



Universidad Autónoma de Ciudad Juárez

Instituto de Ingeniería y Tecnología

Departamento de Ingeniería Eléctrica y Computación

Maestría en Ingeniería Eléctrica

“Clasificación de los desórdenes del sueño a través de polisomnogramas usando un algoritmo de aprendizaje profundo”

Tesis para obtener el grado de:

Maestro en Ingeniería Eléctrica

Ingeniero. Marco Antonio Rodríguez Castrejón

“Becado por el Consejo Nacional de Ciencia y Tecnología”

Bajo la dirección de la

Dra. Leticia Ortega Máynez

y la codirección del

Dr. José Manuel Mejía Muñoz

Ciudad Juárez, Chihuahua, agosto de 2022

Clasificación de los desórdenes del sueño a través de polisomnogramas usando un algoritmo de aprendizaje profundo.

Ing. Marco Antonio Rodríguez Castrejón

Proyecto de tesis presentado como requisito para obtener el grado de:

Maestro en Ingeniería Eléctrica

Directora:

Dra. Leticia Ortega Máynez

Codirector:

Dr. José Manuel Mejía Muñoz

Línea de Investigación:

Procesamiento de señales

Universidad Autónoma de Ciudad Juárez

Instituto de Ingeniería y Tecnología, Departamento de Ingeniería Eléctrica y
Computación

2022

Agradecimientos y dedicatorias

Agradezco a la Dra. Leticia Ortega y al Dr. José Mejía por su experiencia y conocimiento brindado en esta investigación, sin ellos esto no hubiera sido posible. Los admiro por ser parte de mi formación académica como profesores.

Agradezco a la coordinación de la Maestría en Ingeniería Eléctrica por el seguimiento en mi carrera, aportando los recursos necesarios para este trabajo.

Agradezco a mis padres por su apoyo incondicional.

A Dania por tu amor y comprensión en todo momento.

Esta investigación se la dedico a mi hija Renata Daniela, que siempre ha sido mi inspiración para seguir creciendo profesionalmente.

Resumen

La presente investigación tiene como objetivo la clasificación o predicción de los siguientes trastornos del sueño: epilepsia nocturna del lóbulo frontal, apnea obstructiva del sueño, movimiento periódico de piernas y desorden de comportamiento durante la fase REM (movimiento rápido de los ojos), por medio de algoritmos de aprendizaje profundo con señales extraídas de polisomnogramas. Se consultaron las bases de datos públicas y privadas: *CAP Sleep Database* y *Wisconsin Sleep Cohort* (WSC), de las cuales se extrajeron polisomnogramas (PSG) de diez participantes de cada trastorno además de diez sin algún trastorno, de los cuales se utilizó las señales electroencefalograma (EEG1, EEG2), electrocardiograma (ECG) y saturación de oxígeno en sangre arterial (SaO2/SpO2). Primero se filtró el ruido de las señales por medio de un filtro FIR exceptuando la señal de saturación de oxígeno, posteriormente se segmentaron las señales en ventanas de 20 segundos con traslape para ser codificadas en imágenes por medio de funciones de transformación de series en el tiempo, las cuales alimentan a una red neuronal convolucional recurrente. Como resultado se obtuvo una precisión de 0.94.

Como conclusión podemos observar que esta investigación brinda una herramienta a los especialistas del área del sueño (psiquiatría, neurología, neurofisiología, etc.) para clasificar y diagnosticar los trastornos del sueño, ahorrándoles tiempo y automatizando el proceso con precisión sin embargo es necesario trabajar con más datos del PSG para obtener mejores resultados.

Abstract

The current investigation has as objective the classification or prediction for the following sleep disorders: nocturnal frontal lobe epilepsy, sleep-disordered breathing, periodic leg movement and REM behavior disorder, using deep learning algorithms with signals from polysomnography. The following public and private databases were consulted: CAP Sleep Database and Wisconsin Sleep Cohort (WSC), from which ten polysomnography (PSG) records were extracted from each disorder as well as ten records without any disorder, electroencephalogram (EEG1, EEG2), electrocardiogram (ECG) and arterial oxygen saturation (SaO₂/SpO₂) signals were used. At first all signals except oxygen saturation went through a FIR filter process to reduce noise, then they were split into 20 seconds windows to be codified into images with encoding time series functions, these images feed the recurrent convolutional neural network. As a result, we got an accuracy of 0.92.

We can conclude that this work will bring a tool for specialists in the sleep area (neurologists, psychiatrists, etc.), to classify and to diagnostic sleep disorders, saving time and resources by automating the process, in the other hand we need to continue working to improve the results using more PSG signals on it.

Declaración de originalidad

“Yo declaro que esta tesis, así como los resultados en ella reportados, son producto de mi trabajo con el apoyo permitido de terceros en cuanto a su concepción y análisis. Asimismo, declaro que hasta donde yo sé no contiene material previamente publicado o escrito por otra persona excepto donde se reconoce como tal a través de citas y con propósitos exclusivos de ilustración y comparación. En este sentido, afirmo que cualquier información presentada sin citar a un tercero es de mi propia autoría. Finalmente, declaro que la redacción de esta tesis es producto de mi propio trabajo con la dirección y apoyo de mi director de tesis y mi comité tutorial en cuanto a la concepción del proyecto, al estilo de la presentación o a la expresión escrita.”

Ing. Marco Antonio Rodríguez Castrejón

Derechos reservados

“La información confidencial no podrá ser reproducida por ningún medio ni en ningún formato por el receptor sin autorización previa escrita del autor, excepto por aquellas copias que el receptor pueda necesitar para hacer operativo este acuerdo. En caso de que el receptor fuere autorizado por el autor a reproducir total o parcialmente la información confidencial, todas las reproducciones, sean totales o parciales y cualquiera sea el formato en que se registren, deberán hacer expresa mención a la propiedad intelectual del autor sobre la información contenida en ellas, contando con anuncios de confidencialidad y manteniendo las leyendas que contenga la información original, salvo que el autor disponga otra cosa por escrito.”

Contenido

Capítulo 1	22
Planteamiento del problema	22
1.1. Descripción del problema.....	22
1.2. Objetivos	22
1.2.1. Objetivo general	22
1.2.2. Objetivos específicos.....	22
1.3. Justificación.....	22
1.4. Hipótesis.....	23
1.5. Preguntas de investigación	23
Capítulo 2	24
Antecedentes	24
Marco teórico	26
2.1. Polisomnograma.....	26
2.2. Señales del PSG	26
2.2.1. Señal EEG	26
2.2.2. Señal ECG.....	27
2.2.3. Señal SaO2 y SpO2	27
2.3. Trastornos del sueño.....	28
2.3.1. Apnea Obstructiva del sueño.....	28
2.3.2. Epilepsia nocturna del lóbulo frontal	28
2.3.3. Desorden del comportamiento del sueño en la fase REM.....	28
2.3.4. Desorden de movimientos periódicos de pierna.....	29
2.4. Aprendizaje profundo.....	29
2.4.1. Neurona	29
2.4.2. Redes neuronales convolucionales.....	30

2.4.3. Capas convolucionales	30
2.4.4. <i>Dropout</i>	31
2.4.5. Capas <i>pooling</i>	31
2.4.6. Función de activación ReLU	32
2.4.7. Función <i>Softmax</i>	32
2.4.8. Mecanismo general de atención.	33
2.5. Arquitecturas de redes neuronales convolucionales.....	33
2.5.1. AlexNet	33
2.5.2. VGGNet	34
2.5.3. GoogleNet	34
2.5.4. ResNet	36
2.6. Codificación de series en el tiempo en imágenes.....	37
2.6.1. GAF.....	37
2.6.2. MTF.....	38
Capítulo 3.....	40
Metodología	40
3.1. Introducción	40
3.2. Materiales y equipo	41
3.2.1. Hardware	41
3.2.2. Software	41
3.3. Métodos.....	42
3.3.1. Adquisición de datos	42
3.3.2. Preprocesamiento	43
3.3.3. Diseño de la arquitectura.....	46
Capítulo 4.....	48
Resultados	48
4.1. Experimentos.....	48

4.1.1. Experimento 1	48
4.1.2. Experimento 2	51
4.1.3. Experimento 3	54
4.1.4. Comparativa de los experimentos realizados	57
4.2. Discusión.....	58

Lista de Figuras

Figura 1. Bandas de frecuencia del EEG [15].	26
Figura 2. Formas de onda de una señal ECG [16].	27
Figura 3. Oxímetro de pulso [17].	27
Figura 4. Comparación entre neurona biológica y neurona artificial [23].	30
Figura 5. Modelo <i>CNN</i> [24].	30
Figura 6. <i>Kernel</i> 2 x 2 [24].	31
Figura 7. Antes y después de aplicar <i>dropout</i> a una red neuronal durante el entrenamiento [25].	31
Figura 8. <i>Max pooling</i> y <i>average pooling</i> [26].	32
Figura 9. Función ReLU [27].	32
Figura 10. Arquitectura AlexNet [29].	34
Figura 11. Arquitectura VGG16 [31].	34
Figura 12. Módulo <i>Inception</i> , en la izquierda estructura básica y en la derecha con filtros convolucionales 1x1 para reducción de esfuerzo computacional [32].	35
Figura 13. Arquitectura GoogLeNet [32,33].	35
Figura 14. Bloque residual con conexión de salto [34].	36
Figura 15. Arquitectura ResNet-18 [34].	37
Figura 16. Transformación de una serie en el tiempo en una imagen GAF [35].	38
Figura 17. Ilustración del mapa codificado con la matriz MTF. Sea X una secuencia de datos en el tiempo es discretizada en Q contenedores para luego calcular la Matriz de Transición de Markov W y finalmente construir su MTF con la ec. 3.10. Adicionalmente se reduce el tamaño de la imagen promediando los píxeles en cada parche sin traslape de 2x2. [35].	39
Figura 18. Diagrama a bloques de la metodología.	40
Figura 19. Señales en el dominio del tiempo extraídas del PSG.	42
Figura 20. Señal en el dominio del tiempo al dominio de la frecuencia [36].	43
Figura 21. Respuesta en la frecuencia del Filtro FIR para eliminar ruido en las señales EEG y ECG.	44
Figura 22. Imagen resultante de la transformación GASF, empezando por la parte superior de izquierda a derecha las señales EEG1 y EEG2; en la parte inferior de izquierda a derecha ECG y saturación de oxígeno.	45
Figura 23. Esquema del preprocesamiento de las señales del PSG.	45
Figura 24. Red neuronal propuesta.	47
Figura 25 Curva de entrenamiento para el experimento número 1, exactitud en entrenamiento mayor a 0.95, exactitud en validación menor a 0.90.	49

Figura 26 Curvas de la función de costo con el experimento número 1, en entrenamiento menor a 0.1, en validación 0.23 en su mejor desempeño.....	49
Figura 27 Matriz de confusión, la clase mejor clasificada es la <i>SDB</i> mientras que la clase con menor desempeño es la <i>NFLE</i> y <i>NORMAL</i>	50
Figura 28 Curva de entrenamiento para el experimento número 2, exactitud en entrenamiento de 0.98, exactitud en validación de 0.92.	52
Figura 29 Curvas de la función de costo con 0.04 en entrenamiento y 0.25 en validación.	52
Figura 30 Matriz de confusión, la clase mejor clasificada es la <i>SDB</i> mientras que la clase con menor desempeño es la <i>NORMAL</i>	53
Figura 31 Curva de entrenamiento para el experimento número 3, exactitud en entrenamiento de 0.98, exactitud en validación de 0.96.	55
Figura 32 Curvas de la función de costo con 0.03 en entrenamiento y 0.12 en validación.	55
Figura 33 Matriz de confusión, la clase mejor clasificada es la <i>SDB</i> mientras que la clase con menor desempeño es la <i>NORMAL</i>	56
Figura 34. Curvas ROC para las cinco clases, en promedio 0.99 de área.	57

Lista de Tablas

Tabla 1. Parámetros para el experimento número 1.....	48
Tabla 2. Métricas obtenidas experimento 2, el promedio en precisión de 0.79.	50
Tabla 3. Parámetros para el experimento número 3.....	51
Tabla 4. Métricas obtenidas experimento 2, el promedio en precisión de 0.91.	53
Tabla 5. Parámetros para el experimento número 3.....	54
Tabla 6. Métricas obtenidas experimento 3, el promedio en precisión de 0.91.	56
Tabla 7. Comparativa entre los experimentos realizados.....	57
Tabla 8. Comparativa entre el estado del arte actual.....	59

Lista de símbolos y abreviaturas

CNN	<i>Convolutional neural network</i>
ECG	Electrocardiograma
EEG	Electroencefalograma
FFT	<i>Fast fourier transform</i>
NFLE	<i>Nocturnal frontal lobe epilepsy</i>
PLM	<i>Periodic leg movements</i>
PSG	Polisomnograma
REM	<i>Rapid eye movement</i>
RBD	<i>REM behavior disorder</i>
SaO ₂ /SpO ₂	Saturación de oxígeno en la sangre
SDB	<i>Sleep disordering breathing</i>

Introducción

Día a día los trastornos de sueño se vuelven más frecuentes conforme el ritmo de vida se hace más acelerado, el estrés aumenta, la población crece, el clima cambia, nuevas enfermedades aparecen, se incrementan los déficits alimenticios, etc. Este tipo de trastornos se pueden presentar desde la etapa neonatal hasta la adultez. El sueño es considerado un proceso fisiológico de vital importancia para la salud integral de los seres humanos. Además, la falta de sueño es detonante de alteraciones en el procesamiento y retención de la memoria.

Los especialistas del área del sueño se enfrentan a tener que diagnosticar diversos trastornos del sueño tales como: narcolepsia, insomnio, bruxismo, epilepsia nocturna del lóbulo frontal, apnea obstructiva del sueño, movimientos periódicos de extremidades, desorden de comportamiento durante la fase REM, entre otros [1]. Los cuales, para poder detectarlos y diagnosticarlos, es necesario recurrir a la obtención de polisomnografía, que es la representación de un estudio de señales de diversos canales conectados al paciente por electrodos y sensores en diferentes zonas del cuerpo mientras se está dormido, como es en la parte superior, frontal y lateral de la cabeza [2]. Todo esto en un cuarto controlado; este proceso conlleva horas de obtención de las señales y debe monitorearse en todo momento, siendo el tiempo una limitante en el estudio.

