electronics

Article

Visual Explanations of Deep Learning Architectures in Predicting Cyclic Alternating Patterns Using Wavelet Transforms

Ankit Gupta [1,2,*], Fábio Mendonça [1,2], Sheikh Shanawaz Mostafa [1], Antonio G. Ravelo-García [1,3] and Fernando Morgado-Dias [1,2]

[1] Interactive Technologies Institute (ITI/LARSyS and ARDITI), Caminho da Penteada, 9020-105 Funchal, Portugal; fabioruben@staff.uma.pt (F.M.); sheikh.mostafa@tecnico.ulisboa.pt (S.S.M.); antonio.ravelo@ulpgc.es (A.G.R.-G.); morgado@staff.uma.pt (F.M.-D.)

[2] Universidade da Madeira, Caminho da Penteada, 9020-105 Funchal, Portugal

[3] Institute for Technological Development and Innovation in Communications, Universidad de Las Palmas de Gran Canaria, C. Juan de Quesada, 30, 35001 Las Palmas, Spain

[*] Correspondence: ankit.gupta@iti.larsys.pt

Abstract: Cyclic Alternating Pattern (CAP) is a sleep instability marker defined based on the amplitude and frequency of the electroencephalogram signal. Because of the time and intensive process of labeling the data, different machine learning and automatic approaches are proposed. However, due to the low accuracy of the traditional approach and the black box approach of the machine learning approach, the proposed systems remain untrusted by the physician. This study contributes to accurately estimating CAP in a Frequency-Time domain by A-phase and its subtypes prediction by transforming the monopolar deviated electroencephalogram signals into corresponding scalograms. Subsequently, various computer vision classifiers were tested for the A-phase using scalogram images. It was found that MobileNetV2 outperformed all other tested classifiers by achieving the average accuracy, sensitivity, and specificity values of 0.80, 0.75, and 0.81, respectively. The MobileNetV2 trained model was further fine-tuned for A-phase subtypes prediction. To further verify the visual ability of the trained models, Gradcam++ was employed to identify the targeted regions by the trained network. It was verified that the areas identified by the model match the regions focused on by the sleep experts for A-phase predictions, thereby proving its clinical viability and robustness. This motivates the development of novel deep learning based methods for CAP patterns predictions.

Keywords: continuous wavelet transform; cyclic alternating patterns; deep learning; electroencephalogram; signal processing

check for updates

Citation: Gupta , A.; Mendonça, F.; Mostafa, S.S.; Ravelo-García, A.G.; Morgado-Dias, F. Visual Explanations of Deep Learning Architectures in Predicting Cyclic Alternating Patterns Using Wavelet Transforms. *Electronics* **2023**, *12*, 2954. https://doi.org/10.3390/electronics12132954

Academic Editor: Valentina E. Balas

Received: 16 May 2023
Revised: 21 June 2023
Accepted: 3 July 2023
Published: 5 July 2023

1. Introduction

According to the American Academy of Sleep Medicine, the sleep macrostructure is scored with epochs lasting 30 s using multiple signals, including the ones from electroencephalogram (EEG), and oscillates between two primary states: the Rapid Eye Movement (REM) and the non-REM (NREM). Wake periods can also occur during the night [1]. On the other hand, the sleep microstructure is characterized by transient and phasic events in the brain's electrical activity, with a shorter duration than the conventional scoring epoch. Hence, these events are usually evaluated with epochs lasting 1 s. A way of studying this microstructure is through the CAP, which defines these transient and phasic events as an Activation phase (A-phase), lasting between 2 and 60 s. If the duration between two valid A-phases is also in the 2 to 60 s range, then a quiescent phase (B phase) occurs. The cyclic alteration between the A and B phases defines the CAP cycles [2].

The A-phases can be of three subtypes, defined according to the EEG signal's frequency and amplitude characteristics. These include k-complex sequences with or without spindles, delta and polyphasic bursts, vertex sharp transients, k-alpha, EEG arousals, and intermittent alpha. The EEG spectrum is usually subdivided into five bands with identifiable characteristics. These bands are delta (0.5 to 4 Hz), theta (4 to 8 Hz), alpha (8 to 12 Hz), sigma (12 to 15 Hz), and beta (15 to 30 Hz) [2]. For the first subtype (A1), the EEG signal is characterized by synchronized EEG patterns (high-voltage slow waves), where the desynchronized pattern (low-amplitude fast rhythms) accounts for less than 20% of the total activation time. The opposite occurs in the third subtype (A3), where 50% or more of the total activation time is related to desynchronized patterns. The second subtype (A2) characteristics are between the other two subtypes [3]. As a result, there is an energy increase in the higher frequency bands as we progress from A1 to A3 subtypes, while the opposite is true for the EEG signal's amplitude.

The A-phase characteristics pose a unique challenge as their analysis must consider the variation in amplitude (time) and frequency. Therefore, wavelet analysis is a prime candidate for feature creation, as it can combine both. Nevertheless, a time-frequency analysis can be challenging to work with using conventional machine learning techniques as it leads to a two-dimensional representation. This representation can be viewed as an image; thus, image base models, such as a Convolutional Neural Network (CNN), can be used to evaluate how the frequency content of the EEG signal changes over time using the Continuous Wavelet Transform (CWT) as the information source. Yeh and Shi [4] reported that CWT displays characteristic patterns for the A-phase subtypes, showing that, by using a masking phase-amplitude coupling, the coupling intensity lowers from A1 to A3 subtypes. Furthermore, Medina-Ibarra et al. [5] showed that CWT is suitable for detecting the onset and offset boundaries of the A-phase subtypes. Therefore, the hypothesis of this work is that images created by CWT are appropriate to perform A-phase and its subtypes classification.

CAP was identified as a marker of sleep instability, with substantial clinical applications, especially in examining sleep-disordered subjects [6,7]. However, CAP analysis suffers from a considerably low specialist agreement when analyzing the same signals, which ranges from 69% to 78%. This is likely related to the large amount of information produced during the whole night examination, making the manual scoring impractical and highly susceptible to error [8]. As a result, there is a need for algorithms capable of automating the CAP base analysis. Furthermore, identifying the instantaneous changes due to unpredictable brain dynamics poses challenges in analyzing EEG signals for CAP analysis. An alternative to mitigate this problem is the transformation of amplitude-time space EEG signals to their corresponding frequency domain signals, which can provide better insights. Therefore, the proposed solution uses CWT to combine time and frequency information, creating images that were then fed to a CNN for A-phase classification based on transfer learning. Multiple CNN architectures were tested, and the best performing one was further employed for the A-phase subtypes classification, using transfer learning from the model trained for the A-phase classification, developing a classifier for each of the subtypes (one-versus-all analysis). Furthermore, the Grad-CAM++ [9] method was used to provide visual explanations of the CNN model predictions, leading to a more interpretable model that can be further refined and help identify errors that the manual supervision of a physician can correct.

As a result, the three main novelties of this work are: First, the use of CWT images to develop a machine learning model for A/not-A classification, later transferring this model for the A-phase subtypes classifications (A1/not-A1, A2/not-A2, and A3/not-A3), which is the most relevant problem to be addressed in CAP assessment as the other CAP-based metrics are derived from this analysis; and the use of a visual inspector such as Grad-CAM++ to understand what are the relevant parts of the image that the classifier is focusing on, improving the interpretability of the results.

2. Methods

The study utilized scalograms for A-phase detections from EEG signals with monopolar deviations using deep learning. Scalograms are multidimensional plots that consist of the absolute value of CWT coefficients [10]. Transforming signals to scalograms offers two benefits: (1) transforming the space-time to the frequency-time domain, which provides detailed information about the signal frequency spectrum (required for A-phase identification); (2) increased dimensionality enables multiresolution analysis of the signals. Consequently, scalograms have been exploited in various applications such as biometric analysis using scalograms of ECG signals [11], epilepsy detection using scalograms from EEG signals [12], and classification of winking signals using EEG signals [13]. However, increased dimensionality also increases the complexity of the data analysis, making classification difficult for conventional machine learning models. Deep Learning models can efficiently deal with this complexity; however, developing or selecting an appropriate deep learning model is difficult and non-trivial [14]. To answer this question, this study explores the potential of existing state-of-the-art pre-trained deep learning models.

