IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

Data Visualization Using Machine Learning Based On The Analysis Of Electroencephalogram

Sowmiya.E[1], Sandhiya.K[2], Sathiya Priya.K[3], Sowmiya.R[4], Kirthiga.M[5]

1 Assistant Professor,Vivekanandha College of Engineering for Women,Tiruchengode,Tamilnadu (India),
2,3,4,5 Student Of Vivekanandha College of Engineering for Women,Tiruchengode,Tamilnadu (India)

Abstract— Utilizing Python for the analysis and visualization of electroencephalogram (EEG) data offers significant insights into brain activity patterns. The methodology encompasses preprocessing raw EEG signals, performing spectral analysis using Fast Fourier Transform (FFT), and employing advanced data visualization techniques. Key Python libraries such as NumPy, Pandas, and Matplotlib are employed for data manipulation and visualization. The visualization includes spectrograms, power spectral density plots, and topographic maps, facilitating the understanding of EEG frequency content and spatial distribution. Additionally, machine learning algorithms are considered for feature extraction and classification to identify cognitive states. The outcomes aim to develop robust Python-based scripts for EEG data processing, aiding researchers and clinicians in analyzing brain activity patterns and potentially identifying cognitive states or anomalies. Contributions to neuroscience are made by providing accessible tools for EEG data analysis and visualization using Python.

Index Terms— Electroencephalogram (EEG), Python, Data Analysis, Spectral Analysis, Visualization, Machine Learning, Brain Activity Patterns

I. INTRODUCTION

Understanding brain activity patterns through the analysis and visualization of electroencephalogram (EEG) data is of paramount importance in neuroscience. Python's versatility and robust library ecosystem make it an excellent choice for this task. The methodology implemented involves several critical steps, starting with the preprocessing of raw EEG signals. This step includes filtering and artifact removal to ensure the data's integrity. Spectral analysis is performed using Fast Fourier Transform (FFT), which is instrumental in extracting frequency domain features from the EEG signals. This approach is essential for understanding the underlying frequency components of brain activity. Python's scientific libraries, such as NumPy, Pandas, and Matplotlib, are pivotal in this process, providing the tools necessary for efficient data manipulation and comprehensive visualization.

The visualization phase includes the creation of spectrograms, power spectral density (PSD) plots, and topographic maps. These visual tools are crucial for interpreting the EEG data, offering insights into the frequency content and spatial distribution of brain activity. Furthermore, the integration of machine learning algorithms enhances the analysis by enabling feature extraction and classification, which can identify specific cognitive states from the EEG data. The anticipated outcomes include the development of robust, Python-based scripts that streamline EEG data processing, spectral analysis, and visualization. These tools are designed to assist researchers and clinicians in gaining deeper insights into brain activity patterns and identifying any anomalies or cognitive states of interest. By providing accessible and effective tools for EEG data analysis, significant contributions are made to the field of neuroscience.

II. LITERATURE REVIEW

The analysis and classification of EEG signals for mental state assessment have seen significant advancements due to the integration of modern machine learning techniques. Various existing systems have demonstrated innovative approaches and methodologies to enhance the accuracy and efficiency of EEG signal processing. In this literature review, we evaluated ten significant papers on data visualization using machine learning techniques applied to electroencephalogram (EEG) analysis. The studies reveal various methodologies for enhancing EEG data interpretation through machine learning, focusing on themes such as feature extraction, pattern recognition, and visualization techniques. Key insights include the ability of machine learning models to uncover complex patterns in EEG signals that are not readily apparent through traditional methods, and the improvement in visual clarity and interpretability of EEG data. The summarized table below outlines the contributions, approaches, and results of each paper. This review underscores the potential of machine learning to advance EEG data visualization and identifies opportunities for further research to refine these techniques and enhance their practical applications.

III.SYSTEM ANALYSIS

3.1 EXISTING SYSTEM :

The EEG signals involves capturing brainwave data using EEG devices. In Existing system we used Conventional methodologies, it provides Graphical plotting for data visualization. The current paradigm in EEG dataset analysis involves conventional methodologies predominantly reliant on specialized software such as MATLAB, EEGLAB, or similar proprietary tools. These established systems commonly encompass rudimentary preprocessing steps like noise filtering, baseline correction, and artifact removal to enhance data quality.

