



INSTITUTO TECNOLÓGICO DE CIUDAD MADERO

DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

MAESTRÍA EN INGENIERÍA ELÉCTRICA



"POR MI PATRIA Y POR MI BIEN"

TESIS

**IMPLEMENTACIÓN DE REDES NEURONALES CONVOLUCIONALES Y RECURRENTE PARA
MEJORAR EL RENDIMIENTO DE UN SISTEMA DE ENERGÍA SOLAR FOTOVOLTAICA**

Que para obtener el grado de:
Maestro en Ingeniería Eléctrica

Presenta:
Ing. Miguel Alejandro Govea Armengol
G23073004
CVU 1315920

Director:
Dr. José Genaro González Hernández
CVU 321142

Codirector:
Dr. Marco Antonio Coronel García
CVU 269568



Instituto Tecnológico de Ciudad Madero
Subdirección Académica
División de Estudios de Posgrado e Investigación

Ciudad Madero, Tamaulipas, **13/Marzo/2026**

Oficio No.: U2.016/26

Asunto: Autorización de impresión de tesis

C. MIGUEL ALEJANDRO GOVEA ARMENGOL
No. DE CONTROL G23073004
P R E S E N T E

Me es grato comunicarle que después de la revisión realizada por el Jurado designado para su Examen de Grado de Maestría en Ingeniería Eléctrica, se acordó autorizar la impresión de su tesis titulada:

"IMPLEMENTACIÓN DE REDES NEURONALES CONVOLUCIONALES Y RECURRENTES PARA MEJORAR EL RENDIMIENTO DE UN SISTEMA DE ENERGÍA SOLAR FOTOVOLTAICA"

El Jurado está integrado por los siguientes catedráticos:

PRESIDENTE:	DR. JOSÉ GENARO GONZÁLEZ HERNÁNDEZ
SECRETARIO:	DR. MARCO ANTONIO CORONEL GARCÍA
VOCAL:	DR. LUIS FORTINO CISNEROS SINENCIO
SUPLENTE:	M.C. RAFAEL CASTILLO GUTIÉRREZ
DIRECTOR DE TESIS:	DR. JOSÉ GENARO GONZÁLEZ HERNÁNDEZ
CO-DIRECTOR:	DR. MARCO ANTONIO CORONEL GARCÍA

Es muy satisfactorio para la División de Estudios de Posgrado e Investigación compartir con usted el logro de esta meta. Espero que continúe con éxito su desarrollo profesional y dedique su experiencia e inteligencia en beneficio de México.

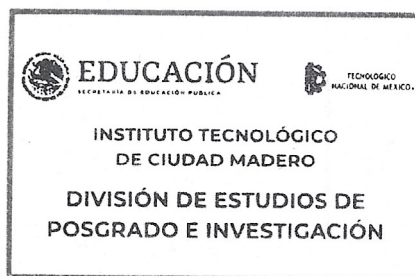
ATENTAMENTE

Excelencia en Educación Tecnológica®

"Por mi patria y por mi bien"®

JOSE AARON MELO BANDA
ENCARGADO DE LA DIVISIÓN DE
ESTUDIOS DE POSGRADO E INVESTIGACIÓN

ccp. Archivo
JAMB/JRC/aecr



2026
año de
Margarita
Maza



Dedicatoria

Primeramente a mi madre, Julieta Adriana Armengol Sánchez, aquella persona que con su amor y enseñanza desde pequeño me ha llevado hasta este punto de mi vida. A mi abuela, Teresa de Jesús Sánchez López q. e. p. d., que siempre me ofreció su sabiduría y consejos, y siempre me levantó mis ánimos cuando me encontraba en momentos difíciles. A mis compañeros de maestría, que con un mutuo apoyo hemos llegado hasta el final de la misma con un proyecto exitoso en todos. A mi pareja, José Ulises Ramírez González, quien en toda mi trayectoria de maestría me brindó todo su apoyo para que todo esto fuese posible. Y a mi director de tesis, el Dr. José Genaro González Hernández, quien desde la universidad y hasta la maestría tuvo fe en mis conocimientos y me brindó su apoyo en este proyecto.