This study included six pre-trained deep learning architectures, covering their major architectural developments, feature extraction techniques (type of convolution operations), and the ability to explore the input's global or local feature space. Furthermore, it is worth mentioning that signal samples corresponding to A-phases indicating unstable sleep are sufficiently less than Not-A-phases. In other words, this binary classification task suffers from class imbalance, which would be alleviated using cost-sensitive learning proposed by Ling and Sheng [15]. Finally, to further explain the visibility of the deep learning models used in this study, Gradcamm++ [9] was utilized to analyze the regions targeted by the best-performing model for accurate classification. This provides an insight into the correctness of the deep learning model for classification with respect to the A-phase annotations provided by the sleep experts. Figure 1 presents the workflow implemented in this study.

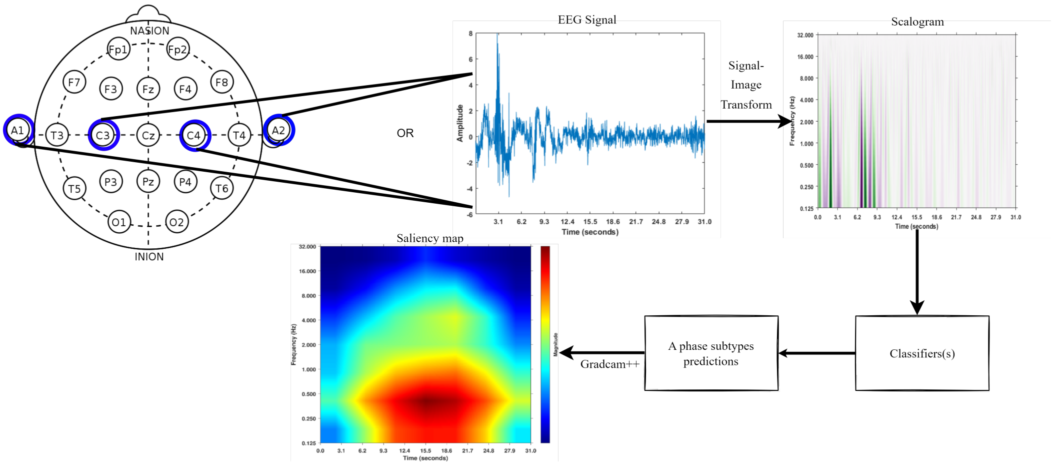


Figure 1. The proposed workflow consists of transforming EEG signal acquired from C4-A1 or C3-A2 to scalogram, feeding to classifiers for A/Not-A-phase classification, and visualizing the region identified by the deep learning network using Gradcam++.

2.1. Dataset

This study employs the Physionet CAP Sleep Database [2], which consists of 19 patients (15 normal and 4 diagnosed with sleep-disordered breathing disorders) and 108 polysomnographic recordings. This study utilized the EEG monopolar deviated signals from C3-A2 or C4-A1. It is worth mentioning that both contain the same information due to their symmetry necessary for CAP scoring along with sufficient requirements for conventional sleep scoring. The study used 592,641 one-second epochs with A-phase annotations by expert neurologists. Table 1 presents the demographic details of the dataset.

Table 1. Dataset summary.

Measure	Mean	Standard Deviation	Range (Min–Max)
Age (years)	40.6	16.8	23.0–78.0
Number of A-phases	439.6	132.4	259.0–844.0
Total A-phase time (s)	4059.2	2194.3	1911.0–10,554.0
Total NREM time (s)	20,505.8	3272.2	13,260.0–27,180.0
Total REM time (s)	5652.6	2505.7	480.0–11,430.0

2.2. Pre-Processing

This step includes filtering followed by resampling the signals to the lowest sampling frequency of the dataset [16]. The filtering step utilized an eighth-order Chebyshev type I filter with a passband ripple of 0.05 dB and cutoff frequency of 0.8/FS (FS is the sampling rate of the signal to be filtered) to avoid aliasing. Subsequently, the filtered sample was resampled to the sampling frequency of 100 Hz, followed by normalizing with zero mean and unit variance to overcome systematic signal variations. Furthermore, despite the recommendation of removing cardiac field and eye movement artifacts [17], these were not removed from the EEG signals for algorithmic simplicity, as it requires including additional signals from other sensors (usually electrooculogram and electrocardiogram).

2.3. Signal to Image Conversion

The unpredictable behavior of brain dynamics induces instantaneous changes that can be recognized in different frequency bands [12]. Hence, frequency domain analysis of EEG signals can provide better information about brain activity. Additionally, due to the non-removal of artifacts, the EEG signals may contain noise, which, when preprocessed, causes information loss [18]. Considering the above-mentioned problems, this study utilizes Continuous Wavelet Transforms (CWT) to transform monopolar deviated EEG signals to the frequency-time domain, followed by creating scalograms for each 31-second EEG signal sample. CWT offers better resolution than other transforms due to its adaptive windowing operations using a mother wavelet during wavelet transform [19]. It is mathematically defined as:

$$W(b,a) = \frac{1}{\sqrt{a}} \int \bar{r}\left(\frac{(t-b)}{a}\right) S(t)dt \tag{1}$$

where $W(b,a), a, b, \bar{r}$, and $S(t)$ are wavelet coefficients, scale, position parameter, the conjugate of a wavelet function, time, and signal in the space-time domain, respectively. This study uses the Ricker wavelet, also known as *Marr* or *Mexican hat wavelet*, as a mother wavelet function [20], defined as:

$$Ricker(t) = \frac{2}{\pi^{\frac{1}{4}}\sqrt{3a}} \left(1 - \left(\frac{x}{a}\right)^2\right) \exp -\frac{1}{2}\left(\frac{x}{a}\right)^2 \tag{2}$$

where a is the width of the wavelet, x is the length of the Mexican Hat wavelet containing zero centered values, and π is constant. Since the Mexican Hat wavelet can be approximated by the Difference of Gaussian's (DoG) for multi-dimensions (e.g., images), and DoG can be separable into multiple components (x and y components for each color channel of RGB image), the computation time for coefficient calculations would be relatively lesser than other wavelets. Additionally, computational biologists have extensively used Gaussian filters due to the Gaussian nature of the responses exhibited by the signaling pathways of the brain [21]. This encourages using the Ricker wavelet as a mother wavelet function for CWT in this study.

Subsequently, scalogram images were created for provided signal samples. A scalogram consists of absolute values of CWT of the provided signal, thereby providing a discrete form of the continuous signal [10]. Being a time-frequency representation, it can give information about the periodic components in the signal [11]. This solves, the space-time representation of a signal, which is limited due to its inability to represent frequency properly. Therefore, transforming space-time representation to frequency provides insights into its frequency and phases, which is very helpful in precisely analyzing the complex and non-stationary signals [22].

2.4. Deep Learning Models

Six deep learning architectures were employed for A and Not-A-phase classification. As mentioned earlier, the selection of these deep learning models depends on the architecture, feature calculation techniques (different types of convolutions), and exploration of feature space (global and local). We focus on using the following deep learning models, namely VGG19 [23], ResNet50 [17,24], InceptionNetV3 [25], Densenet121 [26], MobilenetV2 [27], and EfficientNet-B0 [28] for the proposed binary classification problem. Table 2 briefly describes each deep learning model used in this study.

Table 2. Brief description of the state-of-the-art deep learning models.