Gracias al desarrollo de métodos de aprendizaje profundo, algunas tareas de la sociedad moderna que antes eran complejas ahora son fáciles de hacer, tales como: identificar objetos en imágenes, transcribir voz en texto, encontrar nuevos productos que simpaticen con los intereses de los usuarios, tareas de clasificación, etc. Para realizar lo anterior, existen varias arquitecturas de redes neuronales: las multicapas de perceptrones, que es la arquitectura más antigua y simple de redes neuronales, las redes convolucionales (CNN), de uso muy común en procesamiento de imágenes y redes neuronales recurrentes usadas para datos secuenciales como texto o series en el tiempo [3].

En este proyecto se desarrolló una arquitectura de aprendizaje profundo con redes neuronales convolucionales para la clasificación de trastornos de sueño. Para el entrenamiento será necesario extraer los polisomnogramas de dos bases de datos públicas y privadas seleccionadas de internet [4][5][6][7]. Las señales digitales obtenidas serán preprocesadas digitalmente por medio de filtros que ayudan a mejorar la señal final para posteriormente alimentar la red neuronal.

Los propósitos de este proyecto son: brindar una herramienta a especialistas del área, para que puedan clasificar y diagnosticar los trastornos del sueño ahorrándoles tiempo y automatizando el proceso con alta precisión, así como avanzar en el conocimiento generado en el tema.

Capítulo 1

Planteamiento del problema

1.1. Descripción del problema

Actualmente cada vez más personas, desde niños hasta adultos de la tercera edad, padecen de algún trastorno del sueño, debido a las siguientes razones: alto estrés en sus rutinas diarias, mala alimentación, problemas psicológicos, entre otros. Debido a esto se considera necesario estudiar los padecimientos del sueño debido a su importancia para poder llevar una vida normal. Se han desarrollado varias herramientas para clasificar los diversos trastornos del sueño. Sin embargo, por ser un campo de investigación reciente hay una amplia variedad de arquitecturas de aprendizaje profundo por desarrollar. Sin un óptimo clasificador diagnosticar a una persona hace que la tarea sea de horas de monitoreo de señales PSG por los especialistas.

Entre los padecimientos a clasificar fueron: apnea obstructiva del sueño, epilepsia nocturna del lóbulo frontal, movimientos periódicos de extremidades, desorden de comportamiento durante la fase REM.

1.2. Objetivos

1.2.1. Objetivo general

Clasificar diferentes trastornos del sueño para ayudar al diagnóstico de estos, por medio de un algoritmo de aprendizaje profundo.

1.2.2. Objetivos específicos

- Obtener base de datos con polisomnogramas.
- Diseñar una arquitectura de red de aprendizaje profundo para clasificar o predecir los trastornos de sueño: apnea obstructiva del sueño, epilepsia nocturna del lóbulo frontal, movimientos periódicos de extremidades, desorden de comportamiento durante la fase REM.
- Validar el modelo entrenado empleando datos de prueba.

1.3. Justificación

Los trastornos del sueño como su nombre lo indica, originan que una persona no duerma de forma adecuada ni con calidad, lo cual la puede llevar a padecer otras enfermedades tales como: depresión, diabetes, hipertensión y enfermedades cardiovasculares. Con la clasificación de los trastornos del sueño será más fácil diagnosticar y encontrar una solución para tratar estas enfermedades. Este proyecto está enfocado en encontrar la arquitectura de red neuronal de aprendizaje profundo óptima para clasificar los diferentes trastornos del sueño, la cual pueda ser usada por

médicos especialistas en el área del sueño para rápidamente encontrar la solución o medicación más efectiva y así mejorar la calidad de vida del paciente. En la actualidad este proceso se lleva a cabo de modo manual y requiere muchas horas para completarlo, ya que el paciente es interconectado a varios sensores para extraer las diferentes señales del PSG, de igual modo esto es incómodo para el participante y requiere supervisión de técnicos especialistas durante todo este tiempo.

Se elige usar una arquitectura de aprendizaje profundo por redes neuronales para automatizar el proceso, ya que estos métodos han demostrado ser eficientes y rápidos a la hora de clasificar trastornos de sueño [8][9][10][11][12][13].

1.4. Hipótesis

A través del PSG y aprendizaje profundo es posible clasificar los trastornos del sueño con una exactitud mayor a 0.90.

1.5. Preguntas de investigación

¿Qué tipos de capas de aprendizaje profundo pueden obtener características relevantes para clasificar el PSG?

¿Cuál es la arquitectura básica de la red neuronal de aprendizaje profundo que pueda clasificar mejor?

¿Cómo clasificar los trastornos de sueño por medio del PSG?

¿Qué señales del PSG son relevantes para la clasificación de los trastornos del sueño?

¿Cómo se manejará la longitud de la señal del PSG?

Capítulo 2

Antecedentes

Desde que los trastornos de sueño siguen en aumento constante y afectando actividades cotidianas en la vida de las personas, se han ido desarrollando técnicas de diagnóstico de estos para su temprano tratamiento, la mayoría de ellas por medio de polisomnogramas que son estudios que contienen información de diferentes señales del cuerpo mientras está dormido. Las técnicas han ido avanzando hasta llegar al uso de algoritmos de aprendizaje profundo, los cuales hacen que la tarea de diagnóstico sea más rápida.

Las arquitecturas principales para este tipo de diagnóstico han sido las CNN que han logrado una precisión mayor a 0.90, sin embargo, la mayoría de ellos solo se enfocan en clasificar un trastorno en específico y utilizando un solo tipo de señal.

En 2016, Mellado, *et al.* [8] desarrollaron un algoritmo de aprendizaje profundo para detectar apnea obstructiva, apnea central e hipo apnea del sueño por medio del PSG usando las señales ECG, presión sanguínea, EEG y flujo respiratorio, todas las señales fueron muestreadas a 250 Hz provenientes de 8 participantes distintos. La arquitectura de la red neuronal propuesta se constituye de 6 capas convolucionales con función de activación ReLU y con un abandono (*dropout*) entre cada capa de 0.5 para evitar el sobreajuste del modelo. Con este algoritmo, obtuvieron una exactitud del 0.68 y una precisión del 0.67, sin embargo, al evaluar utilizando una matriz de confusión, el clasificador se confundió al tratar de detectar el desorden hipo apnea de la señal de sueño normal y no logró predecir los eventos de apnea central, siendo útil únicamente para detectar apnea obstructiva.

En 2019 Chowdhury, *et al.* [9] diseñaron una arquitectura de aprendizaje profundo para diagnosticar convulsiones por medio de señales EEG con una red convolucional 1-D. Está compuesta de una capa convolucional de tamaño 173 con 23 filtros y pasos (*strides*) de 1, una capa de *maxpooling* de tamaño 173 y una capa densa con activación sigmoidea; la información fue dividida en segmentos representados con las primeras 5 letras del alfabeto donde A y B pertenecen a participantes con condiciones normales y C, D, E y F que presentan epilepsias en diferentes categorías. Los resultados que obtienen para diferenciar cada grupo de clases siempre dan como resultado una precisión de 0.994 ± 0.005 .

En 2020, Wickramaratne y Mahmud [10] propusieron un método de aprendizaje profundo para detectar esfuerzo respiratorio durante el sueño mediante el PSG por medio de las señales EEG, ECG, EOG y SaO₂, muestreadas a 200 Hz, las cuales se les aplicó un filtro FIR y fueron sub muestreadas a 50 Hz antes de ser inyectadas a una red neuronal recurrente, la arquitectura de la red

consta de capas bidireccionales LSTM con funciones de activación ELU y una capa densa distribuida en el tiempo al final. Esta arquitectura fue aplicada en tres modelos diferentes, uno para la señal EEG y EOG, el segundo para las señales EMG y ECG y el tercero para las EEG y SaO₂ y al final los tres modelos fueron combinados para dar una mejor predicción. Para validar su clasificador utilizaron curvas AUPRC (área debajo de la curva de precisión) y AUROC (área debajo de la curva de recepción de características de operación) las cuales dieron un resultado de 0.59 y 0.95 respectivamente.

En 2020, Pisano, *et al.* [11] construyeron una red neuronal convolucional para detectar epilepsia nocturna del lóbulo frontal por medio de señales EEG muestreadas a 256 Hz, la arquitectura de la red neuronal utilizada se basó en una red profunda residual compuesta de capas convolucionales, *max pooling* y *average pooling*. Dentro de cada capa convolucional se agregó una capa de *dropout* de 0.20 para evitar sobre entrenamiento, y el algoritmo que se utilizó para optimizar el entrenamiento fue el gradiente descendiente estocástico con momento, obtuvieron resultados de precisión entre 0.94 y 0.96.

En 2020, Vaquerizo, *et al.* [12] desarrollaron un modelo de arquitectura *deep learning* para el diagnóstico de la apnea del sueño infantil, por medio de señales SpO₂ obtenidos de base de datos del PSG muestreados a 512 Hz, su red se compuso de módulos *Inception*, los cuales procesan datos de entrada mediante varios caminos en paralelo compuestos de capas convolucionales, cada uno de estos concatenando cada una de sus salidas, lo cual permitió aprender características a distintas escalas en el mismo nivel de abstracción. Los resultados de esta arquitectura están basados en el índice de hipo apnea (IAH), con un IAH=1 alcanzaron una precisión de 0.792, con IAH=5 obtuvieron 0.96 y con IAH=10 alcanzaron 0.98 superando a los trabajos anteriores que no utilizan módulos *Inception*.

En 2020, K. Palani, *et al.* [13] desarrollaron una red neuronal para detectar epilepsia por medio de imágenes generadas a partir de GASF (*Gramian angular summation field*). Utilizando una red convolucional pre entrenada con los pesos de *ImageNet* con tres capas convolucionales con función de activación ReLU seguido de una capa de normalización por lote (BN) y una capa de *MaxPooling*. Después de esto, capas densas con función de activación sigmoidea para la extracción de características de alto nivel y finalmente una capa de clasificación *SoftMax* binaria para las clases normal y focal. Con este modelo ellos logran obtener una precisión de 0.88 en promedio y área bajo la curva de 0.92 los cuales son superiores a otras arquitecturas como AlexNet, VGG16, VGG19, entre otras.

Marco teórico

En esta sección se definirán los conceptos más importantes de los estudios del PSG, como lo son las diferentes señales que lo componen y algunos de los trastornos que se pueden diagnosticar con esta técnica.

2.1. Polisomnograma

El PSG es un estudio que se realiza mientras el participante se encuentra dormido, donde un especialista observa la información grabada de los patrones del sueño y puede identificar trastornos del sueño. Durante un PSG el especialista mide y analiza las siguientes señales: ondas cerebrales, actividad muscular, niveles de oxígeno en la sangre, frecuencia cardíaca, respiración y movimiento ocular [14].

2.2. Señales del PSG

En las siguientes subsecciones se explicarán las señales más importantes del PSG para este trabajo en cuestión.

2.2.1. Señal EEG

El electroencefalograma (EEG) es una herramienta que representa gráficamente las oscilaciones que generan la actividad eléctrica del cerebro, la cual se detecta mediante electrodos conectados en el cuero cabelludo en distintas zonas de la cabeza. Kumar y Bhuvaneshwari [15] clasifican las ondas generadas por el EEG de acuerdo con su rango de frecuencia, amplitud, forma y posición del electrodo en la cabeza en: *delta*, *theta*, *alpha*, *beta* y *gamma* (Figura 1).

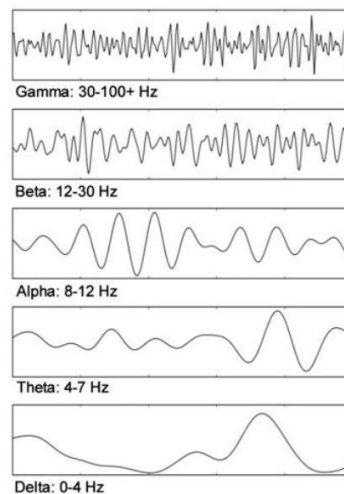


Figura 1. Bandas de frecuencia del EEG [15].

2.2.2. Señal ECG

El electrocardiograma (ECG) representa la actividad eléctrica del corazón. Es una señal que muestra la relación entre el potencial eléctrico del corazón y el tiempo. Los latidos del corazón son caracterizados por ondas eléctricas con picos y valles. El rango de frecuencia oscila entre 0.05 y 100 Hz y su amplitud entre 1 y 10 mV. Para medirla se colocan electrodos o cables en regiones específicas del cuerpo como brazos, pecho o piernas y esta se va registrando en un monitor o en papel. La señal se caracteriza por cinco picos y valles representados por las letras P, Q, R, S, T y en algunos casos U. Un ECG típico se muestra en la Figura 2 [16].

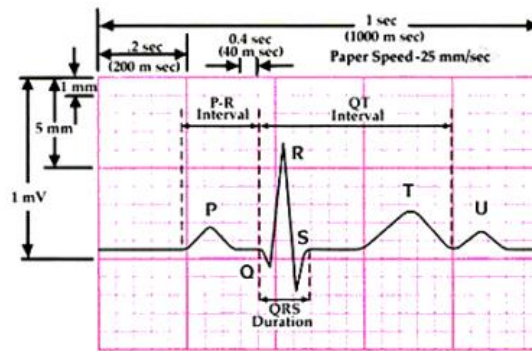


Figura 2. Formas de onda de una señal ECG [16].

2.2.3. Señal SaO2 y SpO2

De acuerdo con [17], SaO₂ es la saturación de oxígeno en la sangre arterial, la cual indica el porcentaje de moléculas de hemoglobina en la sangre arterial saturada con oxígeno; varía entre 0 y 100 % pero los rangos normales van entre 94 y 100% de una persona sana, SpO₂ es la saturación periférica de oxígeno o saturación de oxígeno en la sangre arterial y medida con un oxímetro de pulso (Figura 3).



