.

- **Data Analysis:** The data collected during the fMRI scan is analyzed to identify patterns of brain activity associated with the task or resting state. This typically involves statistical techniques to identify regions of the brain that show significant changes in activity.

- **Spatial and Temporal Resolution:** fMRI has excellent spatial resolution, meaning it can pinpoint the location of brain activity with high precision. However, its temporal resolution (the ability to measure changes in brain activity over time) is relatively poor compared to other techniques like EEG (Electroencephalography).

FMRI is widely used in neuroscience research to study various aspects of brain function, including perception, cognition, emotion, and motor control. It has also found applications in clinical settings, such as the diagnosis and monitoring of neurological disorders like Alzheimer's disease, epilepsy, and schizophrenia. Additionally, fMRI is increasingly being used in fields like psychology, economics, and marketing to understand decision-making processes and consumer behavior.



Figure 1(j). A functional magnetic resonance imaging device.

1.4.1.4 FNIRS (Functional Near Infrared Spectroscopy)

The functioning of the fNIRS device relies on the absorption of light by hemoglobin to assess factors like blood volume, flow, and oxygenation as light traverses the headgear. Its core function involves gauging brain oxygen levels through near-infrared spectroscopy, which detects both oxygenated and deoxygenated blood levels, aiding in the identification of activated brain areas. With its compatibility with multiple imaging methods, fNIRS stands out for its safety and effectiveness as a control mechanism in brain-computer interfaces.

FNIRS offers several advantages over fMRI, including portability, lower cost, and the ability to measure brain activity in naturalistic settings (e.g., during social interactions or mobile tasks). However, it also has limitations, such as limited spatial resolution and sensitivity to scalp and skull artifacts.

FNIRS [5] is used in a wide range of research fields, including neuroscience, psychology, cognitive science, and clinical research. It has applications in studying brain development, cognitive processes, neurological disorders, human-computer interaction, and brain-computer interfaces.



Figure 1(k). A whole brain fNIRS device

1.4.1.5 PET (Positron Emission Tomography)

PET, or Positron Emission Tomography, is a nuclear imaging technique used to create three-dimensional images of the body's metabolism by administering a small amount of radioactive material. It's primarily employed in medical imaging to monitor changes in physiological activities like metabolism, blood flow, regional chemical composition, and absorption. PET works by injecting a short-lived isotope, often FDG (fluoro-2-deoxyglucose), into the body and allowing it to accumulate in the target tissue. Once the tracer is distributed, the subject is positioned in the imaging scanner, and the concentration of the tissue is recorded over time. As the isotopes decay, they emit positrons, which interact with electrons in the body to produce pairs of photons. These photons are detected by the PET scanner and processed by a computer to generate images of the organs or tissues. PET is particularly valuable in neuroimaging as it can visualize oxygen flow in the brain and assess the metabolism of various brain tissues, providing insights into brain function. Additionally, PET finds applications in clinical oncology, musculoskeletal imaging, cardiology, pharmacology, and neuropsychology.



Figure 1(l). Positron Emission Tomography scanner device

1.4.2 Invasive BCIs

Invasive methods of Brain-Computer Interface (BCI) involve directly interfacing with the brain's neural tissue through surgical procedures. These methods typically require implanting electrodes or other devices into specific regions of the brain to record neural activity or stimulate neurons. The devices under the invasive technique are follows:

1.4.2.1 IAT (Intracortical Acquisition Technique)

The Intracortical Acquisition Technique (IAT) is widely recognized as a prominent invasive BCI method. It entails implanting electrodes beneath the surface of the brain's cortex to directly capture action signals generated by individual neurons. This can be accomplished using arrays of electrodes or single electrodes. Ensuring the stability of these electrode arrays over extended periods is crucial, as the tips of the electrodes must remain in close proximity to the source of the neural signals to maintain high spatial resolution. Consequently, IAT is often favored for its exceptional ability to precisely localize neural activity.

However, despite its advantages, intracortical acquisition experiences long-term fluctuations that can impact its reliability. These fluctuations may arise from factors such as an increase in tissue

resistance around the electrodes or the death of neurons over time. Additionally, the presence of significant noise can affect the accuracy of the recorded signals, particularly if the system is subjected to stimuli that activate the affected limb. Therefore, while IAT offers unparalleled spatial resolution and precise source localization, researchers and clinicians must carefully consider and address these potential challenges to ensure the effectiveness and reliability of the technique in BCI applications.

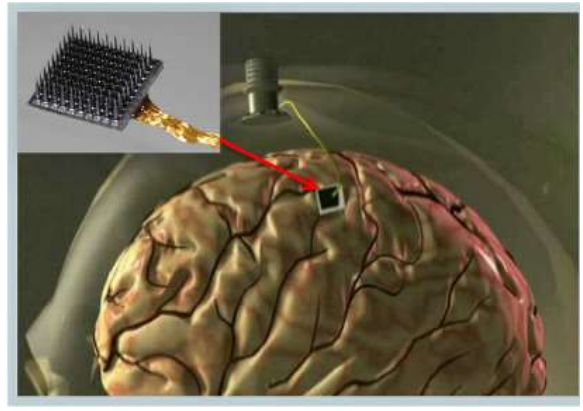


Figure 1(m). Image shows an Intracortical Acquisition Technique, used to measure the activity of single neuron using the micro-electrodes that placed directly into the cortex.

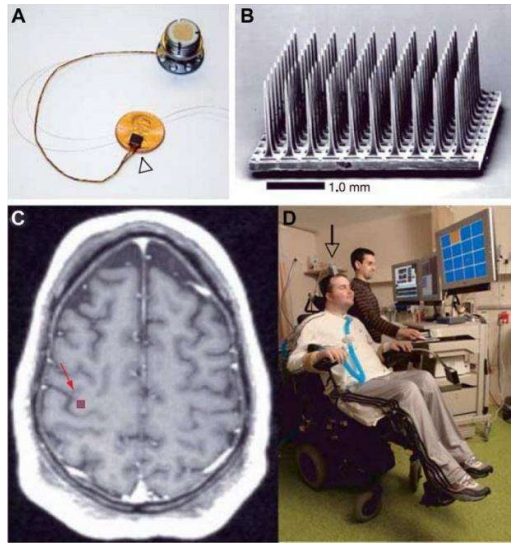


Figure-1(n): The microelectrode array implanted into the cortex

1.4.2.2 ECoG (Electrocorticography)

Electrocorticography (ECoG) [3] is another invasive BCI technique that involves placing electrode grids directly on the surface of the brain, beneath the skull but outside the dura mater (the outermost membrane surrounding the brain). This method allows for the recording of neural signals from the cortical surface with high spatial and temporal resolution. ECoG has been used in various applications, including epilepsy monitoring, brain mapping for surgical planning, and research into brain-computer interfaces.

The ECoG electrodes are typically arranged in a grid pattern and are placed over specific brain regions of interest. These electrodes can detect neural activity associated with sensory, motor, and cognitive processes. ECoG offers several advantages over other invasive BCI techniques, including relatively lower risk compared to intracortical electrodes, as they do not penetrate the brain tissue directly. Additionally, ECoG signals tend to have higher signal-to-noise ratio and stability over time compared to intracortical recordings.

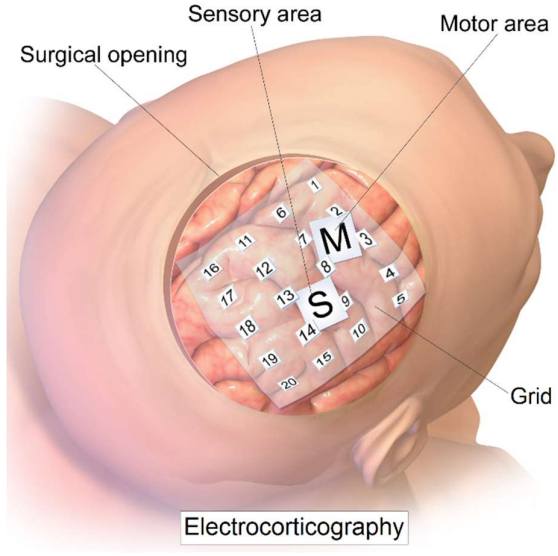


Figure 1(o). The grid placement in ECoG

Invasive BCI methods offer several advantages, including high signal quality, precise spatial localization, and long-term stability. However, they also carry risks associated with surgical procedures, such as infection, tissue damage, and the need for ongoing medical monitoring. Additionally, invasive BCIs require careful consideration of ethical and privacy concerns, particularly regarding patient consent, data security, and potential risks of invasive brain interventions.

1.5 Challenges and Future Directions of BCIs

- **Signal Quality:** Improve signal processing to enhance signal quality and reliability.
- **Information Bandwidth:** Develop hybrid BCIs and closed-loop systems to increase communication bandwidth.
- **User Training:** Design user-friendly interfaces and personalized training protocols for better user adaptation.
- **Long-Term Reliability:** Explore novel electrode materials and implantation techniques for sustained use.
- **Clinical Translation:** Conduct large-scale clinical trials and develop accessible BCI devices for diverse populations.
- **Ethical and Privacy Concerns:** Establish ethical guidelines and privacy-enhancing technologies to protect users' neural data.
- **Neuroethical Considerations:** Address ethical implications of BCI enhancement and promote equitable access and responsible innovation.

1.6 Literature Review

The study of brain-computer interfaces (BCIs) has evolved significantly over the past few decades, driven by the need for more effective communication and control methods, particularly for individuals with severe motor impairments. Leuthardt et al. (2004) [1] demonstrated the feasibility of using electrocorticographic (ECoG) signals for BCI applications in humans, highlighting the potential for high-resolution neural control mechanisms . This foundational work paved the way for further exploration into non-invasive techniques for cognitive load measurement and BCI development.

Wolpaw et al. (2002) [2] provided a comprehensive overview of BCI technologies, emphasizing the importance of developing systems that can reliably interpret user intentions from neural signals . Their work underscored the challenges associated with signal acquisition, processing, and the integration of BCIs into practical applications. This laid the groundwork for subsequent research focusing on improving the accuracy and usability of BCI systems.

In the context of cognitive load measurement, Antonenko et al. (2010) [3] explored the use of electroencephalography (EEG) to assess cognitive load during learning tasks . Their findings indicated that EEG could provide valuable insights into the cognitive processes involved in learning, but also highlighted limitations related to signal noise and the complexity of data interpretation. Complementing this, Martin (2018) [5] discussed various metrics for measuring cognitive load in technology-enhanced learning environments, emphasizing the need for reliable and valid assessment tools .

Functional Near-Infrared Spectroscopy (fNIRS) has emerged as a promising alternative for cognitive load measurement, offering a non-invasive and relatively user-friendly method for monitoring brain activity. Coyle et al. (2007) [10] demonstrated the potential of a simplified fNIRS system for BCI applications, suggesting that fNIRS could be used to measure cognitive load in real-time . Their study provided initial evidence that fNIRS could complement existing techniques like EEG and expand the applicability of BCIs in various contexts.

Further exploring the intersection of neuroimaging and cognitive load, Bhattacharyya et al. (2017) [14] reviewed various brain imaging techniques, including fNIRS, for BCI applications . They highlighted the strengths and weaknesses of each method, pointing out that while fNIRS offers benefits in terms of portability and ease of use, it also faces challenges related to spatial resolution and susceptibility to motion artifacts.

In addition to these non-invasive techniques, other studies have investigated more invasive methods and their applications. For instance, Mesgarani and Chang (2012) [12] explored selective cortical representation using electrocorticography (ECoG), demonstrating its utility in multi-talker speech perception scenarios . Their work illustrated the potential of high-resolution neural recordings to enhance the performance of BCIs in complex auditory environments.

Complementary studies in the field of cognitive neuroscience have utilized other imaging modalities to understand brain function and cognitive processes. For example, Casey et al. (2002) [18] reviewed the principles and applications of functional magnetic resonance imaging (fMRI) in developmental science, providing insights into the neural underpinnings of cognitive development and function . Similarly, Buzsaki (2006) [13] discussed the significance of neural rhythms and oscillations in brain function, highlighting their relevance for understanding cognitive load and developing more effective BCIs .

Collectively, these studies underscore the multifaceted nature of cognitive load measurement and the potential of fNIRS as a valuable tool in this domain. While traditional methods like EEG and fMRI have provided significant insights, the advent of fNIRS offers new opportunities for non-invasive, real-time monitoring of cognitive load. However, challenges related to signal quality, motion artifacts, and data interpretation remain, necessitating further research to enhance the reliability and applicability of fNIRS in cognitive load measurement and BCI applications. By addressing these limitations and integrating insights from various imaging modalities, future research can advance the development of more effective and user-friendly cognitive load measurement tools, ultimately enhancing the efficacy of BCIs and their applications in education, human-computer interaction, and beyond.

1.7 Scope of Work

The measurement of cognitive load has been a significant area of research, particularly due to its implications in instructional design and learning. Various methodologies have been employed to assess cognitive load, with traditional approaches such as self-report questionnaires, dual-task performance, and physiological measures like heart rate variability and pupil dilation being widely used. However, these methods often face limitations in terms of accuracy, reliability, and real-time applicability.

Functional Near-Infrared Spectroscopy (fNIRS) has emerged as a promising alternative for cognitive load measurement due to its ability to non-invasively monitor brain activity in real-time. fNIRS measures changes in blood oxygenation levels, providing insights into neural activity associated with cognitive processes. Studies have shown that fNIRS can effectively capture cognitive load variations during various tasks, offering a more direct and objective measure compared to traditional methods.

Previous research has demonstrated the feasibility of using fNIRS in cognitive load measurement across different domains. For example, studies have explored its application in educational settings to assess the cognitive load of students during learning activities. In the field of human-computer interaction, fNIRS has been utilized to evaluate the mental workload of users interacting with complex systems. Additionally, research has highlighted the potential of fNIRS in combination with other physiological measures, such as EEG, to enhance the accuracy of cognitive load assessment.

Despite the advancements, several limitations remain in the current research on cognitive load measurement using fNIRS. One significant challenge is the presence of motion artifacts, which can distort the fNIRS signal and affect the accuracy of cognitive load estimates. Another limitation is the variability in experimental designs and data analysis techniques, making it difficult to compare results across studies and establish standardized protocols.

Moreover, while fNIRS provides valuable information on neural activity, it does not offer comprehensive insights into other physiological and behavioral aspects of cognitive load. Integrating

fNIRS with other measures, such as eye tracking and behavioral performance, could provide a more holistic understanding of cognitive load dynamics. Additionally, the development of robust algorithms and machine learning models to process fNIRS data and accurately estimate cognitive load levels remains an ongoing area of research.

- **Aim**

This work aims to propose standardized protocols for cognitive load measurement using fNIRS, facilitating broader adoption and comparability of results across studies. The findings from this study are expected to have significant implications for instructional design, human-computer interaction, and other fields where understanding and managing cognitive load is critical.

- **Objective**

The objectives of this research are to review and synthesize existing literature on cognitive load measurement using fNIRS, design and conduct experiments to measure cognitive load during specific tasks with enhanced signal quality, develop and evaluate machine learning algorithms for accurate cognitive load estimation, integrate fNIRS with other physiological and behavioral measures to provide a comprehensive assessment, and propose standardized protocols for cognitive load measurement to promote consistency and comparability across studies. By achieving these objectives, this study aims to contribute to the development of more reliable and effective methods for cognitive load measurement, ultimately enhancing instructional design practices and improving user experience in various domains.

1.8 References

1. E.C.Leuthardt, G.Schalk, J.Wolpaw, J.G.Ojemann,"A brain–computer interface using electrocorticographic signals in humans", *Journal of Neural Engineering*, Volume 1, Issue 2, pp 63-71, June,2004.

2. Wolpaw, J. R., Birbaumer, N., McFarland, D. J., Pfurtscheller, G., & Vaughan, T. M. (2002). "Brain–computer interfaces for communication and control." *IEEE Transactions on Rehabilitation Engineering*, 10(2), 164-173.

3. P. Antonenko, F. Paas, R. Grabner, and T. Van Gog, "Using electroencephalography to measure cognitive load," *Educational Psychology Review*, vol. 22, pp. 425-438, 2010.

4. Y. Ding, F. Guo, M. Hu, and Y. Cao, "Using event related potentials to investigate visual aesthetic perception of product appearance," *Human Factors and Ergonomics in Manufacturing & Service Industries*, vol. 27, no. 5, pp 223-232, Jun 2018.

5. S. Martin, "Measuring cognitive load and cognition: metrics for technology-enhanced learning," *in Technology-Enhanced and Collaborative Learning*, pp. 77-106, Routledge, 2018.

6. M. Grunwald, T. Weiss, W. Krause, L. Beyer, R. Rost, I. Gutberlet, H.J. Gertz , "Theta power in the EEG of humans during ongoing processing in a haptic object recognition task," *Brain Res Cogn Brain Res*, vol. 11, no. 1, pp. 33-37, 2001.

7. A. Khasnobish, A.Jati, G.Singh, A.Konar, D.N.Tibarewala, "Object-shape recognition from tactile images using a feed-forward neural network", *in The 2012 International Joint Conference on Neural Networks (IJCNN)*, Brisbane, QLD, pp. 1-8, 2012.

8. J. Martinovic, R. Lawson, M. Craddock, "Time course of information processing in visual and haptic object classification," *Frontiers in Human Neuroscience*, vol. 6, pp. 1-11, 2012.

9. S. Luo, W. Mou, K. Althoefer, H. Liu, "iclap: shape recognition by combining proprioception and touch sensing", *Autonomous Robots*, pp. 1-12, 2018.

10. S.Coyle, T.E.Ward, C.M.Markham," Brain–computer interface using a simplified functional near-infrared spectroscopy system", *Jouranl of Neural Engineering*,Vol.4, Issue 3, pp. 219-26. Oct 2007.

11. P.Zanzonico, "Positron Emission Tomography: A Review of Basic Principles, Scanner Design and Performance, and Current Systems", *Seminars in Nuclear Medicine*, Vol 34, Issue 2, pp 87-111, May, 2004.

12. N.Mesgarani, E.F.Chang, " Selective cortical representation of attended speaker in multi-talker speech perception", *Nature*, Vol.485, Issue 7397, pp. 233–236, April,2012.

13. G.Buzsaki, "Rhythms of the Brain. Oxford University Press", 2006.

14. S. Bhattacharyya, A.Khasnobish, P.Ghosh, A.Mazumder, and D.Tibarewala,"A review on brain imaging techniques for bci applications, "*in Medical Imaging: Concepts, Methodologies*, Tools, and Applications, pp.300-330,IGI Global, 2017.

15. E. Niedermeyerand F.L.daSilva , "Electroencephalography: basic principles, clinical applications, and related fields". *Lippincott Williams & Wilkins*, Vol.67, Issue 11,Dec,2006.

16. P.T. Lin, K.Sharma, T.Holroyd, H.Battapady, D.Fei and O.Bai,"A High Performance MEG Based BCI Using Single Trial Detection of Human Movement Intention, Functional Brain Mapping and the Endeavor to Understand the Working Brain", *Intech Open, Editors: Francesco Signorelli and Domenico Chirchiglia*, pp.19, Jan, 2013.

17. T.N. Lal, M. Tangermann, N.J.Hill,H.Preissl." A Brain Computer Interface with Online Feedback based on Magnetoencephalography", *International Conference on Machine Learning, Bonn,* Germany, 2005.

18. B.J.Casey, M.Davidson and B.Rosen , "Functional magnetic resonance imaging: basic principles of and application to developmental science", *Developmental Science* , Vol.5, Issue 3, pp 301-309, Aug,2002.

CHAPTER 2

Functional Near-Infrared Spectroscopy

2.1 Introduction

fNIRS, or functional near-infrared spectroscopy, is a method that uses near-infrared light to measure changes in haemoglobin concentration in the brain. This technique is safe, non-invasive, and can be used alongside other imaging methods. By analyzing how light is absorbed and scattered in the brain tissue, fNIRS can estimate haemoglobin concentration, blood volume, and oxygenation levels. This is achieved by measuring the attenuation of light at different wavelengths, taking advantage of the fact that haemoglobin and deoxygenated haemoglobin absorb near-infrared light differently. fNIRS primarily focuses on absorption, using the modified Beer-Lambert law to calculate relative changes in haemoglobin concentration based on the total path length of photons. Placing the light emitter and detector on the same side of the skull allows for the measurement of back-scattered light, making the technique most sensitive to changes in hemodynamics closest to the scalp.

2.2 History

In 1977, Jöbsis [1] discovered that near-infrared light could be used to non-invasively measure tissue oxygen saturation in the brain. This led to the rapid development of NIRS systems, with reflectance-mode techniques replacing transillumination due to better results in adults. The first studies on cerebral oxygenation were conducted in 1985 by M. Ferrari, [2] and in 1989, Hamamatsu [3] developed the first commercial NIRS system, the NIR-1000 cerebral oxygenation monitor, following collaboration with David Delpy at University College London. NIRS methods were primarily used for cerebral oximetry in the 1990s, with further advancements and feasibility demonstrated in adult humans in 1993 by Chance et al., [4] Hoshi & Tamura [5] , Kato et al., and Villringer et al [6] . This work was expanded upon by researchers like Randall Barbour, Britton Chance, Arno Villringer, M. Cope, D. T. Delpy, and Enrico Gratton. Currently, wearable fNIRS devices are being developed.

In Japan, in the mid-1980s, researchers at Hitachi Ltd's central research laboratory began developing a NIRS-based brain monitoring system using short pulses of near-infrared light. This work culminated in the introduction of the Hitachi ETG-100, the company's first fNIRS device based on Frequency Domain, in 2001. Harumi Oishi [7] , a PhD candidate at Nagoya University, published her doctoral dissertation in 2003 on "language learners' cortical activation patterns measured by ETG-

100," providing new insights into fNIRS applications. Hitachi has continued to advance the ETG series since then.

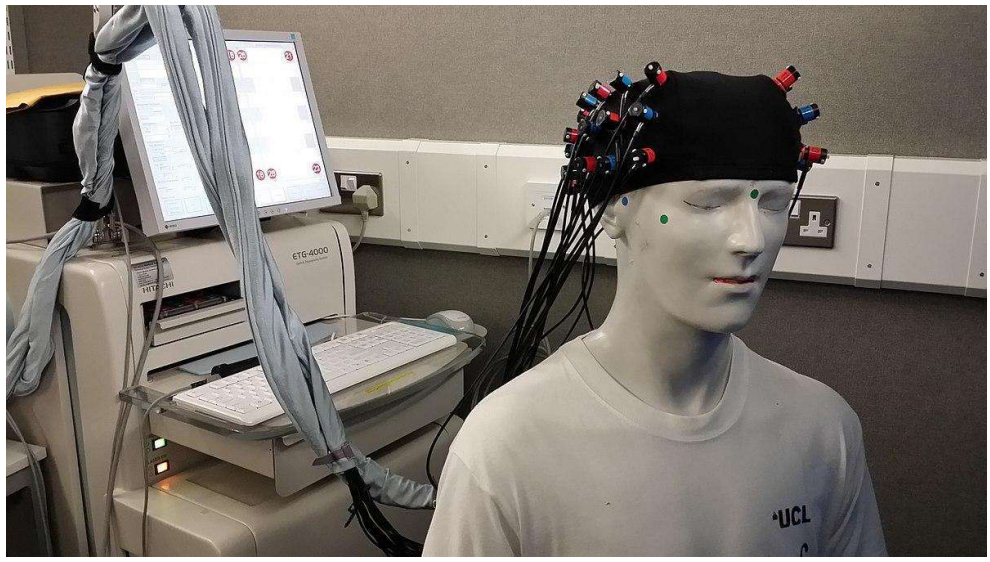


Figure 2(a): Hitachi ETG-4000

2.3 Physical Principle of fNIRS

The fNIRS device measures brain activity by emitting near-infrared (NIR) light onto the scalp. Before reaching the brain, the light passes through various layers such as the scalp, skin, skull, and cerebrospinal fluid, each with unique optical properties [8]. Human tissue is complex, being both anisotropic (having different optical properties in different directions) and inhomogeneous (having varying properties throughout). Despite this complexity, the interaction between NIR light and tissue is primarily affected by absorption and scattering.

There are six main ways [10] in which infrared light can interact with the brain: direct transmission, diffuse transmission, specular reflection, diffuse reflection, scattering, and absorption. fNIRS mainly focuses on absorption, which is the process by which the energy of a photon is converted into internal energy of the medium it is traveling through. Human tissue contains various substances like lipids, water, hemoglobin, melanin, and cytochrome-c-oxides, each with different absorption properties at different wavelengths. Since the human body is mostly water and water have minimal absorption in the NIR range, NIR light can transmit through tissue.