Architecture	Number of Layers					Activations	Parameters
	Convolution	Pooling	Batch Normalization	Fully Connected	Dropout		
VGG19	16	5	0	3	2	ReLU	143,667,240
ResNet50	158	2	53	1	0	ReLU	25,557,032
InceptionNetV3	197	6	96	2	1	-	27,161,264
DenseNet121	120	8	121	1	0	ReLU	7,978,856
MobileNetV2	52	0	52	1	1	ReLU6	3,504,872
EfficientNetB0	81	34	49	1	1	SiLU	5,288,548

2.4.1. Training Parameters

Normalized scalogram images with zero mean and unit standard deviation were resized to 64 × 64 × 3 images for feeding the pre-trained models, except for InceptionNetV3. The latter (PyTorch implementation) was not generalized to use custom-sized input images; hence an image of size 299 × 299 × 3 was used for model training. It is worth mentioning that other than image resizing, no further data augmentation techniques were used to avoid data loss from the scalogram images. The last fully connected classification for all deep learning models was modified to solve the proposed binary classification problem. The models were trained on a batch of 32 images using the stochastic gradient descent method. A weight decay of 0.0001, a momentum value of 0.9, and a learning rate of 0.001 were used as hyperparameters to train the models for 50 epochs. An early stopping criterion based on validation loss was also used to terminate the training process.

2.4.2. VGG19

VGG19 is a plain CNN that consists of a stack of convolution layers followed by a Rectified Linear Unit (ReLU) [29], with kernel size 3 × 3 with a stride of 1 pixel without squeezing the size of the output feature maps (same padding). The architecture also uses max-pooling and adaptive average pooling layers of kernel size of 2 × 2 (stride of 2), and 7 × 7, for spatial pooling and flattening, respectively. Finally, three fully connected layers of sizes 4096, 4096, and 1000 were added for mapping and classification using the softmax function. Each fully connected layer is accompanied by ReLU activation and a dropout layer with a 50% dropout rate to prevent data overfitting.

2.4.3. ResNet50

Inspired by VGG networks, the ResNets, also known as residual networks, consist of convolution layers of kernel size 3 × 3 for convolutions with the following design conditions: the equal number of kernels for same-sized feature maps and; a doubled number of filters for half-sized feature map. The architecture also performs downsampling using convolution layers with a stride of 2. Finally, global average pooling and fully connected layers of size 2048 with softmax were included for feature aggregation and classification, respectively. Following the principle of residual learning, it also consists of the skip connections between ResNet blocks (batch norm (1 × 1 convolution), batch norm (3 × 3 convolution), batch norm (1 × 1 convolution), ReLU) for learning the identity functions that connect the two layers with same sized feature map, or zero-padded feature map (in case of dimension mismatch). The advantage of skip connections is two-fold: (1) no extra trainable parameter is required, and (2) fewer filters and a lower complexity due to identity mappings.

2.4.4. InceptionNetV3

The basic building block of the InceptionNetV3 deep learning architecture is its inception module, which reduces the computational complexity and the number of trainable parameters compared to plain deep neural networks such as VGG. An inception module is an asymmetrical building block consisting of three differently sized convolutional kernels: 1×1, $n \times 1$, and $1 \times n$ for $n = 3$, and 5, respectively. Each 3×3 or 5×5 kernel is followed by a $1 \times n$-sized kernel for dimensionality reductions. It also consists of occasionally used max-pooling operations to reduce the grid space resolution. These inception modules are stacked on each other [30], resulting in InceptionNetV3 architecture. It is essential to mention that the PyTorch implementation of this architecture is not generalized well to use custom-sized input; therefore, an image sizerd 299 × 299 × 3 was used as input for training it, unlike the other deep learning models used in this study.

2.4.5. Densenet121

Similar to ResNets, a DenseNet also consists of shortcut connections between dense layers, which are the main constituents of the dense blocks for all versions of DenseNet. Specifically, the basic building block of DenseNet is the stacked dense blocks connected by transition layers. A dense block consists of several dense layers consisting of shortcut connections for dense layers concatenations. A dense layer consists of the following sequence: batch normalization, ReLU activation, convolution (1×1), batch normalization, ReLU activation, and convolution (3×3). A transition layer consists of a convolution layer succeeded by a pooling layer for feature size alteration [26]. The advantage of this type of architecture includes strengthening the network's feature sharing and propagation capability, reduced trainable parameters, and comparatively fewer chances of the vanishing gradient problem.

2.4.6. MobileNetV2

MobilenetV2 is a lightweight deep neural network exclusively designed for mobile embedded vision applications [31]. The architecture consists of a convolution layer with 32 kernels of size 3 × 3 and 19 residual bottleneck layers. Each bottleneck layer consists of three convolution layers with ReLU6 activation, except for the third layer, with expansion factor and shortcut connections connecting them. The first and third layer performs pointwise convolutions (1×1), while the second layer performs depth-wise convolutions of kernel size 3 × 3. Finally, a fully connected layer of size 1280 with softmax function was included for classification. The bottleneck layers in the architecture make it memory efficient and less vulnerable to information loss due to non-linearities.

2.4.7. EfficientNetB0

The main characteristic of EfficientNet is the gradual scaling of the architecture in terms of depth, width, and resolution. The basic building block of EfficientNet is MBConv blocks [32] with additional squeeze-and-excitation optimization (adaptive average pooling-convolution $(1 \times 1) \times 2$, sigmoid SiLU, sigmoid). The squeeze step squeezes the global feature maps, whereas the excitation step captures the channel-specific dependencies, which leads to an exploration of the global feature space. Besides, EfficientNets use Sigmoid Linear Unit (SiLU) activation [33], unlike other deep learning architectures. An MBConv block consists of pointwise convolution followed by depth-wise convolution of kernel size 3 × 3 and pointwise convolutions. In contrast, the squeeze-and-excitation optimization layer constitutes the following sequence: adaptive average pooling, 2 pointwise convolutions, SiLU activation, and scale activation using a sigmoid function. All EfficientNet architectures consist of seven blocks constituting a varied number of sub-blocks. An EfficientNet-B0 is composed of SiLU-activated convolutional layers (which are batch-normalized) followed by one modified MBconv1 (convolutions, squeeze-and-excitation, convolution), six MBConv6 (convolutions, convolution–squeeze-and-excitation, convolution) blocks, and subsequently convolution, pooling, and fully connected layers with softmax function.

2.5. Understanding the Classifier Decisions

Deep learning models have been extensively used in the computer vision domain but are perceived as a black box due to a lack of understanding of their internal functioning. Consequently, it is challenging to visualize the model predictions. For this purpose, this study used Gradcam++ to visualize the regions selected by the best-performing classifier. Gradcam++ provides a heatmap showing the magnitude scale depicting the human interpretable visualizations for the class labels [9]. This enables investigation of the classifier's ability for making correct predictions by providing their corresponding targeted regions.

3. Results

This study aimed to predict the presence of the A-phase from the monopolar deviated EEG signals extracted from C3-A2 or C4-A1 by exploiting their frequency domain, spatially and temporally. For this purpose, the preprocessed EEG signals with a processing window of 31 s were transformed into scalogram post CWT using the python SciPy library. Subsequently, six pre-trained state-of-the-art deep learning models were analyzed to solve this binary classification algorithm, possessing a class imbalance (relatively lesser samples depicting the presence of A-phase). Finally, Gradcam++ was used to visualize the regions selected by the best-performing deep learning classifier for classification. All experiments were conducted using PyTorch version 1.11.0 library.

3.1. Signal to Scalogram Transformation

The preprocessed signals, after sampling and filtering, were transformed into scalogram images. For each signal, a processing window of 31 s with an overlap of one second underwent continuous wavelet transform using the Ricker mother wavelet function. CWT was performed using the signal submodule of Python's Scipy library. Consequently, 448,883 signal samples were transformed into the corresponding scalogram images.

3.2. Parameters and Data Splitting

A classification task deals with splitting the data into training, validation, and test dataset, among which the training dataset is used for model training. In contrast, validation and test datasets are used for analyzing the performance of the trained model. Following the same protocol, the dataset was divided into training, validation, and test datasets, as presented in Table 3.

