**Enhancing Epileptic Seizure Detection Accuracy Using YOLOv11 Classification**

**Abstract:**

Epilepsy is a long-term neurological condition that causes repeated seizures, demands precise and timely detection to facilitate effective treatment and management. This study introduces an innovative approach for automated seizure detection using the CHB-MIT Scalp EEG Database, a widely utilized resource of pediatric EEG recordings The CHB-MIT Scalp EEG Database is a widely recognized dataset used for seizure detection and classification, comprising EEG recordings from 23 pediatric patients aged 1.5 to 22 years. Collected using the International 10-20 electrode system at a 256 Hz sampling rate, the dataset includes 182 annotated seizures stored in EDF format. To enhance its usability for deep learning applications, we developed a preprocessing pipeline that converts raw EEG signals into image representations, enabling the use of computer vision-based models such as YOLOv11. The dataset, consisting of 6,579 labeled images (seizure and non-seizure), was augmented using techniques like brightness adjustment, grayscale conversion, and noise injection. The images were split into training (92%), validation (4%), and test (4%) subsets. Our YOLOv11-based model achieved an accuracy of 98.8%, precision of 98.7%, recall of 98.8%, and an F1-score of 98.7%, demonstrating its effectiveness in seizure classification. These results highlight how deep learning can help automate seizure detection, making early diagnosis and treatment for epilepsy patients more effective.

**Keywords**: YOLOv11, Seizure Detection, Medical Imaging, Automated Diagnosis,

# **Introduction**

Over 50 million individuals worldwide suffer from epilepsy, a chronic neurological condition characterized by unexpected seizures that seriously jeopardize patient safety and quality of life [1] [2]. Two main ways epilepsy presents itself: focal, which affects just one part of the brain, and non-focal/generalized, which affects several regions. Seizures in 60% of patients are unresponsive to medicine, requiring surgery.[3] [4].

Timely detection of seizures is critical to prevent injuries, enable rapid intervention, and improve long-term outcomes. Electroencephalography (EEG), the gold standard for monitoring brain activity, provides high-resolution insights into seizure dynamics. However, manual interpretation of EEG signals is labor-intensive, error-prone, and impractical for continuous monitoring due to the sheer volume of data and the subtlety of pre-seizure patterns. This challenge has spurred the development of automated systems leveraging machine learning, yet existing solutions often struggle with real-time performance, noise resilience, and generalization across diverse patient populations [5] [6].

Medical signal analysis has changed as a result of recent developments in deep learning, especially convolutional neural networks (CNNs). You Only Look Once (YOLO) models, originally designed for real-time object detection in images, are now being adapted for temporal and spectral data. The latest iteration, YOLOv11, offers unprecedented speed and accuracy improvements, making it a compelling candidate for seizure detection in EEG. Unlike traditional CNNs, which require manual feature engineering, YOLOv11’s end-to-end architecture can learn discriminative features directly from raw or minimally processed EEG data, reducing bias and computational overhead [7] [8] [9].

This study aims to bridge the gap between theoretical deep learning advancements and clinical applicability by optimizing YOLOv11 for real-time seizure detection. We address three core challenges: (1) the variability in seizure morphology and EEG artifacts, (2) class imbalance between seizure and non-seizure states, and (3) the need for low-latency processing in clinical settings. Our approach integrates domain-specific adaptations, such as temporal-aware loss functions and noise-resistant data augmentation, to enhance model robustness. By validating the system across multiple EEG datasets, we demonstrate its potential to serve as a reliable tool for neurologists, enabling faster diagnoses and personalized epilepsy management.

## **Problem Statement**

Epilepsy is a long-term neurological condition that impacts more than 50 million people globally. It’s marked by repeated seizures, which can create serious risks to a person’s safety and overall quality of life. This is why quick and accurate detection of these episodes is so important—it plays a key role in ensuring proper treatment and effective day-to-day management of the condition. Electroencephalography (EEG) is the gold standard for monitoring brain activity, but manual interpretation of EEG signals is labor-intensive, error-prone, and impractical for continuous monitoring due to the volume and complexity of the data. While machine learning and deep learning approaches have shown promise in automating seizure detection, existing solutions often struggle with real-time performance, noise resilience, and generalization across diverse patient populations. By utilizing the sophisticated features of YOLOv11, a cutting-edge deep learning model, this work seeks to overcome these obstacles and improve the precision and effectiveness of automated seizure detection.

## **Research Gap**

Despite significant advancements in deep learning for seizure detection, several gaps remain. Many existing models prioritize accuracy over latency, making them unsuitable for real-time monitoring in clinical settings. Additionally, most models are validated on single datasets, limiting their ability to generalize across diverse patient populations and recording conditions. While deep learning models automate feature extraction, many still rely on manual preprocessing steps, introducing bias and reducing efficiency. EEG signals are often contaminated with noise and artifacts, which can degrade the performance of automated detection systems. Furthermore, seizure events are rare compared to non-seizure events, leading to models that are biased toward the majority class and less sensitive to seizure detection. This study addresses these gaps by proposing an optimized YOLOv11 framework tailored for EEG-based seizure detection, incorporating noise-resistant data augmentation, temporal-aware loss functions, and dynamic weighting to improve robustness and real-time performance.

## **Research Questions**

This study seeks to answer several key research questions. First, how can YOLOv11 be adapted for EEG-based seizure detection to achieve high accuracy and real-time performance? Second, what preprocessing and augmentation techniques are most effective for improving the robustness of seizure detection models? Third, how does the proposed YOLOv11 model perform compared to existing methods in terms of accuracy, precision, recall, and F1-score? Fourth, can the model generalize across diverse patient populations and recording conditions, as validated on multiple EEG datasets? Finally, what are the key challenges and limitations in deploying deep learning models for real-time seizure detection in clinical settings? By addressing these questions, this study aims to advance the field of automated seizure detection and improve patient outcomes.

## **Contributions**

This study makes several key contributions to the field of epilepsy detection and deep learning. First, it introduces the novel application of YOLOv11 to EEG-based seizure detection, leveraging its real-time object detection capabilities for temporal EEG data. The integration of domain-specific adaptations, such as temporal-aware loss functions and noise-resistant data augmentation, enhances model robustness. Second, a preprocessing pipeline is developed to convert raw EEG signals into image representations, enabling the use of computer vision-based models like YOLOv11. Advanced augmentation techniques, including brightness adjustment, grayscale conversion, and noise injection, are applied to improve dataset variability and model generalization. Third, the study achieves state-of-the-art performance metrics, including 98.8% accuracy, 98.7% precision, 98.8% recall, and 98.7% F1-score on the CHB-MIT Scalp EEG Database. The model demonstrates the ability to handle class imbalance and noise, making it suitable for real-world clinical applications. Fourth, the optimization of YOLOv11 for low-latency processing enables real-time seizure detection in clinical settings, with validation on multiple EEG datasets ensuring generalizability across diverse patient populations. Fifth, the study promotes open science by publicly releasing the preprocessing pipeline, trained models, and code to foster reproducibility and further research. Finally, the clinical impact of this work is significant, as it has the potential to improve early diagnosis and intervention for epilepsy patients by enabling continuous, real-time monitoring in hospitals and home settings, reducing diagnostic delays, and improving patient outcomes.

# **Related Work**

Deep learning models have been increasingly popular in recent years in a variety of fields, such as EEG signal processing, where several models have been put out for seizure identification. We examine a few noteworthy deep learning techniques created by scholars in this area below:

Jaafar and Mohammadi [10] developed a deep learning system to detect seizures that skips the usual step of manually identifying key features in the data. They tested their method using the Freiburg dataset—a collection of EEG recordings from 21 patients of varying ages. After cleaning up the data by normalizing and filtering it, they split the EEG signals into short, distinct segments. For detection, they used an LSTM classifier (a type of AI that learns patterns over time) and trained it with a five-part testing method to ensure reliability. The results showed strong performance, highlighting the potential of their streamlined approach.

A seizure detection method using Deep Neural Networks (DNN) with feature scaling was created by Thara et al [11] They tested four distinct feature scaling strategies and loss functions using the Bonn University database. According to their research, standard scalar and robust scalar approaches fared better in terms of accuracy and dependability than the other approaches.

Faust and colleagues [12] created a personalized approach to detect seizures by analyzing visual snapshots of brain activity (called spectrogram images). They started by cleaning raw EEG data with specialized filters, then split the signals into smaller chunks. Using a technique called Fourier transforms, they converted these chunks into visual representations. To classify the data, they designed a simple three-layer neural network with built-in quality checks (batch normalization). While accuracy varied between patients, their method averaged 77.57%—a solid starting point for personalized detection.