Haemoglobin is a significant absorbing chromophore in the NIR range, with different absorption properties for its oxygenated (oxyhemoglobin) and deoxygenated (deoxyhemoglobin) forms [12] . This difference in absorption is used to measure changes in hemoglobin concentration, as oxyhemoglobin appears redder while deoxyhemoglobin appears more purple. Increased brain activity during tasks leads to increased regional cerebral blood flow, resulting in an increase in oxyhemoglobin and a decrease in deoxyhemoglobin concentrations.



Figure 2 (b): Functional near-infrared spectroscopy

The difference in absorption spectra between deoxyhemoglobin and oxyhemoglobin allows for the calculation of relative changes in hemoglobin concentration using the Modified Beer-Lambert law (mBLL), which considers the total path length of photons. Typically, the light emitter and detector are placed on the same side of the head to measure back-scattered light, optimizing sensitivity to changes in hemodynamics near the scalp.

- **Modified Beer-Lambert Law**

The changes in the intensity of light can be related to the changes in relative concentration of hemoglobin through the Modified beer-Lambert Law (mBLL). This law deals with the hemoglobin concentration

Formula:

$$\mu_a(1) = \ln(10)\varepsilon_{ox}(\lambda_1)C_{ox} + \ln(10)\varepsilon_{de}(\lambda_1)C_{de} \quad (2.1)$$

$$\mu_a(2) = \ln(10)\varepsilon_{ox}(\lambda_2)C_{ox} + \ln(10)\varepsilon_{de}(\lambda_2)C_{de} \quad (2.2)$$

The total concentration of HbT(C_{HbT}) is

$$C_{HbT} = C_{ox} + C_{de} \quad (2.3)$$

In these equations ε_{ox} and ε_{de} are the molar coefficient at the wavelength (λ_1) and (λ_2). C_{ox} and C_{de} are the concentrations of the HbO and Hb (Wang, Wu, 2012).

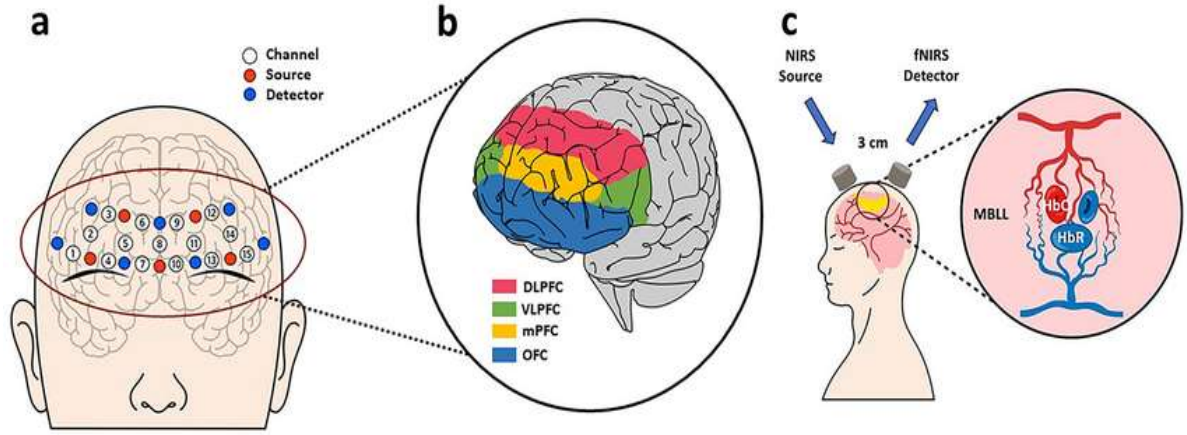


Figure 2(c): Schematic channel location and measurement principle of fNIRS device

2.4 fNIRS Cap

The fNIRS cap [14] and probes are crucial components of any NIRS system setup, as they are in direct contact with both the subject (who is providing data) and the operator (who is collecting the data). The system offers complete flexibility in montage design, allowing for a wide range of configurations. Probes can be placed according to standard layouts like the 10-10 or 10-5 systems, or in arbitrary positions. Measurements can be taken for short-distance (tomography) or long-distance (topography) purposes. The setup can include integrated EEG or other modalities (such as tDCS, accelerometers, etc.) alongside NIRS, or use NIRS alone with a dedicated NIRs cap. It is also possible to use the fNIRS device concurrently with other imaging modalities like fMRI, TMS, EKG, MEG, pulse oximetry, EMG, EOG, eye-tracking, motion-tracking, etc.

For the data acquisition purpose, the NIR scout system by NlRx medical technologies, USA has been used for data acquisition which consists of 8 optode pairs(i.e., 8 sources and8 detectors) that yields two distinct wavelength which is ranging from approximately from 700 and 900 nm. The absorption of HbO and Hb are the strong absorber in NIR light. The absorption spectra for HbO and Hb is shown in Figure 2(d).

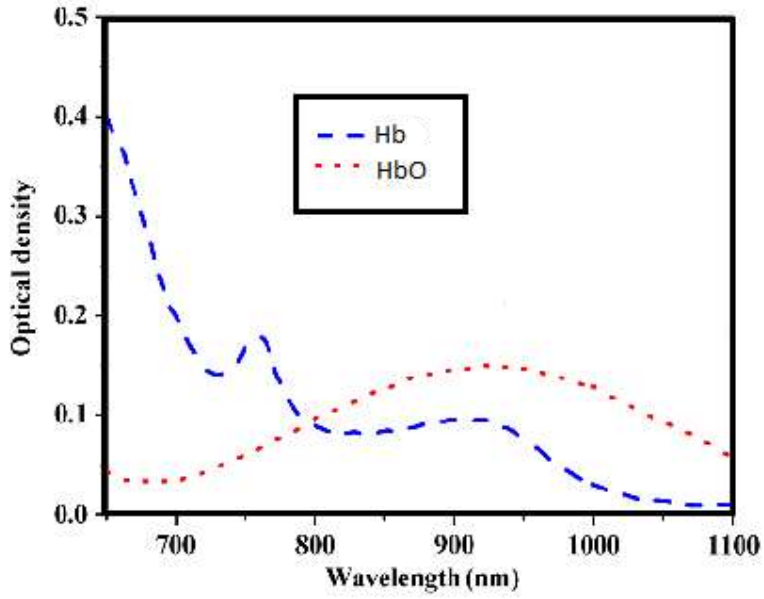


Figure 2(d): Absorption spectra for Hb and HbO

The optodes in the fNIRS system are placed according to the standard 10-10 and 10-5 placement systems [16] , with a separation distance of 3 cm between each sensor. The system has a sampling rate of 7.81 Hz and consists of 8 sensors and 8 detectors, resulting in 64 channels in total. The selection of channels is based on the experiment's topolayout combination, either using the nearest neighbor source-detector combination or as per the montage design requirements.

The NIRx system allows for flexible placement of probes, enabling data collection from various areas of the head. Some examples of areas that can be measured with a NIRx fNIRS system include the pre-frontal cortex, dorsolateral pre-frontal cortex, supplementary motor area, premotor cortex, primary motor cortex, somatosensory cortex, posterior parietal cortex, and primary auditory cortex. Proper spacing between probes is essential for optimal light penetration and signal quality, regardless of whether the probes are placed according to the standard layout or in arbitrary positions. It is crucial for the probes and caps to be fitted properly and comfortably, as discomfort can cause the subject to move, leading to poor data quality.

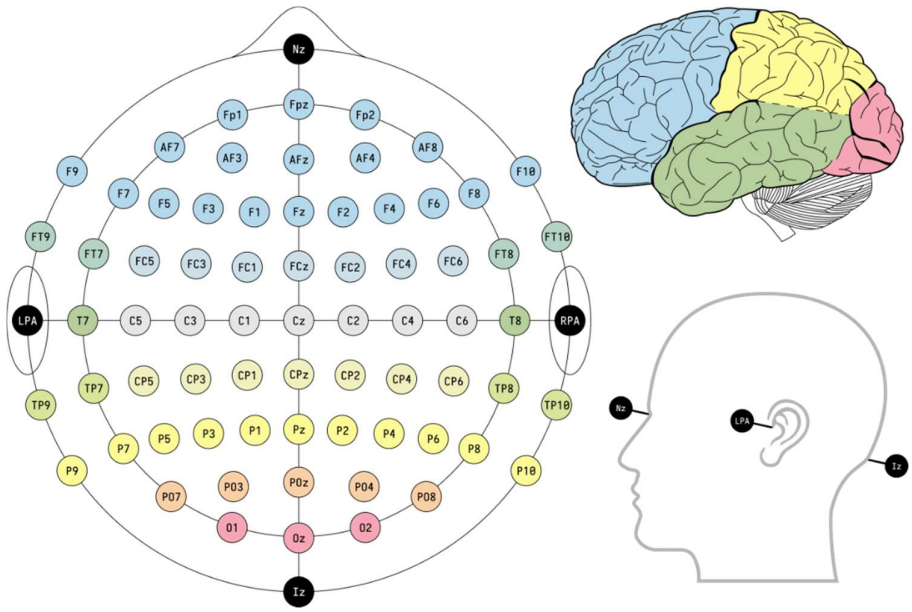


Figure 2(e): The international 10–10 electrode placement system

The international 10-10 electrode placement system is a standardized method for positioning EEG electrodes on the scalp. It is used to ensure consistent and reproducible electrode placement across different subjects and studies. The "10-10" refers to the fact that the system divides the scalp into regions where electrodes are placed at intervals of 10% and 20% of the total front-back and right-left distances, respectively. This results in a grid-like pattern of electrode positions, with each position labelled based on its location relative to anatomical landmarks on the head.

2.5 Spectroscopic Techniques

Functional near-infrared spectroscopy (fNIRS) is a non-invasive neuroimaging technique that measures brain activity by detecting changes in blood oxygenation and volume in the cerebral cortex. There are three main modalities of fNIRS: Continuous wave (CW), Frequency domain (FD), and Time Domain (TD).

1. **Continuous Wave (CW)**

CW-fNIRS systems use continuous light sources with constant intensity and frequency. While simple and cost-effective, CW-fNIRS does not provide absolute measurements of hemoglobin concentration. Instead, it measures relative changes in hemoglobin concentration by comparing the intensity of the light emitted from the source with the intensity of the light detected after passing through the tissue. CW-fNIRS systems are widely used due to their simplicity and cost-effectiveness, making them suitable for various applications, including neonatal care, patient monitoring, and research.

2. **Frequency Domain (FD)**

FD-fNIRS systems use light sources that emit modulated light at multiple frequencies. By measuring the phase shift and attenuation of the modulated light as it passes through the tissue, FD-fNIRS can provide more detailed information about hemoglobin concentration changes compared to CW-fNIRS. FD-fNIRS is capable of measuring absolute hemoglobin concentrations and is less affected by variations in skin colour, making it a valuable tool for studying brain function in diverse

populations. However, FD-fNIRS systems are more complex and expensive than CW systems, limiting their use in some settings.

3. Time Domain (TD)

TD-fNIRS systems use short pulses of light to measure the time it takes for the light to travel through the tissue and be detected. By analyzing the time-of-flight of the light pulses, TD-fNIRS can determine the depth of the tissue being measured, providing high depth sensitivity. TD-fNIRS is also capable of measuring absolute hemoglobin concentrations and is less affected by scattering in the tissue compared to CW and FD systems. However, TD-fNIRS systems are the most technically complex and expensive of the three modalities, making them less accessible for some applications.

In summary, each modality of fNIRS has its strengths and limitations, and the choice of modality depends on the specific requirements of the study or application. CW-fNIRS is simple and cost-effective but provides relative measurements of hemoglobin concentration. FD-fNIRS provides more detailed information and can measure absolute hemoglobin concentrations but is more complex and expensive. TD-fNIRS offers high depth sensitivity and accurate measurements of hemoglobin concentration but is the most technically challenging and expensive option.

2.6 Steps Involved in fNIRS Device

Functional near-infrared spectroscopy (fNIRS) devices are used to measure changes in cerebral blood flow and oxygenation levels in the brain. The following are the general steps involved in using an fNIRS device:

a) **Setup:** Place the fNIRS cap on the subject's head. The cap contains optodes (light emitters and detectors) that are positioned according to a specific layout (e.g., 10-20 system) to cover the desired brain regions.

b) **Optode Placement:** Ensure that the optodes are placed correctly on the scalp, with the emitters and detectors aligned properly. The distance between emitters and detectors determines the depth of measurement.

c) **Calibration:** Before starting the measurement, calibrate the fNIRS device. This involves checking the light intensity, wavelength, and any other relevant parameters to ensure accurate data collection.

d) **Data Acquisition:** The goal of this step is to collect appropriate brain signals, often focusing on the primary prefrontal cortex and primary motor cortex.

- Motor cortex activities are commonly used for fNIRS Brain-Computer Interface (BCI) applications.

- Motor execution involves actual movements or muscular tension, such as finger tapping, hand tapping, etc.

- Motor imagery is a mental rehearsal of a specific action without physical movement, like imagining squeezing a ball or tapping feet.

- Prefrontal cortex activities, suitable for patients with Locked-In Syndrome (LIS), involve simple imagination tasks like mental arithmetic, mental counting, and music imagery.

i. Motor Cortex Activities

- The primary motor cortex is often chosen for fNIRS BCI applications.

- Motor execution tasks include finger tapping, hand tapping, etc., activated by physical movement and affected by proprioceptive feedback.

- Motor imagery tasks involve imagining actions without physical movement and are used in BCI devices for communication with physically disabled individuals.

ii. Prefrontal Cortex Activities

- Prefrontal cortex activities are suitable for LIS patients and do not require motor tasks.
- Mental arithmetic involves performing calculations mentally without external aids.
- Mental counting requires patients to count backwards from a three-digit number, activating the prefrontal area.
- Music imagery involves organizing and analyzing music in the brain without external auditory stimuli.

e) Pre-processing

Pre-processing plays a crucial role in refining the signals acquired from fNIRS devices, as they often contain various types of unwanted noise, including instrumental noise, experimental errors, and physiological noise. The primary objective of pre-processing is to filter out these undesirable components from the fNIRS data, allowing for the accurate extraction of changes in hemoglobin concentrations (HbO and Hb) using the Modified Beer-Lambert Law.

Instrumental noise originates from external factors or hardware-related issues and typically exhibits frequencies within the range of 3-5 Hz. This noise can be effectively attenuated using a low-pass filter, which removes high-frequency components from the signal. Experimental errors, caused by movement of the optodes due to head motion, can introduce abrupt fluctuations in light intensity, resulting in spike-like artifacts in the data. Various filtering techniques, such as band-pass filtering, adaptive filtering, independent component analysis (ICA), and principal component analysis (PCA), can be applied to mitigate these artifacts and enhance signal clarity.

Physiological noise, arising from natural bodily processes such as heartbeat (1.5 Hz) and respiration (0.20.5 Hz), poses another challenge in fNIRS signal processing. Additionally, fluctuations in blood pressure can contribute to this type of noise. To address physiological noise, similar filtering approaches can be employed to isolate and remove unwanted components from the signal.

Effective pre-processing of fNIRS data is essential for improving system performance by minimizing noise interference and ensuring the reliability of subsequent analyses. By refining the signal quality, pre-processing facilitates more accurate interpretation and insights into brain activity patterns.

f) Feature Extraction and Selection

This Step involves the classification of the brain activities based on certain features after the preprocessing process. Most of the features are extracted from the hemodynamic signals. The reason behind the selection is that the HbO, Hb, HbT (total hemoglobin), Central oxygen exchange (COE = HbO - Hb) contribute more opportunities for selection of convenient features. Hemodynamic signal's shapes are mostly clear, after the removal of the noises. The differences in the signal can easily be determined by observing the hemodynamic signals that originate from the different activities. Among all the features of HbO and Hb. Most commonly used features are mean, peak, slope, skewness and kurtosis.

- **Signal Mean**

The signal mean of ΔHbO and ΔHb is computed as the average value within a specified window.

The signal mean of ΔHbO and ΔHb are determined as follows:

$$\mu_w = \frac{1}{N_w}\sum_{k=k_1}^{k_2} \Delta HbX(k) \tag{2.4}$$

The script w denotes the "window", μ denotes the mean value for the window, k_1 and k_2 denote the start time and end time of the window. Number of windows is denoted by N_w . The HbO and Hb data is denoted by ΔHbX. In some fNIRS based BCI device the signal mean is sometimes used as a classification feature.

- **Signal Slope**

In order to determine the signal slope, two methods are used. The first one is the shortest route to determine the signal slope that acquired from the predetermined intervals by obtaining the difference between the start and the end points and that can be apply to estimate the slope. Again, for the pre-determined intervals, second one uses a curve-fittings approach to fit a line to the hemodynamic signal-for the determination of signal slope, the first method is used widely.

- **Signal Peak**

To determine the signal peak of ΔHbO (change in oxyhemoglobin) and ΔHb (change in deoxyhemoglobin) within a specified window, you need to identify the maximum value of the signal within that window. This can be expressed mathematically as follows:

$$Peak_w = max_{k_1 \le k \le k_2} \Delta HbX(k) \tag{2.5}$$

where:

$Peak_w$ is the peak value of the signal (ΔHbO or ΔHbR) within the window, $\Delta HbX(k)$ represents the value of the signal (ΔHbO or ΔHb) at the k-th sample. k_1 and k_2 are the starting and ending indices of the window, respectively. This formula identifies the highest value that the signal (ΔHbO or ΔHb) reaches within the given window.

- **Skewness and Kurtosis**

The skew is calculated as follows:

$$skew_w = E_x[(\Delta HbX_w - \mu_w)^3]/\sigma^3 \tag{2.6}$$

where $skew_w$ is the skewness, σ is the standard deviation of HbX for the given window, and E_x is the expectation of ΔHbX.

The kurtosis is calculated as follows:

$$kurt_w = E_x[(\Delta HbX_w - \mu_w)^4]/\sigma^4 \quad (2.7)$$

where kurt is the kurtosis of ΔHbX and the other terms are defined as above. Both skewness and kurtosis have been reported to work moderately well for fNIRS.

2.7 Noises in fNIRS Device

In fNIRS (functional Near-Infrared Spectroscopy) devices, various types of noise can affect the quality of the acquired signals. These noises can arise from different sources and can interfere with the accurate measurement of brain activity. Some common types of noise in fNIRS devices include:

a) Instrumental Noise

This type of noise originates from the fNIRS device itself or its surroundings. It can be caused by electronic components, ambient light, or electromagnetic interference. Instrumental noise can introduce unwanted fluctuations in the measured signals, affecting the accuracy of the data.

b) Lid and Eye Artifacts

The most common artifacts are the extra-cerebral noises. In order to reduce those types of noises one of the ways is that throughout the duration of the test the subject is asked not to blink. Some subjects find those types of processes tiring and avoid it. In order to measure those types of artifacts, it is better to measure with an Electrooculogram (EOG). The process of EOG can be done by placing the electrodes next or onto the eyebrows and excluding the samples having potential changes in those electrodes as a whole.

c) Motion Artifacts

Motion artifacts are the artifacts generally occurred due to movement mostly close to the scalp area. The main purpose of intervention is due to face and muscle movement and this artifact are

recorded mainly over the temporal and frontal area of the scalp. The baseline values may shift or fast and narrow spikes occur as the fNIRS signals corrupt because of the motion artifacts. This narrow and fast spike can be characterized by a positive interrelation between the HbO and Hb. It can be removed by using wavelet-based filters and the targeted principal component analysis (tPCA). This artifact can also be removed by grand averaging, because muscle artifacts are random in time and amplitude.

d) Environmental Noise

Environmental factors such as ambient light, temperature variations, and electromagnetic interference from electronic devices can also contribute to noise in fNIRS signals. Environmental noise can interfere with the detection of the near-infrared light and affect the accuracy of the measurements.

e) Physiological Noise

Physiological noise arises from natural physiological processes within the body, such as heartbeat, respiration, and blood pressure fluctuations. These physiological changes can introduce rhythmic variations in the fNIRS signals, making it difficult to isolate the brain-related hemodynamic responses from the noise.

2.8 Data Analysis Software

Functional Near-Infrared Spectroscopy (fNIRS) data analysis typically involves the use of specialized software tools designed to process, analyze, and interpret the signals acquired from fNIRS devices. Some commonly used software packages for fNIRS data analysis include:

- **NIRSLab**

NIRSLab is a comprehensive software suite developed for fNIRS data analysis. It provides tools for preprocessing raw data, including motion artifact removal, filtering, and baseline correction. NIRSLab also offers various analysis techniques such as general linear model (GLM) analysis,

correlation analysis, and connectivity analysis. It allows for visualization of results through statistical parametric mapping (SPM) and other graphical representations.

- **Homer2**

Homer2 is an open-source software package widely used for fNIRS data analysis. It offers a range of features for preprocessing, including motion artifact correction, filtering, and signal quality assessment. Homer2 also provides tools for statistical analysis, including task-related response analysis, functional connectivity analysis, and group-level statistics. It is highly customizable and allows for integration with other neuroimaging modalities.

- **NIRS-SPM**

NIRS-SPM is an extension of the Statistical Parametric Mapping (SPM) software specifically tailored for fNIRS data analysis. It offers a user-friendly interface for preprocessing, statistical analysis, and visualization of fNIRS data. NIRS-SPM incorporates advanced techniques such as spatial registration, artifact detection, and multivariate analysis methods for studying brain function.

- **fNIRS Toolbox for MATLAB**

The fNIRS Toolbox is a MATLAB-based software package designed for processing and analyzing fNIRS data. It provides a range of functions for preprocessing, including motion artifact removal, filtering, and signal normalization. The toolbox also offers tools for statistical analysis, including GLM-based analysis, time-frequency analysis, and functional connectivity analysis. It allows for customization and integration with other MATLAB-based neuroimaging tools.

- **NIRS Brain AnalyzIR Toolbox**

The NIRS Brain AnalyzIR Toolbox is a MATLAB-based software package developed for analyzing fNIRS data. It offers a variety of preprocessing techniques, including motion artifact

correction, filtering, and signal normalization. The toolbox also provides tools for statistical analysis, including task-related response analysis, functional connectivity analysis, and machine learning-based classification methods.

2.9 Advantages of fNIRS

Functional Near-Infrared Spectroscopy (fNIRS) offers several advantages, making it a valuable tool for studying brain function in various research and clinical applications.

a) **Non-Invasive:** fNIRS is a non-invasive neuroimaging technique that does not require the use of ionizing radiation or contrast agents. It involves placing optodes on the scalp, making it safe and comfortable for participants, including infants and patients with sensitive conditions.

b) **Portability:** fNIRS devices are typically lightweight and portable, allowing for flexibility in experimental settings. They can be used in laboratory settings, clinical environments, or even in naturalistic settings such as schools or homes, facilitating research in diverse populations and real-world scenarios.

c) **High Temporal Resolution**: fNIRS provides relatively high temporal resolution, capturing changes in brain activity on the order of milliseconds. This allows researchers to investigate dynamic cognitive processes such as language processing, motor control, and decision-making with high precision.

d) **Tolerant to Motion:** Compared to some other neuroimaging techniques like functional Magnetic Resonance Imaging (fMRI), fNIRS is less susceptible to motion artifacts. Participants can engage in naturalistic behaviours such as talking, moving, or performing tasks without significantly affecting the quality of the data.

e) **Versatility:** fNIRS can be used to study various cognitive functions and brain processes, including language processing, memory, attention, emotion regulation, and motor control. It is also suitable for studying developmental changes in brain function across the lifespan.

f) **Accessibility:** fNIRS systems are generally more affordable and easier to set up compared to other neuroimaging modalities such as fMRI or EEG (Electroencephalography). This accessibility makes fNIRS a viable option for researchers with limited resources or in resource-constrained settings.

g) **Complementary to Other Techniques:** fNIRS can be used in conjunction with other neuroimaging techniques such as fMRI, EEG, or TMS (Transcranial Magnetic Stimulation) to provide complementary information about brain function. This multimodal approach allows researchers to gain a more comprehensive understanding of brain activity and connectivity.