Agradecimientos

Primero que nada, quiero agradecer a mi madre, Julieta Adriana Armengol Sánchez, y a mi abuela, Teresa de Jesús Sánchez López q. e. p. d., quienes fueron las peronas que me criaron hasta convertirme en lo que soy ahora, y me han proporcionado siempre su amor incondicional.

A todos los maestros que he tenido a lo largo de mi carrera, quienes han sido la base del conocimiento que actualmente tengo, y de quienes he podido aplicar el mismo en este proyecto y la maestría.

A mi director de proyecto, el Dr. José Genaro González Hernández, que gracias a su inspiración, muestra de perseverancia, y la orientación que me dió, puedo completar con éxito esta siguiente etapa de mi vida.

A los Maestros Rafael Castillo Gutiérrez y Jesús Rodolfo castillo Ibarra, por su apoyo en la entrega de papeles y el proporcionar sus experiencias a la hora de solicitar la beca CONAHCYT.

Al propio CONAHCYT (ahora SECIHTI) por brindarme el apoyo económico para llevar a cabo este trabajo.

A todos mis compañeros de maestría ya que sin la ayuda de todos no hubieramos podido llegar todos hasta esta etapa.

Y finalmente a todo el Tecnológico de Ciudad Madero por proporcionarme una casa de estudios, por aceptarme en su momento, y de todo su personal por haber sido tan amables conmigo y ayudarme en lo solicitado.

Muchas gracias a todos por hacer esto posible!

Resumen

Este trabajo de tesis propone la realización de una solución basada en inteligencia artificial para optimizar la captación de energía de los paneles solares a partir de un análisis de imágenes del cielo y una estimación de la irradiancia. Para ello, este trabajo se beneficia de un conjunto de datos de más de 100,000 imágenes del cielo de la Universidad de Stanford, de marzo a diciembre del año 2017. Para este conjunto de datos de imágenes del cielo, se utiliza Python para extraer un conjunto de variables radiométricas, así como variables de la atmósfera del cielo, como la irradiancia media, la irradiancia ponderada, la cobertura de nubes, la varianza óptica, la ubicación del centro del sol y la uniformidad del brillo de la atmósfera, utilizando modelos basados en el cielo para aislar los píxeles que contienen información menos significativa para extraer con precisión. Valores de irradiancia en la atmósfera del cielo.

Luego se diseñó una Red Neuronal Convolutiva (CNN, por sus siglas en inglés) utilizando TensorFlow y Keras para identificar relaciones complejas entre características extraídas de datos y valores medidos de irradiancia solar. El modelo CNN diseñado es capaz de aprender diferentes características relacionadas con la irradiancia solar, la dispersión atmosférica y el movimiento de las nubes. El modelo diseñado es capaz de predecir con precisión valores de irradiancia solar en tiempo real, así como detectar con precisión condiciones favorables para cambiar las orientaciones de los paneles solares en términos de acimut y posiciones de inclinación.

Los resultados de este estudio demuestran que los enfoques de visión por computadora, aprendizaje automático y aprendizaje profundo crean una alternativa precisa basada en datos para los enfoques de modelado de irradiancia. Esta solución llena un vacío significativo entre los enfoques de imágenes meteorológicas y los sistemas de

optimización solar, lo que permite controlar dinámicamente los sistemas fotovoltaicos utilizando condiciones visuales, ambientales y de otro tipo. Finalmente, este trabajo ayuda a promover la investigación sobre sistemas avanzados de gestión de energía solar que utilizan inteligencia artificial para aumentar la eficiencia, la sostenibilidad y la eficacia de la energía solar.

Abstract

This thesis proposes an AI-based solution to optimize solar panel energy capture by analyzing sky images and estimating irradiance. This work utilizes a dataset of over 100,000 sky images from Stanford University, collected between March and December 2017. Python was used to extract a set of radiometric variables from this sky image dataset, as well as atmospheric variables such as mean irradiance, weighted irradiance, cloud cover, optical variance, the location of the sun's center, and atmospheric brightness uniformity. Sky-based models were used to isolate pixels containing less significant information, allowing for the accurate extraction of irradiance values from the sky atmosphere.