Figura 3. Oxímetro de pulso [17].

2.3. Trastornos del sueño

Los trastornos del sueño son los problemas que involucran la calidad, tiempo y cantidad de sueño, los cuales a menudo traen consigo condiciones médicas o mentales como la depresión, ansiedad, trastornos cognitivos, entre otros. En esta sección se describirán los siguientes: apnea obstructiva del sueño, epilepsia nocturna del lóbulo frontal, desorden del comportamiento en la fase REM y desorden de movimiento periódico de pierna.

2.3.1. Apnea Obstructiva del sueño

Este trastorno consiste en episodios de cierre parcial o total de las vías aéreas superiores que se producen durante el sueño y que dan lugar a una respiración interrumpida (un periodo de apnea mayor a 10 segundos). Los síntomas incluyen somnolencia diurna excesiva, inquietud, ronquidos, despertar recurrente y dolor de cabeza. Para el diagnóstico se requiere del PSG con el cual se mide el flujo constante de aire en la nariz y boca por medio de sensores de flujo, saturación de oxígeno por oximetría, arquitectura del sueño por EEG, electromiografía y electro-oculograma para evaluar la aparición de movimientos oculares rápidos [18].

2.3.2. Epilepsia nocturna del lóbulo frontal

La epilepsia nocturna del lóbulo frontal (NFLE) se caracteriza por convulsiones del lóbulo frontal que ocurren durante el sueño y a su vez causa problemas de agitación en el paciente. Principalmente se da entre los 7 y los 12 años, pero se han documentado casos desde la niñez hasta la adultez. Para su diagnóstico se somete al paciente a grabaciones de EEG durante la noche mientras duerme donde un especialista analiza el tiempo y el número de episodios de convulsiones [19].

2.3.3. Desorden del comportamiento del sueño en la fase REM

El desorden de comportamiento REM (RBD), es un trastorno del sueño caracterizado por comportamientos inusuales al perder la atonía del sueño en la fase REM, estos se manifiestan en violentos movimientos con las extremidades del cuerpo que pueden hacer daño al participante o al compañero de cama. Se ha encontrado que la gran mayoría de personas que padecen RBD eventualmente desarrollan signos de la enfermedad de Parkinson o un trastorno similar como atrofia del sistema múltiple o demencia por cuerpos de Lewi [20].

2.3.4. Desorden de movimientos periódicos de pierna

El movimiento periódico de piernas (PLM), es un trastorno del sueño que afecta del 4% al 11% de la población. Se caracteriza por espasmos involuntarios de las extremidades del cuerpo que suceden cada periodo de tiempo (entre cada 5 a 90 segundos) hasta por una hora. Esto causa que el individuo tenga cansancio durante el día. Este trastorno puede ocurrir en cualquier edad [21].

2.4. Aprendizaje profundo

Aprendizaje profundo es un conjunto de métodos que transforman la representación de los datos de un bajo nivel (comúnmente datos crudos) en una representación de un alto nivel más abstracto, por medio de funciones no lineales e iterativas. Una imagen, por ejemplo, que entra a una red como un arreglo de valores de píxeles, en la primera capa las características aprendidas representan la presencia o ausencia de los bordes en diferentes regiones de la imagen. La segunda capa detecta los patrones en cómo están distribuidos los bordes. La tercera capa encuentra las combinaciones de un objeto en particular y las capas posteriores detectarán todas las combinaciones de los objetos en la imagen [22].

2.4.1. Neurona

La unidad fundamental del cerebro humano es la neurona, se puede decir que en un grano de arroz se encuentran 10,000 neuronas, las cuales forman en promedio 6,000 conexiones con otras neuronas, esta red nos permite percibir el mundo que nos rodea. La neurona recibe información de otras neuronas, la procesa y manda el resultado a otras células. La neurona recibe en sus entradas por medio de unas antenas llamadas dendritas. Cada una de estas conexiones de fortalecerá o debilitará dependiendo de qué tan a menudo sean usadas, y es la fuerza de cada una de estas conexiones la que determina la contribución de la entrada con respecto a su salida. Las entradas son sumadas en el cuerpo humano, esta suma es transformada en una nueva señal que se propaga por las células del axón y mandado a otras neuronas. Todo este funcionamiento de las neuronas se puede trasladar en un modelo artificial que se puede representar en un computador (Figura 4). Justo como las neuronas biológicas, las neuronas artificiales toman las entradas x_1, x_2, \dots, x_n , cada una de las cuales es multiplicada por un coeficiente específico, w_1, w_2, \dots, w_n . Posteriormente se obtiene la suma ponderada $z = \sum_0^n w_i x_i$ la cual pasa por una función f para producir la salida $y = f(z)$, esta a su vez puede ser transmitida a otras neuronas [23].

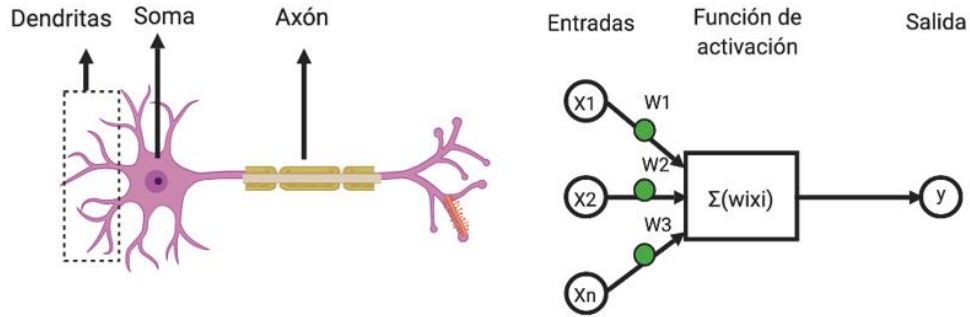


Figura 4. Comparación entre neurona biológica y neurona artificial [23].

La neurona artificial se puede expresar en forma vectorial donde las entradas son el vector $\mathbf{x} = [x_1, x_2, \dots, x_n]$ y los coeficientes o pesos el vector $\mathbf{w} = [w_1, w_2, \dots, w_n]$. De esta manera la salida de la neurona se representa como $y = f(\mathbf{x} \cdot \mathbf{w} + b)$ donde b es el sesgo de la función.

2.4.2. Redes neuronales convolucionales

De acuerdo con [24], las CNN utilizan una arquitectura de alimentación hacia delante (*feed forward*) y pueden aprender características de objetos e identificarlos con alta precisión. Un modelo profundo de CNN consiste en un número finito de set de capas que aprenderán y extraerán características de las entradas y de capas profundas que se encargarán de las características de bajo nivel, un concepto básico de este modelo se puede observar en la Figura 5.

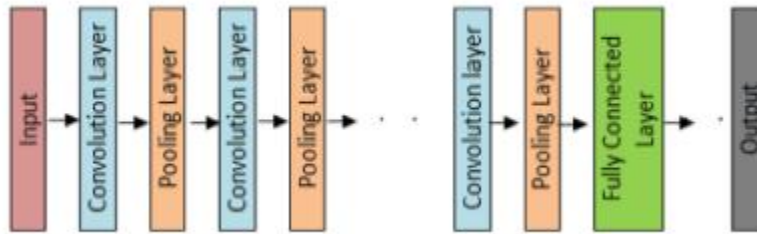


Figura 5. Modelo CNN [24].

2.4.3. Capas convolucionales

Una capa convolucional contiene una cantidad de *kernels* (también llamados filtros) que se convolucionan con la imagen de entrada para generar un mapa de características a la salida.

Un *kernel* puede describirse como una matriz de valores discretos donde cada valor es conocido como el peso del *kernel*. Durante el comienzo del entrenamiento de una CNN, los pesos del *kernel* son asignados con valores aleatorios. En cada época los pesos son afinados para que el filtro

aprenda a extraer las características más importantes, en la Figura 6 se observa un ejemplo de este *kernel* [24].

0	1
-1	2

Figura 6. *Kernel* 2 x 2 [24].

2.4.4. *Dropout*

Es una técnica para optimizar las redes neuronales reduciendo el efecto de *overfitting*. Se refiere a temporalmente (durante el entrenamiento) desactivar unidades de la red junto con sus conexiones de entrada y salida como se observa en la Figura 7. La elección de que unidades desactivar se realiza de manera aleatoria. Cada unidad permanecerá en la red si cumple con una probabilidad p previamente definida. Típicamente se sugiere asignar el valor a 0.5 para las unidades de salida y un valor cercano a 1.0 para las unidades de entrada [25].



Figura 7. Antes y después de aplicar *dropout* a una red neuronal durante el entrenamiento [25].

2.4.5. Capas *pooling*

Las capas de agrupación permiten reducir la dimensión de la *CNN*, también conocidas como submuestreo (*subsampling*), tomando el promedio o el máximo valor en los parches de la imagen (*mean/average-pooling* o *max-pooling*), ver Figura 8. Estas actúan en pequeños parches (*patches*) de la imagen, también tienen paso (*stride*). Si consideramos 2 x 2 parches, de los cuales obtenemos el valor máximo para definir la capa de salida, y un paso $s = 2$, se divide entre 2 el ancho y alto de la imagen. Aunque también es posible reducir la dimensión con una capa convolucional tomando un

paso más grande que 1, y sin llenado de ceros, pero la ventaja de la capa de agrupación es que hace la red menos sensible a pequeñas traslaciones de las imágenes de entrada [26].

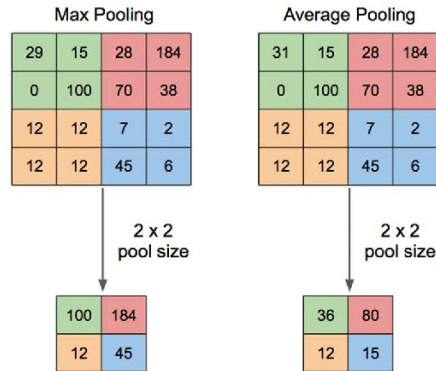


Figura 8. Max pooling y average pooling [26].

2.4.6. Función de activación ReLU

Como se menciona en [27], la función ReLU es utilizada para convertir las entradas en números positivos y está definida por la siguiente expresión:

$$f(x) = \max(0, x) \tag{3.1}$$

Donde x es la entrada de cada neurona. Esta función se ve representada en la Figura 9.

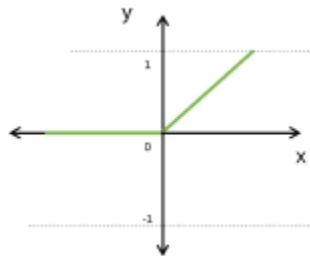


Figura 9. Función ReLU [27].

2.4.7. Función Softmax

La función *Softmax* [27], es una función aplicada en la última capa de la red neuronal para realizar la clasificación de los distintos elementos. El resultado de esta función es una distribución de probabilidades (categórica), dentro de un vector \mathbf{z} con elementos $[z_1, z_2, \dots, z_x]$. Por lo tanto, el

resultado estará definido entre $[0,1]$. En la ecuación (3.2) se define la función $\sigma(\cdot)$ de *softmax* para un vector \mathbf{z} .

$$\sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_0^k e^{z_k}} \quad \forall j \dots k \quad (3.2)$$

2.4.8. Mecanismo general de atención.

El mecanismo de atención general utiliza tres componentes principales, las consultas Q (*queries*), las llaves K (*keys*) y los valores V (*values*), realiza las siguientes operaciones computacionales [28]:

1. Cada valor de consulta, $q = s_{t-1}$, es relacionado con una base de datos de llaves para obtener un valor de puntuación. Esto se realiza con la operación del producto punto de la consulta en específico bajo consideración con cada llave del vector k_i :

$$e_{q,k_i} = q \cdot k_i \quad (3.3)$$

2. Las puntuaciones son transferidas por medio de la operación *softmax* para generar los pesos:

$$\alpha_{q,k_i} = \text{softmax}(e_{q,k_i}) \quad (3.4)$$

3. La atención generalizada es computada por una suma de los vectores de valores V_{k_i} , donde cada vector de valor es relacionado con su correspondiente llave:

$$\text{attention}(q, K, V) = \sum_i \alpha_{q,k_i} V_{k_i} \quad (3.5)$$

2.5. Arquitecturas de redes neuronales convolucionales

A lo largo de la última década se han desarrollado diferentes arquitecturas de redes neuronales, en las siguientes secciones se mencionan las más importantes en la clasificación de imágenes.

2.5.1. AlexNet

En 2012 Alex Krizhevsky, et al. [29] proponen una arquitectura de red neuronal convolucional compuesta de 5 capas convolucionales y 3 capas completamente conectadas, la función ReLU es aplicada a la salida de cada una de ellas, utiliza una capa de *Maxpooling* en la última capa convolucional, además de *dropout* y aumento de datos para evitar sobre entrenamiento; como

optimizador utilizan SGD con tamaño de lote de 128, *weight decay* de 0.0005 y factor de momento 0.9. En la Figura 10 se ve la representación de la red.

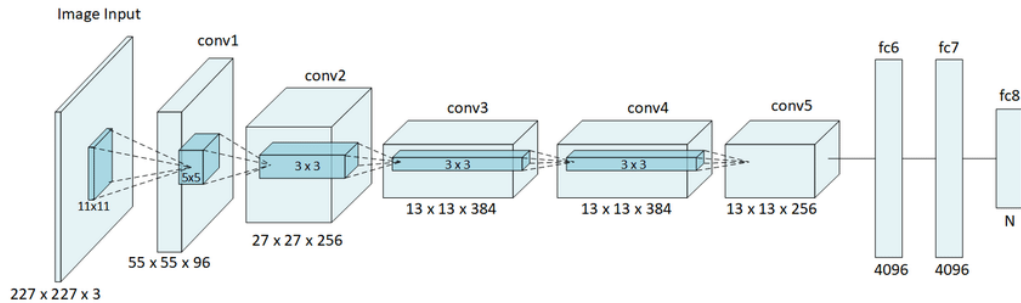


Figura 10. Arquitectura AlexNet [29].