2.10 Disadvantages of fNIRS

While functional Near-Infrared Spectroscopy (fNIRS) offers several advantages, it also has some limitations and disadvantages:

a) **Limited Penetration Depth:** Near-infrared light has limited penetration depth in biological tissue, typically reaching only a few centimetres into the cortex. This restricts the depth of brain tissue that can be studied, limiting access to deeper brain regions.

b) **Shallow Brain Coverage:** Due to the limited penetration depth, fNIRS primarily measures cortical brain activity, providing less insight into subcortical structures compared to other neuroimaging techniques like fMRI.

c) **Interference from Scalp and Skull:** Near-infrared light can be absorbed and scattered by scalp and skull tissues before reaching the brain, leading to distortions in the measured signals and reducing spatial specificity.

d) **Limited Spatial Resolution:** While fNIRS provides high temporal resolution, its spatial resolution is relatively lower compared to techniques like fMRI. This can make it challenging to precisely localize brain activation to specific anatomical regions.

e) **Sensitivity to Optode Placement:** Accurate placement of optodes is crucial for obtaining reliable fNIRS measurements. However, achieving consistent and optimal optode placement across participants can be challenging, leading to variability in data quality.

f) **Limited Availability of Analysis Tools**: Compared to other neuroimaging techniques, there is a relatively smaller pool of software tools and standardized analysis pipelines available for fNIRS data analysis. This can pose challenges for researchers, particularly those new to the field.

2.11 Applications

Functional Near-Infrared Spectroscopy (fNIRS) has a wide range of applications across various fields due to its versatility, portability, and ability to non-invasively measure changes in cerebral blood oxygenation. Some key applications of fNIRS include:

- **Neuroimaging Research:** fNIRS is extensively used in neuroscience research to investigate brain function and cognitive processes. It can be applied to study language processing, memory, attention, emotion regulation, motor control, and decision-making, among other cognitive functions.

- **Clinical Assessment:** fNIRS has promising applications in clinical settings for assessing brain function in patients with neurological disorders, monitoring cognitive function during neurorehabilitation, and guiding treatment interventions. It has been used in studies related to stroke recovery, autism spectrum disorders, schizophrenia, traumatic brain injury, and neurodevelopmental disorders.

- **Brain-Computer Interfaces (BCIs):** fNIRS can be integrated into BCI systems to enable direct communication between the brain and external devices. This technology has potential applications in assistive technology, rehabilitation, and control of prosthetic devices for individuals with motor disabilities.

- **Cognitive Load and Mental Workload Assessment:** fNIRS is used to assess cognitive load and mental workload in various tasks and environments, such as driving, aviation, education, and human-computer interaction. Understanding cognitive load can inform the design of systems and interfaces to optimize human performance and user experience.

- **Infant and Child Development:** fNIRS is particularly well-suited for studying brain development in infants and young children due to its non-invasive nature and tolerance to motion. It has been used to investigate early language acquisition, social cognition, executive function, and sensory processing in paediatric populations.

- **Psychological and Psychophysiological Studies:** fNIRS is employed in psychology research to investigate cognitive processes, emotional responses, and social interactions. It has applications in studying empathy, theory of mind, decision-making under uncertainty, and emotional regulation.

Overall, fNIRS offers valuable insights into brain function and behavior across a wide range of applications, contributing to advancements in neuroscience, clinical practice, human-computer interaction, and other interdisciplinary fields.

2.12 References

1. Jöbsis, F.F. "Noninvasive, infrared monitoring of cerebral and myocardial oxygen sufficiency and circulatory parameters." *Science*, vol. 198, no. 4323, pp. 1264–1267, 1977.

2. Ferrari, M.; Giannini, I.; Sideri, G.; Zanette, E. "Continuous non-invasive monitoring of human brain by near infrared spectroscopy." *Advances in Experimental Medicine and Biology*, vol. 191, pp. 873–882, 1985.

3. Hamamatsu Photonics. "Development of the NIR-1000 cerebral oxygenation monitor." *Hamamatsu Technical Digest*, vol. 6, pp. 11–14, 1989.

4. Chance, B.; Zhuang, Z.; UnAh, C.; Alter, C.; Lipton, L. "Cognition-activated low-frequency modulation of light absorption in human brain." *Proceedings of the National Academy of Sciences*, vol. 90, no. 8, pp. 3770–3774, 1993.

5. Hoshi, Y.; Tamura, M. "Detection of dynamic changes in cerebral oxygenation coupled to neuronal function during mental work in man." *Neuroscience Letters*, vol. 150, no. 1, pp. 5–8, 1993.

6. Villringer, A.; Planck, J.; Hock, C.; Schleinkofer, L.; Dirnagl, U. "Near infrared spectroscopy (NIRS): A new tool to study hemodynamic changes during activation of brain function in human adults." *Neuroscience Letters*, vol. 154, no. 1-2, pp. 101–104, 1993.

7. Oishi, H. "Language learners' cortical activation patterns measured by ETG-100." *PhD dissertation*, Nagoya University, 2003.

8. Oka, T.; Yoshino, K.; Yamamoto, K.; Takahashi, H.; Li, Z.; Kawase, T.; and Maeda, Y. "Noninvasive optical topography of human brain function with near-infrared light." *Medical Physics*, vol. 26, no. 5, pp. 809–819, 1999.

9. Molavi, B.; and Dumont, G. "Wavelet-based motion artifact removal for functional near-infrared spectroscopy." *Physiological Measurement*, vol. 33, no. 2, pp. 259–270, 2012.

10. Boas, D.A.; Elwell, C.E.; Ferrari, M.; Taga, G. "Twenty years of functional near-infrared spectroscopy: Introduction for the special issue." *NeuroImage*, vol. 85, Part 1, pp. 1–5, 2014.

11. Huppert, T.J.; Diamond, S.G.; Franceschini, M.A.; Boas, D.A. "HomER: A review of time-series analysis methods for near-infrared spectroscopy of the brain." *Applied Optics*, vol. 48, no. 10, pp. D280-D298, 2009.

12. Strangman, G.; Franceschini, M.A.; Boas, D.A. "Factors affecting the accuracy of near-infrared spectroscopy concentration calculations for focal changes in oxygenation parameters." *NeuroImage*, vol. 18, no. 4, pp. 865–879, 2003.

13. Owen, J.P.; Marco, E.J.; Desai, S.; Fourie, E.; Harris, J.; Hill, S.S.; Arnett, A.B.; Mukherjee, P. "Abnormal white matter microstructure in children with sensory processing disorders." *NeuroImage: Clinical*, vol. 2, pp. 844–853, 2013.

14. Scholkmann, F.; Kleiser, S.; Metz, A.J.; Zimmermann, R.; Mata Pavia, J.; Wolf, U.; Wolf, M. "A review on continuous wave functional near-infrared spectroscopy and imaging instrumentation and methodology." *NeuroImage*, vol. 85, Part 1, pp. 6–27, 2014.

15. Liao, S.M.; Gregg, N.M.; White, B.R.; Zeff, B.W.; Bjerkaas, K.A.; Inder, T.E.; Culver, J.P. "Neonatal hemodynamic response to visual cortex activity: High-density near-infrared spectroscopy study." *Journal of Biomedical Optics*, vol. 17, no. 8, p. 086006, 2012.

16. Gagnon, L.; Cooper, R.J.; Yücel, M.A.; Perdue, K.L.; Greve, D.N.; Boas, D.A. "Short separation channel location impacts the performance of short channel regression in NIRS." *NeuroImage*, vol. 59, no. 3, pp. 2518–2528, 2012.

17. Huppert, T.J.; Hoge, R.D.; Diamond, S.G.; Franceschini, M.A.; Boas, D.A. "A temporal comparison of BOLD, ASL, and NIRS hemodynamic responses to motor stimuli in adult humans." *NeuroImage*, vol. 29, no. 2, pp. 368–382, 2006.

18. Schmitz, C.H.; Klemer, D.P.; Hardin, R.; Katz, M.S.; Pei, Y.; Graber, H.L.; Levin, M.B.; Lev, M.H.; Barbour, R.L. "Design and implementation of dynamic near-infrared optical tomographic imaging instrumentation for simultaneous dual-breast measurements." *Applied Optics*, vol. 44, no. 11, pp. 2140–2153, 2005.

19. Scholkmann, F.; Spichtig, S.; Muehlemann, T.; Wolf, M. "How to detect and reduce movement artifacts in near-infrared imaging using moving standard deviation and spline interpolation." *Physics in Medicine and Biology*, vol. 57, no. 22, pp. 7793–7809, 2012.

20. Funane, T.; Atsumori, H.; Katura, T.; Obata, A.; Sato, H.; Tanikawa, Y.; Kiguchi, M.; Maki, A. "Quantitative evaluation of deep and shallow tissue layers' contribution to fNIRS signal using multi-distance optodes and independent component analysis." *NeuroImage*, vol. 85, Part 1, pp. 150–165, 2014.

CHAPTER 3

Signal Processing

3.1 Introduction

Signal processing, a branch of electrical engineering, deals with the examination, alteration, and creation of signals encompassing various forms such as audio, visual, seismic, and scientific data. Its methodologies are applied to enhance transmission quality, improve digital storage efficiency, rectify distorted signals, enhance video quality, and identify specific elements within measured signals.

3.2 Categories

- **Analog**

Analog signal processing pertains to signals that haven't undergone digitization, as seen in traditional 20th-century radio, telephone, and television systems [1] . This involves both linear and nonlinear electronic circuits. Linear circuits include passive and active filters, mixers, integrators, and delay lines. Nonlinear circuits consist of compandors, multipliers (such as frequency mixers and voltage-controlled amplifiers), filters, oscillators, and phase-locked loops.

- **Continuous time**

Continuous-time signal processing deals with signals that change continuously (without discrete points). It encompasses methods like time domain, frequency domain, and complex frequency domain. This field primarily focuses on modelling linear time-invariant continuous systems, calculating the integral of the system's zero-state response, establishing system functions, and continuous-time filtering of deterministic signals.

- **Discrete time**

Discrete-time [2] signal processing handles sampled signals, defined only at discrete time points and quantized in time but not magnitude. Analog discrete-time signal processing relies on electronic devices like sample and hold circuits, analog time-division multiplexers, delay lines, and feedback shift registers. This technology predates digital signal processing and is still used for advanced processing of gigahertz signals. The theoretical discipline of discrete-time signal processing

establishes a mathematical foundation for digital signal processing without considering quantization error.

- **Digital**

Digital signal processing involves processing digitized discrete-time sampled signals. This processing is conducted using general-purpose computers or digital circuits such as ASICs, field-programmable gate arrays, or specialized digital signal processors (DSP chips). Typical operations include fixed-point and floating-point arithmetic, real-valued and complex-valued arithmetic, multiplication, addition, circular buffers, and lookup tables. Algorithms commonly used are the fast Fourier transform (FFT), finite impulse response (FIR) filter, infinite impulse response (IIR) filter, and adaptive filters like the Wiener and Kalman filters.

- **Nonlinear**

Nonlinear signal processing deals with the analysis and processing of signals generated by nonlinear systems, encompassing time, frequency, or spatiotemporal domains. Nonlinear systems can exhibit complex behaviours such as bifurcations, chaos, harmonics [3] , and subharmonics, which linear methods cannot analyze or produce. Polynomial signal processing is a subtype of nonlinear signal processing where polynomial systems extend linear systems to the nonlinear realm.

- **Statistical**

Statistical signal [8] processing treats signals as stochastic processes, utilizing their statistical properties for signal processing tasks. Statistical techniques are widely employed in signal processing applications, such as modelling the probability distribution of noise in image photography and developing noise reduction techniques based on this model.

3.3 How signal Processing involve in FNIRs?

Signal processing [4] plays a crucial role in functional near-infrared spectroscopy (fNIRS) throughout various stages of data acquisition, analysis [5] , and interpretation. Here's how signal processing is involved in fNIRS:

- **Data Preprocessing**

1. **Motion Artifact Removal:** fNIRS data can be affected by motion artifacts caused by subject movement. Signal processing techniques are used to identify and remove these artifacts to ensure the quality of the data.

2. **Baseline Correction:** Preprocessing involves correcting for baseline shifts in the raw optical data to eliminate slow drifts and ensure that the signals are centred around zero.

3. **Filtering:** Filtering techniques, such as high-pass, low-pass, or band-pass filters [12] , are applied to remove noise from the data, including physiological noise like heartbeat and respiration.

- **Hemodynamic Response Estimation**

1. **Concentration Changes:** fNIRS measures changes in oxyhemoglobin (HbO) and deoxyhemoglobin (HbR) concentrations in the brain tissue. Signal processing techniques are used to convert changes in light intensity into concentration changes of HbO and HbR.

2. **Hemodynamic Response Function (HRF):** Signal processing methods, such as deconvolution techniques, estimate the shape and timing of the HRF from the concentration changes. This helps in identifying neural activity patterns.

- **Spatial Registration**

1. **Alignment with Brain Anatomy:** fNIRS channels need to be spatially registered with specific brain regions for accurate interpretation. Signal processing methods, including atlas-based registration and Monte Carlo simulations, aid in aligning fNIRS measurements with brain anatomy.

- **Statistical Analysis:**

1. **Activation Detection:** Statistical methods [13] , such as general linear modelling (GLM), are applied to identify brain regions activated during specific tasks or conditions based on fNIRS data.

2. **Machine Learning:** Machine learning techniques, including classification algorithms, can be employed to analyze fNIRS data for pattern recognition tasks or to predict cognitive states.

- **Connectivity Analysis:**

1. **Functional Connectivity:** Signal processing methods, such as coherence analysis or correlation analysis, are used to investigate functional connectivity between different brain regions based on fNIRS signals.

2. **Network Analysis:** Advanced signal processing techniques help in analyzing complex networks of brain activity derived from fNIRS data to understand brain functioning in different conditions.

Each distinct signal processing methods had its own processing sequence. Signal processing and data analysis were conducted using MATLAB (Version R2013b, MathWorks, Natick, MA, USA) along with custom scripts, as well as with a commercially available MATLAB-based fNIRS software (nirsLAB, version v2016.05, NIRx Medical Technologies, Glen Head, NY, USA). MATLAB codes are accessible from the corresponding author upon reasonable request.

Initially, raw optical density (OD) data were imported into the nirsLAB software alongside probe details [15] , and any channels with a coefficient of variation exceeding 15% (Piper et al., 2014) were excluded from further analysis (two channels in one participant and one channel in two other participants). Subsequently, data processing was conducted individually for each signal processing method based on the outlined pipelines (refer to Figure 3.1 for visualization)

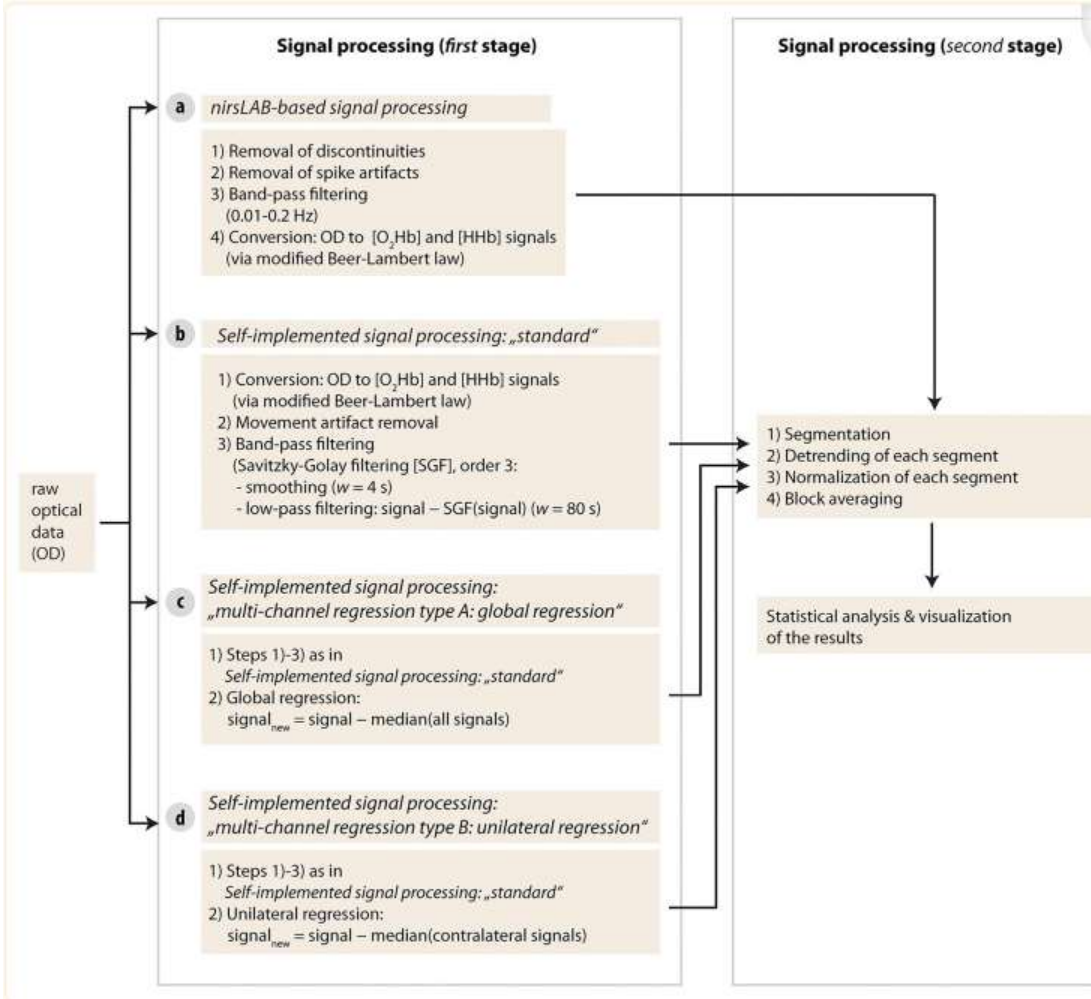


Figure 3(a): Visualization of the signal processing approaches used

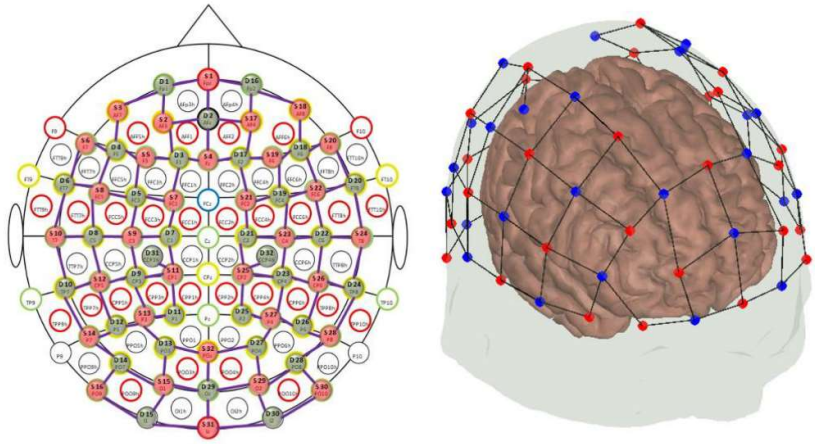


Figure 3(b) : FNIRs Signal Processing pipelines

3.4 Feature Extraction

Feature extraction [20] is a fundamental process in machine learning and data analysis, essential for identifying and extracting pertinent information from raw data. These extracted features serve as the building blocks for creating a more informative dataset, facilitating various tasks such as-

- Classification
- Prediction
- clustering.

The primary objective of feature extraction is to reduce the complexity of the data, commonly referred to as "data dimensionality", while preserving as much relevant information as possible. By doing so, feature extraction enhances the performance and efficiency of machine learning algorithms and simplifies the subsequent analysis process.

Feature extraction encompasses several key aspects:

1. **Identification of Relevant Features:** The process begins with identifying features in the raw data that are likely to be relevant to the task at hand. This may involve domain knowledge, statistical analysis, or automated feature selection techniques.

2. **Extraction of Features:** Once relevant features are identified, they are extracted from the raw data. This extraction process can involve various methods, such as statistical calculations, transformation techniques, or signal processing algorithms.

3. **Creation of New Features (Feature Engineering):** In some cases, existing features may not adequately capture the underlying patterns in the data. Feature engineering involves creating new features derived from the original data, often through combinations, transformations, or aggregations of existing features.

4. **Data Manipulation and Preprocessing:** Feature extraction may also involve preprocessing steps to clean and prepare the data for analysis. This may include handling missing values, normalization, or scaling of features to ensure consistency and improve algorithm performance.

5. **Dimensionality Reduction:** A key goal of feature extraction is to reduce the dimensionality of the dataset while retaining relevant information. Dimensionality reduction techniques, such as principal component analysis (PCA) or t-distributed stochastic neighbor embedding (t-SNE), are commonly employed to achieve this objective.

6. **Evaluation and Validation:** It is essential to evaluate the extracted features to ensure their quality and effectiveness for the intended task. This may involve assessing their discriminatory power, relevance, or contribution to model performance through validation techniques such as cross-validation or holdout validation.

3.4.1 What is Feature?

A feature is an individual measurable property within a recorded dataset. In machine learning and statistics, features are often called “variables” or “attributes.” Relevant features have a correlation or bearing on a model’s use case. In a patient medical dataset, features could be age, gender, blood pressure, cholesterol level, and other observed characteristics relevant to the patient.

Another visual example for feature:

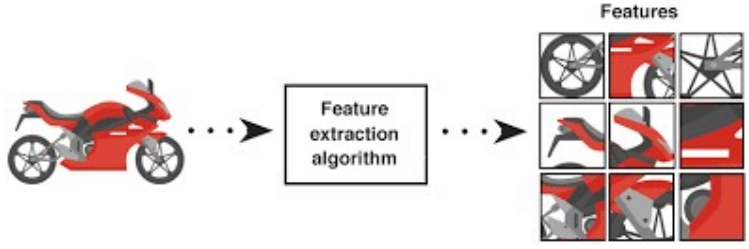


Figure 3(c) : Visual representation of feature Extraction

3.4.2 Importance of Feature Extraction

Feature extraction is essential across various practical applications, including image and speech recognition, predictive modeling, and Natural Language Processing (NLP). In these contexts, the raw data often comprises numerous features, many of which may be irrelevant or redundant. This abundance of features poses challenges for algorithms, hindering their ability to accurately process the data.

Through feature extraction, the pertinent features are isolated or "extracted" from the irrelevant ones. This process simplifies the dataset by reducing the number of features that need to be processed. Consequently, the analysis becomes more accurate and efficient as algorithms can focus on the most relevant aspects of the data. In essence, feature extraction enhances the effectiveness of machine learning and data analysis tasks by streamlining the data representation and removing unnecessary complexity.

3.4.3 Common Feature Types

- **Numerical Features:** These are features represented by numeric values, such as integers or floats. Examples include age, salary, and height.

- **Categorical Features:** These features have a finite set of possible values and represent qualitative characteristics. Examples include gender (male, female, X) and color (red, blue, green).

- **Ordinal Features:** These are categorical features that have a defined order or ranking. Examples include T-shirt sizes (S, M, L, XL), where there's a clear hierarchy.

- **Binary Features:** This category includes features with only two possible values, representing yes/no or true/false conditions. Examples include is smoker (yes, no) and has subscription (true, false).

- **Text Features:** These features contain textual data, such as words or sentences. Textual data typically requires preprocessing steps like tokenization to convert it into a suitable format for machine learning models.

- **Image Features:** Image data consists of pixels representing visual information. Features may include color histograms, texture descriptors, edge detectors, or deep learning features extracted from pre-trained convolutional neural networks (CNNs).

- **Temporal Features:** Temporal data involves time-series information, where observations are collected over time. Features may include timestamps, periodic patterns, trends, or seasonality.

- **Spatial Features:** Spatial data represents information about physical locations or spatial relationships. Features may include coordinates (latitude, longitude), distances, areas, or geometric properties.

- **Derived Features:** Derived features are created by transforming or combining existing features. Examples include polynomial features, logarithmic transformations, interaction terms, or domain-specific feature engineering techniques.

- **Frequency Features:** Frequency domain features represent the frequency content of signals or time-series data. Examples include Fourier transform coefficients, power spectral density, or wavelet coefficients.

3.4.4 Feature Normalization

Feature normalization [17] , also known as feature scaling, is a preprocessing technique used to standardize the range of features in a dataset. Normalization standardizes the range of independent variables or features of the data. This process can make certain algorithms converge faster and lead to better model performance, especially for algorithms sensitive to the scale of input features. It involves adjusting the values of the features so that they fall within a similar scale, typically between 0 and 1 or with a mean of 0 and a standard deviation of 1. This normalization process ensures that all features contribute equally to the learning process and prevents features with larger scales from dominating those with smaller scales.

- **Motivation**

In many machine learning algorithms, features with larger scales can have a disproportionate impact on the model's behaviour compared to features with smaller scales. This can lead to biased model predictions and slow convergence during training.

Feature normalization addresses this issue by scaling features to a similar range, ensuring that each feature contributes proportionally to the learning process.