Table 3. Training, validation, and testing dataset distribution for A, A subtypes, and Not-A-phase.

Classes	Training	Validation	Testing
Not-A	247,448	75,222	77,532
A-phase	27,959	9178	11,949
A1 phase	13,353	4589	4733
A2 phase	5107	1872	3418
A3 phase	9499	2717	3343

It is clear from Table 3 that this classification problem suffered from the class imbalance problem. To alleviate the problem, cost-sensitive learning was employed, assigning the relatively higher cost to Type I (False Negatives) errors than Type II (False positives).

The cost for penalizing the False Positives depends on the number of samples belonging to the corresponding classes in the dataset. Precisely, the cost of penalizing the incorrect classification for the Not-A-phase is 0.5565 (False Positive). In contrast, incorrect classification of the A-phase would incur the cost of 4.9252 (False Negative). The loss function used for performance analysis is cross-entropy loss, or maximum likelihood using predefined costs for penalizing incorrectly classified samples for both classes. The loss will be optimized using a stochastic gradient descent with a learning rate of 0.001 with a weight decay of 0.0001 and momentum of 0.9, for fair comparison between classifiers.

A total of 2 stopping criteria, 50 epochs, and early stopping were used to guide model training. Validation loss was defined as an early stopping criterion with a patience value of 20 epochs. Concretely, an increase in the validation loss is monitored for up to 20 consecutive epochs, and training is stopped if the validation loss continues to increase post 20 consecutive epochs. It is important to mention that a fine-tuned model for A/not-A-phase classification was used as a starting point for training the A-phase subtypes classification model. Therefore, similar settings were maintained for model training (except for class weights and number of classes). The distribution for training, validation, and test datasets for A-phase subtypes is presented in Table 3.

3.3. *Performance Analysis*

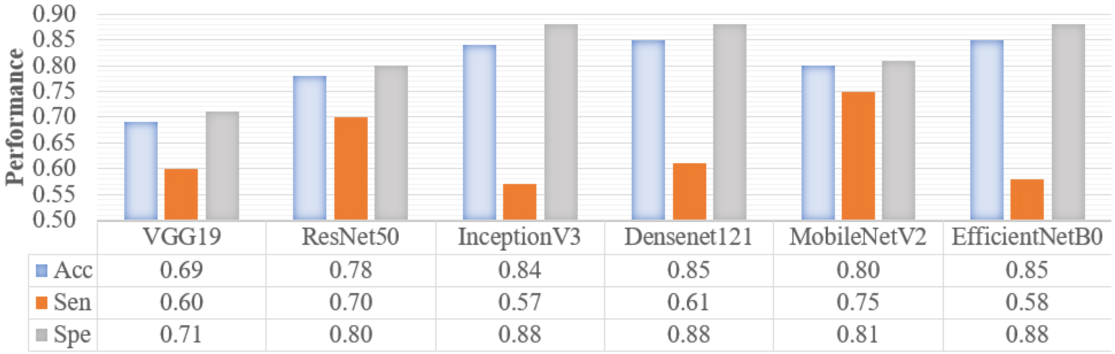
This subsection is divided into two parts: binary classification for A and Not-A-phase classification; A-phase subtypes classification. Specifically, various deep-learning architectures were used to solve the binary classification problem (A/Not-A-phase prediction), subsequently, the best-performing classifier was further used for A-phase subtypes (A1, A2, and A3) classification.

3.3.1. A/Not-A-Phase Prediction

As mentioned earlier, six pre-trained deep learning classifiers were used; namely, VGG19, ResNet50, InceptionNetV3, DenseNet 121, MobileNet-V2, and EfficientNet-B0, to classify the A and Not-A-phases from the EEG signals. The performance of each classifier was tested using three metrics: accuracy, sensitivity, and specificity on the test dataset. Accuracy gives insight into correctly classified samples, where sensitivity and specificity define the classifier's ability to correctly predict the presence (True Positives) and absence (True Negatives) of A-phases in the given EEG signal, respectively.

Since the dataset possesses a class imbalance, accuracy alone is not a good estimator for the performance analysis of the classifier. The reason is that a higher number of true negatives will contribute more to the accuracy than true positives. Hence, alternative performance metrics would be required to quantify the performance of classifiers for correctly classifying the samples from both classes (A and Not-A). Two such performance metrics are sensitivity and specificity. Sensitivity defines the percentage of correctly classified A-phases, whereas specificity depicts the percentage of correctly classified Not-A-phases. Figure 2 illustrates the accuracy, sensitivity, and specificity of various classifiers used in this study.

VGG19, which is a plain deep network, achieved lower accuracy than other sophisticated networks used in this study. Since VGG19 has relatively higher parameters to train (Table 2), VGG19 might have suffered from the overfitting problem. In contrast, all other architectures possessed shortcut connections for learning identity mapping, which avoids the risk of overfitting due to relatively fewer trainable parameters. On the other hand, increasing the depth of the network and skip connections significantly improved the accuracy. However, a very slight change in accuracy was reported by InceptionNetV3, DenseNet 121, and EfficientNet-B0, respectively. The best accuracy was achieved by EfficientNet-B0.



	VGG19	ResNet50	InceptionV3	Densenet121	MobileNetV2	EfficientNetB0
■ Acc	0.69	0.78	0.84	0.85	0.80	0.85
■ Sen	0.60	0.70	0.57	0.61	0.75	0.58
■ Spe	0.71	0.80	0.88	0.88	0.81	0.88

Figure 2. Performance analysis of different deep learning architectures for A and Not-A-phase classification.

Regarding sensitivity, EfficientNet-B0 and InceptionNetV3 performed the worst, whereas VGG19 and DenseNet 121 achieved similar sensitivity values. Likewise, the performance difference in sensitivity between ResNet50 and MobileNet-V2 is insignificant, though the latter achieved a better and highest sensitivity value. In other words, MobileNet-V2 could correctly classify A-phases from the scalogram images better than any other deep learning model. The worst performance by EfficientNet-B0 was potentially due to its squeeze-excitation layer, which aims to add a few parameters for adaptive modification of the network. The squeeze step applies global average pooling, whereas excitation consists of fully connected layers for producing per-channel activations to capture channel-specific dependencies [21]. Since EfficientNet-B0 is inclined more towards exploring the global feature space, thereby decreasing the contribution of local feature space, it suffered from targeting the local region consisting of the A-phase. On the other hand, for consistency, the auxiliary classifier of the InceptionNetV3 was not fine-tuned, resulting in poor network convergence and a vanishing gradient problem. DenseNet 121 could not perform better due to its ability to discard redundant features, which also discarded a subtle variation for distinguishing A-phase. In other words, due to symmetricity, the network might discard features that are redundant but with subtle variations, thereby resulting in the inability to detect these variations due to discarding [19]. On the other hand, VGG19 suffered from a vanishing gradient problem, which resulted in poor discrimination between A and Not-A-phase samples. On the other hand, MobileNet-V2 outperformed all the classifiers in terms of sensitivity by achieving the highest sensitivity value of 0.75.

Almost all models, except for VGG19, achieved a similar specificity (the difference is insignificant). Additionally, DenseNet 121, EfficientNet-B0, and InceptionNetV3 achieved similar specificity values. Higher specificity values as compared to sensitivity were expected due to the lower number of A-phase samples. Surprisingly, MobileNet-V2 achieved a little bit lower sensitivity value. However, it maintained a good balance between accuracy, sensitivity, and accuracy.

Overall, MobileNet-V2 (Figure 3) performs the best due to its ability to extract subtle details, the same as ResNet50. The difference in performance is due to the difference in building blocks, i.e., inverted residual blocks in MobileNet-V2 and residual blocks in ResNet50. Inverted residual blocks are connected through bottleneck layers (with a lower number of channels), which aims at aggregating features through pointwise convolutions. In contrast, residual blocks are connected through layers with a higher number of channels. Efficient feature aggregation in bottleneck layers in MobileNet-V2 and aggregating redundant information due to the higher number of channels in the ResNet50 make the MobileNet-V2 perform better than ResNet50.