In a similar vein, Waqar Hussain’s team [13] built a model to pinpoint seizure phases (ictal, preictal, interictal) tailored to individual patients. They pulled insights from three angles: raw signal patterns, frequency shifts, and time-frequency changes. These features were then analyzed by a hybrid AI system blending a CNN (great for spotting patterns) with an LSTM (ideal for tracking changes over time).

Meanwhile, Xinghua Yao et al.[14] tested a bidirectional LSTM model on the CHB-MIT dataset to distinguish seizure vs. non-seizure brain signals. To ensure their results weren’t a fluke, they rigorously cross-checked performance across all patient data. Rajendra Acharya’s team (2018) took a different route, using a 13-layer CNN to both extract features and classify EEG signals—proving just how versatile deep learning can be in decoding brain activity.

**Table 1** breaks down and compares 18 different seizure detection methods, highlighting the datasets they used, the categories they analyzed (like "Normal" or "Seizure"), and how accurate each approach turned out to be. The studies pulled data from hospitals worldwide—including KK Women and Children’s Hospital in Singapore, CHB-MIT in the U.S., and Freiburg Hospital in Germany—and focused on everything from typical brain activity to pre-seizure warning signs.

The techniques themselves are a mix of cutting-edge machine learning approaches. For example, some teams used classic tools like Convolutional Neural Networks (CNNs), while others got creative with hybrids like 3D-CNNs paired with GRUs (a type of neural network for sequences) or even combined Gramian Angular Fields with Particle Swarm Optimization (a bio-inspired algorithm). There’s also a Temporal Graph Convolutional Network (TGCN) for mapping brain connections over time and a Multivariate Autoencoder with EM-PCA for simplifying complex data.

What stands out is how varied the strategies are—from analyzing spectrograms to tweaking wavelet transforms—but they all share the same goal: catching seizures more reliably. The table shows that while accuracy varies, many of these methods hold real promise for improving patient care. These studies collectively highlight the diversity and innovation in deep learning approaches for seizure detection.

**Table 1:** A comparison of several different techniques of seizure detection.

| Authors | Technique | Dataset | Classes | Accuracy % |
| --- | --- | --- | --- | --- |
| M. Talha et al. [15] | CNN | KK women and children's hospital | Normal-ictal | 93.3 |
| Yang et al. [16] | NLSTM | Boon University | Normal-ictal | 98.44 |
| CHB-MIT | 97.47 |
| Covert et al. [17] | TGCN | Boston children's hospital | Normal-ictal | 98.05 |
| B. Bouaziz et al. [18] | CNN | CHB-MIT | Normal-ictal | 99.48 |
| Rajaguru et al. [19] | MAE + EM-PCA | CHB-MIT | Normal-ictal | 93.78 |
| Roy et al. [20] | ChronoNet | CHB-MIT | Normal-ictal | 86.57 |
| Choi et al. [21] | 3D-CNN + GRU | CHB-MIT | Normal-ictal | 89.4 |
| SNUH | 97 |
| Truong et al. [22] | Spectrograms + STFT | Freiburg hospital intra-cranial EEG | Normal-ictal | 81.4 |
| CHB-MIT | 81.2 |
| Subasi et al. [23] | GA + PSO | Boon University | Normal-ictal | 99.38 |
| M. Zhou et al. [24] | CNN | CHB-MIT | Ictal-pre-ictal  Normal-ictal  Normal-ictal-pre-ictal | 95.6  97.5  93 |
| Freiburg | Ictal-pre-ictal  Normal-ictal  Normal-ictal-pre-ictal | 96.7  95.4  92.3 |
| Qaisar et al. [25] | CNN | Andrzejak | Normal-ictal-pre-ictal | 96.4 |
| Hassan et al. [26] | TQWT | Boon University | Normal-ictal-pre-ictal | 98.4 |
| Hassan et al. [27] | CEEMDAN + NIG | Boon University | Normal-ictal-pre-ictal | 97.6 |
| Sharma et al. [28] | 2D PSRs + EMD + LS-SVM | CHB-MIT | Normal-ictal | 98.67 |
| Shankar et al. [29] | PSR + CNN | Boon University | Normal-ictal | 93 |
| CHB-MIT | Normal-ictal | 85 |
| [Fatma E. Ibrahim](https://onlinelibrary.wiley.com/authored-by/Ibrahim/Fatma+E.) et.al. [30] | Spectrogram images  five-layer CNN | CHB-MIT | Normal-ictal  Normal-pre-ictal  Normal-ictal-pre-ictal | 91.28  92.49  90.21 |

# **Methodology**

## **Preliminaries:**

The YOLO (You Only Look Once) series has been a leader in real-time object detection since its introduction by Redmon et al. [31] Known for its efficiency in predicting bounding boxes and class probabilities in a single network pass, YOLO has evolved significantly from YOLOv1 to YOLOv11, enhancing speed and accuracy for applications like autonomous vehicles, surveillance, healthcare, and agriculture as shown in **Table 2**. Early versions (YOLOv2, YOLOv3) introduced multi-scale feature extraction and advanced training strategies, while later iterations (YOLOv4-YOLOv6) focused on balancing computational efficiency and precision using techniques like mosaic data augmentation and CSPNet. Versions YOLOv7-YOLOv9 improved adaptability across hardware, and YOLOv10-YOLOv11 integrated advanced deep learning methods like attention mechanisms and transformer-inspired components. However, challenges remained in detecting small, occluded, or overlapping objects in real time.

## **YOLOv11: Revolutionizing Real-Time Object Detection**

YOLOv11 marks a transformative leap in real-time object detection, introducing a paradigm shift through its innovative integration of attention-based mechanisms, refined architectural designs, and optimized training pipelines. Building on the strong legacy of its predecessors, YOLOv11 introduces a series of enhancements designed to maximize both accuracy and computational efficiency. Central to its architecture is a reimagined feature extraction strategy that incorporates the Residual Efficient Layer Aggregation Network (R-ELAN), FlashAttention, and 7×7 separable convolutions, as depicted in Figure 1. These advancements enable YOLOv11 to achieve exceptional throughput and precision, setting new standards in object detection and instance segmentation tasks. The model excels in handling complex visual scenes, even in scenarios with varying levels of detail and occlusion.

A defining feature of YOLOv11 is its ability to adapt to challenging detection environments. Its advanced area attention module, powered by FlashAttention, allows the model to effectively identify and focus on critical regions within cluttered or dynamic settings. This capability enhances the localization of objects, including those that are small, partially obscured, or overlapping, ensuring robust performance in demanding scenarios.

**Table 2: Evolution of YOLO Frameworks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Release | Year | Tasks | Contributions | Framework |
| YOLOv1 [32] | 2015 | Object Detection, Basic Classification | Single-stage object detector | Darknet |
| YOLOv2 [33] | 2016 | Object Detection, Enhanced Classification | Multi-scale training, dimension clustering | Darknet |
| YOLOv3 [34] | 2018 | Object Detection, Multi-scale Object Detection, Basic Tracking | SPP block, Darknet-53 backbone | Darknet |
| YOLOv4 [35] | 2020 | Object Detection, Instance Segmentation | Mish activation, CSPDarknet-53 backbone | Darknet |
| YOLOv5 [36] | 2020 | Object Detection, Instance Segmentation | Anchor-free detection, SWISH activation, PANet | PyTorch |
| YOLOv6 [37] | 2022 | Object Detection, Tracking, Segmentation | Self-attention, anchor-free OD | PyTorch |
| YOLOv7 [38] | 2022 | Object Detection, Instance and Panoptic Segmentation | Transformers, E-ELAN reparameterization | PyTorch |
| YOLOv8 [39] | 2023 | Object Detection, Instance Segmentation | GANs, anchor-free detection | PyTorch |
| YOLOv9 [40] | 2024 | Object Detection | PGI and GELAN | PyTorch |
| YOLOv10 [41] | 2024 | Object Detection, Instance Segmentation | Consistent dual assignments for NMS-free training | PyTorch |
| YOLOv11 [42] | 2024 | Object Detection, Instance Segmentation | Expanded capabilities, improved efficiency | PyTorch |

YOLOv11 maintains the real-time speed of its predecessors, making it ideal for latency-sensitive applications like autonomous navigation and urban surveillance. With improved object detection performance, it pushes the boundaries of computer vision capabilities.

## **Architectural Blueprint of YOLOv11**

YOLO’s success is built on its unified architecture, allowing seamless bounding box regression and object classification through end-to-end training. YOLOv11 enhances this framework with innovations designed to improve accuracy, reduce latency, and increase adaptability. As shown in **Table 3**, its structure consists of three key components: the backbone for multi-scale feature extraction, the neck for feature refinement, and the head for final predictions.