- **Techniques**

1. **Min-Max Scaling:** This technique [18] scales features to a specified range, typically [0, 1] or [-1, 1]. It subtracts the minimum value from each feature and divides by the range (maximum value minus minimum value).

2. **Standardization (Z-score Normalization):** This technique standardizes features to have a mean of 0 and a standard deviation of 1. It subtracts the mean from each feature and divides by the standard deviation.

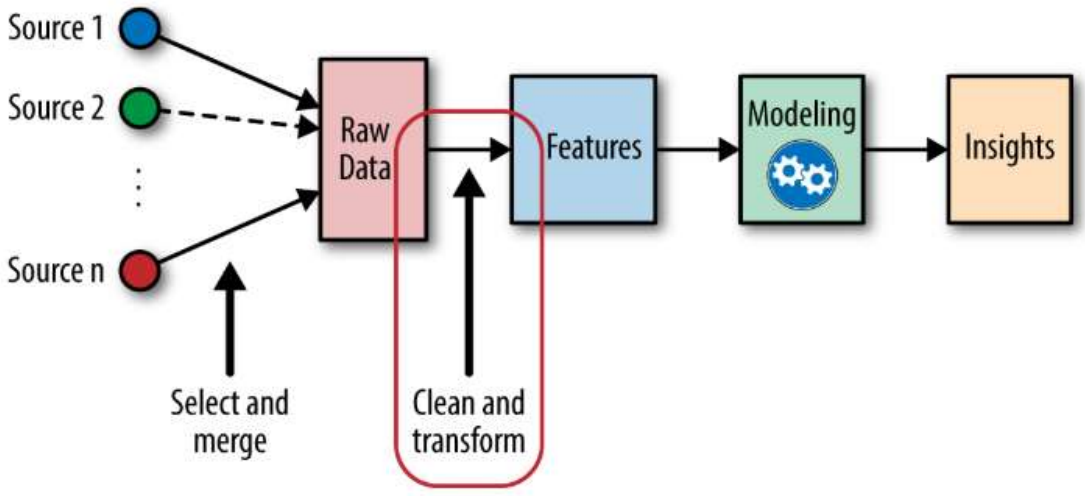


Figure 3(d) : Feature Engineering Lifecycle

- **Advantages**

1. **Improved Algorithm Performance:** Normalizing features can improve the performance of many machine learning algorithms, particularly those sensitive to feature scales or distances,

such as gradient descent-based algorithms, k-nearest neighbours (KNN), and support vector machines (SVM).

2. **Faster Convergence:** Normalized features can lead to faster convergence during training, as features are on a similar scale, reducing the oscillations and instability often observed with unscaled features.

- ## Considerations

1. **Outliers:** Feature normalization may be sensitive to outliers, particularly with Min-Max scaling. Outliers can disproportionately affect the scaling process and may require special handling, such as outlier removal or robust scaling techniques.

2. **Interpretability:** While normalization improves algorithm performance, it can make the interpretation of feature importance less intuitive, especially with standardization, where feature values are no longer in their original units.

- ## Implementation

Feature normalization is typically applied separately to each feature in the dataset. This ensures that each feature is scaled independently and maintains its individual characteristics. Many libraries and frameworks for machine learning, such as scikit-learn in Python, provide built-in functions for feature normalization, making it easy to implement as part of the data preprocessing pipeline.

In summary, feature normalization is a crucial preprocessing step in machine learning that standardizes the scale of features to improve algorithm performance and convergence. It helps mitigate the impact of feature scales on model training and ensures that all features contribute equally to the learning process.

3.5 References

1. L. You, X. Qiang, Y. Zhu, F. Jiang, C.G. Tsinos, W. Wang, H. Wymeersch, X. Gao, and B. Ottersten, "Integrated Communications and Localization for Massive MIMO LEO Satellite Systems," *IEEE Transactions on Wireless Communications*, vol. 23, no. 5, pp. 1234-1245, May 2024.

2. V. Oppenheim and R. W. Schafer, "Discrete-Time Signal Processing," *Proceedings of the IEEE*, vol. 61, no. 6, pp. 724-725, Jun. 1973.

3. R. E. Blahut, "Theory and Practice of Error Control Codes," *IEEE Transactions on Information Theory*, vol. 28, no. 4, pp. 571-572, Jul. 1982.

4. J. G. Proakis and D. G. Manolakis, "Digital Signal Processing: Principles, Algorithms, and Applications," *IEEE Transactions on Education*, vol. 34, no. 2, pp. 175-176, May 1991.

5. V. R. Algazi, "Applications of Time-Frequency Signal Processing," *IEEE Signal Processing Magazine*, vol. 9, no. 5, pp. 65-66, Sep. 1992.

6. R. C. Gonzalez and R. E. Woods, "Digital Image Processing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 1, pp. 121-122, Jan. 1988.

7. S. G. Mallat, "A Wavelet Tour of Signal Processing: The Sparse Way," *IEEE Transactions on Information Theory*, vol. 49, no. 7, pp. 2061-2061, Jul. 2003.

8. Papoulis and S. U. Pillai, "Probability, Random Variables, and Stochastic Processes," *IEEE Transactions on Information Theory*, vol. 35, no. 4, pp. 595-595, Jul. 1989.

9. V. Oppenheim, "Signals and Systems," *IEEE Transactions on Audio and Electroacoustics*, vol. 21, no. 1, pp. 73-74, Feb. 1973.

10. G. Kramer, I. Marić, and R.D. Yates, "Cooperative communications," *Foundations and Trends® in Networking*, vol. 1, no. 3-4, pp. 271-425, 2007.

11. T. S. Rappaport, "Wireless Communications: Principles and Practice," *IEEE Transactions on Vehicular Technology*, vol. 42, no. 4, pp. 781-782, Nov. 1993.

12. Boashash, "Time-Frequency Signal Analysis: Methods and Applications," *IEEE Signal Processing Magazine*, vol. 18, no. 6, pp. 85-86, Nov. 2001.

13. Papoulis, "Probability, Random Variables, and Stochastic Processes," *Proceedings of the IEEE*, vol. 86, no. 2, pp. 274-275, Feb. 1998.

14. R.G. Brown and P.Y. Hwang, "Introduction to random signals and applied Kalman filtering: with MATLAB exercises and solutions," *IEEE Press*, 1997

.

15. W. Van Drongelen, "Signal processing for neuroscientists," *Academic press*, 2018.

16. Bruderlin and L. Williams, "Motion signal processing," *in Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pp. 97-104, Sep. 1995.

17. Li, F. Wu, S.N. Lim, S. Belongie, and K.Q. Weinberger, "On feature normalization and data augmentation," *in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12383-12392, 2021.

18. Singh and B. Singh, "Feature wise normalization: An effective way of normalizing data," *Pattern Recognition*, vol. 122, p. 108307, 2022.

19. H.K. Ekenel and R. Stiefelhagen, "Analysis of local appearance-based face recognition: Effects of feature selection and feature normalization," *in 2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)*, pp. 34-34, Jun. 2006.

20. W.K. Mutlag, S.K. Ali, Z.M. Aydam, and B.H. Taher, "Feature extraction methods: a review," *in Journal of Physics: Conference Series*, vol. 1591, no. 1, p. 012028, Jul. 2020.

CHAPTER 4

Classification

4.1 Introduction to Machine Learning

Arthur Samuel, a pioneer in computer gaming and artificial intelligence, introduced the term "Machine Learning" in 1959 at IBM. He defined it as the field where computers learn without being explicitly programmed. However, there isn't a universally accepted definition for machine learning, and different authors define it in various ways. Here are two more definitions:

Machine learning [1] involves programming computers to optimize performance using example data or past experience. It entails defining a model with certain parameters and using a computer program to adjust these parameters using training data or past experience. The model can be predictive, making future predictions, or descriptive, extracting knowledge from data.

The field of machine learning focuses on creating computer programs that improve automatically with experience. It is a subset of artificial intelligence and involves developing algorithms and models that enable computers to enhance their performance in tasks through experience. These algorithms learn from data and make predictions or decisions without explicit programming.

Machine learning encompasses various types [2], including supervised learning (training on labeled data), unsupervised learning (training on unlabeled data), and reinforcement learning (training through trial and error). It is applied in diverse areas such as image and speech recognition, natural language processing, and recommendation systems.

Machine learning algorithms can be broadly categorized into three main types:

a. Supervised Learning

In supervised learning, the algorithm is trained on a labeled dataset, where each example is paired with the correct label. The algorithm learns to map input data to the correct output based on these examples. Common tasks in supervised learning include classification (assigning labels to input data) and regression (predicting a continuous value).

b. Unsupervised Learning

In unsupervised learning, the algorithm is trained on an unlabeled dataset, and its goal is to find hidden patterns or structures in the data. Clustering (grouping similar data points together) and dimensionality reduction (reducing the number of features in the data) are common tasks in unsupervised learning.

c. Reinforcement Learning

Reinforcement learning involves training an agent to interact with an environment and learn the optimal actions to take in different situations. The agent receives feedback in the form of rewards or penalties based on its actions, and it learns to maximize its cumulative reward over time.

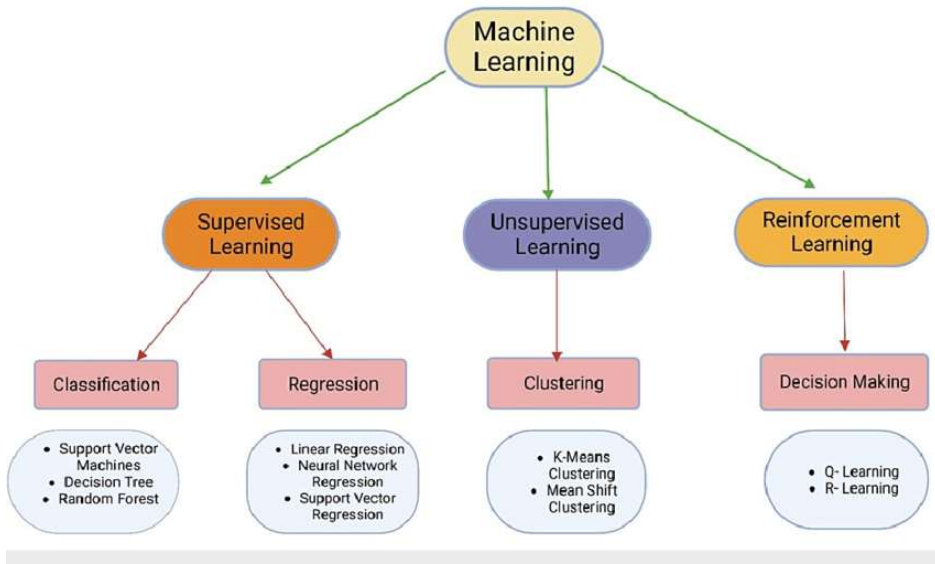


Figure 4 (a): Types of Machine Learning

4.1.1 How Does Machine Learning Work?

Machine learning works by enabling computers to learn from data and make predictions or decisions based on that data, without being explicitly programmed [3] to do so. The process typically involves the following steps:

- **Data Collection**: The first step in any machine learning project is to collect relevant data. This data can come from various sources, such as sensors, databases, or the internet.

- **Data Preprocessing:** Once the data is collected, it needs to be preprocessed to make it suitable for training a machine learning model. This may involve cleaning the data, handling missing values, and encoding categorical variables.

- **Feature Selection/Extraction:** In this step, the relevant features (or variables) that will be used to train the model are selected or extracted from the data. This helps reduce the dimensionality of the data and focus on the most important features.

- **Model Selection:** Next, a suitable machine learning model is selected based on the type of problem and the nature of the data. Common types of models include decision trees, neural networks, and support vector machines.

- **Model Training:** The selected model is trained on the training data, where it learns the underlying patterns and relationships in the data. During training, the model adjusts its parameters to minimize the difference between its predictions and the actual values in the training data.

- **Model Evaluation:** Once the model is trained, it is evaluated using a separate dataset called the validation or test set. This helps assess the model's performance and determine if it can generalize well to new, unseen data.

- **Model Tuning:** If the model's performance is not satisfactory, it may be necessary to tune its hyperparameters (parameters that control the learning process) to improve its performance.

- **Prediction/Inference:** Once the model is trained and evaluated, it can be used to make predictions or decisions on new, unseen data. This is known as the inference phase.

- **Deployment:** Finally, if the model performs well, it can be deployed to production to make real-time predictions or decisions. It is important to monitor the model's performance over time and retrain it as needed to ensure its accuracy.

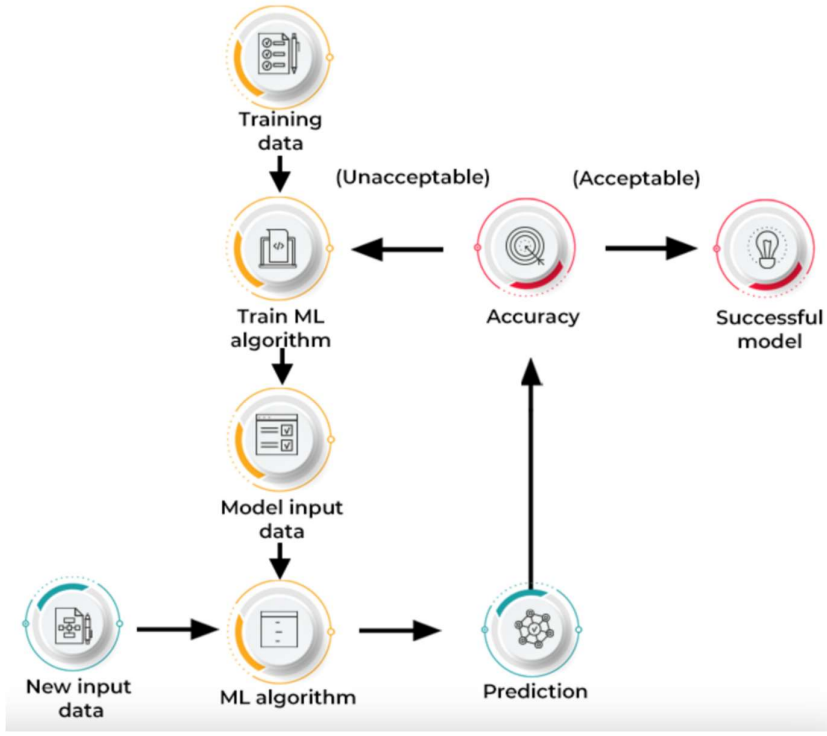


Figure 4 (b): Steps for Machine Learning Algorithm

4.1.2 Classification with Machine Learning

In machine learning, classification is a type of supervised learning that aims to categorize data points into distinct classes or categories based on their features [4]. This process involves training a model using a dataset where each data point is labeled with its corresponding class. The model learns patterns and relationships in the data to make predictions about the class of new, unseen data points.

During the training phase, the model uses algorithms to find the best way to divide the data into classes based on the features provided. This trained model can then be used to predict the class of new data points, helping to classify them into the appropriate categories. Classification is used in various applications, such as spam detection in emails, sentiment analysis in text, and medical diagnosis based on patient data. It is a fundamental concept in machine learning and is essential for many real-world problems [5] where categorization is necessary.

4.1.3 List of Machine Learning Classification Algorithms

Here is a list of some popular classification algorithms:

1. Logistic Regression
2. Decision Tree
3. Random Forest
4. Support Vector Machine (SVM)
5. Naive Bayes
6. K-Nearest Neighbors (KNN)

These algorithms [6] are used to automate decision-making and identify patterns in data, making them valuable tools in many fields.

4.2 Deep Learning

4.2.1 Introduction to Deep Learning

Deep learning [16] is a subset of machine learning that relies on artificial neural networks, specifically deep neural networks (DNNs). It is adept at recognizing intricate patterns and relationships in data, negating the need for explicit programming of all rules. Its popularity has surged recently due to advancements in processing capabilities and the abundance of large datasets. DNNs are inspired by the human brain's neurons and are designed to learn from vast amounts of data.

Deep learning leverages neural networks, which mimic the brain's structure, consisting of interconnected nodes that process and transform data. Its hallmark feature is the use of deep neural networks, with multiple layers of nodes, enabling the discovery of complex data representations through hierarchical patterns.

This technology has made significant strides in various domains like image and speech recognition, natural language processing, and recommendation systems. It excels in automatically learning and improving from data, eliminating the need for manual feature engineering.

Training deep neural networks traditionally demands substantial data and computational resources. However, with the advent of cloud computing and specialized hardware like GPUs, training such networks has become more feasible.

In essence, deep learning is a powerful tool within machine learning, using deep neural networks to address complex problems. Its applications are vast and continue to expand as data availability increases and computing resources become more potent.

- **Applications of Deep Learning**

The primary applications of deep learning are in computer vision, natural language processing (NLP), and reinforcement learning. In computer vision, deep learning models enable machines to interpret and process visual data. They can detect and recognize objects in images and videos, classify images into categories, and segment images into different regions. These applications are used in self-driving cars, surveillance, robotics, medical imaging, quality control, and image retrieval.

In natural language processing (NLP), deep learning models help machines understand and generate human language. They can automatically generate text, translate text between languages, analyze the sentiment of text, and transcribe spoken words. These applications are used in automatic text generation, language translation, sentiment analysis, customer service, social media monitoring, and speech-to-text conversion.

- **Challenges in Deep Learning**

Deep learning has made significant strides in various fields, but it still faces several challenges. These include:

a. **Data Availability:** Deep learning requires large datasets for training, which can be challenging to gather and label.

b. **Computational Resources:** Training deep learning models is computationally expensive, requiring specialized hardware like GPUs and TPUs.

c. **Time-Consuming:** Training deep learning models on sequential data can be time-consuming, sometimes taking days or even months.

d. **Interpretability:** Deep learning models are often considered black boxes, making it difficult to interpret their results and understand how they arrive at their decisions.

e. **Overfitting:** Deep learning models can become too specialized to the training data, leading to overfitting and poor performance on new data.

- ## Advantages of Deep Learning

Deep learning offers several advantages:

a. **High Accuracy:** Deep learning algorithms can achieve exceptional performance levels in tasks like image recognition and natural language processing, often surpassing traditional machine learning methods.

b. **Automated Feature Engineering:** Unlike traditional machine learning approaches that require manual feature engineering, deep learning algorithms can automatically learn and extract relevant features from data.

c. **Scalability:** Deep learning models are highly scalable and can effectively handle large and complex datasets, making them suitable for applications with massive amounts of data.

d. **Flexibility:** Deep learning models are versatile and can be applied to a wide range of tasks, including image analysis, speech recognition, and text processing.

e. **Continual Improvement:** Deep learning models have the capability to continuously enhance their performance as they are exposed to more data, allowing for continual learning and adaptation.

● Disadvantages of Deep Learning

Deep learning comes with several disadvantages:

a. **High Computational Requirements:** Training and optimizing deep learning models demand significant computational resources, including high-performance GPUs or TPUs, which can be costly.

b. **Large Data Requirements:** Deep learning models often require vast amounts of labelled data for effective training, posing challenges in data collection and labelling.

c. **Interpretability Challenges:** The inner workings of deep learning models can be complex and opaque, making it challenging to interpret how they make decisions.

d. **Overfitting Risk:** Deep learning models can be prone to overfitting, where they perform well on training data but generalize poorly to new, unseen data.

In summary, while deep learning offers significant advantages, such as high accuracy and scalability, it also presents challenges that need to be carefully considered, such as high computational requirements and interpretability issues.

● Classification in Deep Learning

Classification with deep learning involves using deep neural networks to categorize data into predefined classes or categories. Deep learning models, such as convolutional neural networks (CNNs) [17] for image classification or recurrent neural networks (RNNs) for sequential data, are trained on labeled datasets to learn the patterns and features that distinguish different classes.

The process typically involves the following steps:

- **Data Preparation**: Collect and preprocess the dataset, ensuring it is labeled and appropriately formatted for training.

- **Model Selection:** Choose a deep learning model suitable for the classification task based on the nature of the data and the problem requirements.

- **Model Training:** Train the selected model on the labeled dataset using optimization techniques such as stochastic gradient descent (SGD) to minimize the loss function.

- **Model Evaluation:** Evaluate the trained model using a separate validation dataset to assess its performance and adjust hyperparameters if necessary.

- **Inference:** Use the trained model to predict the classes of new, unseen data samples.

Deep learning models have demonstrated state-of-the-art performance in various classification tasks, including image recognition, speech recognition, natural language processing, and many others. Their ability to automatically learn complex features from data makes them highly effective for classification tasks across different domains.

4.2.2 Types of Algorithms used in Deep Learning

The following is a list of the top 10 most popular deep learning algorithms:

1. Convolutional Neural Networks (CNNs)
2. Long Short-Term Memory Networks (LSTMs)

3. Recurrent Neural Networks (RNNs)
4. Generative Adversarial Networks (GANs)
5. Radial Basis Function Networks (RBFNs)
6. Multilayer Perceptron (MLPs)
7. Self-Organizing Maps (SOMs)
8. Deep Belief Networks (DBNs)
9. Restricted Boltzmann Machines (RBMs)
10. Autoencoders

These algorithms are widely used in various deep learning applications due to their effectiveness in handling different types of data and tasks.

4.2.3 Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) [18] are a type of deep learning algorithm that is particularly effective for image recognition and classification tasks. CNNs are inspired by the organization of the animal visual cortex and use a variation of multilayer perceptron designed to require minimal preprocessing. They are able to capture spatial hierarchies of features in an image through the application of convolutional filters. This allows them to learn features such as edges, textures, and shapes at different levels of abstraction, making them highly effective for tasks such as object recognition in images and video analysis.

CNNs typically consist of multiple layers, including convolutional layers, pooling layers, and fully connected layers. Convolutional layers apply convolutional filters to the input image, extracting features and creating feature maps. Pooling layers reduce the spatial dimensions of the feature maps, reducing computational complexity and controlling overfitting. Fully connected layers are used for classification, taking the features learned by the previous layers and using them to classify the input image into different categories.

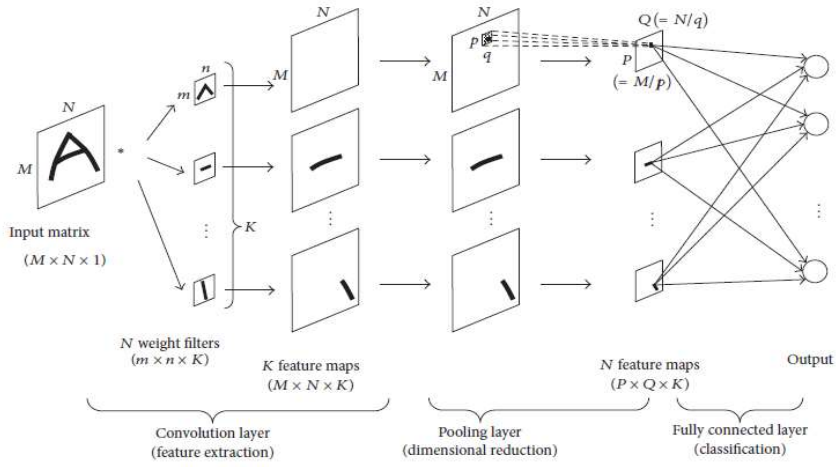


Figure 4 (c): Convolutional Neural Network

A type of feed-forward artificial neural network (ANN) is convolution neural network (CNN) which has two hidden layers, pooling and convolution. Through the process of repetition, the weight of the hidden layers is learned within the connected network.

Suppose a CNN is applied on an image recognition process that is shown in figure 4(h). The input to the convolution layer is M×N pixel image. In the convolution layer "kernels" are referred to as the K numbers of feature extraction filters. This kernel is convoluted over the entire image using m×n window. The convolution process after the convolution that takes place between the image and the kernels is computed as

$$U_{ijk} = \sum_{a=0}^{m-1} \sum_{b=0}^{n-1} x_{(i+a)(j+b)} w_{abk} \qquad (4.2)$$

In the above equation x_{ij} is defined as the pixel of (i, j) of the input matrix of the image, w_{ijk} is the weight to the pixel (i, j) of the k^{th} kernel, and u_{ijk} is the convoluted output with the k^{th} kernel at the pixel (i, j). The final output in the convolution layer after processed by the activation function:

$$y_{ijk} = f(u_{ijk}) \tag{4.3}$$

In the convolution layer, finally the K feature map is obtained. A CNN doesn't require any pre-processing of feature extraction like the ordinary ANN. The feature extraction process is included in its architecture. The feature map output obtained from the convolution layer is then processed by the pooling layer. The extra feature information for learning is discarded in the pooling layer thereby controlling the overfitting. Different types of pooling are there. Among them the most famous one is the mixed pooling.