2.5.2.VGGNet

Propuesta en 2014 por Karen Simonyan y Andrew Zisserman [30], la entrada de una VGG es básicamente una imagen RGB de (224x224x3), compuesta de capas convolucionales con filtros de 3x3, capas *Maxpooling* de 2x2 y capas completamente conectadas al final dando un total de 16 capas como se ve en la Figura 11.



Figura 11. Arquitectura VGG16 [31].

2.5.3.GoogleNet

Propuesta en 2014 por Szegedy, et al. [32], compuesto de 22 capas, dentro de las que destacan los módulos *Inception*, los cuales consisten en capas convolucionales con filtros 1x1, 3x3, 5x5 cada uno y una capa *Maxpooling* ocurriendo todas ellas en paralelo, la salida de cada uno de estos módulos se convierte en la entrada del siguiente, y al final de la red, una capa de *Average Pooling*. Debido al procesamiento computacional alto, a cada uno de estos módulos se les agregaron capas convolucionales 1x1 para reducir la dimensión de las salidas de cada módulo. En la Figura 12 se observa la estructura de un módulo *Inception*.

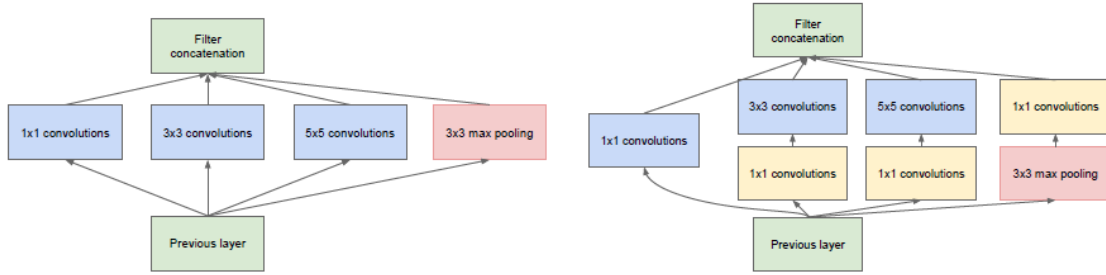


Figura 12. Módulo *Inception*, en la izquierda estructura básica y en la derecha con filtros convolucionales 1x1 para reducción de esfuerzo computacional [32].

La arquitectura completa de la GoogLeNet usa un total de nueve bloques de tipo *Inception* y esta se observa en la Figura 13. Como lo explican en [33], un bloque *Inception* es equivalente a una subred con cinco caminos que extraen información en paralelo con las capas convolucionales de tamaño 3 x 3 y 5 x 5. Las convoluciones 1x1 reducen las dimensiones del canal a nivel por píxel mientras que el *Max Pooling* 3 x 3 reduce la resolución.

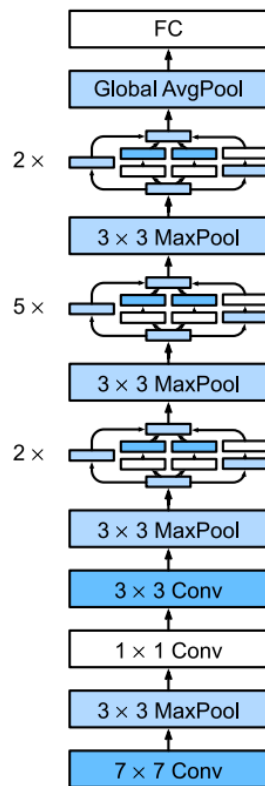


Figura 13. Arquitectura GoogLeNet [32,33].

2.5.4. ResNet

Diseñada por He, et al. [34] de Microsoft, se compone por bloques residuales que utilizan conexiones de salto como se ve en la Figura 14, para atacar el problema del desvanecimiento del gradiente. Como el gradiente se mantiene fluyendo hacia las capas iniciales, su valor permanece siendo multiplicado por cada gradiente local (por la regla de la cadena) y por esto cada vez más pequeño, haciendo las actualizaciones muy pequeñas e incrementando el tiempo de entrenamiento. Para resolver este problema es necesario que el gradiente local sea 1 por medio de la función identidad, ya que la derivada de esta es siempre 1. La ResNet se encarga de esto modelando la función identidad por medio de las conexiones de salto.

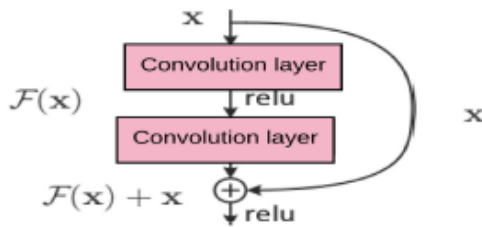


Figura 14. Bloque residual con conexión de salto [34].

Al igual que *GoogLeNet* usa cuatro módulos elaborados con bloques *Inception*, *ResNet* usa cuatro módulos hechos de bloques residuales, cada uno de los cuales contiene varios bloques residuales con el mismo número de canales de salida. En la Figura 15 se observa la arquitectura de una ResNet-18, cuatro capas convolucionales en cada módulo *ResNet*, una capa convolucional al inicio de la red de 7×7 y una capa completamente conectada.

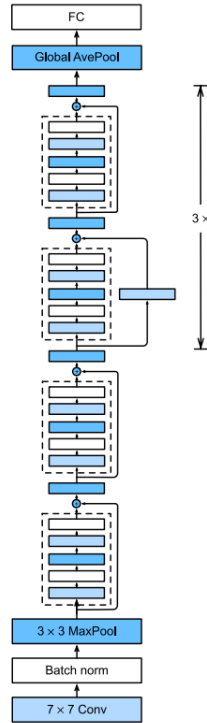


Figura 15. Arquitectura ResNet-18 [34].

2.6. Codificación de series en el tiempo en imágenes

En la última década se han ido desarrollando algoritmos de redes neuronales a base de alimentación por imágenes, aprovechando esto, muchos autores han optado por utilizar métodos para transformar series en el tiempo en imágenes, dos técnicas se mencionan a continuación, introducidas por Wang y Oates en 2015 [35].

2.6.1. GAF

Gramian Angular Field (Wang y Oates 2015) es una representación de series en el tiempo en coordenadas polares. Dada una serie en el tiempo $X = \{x_1, x_2, \dots, x_n\}$ de n observaciones de valor real, primero escalamos X de manera que todos los valores estén en el intervalo $[-1,1]$:

$$\bar{x}_i = \frac{(x_i - \max(X)) + (x_i - \min(X))}{\max(X) - \min(X)} \quad (3.6)$$

Posteriormente se representan los nuevos valores de la serie en coordenadas polares codificando el valor como un coseno angular y los pasos en el tiempo como radios:

$$\begin{cases} \phi = \arccos(\bar{x}_i), -1 \leq \bar{x}_i \leq 1, \bar{x}_i \in \bar{X} \\ r = \frac{t_i}{N}, t_i \in \mathbb{N} \end{cases} \quad (3.7)$$

Donde t_i son los saltos en el tiempo y N es un factor constante para regularizar la amplitud del sistema de coordenadas polares. Una vez hecho lo anterior, se puede calcular la matriz GAF de imágenes aprovechando las propiedades trigonométricas.

$$G = \begin{pmatrix} \cos(\phi_1 + \phi_1) & \cdots & \cos(\phi_1 + \phi_n) \\ \cos(\phi_2 + \phi_1) & \cdots & \cos(\phi_2 + \phi_n) \\ \vdots & \ddots & \vdots \\ \cos(\phi_n + \phi_1) & \cdots & \cos(\phi_n + \phi_n) \end{pmatrix} \quad (3.8)$$

$$G = \tilde{X} \cdot \tilde{X} - \sqrt{I - \tilde{X}^2} \cdot \sqrt{I - \tilde{X}^2} \quad (3.9)$$

Donde I es un vector fila unitario $[1, 1, \dots, 1]$. La Figura 16 ilustra este procedimiento.

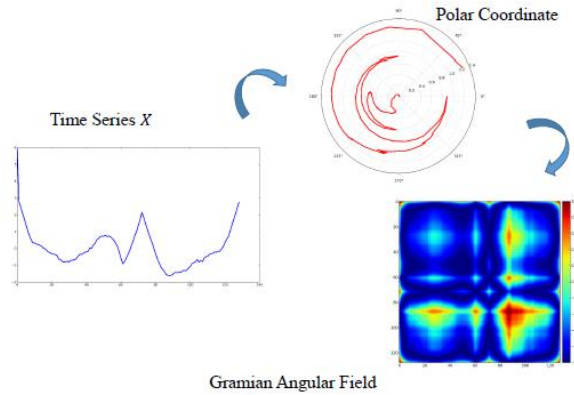


Figura 16. Transformación de una serie en el tiempo en una imagen GAF [35].

2.6.2.MTF

Markov Transition Field (Wang y Oates 2015), consiste en una matriz de probabilidades para observar cambios de valor en cualquier par de puntos en una serie en el tiempo. El campo de transición de Markov está definido por la siguiente matriz:

$$M = \begin{pmatrix} w_{ij}|x_1 \in q_i, x_1 \in q_j & \cdots & w_{ij}|x_1 \in q_i, x_n \in q_j \\ w_{ij}|x_2 \in q_i, x_1 \in q_j & \cdots & w_{ij}|x_2 \in q_i, x_n \in q_j \\ \vdots & \ddots & \vdots \\ w_{ij}|x_n \in q_i, x_1 \in q_j & \cdots & w_{ij}|x_n \in q_i, x_n \in q_j \end{pmatrix} \quad (3.10)$$

Dividiendo la magnitud de los datos en Q contenedores genera la matriz $Q \times Q$. Los contenedores con los datos en pasos en el tiempo i, j se definen como q_i, q_j . Mientras que w_i es la probabilidad de $q_i \rightarrow q_j$. La metodología para obtener una imagen MTF se presenta en la Figura 17.

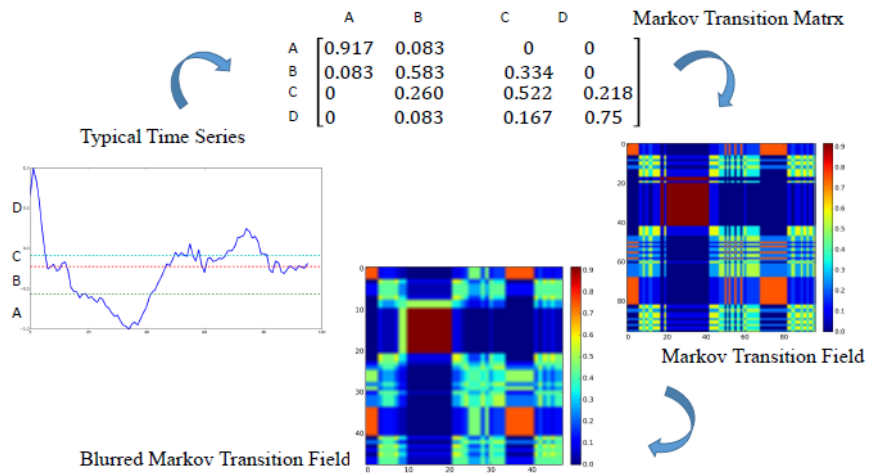


Figura 17. Ilustración del mapa codificado con la matriz MTF. Sea X una secuencia de datos en el tiempo es discretizada en Q contenedores para luego calcular la Matriz de Transición de Markov W y finalmente construir su MTF con la ec. 3.10. Adicionalmente se reduce el tamaño de la imagen promediando los pixeles en cada parche sin traslape de 2×2 . [35].

Capítulo 3

Metodología

En este capítulo se mostrará la metodología empleada para la realización de esta investigación, así como las herramientas utilizadas.

3.1. Introducción

Los pasos para el desarrollo de este proyecto fueron: la adquisición de datos de diferentes bases de datos de la red, públicas y privadas, preprocesamiento de los datos para eliminar el ruido, así como convertirlos a imágenes y el diseño de la red de aprendizaje profundo para la clasificación de los desórdenes del sueño. El diagrama a bloques de la Figura 18 ilustra la metodología de este proyecto.

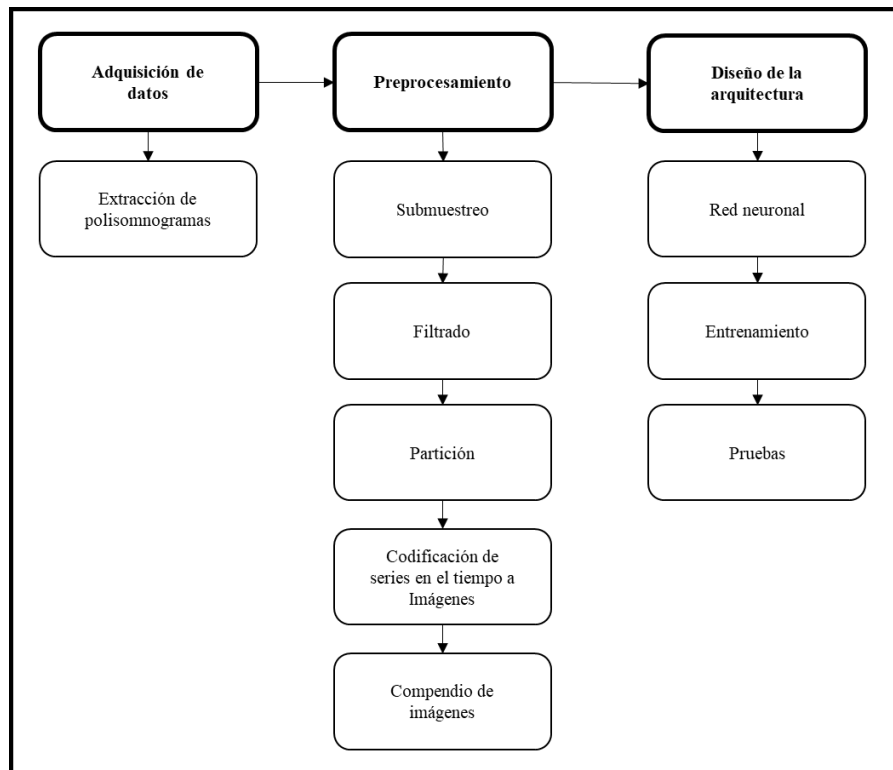


Figura 18. Diagrama a bloques de la metodología.