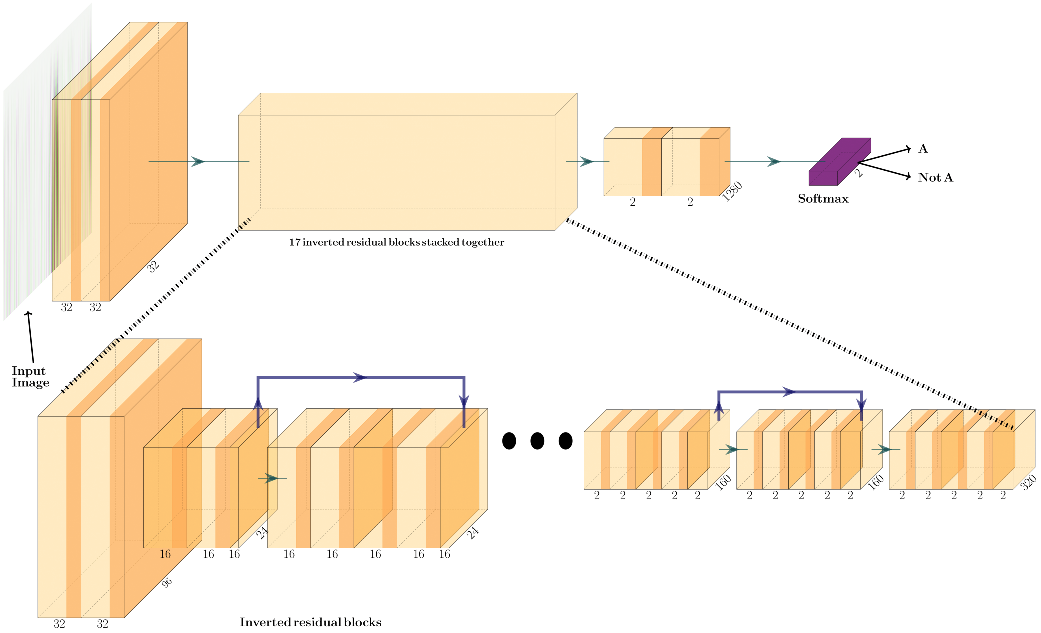


Figure 3. The architecture of MobileNetV2 consisting of inverted residual blocks (stack of yellow-saffron blocks) connected using skip connections and softmax for classification.

3.3.2. A-Subtypes Classification

MobileNetV2 trained for binary classification task (A/Not-A-phase), was used as starting point to classify A-phase subtypes. The model was trained by applying cost-sensitive learning, followed by threshold tuning. The latter was also employed using Receiver Operating Curve (ROC) [34], and geometric mean [35], for better performance. The performance of the aforementioned deep learning model for A-phase subtypes classification is presented in Figure 4. It is worth mentioning that, despite the relatively lesser number of A2 samples compared to A1 and A3 samples in the test dataset, the classification performance of A2 phases was relatively better than A1, and A3. Therefore, it can be concluded that MobileNetV2 performed better for both A and Not-A-phases and subtypes classification, respectively. This is due to its inverted residual blocks, which exhibit efficient feature aggregation and better identity mapping and learning. This study did not test other deep learning classifiers used for binary classification, since it is expected to perform inferior to the best-performing model (MobileNetV2) for binary classification.

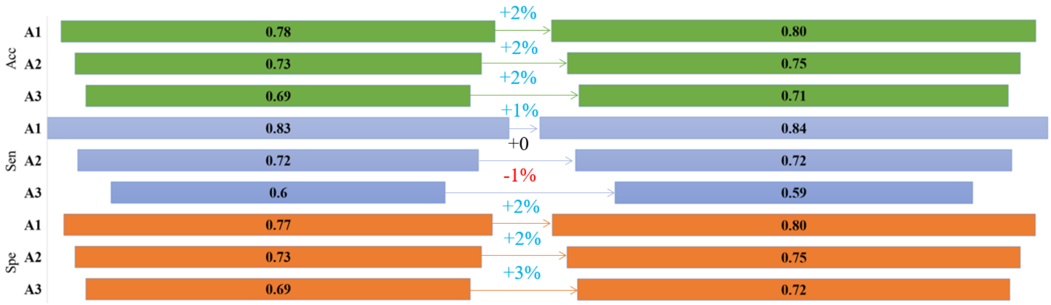


Figure 4. Performance analysis of A−phase subtypes classification before (**left**) and after post processing (**right**).

3.4. *Visualizing the Regions Targeted Using Gradcam++*

3.4.1. Binary Classification A/Not-A-Phase Classification

Gradcam++ was used to identify the regions from the scalogram images, which were used by the best-performing deep learning model. This includes visualizing the regions targeted by MobileNetV2 from the correct and incorrect classifications. Figure 5 presents the A/not-A predictions of a continuous EEG signal, depicting the CAP cycle. Analyzing the correctly classified samples gives insight into the regions identified by the MobileNetV2. In contrast, analysis of incorrectly classified samples would help exclude the regions causing the classifier to learn wrongly and understand the potential reasons behind it. For this purpose, the last feature extraction layer of the MobileNetV2 was selected, followed by feeding an input image to it, then Gradcam++ was used to produce a saliency map and its superposition with the original image, as shown in Figure 6.

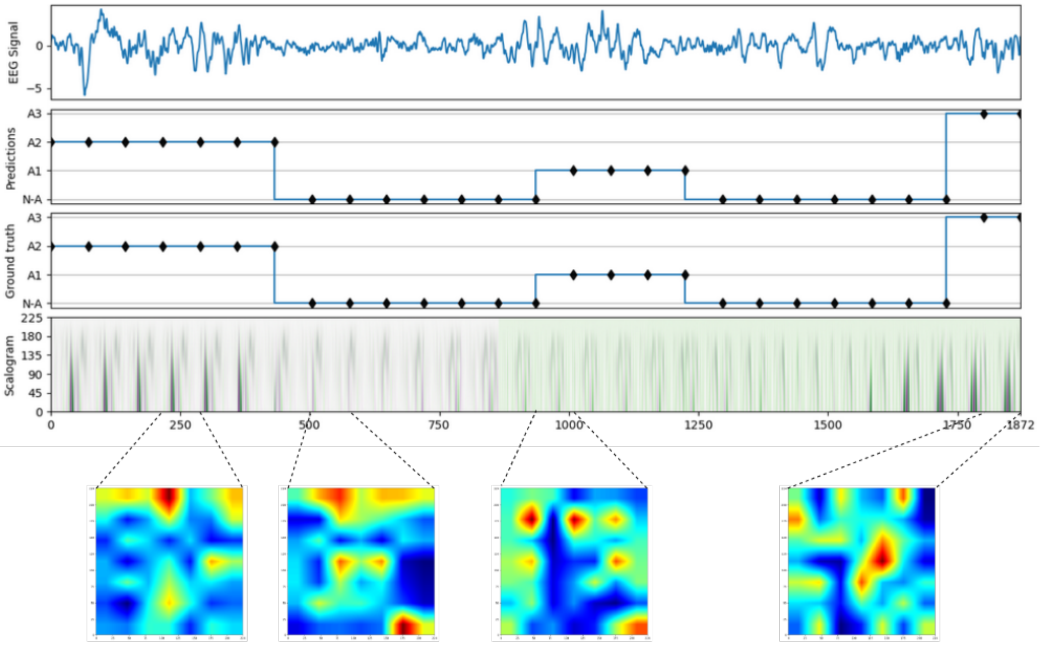
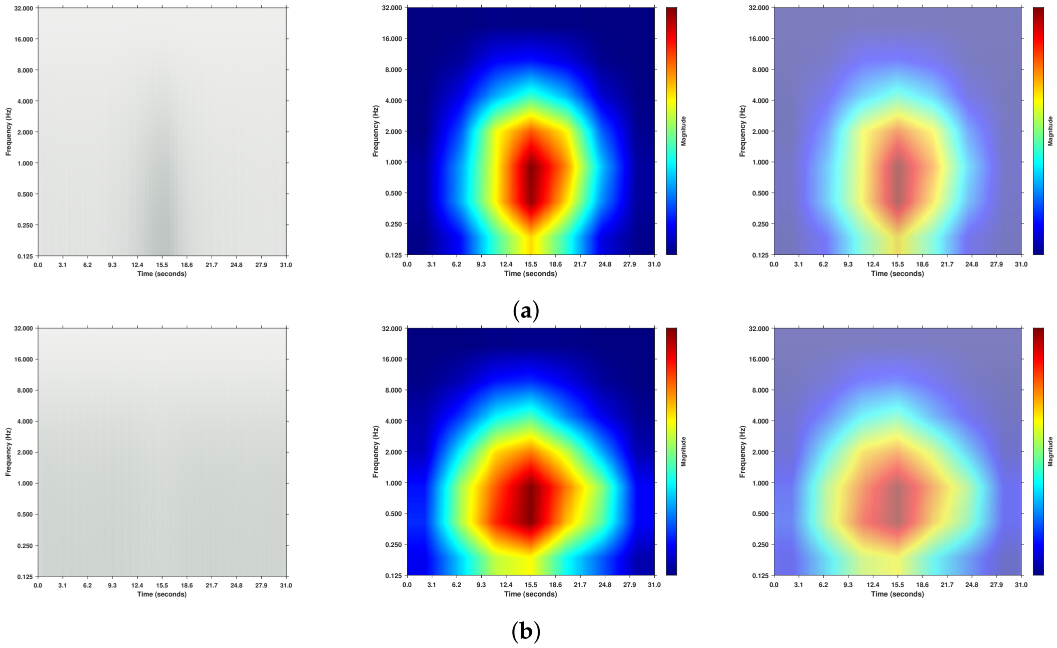