- **Pooling Operation**

In this method, (p× q) pixel that referred to as a pooling size are extracted from each feature map and maximum value is adopted as a single representative output value as

$$y'_{cdk} = y_{ijk} \tag{4.4}$$

$p_{ijk}$is the pooling block with pixel (i, j) in the k^{th} feature map. y'_{cdk} is the output of each pixel (c, d) in the $\frac{M}{p} \times \frac{N}{q}$ image. There are several years of convolution and pooling layers are repetitively connected.

Similar to the ordinary ANN the output layer comprises a fully connected neural network. In the data classification process the number of neurons is equivalent to the number of data classification. In the output layer each neuron is connected to every neuron in the previous layers (final pooling layer) and its output is computed as

$$u_j = \sum_{i=1}^{\frac{M}{p}\times\frac{N}{q}\times k} w_{ij} y'_i + b_j \tag{4.5}$$

$$p_j = \frac{e^{u_j}}{\sum_{k=1}^{c} e^{u_k}} \tag{4.6}$$

Where, $u_j$is the input to the neuron j in the output layer, the output of the pooling layer is denoted as y_i', w_{ij} is the weight from neuron i in the pooling layer to the neuron j in the output layer. The bias is denoted as b_j, c is the number of classes in the data classification and the probability that the input data is classified to class j is p_j

The process of back propagation is used in order to learn the weight like an ordinary ANN. The CNN is trained to classify the input data by repeating the forward and the back propagation.

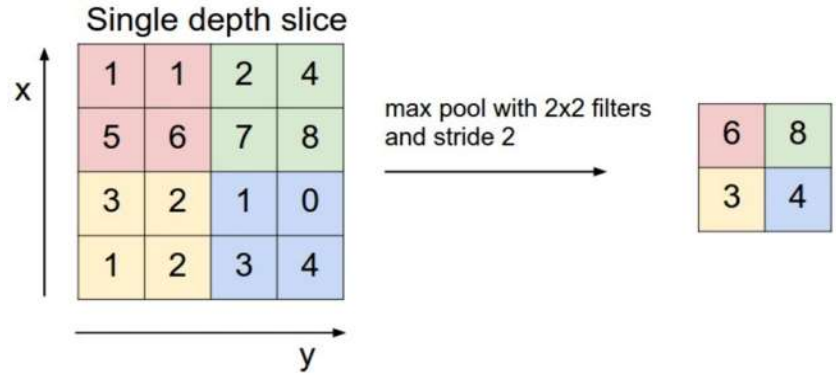


Figure 4 (d): Max. Pooling operation

- ## CNN Architecture

1. **Architecture:** A typical CNN architecture [19] consists of three main types of layers: convolutional layers, pooling layers, and fully connected layers.

    - **Convolutional Layers:** These layers apply convolution operations to the input image using learnable filters. Each filter detects specific patterns or features in the input, such as edges or textures. The output of a convolutional layer is a set of feature maps that represent the presence of these features at different spatial locations in the input.

    - **Pooling Layers:** Pooling layers reduce the spatial dimensions of the feature maps while retaining the most important information. This helps in reducing the computational

complexity of the network and controlling overfitting. Common pooling operations include max pooling and average pooling.

- **Fully Connected Layers:** These layers perform classification based on the features extracted by the convolutional and pooling layers. They connect every neuron in one layer to every neuron in the next layer, allowing the model to learn complex patterns in the data.

2. Activation Functions

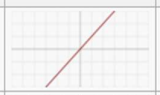
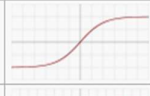
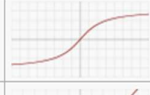
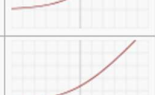
Name	Plot	Equation	Derivative
Identity		$f(x) = x$	$f'(x) = 1$
Binary step		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x \neq 0 \\ ? & \text{for } x = 0 \end{cases}$
Logistic (a.k.a Soft step)		$f(x) = \frac{1}{1+e^{-x}}$	$f'(x) = f(x)(1-f(x))$
TanH		$f(x) = \tanh(x) = \frac{2}{1+e^{-2x}} - 1$	$f'(x) = 1 - f(x)^2$
ArcTan		$f(x) = \tan^{-1}(x)$	$f'(x) = \frac{1}{x^2+1}$
Rectified Linear Unit (ReLU)		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Parameteric Rectified Linear Unit (PReLU) [2]		$f(x) = \begin{cases} \alpha x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} \alpha & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Exponential Linear Unit (ELU) [3]		$f(x) = \begin{cases} \alpha(e^x - 1) & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} f(x) + \alpha & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
SoftPlus		$f(x) = \log_e(1+e^x)$	$f'(x) = \frac{1}{1+e^{-x}}$

Figure 4 (e): Some Activation Functions and their properties

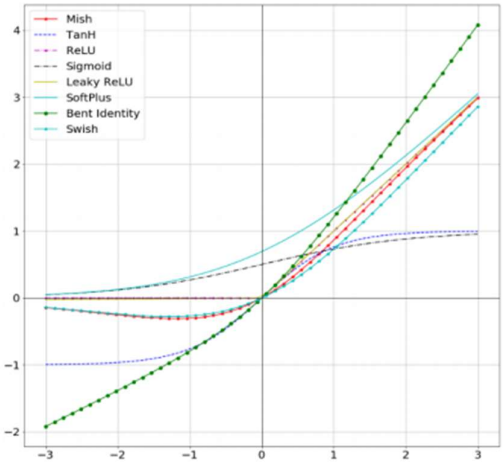


Figure 4 (f): Characteristics Comparison of Mish and Swish with other Activation functions

3. **Training:** CNNs are trained using the backpropagation algorithm with an optimization technique like stochastic gradient descent (SGD) to minimize a loss function. The weights of the filters and the parameters of the network are adjusted during training to improve the model's performance.

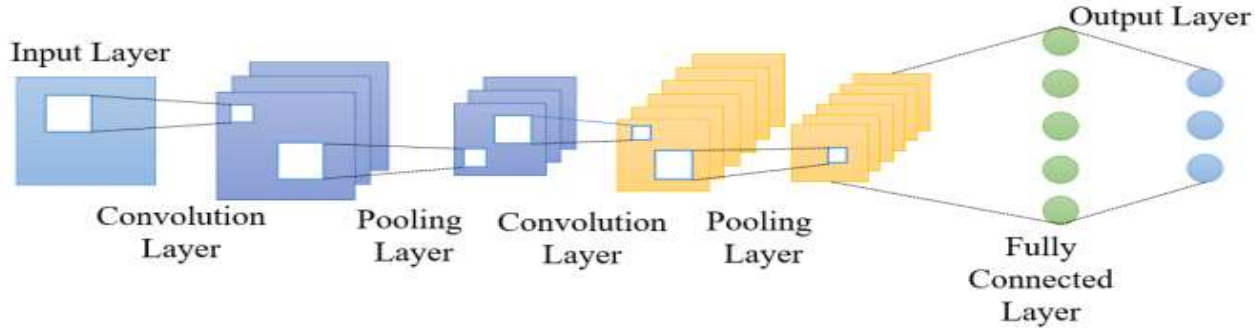


Figure 4 (g): Convolutional Neural Network

- **Algorithm for CNN**

The algorithm for Convolutional Neural Networks (CNNs) can be summarized as follows:

a. **Input:** Receive an input image or a batch of images. Each image is represented as a matrix of pixel values.

b. **Convolutional Layer:** Apply convolutional filters (kernels) to the input image to extract features. Each filter performs a convolution operation, resulting in a feature map that highlights certain features in the input image.

c. **Activation Function:** Apply a non-linear activation function (such as ReLU) to introduce non-linearity into the model and allow it to learn complex patterns in the data.

d. **Pooling Layer**: Use a pooling operation (such as max pooling or average pooling) to reduce the spatial dimensions of the feature map while retaining important information. This helps in reducing the computational complexity of the network and controlling overfitting.

e. **Repeat Steps 2-4:** Repeat the convolutional and pooling layers multiple times to extract higher-level features from the input image.

f. **Flattening:** Flatten the final feature map into a one-dimensional vector to prepare it for input into the fully connected layers.

g. **Fully Connected Layers:** Use one or more fully connected layers to perform classification based on the features extracted by the convolutional layers. Each neuron in the fully connected layer is connected to every neuron in the previous layer.

h. **Output:** Use an appropriate activation function (such as SoftMax for classification) to obtain the final output of the network, which represents the predicted class probabilities for the input image.

i. **Training:** Train the CNN using backpropagation and an optimization algorithm (such as stochastic gradient descent) to minimize a loss function. Adjust the weights of the filters and the parameters of the network to improve its performance.

j. **Prediction:** Use the trained CNN to predict the class label of new images by passing them through the network and obtaining the output probabilities.

This algorithm is a high-level overview of how CNNs work. The specific implementation details, such as the number of layers, the size of the filters, and the choice of activation functions, can vary depending on the specific architecture and task.

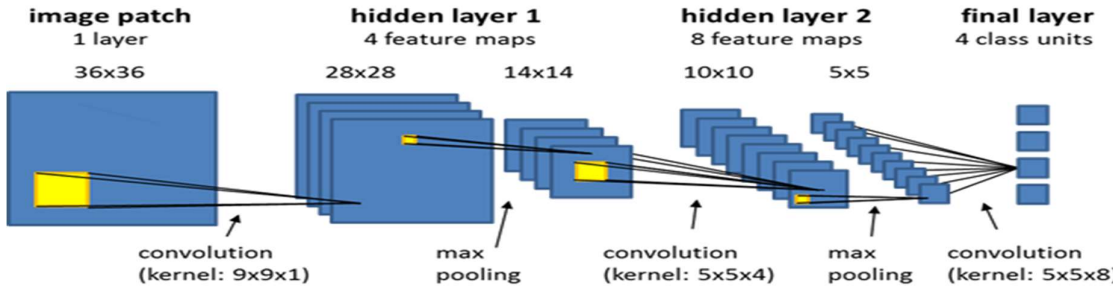


Figure 4 (h): Visual Representation of Convolutional Neural Network

- **Some common regularization techniques used in CNNs**

Here are some common regularization techniques used in Convolutional Neural Networks (CNNs):

a. **Dropout:** Dropout randomly sets a fraction of input units to zero during training, which helps prevent overfitting by reducing interdependencies between neurons. It forces the network to learn more robust features.

b. **Weight Decay (L2 Regularization):** Weight decay adds a regularization term to the loss function, penalizing large weights. This encourages the model to learn simpler patterns and prevents overfitting by discouraging complex models.

c. **Batch Normalization:** Batch normalization normalizes the activations of each layer across the mini-batch, stabilizing and accelerating the training process. It helps prevent overfitting by reducing internal covariate shift and acts as a regularizer.

d. **Data Augmentation:** Data augmentation artificially increases the size of the training dataset by applying transformations such as rotation, flipping, scaling, cropping, or adding noise to the input images. It introduces variability into the training data, helping the model generalize better to unseen examples.

e. **Early Stopping:** Early stopping stops the training process when the performance on a validation dataset starts to degrade, preventing the model from overfitting to the training data.

f. **Drop Connect:** Drop Connect is an extension of dropout where instead of dropping input units, it randomly drops connections between neurons within the network. It helps regularize the model by reducing the capacity of individual neurons.

g. **Data Normalization:** Data normalization scales input data to a standard range (e.g., [0, 1] or [-1, 1]), helping stabilize the training process and improve convergence.

h. **Sparse Regularization:** Sparse regularization encourages sparsity in the model by adding penalties for non-zero weights, leading to a simpler and more interpretable model.

i. **Data Dropout:** Data dropout randomly removes features or channels from input data during training, similar to dropout for neurons. It encourages the model to learn from more diverse features and improves generalization.

These regularization techniques help prevent overfitting, improve model generalization, and enhance the robustness and performance of CNNs on unseen data.

- **Advantages of Convolutional Neural Networks (CNNs)**

a. **Feature Learning:** CNNs can automatically learn hierarchical representations of features from raw data, eliminating the need for manual feature extraction.

b. **Spatial Hierarchies:** They can capture spatial hierarchies of features in images, learning to detect low-level features like edges and textures, and then combining them to detect higher-level features like objects or faces.

c. **Translation Invariance:** CNNs are able to recognize objects in images regardless of their position, making them robust to translation.

d. **Parameter Sharing:** By using shared weights in convolutional layers, CNNs are more efficient in terms of memory and computation compared to fully connected networks.

e. **Effective in High-Dimensional Data:** CNNs perform well in tasks with high-dimensional data, such as image and video processing.

f. **Transfer Learning**: Pre-trained CNN models can be used as a starting point for new tasks, reducing the need for large amounts of labeled data and computation resources.

- **Disadvantages of Convolutional Neural Networks (CNNs)**

a. **Computationally Intensive:** Training and using CNNs can be computationally intensive, requiring powerful hardware, especially for large datasets and complex models.

b. **Overfitting:** CNNs can be prone to overfitting, especially with small datasets or overly complex models. Regularization techniques like dropout are often used to mitigate this issue.

c. **Interpretability:** The internal workings of CNNs can be difficult to interpret, making it challenging to understand how decisions are made, especially in complex models.

d. **Data Requirements:** CNNs require large amounts of labeled training data to perform well, which may not always be available or easy to obtain.

e. **Hyperparameter Sensitivity:** CNNs have many hyperparameters that need to be tuned, such as the number of layers, filter sizes, and learning rates, which can be challenging and time-consuming.

f. **Limited Contextual Understanding:** While CNNs are effective at recognizing patterns in local regions of an image, they may struggle with tasks that require understanding broader contextual information.

- ## Applications

Convolutional Neural Networks (CNNs) have a wide range of applications across various domains due to their ability to effectively process and analyze visual data. Some common applications [19] [20] of CNNs include:

a. **Image Classification:** CNNs are widely used for image classification tasks, such as identifying objects in images. They can classify images into predefined categories, such as recognizing different species of animals or types of vehicles.

b. **Object Detection:** CNNs can also be used for object detection, where they not only classify objects but also locate them within an image. This is useful in applications such as autonomous driving, where vehicles need to detect pedestrians and other vehicles.

c. **Facial Recognition:** CNNs are used in facial recognition systems to identify and verify individuals based on their facial features. This technology is used in security systems, access control, and photo tagging applications.

d. **Medical Image Analysis:** CNNs are used in medical imaging for tasks such as detecting tumors in MRI scans, classifying skin lesions in dermatology, and analyzing retinal images for diabetic retinopathy.

e. **Video Analysis:** CNNs can analyze video data for tasks such as action recognition, object tracking, and video classification. This is useful in surveillance, sports analysis, and video recommendation systems.

f. **Natural Language Processing (NLP):** While primarily used for image processing, CNNs can also be applied to NLP tasks such as text classification and sentiment analysis, where the input is represented as a sequence of words.

g. **Autonomous Vehicles:** CNNs are used in autonomous vehicles for tasks such as detecting traffic signs, pedestrians, and other vehicles, as well as for lane detection and path planning.

4.3. References

1. M.I. Jordan and T.M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255-260, 2015.

2. B. Mahesh, "Machine learning algorithms-a review*," International Journal of Science and Research (IJSR)*, vol. 9, no. 1, pp. 381-386, 2020.

3. H. Luan and C.C. Tsai, "A review of using machine learning approaches for precision education," *Educational Technology & Society*, vol. 24, no. 1, pp. 250-266, 2021.

4. Y. Singh, P.K. Bhatia, and O. Sangwan, "A review of studies on machine learning techniques," *International Journal of Computer Science and Security*, vol. 1, no. 1, pp. 70-84, 2007.

5. M. Bkassiny, Y. Li, and S.K. Jayaweera, "A survey on machine-learning techniques in cognitive radios," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1136-1159, 2012.

6. B.K. Bhavitha, A.P. Rodrigues, and N.N. Chiplunkar, "Comparative study of machine learning techniques in sentimental analysis," *in 2017 International conference on inventive communication and computational technologies (ICICCT),* pp. 216-221, IEEE, Mar. 2017.

7. D.T. Pham and A.A. Afify, "Machine-learning techniques and their applications in manufacturing," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 219, no. 5, pp. 395-412, 2005.

8. S.B. Kotsiantis, I. Zaharakis, and P. Pintelas, "Supervised machine learning: A review of classification techniques," *Emerging artificial intelligence applications in computer engineering*, vol. 160, no. 1, pp. 3-24, 2007.

9. S. Omar, A. Ngadi, and H.H. Jebur, "Machine learning techniques for anomaly detection: an overview*," International Journal of Computer Applications*, vol. 79, no. 2, 2013.

10. V. Nasteski, "An overview of the supervised machine learning methods," *Horizons. b*, vol. 4, pp. 51-62, 2017.

11. N. Burkart and M.F. Huber, "A survey on the explainability of supervised machine learning," *Journal of Artificial Intelligence Research*, vol. 70, pp. 245-317, 2021.

12. A. Singh, N. Thakur, and A. Sharma, "A review of supervised machine learning algorithms," *in 2016 3rd international conference on computing for sustainable global development (INDIACom*), pp. 1310-1315, IEEE, Mar. 2016.

13. Muhammad and Z. Yan, "SUPERVISED MACHINE LEARNING APPROACHES: A SURVEY," *ICTACT Journal on Soft Computing*, vol. 5, no. 3, 2015.

14. F.Y. Osisanwo, J.E.T. Akinsola, O. Awodele, J.O. Hinmikaiye, O. Olakanmi, and J. Akinjobi, "Supervised machine learning algorithms: classification and comparison," *International Journal of Computer Trends and Technology (IJCTT)*, vol. 48, no. 3, pp. 128-138, 2017.

15. A.E. Mohamed, "Comparative study of four supervised machine learning techniques for classification," *International Journal of Applied*, vol. 7, no. 2, pp. 1-15, 2017.

16. S. Minaee et al., "Deep learning--based text classification: a comprehensive review," *ACM computing surveys (CSUR)*, vol. 54, no. 3, pp. 1-40, 2021.

17. J. Wu, "Introduction to convolutional neural networks," *National Key Lab for Novel Software Technology*. Nanjing University. China, vol. 5, no. 23, p. 495, 2017.

18. S. Albawi, T.A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," *in 2017 international conference on engineering and technology (ICET*), pp. 1-6, IEEE, Aug. 2017.

19. Z. Li et al., "A survey of convolutional neural networks: analysis, applications, and prospects," *IEEE transactions on neural networks and learning systems*, vol. 33, no. 12, pp. 6999-7019, 2021.

20. Montesinos López, O.A., Montesinos López, A., and Crossa, J., "Convolutional neural networks," *in Multivariate statistical machine learning methods for genomic prediction,* pp. 533-577, Cham, Springer International Publishing, 2022.

CHAPTER 5

Cognitive Load Analysis

5.1 Introduction

Cognitive load theory incorporates principles from evolutionary theory to understand human cognitive architecture and to devise innovative instructional methods. The theory posits that our knowledge can be categorized into two types: biologically primary knowledge, which we instinctively acquire through evolution, and biologically secondary knowledge, which is culturally significant and forms the focus of instructional efforts. Secondary knowledge is processed in a manner akin to how biological evolution processes information. When dealing with secondary knowledge, human cognition relies on a substantial information store, primarily built by acquiring knowledge from existing sources. Generating new information involves a process resembling trial and error, with limited capacity to process novel information concurrently. However, our cognitive system can effectively process vast amounts of organized information stored in the knowledge base to facilitate the execution of complex tasks. This cognitive framework informs the development of instructional strategies aimed at enhancing learning outcomes.

5.2 History

The origins of cognitive load theory can be traced back to the early days of cognitive science in the 1950s, notably with the groundbreaking work of G.A. Miller. In his seminal paper, Miller was among the first to propose that there are limitations to our working memory capacity, suggesting that humans can typically hold only a limited number of items in short-term memory, around seven plus or minus two.

In 1973, Simon and Chase introduced the concept of "chunking" to explain how individuals organize information in short-term memory. This idea of chunking, akin to constructing mental schemas, became pivotal in understanding how people manage information.

John Sweller further advanced cognitive load theory in the late 1980s, particularly in the study of problem-solving. His research highlighted that learners often employ a strategy called means-ends analysis, which requires significant cognitive processing capacity that could otherwise be utilized for schema construction. Sweller proposed that instructional designers should mitigate this unnecessary cognitive load by designing instructional materials that minimize the need for extensive problem-

solving. Examples of such materials include worked examples and goal-free problems.Throughout the 1990s, cognitive load theory gained traction in various domains, leading to the identification of several learning phenomena, such as the completion-problem effect.

5.3 What is Cognitive Load?

In cognitive psychology, cognitive load refers to how much mental effort is used, but it's important to distinguish it from Cognitive Load (CL) or Mental Workload (MWL), which are studied across various disciplines. In instructional design and pedagogy, three types of cognitive load are recognized: intrinsic (effort related to the topic), extraneous (due to how information is presented), and germane (related to building understanding). However, researchers have found that these types of loads don't just add up independently; they interact in complex ways.

Cognitive load theory emerged in the late 1980s from John Sweller's research on problem-solving, suggesting that instructional design can lessen the mental effort learners expend. Later, researchers developed methods to measure perceived mental effort, like task-invoked pupillary response, which correlates with working memory. Learning relies on information passing through working memory into long-term memory, but working memory has limited capacity and duration, which can hinder learning under certain conditions. High cognitive load can negatively impact task performance, with different groups, such as the elderly, students, and children, experiencing varying levels of cognitive load.

Cognitive load refers to the amount of mental effort or resources required to perform a task or process information. It is a concept derived from cognitive psychology and refers to the burden placed on working memory during cognitive tasks. Working memory has limited capacity and can become overwhelmed when too much information needs to be processed simultaneously. The core idea of cognitive load theory is that better instructional design considers the role and limitations of working memory, aiming to enhance learning outcomes. With increasing distractions, like cell phone use, students are more likely to experience high cognitive load, potentially reducing academic success.

5.4 Types of Cognitive Load

Cognitive load theory identifies three main types of cognitive load: intrinsic, extraneous, and germane.

1. Intrinsic Cognitive Load

This type of load is inherent to the complexity of the learning task itself. It's determined by the difficulty of the material being learned. For example, solving a simple math problem would have a lower intrinsic cognitive load compared to solving a complex calculus equation. Intrinsic cognitive load is relatively fixed and cannot be directly manipulated by instructional designers.

2. Extraneous Cognitive Load

Extraneous cognitive load refers to the mental effort required to process the instructional materials or the presentation format. It arises from how the information is presented to learners and is influenced by factors such as the clarity of instructions, the organization of information, or the complexity of multimedia elements. Instructional designers have control over extraneous cognitive load and can reduce it by optimizing the design of instructional materials to minimize unnecessary cognitive effort, thus allowing learners to focus more effectively on the learning task itself.

3. Germane Cognitive Load

Germane cognitive load is related to the mental effort involved in processing and constructing meaningful schemas or mental models. It encompasses the cognitive processing that leads to the formation of long-term memory and the integration of new information with existing knowledge. Unlike intrinsic load, which is inherent to the task, and extraneous load, which is influenced by instructional design, germane load is directed towards productive learning processes. Instructional designers can facilitate germane load by providing opportunities for learners to engage in activities that promote deeper understanding, critical thinking, and schema development.

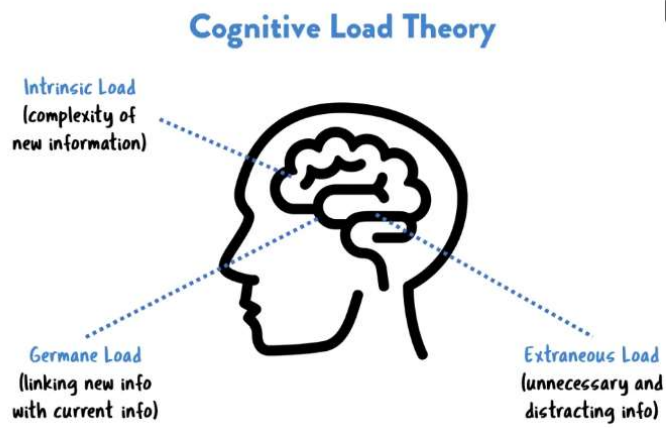


Figure 5(a). Cognitive Load theory

5.5 Human Cognitive Architecture

Human cognitive architecture refers to the underlying structure and organization of cognitive processes in the human mind. It encompasses the various mental systems and mechanisms that enable perception, attention, memory, language, reasoning, problem-solving, decision-making, and other cognitive functions. Understanding human cognitive architecture involves elucidating how these different cognitive processes interact and operate within the brain to generate behavior and cognition.