3.2. Materiales y equipo

Para este proyecto basado en procesamiento digital de señales se utilizaron materiales de hardware y software.

3.2.1. Hardware

A continuación, se enlista el hardware o equipo físico utilizado.

Computadora

La computadora que se utilizó para el desarrollo de esta investigación y con la que se realizó todo el preprocesamiento de las señales adquiridas de las bases de datos, tiene las especificaciones que se muestran a continuación:

- ◆ PC Asus G75vw
- ◆ Procesador Intel Core i7 a 2.4 GHz
- ◆ Memoria RAM de 20 GB
- ◆ Tarjeta de video NVIDIA GEFORCE GTX 660M de 2GB

3.2.2. Software

Los elementos de software o elementos no físicos se describen en los siguientes párrafos.

Bases de datos

Toda la información se extrajo de bases de datos que se encuentran en la red las cuales se enumeran a continuación:

- ◆ Base de datos publica: CAP Sleep Database v1.0.0
- ◆ Base de datos privada: Wisconsin Sleep Cohort (WSC)

Python 3.7

El desarrollo de todo el código tanto del preprocesamiento como el entrenamiento se realiza con el entorno de programación de Python, siendo este un recurso gratuito se ha ido popularizando en los últimos años, las librerías que se utilizaron para ello son:

- ◆ numpy
- ◆ matplotlib.pyplot
- ◆ mne.io
- ◆ scipy
- ◆ yasa
- ◆ tensorflow
- ◆ keras
- ◆ sklearn
- ◆ pandas

Google Colab

El entrenamiento de la red se vuelve una tarea que consume grandes recursos, así como tiempo de ser realizado en una computadora comercial, Google pone a disposición de los usuarios sus computadoras virtuales que cuentan con mayores recursos tecnológicos lo cual hace que la tarea de entrenamiento se vuelva más rápida, algunas de sus especificaciones de la versión Google Colab Pro son:

- ◆ GPU T4 & P100
- ◆ Memoria RAM 25 GB
- ◆ 24 horas continuas de entrenamiento

3.3.Métodos

A continuación, se describen los métodos utilizados para el desarrollo del proyecto.

3.3.1.Adquisición de datos

Se obtuvieron polisomnogramas de las bases de datos públicas y privadas: *CAP Sleep Database* v1.0.0 la cual contiene las clases: NFLE, NORMAL, PLM y RBD; y *Wisconsin Sleep Cohort* (WSC) de la cual se extrajo la clase SDB. El formato para estos archivos es EDF, el cuál es un estándar europeo para almacenar polisomnogramas. De cada clase se obtuvo diez polisomnogramas de cuatro horas de sueño aproximadamente. Para acceder a ellos se usó la librería de mne de Python con la cual se seleccionan dos canales de EEG (EEG1 y EEG2), ECG y SaO2/SpO2 para su posterior procesamiento (Figura 19).

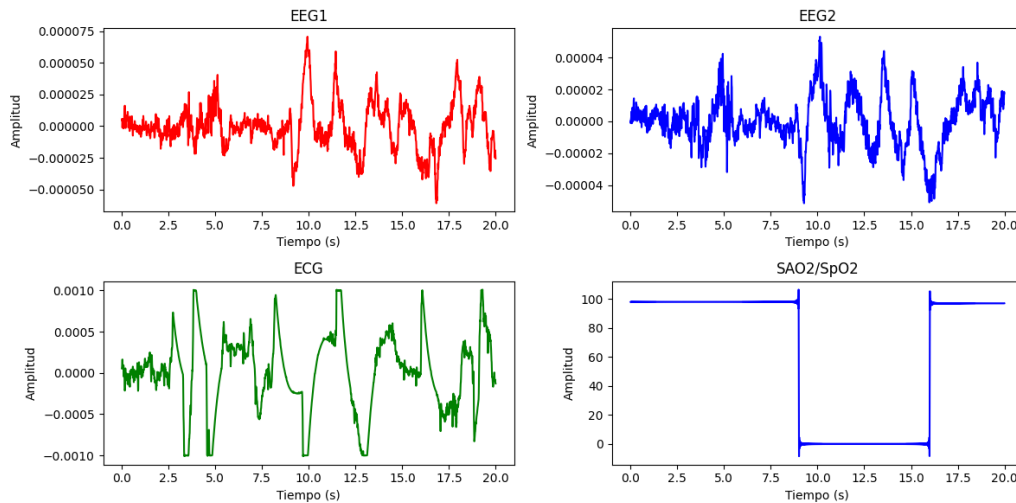


Figura 19. Señales en el dominio del tiempo extraídas del PSG.

3.3.2.Preprocesamiento

El preprocesamiento se llevó a cabo de la siguiente manera:

- I. Submuestreo. Debido a que las señales están a diferentes frecuencias de muestreo, se realizó un submuestreo de todas las señales por medio de la librería de *mne* utilizando su función *resample*, todas fueron submuestreadas a 100 Hz.
- II. Filtrado de ruido. Se analizaron las señales obtenidas en la sección anterior, en el dominio de la frecuencia por medio de la transformada de *Fourier* rápida *FFT*, utilizando la función de *numpy fft* (ver Figura 20), y se observa que las frecuencias más importantes para las señales EEG están en el rango de 0.5 y 40 Hz, mientras que para la señal ECG están entre 5 y 35 Hz, para la señal de saturación de oxígeno no se realizó análisis de frecuencias ya que esta viene dada en porcentaje de 0 a 100%.

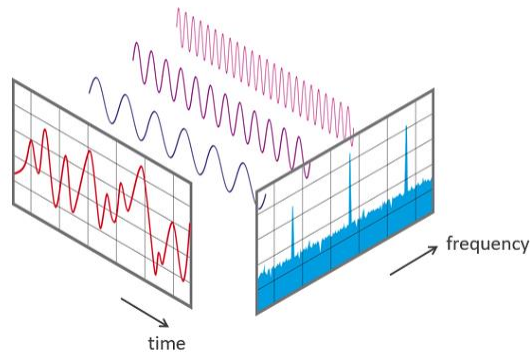


Figura 20. Señal en el dominio del tiempo al dominio de la frecuencia [36].

Se diseñaron dos filtros FIR con los siguientes parámetros:

Frecuencia de corte baja para el EEG: 0.5 Hz.

Frecuencia de corte alta para la señal EEG: 30 Hz.

Frecuencia de corte baja para el ECG: 0.5 Hz.

Frecuencia de corte alta para la señal ECG: 40 Hz.

Número de coeficientes del filtro: 101.

Tipo de ventana: *Hamming*.

En la Figura 21, se observan los filtros diseñados para eliminar frecuencias fuera del rango deseado.

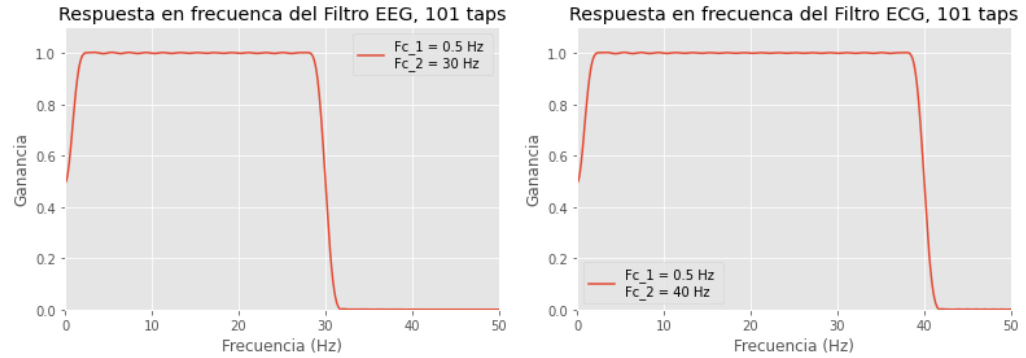


Figura 21. Respuesta en la frecuencia del Filtro FIR para eliminar ruido en las señales EEG y ECG.

Una vez creado el filtro se realizó la operación de la convolución con las señales del PSG (EEG y ECG) utilizando la función *signal.convolve* de la librería *scipy*.

III. Partición de datos en ventanas o épocas. Se dividieron los datos del arreglo de señales filtradas en ventanas de 20 segundos de longitud con solapamiento entre ventanas de 10 segundos. La función con la que se generó estas ventanas fue *sliding_window* de la librería de *yasa*, con los siguientes parámetros:

- i. *Data* = arreglo bidimensional de n canales por puntos en el tiempo.
- ii. *Sf* = Frecuencia de muestreo.
- iii. *Window* = 20
- iv. *step* = 10

IV. Codificación de series en el tiempo a imágenes. Este proceso se realizó primero creando el objeto *transformerGASF* por medio de la clase *GramianAngularField* de la librería *pyts.image* con los siguientes parámetros:

- i. *Image_size* = tamaño del arreglo de la serie de datos en el tiempo t .
- ii. *Method*=*summation*

Una vez que se creó el objeto se llamó al método *fit_transform* pasándole el arreglo bidimensional de la serie en el tiempo, lo que dio como resultado una imagen de 20 segundos para cada una de las señales, cuatro imágenes en total, posteriormente estas

fueron agrupadas en una sola imagen utilizando los métodos de la librería *matplotlib.pyplot* (ver Figura 22), las imágenes resultantes fueron guardadas en un directorio temporal.

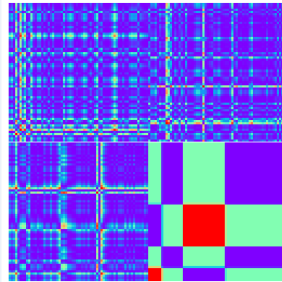


Figura 22. Imagen resultante de la transformación GASF, empezando por la parte superior de izquierda a derecha las señales EEG1 y EEG2; en la parte inferior de izquierda a derecha ECG y saturación de oxígeno.

- V. Compendio de imágenes. Se optó por guardar las imágenes resultantes en videos de 3 minutos cada uno, compuesto por 15 fotogramas con frecuencia de 1 fotograma por segundo, para ser utilizados posteriormente en la red neuronal, esto con el fin de reducir tamaño en memoria en un 50%. Para este paso se usó la librería de Open CV de la cual se extrajo el método *VideoWriter* para escribir los videos a partir de las imágenes generadas en el paso anterior, se le especificó el código de video *mp4v el cual dio a la salida videos de extensión .mp4 que se usaron para alimentar la red neuronal.

En la Figura 23 se presenta el esquema con los pasos realizados anteriormente, desde el PSG en dominio del tiempo hasta la creación del compendio de imágenes en videos.

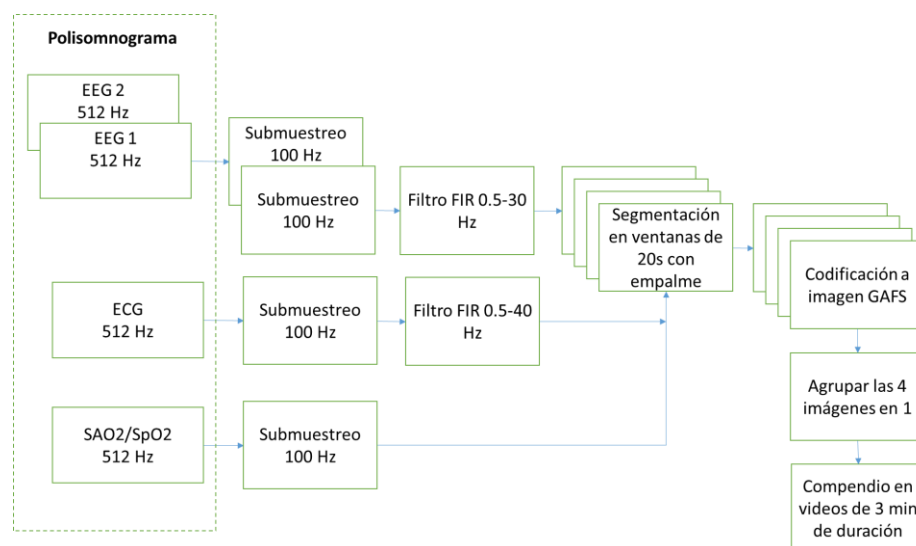


Figura 23. Esquema del preprocesamiento de las señales del PSG.

3.3.3. Diseño de la arquitectura

Ahora se describen los pasos realizados para obtener el modelo final de la red.

- I. Preparación de datos. Los datos fueron preparados antes de entrar en la red neuronal, por medio de la librería *ImageDataGenerator* y la clase *VideoFrameGenerator*, primero se realizó el aumento de datos con las siguientes características:

Rescale: 1. /255	Shear_range:0.2
Rotation_range:8	Zoom_range:0.2
Width_shift_range:0.2	Horizontal_flip: True
Height_shift_range:0.2	

Posteriormente se creó el objeto generador de fotogramas(*frames*) a partir del video con los siguientes parámetros:

Target_size: 224, 224
Color_mode: rgb
Batch_size=16
Shuffle: True
Nb_frames:10

- II. Arquitectura. Se diseñó una arquitectura basada en una red *VGG16*, con sus cuatro primeras capas convolucionales para la extracción de características de las imágenes, una red recurrente conformada por una capa distribuida en el tiempo y un bloque de atención, y para la clasificación se usaron cinco capas densas (la quinta con función de activación *softmax* para la clasificación de las cinco clases). Los hiper parámetros de la red como el número de filtros y tamaño de *kernel* se muestra en la Figura 24.

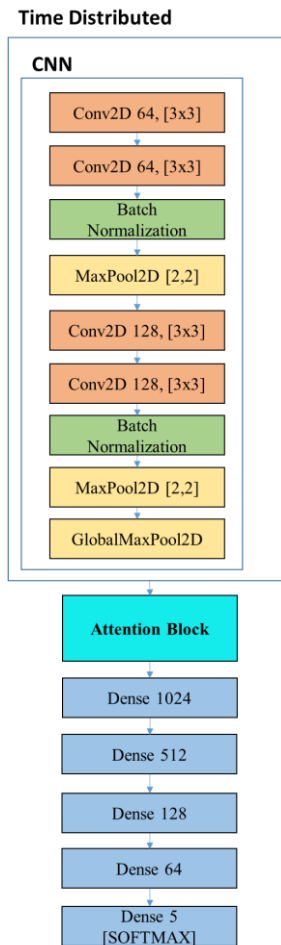


Figura 24. Red neuronal propuesta.