The well-established systems typically include simple preprocessing steps to improve the quality of the data. Visualization, typically facilitated by basic plotting functions like scalp topographies, time-series plots, and frequency spectra. These systems are limited in their flexibility and advanced analytical capabilities, even though they offer a basic framework for the analysis and visualisation of EEG data.The lack of Modern programming languages like Python offer state-of-the-art algorithms and interactive visualization approaches. The absence of machine learning-driven analytical modules and interactive visualization tools. Some systems focus on real-time visualization for immediate feedback during EEG recording sessions, while others emphasize post-processing and analysis of recorded data. Despite the diversity of existing systems, ongoing research and development continue to drive innovation in EEG data visualization, aiming to improve usability, interpretability, and utility across various domains such as neuroscience, clinical neurology, cognitive psychology, and brain-computer interface (BCI) technology.

3.1.1 PROBLEM STATEMENT:

1. Plotting system is one of its drawbacks it cannot be accurately displayed or understand by the people.
2. It is also difficult to analyse by the doctor.
3. A model that is too complicated or that was trained on a small dataset may identify patterns or noise that are not properly generalized to new data, resulting in inaccurate visualizations.

3.2 PROPOSED SYSTEM :

CNNs in EEG signal analysis include tasks such as brain-computer interface (BCI) systems, where the model can interpret brain activity for control commands, and medical diagnosis, where abnormal patterns in EEG signals are detected. A doctor can quickly analyse and provide what issue they are facing by using our suggested system. By using this common people can also able to understand it. Machine learning (ML) and python is then applied for data analysis and interactive visualization. Data visualization is done as Graphical Animation(Topography) by using AI Model. It will get 100% accuracy result. So, By creating this Website, hope to identify things that are only recognized by doctors and that regular people can also grasp. The system aims to introduce innovative methodologies for EEG data analysis and visualization, emphasizing originality in methodologies and approach, thereby advancing the domain with sophisticated techniques and accessible tools for comprehensive data exploration and understanding. Through customizable visualization parameters and real-time feedback, users can explore EEG data with enhanced clarity, facilitating insights into brain activity patterns associated with various cognitive states or neurological conditions. Moreover, the system could incorporate features for comparative analysis, trend identification, and anomaly detection, empowering users to uncover meaningful patterns and correlations within EEG datasets efficiently. Ultimately, this proposed system holds the potential to revolutionize EEG analysis by providing a comprehensive and accessible platform for visualizing brain activity dynamics, thus advancing our understanding of the human brain and its functions.

3.2.1 ADVANTAGES :
1. CNNs are known for their ability to reduce computational time and adapt to different variations of images Python will be used; it's easy to use and provides powerful visualization.
2. Physicians are clearly recognized, and the people can understand.
3. CNNs work by applying a series of convolution and pooling layers to an output image or video.

3.3 FEASIBILITY STUDY
3.3.1 TECHNICAL FEASIBILITY :
Streamlit Framework:
Streamlit is a widely used Python library for building interactive web applications for data science and machine learning projects. It provides easy integration with data visualization libraries like Matplotlib and Pandas, making it suitable for our EEG data visualization application.
Python Ecosystem:
Python offers extensive libraries and tools for data manipulation, signal processing, and visualization, which are essential for handling EEG data effectively.
PyEDFlib:
The PyEDFlib library allows reading EDF (European Data Format) files, which is the standard format for storing EEG data. It provides the necessary functionality to load EEG data into the application.
3.3.2 ECONOMIC FEASIBILITY:
Open Source Tools:
Streamlit, Python, PyEDFlib, and other required libraries are open-source and free to use. There are no licensing costs. associated with developing or deploying the application.
Minimal Hardware Requirements:
The application can run on standard laptops or desktop computers, requiring minimal hardware resources. Therefore, there are no significant hardware costs involved.

3.3.3 OPERATIONAL FEASIBILITY:
User-Friendly Interface:
Streamlit allows for the creation of intuitive and interactive web interfaces with minimal coding effort. The application will have straightforward controls and visualizations, making it easy for users to navigate and understand.
Data Accessibility:
The EEG dataset in EDF format is readily available and can be easily accessed for testing and development purposes. This ensures that the application can be operated efficiently with the necessary input data.