A convolutional neural network (CNN) was then designed using TensorFlow and Keras to identify complex relationships between extracted data features and measured solar irradiance values. The designed CNN model is capable of learning various characteristics related to solar irradiance, atmospheric scattering, and cloud movement. It can accurately predict real-time solar irradiance values and precisely detect conditions favorable for adjusting solar panel orientations in terms of azimuth and tilt positions.

The results of this study demonstrate that computer vision, machine learning, and deep learning approaches offer a precise, data-driven alternative to traditional irradiance modeling methods. This solution fills a significant gap between meteorological imaging approaches and solar optimization systems, enabling the dynamic control of photovoltaic systems using visual, environmental, and other factors. Finally, this work contributes to advancing research on advanced solar energy management systems that utilize artificial intelligence to enhance the efficiency, sustainability, and effectiveness of solar power.

Índice

Dedicatoria	I
Agradecimientos	II
Resumen	III
Abstract	V
Índice	VI
Índice de Figuras	XI
Índice de Tablas	XII
Índice de Acrónimos	XIII
Índice de Códigos	XIV
1. Capítulo I - Introducción	1
1.1. Antecedentes	1
1.1.1. Revisión del Laboratorio Nacional de Energías Renovables: Avances en la Predicción Solar en 2023	2
1.1.2. Predicción Mediante Imágenes del Cielo y Métodos Basados en Características	3
1.1.3. Estudios de Pronóstico Híbridos	3
1.1.4. Aprendizaje Profundo a Partir de Cámaras Celestes: Modelos Aplicados y Evaluaciones	4

1.1.5.	Pronóstico a Nivel de Minutos o Multipasos Mediante Modelos Temporales Avanzados	4
1.1.6.	Análisis Comparativo y Reproducibilidad	5
1.1.7.	Aprendizaje por Refuerzo, Control de la Inteligencia Artificial para Seguimiento	5
1.1.8.	Aprendizaje por Refuerzo y Orientación de Helióstatos	6
1.1.9.	Sistemas Aplicados Recientes y Estudios de Caso	6
1.2.	Planteamiento del Problema	7
1.2.1.	Intermitencia y Variabilidad	7
1.2.2.	Eficiencia y Avances Tecnológicos	7
1.2.3.	Dependencias Geográficas y Climáticas	8
1.3.	Justificación	8
1.3.1.	Optimización Basada en Datos	8
1.3.2.	Optimización Multivariante	9
1.3.3.	Reconocimiento y Generalización de Patrones	9
1.3.4.	Estimación Energética	9
1.4.	Objetivos	10
1.4.1.	Objetivo General	10
1.4.2.	Objetivos Específicos	10
1.5.	Hipótesis	10
1.6.	Alcances y Limitaciones	10
1.6.1.	Alcances	10
1.6.2.	Limitaciones	11
2.	Capítulo II - Marco Teórico	12
2.1.	Introducción	12
2.2.	Geometría Solar y Captura de Energía	12
2.3.	Redes Neuronales en la Extracción de Energía Solar	14

2.4.	Efectos Atmosféricos y Modelado de Irradiancia	16
2.4.1.	Desglose de la Irradiancia Solar	16
2.4.1.1.	Atenuación Atmosférica y Profundidad Óptica	17
2.4.2.	Influencia de las Nubes y la Dispersión	18
2.4.3.	Modelado Atmosférico Basado en Imágenes	19
2.4.4.	Integración en el Marco de la Red Neuronal	19
2.5.	Inclinación y Azimut	20
2.6.	Red Neuronal Convolutiva	22
2.6.1.	Capas Convolutivas	22
2.6.2.	Funciones de Activación	23
2.6.3.	Capas de Agrupación	23
2.6.4.	Capas Totalmente Conectadas	24
2.6.5.	Capa de Salida	24
2.7.	Red Neuronal Recurrente	25
2.7.1.	Desglosando Paso a Paso el Funcionamiento de una RNN	26
2.7.1.1.	Estructura y Flujo de Información	26
2.7.1.2.	Memoria y Dependencias Temporales	27
2.7.1.3.	Entrenamiento con Retropropagación a Través del Tiempo	27
2.7.1.4.	Variantes: Memoria Larga a Corto Plazo y Unidades Recurrentes Cerradas	27
2.7.2.	Aplicaciones en la Optimización de la Energía Solar	28
3.	Capítulo III - Metodología	29
3.1.	Usando Redes Neuronales en una Base de Datos de 100,000 Imágenes	29
3.1.1.	Composición y Etiquetado del Conjunto de Datos	29
3.1.2.	Preprocesamiento y Preparación de Datos	30
3.1.3.	Extracción de Características Espaciales Mediante CNN	31