III. Compilación y Entrenamiento

La compilación y el entrenamiento de la red se realizó con los siguientes datos:

Optimizer: Adam

Learning rate: 1e-4

Loss: categorical_crossentropy

Épocas: 100

Capítulo 4

Resultados

En esta sección se presentan los resultados obtenidos con diferentes experimentos utilizando diferentes configuraciones de la red.

4.1.Experimentos

A continuación, se explican los experimentos realizados para llegar al modelo con mejor precisión.

4.1.1.Experimento 1

El experimento número uno utilizó diez participantes con las siguientes clases: *SDB*, *NFLE*, *NORMAL*, *PLM* y *RBD*; de cada uno se extrajeron cuatro horas del PSG con cuatro canales de este (EEG1, EEG2, SaO2/SpO2, ECG) lo que generó 14390 imágenes de 20 segundos con traslape de 10 segundos entre cada una, las cuales fueron almacenadas en 767 videos de 3 minutos por clase. La red fue alimentada con 80% del total de los videos para entrenamiento, 10 % para validación y 10% para pruebas, se utilizaron 100 épocas de entrenamiento. Los parámetros del experimento se pueden ver en la Tabla 1.

Tabla 1. Parámetros para el experimento número 1.

Número de compendios de imágenes en diez participantes.				
Clase	Entrenamiento	Validación	Prueba	Total
	80%	10%	10%	
NFLE	767	95	97	959
SDB	767	95	97	959
NORMAL	767	95	97	959
PLM	767	95	97	959
RBD	767	95	97	959
Total	3835	475	485	4795
Entrenamiento				
Tamaño imagen entrada	224x224			
Tamaño de lote de entrada	10 fotogramas			
Número de épocas	100			
Índice de aprendizaje	1e-4			
Optimizador	Adam			
Función de costo	<i>Categorical cross-entropy</i>			
CNN Layers	4			
RNN Layers/GRU	3			
Dense Layers	5			

El resultado en entrenamiento después de 100 épocas alcanza una exactitud de 0.97, mientras que en validación alcanza 0.89 de exactitud (Figura 25). La curva de la función de costo alcanza a decrecer hasta 0.07 en entrenamiento y hasta 0.23 en validación como se observa en la Figura 26. Los resultados no son satisfactorios para este experimento ya que la exactitud en validación aún es menor al objetivo 0.90.

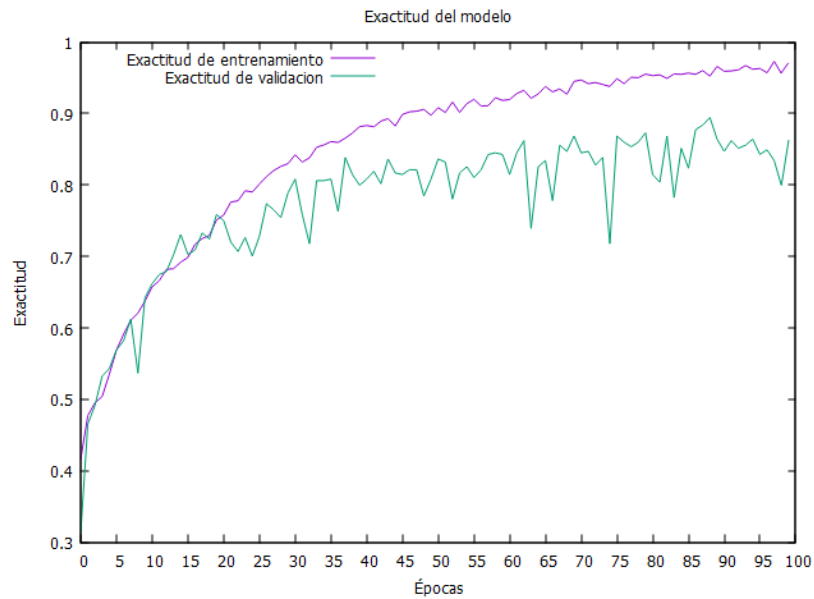


Figura 25 Curva de entrenamiento para el experimento número 1, exactitud en entrenamiento mayor a 0.95, exactitud en validación menor a 0.90.

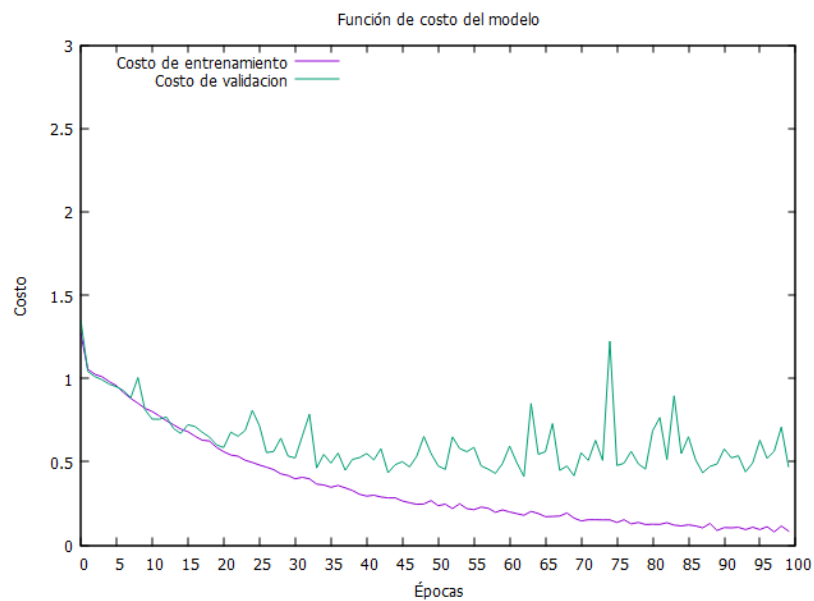


Figura 26 Curvas de la función de costo con el experimento número 1, en entrenamiento menor a 0.1, en validación 0.23 en su mejor desempeño.

Utilizando los datos del conjunto de prueba se obtuvieron las métricas de precisión (*Precision*), sensibilidad (*Recall*) y el valor F (*F1-Score*) las cuales se muestran en la Tabla 2, obteniendo un promedio de precisión de 0.90, siendo la mayor precisión en el trastorno *SDB* con 1.00 y la menor con la condición *PLM* con 0.82.

Tabla 2. Métricas obtenidas experimento 2, el promedio en precisión de 0.79.

Sleep disorder	Precision	Recall	F1-score	Support
nfle	0.92	0.80	0.86	97
normal	0.88	0.80	0.84	97
plm	0.82	0.91	0.86	97
rbd	0.88	0.97	0.92	97
sdb	1.00	1.00	1.00	97
Average	0.90	0.90	0.90	485

La matriz de confusión (Figura 27) muestra que clasifica mejor la clase *SDB* con 1.00 de efectividad y la clase que más confunde *NFLE* y *normal* ambas con 0.80.

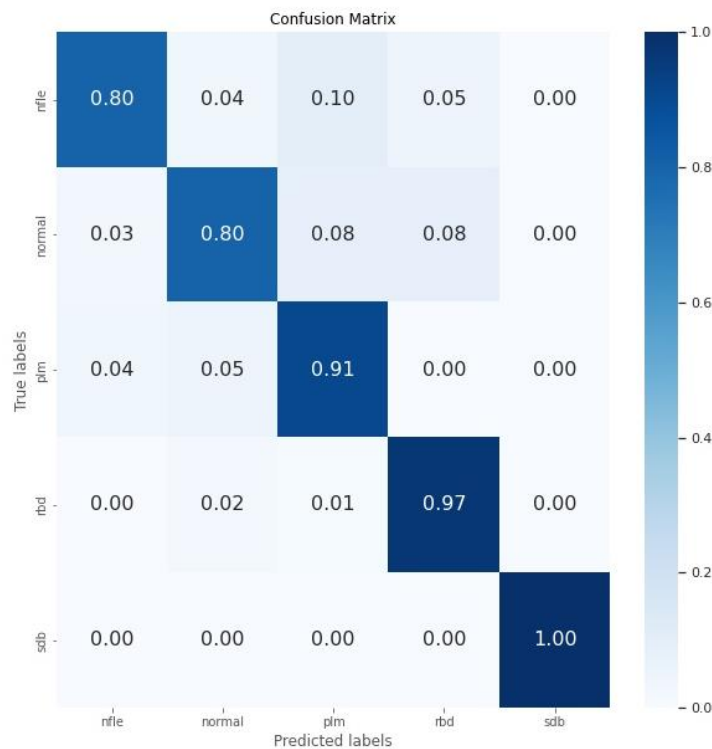


Figura 27 Matriz de confusión, la clase mejor clasificada es la *SDB* mientras que la clase con menor desempeño es la *NFLE* y *NORMAL*.

4.1.2. Experimento 2

El experimento dos utilizó igual que el anterior diez participantes al igual que las cinco clases; de cada uno se extrajeron cuatro horas del PSG con cuatro canales de este (EEG1, EEG2, SaO2/SpO2, ECG) lo que generó 14390 imágenes de 20 segundos con traslape de 10 segundos entre cada una, pero a diferencia del experimento anterior la red es modificada cambiando las capas GRU por un bloque de atención y además la resolución de entrada de las imágenes de 228x228 a 128x128. La red fue alimentada con 80% del total de los videos para entrenamiento, 10 % para validación y 10% para pruebas, se utilizaron 100 épocas de entrenamiento, además, las imágenes fueron escaladas a 128 x 128. Los parámetros del experimento se pueden ver en la Tabla 3.

Tabla 3. Parámetros para el experimento número 3.

Número de compendios de imágenes en diez participantes.				
Clase	Entrenamiento	Validación	Prueba	Total
	80%	10%	10%	
NFLE	767	95	97	959
SDB	767	95	97	959
NORMAL	767	95	97	959
PLM	767	95	97	959
RBD	767	95	97	959
Total	3835	475	485	4795
Entrenamiento				
Tamaño imagen entrada	128x128			
Tamaño de lote de entrada	10 fotogramas			
Número de épocas	100			
Índice de aprendizaje	1e-4			
Optimizador	Adam			
Función de costo	<i>Categorical cross-entropy</i>			
CNN Layers	4			
Attention block	1			
Dense Layers	5			

El resultado en entrenamiento después de 100 épocas alcanza una exactitud de 0.98, mientras que en validación alcanza 0.92 de exactitud (Figura 28). La curva de la función de costo alcanza a decrecer hasta 0.04 en entrenamiento y hasta 0.21 en validación como se observa en la Figura 29. Los resultados en las curvas de entrenamiento están por arriba del objetivo mayor a 0.90.

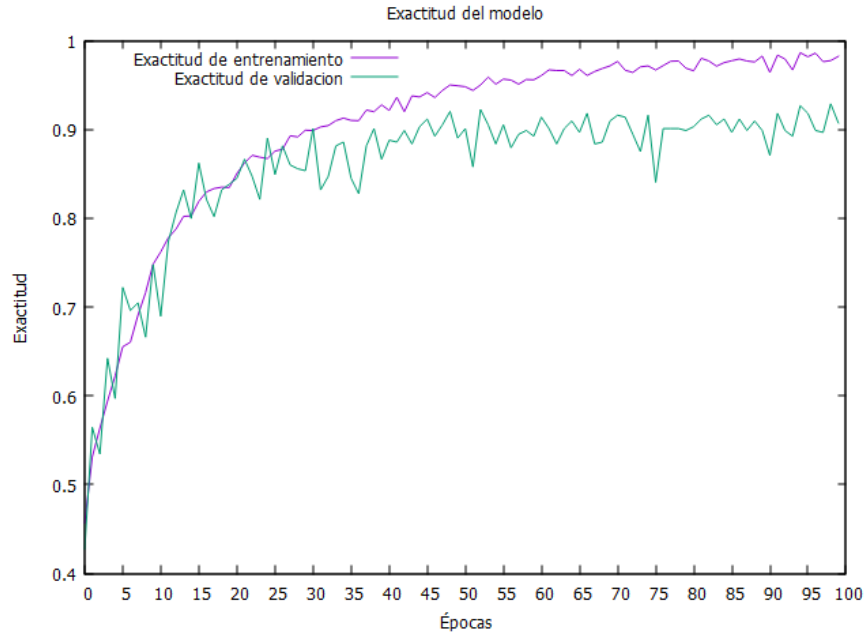


Figura 28 Curva de entrenamiento para el experimento número 2, exactitud en entrenamiento de 0.98, exactitud en validación de 0.92.

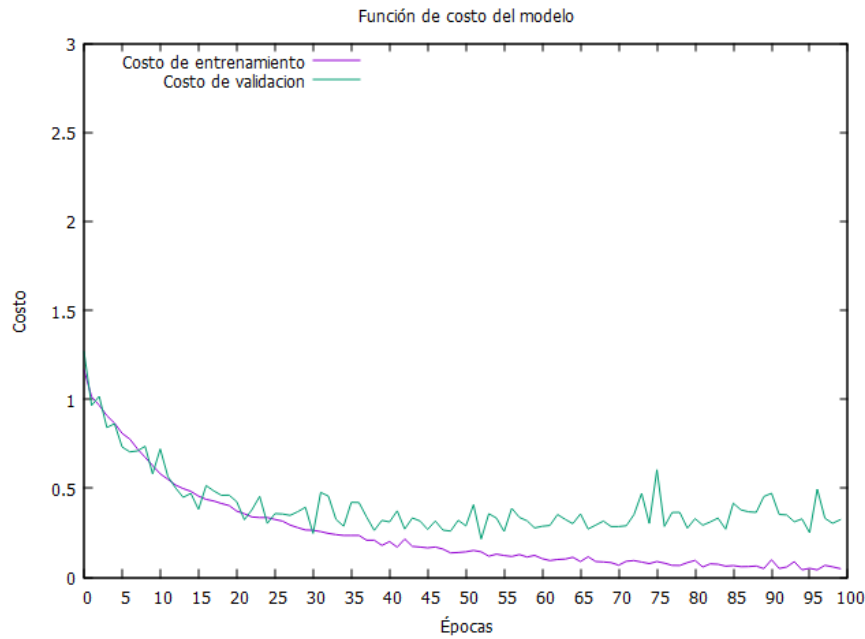


Figura 29 Curvas de la función de costo con 0.04 en entrenamiento y 0.25 en validación.