## **Backbone**

The backbone of YOLOv11 plays a vital role in transforming raw image data into multi-scale feature maps, forming the foundation for accurate object detection. At its core, the Residual Efficient Layer Aggregation Network (R-ELAN) integrates deeper convolutional layers with strategically placed residual connections. This architecture mitigates gradient bottlenecks and enhances feature reuse, improving the model’s ability to detect objects of varying sizes and shapes with greater precision. **Figure 1 illustrates the Architecture comparison of popular modules across YOLO versions** [43][44]**.**

|  |  |
| --- | --- |
|  |  |
|  | |

**Figure 1: Architecture comparison of popular modules across YOLO versions:** (a) **CSPNet** [45] – utilized in YOLOv4/YOLOv5; (b) **ELAN** [46] – employed in YOLOv8; (c) **C3K2 (a case of GELAN)** [47] – implemented in YOLOv11.

**Table 3**: Core Architectural Elements of YOLOv11 [48]

|  |  |  |
| --- | --- | --- |
| Component | Functionality | Innovations in YOLOv11 |
| Backbone | Extracts multi-scale features from input images using convolutional layers. | Introduces R-ELAN for enhanced residual connectivity and 7×7 separable convolutions to maintain spatial context with reduced parameters. |
| Neck | Aggregates and transmits multi-scale features to the head for predictions. | Implements area attention mechanisms driven by FlashAttention, enabling efficient focus on critical regions. |
| Head | Produces final predictions, including bounding box coordinates and class labels. | Features refined prediction pathways for improved multi-scale detection and optimized loss functions for real-time performance. |

## **Advanced Convolutional Blocks**

Compared to earlier versions, YOLOv11 employs a new convolutional block class emphasizing lightweight operations and higher parallelization. These blocks utilize a series of smaller kernels, represented generically as [48]:

|  |  |
| --- | --- |
|  | (1) |

where *F*out is the output feature map, *Wi* is the convolutional filters, *F*in is the input feature map, and *bi* is the bias term. By distributing the computation across multiple small convolutions instead of fewer large ones, YOLOv11 achieves faster processing without compromising feature extraction quality.

## **Enhanced Backbone Architecture**

Beyond introducing advanced convolutional blocks, YOLOv11 leverages techniques like *7x7 separable convolutions* to reduce the computational burden. This approach effectively replaces conventional large-kernel operations or positional encodings, maintaining spatial awareness with fewer parameters. Additionally, multi-scale feature pyramids ensure that objects of varied sizes, including small or partially occluded ones, are represented distinctly within the network.

×

## **Neck**

Functioning as a conduit between the backbone and head, the neck in YOLOv11 aggregates and refines multi-scale features. One of its key innovations is an *area attention* mechanism accelerated by FlashAttention, which enhances the model’s focus on critical regions in cluttered scenes. Mathematically, this can be interpreted as a segmented attention operation [48]:

|  |  |
| --- | --- |
|  | (2) |

where *Q, K, and V* are query, key, and value matrices, and *dk* is the dimensionality of the key. By segmenting feature maps into areas and applying fast attention routines, YOLOv11 reduces memory transfers and computational overhead, enabling real-time inference even at higher input resolutions.

## **Head**

The head of YOLOv11 transforms the refined feature maps from the neck into final predictions, generating bounding box coordinates and classification scores. Key improvements include streamlined multi-scale detection pathways, and specialized loss functions that better balance localization and classification objectives. For example, a typical YOLO-style loss might be extended to incorporate new attention or confidence terms [48]:

|  |  |
| --- | --- |
|  | (3) |

where *x*ˆ*, y*ˆ*, and C*ˆ denote predicted bounding box coordinates and confidence, respectively. Such refinements further enhance YOLOv11’s performance in real-time applications.

As shown in **Table 4**, YOLOv11 is designed for high efficiency and adaptability in modern computer vision tasks. By integrating advanced architectural innovations and attention mechanisms, it ensures real-time performance while expanding its applicability across various industries.

**Table 4:** YOLOv11 Key Architectural Features [48]

| Feature | Technical Details | Benefits |
| --- | --- | --- |
| Enhanced Backbone Architecture | - Utilizes R-ELAN with deeper residual links.  - Employs 7×7 separable convolutions for efficient spatial encoding.  - Incorporates multi-scale feature pyramids. | - Improved capture of small or complex objects.  - Faster, more accurate feature extraction.  - Greater robustness under varied scene complexities. |
| Advanced Attention Mechanisms | - Implements area-based FlashAttention to reduce computational overhead.  - Applies channel and spatial weighting for refined feature extraction.  - Utilizes context-aware dynamic weighting. | - Enhanced focus on salient regions.  - Improved detection in cluttered or dynamic environments.  - Fewer false positives via targeted weighting. |
| Optimized Neck Design | - Integrates enhanced feature aggregation strategies.  - Uses depthwise separable layers to lower computational cost.  - Supports flexible up/down-sampling processes. | - Superior multi-scale feature integration.  - Lower computational overhead and faster inference.  - Better adaptability for different object sizes. |
| Refined Head Modules | - Expands receptive fields for improved contextual cues.  - Utilizes non-linear activations (e.g., SiLU) to boost expressivity.  - Fine-tunes bounding box regression. | - Higher precision in object localization.  - More robust classification under varied conditions.  - Smoother training convergence. |
| Parameter Optimization | - Deploys lightweight convolutional blocks to reduce trainable parameters.  - Applies pruning and quantization for edge deployment.  - Streamlines architecture for memory efficiency. | - Reduced model footprint for resource-constrained devices.  - Maintains competitive accuracy with fewer parameters.  - Scalable performance across hardware platforms. |
| Enhanced Training Pipeline | - Employs advanced data augmentations (e.g., Mosaic, MixUp).  - Utilizes dynamic learning rate schedules with high-performance optimizers.  - Leverages transfer learning from large-scale datasets. | - Elevated model generalization and robustness.  - Faster convergence with diverse data distributions.  - Consistent performance in real-world conditions. |

# **The proposed framework**

## **Dataset Description**

The **CHB-MIT Scalp EEG Database** available at (<https://physionet.org/content/chbmit/1.0.0/>) is a widely recognized and extensively used dataset in the field of neuroscience and biomedical signal processing. It contains electroencephalogram (EEG) recordings collected from pediatric patients with intractable epilepsy, primarily for the purpose of seizure detection and analysis. The database is a collaborative effort between **Boston Children's Hospital (CHB)** and the **Massachusetts Institute of Technology (MIT)**, and it has become a benchmark resource for researchers developing algorithms for seizure prediction, detection, and classification.

#### **Key Features of the Database**

1. **Patient Demographics**:
   * The dataset includes EEG recordings from **23 pediatric patients** (5 males, 17 females, and 1 of unspecified gender).
   * The age range of the patients is between **1.5 and 22 years**.
2. **Recording Details**:
   * The EEG signals were recorded using the **International 10-20 system** for electrode placement.
   * Each recording consists of **23 channels** of EEG data, sampled at **256 Hz**.
   * The recordings vary in duration, ranging from **1 hour to several hours**, and include both **ictal (seizure)** and **interictal (non-seizure)** periods.
3. **Seizure Annotations**:
   * Seizure events are annotated by clinical experts, providing precise start and end times for each seizure.
   * The database contains a total of **182 seizures** across all patients.
4. **Data Format**:
   * The data is stored in **EDF (European Data Format)**, a standard format for EEG recordings.
   * Annotations are provided in separate files, detailing the timing and type of events (e.g., seizures, artifacts).
5. **Clinical Relevance**:
   * The dataset is particularly valuable for studying **intractable epilepsy**, a condition where seizures cannot be controlled with medication.
   * It is widely used for developing and evaluating algorithms for **seizure detection**, **prediction**, and **classification**.

#### **Applications of the CHB-MIT Scalp EEG Database**

1. **Seizure Detection**:
   * The dataset is a benchmark for developing machine learning and signal processing algorithms to detect seizures in real-time or offline analysis.
2. **Seizure Prediction**:
   * Researchers use the dataset to identify pre-seizure patterns and develop predictive models to anticipate seizures before they occur.
3. **Feature Extraction and Classification**:
   * The database is used to extract meaningful features from EEG signals and classify them into seizure and non-seizure categories.
4. **Algorithm Validation**:
   * The dataset serves as a standard for validating the performance of new algorithms against established methods.
5. **Neurological Research**:
   * It provides insights into the brain's electrical activity during seizures, aiding in the understanding of epilepsy and related disorders.

#### **Dataset Structure**

The CHB-MIT Scalp EEG Database is organized as follows:

* **Patient Folders**: Each patient has a dedicated folder containing their EEG recordings and annotation files.
* **Recording Files**: EEG recordings are stored in EDF format, with each file representing a continuous segment of data.
* **Annotation Files**: Seizure events and other annotations are provided in separate files, detailing the timing and type of events.