3.1.4.	Modelado Temporal con RNN o LSTM	32
3.1.5.	Entrenamiento y Validación del Modelo	32
3.1.6.	Implementación y Aplicaciones en Tiempo Real	33
3.1.7.	Ventajas de Usar una Base de Datos de 100,000 Imágenes	34
3.2.	Dataset a Utilizar	34
3.3.	Extracción de los Datos de Imagen para la Estimación de Irradiancia Solar	36
3.3.1.	Extracción de Datos Temporales	36
3.3.2.	Extracción de Datos Radiométricos a Partir de la Intensidad de los Píxeles	37
3.3.3.	Localización del Sol y Análisis de Distancia	37
3.3.4.	Selección de la Región de Interés y Rechazo de Píxeles Irrelevantes	38
3.3.5.	Estimación de la Condición Atmosférica	39
3.3.6.	Integración y Estructuración de Datos	40
3.4.	Geometría Solar	41
3.4.1.	Ecuación General de la Irradiancia Solar	42
3.4.2.	Corrección de Declinación Solar	46
3.4.3.	Computando las Ecuaciones y su Muestreo en Gráficas	47
3.4.4.	Ecuación General de la Irradiancia Solar	50
3.4.5.	Computando la Irradiancia Solar	53
3.5.	Determinando las Condiciones Atmosféricas Usando Inteligencia Artificial	55
3.6.	Ubicación Ideal de Granjas de Paneles Solares	56
4.	Capítulo IV - Extracción de Datos Previos al Entrenamiento	59
4.1.	Extracción de Datos de las Imágenes Usando Python	59
4.1.1.	Preprocesamiento de Imágenes	59
4.1.2.	Estimación de la Irradiancia Solar	60
4.1.3.	Intensidad Ponderada	61
4.1.4.	Coordenadas del Centro Solar	61

4.1.5.	Estimación de la Cobertura de Nubes	62
4.1.6.	Varianza Óptica	62
4.1.7.	Índice de Uniformidad del Brillo Atmosférico	63
4.1.8.	Extracción de Metadatos Temporales	64
4.1.9.	Estructuración y Almacenamiento de Datos	67
4.2.	Estimación de la Posición Solar	68
4.2.1.	Posición del Sol como un Conjunto de Vectores Unitarios 3D . .	69
4.2.2.	Interpretación de las Componentes del Vector Unitario	70
4.2.3.	Corrección de la Proyección Solar	71
4.2.4.	Proyección Sobre un Plano 2D	73
4.2.5.	Eliminación de Datos Innecesarios	73
4.3.	Integración de los Resultados de la Ecuación al Dataset	74
5.	Capítulo V - Elaboración de las Redes Neuronales	78
5.1.	Desglose de la Red Neuronal Convolutacional	78
5.2.	Desglose de la Red Neuronal Recurrente	86
6.	Capítulo VI - Análisis y Resultados	90
6.1.	Resultados Obtenidos de Ambas Redes	90
6.2.	Trabajos a Futuro con Estos Resultados	95
7.	Capítulo VII - Conclusiones	98