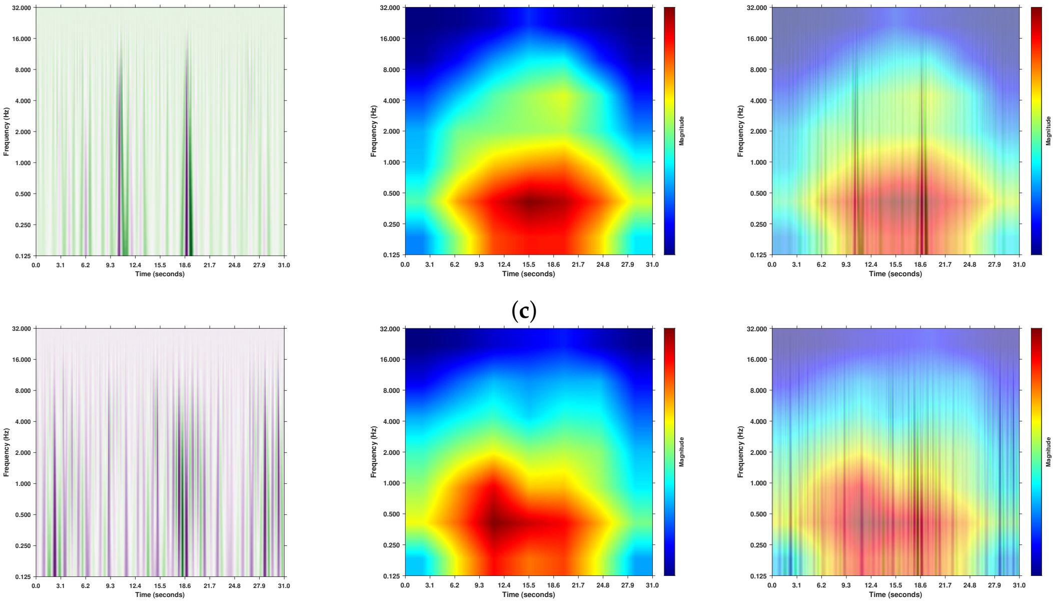
Figure 5. A figure depicting random scalogram image samples with their ground truth, model predictions, and gradcam outputs for A-phase subtypes. The red color in the gradcam image depicts the highest number of active neurons in the region, and blue region corresponds to region with inactive neurons of the neural network.



(**a**)

(**b**)

Figure 6. *Cont.*



(**c**)

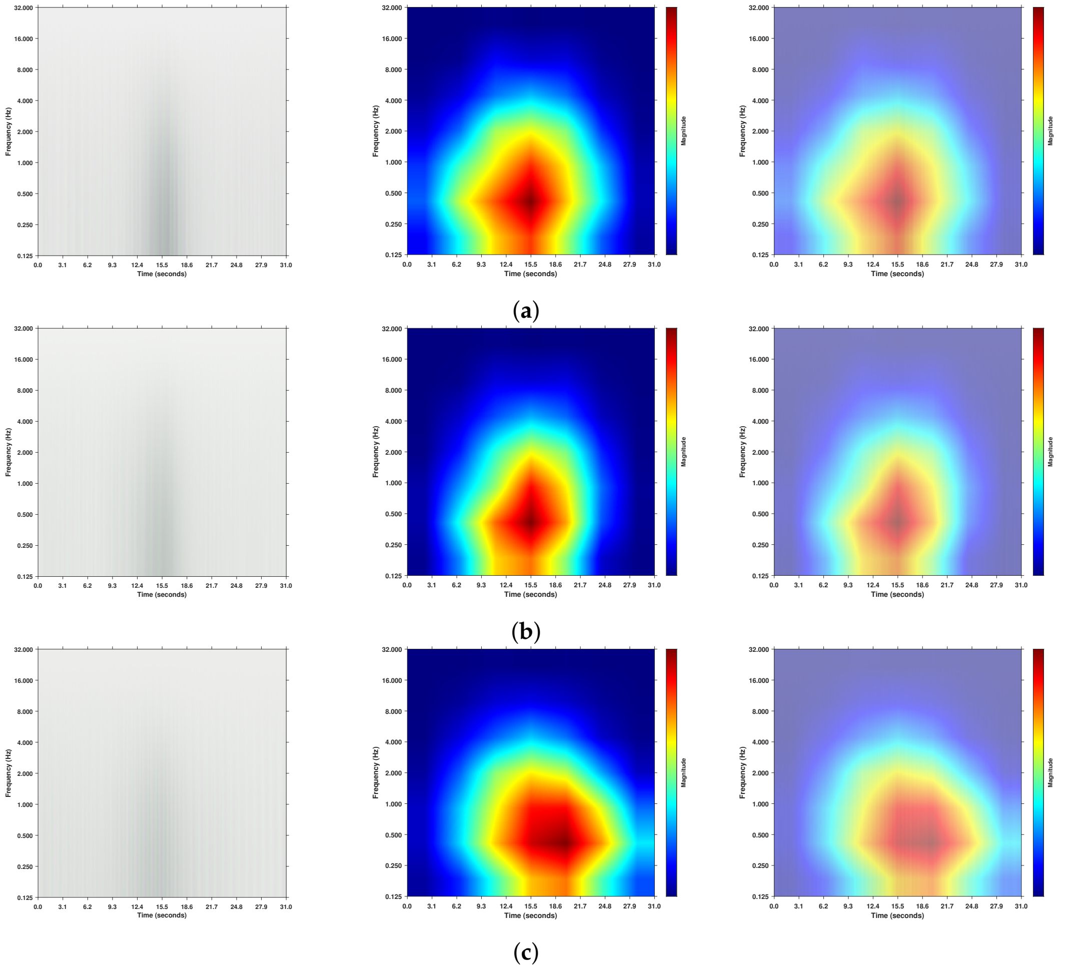
(**d**)

Figure 6. Gradcam++ visualization of trained model representing a sample image (**left**), saliency map (**middle**), and their superimposition (**right**). (**a**) True Positives (Correctly Classified A-phases); (**b**) Negatives (Correctly Classified Not-A-phases); (**c**) False Positive (Incorrectly classified Not-A-phases); (**d**) False Negative (wrongly classified A-phases).