#### **Preprocessing and Augmentation (for Computer Vision Applications)**

This dataset, **CHB-MIT Scalp EEG Database Expo - v1**, was exported via **Roboflow.com** on **March 5, 2025, at 10:21 AM GMT**. Roboflow is a comprehensive computer vision platform designed to streamline the development of computer vision projects. It offers tools for collaboration, image collection and organization, dataset annotation, model training, and deployment, as well as active learning to iteratively improve datasets over time.

The dataset is derived from the **CHB-MIT Scalp EEG Database**, a well-known resource for EEG (electroencephalogram) data, and has been adapted for computer vision tasks. It includes pre-processed and augmented images, making it suitable for training and evaluating machine learning models for EEG-related applications such as seizure detection, brain-computer interfaces, and other neurological studies.

### Key Features of the Dataset

#### **Dataset Overview**

* **Total Images**: 6,579 images.
* **Annotations**: Objects related to the **CHB-MIT Scalp EEG Database** are annotated in folder format.
* **Pre-Processing**:
  + **Auto-Orientation**: Pixel data was auto-oriented, and EXIF orientation metadata was stripped.
  + **Resizing**: Images were resized to **640x640 pixels** using a stretch method.
* **Augmentations**:
  + **Brightness Adjustment**: Random brightness adjustments between **-15% and +15%** were applied.
  + **Noise**: Salt-and-pepper noise was added to **0.1% of pixels**.
  + **Outputs per Training Example**: Each source image was augmented to create **3 versions**.

#### **Dataset Splits**

Table 5: Three subsets of the dataset are separated for testing, validation, and training:

|  |  |  |
| --- | --- | --- |
| **Split** | **Percentage** | **Number of Images** |
| **Training Set** | 92% | 6,072 |
| **Validation Set** | 4% | 255 |
| **Test Set** | 4% | 252 |

#### **Additional Augmentations Applied**

Table 6: To enhance the dataset's robustness and variability, the following augmentations were applied:

|  |  |
| --- | --- |
| **Augmentation** | **Details** |
| **Grayscale** | Applied to **15% of images** |
| **Saturation Adjustment** | Adjusted between **-25% and +25%** |
| **Brightness Adjustment** | Adjusted between **-15% and +15%** |
| **Noise Addition** | Added to up to **0.1% of pixels** |

#### **Dataset Link**

* Access the dataset directly at: (<https://universe.roboflow.com/chbmit-scalp-eeg-database/chb-mit-scalp-eeg-database-expo/dataset/1>) .

**Workflow and Algorithm for Processing EEG Data from the CHB-MIT Scalp EEG Database**

This Python-based workflow processes EEG data from the **CHB-MIT Scalp EEG Database** to generate training-ready PNG images for seizure detection. The pipeline converts raw EEG signals into labeled images, distinguishing between seizure and non-seizure events, and organizes them into structured directories for machine learning. Below is a step-by-step breakdown of the process:

**Step 1: Data Acquisition**

* **Input**: Raw EEG data in **EDF (European Data Format)** files and patient-specific summary files (e.g., chb24-summary.txt) from the CHB-MIT database.
* **Purpose**: Source the necessary data files containing EEG recordings and seizure timing metadata.

**Step 2: Metadata Extraction**

* **Action**:
  + Parse the summary files to extract seizure start and end times for each EDF file.
* **Output**: A structured list of seizure intervals for each patient.

**Step 3: EEG Signal Processing**

1. **Channel Selection**:
   * Isolate **18 standard EEG channels** (e.g., FP1-F7, CZ-PZ) for consistency across recordings.
2. **Segmentation**:
   * **Seizure Segments**:
     + Extract EEG data during annotated seizure intervals.
   * **Non-Seizure Segments**:
     + Sample non-overlapping intervals outside seizure periods to ensure balanced representation.
     + Avoid temporal overlaps with seizure events.

**Step 4: Visualization**

* **Action**:
  + Use **Matplotlib** to plot multi-channel EEG signals as PNG images.
  + Vertically offset channels by **100µV** for visual separation.
  + Remove axes, labels, and text to focus on waveform morphology.
  + Save images in high resolution (**300 DPI**) to preserve signal details.
* **Output**: PNG images representing seizure and non-seizure EEG segments.

**Step 5: Dataset Organization**

* **Action**:
  + Save seizure and non-seizure PNG images into separate folders (e.g., seizure\_chb24, non-seizure\_chb24).
* **Purpose**: Organize data for easy access and labeling.

**Step 6: Train/Test/Validation Split**

* **Action**:
  + Randomly shuffle the images and partition them into:
    - **Training Set (92%)**: For model development.
    - **Validation Set (4%)**: For hyperparameter tuning.
    - **Test Set (4%)**: For final evaluation.
  + Maintain identical directory structures for each subset to ensure compatibility with YOLO.
* **Output**: A structured dataset ready for machine learning.

**Block Diagram of the Workflow**

1. **Input**: Raw EDF files and summary metadata.
2. **Processing**:
   * Metadata extraction.
   * EEG signal segmentation (seizure and non-seizure).
   * Visualization of EEG signals as PNG images.
3. **Output**:
   * Organized PNG images in seizure and non-seizure folders.
   * Partitioned into train, test, and validation sets.

|  |
| --- |
| ***Algorithm : EDF to images***  ***Input****: EDF files*  ***Output****: Images (Seizure and Non-Seizure)*  *1. Load EDF files and summary metadata.*  *2. Parse summary files to extract seizure intervals.*  *3.* ***foreach*** *EDF file:*  *a. Select 18 standard EEG channels.*  *b. Extract seizure segments based on annotated intervals.*  *c. Sample non-seizure segments, ensuring no overlap with seizures.*  *4.* ***foreach*** *segment (seizure and non-seizure):*  *a. Plot multi-channel EEG signals using Matplotlib.*  *b. Vertically offset channels by 100µV.*  *c. Remove axes and labels for clean visualization.*  *d. Save as high-resolution PNG images.*  *5. Organize PNG images into folders:*  *a. Separate folders for seizure and non-seizure images.*  *6. Split the dataset:*  *a. Randomly shuffle images.*  *b. Partition into training (92%), validation (4%), and test (4%) sets.*  *7. Save the final structured dataset for YOLO-compatible training.* |

**Figure 2** presents a comprehensive overview of liver disease classes, accompanied by representative sample images for each category.

| **Class Name** | **Sample Images** |
| --- | --- |
| Seizure |  |
| Non-Seizure |  |

**Figure 2:** Sample Images of Seizure and non-seizure.

## The Proposed Model Steps

**Figures 3 and 4** outline the comprehensive workflow for developing a seizure detection system using the YOLO 11 model. The methodology is divided into two main phases: **Data Preparation** and **Model Development**.

The workflow starts by gathering the CHB-MIT Scalp EEG Databasea popular dataset packed with brainwave recordings from children with epilepsy. These raw EEG files are stored in EDF format (a common standard for medical data), which researchers unpack to pull out the actual brain signals and details like timestamps or patient info. Next comes the cleanup phase: the raw signals are scrubbed to remove background noise, blips from movement, or other interference. This polishing step is crucial—it ensures the data is clear and reliable before feeding it into any analysis tools.

The preprocessed EEG signals are then converted into visual representations (images) by plotting the multi-channel EEG data. Each image represents a time window of EEG activity, with channels vertically offset for clarity. These generated images are uploaded to **Roboflow**, a platform for managing and augmenting datasets, facilitating efficient dataset organization and augmentation.

To enhance the dataset's variability and improve model generalization, several augmentation techniques are applied. These include converting images to grayscale to reduce complexity, adjusting saturation levels to simulate different recording conditions, modifying brightness to account for variations in signal intensity, and adding random noise to simulate real-world recording artifacts. The augmented dataset is then split into three subsets: a **training set (92%)** for model training, a **validation set (4%)** for hyperparameter tuning, and a **test set (4%)** for final evaluation.