Índice de Figuras

3.1. Imagen a utilizar en la base de datos de la red, todas las imágenes com- parten esta apariencia	35
3.2. Elevación solar durante todo el día, en todo el año	49
3.3. Patrón de elevación en un día	49
3.4. Puntos de elevación máxima en el día, durante todo el año	50
3.5. Presencia de irradiancia en cada minuto del día durante todo el año . .	54
4.1. Analemas solares en un punto, por intervalos de 1 hora los 365 días del año	72
4.2. Analemas solares vistas desde un punto elevado, con la limpieza de datos carentes de luz solar	72
5.1. Métricas de error de la CNN	85
5.2. Métricas de error de la RNN	89
6.1. Gráfica de la irradiancia, comparando datos reales y datos generados .	91
6.2. Gráfica de coordenadas solares, comparando datos reales y datos generados	92
6.3. Gráfica de indicador atmosférico, comparando datos reales y datos ge- nerados	94
6.4. Gráfica de irradiancia mediante la ecuación, comparando datos reales, datos generados y datos de irradiancia de tanto la red como los datos reales	94

Índice de Tablas

3.1. Tabla de parámetros a utilizar en las redes neuronales	40
---	----

Índice de Acrónimos

BPTT Retropropagación a Través del Tiempo. 27

CNN Red Neuronal Convolutacional. 3–6, 14, 15, 20, 22–25, 28, 29, 31–33, 85, 87, 90, III, V, VIII, XI

CSP Energía Solar Concentrada. 6, 57

GRU Unidades Recurrentes Cerradas. 28, 32

IA Inteligencia Artificial. 12, 21, 22, 28, 41, 45, 47, 55, 56

LSTM Memoria Larga a Corto Plazo. 3–6, 28, 32, 86, 88, 95, IX, XIV

MLP Perceptrón Multicapa. 3

RL Aprendizaje por Refuerzo. 5, 6

RNN Red Neuronal Recurrente. 25–29, 32, 33, 86, 89, 90, VIII, IX, XI

RoI Región de Interés. 38–40

Índice de Códigos

3.1. Generando el Ángulo de Elevación	48
3.2. Añadiendo la Ecuación de Irradiancia	54
4.1. Extracción de Datos	64
4.2. Uniendo los Archivos	68
4.3. Aumentando la Precisión de la Red Incluyendo Datos Computados . .	74
5.1. Red Neuronal Convolucional	79
5.2. Red Neuronal Recurrente modo LSTM	87

Capítulo I - Introducción

1.1. Antecedentes

En los últimos años, la unión de la tecnología de la energía solar y la inteligencia artificial ha realizado avances revolucionarios en la eficiencia y adaptabilidad de los sistemas de energía solar. Los retos presentes, como la naturaleza intermitente de la luz solar y las condiciones ambientales dinámicas, han impulsado a los investigadores a profundizar en el potencial de las redes neuronales como controladores inteligentes para optimizar el rendimiento de la energía solar.

La búsqueda mundial de fuentes de energía sostenibles y renovables se ha intensificado en respuesta a los retos que plantean el cambio climático y la necesidad de una transición hacia formas más limpias de generación de energía [5]. La energía solar, que se aprovecha de la luz del sol, la cual se encuentra abundantemente, se plasma como una solución prometedora. Sin embargo, la naturaleza intermitente y variable de la luz solar plantea retos a la hora de maximizar la eficiencia de los sistemas de energía solar. Esto mismo ha dado lugar a planteamientos innovadores para superar estos retos, con especial atención a la aplicación de redes neuronales.

Las redes neuronales son cada vez más importantes en el campo de la optimización de la energía solar debido a su capacidad para analizar relaciones complejas dentro de los datos y realizar predicciones, o tomar decisiones basadas en ese análisis [3]. En el contexto de la energía solar, las redes neuronales se emplean para optimizar diversos aspectos, como la orientación de los paneles solares, los mecanismos de seguimiento y la configuración general del sistema. Estas redes aprenden de los datos históricos y ajustan continuamente los parámetros para maximizar la recolección de energía en

diversas condiciones ambientales [11].

La integración de la inteligencia artificial, en particular de las redes neuronales, con los sistemas de energía solar ha sido objeto de numerosas investigaciones en los últimos años. La motivación se centra en la necesidad de mejorar la eficiencia, la adaptabilidad y el rendimiento general de los sistemas de energía solar. Investigaciones anteriores han profundizado en diversos aspectos de la utilización de redes neuronales para la optimización de la energía solar, revelando valiosos conocimientos y contribuyendo a la continua evolución de este campo interdisciplinario.