Key components of human cognitive architecture include:

1. **Sensory Systems:** Sensory systems such as vision, audition, touch, taste, and smell gather information from the environment and transmit it to the brain for processing. These sensory inputs serve as the foundation for perception and awareness.

2. **Attention:** Attentional mechanisms selectively filter and focus on relevant sensory information while ignoring irrelevant stimuli. Attention plays a crucial role in guiding perception, prioritizing information processing, and directing cognitive resources towards specific tasks or goals.

3. **Working Memory:** Working memory is a temporary storage system that holds and manipulates information during cognitive tasks. It allows individuals to maintain and process relevant information in real-time, enabling activities such as problem-solving, reasoning, and decision-making.

4. **Long-Term Memory:** Long-term memory stores vast amounts of information acquired over time, including facts, concepts, experiences, and skills. It is subdivided into declarative memory (knowledge of facts and events) and procedural memory (knowledge of how to perform tasks and skills).

5. **Language Processing:** Language processing involves the comprehension, production, and interpretation of linguistic information. It encompasses various cognitive processes, including phonological processing, semantic understanding, syntactic analysis, and discourse comprehension.

6. **Executive Functions:** Executive functions are higher-order cognitive processes that govern goal-directed behavior, self-regulation, and cognitive control. They include abilities such as planning, inhibition, cognitive flexibility, working memory updating, and decision-making.

7. **Perceptual-Motor Processes:** Perceptual-motor processes integrate sensory information with motor responses to guide action and behavior. They involve sensorimotor coordination, motor planning, motor execution, and feedback processing.

8. **Emotion and Motivation:** Emotion and motivation influence cognitive processes by modulating attention, memory, decision-making, and behavior. Emotions provide affective valence to stimuli, whereas motivation directs goal-directed actions and drives behavior towards desired outcomes.

9. **Social Cognition:** Social cognition refers to the cognitive processes involved in understanding and interacting with others. It encompasses abilities such as theory of mind (understanding others' thoughts and intentions), empathy, social perception, and social decision-making.

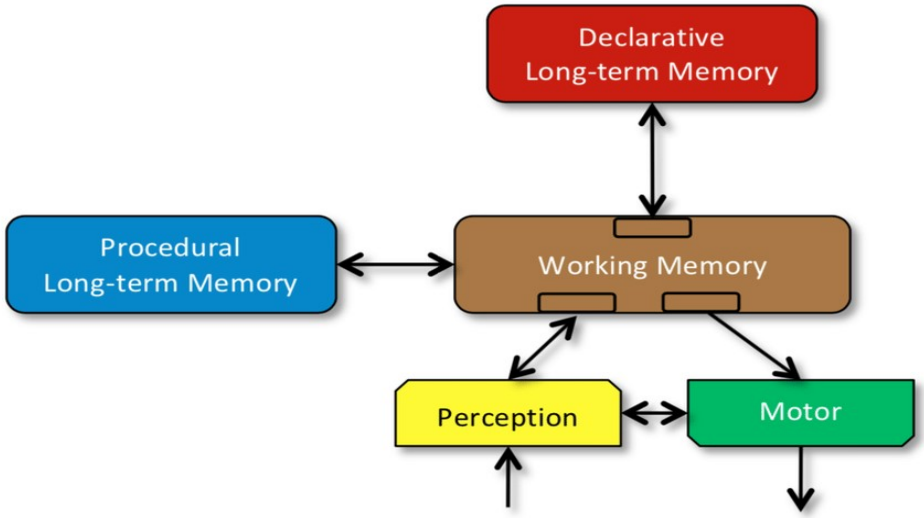


Figure 5(b): Standard model of Cognitive Architecture

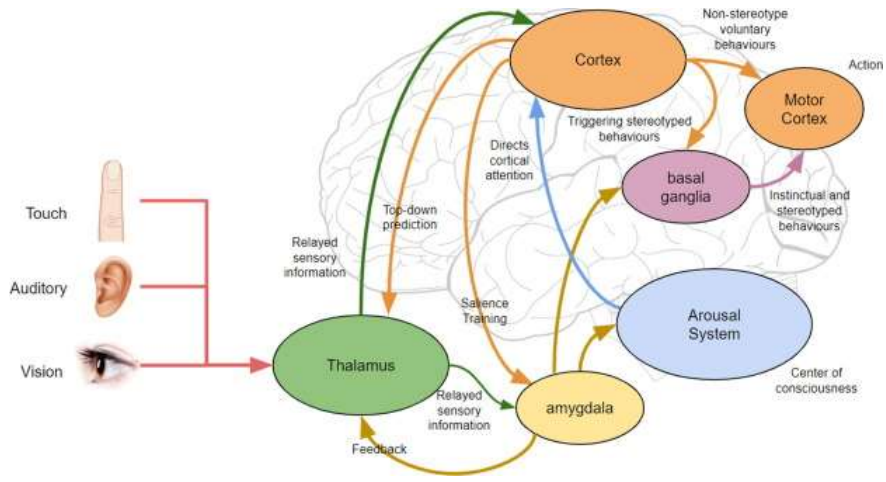


Figure 5(c): Human Brain-inspired distributed cognitive architecture

5.6 Measurement of Cognitive Load

The measurement of cognitive load is crucial in understanding how individuals process information and learn effectively. Various methods have been developed to assess cognitive load, ranging from self-report questionnaires to neurophysiological measures. Here's a detailed explanation of some common approaches:

1. Self-Report Questionnaires

Self-report questionnaires are a widely used method for assessing cognitive load, as they provide a direct way for individuals to express their subjective experience of mental effort. Participants are typically asked to rate their perceived mental effort or difficulty during a learning task using Likert scales or other rating systems. These questionnaires often inquire about aspects such as mental effort, difficulty, frustration, or perceived workload. While self-report questionnaires offer valuable insights into learners' subjective experiences, they may be influenced by factors such as response biases or individual differences in self-awareness.

2. Dual-Task Paradigms

Dual-task paradigms involve performing a primary task while simultaneously completing a secondary task that interferes with cognitive resources. By measuring performance on both tasks, researchers can infer the cognitive load imposed by the primary task. For example, participants might be asked to complete a problem-solving task while simultaneously performing a secondary task such as verbal recitation or pattern recognition. The performance decrement on the secondary task reflects the cognitive load imposed by the primary task.

3. Physiological Measures

Physiological measures provide objective indicators of cognitive load by monitoring changes in physiological responses associated with mental effort and arousal. Common physiological measures include heart rate variability, electrodermal activity, pupillometry, and electroencephalography (EEG). For instance, increases in heart rate or pupil dilation are indicative of heightened cognitive load, as

they reflect changes in autonomic arousal and sympathetic nervous system activity. EEG can also capture neural correlates of cognitive load by measuring brainwave activity associated with attention, memory, and cognitive processing.

4. Performance Measures

Performance-based measures assess task performance as an indirect indicator of cognitive load. Poor performance or errors on a task may suggest that the cognitive demands exceed the individual's capacity, leading to cognitive overload. Conversely, improvements in performance with practice or instructional interventions may indicate a reduction in cognitive load. Performance measures can include accuracy, response time, error rates, and learning curves.

5. Objective Task Analysis

Objective task analysis involves breaking down a learning task into its component parts and quantifying the cognitive demands associated with each part. This approach allows researchers to systematically evaluate the cognitive load imposed by different aspects of the task, such as information processing, decision-making, and memory retrieval. Objective task analysis can inform instructional design by identifying areas of high cognitive load that may require simplification or support.

6. Electroencephalography (EEG) Connectivity Analysis

EEG measures electrical activity in the brain using electrodes placed on the scalp. In addition to analyzing individual EEG signals, researchers can examine functional connectivity patterns between different brain regions to assess cognitive load. Connectivity analysis identifies synchronized activity between brain areas involved in cognitive processes such as attention, memory, and decision-making. Alterations in connectivity patterns may indicate changes in cognitive load and cognitive resource allocation.

7. Behavioral Observation and Task Analysis

Behavioral observation involves systematically recording and analyzing observable behaviors during cognitive tasks, such as problem-solving strategies, verbalizations, and task completion times. Task analysis breaks down complex tasks into discrete steps and quantifies the cognitive demands associated with each step. By observing how individuals approach and complete tasks, researchers can identify cognitive load indicators such as strategy shifts, errors, and task completion times.

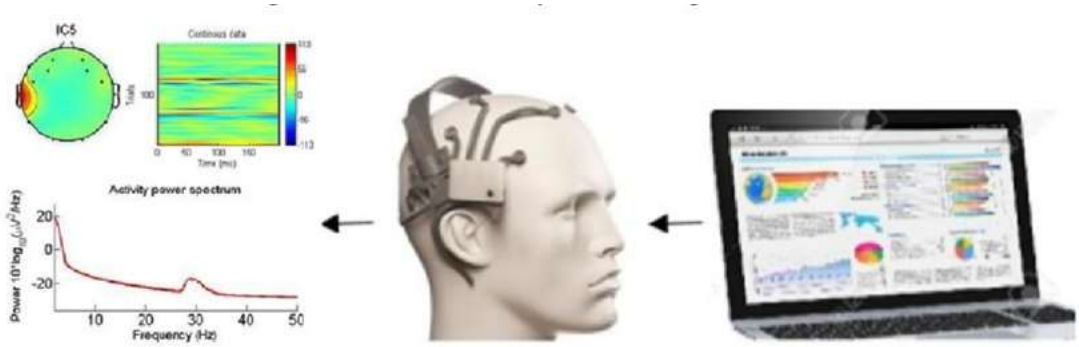


Figure 5(d) : Experimental Set up to measure Cognitive Load

Overall, the measurement of cognitive load involves a combination of subjective and objective methods to capture both the perceived experience and the objective demands of cognitive tasks. By employing a multi-method approach, researchers can gain a comprehensive understanding of cognitive load and its implications for learning and performance.

5.7 Effects of Heavy Cognitive Load

When faced with a heavy cognitive load, individuals often experience errors or interference in the task they are performing. This cognitive burden can lead to the activation of stereotypes and increase the likelihood of making stereotypical judgments. This is because heavy cognitive loads push excess information into subconscious processing, where schemas—patterns of thought and behavior—are utilized to organize information into categories and identify relationships between them. Consequently, automatic activation of stereotypical associations occurs through pattern recognition and schema utilization, resulting in an implicit stereotype effect.

The concepts of cognitive load and arousal contribute to the "Overload Hypothesis" explanation of social facilitation. In situations where individuals are observed by an audience, their performance tends to worsen in tasks perceived as subjectively complex, while it improves in tasks seen as subjectively easy. This suggests that the presence of an audience intensifies cognitive load and arousal, which can impair performance in demanding tasks but enhance it in simpler ones.

Heavy cognitive load can have various effects on individuals' cognitive functioning, behavior, and performance:

- **Impaired Task Performance:** Heavy cognitive load can lead to decreased performance on cognitive tasks, particularly those that require complex problem-solving, decision-making, or information processing. When cognitive resources are overwhelmed, individuals may struggle to maintain focus, make errors, or experience difficulty in completing tasks accurately and efficiently.

- **Increased Errors and Interference:** High cognitive load can result in increased errors and interference in task performance. When cognitive resources are overloaded, individuals may have difficulty filtering out irrelevant information, inhibiting irrelevant responses, or maintaining attention on task-relevant stimuli, leading to mistakes and interference from irrelevant information.

- **Impaired Decision-Making:** Cognitive overload can impair decision-making abilities by hindering individuals' ability to weigh options, consider alternatives, and make informed choices. Under heavy cognitive load, individuals may rely on heuristic decision-making strategies or make impulsive decisions without fully considering the consequences.

- **Reduced Creativity and Innovation:** Cognitive overload can suppress creativity and innovation by limiting individuals' ability to generate novel ideas, think flexibly, or engage in divergent thinking. When cognitive resources are depleted, individuals may struggle to explore alternative solutions or approaches to problems, resulting in rigid thinking and reduced creativity.

- **Increased Stress and Fatigue**: Heavy cognitive load can contribute to increased stress and mental fatigue, as individuals expend additional effort to process information and maintain cognitive performance. Prolonged cognitive overload can lead to feelings of exhaustion, irritability, and burnout, impacting overall well-being and productivity.

- **Impaired Learning and Memory:** Excessive cognitive load can hinder learning and memory consolidation by overwhelming working memory capacity and inhibiting encoding processes. When cognitive resources are strained, individuals may have difficulty encoding new information into long-term memory or retrieving stored information, impairing learning retention and recall.

5.8 Sub-Population Studies

The effects of heavy cognitive load can have significant implications for various sub-populations, such as individuals with cognitive impairments, neurodiverse individuals, or those experiencing stress or anxiety. Understanding how cognitive load impacts different sub-populations can inform interventions and support strategies tailored to their specific needs and challenges. Research on the effects of heavy cognitive load within sub-populations reveals insights into how various groups may be affected differently:

- **Children and Adolescents**: Children and adolescents may be particularly susceptible to the effects of heavy cognitive load due to their developing cognitive abilities and limited attentional resources. Studies show that excessive cognitive demands can impair academic performance, increase stress levels, and affect learning outcomes in this population. Strategies such as scaffolding, chunking information, and providing explicit instructions can help alleviate cognitive load and support learning in educational settings.

- **Elderly Individuals:** Older adults may experience greater challenges in managing heavy cognitive loads due to age-related declines in cognitive function, such as working memory capacity and processing speed. Heavy cognitive load can exacerbate cognitive fatigue, increase cognitive errors, and impair decision-making in older adults. Interventions focused on

simplifying tasks, providing clear instructions, and reducing distractions can help mitigate cognitive load and support older adults in daily activities.

- **Individuals with Cognitive Impairments:** Individuals with cognitive impairments, such as dementia or traumatic brain injury, may be especially vulnerable to the effects of heavy cognitive load. Heavy cognitive demands can exacerbate cognitive deficits, impair functional abilities, and increase caregiver burden in this population. Tailored interventions, including cognitive rehabilitation, environmental modifications, and caregiver support, are essential for managing cognitive load and enhancing quality of life in individuals with cognitive impairments.

- **Individuals with Mental Health Conditions:** Individuals with mental health conditions, such as anxiety disorders or depression, may experience heightened cognitive load due to intrusive thoughts, rumination, and emotional distress. Heavy cognitive demands can exacerbate symptoms, impair concentration and decision-making, and interfere with daily functioning in individuals with mental health conditions. Cognitive-behavioral therapies, mindfulness techniques, and stress management strategies can help individuals with mental health conditions cope with cognitive load and improve cognitive functioning.

Overall, understanding how heavy cognitive load affects different sub-populations is essential for developing targeted interventions and support strategies to mitigate its adverse effects and promote cognitive well-being across diverse populations.

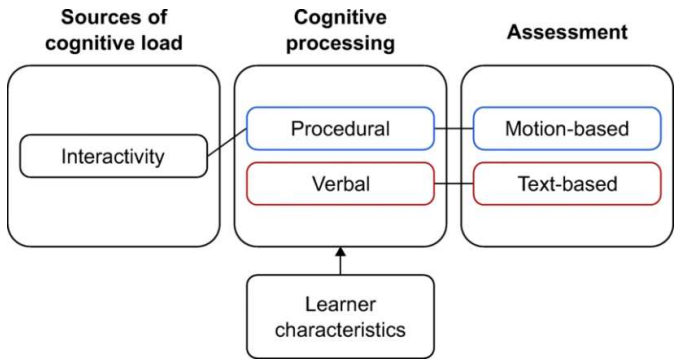


Figure 5(e) : Analysis of Cognitive Load

5.9 Advantages of Cognitive Load Analysis

Cognitive load analysis offers several advantages in understanding and optimizing cognitive performance and learning experiences:

- **Enhanced Learning Outcomes:** By identifying and managing cognitive load, cognitive load analysis helps optimize learning experiences, leading to improved learning outcomes. By minimizing extraneous cognitive load and promoting germane cognitive load, instructional materials can be designed to facilitate better understanding, retention, and transfer of knowledge.

- **Effective Instructional Design:** Cognitive load analysis provides insights into the cognitive demands imposed by instructional materials and tasks, allowing instructional designers to create more effective and engaging learning experiences. By aligning instructional materials with learners' cognitive capabilities and limitations, designers can enhance instructional effectiveness and promote deeper learning.

- **Improved Task Performance:** Understanding the cognitive demands of tasks through cognitive load analysis enables the optimization of task design to support better performance outcomes. By reducing cognitive overload and streamlining task complexity, individuals can perform tasks more efficiently, accurately, and with less cognitive effort.

- **Enhanced User Experience:** By considering cognitive load during the design of interfaces, products, or systems, cognitive load analysis helps improve the user experience. By minimizing cognitive demands, reducing cognitive fatigue, and enhancing usability, designers can create products and interfaces that are more intuitive, user-friendly, and enjoyable to use.

- **Evidence-Based Decision Making:** Cognitive load analysis provides empirical data and insights to inform decision-making in instructional design, task design, and product development. By grounding design decisions in research-based principles of cognitive load theory, designers can make informed choices that are more likely to result in positive outcomes for users and learners.

- **Continuous Improvement:** Cognitive load analysis supports iterative design processes, allowing for ongoing refinement and optimization of instructional materials, tasks, and interfaces. By collecting feedback, monitoring performance, and adjusting design elements based on cognitive load considerations, designers can iteratively improve the effectiveness, efficiency, and usability of their creations over time.

5.10 Challenges in Cognitive Load Analysis

While cognitive load analysis offers valuable insights into understanding and optimizing cognitive performance, it also presents several challenges:

- **Limited Objective Measures**: While subjective measures provide valuable insights into individuals perceived cognitive load, objective measures of cognitive load, such as physiological indicators or performance metrics, have limitations. Physiological measures, such as heart rate or pupil dilation, may be influenced by factors other than cognitive load, such as arousal or emotional states. Performance metrics, while objective, may not capture the full complexity of cognitive processes involved in task performance.

- **Contextual Variability**: Cognitive load is influenced by various contextual factors, including task characteristics, environmental conditions, and individual differences. Analyzing cognitive load in different contexts requires consideration of these factors, which can complicate interpretation and generalization of findings. What may constitute a heavy cognitive load in one context may not be applicable in another, making it challenging to establish universal guidelines or benchmarks for cognitive load assessment.

- **Dynamic Nature of Cognitive Load:** Cognitive load is dynamic and can fluctuate over time based on task demands, individual factors, and environmental conditions. Analyzing cognitive load requires capturing these dynamic changes and understanding how cognitive load evolves throughout task execution. However, existing methods may lack sensitivity to detect subtle changes in cognitive load or may not account for temporal dynamics adequately.

5.11 Applications

Cognitive load analysis has numerous applications across various domains, including education, training, human-computer interaction, healthcare, and beyond. Here are some key applications:

- **Educational/Instructional Design:** Cognitive load analysis informs the design and development of instructional materials, curriculum, and teaching methods to optimize learning outcomes. By understanding the cognitive demands of learning tasks, instructional designers can create learning experiences that effectively manage cognitive load, promote deeper learning, and enhance knowledge retention.

- **Training and Skill Development:** In corporate training, military training, and professional development contexts, cognitive load analysis is used to design training programs that maximize learning efficiency and skill acquisition. By identifying cognitive bottlenecks and optimizing training content and delivery methods, organizations can improve training effectiveness and performance outcomes.

- **Human-Computer Interaction (HCI):** In HCI research and design, cognitive load analysis helps designers create user interfaces and interactive systems that are intuitive, user-friendly, and supportive of cognitive processing. By minimizing cognitive load and maximizing usability, designers can enhance user experience, productivity, and satisfaction with technology products and interfaces.

- **Healthcare and Patient Safety:** Cognitive load analysis is applied in healthcare settings to optimize clinical workflows, decision support systems, and patient safety initiatives. By understanding the cognitive demands of clinical tasks and information processing, healthcare providers can design interventions that reduce cognitive errors, improve diagnostic accuracy, and enhance patient outcomes.

- **Driver Monitoring Systems:** Cognitive load analysis informs the development of driver monitoring systems that assess drivers' cognitive states, workload, and attention levels in real

time. By integrating physiological sensors, eye-tracking technology, and machine learning algorithms, these systems can detect signs of cognitive fatigue, distraction, or inattention, and provide timely alerts or adaptive interventions to support safe driving behavior.

- **Automation and Autonomous Vehicles:** In the context of automation and autonomous vehicles, cognitive load analysis is used to assess drivers' cognitive engagement and workload during supervisory driving tasks. By understanding how drivers interact with automated driving systems and manage transitions between manual and autonomous modes, designers can optimize system interfaces and interaction protocols to facilitate safe and effective human-machine collaboration.

In summary, cognitive load analysis plays a crucial role in improving the design of driver assistance systems, enhancing driver training and education, and promoting safe driving behavior in the automotive industry. By considering the cognitive demands of driving tasks and environments, designers can develop technologies and interventions that prioritize driver attention, workload management, and situational awareness, ultimately leading to safer and more efficient transportation systems.

5.12 References

1. J. Leppink, F. Paas, C.P. Van der Vleuten, T. Van Gog, and J.J. Van Merriënboer, "Development of an instrument for measuring different types of cognitive load," *Behavior Research Methods*, vol. 45, pp. 1058-1072, 2013.

2. Korbach, R. Brünken, and B. Park, "Differentiating different types of cognitive load: A comparison of different measures," *Educational Psychology Review*, vol. 30, no. 2, pp. 503-529, 2018.

3. P. Antonenko, F. Paas, R. Grabner, and T. Van Gog, "Using electroencephalography to measure cognitive load," *Educational Psychology Review*, vol. 22, pp. 425-438, 2010.

4. L. Cabañero, R. Hervás, I. González, J. Fontecha, T. Mondéjar, and J. Bravo, "Analysis of cognitive load using EEG when interacting with mobile devices," *in Proceedings*, vol. 31, no. 1, pp. 70, MDPI, Nov. 2019.

5. L.C. Gómez, R. Hervás, I. Gonzalez, and V. Villarreal, "Studying the generalisability of cognitive load measured with EEG," *Biomedical Signal Processing and Control*, vol. 70, p. 103032, 2021.

6. S. Martin, "A critical analysis of the theoretical construction and empirical measurement of cognitive load," *in Cognitive Load Measurement and Application*, pp. 29-44, 2017.

7. S. Chen, J. Epps, and F. Chen, "A comparison of four methods for cognitive load measurement," *in Proceedings of the 23rd Australian Computer-Human Interaction Conference*, pp. 76-79, Nov. 2011.

8. R. Brunken, J.L. Plass, and D. Leutner, "Direct measurement of cognitive load in multimedia learning," *Educational Psychologist*, vol. 38, no. 1, pp. 53-61, 2003.

9. B.B. Morrison, B. Dorn, and M. Guzdial, "Measuring cognitive load in introductory CS: adaptation of an instrument," *in Proceedings of the tenth annual conference on International computing education research*, pp. 131-138, July 2014.

10. F. Chen et al., "Robust multimodal cognitive load measurement," *Cham, Springer*, pp. 13-32, 2016.

11. P. Schmutz, S. Heinz, Y. Métrailler, and K. Opwis, "Cognitive load in eCommerce applications—measurement and effects on user satisfaction," *Advances in Human-Computer Interaction*, 2009.

12. J.Q. Young and J.L. Sewell, "Applying cognitive load theory to medical education: construct and measurement challenges," *Perspectives on Medical Education*, vol. 4, pp. 107-109, 2015.

13. E. Haapalainen, S. Kim, J.F. Forlizzi, and A.K. Dey, "Psycho-physiological measures for assessing cognitive load," *in Proceedings of the 12th ACM International Conference on Ubiquitous Computing*, pp. 301-310, Sept. 2010.

14. F. Paas, J.E. Tuovinen, H. Tabbers, and P.W. Van Gerven, "Cognitive load measurement as a means to advance cognitive load theory," *in Cognitive Load Theory*, pp. 63-71, Routledge, 2016.

15. S. Martin, "Measuring cognitive load and cognition: metrics for technology-enhanced learning," *in Technology-Enhanced and Collaborative Learning*, pp. 77-106, Routledge, 2018.

16. B. Park and R. Brünken, "Secondary task as a measure of cognitive load," *in Cognitive Load Measurement and Application*, pp. 75-92, Routledge, 2017.

17. J. Leppink and A. van den Heuvel, "The evolution of cognitive load theory and its application to medical education," *Perspectives on Medical Education*, vol. 4, pp. 119-127, 2015.