The YOLO 11 model is employed for seizure detection. YOLO (You Only Look Once) is a state-of-the-art object detection framework adapted here for classification tasks. The model performs feature extraction to identify relevant patterns in the EEG images, followed by classification into two categories: **seizure** and **non-seizure**. A performance matrix comprising measures like accuracy, precision, recall, and F1-score is used to assess the model's performance. This thorough process guarantees the creation of a reliable and accurate seizure detection system.

|  |
| --- |
|  |

**Figure 3:** The Flowchart of the proposed Model.

**Figure 4** illustrates the detailed architecture of the proposed segmentation model, outlining its key components and processing pipeline. The framework is designed to efficiently segment objects within images by leveraging deep learning techniques. It consists of multiple stages, including data preprocessing, feature extraction, segmentation, and post-processing. The model integrates the YOLOv11 technique to enhance segmentation accuracy. The framework ensures robust performance across varying image conditions, making it suitable for real-world applications requiring precise object delineation.

|  |
| --- |
|  |

**Figure 4**. The framework of the proposed segmentation model.

The pseudocode outlines the steps required to build, evaluate, and report on a classification model for seizure detection can be summarized in **Figure 5**.

|  |
| --- |
| ***Algorithm: YOLO Model Training and Evaluation on CHB-MIT Scalp EEG Databas***  ***Input****: Preprocessed EEG images (Seizure and Non-Seizure)* ***Output****: Trained YOLO model and performance metrics*   1. ***Check GPU Availability***    * *Verify GPU resources using a system command to ensure accelerated training.* 2. ***Install Required Libraries***    * *Install necessary Python libraries:*      + *ultralytics (for YOLO).*      + *roboflow (for dataset download).*      + *opencv-python, pandas, matplotlib (for data handling and visualization).* 3. ***Import Required Libraries***    * *Import libraries for data processing, model training, and evaluation.* 4. ***Download Dataset from Roboflow***    * *Initialize Roboflow API with the provided API key.*    * *Access the CHB-MIT Scalp EEG Database project and download the dataset in folder format.* 5. ***Define Dataset Path***    * *Set the path to the downloaded dataset for easy access during training.* 6. ***Initialize the YOLO Model***    * *Load a pre-trained YOLO model (e.g., yolo11n-cls.pt) for classification tasks.* 7. ***Train the Model***    * *Configure the training task as classification.*    * *Set hyperparameters:*      + *epochs=100: Extended training for better convergence.*      + *batch=16: Smaller batch size for efficient gradient updates.*      + *augment=True: Enable data augmentation.*      + *lr0=0.0001: Lower initial learning rate for stability.*      + *lrf=0.00001: Final learning rate decay.*      + *patience=30: Extended patience for early stopping.*    * *Initiate the training process.* 8. ***Display Training Results***    * *Visualize the confusion matrix and training performance graphs.*    * *Check for the existence of result files and display them.* 9. ***Visualize Validation Results***    * *Retrieve validation result images (e.g., val\_batch0\_labels.jpg, val\_batch0\_pred.jpg).*    * *Plot ground truth and predicted labels side-by-side for visual inspection.* 10. ***Predict on Test Data and Calculate Metrics***     * *Iterate through the test dataset directory.*     * *For each image in the test set: a. Use the trained model to predict the class. b. Store the ground truth and predicted labels.*     * *Record the total time taken for predictions.* 11. ***Compute Metrics***     * *Calculate the confusion matrix.*     * *Compute accuracy, precision, recall, and F1-score.*     * *Derive true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) for each class.* 12. ***Display and Save Metrics***     * *Print the confusion matrix and classification report.*     * *Display accuracy, precision, recall, and F1-score.*     * *Save the metrics to a text file, including per-class TP, TN, FP, and FN values.*   ***Output****:*   * *Trained YOLO model for seizure detection.* * *Performance metrics (accuracy, precision, recall, F1-score, confusion matrix).* * *Structured dataset partitioned into training, validation, and test sets.* |

**Figure 5:** Tuned YOLOv11 Algorithm

## Optimal Hyperparameters for YOLOb11 Model Training

The **optimal hyperparameters** for training a YOLO model, specifically tailored for task seizure detection using the CHB-MIT Scalp EEG Database. These values are selected based on best practices and empirical results to achieve high performance. Key hyperparameters, including learning rate, optimizer, and image size, were tuned to optimize model performance. The hyperparameters used for training are summarized in **Table 7.**

**Table 7**: Hyperparameters for YOLOv11 Model Training

| Hyperparameter | Description | Optimal Value |
| --- | --- | --- |
| task | Specifies the task type (e.g., classification, detection). | 'classify' |
| epochs | Number of training epochs. | 100 |
| batch | Batch size for training. | 16 |
| imgsz | Input image size (height and width). | 640 |
| augment | Enables data augmentation during training. | True |
| lr0 | Initial learning rate. | 0.0001 |
| lrf | Final learning rate (learning rate decay factor). | 0.00001 |
| momentum | Momentum for the optimizer. | 0.9 |
| weight\_decay | Weight decay (L2 regularization) for the optimizer. | 0.0005 |
| warmup\_epochs | Number of warmup epochs at the start of training. | 3 |
| warmup\_momentum | Momentum during warmup epochs. | 0.8 |
| warmup\_bias\_lr | Learning rate for bias parameters during warmup. | 0.1 |
| box | Loss weight for bounding box regression. | 0.05 |
| cls | Loss weight for classification. | 0.5 |
| scale | Scale augmentation range (fraction). | 0.5 |
| shear | Shear augmentation range (degrees). | 0.0 |
| perspective | Perspective augmentation range (fraction). | 0.0 |
| flipud | Probability of flipping the image vertically. | 0.0 |
| fliplr | Probability of flipping the image horizontally. | 0.5 |
| mosaic | Enables mosaic data augmentation. | True |
| mixup | Enables mixup data augmentation. | True |
| device | Device to use for training (e.g., GPU, CPU). | '0' (GPU) |
| workers | Number of worker threads for data loading. | 8 |
| optimizer | Optimizer to use (e.g., SGD, Adam). | 'Adam' |
| seed | Random seed for reproducibility. | 42 |
| rect | Enables rectangular training for faster training. | True |
| cos\_lr | Uses cosine learning rate scheduler. | True |
| sync\_bn | Synchronizes batch normalization across GPUs. | False |

## Performance Metrics

To evaluate how well deep learning models work, evaluation measures like F1 score, recall, precision, and accuracy are crucial [49] [50] [51] [52], as shown in **Table 8**. Recall assesses the model's capacity to identify real positive instances, including false negatives, whereas precision calculates the percentage of successfully detected positive samples among all anticipated positives. The F1 score offers a thorough assessment of model performance by striking a balance between precision and recall.

**Table 8.** Standard performance metrics which are quantitatively assessing the effectiveness and accuracy of various machine learning models

|  |  |
| --- | --- |
| **Equation name** | **Equation** |
| Precision (P) |  |
| Recall |  |
| F1 Score |  |
| Accuracy |  |
| Average Precision (AP) |  |

# **Results and Analysis**

The training procedure and performance indicators for the YOLO model applied to the CHB-MIT Scalp EEG Database are thoroughly examined in **Tables 9 and 10**. Key parameters like training loss, validation loss, accuracy, and learning rates are displayed together with the training progress throughout several epochs in **Table 9**. The confusion matrix, precision, recall, F1-score, and overall accuracy of the model are all summarized in **Table 10**. The model's learning processes and its ability to distinguish between seizure and non-seizure events are both revealed by these tables taken together.

**Table 9**: Training Progress Across Epochs

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| epoch | time | train/loss | metrics/accuracy\_top1 | val/loss | lr/pg0 | lr/pg1 | lr/pg2 |
| 1 | 74.9636 | 0.28697 | 0.93254 | 0.33983 | 0.000554204 | 0.000554204 | 0.000554204 |
| 10 | 684.359 | 0.15681 | 0.96032 | 0.12745 | 0.00151697 | 0.00151697 | 0.00151697 |
| 20 | 1367.17 | 0.11821 | 0.96825 | 0.1121 | 0.00135027 | 0.00135027 | 0.00135027 |
| 30 | 2049.07 | 0.09643 | 0.96429 | 0.09262 | 0.00118357 | 0.00118357 | 0.00118357 |
| 40 | 2733.43 | 0.0889 | 0.97222 | 0.0716 | 0.00101688 | 0.00101688 | 0.00101688 |
| 50 | 3414.21 | 0.06399 | 0.97222 | 0.10698 | 0.000850178 | 0.000850178 | 0.000850178 |
| 60 | 4102.38 | 0.05649 | 0.97619 | 0.08864 | 0.00068348 | 0.00068348 | 0.00068348 |
| 70 | 4783.32 | 0.04819 | 0.96825 | 0.09249 | 0.000516782 | 0.000516782 | 0.000516782 |
| 80 | 5471.58 | 0.0378 | 0.97619 | 0.10166 | 0.000350083 | 0.000350083 | 0.000350083 |
| 87 | 5950.09 | 0.03971 | 0.97619 | 0.11081 | 0.000233394 | 0.000233394 | 0.000233394 |

**Table 10**: Model Performance Metrics

|  |  |  |  |
| --- | --- | --- | --- |
| Metric | non-seizure | seizure | Overall |
| Confusion Matrix |  |  |  |
| True Positive (TP) | 234 | 15 |  |
| True Negative (TN) | 15 | 234 |  |
| False Positive (FP) | 2 | 1 |  |
| False Negative (FN) | 1 | 2 |  |
| Precision | 0.99 | 0.94 | 0.9879 |
| Recall | 1.00 | 0.88 | 0.9881 |
| F1-Score | 0.99 | 0.91 | 0.9879 |
| Support | 235 | 17 | 252 |
| Accuracy |  |  | 0.9881 |
| Total Time |  |  | 9.35 seconds |

The training loss decreases steadily from 0.28697 at epoch 1 to 0.03971 at epoch 87, indicating that the model is effectively learning from the data. The top-1 accuracy improves from 93.25% at epoch 1 to 97.62% at epoch 87, demonstrating the model's increasing ability to correctly classify EEG images. The validation loss also decreases significantly, from 0.33983 at epoch 1 to 0.11081 at epoch 87, suggesting that the model generalizes well to unseen data. The learning rates are adjusted dynamically, starting at 0.000554204 and gradually decreasing to 0.000233394, which helps stabilize the training process and avoid overfitting.