Utilizando los datos del conjunto de prueba se obtuvieron las métricas de precisión (*Precision*), sensibilidad (*Recall*) y el valor F (*F1-Score*) las cuales se muestran en la Tabla 4, obteniendo un promedio de precisión de 0.92, siendo la mayor precisión en el trastorno *SDB* con 1.00 y la menor con la condición *PLM* con 0.80.

Tabla 4. Métricas obtenidas experimento 2, el promedio en precisión de 0.91.

Sleep disorder	Precision	Recall	F1-score	Support
NFLE	0.93	0.90	0.91	97
NORMAL	0.98	0.82	0.89	97
PLM	0.80	0.97	0.88	97
RBD	0.92	0.90	0.91	97
SDB	1.00	1.00	1.00	97
Average	0.92	0.92	0.92	485

La matriz de confusión (Figura 30) muestra que clasifica mejor la clase *SDB* con 1.00 de efectividad, siendo la clase *NORMAL* la que más confunde con 0.82.

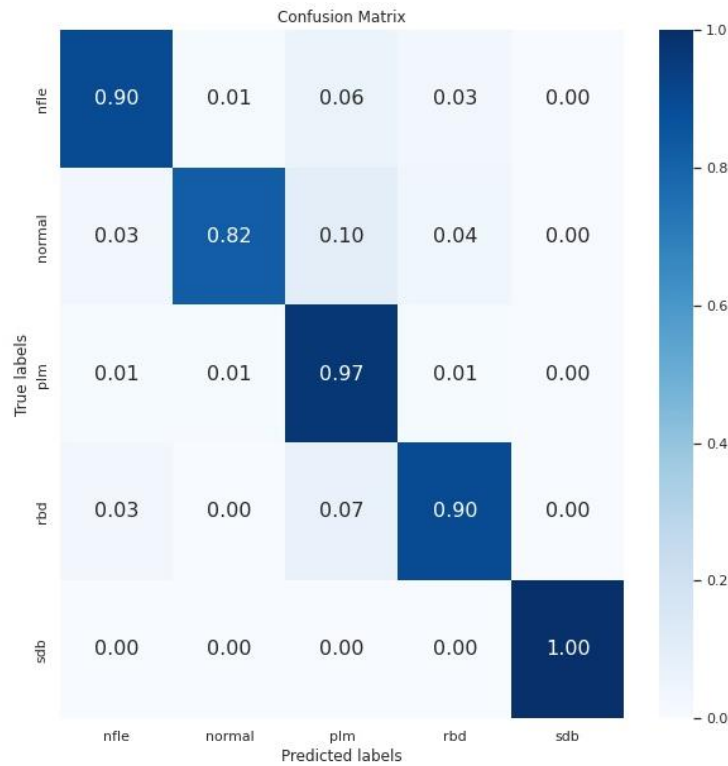


Figura 30 Matriz de confusión, la clase mejor clasificada es la *SDB* mientras que la clase con menor desempeño es la *NORMAL*.

4.1.3. Experimento 3

El experimento tres utilizó igual que el anterior diez participantes al igual que las cinco clases; de cada uno se extrajeron cuatro horas del PSG con cuatro canales de este (EEG1, EEG2, SaO2/SpO2, ECG) lo que generó 14390 imágenes de 20 segundos con traslape de 10 segundos entre cada una, la diferencia en este experimento es la entrada de las imágenes a la red, esta vez de 224 x 224. La red fue alimentada con 80% del total de los videos para entrenamiento, 10 % para validación y 10% para pruebas, se utilizaron 100 épocas de entrenamiento, además, las imágenes fueron escaladas a 224 x 224. Los parámetros del experimento se pueden ver en la Tabla 5.

Tabla 5. Parámetros para el experimento número 3.

Número de compendios de imágenes en diez participantes.				
Clase	Entrenamiento	Validación	Prueba	Total
	80%	10%	10%	
NFLE	767	95	97	959
SDB	767	95	97	959
NORMAL	767	95	97	959
PLM	767	95	97	959
RBD	767	95	97	959
Total	3835	475	485	4795
Entrenamiento				
Tamaño imagen entrada	224 x 224			
Tamaño de lote de entrada	10 fotogramas			
Número de épocas	100			
Índice de aprendizaje	1e-4			
Optimizador	Adam			
Función de costo	<i>Categorical cross-entropy</i>			
CNN Layers	4			
Attention block	1			
Dense Layers	5			

El resultado en entrenamiento después de 100 épocas alcanza una exactitud de 0.98, mientras que en validación alcanza 0.96 de exactitud (Figura 31). La curva de la función de costo alcanza a decrecer hasta 0.03 en entrenamiento y hasta 0.12 en validación como se observa en la Figura 32. Los resultados en las curvas de entrenamiento están por arriba del objetivo mayor a 0.90.

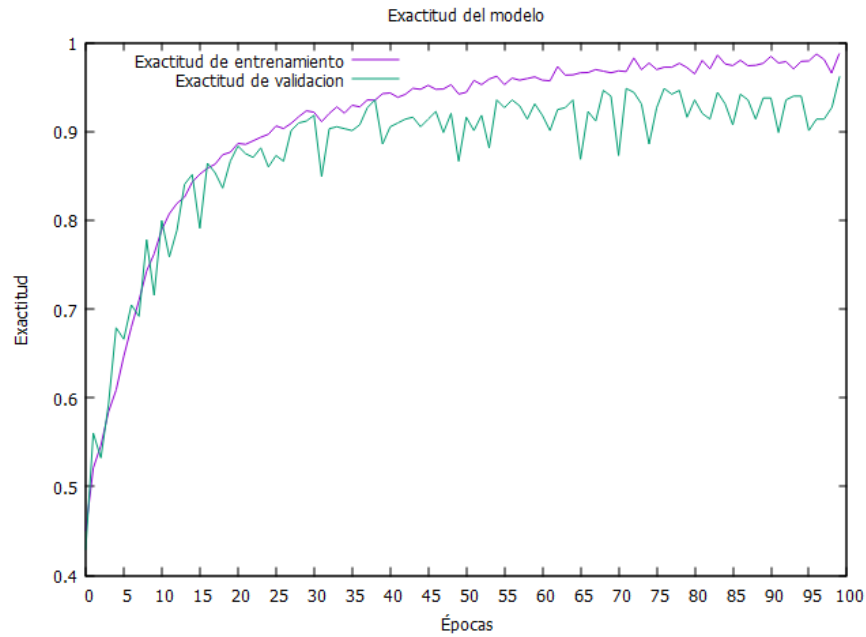


Figura 31 Curva de entrenamiento para el experimento número 3, exactitud en entrenamiento de 0.98, exactitud en validación de 0.96.

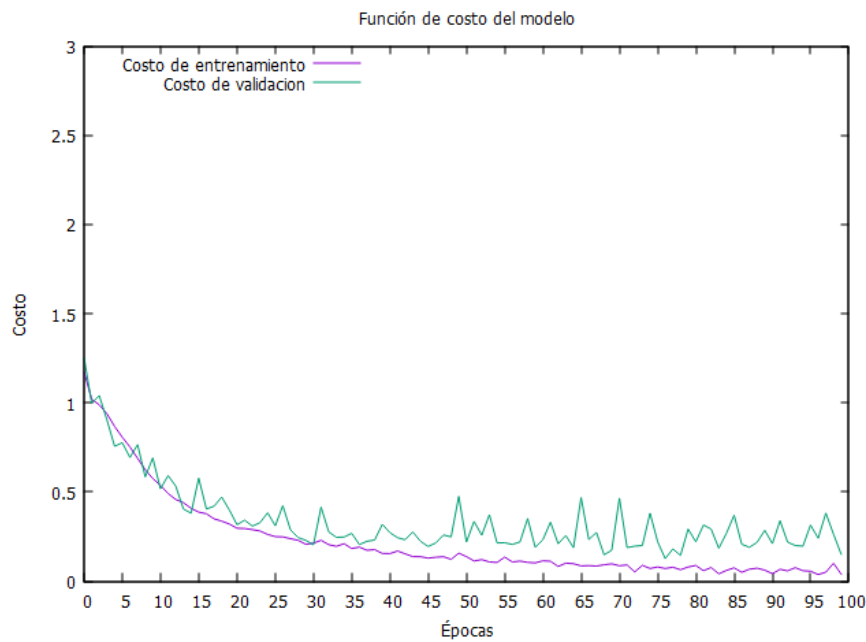


Figura 32 Curvas de la función de costo con 0.03 en entrenamiento y 0.12 en validación.

Utilizando los datos del conjunto de prueba se obtuvieron las métricas de precisión (*Precision*), sensibilidad (*Recall*) y el valor F (*F1-Score*) las cuales se muestran en la Tabla 6, obteniendo un promedio de precisión de 0.94, siendo la mayor precisión en el trastorno *SDB* con 0.99 y la menor con la condición *PLM* con 0.88.

Tabla 6. Métricas obtenidas experimento 3, el promedio en precisión de 0.91.

Sleep disorder	Precision	Recall	F1-score	Support
NFLE	0.95	0.89	0.91	97
NORMAL	0.97	0.93	0.95	97
PLM	0.88	0.94	0.91	97
RBD	0.93	0.95	0.94	97
SDB	0.99	1.00	0.99	97
Average	0.94	0.94	0.94	485

La matriz de confusión (Figura 33) muestra que clasifica mejor la clase *SDB* con 1.00 de efectividad, siendo la clase *NFLE* la que más confunde con 0.89.

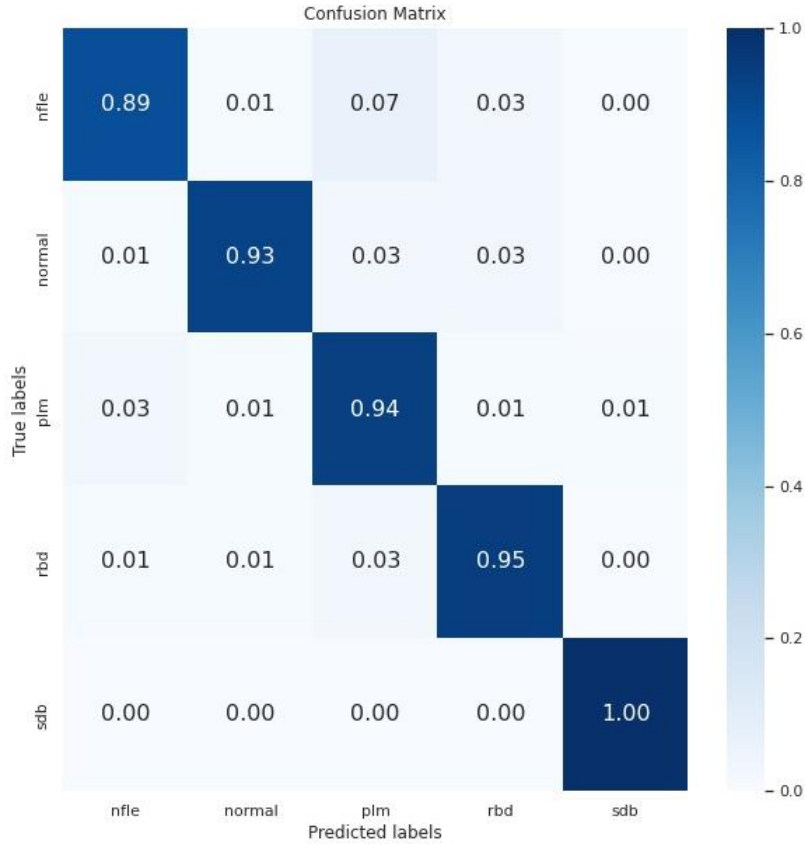


Figura 33 Matriz de confusión, la clase mejor clasificada es la *SDB* mientras que la clase con menor desempeño es la *NORMAL*.

4.1.4. Comparativa de los experimentos realizados

El resumen de los resultados obtenidos se muestra en la Tabla 7, donde claramente el experimento número 3 muestra mejor desempeño en las métricas.

Tabla 7. Comparativa entre los experimentos realizados.

Métricas	Experimento	Experimento	Experimento
	1	2	3
Precisión	0.90	0.92	0.94
Recall	0.90	0.92	0.94
F1-Score	0.90	0.92	0.94

Finalmente obtenemos las curvas AUROC de nuestro mejor modelo las cuales representamos en la Figura 34, la clase con mayor área es la *SDB* con 1.00 y en total tenemos un promedio para las 5 clases de 0.99 de área bajo la curva.

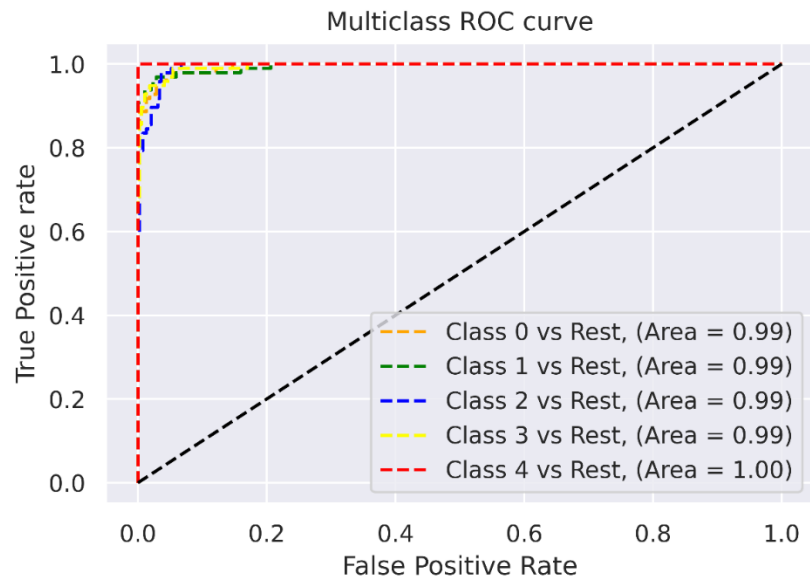


Figura 34. Curvas ROC para las cinco clases, en promedio 0.99 de área.