CHAPTER 6

Cognitive Load Measurement using fNIRS

6.1 Abstract

This work investigates the feasibility and effectiveness of using functional Near-Infrared Spectroscopy (fNIRS) [1] as a method for measuring cognitive load. Cognitive load, the mental effort required to process information, is a critical factor in instructional design and learning. Traditional methods for assessing cognitive load have limitations, and fNIRS presents a promising alternative due to its ability to measure neural activity non-invasively and in real-time. We review the current state of research on cognitive load measurement using fNIRS, including experimental designs, data analysis techniques, and applications in various domains. We also discuss challenges and future directions for utilizing fNIRS in cognitive load research, such as improving signal quality, standardizing measurement protocols, and integrating fNIRS with other physiological and behavioral measures. Overall, fNIRS shows great potential as a tool for advancing our understanding of cognitive load and enhancing instructional design practices.

6.2 Introduction

Several studies have demonstrated that cognitive load affects eye movements, even in tasks where eye movements are not inherently involved. For instance, Collins (1962) [2] found that mental arithmetic tasks elicited stronger vestibular nystagmus responses compared to other tasks like estimating subjective velocity or reproducing durations of sound stimuli. Similarly, Greenberg (1970) [3] observed higher optokinetic nystagmus frequency during difficult mental arithmetic tasks compared to easier mental tasks.

Takeda (1976) [4] showed that mental activity during problem-solving tasks disrupted performance in fixation and eye tracking tasks. Subjects exhibited difficulties maintaining gaze or experienced disturbances in eye tracking while solving problems. Osaka and Koga (1982) [5] also found instability in eye fixation during cognitive tasks, attributing it to heightened central nervous system activity.

Tate (1979) [6] investigated the effects of cognitive tasks on saccade latency, expecting differential effects based on task difficulty and hemispheric activation. However, her results did not show significant differences in saccade latency between cognitive task conditions.

One possible explanation for the lack of significant findings in Tate's experiments could be that the mental tasks were not sufficiently challenging. In contrast, the studies by Takeda (1976) and Osaka and Koga (1982) utilized more demanding mental tasks.

In our experiment, we aim to further explore the relationship between cognitive load and visually elicited saccade latency by administering a set of mathematics questions.

6.3 System Overview

The cognitive load analysis system utilizing functional Near-Infrared Spectroscopy (fNIRS) begins by collecting data from the fNIRS system, capturing changes in blood oxygenation levels as participants engage in cognitive tasks. The subject wore a full-brain cap embedded with fNIRS sensors. The fNIRS system captured changes in oxygenated (HbO) and deoxygenated hemoglobin (Hb) concentrations, reflecting brain activity. Data was collected from 5 sessions, with each session comprising 5 distinct trials. First, data was recorded as high cognitive load (HCL), medium cognitive load (MCL) and low cognitive load (LCL), then transferred to a computer, generating a dataset in an Excel format. In this study, the fNIRS data was sampled at a frequency of 8 Hz using our recording device. Each experimental trial consisted of a fixation cross displayed for 3 seconds, a question period lasting 30 seconds, oral response period of 15 seconds and a rest period of 10 seconds. Based on this experimental protocol, the data was segmented from the raw dataset to analyze specific task-related brain activity. The data was sampled at 8 Hz, yielding 8 data points per second. Each segment was defined based on the timing of the experiment: Fixation cross (3 seconds) = 24 samples, Question period (30 seconds) = 240 samples and Rest period (10 seconds) = 80 samples. We have created new data frames for each question period to facilitate detailed analysis. This segmentation ensures that each data frame accurately represents the brain activity during each question period, allowing for precise analysis of the fNIRS data. Following preprocessing, feature extraction takes place using wavelet transform, allowing for the identification of temporal patterns in hemoglobin concentration changes associated with cognitive processes. These extracted features serve as inputs for the classification stage, where a Convolutional Neural Network (CNN) model is employed to classify the data into one of three cognitive load classes: high (HCL), medium (MCL), or low (LCL). The CNN model learns to differentiate between patterns corresponding to different levels of cognitive load, enabling accurate

classification. The system provides a comprehensive overview of cognitive load levels based on real-time fNIRS data, facilitating insights into cognitive performance during various tasks. Additionally, the system may incorporate a user interface for experiment setup, real-time visualization of fNIRS data, and continuous monitoring of cognitive load estimates, enhancing usability and accessibility for researchers and experimenters. Finally, the system allows for data storage and analysis, enabling researchers to further explore cognitive processes and their dynamics offline, contributing to advancements in cognitive science and related fields.

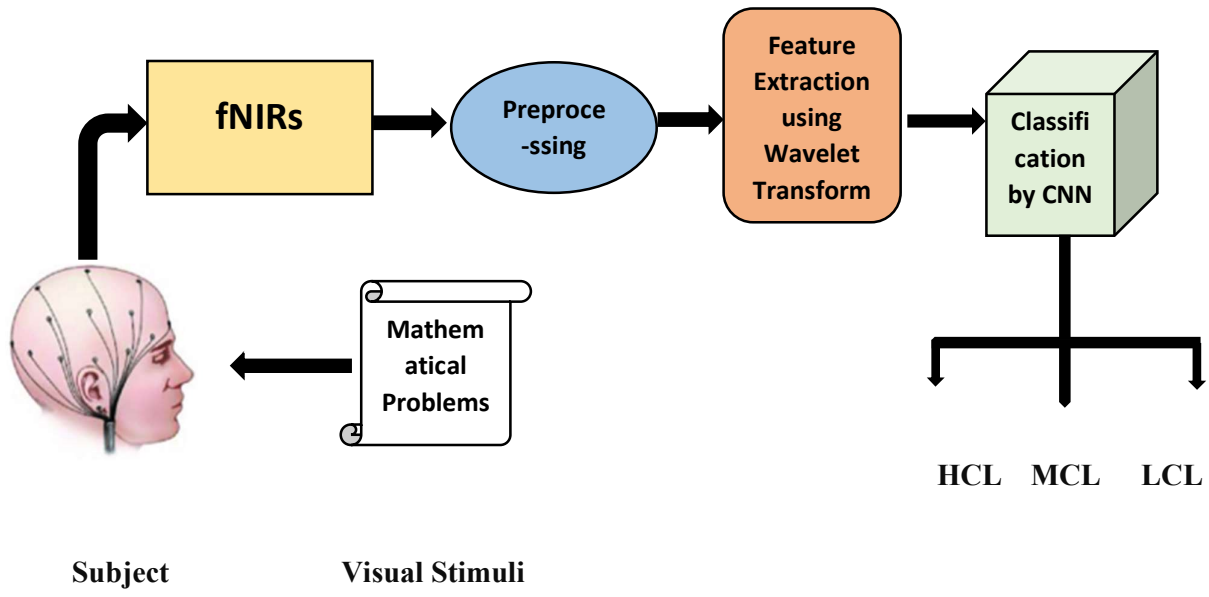


Figure 6 (a) : Block Diagram of Cognitive Load Analysis

6.4 Principle and Methodologies

a) Normalization of Raw Data

The signals obtained from the fNIRS acquisition system consist of oxy-hemoglobin blood concentration (in mmol/L) C_{Hbo} and deoxy-hemoglobin blood concentration (in mmol/L) C_{Hb}. To standardize the oxygenated (HbO) and deoxygenated (Hb) blood concentrations using a min-max scaler operation, we aim to transform the data onto a uniform scale between 0 and 1. First, we determine the minimum and maximum values for both HbO and Hb concentrations. Then, for each data point in the oxygenated blood concentration dataset, we subtract the minimum value and divide by the range (maximum - minimum) to obtain the normalized value. Similarly, we perform the same operation for the deoxygenated blood concentration dataset. Let the minimum and maximum values of C_{HbO} & C_{Hb} denoted as $C_{HbO}^{Max}(\tau)$, $C_{HbO}^{Min}(\tau)$, $C_{Hb}^{Max}(\tau)$ and $C_{Hb}^{Min}(\tau)$ respectively.

$$\hat{C}_{HbO}(\tau) = \frac{C_{HbO}(\tau) - C_{HbO}^{Min}(\tau)}{C_{HbO}^{Max}(\tau) - C_{HbO}^{Min}(\tau)} \tag{6.1}$$

$$\hat{C}_{Hb}(\tau) = \frac{C_{HbO}(\tau) - C_{Hb}^{Min}(\tau)}{C_{Hb}^{Max}(\tau) - C_{Hb}^{Min}(\tau)} \tag{6.2}$$

The oxygen Consumption at a given time instant τ is evaluated by a difference signal illustrated in (6.3).

$$d(\tau) = \hat{C}_{HbO}(\tau) - \hat{C}_{Hb}(\tau) \tag{6.3}$$

b) Feature Extraction using wavelet transform

In this work, we applied wavelet transformation [15] to a dataset of functional near-infrared spectroscopy (fNIRS) signals to improve the quality of the data by reducing noise. We started by loading the dataset from an Excel file named 'differences1.xlsx'. This dataset comprised features located in columns 0-13 and labels in column 14. The features represented the fNIRS signals from

multiple channels, while the labels denoted the class information associated with each data instance. Our primary objective was to enhance the signal quality by utilizing wavelet transforms, specifically targeting noise reduction.

To achieve this, we defined a function called **wavelet_transform** that performed wavelet decomposition on each signal channel. We used the Daubechies wavelet ('db4') with a decomposition level of 4. In the wavelet decomposition process, the signal was decomposed into approximation and detail coefficients. To reduce noise, we set the detail coefficients to zero, effectively filtering out high-frequency components that are often associated with noise. After modifying the coefficients, we reconstructed the signal using the inverse wavelet transform, ensuring the reconstructed signal retained the same length as the original.

We applied the **wavelet_transform** function to each column of the feature dataset using the **apply** method from the Pandas library. This process transformed all the signal channels, producing a new dataset of reconstructed signals. We then converted this transformed data back into a Data Frame and reattached the original labels, creating a new Data Frame that contained the denoised signals along with their corresponding labels.

To visually assess the impact of the wavelet, transform, we plotted the original and reconstructed signals for the data points from each channel. This comparison was crucial for evaluating the effectiveness of the noise reduction process. We set up the plot using Matplotlib, plotting each original signal with a dashed line and each reconstructed signal with a solid line. The visualization clearly demonstrated the reduction in high-frequency noise, highlighting the smoother nature of the reconstructed signals.

Finally, we saved the reconstructed dataset to a new Excel file named 'reconstructed_features_after_wavelet_001.xlsx'. This step ensured that the denoised data could be used for further analysis or machine learning applications. The entire process, from loading the data,

applying the wavelet transform, visualizing the results, and saving the output, was designed to be reproducible and scalable for larger datasets.

In summary, this work leveraged wavelet transformation to enhance the quality of fNIRS signals by reducing noise. By decomposing the signals, zeroing the detail coefficients, and reconstructing the signals, we effectively filtered out noise while preserving the essential characteristics of the original signals. The visual comparison between original and reconstructed signals confirmed the efficacy of our approach. The denoised dataset, saved for future use, provides a cleaner basis for subsequent analyses, potentially leading to more accurate and reliable results in research and clinical applications involving fNIRS data. This method underscores the importance of preprocessing in signal analysis, highlighting wavelet transform as a powerful tool for noise reduction.

c) Classification using CNN

In this section, we employed a Convolutional Neural Network (CNN) for multiclass classification using the TensorFlow and Keras frameworks.

- **Convolutional Neural Network**



Figure 6(b). Block Diagram of Proposed CNN Architecture

The architecture of the presented Convolutional Neural Network (CNN) is designed for multi-class classification tasks, comprising several layers that facilitate feature extraction and classification. The model starts with an input layer that reshapes the normalized feature data into a four-dimensional tensor of shape (14, 1, 1), where 14 is the number of features, and 1 is used to denote the width and channel dimensions. This reshaped input is then fed into the first convolutional layer, which consists

of 16 filters (also called kernels) each of size (3×1). The convolution operation can be mathematically described as:

$$h_{i,j,k} = \sum_{m=0}^{2} \sum_{n=0}^{0} x_{i+m,j+n} \cdot w_{m,n,k} + b_k \quad (6.4)$$

Where $h_{i,j,k}$ is the output of the convolution operation at position (i, j) for the k-th filter, $x_{i+m,j+n}$ represents the input data at position (i+m, j+n), $w_{m,n,k}$ denotes the filter weights, and b_k is the bias term for the k-th filter. The activation function applied here is the Mish function, defined as:

$$f(x) = x \cdot tanh(ln(1 + e^x)) \quad (6.5)$$

This activation function is known for its smooth gradient and better training performance compared to traditional functions like ReLU.

Following the first convolutional layer, an Average Pooling layer with a pooling size of 2 × 1 is applied. Pooling operations reduce the spatial dimensions of the feature maps, which can be expressed as:

$$P_{i,j,k} = \frac{1}{2} \sum_{m=0}^{1} h_{2i+m,j,k} \quad (6.6)$$

where$P_{i,j,k}$ is the pooled feature at position (i, j) for the k-th feature map, and the summation is taken over the pooling window.

The next layer in the architecture is another Conv2D layer with 32 filters of size 3×1. This layer also applies the Mish activation function to its outputs, similar to the first convolutional layer. The addition of multiple convolutional layers allows the network to capture increasingly complex patterns and hierarchies in the data.

After the second convolutional layer, another Average Pooling layer with the same pooling size 2×1 is applied, further reducing the spatial dimensions of the feature maps and thus decreasing the computational load in subsequent layers.

Next, the output from the pooling layer is flattened into a one-dimensional vector to prepare it for the dense (fully connected) layers. Flattening the data from a multidimensional tensor to a vector can be represented as:

$$f_i = h_{i//W,i\%W,k} \tag{6.7}$$

where f_i is the i-th element of the flattened vector, W is the width of the feature map, and k represents the channel index.

Following the flattening, a Dropout layer is applied with a dropout rate of 0.5, which randomly sets 50% of the input units to zero at each update during training to prevent overfitting.

The next layer is a dense layer with 64 units, also using the Mish activation function. The dense layer performs a weighted sum of its inputs, followed by the activation function, mathematically represented as:

$$a_i = f\left(\sum_{j=0}^{N-1} w_{i,j} f_j + b_i\right) \tag{6.8}$$

where a_i is the activation of the i-th unit, $w_{i,j}$ are the weights connecting the j-th input to the i-th unit, f_j is the j-th input, b_i is the bias term, and f is the activation function.

Another Dropout layer with a rate of 0.5 is applied after this dense layer to continue the effort to mitigate overfitting.

Finally, the output layer is a dense layer with 3 units, corresponding to the 3 classes in the classification task, and uses the SoftMax activation function to output a probability distribution over the classes. The SoftMax function is defined as:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}} \tag{6.9}$$

where $\sigma(z)_i$ is the output probability for the i-th class, z_i is the input to the softmax function for the i-th class, and C is the total number of classes.

The model is compiled using the Adam optimizer, which is an adaptive learning rate optimization algorithm designed for efficient and effective training. The loss function used is sparse categorical cross-entropy, appropriate for multi-class classification tasks with integer labels. This loss function is computed as:

$$L\,(y\,,\hat{y}) = -\frac{1}{N}\sum_{i=1}^{N} log\,(\hat{y}_{\,i,y_i}) \tag{6.10}$$

where L is the loss, y is the true label, $\hat{y}$ is the predicted probability distribution, and N is the number of samples.

To further enhance the training process, early stopping and learning rate reduction on plateau callbacks are implemented. Early stopping monitors the validation loss and stops training when it ceases to improve, preventing overfitting. The learning rate reduction callback reduces the learning rate when the validation loss plateaus, ensuring the model converges smoothly.

Overall, this CNN architecture leverages multiple convolutional and pooling layers to extract spatial features, dense layers to perform classification, and regularization techniques to generalize well to unseen data. This comprehensive approach combines state-of-the-art techniques in deep learning to achieve robust and accurate multi-class classification [17] performance.

In our experiment, the CNN model architecture consisted of convolutional layers followed by average pooling, flattening, and dense layers with dropout regularization. Mish activation function was utilized in convolutional and dense layers to introduce non-linearity. The model concluded with a SoftMax activation function in the output layer for multiclass classification. The model was compiled with the Adam optimizer and sparse categorical cross-entropy loss function. Early stopping and learning rate reduction callbacks were employed during model training to prevent overfitting and improve convergence. The model was trained on the training data with 20 epochs and a batch size of 64, utilizing a validation split of 20% for monitoring performance. The trained model was evaluated on the testing set to assess its performance.

6.5 Experiment and Result

6.5.1 Experimental Framework

The experiment utilized a complete brain Functional near Infrared Spectroscopy (fNIRS) setup within the Artificial Intelligence Laboratory at Jadavpur University. Data collection employed the NIRScout system from NIRx Medical technologies, USA, comprising 8 optode pairs emitting two specific wavelengths (760 and 850nm).

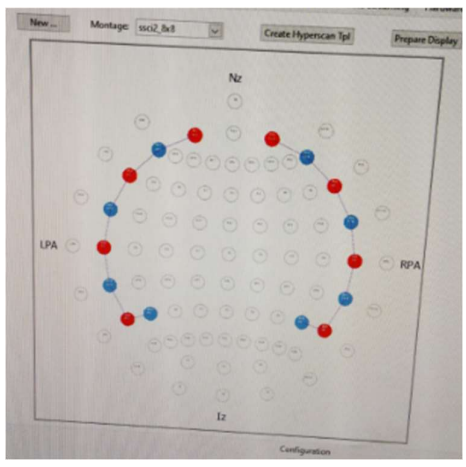


Figure 6(c): Montage

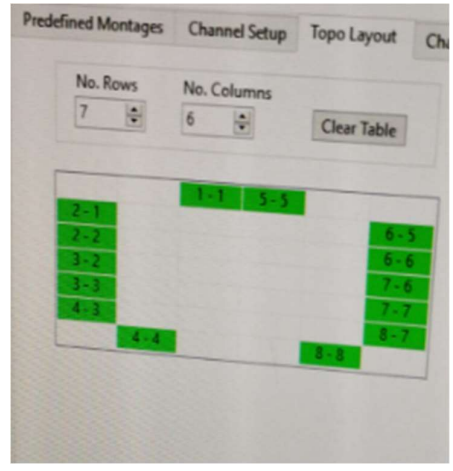


Figure 6(d): Optode Arrangement's topo layout

Given the focus on Cognitive Load Assessment during Mathematical Problem-Solving Tasks, the chosen montage targeted the prefrontal and parietal lobes, as depicted in Figure 6(c). Optodes were positioned using an NIRS-EEG compatible cap following the 10-10 standard placement system, with a 3 cm separation between sensors. The sampling rate of this hemodynamic analysis device was approximately 8Hz. For this study, 14 channels were selected from the available 64 channels based on the nearest neighbor source-detector combination. The optode arrangement's topo layout is illustrated in Figure 6(d). The acquired fNIRS data is analyzed using the nirsLAB software to explore the Brodmann area (BAs) involved in the recognition of various tasks. In our study, we used nirsLAB to map the optode channels to corresponding Brodmann Areas (BAs) using the 10-10 system. Optodes were precisely placed on the participants' scalps according to this system. nirsLAB utilized a standard

head model to spatially register these optode positions, aligning them with anatomical landmarks such as the nasion, inion, and preauricular points. This process enabled nirsLAB to map each channel to specific cortical regions and corresponding BAs, such as BA 7 and BA 40. Verification and adjustments were made using nirsLAB's visualization tools to ensure accuracy. This mapping allowed us to analyze fNIRS data in terms of neural activity in these specific BAs during mathematical problem-solving tasks.

6.5.2 Participants

In our laboratory experiment, there were 12 participants whose ages were between 20-30 years. Each participant engaged in a series of tasks focused solely on mathematical problem-solving. The experiment was divided into 5 sessions, with each session comprising 5 distinct trials. Throughout the experiment, participants were instructed to concentrate on solving various mathematical problems presented to them. This meticulous design resulted in a total of 300 experimental instances (5 trials / sessions × 5 sessions ×12 subjects), providing abundant data for thorough analysis and interpretation of cognitive processes involved in mathematical problem-solving tasks. The experimental set-up with Participant is depicted in Figure 6(e).



Figure 6(e). Experimental Set-Up: Subject Participating in the experiment

6.5.3 Stimuli Presentation

The experimental procedure commenced with the presentation of a fixation cross of 3sec, serving as a visual cue to focus participants' attention. Following this, a 30-second interval ensued, during which participants were tasked with mentally solving mathematical problems displayed on the screen. This interval allowed participants time to deliberate and formulate potential solutions internally.

Subsequently, a 15-second oral response period was initiated, providing participants with the opportunity to vocalize their answers to the presented mathematical questions. This oral response period facilitated the collection of immediate verbal feedback from participants regarding their solutions.

Following the oral response phase, a 10-second rest period was introduced to allow participants brief respite before the commencement of the next trial. This intertrial rest period aimed to mitigate potential fatigue or cognitive overload, ensuring participants remained engaged and responsive throughout the experiment.

This structured sequence of events, comprising fixation, problem-solving interval, oral response phase, and rest period, was repeated for each subsequent trial, maintaining consistency and standardization across the experimental session.

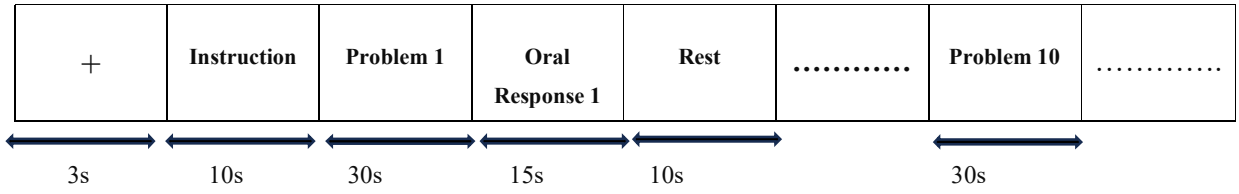


Figure 6(f). Stimuli Diagram

6.5.4 Feature Extraction Result

The main motivation of this experiment involves the determination of important feature sets [19] for cognitive load (High, Medium and Low). In this section, we applied wavelet transform to reconstruct features in the FNIRs dataset, stored in 'data_difference.xlsx'. The dataset was preprocessed, separating features and labels.

Utilizing the pywt library, we implemented a wavelet transform function, wavelet_transform (), which decomposes the features using the 'db4' wavelet with a level 4 decomposition. The approximation coefficients were set to zero, and the signal was reconstructed using modified coefficients.

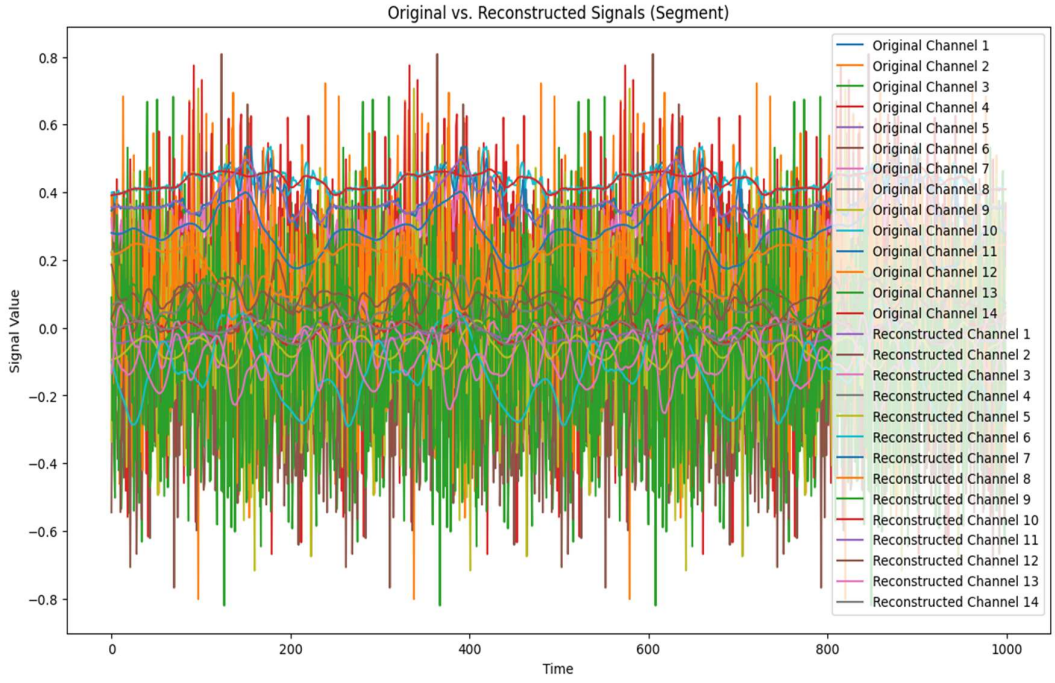


Figure 6(g): Output Graph of Wavelet Transform from trial

The wavelet transform was applied to each feature, resulting in a reconstructed feature dataset, 'reconstructed_features_after_wavelet.xlsx'. This dataset, comprising reconstructed signals, was plotted against the original signals for comparison. The plot visualizes the original and reconstructed

signals for a specific segment, providing insights into the effectiveness of the wavelet transform in capturing signal characteristics.