The confusion matrix shows that the model achieves high true positive (TP) and true negative (TN) rates, with 234 TP for non-seizure and 15 TP for seizure events. The false positive (FP) and false negative (FN) rates are low, with only 2 FP for non-seizure and 1 FP for seizure events, and 1 FN for non-seizure and 2 FN for seizure events. This indicates that the model is highly accurate in distinguishing between seizure and non-seizure events.

The precision, recall, and F1-score metrics further validate the model's performance. For non-seizure events, the precision is 0.99, recall is 1.00, and F1-score is 0.99. For seizure events, the precision is 0.94, recall is 0.88, and F1-score is 0.91. The overall accuracy of the model is 98.81%, demonstrating its effectiveness in classifying EEG data. The total inference time for the test set is 9.35 seconds, indicating efficient processing.

The model's performance was further validated by comparing actual images with their predicted outcomes. Details are provided in **Figures 7 and 8**, which display sample images alongside their corresponding predictions.

|  |
| --- |
|  |
| **Figure 7: Samples of Truth Images** |
|  |
| **Figure 8: Predicted Results for Sample Images** |

**Figure 9** illustrates the model's learning behavior by plotting the training and validation loss curves. These curves are analyzed to assess potential overfitting or underfitting. The top-1 and Confusion matrix metrics further support this analysis by showing the model's classification accuracy on both training and validation sets.

|  |
| --- |
|  |

**Figure 9**: The model's learning behavior for the training and validation loss curves

## **Discussion and Future Directions**

The study's findings show how well the suggested YOLOv11-based model works for seizure identification with the CHB-MIT Scalp EEG Database. The model's precision, recall, and F1-scores were 0.9879, 0.9881, and 0.9879, respectively, and it attained a high accuracy of 98.81%, respectively. With few false positives and false negatives, these measures show that the model is quite accurate at differentiating between seizure and non-seizure events. This is further supported by the confusion matrix, which displays high true positive and true negative rates both of which are essential for clinical applications where precise detection is crucial.

The training process revealed a steady decrease in both training and validation losses, indicating effective learning and generalization. The top-1 accuracy improved significantly from 93.25% at epoch 1 to 97.62% at epoch 87, demonstrating the model's ability to refine its classification capabilities over time. The dynamic adjustment of learning rates contributed to the stability of the training process, preventing overfitting and ensuring robust performance on unseen data.

The integration of advanced preprocessing techniques, including brightness adjustment, grayscale conversion, and noise injection, played a crucial role in enhancing the dataset's robustness. These augmentations helped the model generalize better to varying conditions, which is essential for real-world applications where EEG signals can be subject to noise and artifacts. The structured dataset split into training, validation, and test sets ensured a rigorous evaluation of the model's performance, providing confidence in its reliability and accuracy.

The findings of this study align with and extend previous research in EEG-based seizure detection. The use of deep learning models, particularly YOLOv11, is a major improvement over conventional machine learning techniques, which can call for a great deal of human feature designing. This work shows how deep learning may be used to create automated, highly accurate seizure detection systems that can function in real-time, providing substantial advantages for patient care and management.

**Table 11** provides a comprehensive comparison of various methodologies employed in EEG-based seizure detection, highlighting their key features, techniques, classifiers/models, performance metrics, and limitations. The evaluated works range from sophisticated deep learning methods like CNNs and the suggested YOLOv11 optimization to more conventional machine learning approaches like SVM and logistic regression. Accuracy, sensitivity, and computing efficiency are performance measures used to assess each method's suitability for handling spectral, temporal, and multi-feature EEG data. The proposed work introduces a novel YOLOv11-based approach, demonstrating superior accuracy and robustness, though it requires further validation on larger, multi-center datasets. This comparison underscores the trade-offs between manual feature engineering, computational complexity, and real-time applicability in EEG analysis.

**Table 11**: Comparative Analysis of EEG-Based Seizure Detection Methodologies: Techniques, Models, Performance, and Limitations

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Study | Methodology | Key Features/Techniques | Classifier/Model | Performance | Limitations |
| Park et al. [53] | Spectral analysis of EEG signals | Power spectral density (PSD) in high gamma band (30–100 Hz) | Cost-sensitive SVM | High interictal/preictal separation | Manual feature engineering; limited to spectral features |
| Pang et al. [54] | Multi-feature extraction across EEG channels | Shannon entropy in 46 frequency bands (60s windows, 50% overlap) | SVM (RBF, KNN), Logistic Regression | Evaluated on human/animal datasets | Computationally intensive; lacks real-time applicability |
| Zhang et al. [55] | Feature fusion and Kalman filtering | Relative power spectrum, cross-correlation coefficients | AdaBoost, SVM, Neural Networks | Robust multi-class classification | Feature redundancy; dataset-specific optimization |
| Truong et al. [56] | Deep learning for raw EEG analysis | Raw EEG data (30s windows, 50% overlap) | CNN | 89.1% sensitivity | High computational latency; sensitive to noise |
| Sharma et al. [57] | Time-frequency and fractal analysis | Wavelet transforms, fractal dimensions | LS-SVM | 98.5% accuracy | Limited generalizability across datasets |
| Acharya et al. [58] | End-to-end deep learning | 10 convolutional + 3 fully connected layers | CNN | 88% accuracy, 95% sensitivity | Overfitting on small datasets; no real-time implementation |
| Proposed Work | YOLOv11 optimization for EEG | Noise-resistant augmentation, temporal-aware loss functions | YOLOv11 (modified) | accuracy of 98.81%, with precision, recall, and F1-scores of 0.9879, 0.9881, and 0.9879, respectively. | Requires validation on larger multi-center datasets |

## **Limitations**

Despite the promising results, several limitations must be acknowledged. Firstly, the reliance on the CHB-MIT Scalp EEG Database, while providing a robust foundation for this study, may limit the generalizability of the findings. The dataset, though comprehensive, may not fully capture the diversity of seizure types, patient demographics, and recording conditions encountered in clinical practice. This could have an impact on how well the model performs when used with larger or more diverse populations.   
Second, the YOLOv11 model's processing requirements make it difficult to implement in situations with limited resources. Significant memory and processing capacity are needed for training and inference, and these resources might not always be available in clinical situations. This could hinder the widespread adoption of the model, particularly in low-resource healthcare systems.

The study's use of a binary classification technique is another drawback. Although this simplification works well for differentiating seizure from non-seizure episodes, it does not fully represent the intricacy and temporal dynamics of actual seizure activity. Multi-class classification techniques should be investigated in future studies to offer more complex and therapeutically applicable findings. Lastly, there is still a problem with how interpretable the model's predictions are. Understanding the fundamental causes impacting deep learning models' judgments can be challenging due to their "black box" nature. Because healthcare practitioners would be hesitant to depend on a system whose decision-making process is difficult to understand, this lack of openness could impede clinical acceptability.

## **Conclusions and Future Work**

This study successfully demonstrated the efficacy of the YOLOv11 model for automated seizure detection using the CHB-MIT Scalp EEG Database. By converting raw EEG signals into visual representations and employing robust data augmentation techniques, we achieved a high classification accuracy of 98.81% on the test dataset. Comprehensive performance metrics, including precision, recall, and F1-score, validate the model's ability to accurately distinguish between seizure and non-seizure events, with minimal false positives and false negatives. The convergence of training and validation loss, along with insights from the confusion matrix, further confirmed the robustness and reliability of our approach. The success of this methodology can be attributed to the meticulous preprocessing pipeline, optimized hyperparameters, and the adoption of the YOLOv11 framework, which collectively ensured high accuracy and efficiency. This work contributes significantly to the growing body of research utilizing deep learning for EEG analysis and offers a promising avenue for developing scalable, real-time seizure detection systems. Such advancements hold immense potential for improving patient outcomes by enabling timely intervention and reducing diagnostic delays.