3.3.4. LEGAL AND ETHICAL FEASIBILITY:
Data Privacy and Security:
The application should adhere to data privacy regulations and ensure the security of user data. Any sensitive information, such as personal identifiers, should be handled responsibly to maintain user confidentiality.
Compliance with Data Usage Policies:
Ensure that the EEG dataset used in the application is obtained legally and ethically, with proper attribution provided to the data source as per licensing agreements or data usage policies.

IV.SOFTWARE DESCRIPTION
4.1 FRONT END:
STREAMLIT :
Streamlit is a powerful and user-friendly open-source Python library designed for creating interactive and data-driven web applications with ease. Developed with a focus on simplicity and rapid prototyping, Streamlit enables data scientists and engineers to effortlessly convert Python scripts into shareable and interactive web apps. Its minimalistic syntax allows users to build web applications without the need for extensive knowledge of web development technologies such as HTML, CSS, or JavaScript. Streamlit provides a wide range of widgets, including sliders, buttons, and charts, facilitating dynamic interactions with data.
The library seamlessly integrates with popular data science tools and libraries, such as Pandas, Matplotlib, and Plotly, enabling users to leverage their existing data analysis workflows. Streamlits capabilities extend to

data exploration, real-time visualization, and customization of web applications. Furthermore, the deployment process is straightforward, allowing users to easily share their applications on various platforms. With its simplicity, versatility, and rapid development capabilities, Streamlit has become a go-to choice for creating and deploying web applications in the data science and machine learning communities.

USE OF STREAMLIT IN EEG DATA VISUALIZATION :

Streamlit proves to be an invaluable tool for EEG (Electroencephalogram) data visualization, providing an accessible and interactive platform for researchers and practitioners. With its straightforward syntax and versatile widgets, Streamlit allows for real-time visualization of EEG signals, enabling dynamic exploration of brain activity. Users can interactively select specific channels, apply filters, and preprocess data, facilitating on-the-fly analysis. Streamlits capabilities extend to spectral analysis, allowing users to examine frequency-domain characteristics of EEG signals with ease.

The platform interactive event markers and annotations make it simple to correlate specific events with corresponding EEG patterns. Additionally, Streamlit supports the integration of machine learning models, providing a seamless interface to visualize and analyze model predictions on EEG data. Whether for educational purposes or research, Streamlit empowers users to create customized dashboards, export visualizations, and share insights, making it a valuable asset in the field of EEG data visualization.

4.2 ALGORITHM :

The essential parameters of CNN and regression models include convolutional layers, average pooling, relu activation, hidden layers, and FNN .Generally, as the network gets deeper, the neural network typically learns better .However, this may encourage the model to overfit during training. Therefore, we carefully designed a neural network architecture that required few parameters. After the training phase, the highest classification performance was calculated based on the test data. CNN has been demonstrated high performance on image classification and pattern detection. In this paper, we combine the continuous wavelet transform (CWT) and CNN to classify epileptic seizure. This experiment uses the wavelet transform to convert signal data of EEG to time- frequency domain images. A convolutional neural network (CNN) is a specific type of artificial neural network that uses perceptrons, a machine learning unit algorithm, for supervised learning, to analyze data. CNNs apply to image processing, natural language processing and other kinds of cognitive tasks. CNNs work by applying a series of convolution and pooling layers to an input image or video.

A Convolutional Neural Network (CNN) is a type of deep learning algorithm commonly used for image recognition tasks due to its ability to effectively capture spatial patterns within the data. While CNNs are not traditionally used for data visualization in the same way as they are for image recognition, they can still be applied to analyze data from sources like electroencephalogram (EEG)

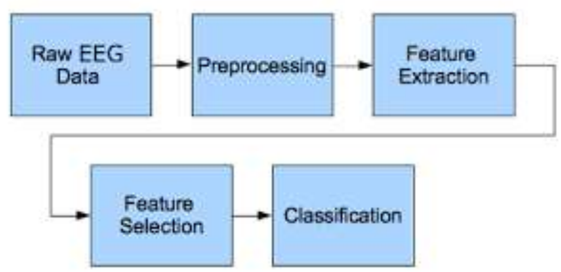


Fig 4.1 EEG Data

Here's how CNNs could be utilized for analyzing EEG data and potentially for data visualization:

1. Data Preprocessing:

EEG data usually consists of time-series data representing electrical activity in the brain. Before feeding this data into a CNN, preprocessing steps such as filtering, normalization, and feature extraction might be necessary to prepare the data for analysis.