Las primeras investigaciones sobre la optimización de la energía solar mediante redes neuronales se llevaron a cabo debido a los retos asociados a los sistemas tradicionales de energía solar. La intermitencia de la luz solar, las variaciones de las condiciones ambientales y la complejidad de la optimización de los parámetros del sistema llevaron a los investigadores a explorar soluciones relacionadas a la inteligencia artificial. El objetivo era aprovechar la capacidad de aprendizaje de las redes neuronales para controlar de forma adaptativa los sistemas de energía solar y maximizar la captación de energía.

1.1.1. Revisión del Laboratorio Nacional de Energías

Renovables: Avances en la Predicción Solar en 2023

Una revisión técnica exhaustiva de la revista del Laboratorio Nacional de Energías Renovables (NREL, por sus siglas en inglés) analiza los avances recientes en la predicción solar, destacando el auge del aprendizaje profundo (redes neuronales convolucionales, recurrentes y modelos híbridos) y la fusión de múltiples fuentes (cámaras terrestres del cielo, satélites y modelos numéricos de predicción meteorológica) como la principal línea de investigación para las predicciones a muy corto y corto plazo. La revisión enfatiza que los modelos basados en imágenes capturan la estructura y el movimiento a escala

de nubes, algo que los modelos numéricos no logran a escalas subhorarias, y recomienda combinar características de visión artificial con información física previa (modelos de cielo despejado, geometría solar) para un rendimiento robusto en sistemas de producción. Esta revisión resume por qué el enfoque de redes neuronales convolucionales (CNN) + preprocesamiento de imágenes + guiado por la física es la mejor práctica actual. [16]

1.1.2. Predicción Mediante Imágenes del Cielo y Métodos Basados en Características

Al-lahham et al. Se propone un método para estimar la irradiancia solar a corto plazo directamente a partir de imágenes de cielo completo, extrayendo características de imagen diseñadas manualmente (índices de nubes, textura, intensidad de la luz) y utilizándolas en modelos de regresión/redes neuronales superficiales. Los experimentos muestran que las características de los generadores de imágenes del cielo reducen significativamente los errores de pronóstico a nivel de minuto en comparación con los modelos de referencia basados en persistencia, especialmente en condiciones de nubosidad variable. Este estudio es representativo de la generación de enfoques que evolucionaron desde características diseñadas manualmente hasta CNN de extremo a extremo. [2]

1.1.3. Estudios de Pronóstico Híbridos

Diversos estudios han demostrado que las arquitecturas híbridas utilizando CNN para la codificación espacial y Perceptrón Multicapa (MLP, por sus siglas en inglés) y Memoria Larga a Corto Plazo (LSTM, por sus siglas en inglés) para el modelado temporal— ofrecen un rendimiento excelente para el pronóstico a corto y largo plazo. Por ejemplo, El Alani et al. (2021) presentaron un híbrido CNN-MLP para pronósticos

de 15 minutos y reportaron reducciones significativas del RMSE en comparación con modelos de referencia simples. Trabajos posteriores extienden este enfoque a combinaciones de CNN-LSTM o CNN-Transformer para pronósticos a varios pasos y demuestran una mayor precisión en horizontes de varios minutos. Estos modelos híbridos son particularmente efectivos porque la CNN aprende la morfología de las nubes y el módulo recurrente/temporal aprende su movimiento y persistencia. [9]

1.1.4. Aprendizaje Profundo a Partir de Cámaras Celestes: Modelos Aplicados y Evaluaciones

Rajagukguk et al. y trabajos similares entrenan modelos profundos de CNN con conjuntos de datos de cámaras celestes para predecir la irradiancia y la cobertura de nubes; enfatizan el preprocesamiento robusto, la segmentación de nubes y el uso de estadísticas de luminancia por píxel como etiquetas o objetivos auxiliares. Estos estudios suelen evaluarse comparándolos con datos reales de piranómetros y encuentran que los modelos profundos reducen los errores a nivel de minuto, especialmente cuando se entrenan con conjuntos de datos grandes y etiquetados. También abordan el ruido en las etiquetas, la calibración de los sensores y la necesidad de una cuidadosa división entre entrenamiento y validación para evitar la fuga temporal. [19]