Future research will focus on extending this methodology to larger and more diverse EEG datasets, incorporating temporal information to enhance prediction accuracy, and exploring real-time deployment scenarios. Also, focus on expanding the diversity of datasets to include a wider range of seizure types and patient populations. Additionally, efforts should be made to optimize the computational efficiency of deep learning models to facilitate their deployment in resource-constrained environments. Additionally, efforts will be directed toward enhancing the interpretability of the model’s predictions, providing clinicians with actionable insights into the underlying EEG patterns associated with seizures. By refining and validating our approach, we aim to contribute to the development of robust, reliable, and clinically deployable tools for epilepsy management, ultimately transforming the landscape of neurological disorder diagnostics and care.

**Declarations**

**Authors' Contributions**

This work was a collaborative effort among all authors. Together, the authors conceptualized the study, conducted the statistical analyses, and developed the protocol. Each author contributed to the writing process and collectively reviewed and approved the final manuscript.

Data and code availability

* The **CHB-MIT Scalp EEG Database** available at ([**https://physionet.org/content/chbmit/1.0.0/**](https://physionet.org/content/chbmit/1.0.0/))**.**
* The dataset in Roboflow of this study is available at: (<https://universe.roboflow.com/chbmit-scalp-eeg-database/chb-mit-scalp-eeg-database-expo/dataset/1>)

**Funding:** N/A

Declaration of competing interest

No conflicts of interest exist, according to the authors. They affirm that there are no known conflicting financial interests or personal ties that may have influenced the work described in this publication.

Top of Form

Disclaimer (Artificial intelligence)

Option 1:

Author(s) hereby declare that NO generative AI technologies such as Large Language Models (ChatGPT, COPILOT, etc.) and text-to-image generators have been used during the writing or editing of this manuscript.

Option 2:

Author(s) hereby declare that generative AI technologies such as Large Language Models, etc. have been used during the writing or editing of manuscripts. This explanation will include the name, version, model, and source of the generative AI technology and as well as all input prompts provided to the generative AI technology

Details of the AI usage are given below:

1.

2.

3.

Bottom of Form

**References**

[1] J. Shi *et al.*, “The Role of Neuroinflammation and Network Anomalies in Drug-Resistant Epilepsy,” *Neurosci. Bull.*, Feb. 2025, doi: 10.1007/s12264-025-01348-w.

[2] B. N. Aljafen, “Exploring the Link Between Environmental Pollution and Epilepsy,” *Ann. PIMS-Shaheed Zulfiqar Ali Bhutto Med. Univ.*, vol. 21, no. 1, Art. no. 1, Jan. 2025, Accessed: Mar. 16, 2025. [Online]. Available: https://www.apims.net/apims/article/view/1427

[3] Chavan, P. A., & Desai, S. (2024). An efficient epileptic seizure detection by classifying focal and non-focal EEG signals using optimized deep dual adaptive CNN-HMM classifier. *Multimedia Tools and Applications*, *83*(19), 57347-57388.

[4] Chavan, P. A., & Desai, S. (2023). Effective epileptic seizure detection by classifying focal and non-focal EEG signals using human learning optimization-based hidden Markov model. *Biomedical Signal Processing and Control*, *83*, 104682.

[5] M. X. Fayzullayevna, “EPILEPSY AND NERVOUS SYSTEM DISORDERS,” *Ethiop. Int. J. Multidiscip. Res.*, vol. 12, no. 02, Art. no. 02, Feb. 2025.

[6] V. Gabeff *et al.*, “Interpreting deep learning models for epileptic seizure detection on EEG signals,” *Artif. Intell. Med.*, vol. 117, p. 102084, Jul. 2021, doi: 10.1016/j.artmed.2021.102084.

[7] D. Ahmedt-Aristizabal *et al.*, “Deep learning approaches for seizure video analysis: A review,” *Epilepsy Behav.*, vol. 154, p. 109735, May 2024, doi: 10.1016/j.yebeh.2024.109735.

[8] F. A. Jibon *et al.*, “Epileptic seizure detection from electroencephalogram (EEG) signals using linear graph convolutional network and DenseNet based hybrid framework,” *J. Radiat. Res. Appl. Sci.*, vol. 16, no. 3, p. 100607, Sep. 2023, doi: 10.1016/j.jrras.2023.100607.

[9] N. M. Pour and Y. Özbek, “Epileptic Seizure Detection based on EEG Signal using Boosting Classifiers,” *Erzincan Univ. J. Sci. Technol.*, vol. 14, no. 1, Art. no. 1, Mar. 2021, doi: 10.18185/erzifbed.893492.

[10] S. T. Jaafar and M. Mohammadi, “Epileptic Seizure Detection using Deep Learning Approach,” *UHD J Sci Technol*, vol. 3, no. 2, pp. 41–50, 2019.

[11] D. K. Thara, B. G. PremaSudha, and F. Xiong, “Auto-detection of epileptic seizure events using deep neural network with different feature scaling techniques,” *Pattern Recognit Lett*, vol. 128, pp. 544–550, 2019.

[12] O. Faust, Y. Hagiwara, T. J. Hong, O. S. Lih, and U. R. Acharya, “Deep learning for healthcare applications based on physiological signals: A review,” *Comput Methods Programs Biomed*, vol. 161, pp. 1–13, 2018.

[13] W. Hussain, M. T. Sadiq, S. Siuly, and A. U. Rehman, “Epileptic seizure detection using 1 D-convolutional long short-term memory neural networks,” *Appl Acoust*, vol. 177, p. 107941, 2021.

[14] “X. Yao, X. Li, Q. Ye, Y. Huang, Q. Cheng, and G.-Q. Zhang, ‘A robust deep learning approach for automatic classification of seizures against non-seizures,’ Biomed Signal Process Control, vol. 64, p. 102215, 2021”.

[15] K. Meddah, H. Zairi, B. Bessekri, H. Cherrih, and M. Kedir-Talha, “FPGA implementation of Epileptic Seizure detection based on DWT, PCA and Support Vector Machine,” in *2020 Second International Conference on Embedded & Distributed Systems (EDiS)*, 2020, pp. 141–146.

[16] Y. Zhang, Y. Guo, P. Yang, W. Chen, and B. Lo, “Epilepsy seizure prediction on EEG using common spatial pattern and convolutional neural network,” *IEEE J Biomed Health Inf.*, vol. 24, no. 2, pp. 465–474, 2019.

[17] I. C. Covert and others, “Temporal graph convolutional networks for automatic seizure detection,” in *Machine Learning for Healthcare Conference*, 2019, pp. 160–180.

[18] B. Bouaziz, L. Chaari, H. Batatia, and A. Quintero-Rincón, “Epileptic Seizure Detection Using a Convolutional Neural Network,” vol. 10, Springer, 2019, pp. 79–86. doi: 10.1007/978-3-030-11800-6\_9.

[19] H. Rajaguru and S. K. Prabhakar, “Multilayer autoencoders and em-pca with genetic algorithm for epilepsy classification from EEG,” in *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, 2018, pp. 353–358.

[20] S. Roy, I. Kiral-Kornek, and S. Harrer, “ChronoNet: A deep recurrent neural network for abnormal EEG identification,” in *Artificial Intelligence in Medicine: 17th Conference on Artificial Intelligence in Medicine, AIME 2019, Poznan, Poland, June 26–29, 2019, Proceedings 17*, 2019, pp. 47–56.

[21] G. Choi and others, “A novel multi-scale 3D CNN with deep neural network for epileptic seizure detection,” in *2019 IEEE International Conference on Consumer Electronics (ICCE)*, 2019, pp. 1–2.

[22] N. D. Truong and others, “Convolutional neural networks for seizure prediction using intracranial and scalp electroencephalogram,” *Neural Netw*, vol. 105, pp. 104–111, 2018.

[23] A. Subasi, J. Kevric, and M. A. Canbaz, “Epileptic seizure detection using hybrid machine learning methods,” *Neural Comput Appl*, vol. 31, pp. 317–325, 2019.

[24] X. J. Zhou M, Tian C, Cao R, Wang B, Niu Y, Hu T, Guo H, “Epileptic seizure detection based on EEG signals and CNN,” *Front Neuroinformatics*, vol. 12, p. 95, 2018, doi: 0.3389/fninf.2018.00095.

[25] S. M. Qaisar and A. Subasi, “Efficient epileptic seizure detection based on the event-driven processing,” *Procedia Comput Sci*, vol. 163, pp. 30–34, 2019.

[26] A. R. Hassan and A. Subasi, “Automatic identification of epileptic seizures from EEG signals using linear programming boosting,” *Comput Methods Programs Biomed*, vol. 136, pp. 65–77, 2016.

[27] A. R. Hassan, A. Subasi, and Y. Zhang, “Epilepsy seizure detection using complete ensemble empirical mode decomposition with adaptive noise,” *Knowl-Based Syst*, vol. 191, p. 105333, 2020.