2. CNN Architecture:

A CNN architecture suitable for analyzing EEG data needs to be designed. This architecture would typically consist of convolutional layers, pooling layers, and possibly fully connected layers. The convolutional layers are crucial for capturing local patterns in the EEG signals, while pooling layers help reduce the spatial dimensions of the data.

3.Training the CNN:

The CNN is trained using labeled EEG data. Labels might represent different brain states or activities, such as different mental states (e.g., attention, relaxation) or detecting anomalies (e.g., epileptic seizures). During training, the CNN learns to extract features from the EEG data that are relevant for predicting the corresponding labels.

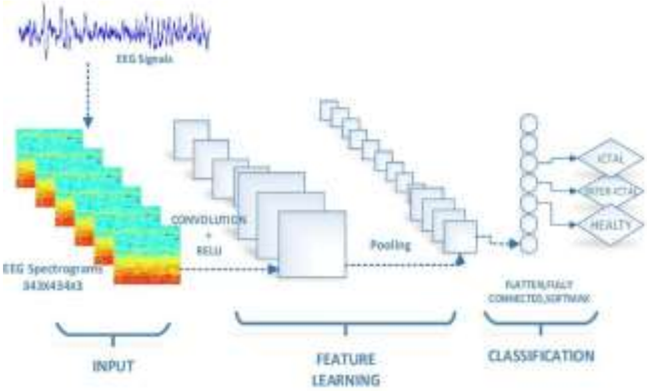


Fig 4.2 CNN in EEG Data

4.Evaluation:

The trained CNN is evaluated on a separate dataset to assess its performance in classifying EEG data accurately. Metrics such as accuracy, precision, recall, and F1-score are commonly used to evaluate classification performance.

5.Visualization:

While CNNs themselves don't typically produce visualizations in the same way as other machine learning models (e.g., decision trees, t-SNE), techniques can be applied to visualize the learned features or activations within the CNN. For instance, methods like activation maximization or gradient ascent can be used to visualize what features in the EEG data are most relevant for a particular classification decision.

6.Interpretation:

Interpretation of CNN results can involve understanding which parts of the EEG signals are most influential in the classification decisions. Visualization techniques combined with domain knowledge can help interpret the learned features and understand how different brain activities are represented in the EEG data.

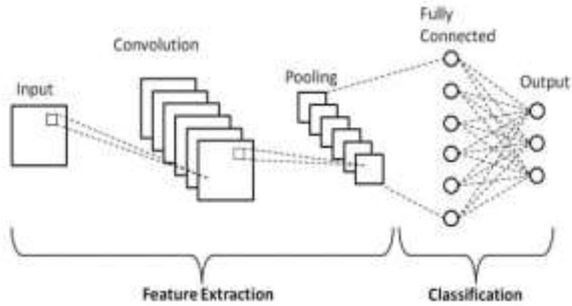


Fig 4.3 Basic CNN Classification

V.PROJECT DESCRIPTION

5.1 PROBLEM DEFINITION :

The EEG Data Visualization project aims to tackle various challenges associated with the analysis and interpretation of EEG (Electroencephalography) data obtained from Emotiv 14 channel headsets. One of the primary challenges lies in the complex nature of interpreting EEG signals, which demands expertise in signal processing and neurophysiology. This complexity often poses difficulties for individuals lacking specialized knowledge, hindering their ability to effectively decipher the data. Moreover, the limited accessibility of EEG data further compounds the issue, as accessing and analyzing such data typically requires specialized software and expertise, thereby restricting widespread adoption and exploration of EEG datasets.

Additionally, the absence of interactive visualization tools exacerbates the problem, as existing analysis tools often lack intuitive interfaces for exploring EEG data interactively. Consequently, researchers and enthusiasts encounter obstacles in effectively exploring and interacting with the data. Moreover, the reliance on traditional, manual methods for EEG data analysis contributes to time-consuming and labor-intensive procedures, further impeding efficient analysis and interpretation efforts. By addressing these challenges, the EEG Data Visualization project aims to democratize EEG data analysis, fostering broader accessibility and facilitating insightful exploration of brain activity patterns.