6.5.5 Relative Performance Analysis

In this section, we analyze the performance of our Convolutional Neural Network (CNN) model employed for multiclass classification. Evaluating the model's performance is crucial to ensure its efficiency and reliability in real-world applications. This analysis provides insights into the model's strengths and areas for improvement, guiding future enhancements.

The relative performance of the proposed classifier has been evaluated on the basis of three metrics- Classification Accuracy (*CA*), Sensitivity and Specificity which are defined by equations (6.11), (6.12), (6.13).

$$CA = \frac{TP + TN}{TP + TN + FP + FN} \tag{6.11}$$

$$Sensitivity = \frac{TP}{TP + F} \tag{6.12}$$

$$Specificity\ or\ Recall\ = \frac{TN}{TN + FP} \tag{6.13}$$

where, *TP*, *TN*, *FP*, *FN* indicates the number of true positives, true negatives, false positives and false negatives respectively. Table 6.1 depicts the comparative analysis of the proposed classifier with other primitive ones. It is quite evident from this table that the performance of the proposed classifier surpasses other classifiers by an enhanced marginal level. Hence, this classifier is able to provide a precise and accurate categorization of five object class labels.

Table 6.1: Classification Accuracy of Various Algorithms for Cognitive Load Analysis

SL. No.	Classifier Algorithm	Classification Accuracy (%)
1	Naïve bayes	76.54
2	Support Vector machine	82.11
3	Random Forest	87.61
4	Convolutional Neural Networks (CNN) without feature extraction	94.03
5	**Convolutional Neural Networks (CNN) with feature extraction**	**97.25**

The classification accuracy of various machine learning algorithms shows a clear trend of improvement with increasing model complexity and feature extraction techniques. Naïve Bayes, a straightforward probabilistic classifier, achieves a modest accuracy of 76.54%. Support Vector Machine (SVM), a more sophisticated method, improves the accuracy to 82.11%. Random Forest, an ensemble learning method, further enhances the performance to 87.61%. The significant leap in accuracy is observed with Convolutional Neural Networks (CNNs), a deep learning approach, which achieves 94.03% accuracy even without explicit feature extraction. When feature extraction is incorporated, CNNs reach an impressive accuracy of 97.25%, highlighting the powerful combination of deep learning and feature engineering in complex classification tasks

.

6.5.6 Comparison Analysis on the Basis of Activation Functions

To further evaluate the performance of the proposed classifier, we conducted a series of experiments where the activation function [23] in the first convolutional layer was varied. The impact of different activation functions on the classification accuracy was assessed, and the results are summarized in Table 6.2.

Table 6.2: Classification Accuracy with Activation Functions

SL. No.	Activation Function	Classification Accuracy (%)
1	Sigmoid	51.2
2	ReLu	78.7
3	Leaky ReLu	85.3
4	Elu	87.2
5	Swish	94.6
6	**Mish**	**98.7**

In this study, we evaluated the performance of various activation functions in the model for classification tasks by measuring their classification accuracy. The activation functions tested were Sigmoid, ReLU, Leaky ReLU, ELU, Swish, and Mish. The Sigmoid function yielded the lowest classification accuracy at 51.2%, indicating its limitations in capturing complex patterns within the data. The ReLU function showed a significant improvement with an accuracy of 78.7%, demonstrating its effectiveness in overcoming the vanishing gradient problem commonly associated with Sigmoid. Further enhancement was observed with the Leaky ReLU, achieving an accuracy of 85.3%, as it mitigates the issue of dying neurons by allowing a small gradient when the unit is not active. The ELU function provided a higher accuracy of 87.2%, benefiting from its ability to bring mean activations closer to zero, thus speeding up learning. Notably, the Swish function achieved a substantial boost in performance with an accuracy of 94.6%, attributed to its smooth and non-monotonic properties that

improve model expressiveness. The Mish function outperformed all others, achieving the highest classification accuracy of 98.7%, thanks to its ability to maintain a balance between complexity and ease of optimization, leading to superior gradient flow and model generalization. These results underscore the critical role of activation functions in neural network performance, with advanced functions like Swish and Mish offering significant advantages over traditional ones.

6.6 References

1. S. Tak and J.C. Ye, "Statistical analysis of fNIRS data: a comprehensive review," *NeuroImage*, vol. 85, pp. 72-91, 2014.

2. J. G. Collins, "Effect of mental arithmetic on vestibular nystagmus," *Journal of Experimental Psychology*, vol. 63, no. 6, pp. 550-555, 1962.

3. G. Greenberg, "Optokinetic nystagmus frequency and difficulty of mental arithmetic," *Perception & Psychophysics*, vol. 8, no. 3, pp. 167-170, 1970.

4. Y. Takeda, "The influence of mental activity on eye movements during fixation and tracking tasks," *Japanese Journal of Physiology*, vol. 26, no. 4, pp. 329-337, 1976.

5. N. Osaka and Y. Koga, "Eye fixation instability during cognitive tasks and its relation to central nervous system activity," *Neuroscience Letters*, vol. 29, no. 1, pp. 99-103, 1982.

6. R. Tate, "Effects of cognitive tasks on saccade latency: An investigation of task difficulty and hemispheric activation," *Neuropsychologia*, vol. 17, no. 5, pp. 595-604, 1979.

7. T. Wilcox and M. Biondi, "fNIRS in the developmental sciences," *Wiley Interdisciplinary Reviews: Cognitive Science*, vol. 6, no. 3, pp. 263-283, 2015.

8. N. Naseer and K.S. Hong, "fNIRS-based brain-computer interfaces: a review," *Frontiers in Human Neuroscience*, vol. 9, p. 3, 2015.

9. M. Ferrari and V. Quaresima, "A brief review on the history of human functional near-infrared spectroscopy (fNIRS) development and fields of application," *NeuroImage*, vol. 63, no. 2, pp. 921-935, 2012.

10. X. Hou, Z. Zhang, C. Zhao, L. Duan, Y. Gong, Z. Li, and C. Zhu, "NIRS-KIT: a MATLAB toolbox for both resting-state and task fNIRS data analysis," *Neurophotonics*, vol. 8, no. 1, p. 010802, 2021.

11. D. Zhang and D. Zhang, "Wavelet transform," in Fundamentals of Image Data Mining: *Analysis, Features, Classification and Retrieval*, pp. 35-44, 2019.

12. M.J. Shensa, "The discrete wavelet transform: wedding the a trous and Mallat algorithms," *IEEE Transactions on Signal Processing*, vol. 40, no. 10, pp. 2464-2482, 1992.

13. C.E. Heil and D.F. Walnut, "Continuous and discrete wavelet transforms," *SIAM Review*, vol. 31, no. 4, pp. 628-666, 1989.

14. D. Lopez-Martinez, K. Peng, A. Lee, D. Borsook, and R. Picard, "Pain detection with fNIRS-measured brain signals: a personalized machine learning approach using the wavelet transform and Bayesian hierarchical modeling with Dirichlet process priors," *in 2019 8th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*, pp. 304-309, IEEE, Sept. 2019.

15. D. Perpetuini, D. Cardone, C. Filippini, A.M. Chiarelli, and A. Merla, "A motion artifact correction procedure for fNIRS signals based on wavelet transform and infrared thermography video tracking," *Sensors,* vol. 21, no. 15, p. 5117, 2021.

16. A.M. Chiarelli, E.L. Maclin, M. Fabiani, and G. Gratton, "A kurtosis-based wavelet algorithm for motion artifact correction of fNIRS data," *NeuroImage*, vol. 112, pp. 128-137, 2015.

17. B. Xu, Y. Fu, L. Miao, Z. Wang, and H. Li, "Classification of fNIRS data using wavelets and support vector machine during speed and force imagination," *in 2011 IEEE International Conference on Robotics and Biomimetics*, pp. 1224-1229, IEEE, Dec. 2011.

18. P. Reddy, K. Izzetoglu, and P.A. Shewokis, "Wavelet-based analysis of fNIRS measures enable assessment of workload," *in International Conference on Human-Computer Interaction, Cham: Springer International Publishing*, pp. 173-182,2022.

19. M. Saadati, J. Nelson, and H. Ayaz, "Convolutional neural network for hybrid fNIRS-EEG mental workload classification," *in Advances in Neuroergonomics and Cognitive Engineering: Proceedings of the AHFE 2019 International Conference on Neuroergonomics and Cognitive Engineering, and the AHFE International Conference on Industrial Cognitive Ergonomics and Engineering Psychology*, Washington DC, USA, vol. 10, Springer International Publishing, 2020, pp. 221-232 July 24-28, 2019.

20. H. Ghonchi, M. Fateh, V. Abolghasemi, S. Ferdowsi, and M. Rezvani, "Deep recurrent–convolutional neural network for classification of simultaneous EEG–fNIRS signals," *IET Signal Processing*, vol. 14, no. 3, pp. 142-153, 2020.

21. T. Ma, S. Wang, Y. Xia, X. Zhu, J. Evans, Y. Sun, and S. He, "CNN-based classification of fNIRS signals in motor imagery BCI system," *Journal of Neural Engineering*, vol. 18, no. 5, p. 056019, 2021.

22. H. Hush, "Classification with neural networks: a performance analysis," *in IEEE 1989 International Conference on Systems Engineering*, pp. 277-280, IEEE, Aug. 1989.

23. A.D. Rasamoelina, F. Adjailia, and P. Sinčák, "A review of activation function for artificial neural network," *in 2020 IEEE 18th World Symposium on Applied Machine Intelligence and Informatics (SAMI)*, pp. 281-286, IEEE, Jan. 2020.

CHAPTER 7

Conclusion

In this thesis, we investigated the effects of functional near-infrared spectroscopy (fNIRS) in assessing cognitive load during mathematical problem-solving tasks. The hemodynamic responses measured via fNIRS provided significant insights into the brain's activity, particularly in the prefrontal cortex, which is crucial for tasks involving complex cognitive functions. Our findings indicate that there is a clear correlation between the difficulty of mathematical problems and the level of cognitive load, as evidenced by the variations in oxygenated and deoxygenated hemoglobin concentrations. Additionally, we examined the use of wavelet transformation and Convolutional Neural Networks (CNNs) for enhancing the analysis and classification of cognitive load states using fNIRS data.

Our analysis revealed that as the cognitive load increased, there was a corresponding increase in prefrontal cortex activity, which underscores the utility of fNIRS in capturing the nuances of cognitive engagement and workload. This highlights the potential of fNIRS as a practical tool for real-time assessment of cognitive load, offering a non-invasive and relatively accessible means of studying brain function compared to traditional neuroimaging techniques. Similarly, Greenberg (1970) observed higher optokinetic nystagmus frequency during difficult mental arithmetic tasks compared to easier mental tasks. Takeda (1976) showed that mental activity during problem-solving tasks disrupted performance in fixation and eye tracking tasks.

To standardize the oxygenated (HbO) and deoxygenated (Hb) blood concentrations, we applied a min-max scaler operation to transform the data onto a uniform scale between 0 and 1. This normalization process enhanced the stability and convergence of machine learning models trained on the dataset by mitigating the effects of varying feature scales. Additionally, the difference between the two datasets for each data point was computed to elucidate potential hemodynamic responses during mathematical problem-solving tasks. This approach allowed for the examination of fluctuations in cerebral oxygenation levels and their correlation with cognitive processing demands in our study. We leveraged wavelet transformation to enhance the quality of fNIRS signals by reducing noise. By decomposing the signals, zeroing the detail coefficients, and reconstructing the signals, we effectively filtered out noise while preserving the essential characteristics of the original signals. The visual comparison between original and reconstructed signals confirmed the efficacy of our approach. The denoised dataset provided a cleaner basis for subsequent analyses, potentially leading to more accurate and reliable results in research and clinical applications involving fNIRS data.

The architecture of the presented Convolutional Neural Network (CNN) was designed for multi-class classification tasks, comprising several layers that facilitate feature extraction and classification. The model demonstrated high classification accuracy, with CNNs achieving 94.03% accuracy even without explicit feature extraction, and 97.25% with feature extraction. This significant improvement highlights the powerful combination of deep learning and feature engineering in complex classification tasks.

The relative performance of the proposed classifier was evaluated based on classification accuracy, sensitivity, and specificity. Our CNN model outperformed traditional classifiers like Naïve Bayes, Support Vector Machine (SVM), and Random Forest, achieving the highest classification accuracy. Additionally, we conducted experiments to assess the impact of different activation functions on the classification accuracy. The Mish activation function outperformed others, achieving the highest classification accuracy of 98.7%, demonstrating its superior gradient flow and model generalization capabilities. This study demonstrates the potential of combining advanced signal processing techniques and machine learning models to enhance the measurement and analysis of cognitive load using fNIRS. These advancements pave the way for more effective educational tools and a better understanding of cognitive processes, ultimately contributing to improved learning outcomes and mental health assessments.

CHAPTER 8

Future Scope of Work

The research conducted in this thesis has laid a solid foundation for understanding the relationship between cognitive load and eye movements, as well as the utilization of functional Near-Infrared Spectroscopy (fNIRS) for cognitive load measurement. The promising results of this study open numerous avenues for future research and practical applications. Here, we outline potential directions for further exploration and development.

1. Advanced Signal Processing Techniques

The application of wavelet transform for noise reduction in fNIRS signals proved effective in this study. Future research could explore other advanced signal processing techniques such as empirical mode decomposition (EMD), independent component analysis (ICA), and machine learning-based denoising methods. These techniques could provide better noise reduction, leading to more accurate cognitive load measurements.

- **Adaptive Signal Processing:** Developing adaptive algorithms that can adjust to the dynamic nature of fNIRS signals in real-time could enhance the robustness of cognitive load measurements. These algorithms could automatically identify and mitigate noise and artifacts, ensuring high-quality data for analysis.

2. Integration with Other Physiological Measures

While this study focused on fNIRS data, integrating it with other physiological measures such as electroencephalography (EEG), eye-tracking, and heart rate variability could provide a more comprehensive understanding of cognitive load.

- **Multimodal Data Fusion:** Combining fNIRS with EEG can provide complementary information about brain activity. EEG offers high temporal resolution, while fNIRS provides better spatial resolution. Fusing these data types could lead to more accurate and detailed cognitive load assessments.

- **Eye-Tracking Integration:** Incorporating eye-tracking data could help validate the findings related to eye movements and cognitive load. It could also offer insights into how visual attention and cognitive load interact during complex tasks.

3. Machine Learning and Artificial Intelligence

The use of Convolutional Neural Networks (CNNs) for classifying cognitive load states demonstrated high accuracy. Future work could explore other machine learning and artificial intelligence (AI) techniques to enhance classification performance and expand the range of detectable cognitive states.

- **Deep Learning Models:** Exploring different deep learning architectures such as Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and Transformer models could improve the ability to capture temporal dependencies and complex patterns in fNIRS data.

- **Explainable AI:** Developing explainable AI models that can provide insights into how decisions are made could enhance the interpretability and trustworthiness of cognitive load measurements. This is particularly important in educational and clinical applications where understanding the underlying reasons for cognitive load changes is crucial.

4. Real-Time Cognitive Load Monitoring

One of the exciting future directions is the development of real-time cognitive load monitoring systems. These systems could be used in educational settings, workplaces, and clinical environments to provide immediate feedback and interventions.

- **Wearable fNIRS Devices:** Advances in wearable technology could lead to the development of portable fNIRS devices that can be worn comfortably for extended periods. These devices could continuously monitor cognitive load in real-world environments.

- **Real-Time Algorithms:** Developing real-time algorithms for processing and analyzing fNIRS data could enable instantaneous detection of cognitive load changes. This could be useful for adaptive learning systems that adjust instructional materials based on the learner's cognitive state.

5. Personalized Learning and Adaptive Systems

The findings from this study can be applied to develop personalized learning systems that adapt to the cognitive load of individual learners. This could enhance learning outcomes by ensuring that instructional materials are neither too easy nor too difficult.

- **Adaptive Learning Platforms:** Integrating cognitive load measurement into e-learning platforms could enable the automatic adjustment of content difficulty based on the learner's current cognitive load. This could help maintain optimal learning conditions and prevent cognitive overload.

- **Personalized Feedback:** Providing personalized feedback based on cognitive load measurements could help learners become more aware of their cognitive states and develop better self-regulation strategies.

6. Clinical and Therapeutic Applications

In clinical settings, cognitive load measurement can be used to assess the cognitive function of patients with neurological disorders, mental health conditions, or brain injuries. This could aid in diagnosis, treatment planning, and monitoring of therapeutic interventions.

- **Neurorehabilitation:** Cognitive load monitoring could be used in neurorehabilitation programs to tailor exercises to the cognitive capabilities of patients, ensuring that tasks are appropriately challenging without causing excessive cognitive strain.

- **Mental Health Assessments:** In mental health care, cognitive load measurements could help assess the cognitive impact of various conditions such as depression, anxiety, and ADHD. This could inform treatment strategies and monitor progress over time.

7. Challenges and Ethical Considerations

As cognitive load measurement technologies advance, addressing the associated challenges and ethical considerations will be crucial.

- **Data Privacy and Security:** Ensuring the privacy and security of fNIRS data is essential, particularly when used in sensitive environments such as healthcare and education.

- **Ethical Use of AI:** Developing guidelines for the ethical use of AI in cognitive load measurement is important to prevent misuse and ensure that the technology is used to benefit individuals and society.

The future scope of cognitive load research and its applications is vast and promising. By advancing signal processing techniques, integrating multimodal data, leveraging machine learning, and developing real-time monitoring systems, we can enhance our understanding and measurement of cognitive load. These advancements have the potential to transform educational practices, clinical assessments, and various other fields, ultimately leading to better outcomes and improved quality of life. As we move forward, addressing the challenges and ethical considerations will be key to ensuring the responsible and beneficial use of these technologies.

Appendix

User Guide to Run the Codes

This section provides step wise guidance for the execution of the codes given in the accompanying CD at the end of the thesis.

- **Step 1:**

  For the work we are using Google Colab platform and first we have to upload the required dataset of a session (as we have to work with multiple session's data) as shown in Figure: 1

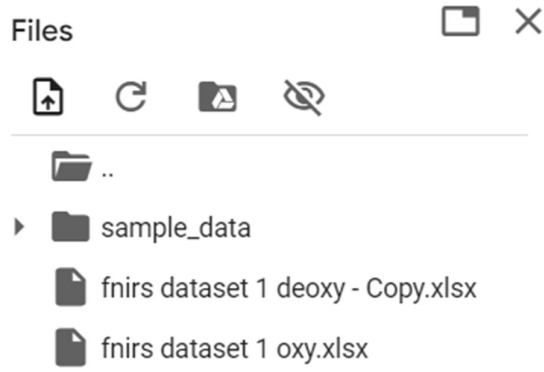


Figure 1: Uploading the Data

- **Step 2:**

  In the context of Python programming, import all the required library. If a library is not available , install it using pip command.

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from tensorflow.keras import layers, models, callbacks
```

Figure 2: Importing Library

- **Step 3:**

The next step is the data cutting process and labelling based on subject responses for both oxygenated and deoxygenated datasets.

```
# Assign labels to each subset
df4.loc[65:306, 'Label'] = 1
df4.loc[508:749, 'Label'] = 1
df4.loc[951:1192, 'Label'] = 2
df4.loc[1394:1635, 'Label'] = 0
df4.loc[1837:2078, 'Label'] = 2
df4.loc[2280:2521, 'Label'] = 1
df4.loc[2723:2964, 'Label'] = 2
df4.loc[3166:3407, 'Label'] = 1
df4.loc[3609:3850, 'Label'] = 0
df4.loc[4052:4293, 'Label'] = 2
```

Figure 3: Data Cutting and Labelling

- **Step 4:**

Then merge all datasets of different classes (considering three classes) for both oxygenated and deoxygenated datasets

```
# Concatenate all the noisy matrices into a single DataFram
merged_data1 = pd.concat(all_matrices1, ignore_index=True)


# Save the merged DataFrame to an Excel file
excel_file_path6 = 'merged_matrices_corrected_hbO3.xlsx'  #
merged_data1.to_excel(excel_file_path6, index=False)

print(f'Data saved to {excel_file_path6}')
```

Figure 4: Merge the Datasets

- **Step 5:**

Normalize the merge datasets.

```
import pandas as pd
from sklearn.preprocessing import MinMaxScaler


# Extract labels before normalization
labels = hb3['Label']

# Drop the 'Label' column for normalization
hb3_updated = hb3.drop('Label', axis=1)

# Perform Min-Max scaling
scaler = MinMaxScaler()
normalized_data = scaler.fit_transform(hb3_updated)

# Create a DataFrame with normalized data
normalized_hb3 = pd.DataFrame(normalized_data, columns=hb3_updated.columns)

# Add the 'Label' column back to the normalized DataFrame
normalized_hb3['Label'] = labels


# Save the normalized data to a new Excel file or overwrite the existing one
normalized_file_path1 = 'normalized_data_hb3.xlsx'
normalized_hb3.to_excel(normalized_file_path1, index=False)

print(f"Normalized data saved to {normalized_file_path1}")
```

Figure 5: Normalize the Datasets

- **Step 6:**

In this step we have to find out differences between the normalized oxygenated data and normalized deoxygenated data.

```
# Create a new dataframe for the differences, copying the labels from one of the datasets
data_difference3 = data_hb3.copy()
data_difference3.iloc[:, :-1] = data_hbO3.iloc[:, :-1] - data_hb3.iloc[:, :-1]
```

Figure 6: Difference of the Datasets

- **Step 7:**

Then we are doing feature extraction using wavelet transform.

```
# Separate the features (columns 0-13) and the labels (column 14)
features = data_difference.iloc[:, :-1]
labels = data_difference.iloc[:, -1]

# Function to perform wavelet transform on a single dataset
def wavelet_transform(data_difference):
    # Perform wavelet transform (using 'db4' wavelet and level 4 decomposition)
    coeffs = pywt.wavedec(data_difference, 'db4', level=4)
    # Set the approximation coefficients (cA) to zero
    for i in range(1, len(coeffs)):
        coeffs[i] = np.zeros_like(coeffs[i])
    # Reconstruct the signal from the modified coefficients
    reconstructed_signal = pywt.waverec(coeffs, 'db4', mode='constant')
    return reconstructed_signal[:len(data_difference)]

# Apply wavelet transform to the features dataset (all 14 channels)
reconstructed_features = features.apply(wavelet_transform, axis=0)

# Convert the reconstructed features to a DataFrame
reconstructed_features_df = pd.DataFrame(reconstructed_features.values.tolist(),
```

Figure 7: Feature Extraction by Wavelet Transform

- **Step 8:**

Then we are performing Classification by Convolutional Neural Network using all the output datasets obtained after feature extraction step .

```
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Reshape your data for Conv2D
X_train = X_train.reshape((X_train.shape[0], X_train.shape[1], 1, 1))
X_test = X_test.reshape((X_test.shape[0], X_test.shape[1], 1, 1))

# Define the CNN model with Conv2D
model = models.Sequential()
model.add(layers.Conv2D(16, kernel_size=(3, 1), activation='mish', input_shape=(X_train.shape[1],
model.add(layers.AveragePooling2D((2, 1)))
model.add(layers.Conv2D(32, kernel_size=(3, 1), activation='mish'))
model.add(layers.AveragePooling2D((2, 1)))
model.add(layers.Flatten())
model.add(layers.Dropout(0.5))
model.add(layers.Dense(64, activation='mish'))
model.add(layers.Dropout(0.5))
model.add(layers.Dense(3, activation='softmax'))

# Compile the model
model.compile(optimizer='adam', loss='sparse_categorical_crossentropy', metrics=['accuracy'])

# Define callbacks for early stopping and learning rate reduction
early_stopping = callbacks.EarlyStopping(monitor='val_loss', patience=5, restore_best_weights=True
reduce_lr = callbacks.ReduceLROnPlateau(monitor='val_loss', factor=0.2, patience=3, min_lr=1e-6)

# Train the model with early stopping and learning rate reduction
history = model.fit(X_train, y_train, epochs=20, batch_size=64, validation_split=0.2, callbacks=[e
```

Figure 8: Classification by CNN