To visualize the regions corresponding to correct A and Not-A-phase classification, 1000 randomly selected True Positive (TP) and True Negative (TN) samples were selected to create an average image. On the other hand, to identify the incorrect regions targeted by MobileNet-V2, random false negatives for A and its subtypes, along with an False Positive (FP) sample, were selected. As mentioned earlier, since the signal samples are of length 31 s with the labeled sample in the middle (16th second), MobileNet-V2 looked for the patterns in the middle section of the input scalogram, as depicted in Figure 6, resulting in correct predictions. At the same time, correct classifications of the Not-A-phases were also due to targeting almost the same region. On the other hand, for FPs and False Negatives (FNs), the network looked at the whole image due to the presence of similar patterns such as the A-phase located at the middle-bottom of the scalogram, which led to false predictions. This is due to the coinciding frequencies of artifacts with A-phase, present in the EEG signal, which were not removed in this study.

3.4.2. A-Phase Subtype Classification

Gradcam++ was also used to identify the regions targeted by A-phase subtypes. Similar to binary classification, 1000 random TP samples, its subtypes A1, A2, and A3 samples were selected, subsequently following the same procedure (average image creation and heatmap generation) was followed. The visualization results using Gradcam++ for TPs for A, A1, A2, and A3 are shown in Figure 7. Similar to the region targeted by A-phase, MobileNet-v2 targeted the same region, thereby correctly classifying A-phase subtypes with minor differences, depending on their frequency.



(**a**)

(**b**)

(**c**)

Figure 7. Gradcam++ visualization of the trained model representing an averaged image over 1000 samples for each subtype (**left**), saliency map (**middle**), and their superimposition (**right**). %MDPI: We moved the subfigure explanations into the figure caption. Please confirm. (**a**) A1 Phase; (**b**) A2 Phase; (**c**) A3 Phase.

4. Discussion

The baselines for the performance analysis of the best-performing models are considered from the review studies conducted by Mendonca et al. [36], and Sharma et al. [37]. Table 4 presents the performance metrics provided by the baselines from the reviews. It is important to mention that this work did not focus on improving the performance of models, but rather on the model's explainability for A-phase and its subtypes prediction. Specifically, there is a trade-off between accuracy and explainability, as explained in the study conducted by Van der Veer et al. [38]. Despite this trade-off, the proposed model surpassed the accuracy and sensitivity values of the baselines i.e., Average (Avg) performance) identified in the reviews. Additionally, most works in the literature have manually removed REM sleep periods since CAP is only defined in NREM, which resulted in an accuracy increase [39,40], however, doing so leads to a model that is not suitable for fully automatic analysis. In contrast, this work did not perform this preprocessing to ensure the automaticity and simplicity of the method.

Meanwhile, the dataset consists of significantly more B phase samples than A-phase samples (class imbalance), which may lead to higher contributions of specificity to the global accuracy than sensitivity.

Table 4. Performance analysis of the proposed MobileNetV2 model for A-phase predictions with baselines.

Authors	Accuracy	Sensitivity	Specificity
Mendonca et al. [36]	0.76 (Avg)	0.72 (Avg)	0.78 (Avg)
Sharma et al. [37]	0.79 (Avg)	0.73 (Avg)	**0.88 (Avg)**
Proposed study	**0.80**	**0.75**	0.81

Furthermore, the best-performing model, MobileNetV2, was further tested to predict the subtypes of A-phases for which the pre-trained model for A-phase was used as a starting point. To avoid misclassifications of A-phase subtypes, it is vital to measure the CAP events boundary conditions suggested by Largo et al. [41]. Therefore, a correction procedure was applied, which increased the accuracy, specificity, and sensitivity value by 2%. However, the sensitivity of the A2 phase did not improve, and it was reduced by 1% in A3. This is likely related to the tendency of the model to misclassify the A3 phase offset, a problem previously reported to be persistent in this subtype, especially due to uncertainty of where the optimal cutting point to define the subtype ending is [36]. Therefore, by using the correction procedure, the isolated classifications on the offset are removed, leading to a lower sensitivity. The A2 phases were likely unaffected because they comprise a mix of patterns between the other two subtypes. Hence it is likely that the correction procedure is improving the performance in some cases while reducing it in others. However, using this procedure leads to a better performance because it boosts the A1 phase detection, which is the most common, leading to a better A-phase detection performance [42].

To further explain the visibility and get insights into the models' correctness in predicting the samples, Gradcam++ [9] was utilized. It was found that the model focused on similar locations as human sleep experts. In addition, Largo et al. [41] verified that the global average of the inter-scorer agreement was 69.9%. Both observations proved the clinical viability and robustness of the proposed model.

5. Conclusions

This study explores the potential of various deep learning architectures based on their global and local feature space exploration abilities for A-phase and its subtypes prediction. The EEG signals with monopolar deviations are transformed to the scalograms and fed to six classifiers, namely, VGG19, ResNet50, InceptionNetV3, Densnet121, MobileNetV2, and EfficientNetB0.

The best-performing model for A-phase prediction was MobileNetV2, which was further tuned to predict A-subtypes. Precisely, MobileNetV2 achieved average accuracy, specificity, and sensitivity values of 0.85, 0.75, and 0.81, while these values range between 0.71 and 0.80, 0.59 and 0.84, and 0.72 and 0.80, respectively.

Furthermore, to understand the training model's visibility, Gradcam++ was used. It was found that the model's visibility matches the locations identified by human sleep experts for A-phase and its subtypes. Considering this observation and the global average of inter-scorer agreement, this model is clinically viable and robust for prediction of A-phase and its subtypes.

Author Contributions: A.G. contributed to the conceptualization, methodology, and implementation of various experiments in the study, preparing the first manuscript draft. F.M. and S.S.M. contributed substantially to conceptualization, data analysis, and manuscript preparation. A.G.R.-G. and F.M.-D. helped in providing constructive discussion and manuscript refinement for publication purposes. All authors have read and agreed to the published version of the manuscript.

Funding: This research received funding from LARSyS (Project- UIDB/50009/2020) and ARDITI—Agência Regional para o Desenvolvimento da Investigação, Tecnologia e Inovação under the scope of the project M1420-09-5369-FSE-000002—Post-Doctoral Fellowship, co-financed by the Madeira 14-20 Program—European Social Fund, and Projeto RRSO—Restaurant Review Sentiment Output (M1420-01-0247-FEDER-000055).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: This work includes the usage of EEG signals from Physionet CAP sleep database [2], publicly available under Open Data Commons Attribution License v1.0. The EEG signals were transformed into scalogram images, resulting in a dataset that is freely available at https://www.kaggle.com/datasets/ankitgupta1991/cap-image-database, accessed on 15 May 2023.

Acknowledgments: The authors are thankful to anonymous reviewers for their suggestions and recommendations to improve this manuscript for journal publication.