5.2 MODULES DESCRIPTION :

Data Processing :

Processes EEG signals, applying bandpass filtering to isolate desired frequency bands. Computes correlation between EEG channels and calculates. entropy of signals .Provides functions for basic preprocessing and feature extraction from EEG data. Enables initial preparation of EEG data for machine learning tasks.

1.Filtering Data:

Implements functions to apply bandpass filtering to EEG signals, Which helps remove unwanted frequencies.

2.Correlation Calculation:

Provides methods calculate the correlation between two signals, which is useful for analyzing relationships between EEG

3.Entropy Calculation:

Computes the entropy of a signal, which measures unpredictability disorder within the signal's distribution of values.

Plotting :

Visualizes topographic plots of EEG data to represent spatial distribution. Customizes plots with contouring, scatter plots, and annotations for electrode positions. Facilitates the visualization of EEG signals and electrode configurations. Offers a standardized format for visualizing EEG data for analysis and interpretation.

1.Topomap Visualization :

Constructs topographic plots of EEG data, displaying spatial distributions of signal power across electrode positions.

2.Customized Plotting:

Allows customization of plots, including contouring, scatter plots, and adding annotations to represent EEG electrode positions.

3.Visualization Utility:

Provides a convenient way to visualize EEG data and electrode position in standardized format.

Filtering :

Implements Butterworth bandpass filtering to remove unwanted frequency components. Enhances the quality of EEG data by applying configurable filter parameters. Provides functions for signal processing tasks such as noise reduction and artifact removal. Enables precise control over filtering parameters for optimal EEG signal extraction.

1.Butterworth Filtering:

Implements Butterworth bandpass filtering, a common technique signal processing for isolating specific frequency bands

2. Signal Filtering:

Enables the removal of unwanted frequency components from EEG signals, enhancing the quality of data for analysis.

3.Configurable Filtering:

Offers flexibility in configuring filter parameters such as cutoff frequencies and filter order to suit different EEG signal.

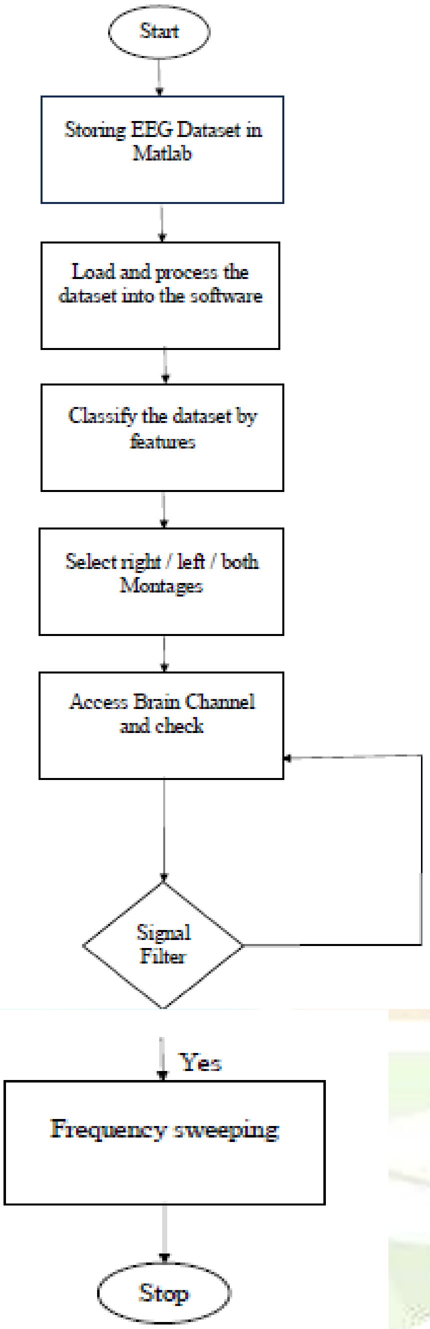
Analysis:

Computes correlation coefficients to identify patterns between EEG channels.Calculates entropy to quantify the complexity of EEG signals. Visualizes EEG data through power spectral density plots, histograms, and scatter plots. Provides insights into EEG signal characteristics and supports data-driven analysis.

1.Correlation Analysis: Computes the correlation coefficient between two signals, helping to identify patterns or relationships between different EEG channels.

2.Entropy Analysis: Calculates the entropy of a signal to quantify its randomness or complexity, providing insights into the information content of EEGdata.