[28] R. Sharma and R. B. Pachori, “Classification of epileptic seizures in EEG signals based on phase space representation of intrinsic mode functions,” *Classif Epileptic Seizures EEG Signals Based Phase Space Represent Intrinsic Mode Funct*, vol. 42, no. 3, pp. 1106–1117, 2015.

[29] A. Shankar, H. K. Khaing, S. Dandapat, and S. Barma, “Analysis of epileptic seizures based on EEG using recurrence plot images and deep learning,” *Biomed Signal Process Control*, vol. 69, p. 102854, 2021, doi: 10.1016/j.bspc.2021.102854.

[30] F. E. Ibrahim and others, “Deep-learning-based seizure detection and prediction from electroencephalography signals,” *Int J Numer Methods Biomed Eng*, vol. 38, no. 6, p. e3573, 2022, doi: 10.1002/cnm.3573.

[31] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.

[32] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788. Accessed: Feb. 26, 2025. [Online]. Available: https://www.cv-foundation.org/openaccess/content\_cvpr\_2016/html/Redmon\_You\_Only\_Look\_CVPR\_2016\_paper.html

[33] J. Redmon and A. Farhadi, “YOLO9000: Better, Faster, Stronger,” presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 7263–7271. Accessed: Feb. 26, 2025. [Online]. Available: https://openaccess.thecvf.com/content\_cvpr\_2017/html/Redmon\_YOLO9000\_Better\_Faster\_CVPR\_2017\_paper.html

[34] J. Redmon and A. Farhadi, “Yolov3: An incremental improvement,” *ArXiv Prepr. ArXiv180402767*, 2018, Accessed: Feb. 26, 2025. [Online]. Available: http://175.27.250.89:10000/media/attachment/2024/08/YOLOV3.pdf

[35] A. Bochkovskiy, C.-Y. Wang, and H. Liao, “YOLOv4: Optimal Speed and Accuracy of Object Detection,” *ArXiv*, Apr. 2020, Accessed: Feb. 26, 2025. [Online]. Available: https://www.semanticscholar.org/paper/YOLOv4%3A-Optimal-Speed-and-Accuracy-of-Object-Bochkovskiy-Wang/2a6f7f0d659c5f7dcd665064b71e7b751592c80e

[36] J. Solawetz, “What is Yolov5? A guide for beginners,” *Roboflow Blog*, vol. 29, p. 2020, 2020.

[37] C. Li *et al.*, “YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications,” Sep. 07, 2022, *arXiv*: arXiv:2209.02976. doi: 10.48550/arXiv.2209.02976.

[38] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, “YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors,” presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 7464–7475. Accessed: Feb. 26, 2025. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2023/html/Wang\_YOLOv7\_Trainable\_Bag-of-Freebies\_Sets\_New\_State-of-the-Art\_for\_Real-Time\_Object\_Detectors\_CVPR\_2023\_paper.html

[39] M. Sohan, T. Sai Ram, and Ch. V. Rami Reddy, “A Review on YOLOv8 and Its Advancements,” in *Data Intelligence and Cognitive Informatics*, I. J. Jacob, S. Piramuthu, and P. Falkowski-Gilski, Eds., Singapore: Springer Nature, 2024, pp. 529–545. doi: 10.1007/978-981-99-7962-2\_39.

[40] M. Yaseen, “What is YOLOv9: An In-Depth Exploration of the Internal Features of the Next-Generation Object Detector,” Sep. 12, 2024, *arXiv*: arXiv:2409.07813. doi: 10.48550/arXiv.2409.07813.

[41] A. Wang *et al.*, “YOLOv10: Real-Time End-to-End Object Detection,” *Adv. Neural Inf. Process. Syst.*, vol. 37, pp. 107984–108011, Jan. 2025.

[42] M. Al Rabbani Alif, “YOLOv11 for Vehicle Detection: Advancements, Performance, and Applications in Intelligent Transportation Systems,” Oct. 01, 2024. doi: 10.48550/arXiv.2410.22898.

[43] 田运杰, *sunsmarterjie/yolov12*. (Feb. 26, 2025). Python. Accessed: Feb. 26, 2025. [Online]. Available: https://github.com/sunsmarterjie/yolov12

[44] Y. Tian, Q. Ye, and D. Doermann, “YOLOv12: Attention-Centric Real-Time Object Detectors,” Feb. 18, 2025, *arXiv*: arXiv:2502.12524. doi: 10.48550/arXiv.2502.12524.

[45] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, “CSPNet: A New Backbone That Can Enhance Learning Capability of CNN,” presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020, pp. 390–391. Accessed: Feb. 26, 2025. [Online]. Available: https://openaccess.thecvf.com/content\_CVPRW\_2020/html/w28/Wang\_CSPNet\_A\_New\_Backbone\_That\_Can\_Enhance\_Learning\_Capability\_of\_CVPRW\_2020\_paper.html

[46] C.-Y. Wang, H.-Y. M. Liao, and I.-H. Yeh, “Designing Network Design Strategies Through Gradient Path Analysis,” Nov. 09, 2022, *arXiv*: arXiv:2211.04800. doi: 10.48550/arXiv.2211.04800.

[47] C.-Y. Wang, I.-H. Yeh, and H.-Y. Mark Liao, “YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information,” in *Computer Vision – ECCV 2024*, A. Leonardis, E. Ricci, S. Roth, O. Russakovsky, T. Sattler, and G. Varol, Eds., Cham: Springer Nature Switzerland, 2025, pp. 1–21. doi: 10.1007/978-3-031-72751-1\_1.

[48] M. A. R. Alif and M. Hussain, “YOLOv12: A Breakdown of the Key Architectural Features,” Feb. 20, 2025, *arXiv*: arXiv:2502.14740. doi: 10.48550/arXiv.2502.14740.

[49] G. Mostafa, H. Mahmoud, T. Abd El-Hafeez, and M. E.ElAraby, “The power of deep learning in simplifying feature selection for hepatocellular carcinoma: a review,” *BMC Med. Inform. Decis. Mak.*, vol. 24, no. 1, p. 287, Oct. 2024, doi: 10.1186/s12911-024-02682-1.

[50] A. Ameen, I. E. Fattoh, T. Abd El-Hafeez, and K. Ahmed, “Advances in ECG and PCG-based cardiovascular disease classification: a review of deep learning and machine learning methods,” *J. Big Data*, vol. 11, no. 1, p. 159, Nov. 2024, doi: 10.1186/s40537-024-01011-7.

[51] E. H. I. Eliwa, A. M. E. Koshiry, T. A. El-Hafeez, and A. Omar, “Secure and Transparent Lung and Colon Cancer Classification Using Blockchain and Microsoft Azure,” *Adv. Respir. Med.*, vol. 92, no. 5, p. 395, Oct. 2024, doi: 10.3390/arm92050037.

[52] W. M. Elmessery *et al.*, “Semantic segmentation of microbial alterations based on SegFormer,” *Front. Plant Sci.*, vol. 15, p. 1352935, 2024.

[53] Y. Park, L. Luo, K. K. Parhi, and T. Netoff, “Seizure prediction with spectral power of EEG using cost-sensitive support vector machines,” *Epilepsia*, vol. 52, no. 10, pp. 1761–1770, 2011, doi: 10.1111/j.1528-1167.2011.03138.x.

[54] M. Pang, H. Wang, J. Huang, C.-M. Vong, Z. Zeng, and C. Chen, “Multi-Scale Masked Autoencoders for Cross-Session Emotion Recognition,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 32, pp. 1637–1646, 2024, doi: 10.1109/TNSRE.2024.3389037.

[55] S. Zhang, C. Tang, and C. Guan, “Visual-to-EEG cross-modal knowledge distillation for continuous emotion recognition,” *Pattern Recognit.*, vol. 130, p. 108833, Oct. 2022, doi: 10.1016/j.patcog.2022.108833.

[56] D. Truong, M. A. Khalid, and A. Delorme, “EEG-SSL: A Framework for Self-Supervised Learning on EEG,” in *2024 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, Dec. 2024, pp. 4193–4200. doi: 10.1109/BIBM62325.2024.10822136.

[57] R. R. Sharma and R. B. Pachori, “Time–frequency representation using IEVDHM–HT with application to classification of epileptic EEG signals,” *IET Sci. Meas. Technol.*, vol. 12, no. 1, pp. 72–82, 2018, doi: 10.1049/iet-smt.2017.0058.

[58] B. Ay *et al.*, “Automated Depression Detection Using Deep Representation and Sequence Learning with EEG Signals,” *J. Med. Syst.*, vol. 43, no. 7, p. 205, May 2019, doi: 10.1007/s10916-019-1345-y.