Conflicts of Interest: The authors declare that they have no known competing financial interest or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Berry, R.B.; Brooks, R.; Gamaldo, C.; Harding, S.M.; Lloyd, R.M.; Quan, S.F.; Troester, M.T.; Vaughn, B.V. AASM scoring manual updates for 2017 (version 2.4). *J. Clin. Sleep Med.* **2017**, *13*, 665–666. [CrossRef] [PubMed]
2. Terzano, M.G.; Parrino, L.; Smerieri, A.; Chervin, R.; Chokroverty, S.; Guilleminault, C.; Hirshkowitz, M.; Mahowald, M.; Moldofsky, H.; Rosa, A.; et al. Atlas, rules, and recording techniques for the scoring of cyclic alternating pattern (CAP) in human sleep. *Sleep Med.* **2002**, *3*, 187–199. [CrossRef] [PubMed]
3. Terzano, M.G.; Parrino, L. Origin and significance of the cyclic alternating pattern (CAP). *Sleep Med. Rev.* **2000**, *4*, 101–123. [CrossRef] [PubMed]
4. Yeh, C.H.; Shi, W. Identifying phase-amplitude coupling in cyclic alternating pattern using masking signals. *Sci. Rep.* **2018**, *8*, 2649. [CrossRef] [PubMed]
5. Medina-Ibarra, D.; Mendez, M.O.; Chouvarda, I.; Murguía, J.; Alba, A.; Arce-Santana, E.R.; Bianchi, A. Assessment of Singularities in the EEG during A-phases of Sleep based on Wavelet Decomposition. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2022**, *30*, 2721–2731. [CrossRef] [PubMed]
6. Parrino, L.; Ferri, R.; Bruni, O.; Terzano, M.G. Cyclic alternating pattern (CAP): The marker of sleep instability. *Sleep Med. Rev.* **2012**, *16*, 27–45. [CrossRef]
7. Terzano, M.G.; Parrino, L. Clinical applications of cyclic alternating pattern. *Physiol. Behav.* **1993**, *54*, 807–813. [CrossRef]
8. Rosa, A.; Alves, G.R.; Brito, M.; Lopes, M.C.; Tufik, S. Visual and automatic cyclic alternating pattern (CAP) scoring. *Arq. Neuro-Psiquiatr.* **2006**, *64*, 578–581. [CrossRef]
9. Chattopadhay, A.; Sarkar, A.; Howlader, P.; Balasubramanian, V.N. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018; pp. 839–847.
10. Arino, M.A.; Morettin, P.A.; Vidakovic, B. Wavelet scalograms and their applications in economic time series. *Braz. J. Probab. Stat.* **2004**, *18*, 37–51.
11. Byeon, Y.H.; Pan, S.B.; Kwak, K.C. Intelligent deep models based on scalograms of electrocardiogram signals for biometrics. *Sensors* **2019**, *19*, 935. [CrossRef]
12. Türk, Ö.; Özerdem, M.S. Epilepsy detection by using scalogram based convolutional neural network from EEG signals. *Brain Sci.* **2019**, *9*, 115. [CrossRef]
13. Kumar, J.L.M.; Rashid, M.; Musa, R.M.; Razman, M.A.M.; Sulaiman, N.; Jailani, R.; Majeed, A.P.A. The classification of EEG-based winking signals: A transfer learning and random forest pipeline. *PeerJ* **2021**, *9*, e11182. [CrossRef]
14. Voulodimos, A.; Doulamis, N.; Doulamis, A.; Protopapadakis, E. Deep learning for computer vision: A brief review. *Comput. Intell. Neurosci.* **2018**, *2018*, 7068349. [CrossRef]
15. Ling, C.X.; Sheng, V.S. Cost-sensitive learning and the class imbalance problem. *Encycl. Mach. Learn.* **2008**, *2011*, 231–235.
16. Weinstein, C.J. Programs for digital signal processing. *Proc. IEEE* **1979**, *69*, 856–857.
17. Hartmann, S.; Baumert, M. Automatic a-phase detection of cyclic alternating patterns in sleep using dynamic temporal information. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2019**, *27*, 1695–1703. [CrossRef]
18. Hatipoglu, B.; Yilmaz, C.M.; Kose, C. A signal-to-image transformation approach for eeg and meg signal classification. *Signal Image Video Process.* **2019**, *13*, 483–490. [CrossRef]
19. Sadowsky, J. Investigation of signal characteristics using the continuous wavelet transform. *Johns Hopkins Apl Tech. Dig.* **1996**, *17*, 258–269.
20. Wang, Y. Frequencies of the Ricker wavelet. *Geophysics* **2015**, *80*, A31–A37. [CrossRef]
21. Bahg, G.; Evans, D.G.; Galdo, M.; Turner, B.M. Gaussian process linking functions for mind, brain, and behavior. *Natl. Acad. Sci.* **2020**, *117*, 29398–29406. [CrossRef]
22. Arefnezhad, S.; Eichberger, A.; Frühwirth, M.; Kaufmann, C.; Moser, M.; Koglbauer, I.V. Driver monitoring of automated vehicles by classification of driver drowsiness using a deep convolutional neural network trained by scalograms of ECG signals. *Energies* **2022**, *15*, 480. [CrossRef]
23. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.

24. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
25. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826.
26. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
27. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Munich, Germany, 8–14 September 2018; pp. 4510–4520.
28. Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In Proceedings of the International Conference on Machine Learning, PMLR, Long Beach, CA, USA, 9–15 June 2019; Volume 97, pp. 6105–6114.
29. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
30. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
31. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
32. Tan, M.; Chen, B.; Pang, R.; Vasudevan, V.; Sandler, M.; Howard, A.; Le, Q.V. Mnasnet: Platform-aware neural architecture search for mobile. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 2820–2828.
33. Elfwing, S.; Uchibe, E.; Doya, K. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Netw.* **2018**, *107*, 3–11. [CrossRef]
34. Brown, C.D.; Davis, H.T. Receiver operating characteristics curves and related decision measures: A tutorial. *Chemom. Intell. Lab. Syst.* **2006**, *80*, 24–38. [CrossRef]
35. Batuwita, R.; Palade, V. Adjusted geometric-mean: A novel performance measure for imbalanced bioinformatics datasets learning. *J. Bioinform. Comput. Biol.* **2012**, *10*, 1250003. [CrossRef]
36. Mendonça, F.; Fred, A.; Mostafa, S.S.; Morgado-Dias, F.; Ravelo-García, A.G. Automatic detection of cyclic alternating pattern. *Neural Comput. Appl.* **2022**, 34, 11097–11107. [CrossRef]
37. Sharma, M.; Lodhi, H.; Yadav, R.; Elphick, H.; Acharya, U.R. Computerized detection of cyclic alternating patterns of sleep: A new paradigm, future scope and challenges. *Comput. Methods Programs Biomed.* **2023**, *235*, 107471. [CrossRef]
38. van der Veer, S.N.; Riste, L.; Cheraghi-Sohi, S.; Phipps, D.L.; Tully, M.P.; Bozentko, K.; Atwood, S.; Hubbard, A.; Wiper, C.; Oswald, M.; et al. Trading off accuracy and explainability in AI decision-making: Findings from 2 citizens' juries. *J. Am. Med. Inform. Assoc.* **2021**, *28*, 2128–2138. [CrossRef]
39. Mariani, S.; Grassi, A.; Mendez, M.O.; Milioli, G.; Parrino, L.; Terzano, M.G.; Bianchi, A.M. EEG segmentation for improving automatic CAP detection. *Clin. Neurophysiol.* **2013**, *124*, 1815–1823. [CrossRef]
40. Gnoni, V.; Drakatos, P.; Higgins, S.; Duncan, I.; Wasserman, D.; Kabiljo, R.; Mutti, C.; Halasz, P.; Goadsby, P.J.; Leschziner, G.D.; et al. Cyclic alternating pattern in obstructive sleep apnea: A preliminary study. *J. Sleep Res.* **2021**, *30*, e13350. [CrossRef]
41. Largo, R.; Lopes, M.; Spruyt, K.; Guilleminault, C.; Wang, Y.; Rosa, A. Visual and automatic classification of the cyclic alternating pattern in electroencephalography during sleep. *Braz. J. Med. Biol. Res.* **2019**, *52*. [CrossRef]
42. Mendonça, F.; Mostafa, S.S.; Gupta, A.; Arnardottir, E.S.; Leppänen, T.; Morgado-Dias, F.; Ravelo-García, A.G. A-phase index: An alternative view for sleep stability analysis based on automatic detection of the A-phases from the cyclic alternating pattern. *Sleep* **2023**, *46*, zsac217. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.