3.Signal Visualization: Provides functions to visualize EEG signals through power spectral density (PSD) plots, histograms, and scatter plots, aiding in the exploration and interpretation of EEG data.



5.3 Input Design:

For the input design of your Streamlit application, you can create an intuitive and user-friendly interface using Streamlit's built-in widgets. Begin by allowing users to select a session from a dropdown menu, providing a list of available sessions. Next, offer options for selecting channels and frequency bands, enabling users to customize their visualization preferences. Utilize checkboxes or select boxes for these options to facilitate easy selection. Additionally, Incorporate a chechbox to allow users to choose whether to Display the topological map of the brain.Ensure that the Interface is responsive and provides clear instructions to guide users through the selection process.

5.4 Output Design :

In the output design, focus on presenting the EEG data visualization in an informative and visually appealing manner. Begin by displaying the first five channels of the selected session as a dataframe, providing users with an overview of the raw data. If the user chooses to show the topographical map, render the map using matplotlib and display it within the Streamlit app. Additionally, generate a line chart to visualize the data of the selected channel within the specified frequency band. Ensure that the chart includes axis labels and a title to provide context. Provide clear annotations to highlight important features or trends in the data. Finally, include a markdown section to cite the source of the dataset used in the project, providing transparency and credibility. Overall, aim for a clean and organized layout that enables users to interpret the EEG data effectively. Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the abstract or reference list. Use letters for table footnotes.

VI.SYSTEM MAINTENANCE

6.1 CORRECTIVE MAINTENANCE:

This involves addressing and rectifying any bugs, errors, or issues that arise in the system after deployment. For the EEG Data Visualization project, corrective maintenance would involve identifying and fixing any software bugs or glitches that affect the functionality or performance of the Streamlit web application, app.py, visu.py, and streamlit.py. It includes debugging code, resolving runtime errors, and ensuring the smooth operation of the application to maintain.

6.2 ADAPTIVE MAINTENANCE:

As the project evolves and user requirements change, adaptive maintenance becomes crucial to ensure that the system remains compatible with new environments, technologies, or platforms. In the context of the EEG Data Visualization project, adaptive maintenance may involve updating the application to support new versions of Streamlit or incorporating compatibility with different EEG headset models. It aims to adapt the system to evolving user needs and technological advancements, enhancing its longevity and relevance.

6.3 PERFECTIVE MAINTENANCE:

This type of maintenance focuses on improving the system's functionality, performance, and usability based on user feedback and evolving requirements. For the EEG Data Visualization project, perfective maintenance may involve enhancing the user interface of the Streamlit web application to provide a more intuitive and seamless experience for data visualization and analysis. It could also include optimizing code efficiency, adding new features requested by users, or refining existing functionalities to improve overall user satisfaction and productivity.

VII. SYSTEM TESTING AND IMPLEMENTATION

7.1 SYSTEM TESTING :

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the requirement.

7.1.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test.Each test type addresses a specific testing application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the document.

7.1.2 INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

7.1.3 FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals. Functional testing is centered on the following items:Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows;

data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests.

7.2 SYSTEM IMPLEMENTATION :

System Test :

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizingpre- driven process links and integration points. outputs without considering how the software works.

White Box Testing :

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing :

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box. you cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works.

Unit Testing :

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

VII. CONCLUSION AND FUTURE WORK

8.1 CONCLUSION :

In conclusion, the EEG Data Visualization project represents a significant advancement in the field of neurophysiology by addressing several challenges associated with EEG data analysis and interpretation. By developing a user-friendly web application using Streamlit, the project enables researchers and enthusiasts to explore EEG signals intuitively. The integration of interactive topographical maps and frequency band selection enhances the system's utility and accessibility. Furthermore, ongoing maintenance efforts ensure the system's reliability, adaptability, and continued improvement over time. Overall, this project contributes to democratizing EEG data analysis, fostering broader access, and facilitating insightful exploration of brain activity patterns.

8.2 FUTURE WORK:

Further development opportunities for the EEG Data Visualization project include expanding compatibility with a wider range of EEG headset models, enabling broader accessibility and applicability. Enhancements in data preprocessing techniques, such as artifact removal and signal denoising algorithms, could improve the quality and accuracy of EEG data analysis. Integration with cloud-based storage and collaboration platforms would facilitate data sharing and collaborative research efforts. Additionally, implementing immersive visualization technologies, such as virtual reality or augmented reality, could offer new perspectives for exploring and interacting with EEG data, enhancing user engagement and understanding in research and educational contexts.