4.2. Discusión

En otros trabajos relacionados, Palani, Parvathavarthini, *et al.* (2020), obtienen una precisión de 0.80 con su modelo propuesto, el cual consta de una red convolucional entrenada con señales provenientes del EEG codificadas en imágenes, la cual fue desarrollada para clasificar el trastorno de NFLE; con nuestro modelo se obtiene una mayor precisión de 0.95 (para esta clase), como se puede observar en la Tabla 8.

Korkalainen, Aakko, *et al.*, en 2020 [37], usaron un modelo CNN/LSTM para clasificar SDB, aunque ellos solo clasifican este trastorno en diferentes niveles de severidad, su mejor resultado en entrenamiento obtiene 0.89 mientras que en validación 0.84 nuestro modelo propuesto alcanza arriba de 0.94 de precisión en promedio para cinco trastornos diferentes. La matriz de confusión del modelo de Korkalainen se generó con las distintas fases del sueño y los diferentes niveles de trastorno respiratorio (leve, moderado, severo) su mayor puntaje alcanza 0.92 mientras que nuestro modelo alcanza 1.00 para este trastorno, pero sin distinguir entre los diferentes niveles, la Tabla 8 muestra la comparativa entre estos modelos.

En 2021, Sharma, Tiwari, *et al* [38], desarrollaron un clasificador para identificar los trastornos, insomnio, narcolepsia, NFLE, PLM, RBD, SDB, usando dos canales EEG (C4-A1 y F4-C4). Al compararnos con su modelo tomando en cuenta únicamente las mismas clases (Normal, NFLE, PLM, RBD y SDB) obtuvieron un promedio en precisión de 0.89 mientras que nuestro modelo obtuvo 0.94. En lo que respecta a la matriz de confusión el modelo de los autores mencionados la clase que peor clasifica es la SDB con 0.52, nuestro modelo la clasifica con 1.00, seguido por la clase PLM con 0.74 mientras que el nuestro obtuvo 0.94 la clase RBD obtuvo 0.94 con nosotros fue de 0.95, la NFLE 0.93 contra 0.89 de nosotros y finalmente la clase normal 0.74 contra 0.93 de nuestro modelo propuesto; pues observamos que nuestro modelo es superior en 4 clases mientras que ellos en 1. La Tabla 8 muestra una comparativa entre ambas propuestas.

Tabla 8. Comparativa entre el estado del arte actual.

Estudio	Señal	Clasificador	Trastornos del sueño	Exactitud	Precisión	AUROC
Korkalainen, Aakko, et al. [37]	EEG1, EEG2, EOG	CNN(1D) /LSTM	SDB	0.84	NA	NA
Palani, Parvathavarthini, et al. [13]	EEG	<i>Custom</i> CNN	NFLE	NA	0.88	0.92
Sharma, Tiwari, et al. [38]	EEG	<i>Ensemble</i> <i>Bagged and</i> <i>Boosted</i> <i>trees</i>	Normal, Narcolepsia, PLM, SDB, RBD, NFLE	0.91	0.89	0.99
Modelo propuesto	EEG1, EEG2, ECG, SaO2/SpO2	CNN(2D) /Attention	Normal, NFLE, SDB, PLM, RBD	0.96	0.94	0.99

Conclusiones

El modelo propuesto compuesto por capas convolucionales y un bloque de atención logró clasificar los cinco trastornos de sueño propuestos para este trabajo con un promedio de 0.94 en precisión utilizando señales EEG1, EEG2, ECG, SaO2/SpO2 del PSG las cuales son convertidas a imágenes y almacenadas en compendios *.mp4* de tres minutos de duración.

La metodología propuesta basada en la codificación de señales en el tiempo en imágenes es un método novedoso que basándose en el principio que las redes neuronales han avanzado más en clasificación de imágenes, busca competir con modelos basados en señales unidimensionales en el tiempo o capas de una dimensión.

La red propuesta está diseñada con capas convolucionales para la extracción de características de las imágenes, capas recurrentes y un bloque de atención para guardar la información en función del tiempo.

El trabajo propuesto queda abierto para mejorar en precisión utilizando un mayor número de participantes de entrenamiento, así como también con arquitecturas más modernas. También un trabajo a futuro puede ser su implementación en una interfaz de usuario llámese computador o teléfono móvil inteligente.

Bibliografía

- [1] Cleveland Clinic. "Common Sleep Disorders." [my.clevelandclinic.org. https://my.clevelandclinic.org/health/articles/11429-common-sleep-disorders#:~:text=Common%20sleep%20disorders%20like%20insomnia,of%20diabetes%20and%20heart%20disease](https://my.clevelandclinic.org/health/articles/11429-common-sleep-disorders#:~:text=Common%20sleep%20disorders%20like%20insomnia,of%20diabetes%20and%20heart%20disease) (accessed Nov. 1, 2020).
- [2] C. Armon, MD, MSc, MHS. "Polysomnography." [emedicine.medscape.com. https://emedicine.medscape.com/article/1188764-overview](https://emedicine.medscape.com/article/1188764-overview) (accessed: Nov. 1, 2020).
- [3] A. Shrestha and A. Mahmood, "Review of Deep Learning Algorithms and Architectures," in *IEEE Access*, vol. 7, pp. 53040-53065, 2019, doi: 10.1109/ACCESS.2019.2912200.
- [4] M.G. Terzano, L. Parrino, A. Sherieri, R. Chervin, S. Chokroverty, C. Guilleminault, M. Hirshkowitz, M. Mahowald, H. Moldofsky, A. Rosa, R. Thomas, A. Walters. "Atlas, rules, and recording techniques for the scoring of cyclic alternating pattern (CAP) in human sleep," *Sleep Med.* 2001 Nov;2(6):537-53. doi: 10.1016/s1389-9457(01)00149-6. Erratum in: *Sleep Med.* 2002 Mar;3(2):185. PMID: 14592270.
- [5] A. Goldberger, L. Amaral, L. Glass, J. Hausdorff, P. C. Ivanov, R. Mark, and H.E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *2000 Circulation [Online]*. 101 (23), pp. e215–e220.
- [6] G. Q. Zhang, L. Cui, R. Mueller, S. Tao, M. Kim, M. Rueschman, S. Mariani, D. Mobley, S. Redline, "The National Sleep Research Resource: towards a sleep data commons," *J Am Med Inform Assoc.* 2018 Oct 1;25(10):1351-1358. doi: 10.1093/jamia/ocy064. PMID: 29860441; PMCID: PMC6188513.
- [7] T. Young, M. Palta, J. Dempsey, P. E. Peppard, F. J. Nieto, K. M. Hla, "Burden of sleep apnea: rationale, design, and major findings of the Wisconsin Sleep Cohort study," *WMJ.* 2009 Aug;108(5):246-9. PMID: 19743755; PMCID: PMC2858234.
- [8] D. Mellado, C. Saavedra and R. Salas. (2016). "Aplicación de Deep Learning para la detección de Apnea e Hipoapnea del Sueño basado en Señales de Polisomnografía,".
- [9] T. T. Chowdhury, A. Hossain, S. A. Fattah and C. Shahnaz, "Seizure and Non-Seizure EEG Signals Detection Using 1-D Convolutional Neural Network Architecture of Deep Learning Algorithm," 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), 2019, pp. 1-4, doi: 10.1109/ICASERT.2019.8934564.
- [10] S. Wickramaratne, and M. D. Mahmud, "Automatic Detection of Respiratory Effort Related Arousals with Deep Neural Networks from Polysomnographic Recordings," 2020. 154-157. 10.1109/EMBC44109.2020.9176413.
- [11] F. Pisano, G. Sias, A. Fanni, B. Cannas, A. Dourado, B. Pisano, and C. Teixeira, "Convolutional Neural Network for Seizure Detection of Nocturnal Frontal Lobe Epilepsy," *Complexity*, 2020. 1-10. 10.1155/2020/4825767.
- [12] F. Vaquerizo, D. Alvarez, L. Gozal, G. Gutiérrez, V. Barroso, E. Santamaría, F. del Campo, D. Gozal, R. Hornero, "Modelo de deep learning basado en la arquitectura Inception para el diagnóstico de la apnea del sueño infantil mediante la señal de oximetría," 2020.
- [13] K. Palani Thanaraj, B. Parvathavarthini, Urcun John Tanik, V. Rajinikanth, Seifedine Nimer Kadry, Krishnamurthy Kamalanand: Implementation of Deep Neural Networks to Classify

- EEG Signals using Gramian Angular Summation Field for Epilepsy Diagnosis. CoRR abs/2003.04534 (2020).
- [14] A. Giorgi and A. Biggers, M.D., MPH. “Polysomnography.” healthline.com. <https://www.healthline.com/health/polysomnography> (accessed dic 25, 2020).
- [15] J. Satheesh Kumar, P. Bhuvaneshwari, “Analysis of Electroencephalography (EEG) Signals and Its Categorization—A Study,” *Procedia Engineering*. Volume 38, 2012. Pages 2525-2536.
- [16] Rajni, and K. Inderbir. “Electrocardiogram Signal Analysis - An Overview,” 2013 *International Journal of Computer Applications*. 84. 22-25. 10.5120/14590-2826.
- [17] Lakna. “What is the difference between SAO2 and SPO2,” pediaa.com, <https://pediiaa.com/what-is-the-difference-between-sao2-and-spo2/> (accessed Jan. 11, 2021).
- [18] P. Kingman, MD, msdmanuals.com, <https://www.msdmanuals.com/es/professional/trastornos-pulmonares/apnea-del-sue%C3%B1o/apnea-obstruictiva-del-sue%C3%B1o?query=apnea>(accessed Jan. 11, 2021)
- [19] J. Garcia and N. Williams, “Sleep Terrors or Nocturnal Frontal Lobe Epilepsy?” *A pediatric perspective*, vol. 20, 2011.
- [20] “Rapid eye movement sleep behaviour disorder.” uptodate.com. [https://www.uptodate.com/contents/rapid-eye-movement-sleep-behavior-disorder/print#:~:text=INTRODUCTION%20Rapid%20eye%20movement%20\(REM,thras hing%2C%20punching%2C%20and%20kicking](https://www.uptodate.com/contents/rapid-eye-movement-sleep-behavior-disorder/print#:~:text=INTRODUCTION%20Rapid%20eye%20movement%20(REM,thras hing%2C%20punching%2C%20and%20kicking) (accessed May. 1, 2022).
- [21] F. Logan. “Periodic Limb Movements Disorder.” Sleepfoundation.org. <https://www.sleepfoundation.org/periodic-limb-movement-disorder>(accessed May 1, 2022).
- [22] Y. LeCun, Y. Bengio, and G. Hinton, “Deep Learning,” 2015 *Nature*. 521. 436-44. 10.1038/nature14539.
- [23] N. Buduma, and N. Locascio, “Fundamentals of Deep learning,” O'Reilly Media, 2017. [Online]. Available: books.google.com.
- [24] A. Ghosh, A. Sufian, F. Sultana, A. Chakrabarti, and D. De, “Fundamental Concepts of Convolutional Neural Network,” 2020, 10.1007/978-3-030-32644-9_36.
- [25] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, “Dropout: A Simple Way to Prevent Neural Networks from Overfitting,” *Journal of Machine Learning Research*, vol. 15, pp. 1929-1958, Jun. 2014.
- [26] Université de Toulouse “Neural Networks and Introduction to Deep Learning” [Online]. Available: <https://www.math.univ-toulouse.fr/~besse/Wikistat/pdf/st-m-hdstat-rnn-deep-learning.pdf>.
- [27] A. Vilagran, “Facial Expression Detection using Convolutional Neural Networks,” 2018, Engineering, Universidad Politécnic de Catalunya.
- [28] S. Cristina. “The Attention Mechanism from Scratch” machinelearningmastery.com <https://machinelearningmastery.com/the-attention-mechanism-from-scratch/> (accessed May 10, 2022).
- [29] A. Krizhevsky, I. Sutskever, and G. E. Hinton. “Imagenet classification with deep convolutional neural networks,” In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097-1105. Curran Associates, Inc., 2012.

- [30] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image Recognition," CoRR, abs/1409.1556, 2014.
- [31] P. Varshney. "VGGNet-16 Architecture: A Complete Guide" kaggle.com. <https://www.kaggle.com/blurredmachine/vggnet-16-architecture-a-complete-guide> (accessed April 10, 2022).
- [32] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015.
- [33] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, "Dive into deep learning," 2021, *arXiv preprint arXiv:2106.11342*.
- [34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016.
- [35] Z. Wang and T. Oates, "Encoding Time Series as Images for Visual Inspection and Classification Using Tiled Convolutional Neural Networks," 2015.
- [36] File: [FFT-Time-Frequency-View.png](https://commons.wikimedia.org/wiki/File:FFT-Time-Frequency-View.png). [commons.wikimedia.org](https://commons.wikimedia.org/wiki/File:FFT-Time-Frequency-View.png). <https://commons.wikimedia.org/wiki/File:FFT-Time-Frequency-View.png> (accessed May 1, 2022).
- [37] H. Korkalainen et al., "Accurate Deep Learning-Based Sleep Staging in a Clinical Population With Suspected Obstructive Sleep Apnea," in *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 7, pp. 2073-2081, July 2020, doi: 10.1109/JBHI.2019.2951346.
- [38] M. Sharma, J. Tiwari, V. Patel, U. R. Acharya, "Automated Identification of Sleep Disorder Types Using Triplet Half-Band Filter and Ensemble Machine Learning Techniques with EEG Signals. *Electronics*," 2021; 10(13):1531. <https://doi.org/10.3390/electronics10131531>.