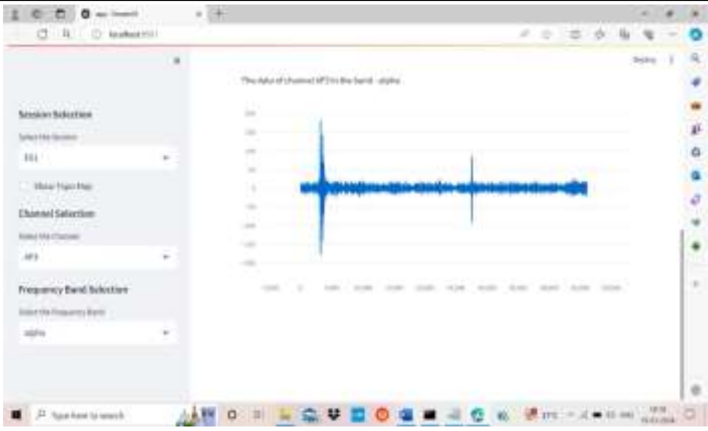


Fig 8.1 Output in Frequency

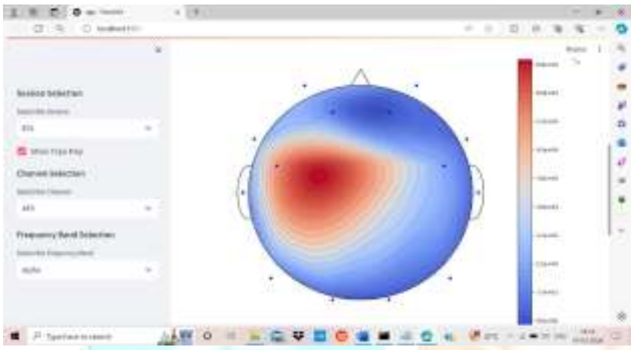


Fig 8.2 Output in Topography

REFERENCES

[1] A. Bhattacharyya, R. K. Tripathy, L. Garg, and R. B. Pachori,“A novel multivariate-multiscale approach for computing EEG spectral and temporal complexity for human emotion recognition,” IEEE Sensors J., vol. 21, no. 3, pp., Feb. 2021.

[2] C. Ieracitano, N. Mammone, A. Hussain, and F. C. Morabito, “A novel multi-modal machine learning based approach for automatic classification of EEG recordings in dementia,” Neural Netw., vol. 123, pp. 176–190, Mar. 2020.

[3] J. W. Li et al., “Single-channel selection for EEG-based emotion recognition using brain rhythm sequencing,” IEEE J. Biomed. Health Informat., vol. 26, no. 6, pp. 2493–2503, Jun. 2022.

[4] R. B. Pachori, “A multivariate approach for patient-specific EEG seizure detection using empirical wavelet transform,” IEEE Trans. Biomed. Eng., vol. 64, no. 9, pp. 2015

[5] N. Mammone, A. Hussain, and F. C. Morabito,“A novel multi-modal machine learning based approach for automatic classification of EEG recordings in dementia,” Neural Netw., vol. 123,pp. 176–190, Mar. 2020.

[6] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, “EEGNet: A compact convolutional neural network for EEG-based brain–computer interfaces,” J. Neural Eng., vol. 15, no. 5, Oct. 2018.

[7] X. Hong et al., “Dynamic joint domain adaptation network for motor imagery classification,” IEEE Trans. Neural Syst. Rehabil. Eng., vol. 29, pp. 556–565, 2021.

[8] Wu, W.-L. Zheng, Z. Li, and B.-L. Lu, “Investigating EEG-based functional connectivity patterns for multimodal emotion recognition,”J. Neural Eng., vol. 19, no. 1, Feb. 2022.

[9] YonghaoSong , Qingqing Zheng , Bingchuan Liu , Xiaorong Gao,” Convolutional Transformer for EEG Decoding and Visualization”. May-31,2023.

[10] Yu Chen1 ,Xiuxiu Hu2 , Lihua Xia3 “Local-Global Graph Convolutional Network for Depression Recognition using EEG Signals “.June-2022.