1.1.5. Pronóstico a Nivel de Minutos o Multipasos Mediante Modelos Temporales Avanzados

Un trabajo reciente (2024) propone híbridos LSTM-InformerStack y otros modelos temporales avanzados para generar pronósticos a nivel de minutos/multipaso (varios pasos adelante con una cadencia de 1 a 5 minutos). Estas arquitecturas buscan capturar tanto las dependencias locales (advección de nubes a corto plazo) como las globales

(estacionales lentas) y muestran mejoras con respecto a los enfoques estándar basados únicamente en LSTM. La tendencia demuestra que un modelado temporal preciso es fundamental al utilizar secuencias de imágenes del cielo. [27]

1.1.6. Análisis Comparativo y Reproducibilidad

Los artículos de análisis comparativo (p. ej., un análisis comparativo de arXiv de 2025) comparan diversas arquitecturas de aprendizaje profundo (solo CNN, CNN-LSTM, redes convolucionales temporales, modelos generativos) con conjuntos de datos estándar (p. ej., SIRTa, Folsom, conjuntos de datos del NREL) y demuestran que:

- La predicción inmediata basada en imágenes supera a los modelos de referencia básicos para horizontes temporales muy cortos.
- La elección del modelo es menos importante que un buen preprocesamiento y una correcta división del entrenamiento.
- La estimación de la incertidumbre y la calibración no se abordan suficientemente en la literatura.

Los análisis comparativos son útiles para posicionar la arquitectura de la red, los hiperparámetros y el protocolo de evaluación con respecto a los estándares de la comunidad. [4]

1.1.7. Aprendizaje por Refuerzo, Control de la Inteligencia Artificial para Seguimiento

Varios artículos aplicados (p. ej., Tsuchida et al., 2022) utilizan Aprendizaje por Refuerzo (RL, por sus siglas en inglés) profundo para controlar la inclinación de paneles (sistemas de un solo eje o bifaciales). Estos estudios definen funciones de recompensa que

equilibran la ganancia de energía inmediata y el costo del actuador, y utilizan métodos de gradiente de políticas / actor-crítico para aprender estrategias de seguimiento casi óptimas en condiciones de nubosidad variable. Los resultados suelen mostrar que las políticas del RL superan el seguimiento solar convencional en regímenes nublados y de alta variabilidad, ya que el RL aprende implícitamente a evitar movimientos innecesarios y a anticipar el paso de las nubes. Sin embargo, los autores señalan las dificultades de la implementación en el mundo real (eficiencia de muestreo, exploración segura, necesidad de simuladores). [25]

1.1.8. Aprendizaje por Refuerzo y Orientación de Helióstatos

Trabajos recientes extienden el RL a la orientación de heliostatos en sistemas de Energía Solar Concentrada (CSP, por sus siglas en inglés). Estos enfoques sin modelo optimizan las estrategias de orientación para maximizar el rendimiento de la planta ante la incertidumbre de las nubes y la atmósfera. Si bien los heliostatos constituyen una clase de hardware diferente, los métodos y las lecciones aprendidas (configuración de recompensas, restricciones de seguridad, integración de pronósticos) son aplicables a los sistemas de seguimiento fotovoltaico. Esta línea de investigación muestra un creciente interés en el RL para problemas complejos de control solar con un alto grado de libertad. [6]

1.1.9. Sistemas Aplicados Recientes y Estudios de Caso

Diversos estudios de caso y artículos de congresos (2021-2024) demuestran sistemas integrales: cámara celeste, extracción de características, CNN-LSTM, control de modelo predictivo con RL, actuador de campo. Las principales conclusiones de estos trabajos aplicados son consistentes:

- La calidad de la imagen y el registro preciso de la hora son fundamentales.
- La generación de etiquetas debe sincronizarse cuidadosamente con las lecturas del piranómetro.
- Las mejoras en el rendimiento dependen en gran medida de la climatología de nubes local.
- La cuantificación de la incertidumbre es esencial para la toma de decisiones robustas por parte del controlador. [17]

1.2. Planteamiento del Problema

La utilización de sistemas de energía solar presenta varios retos y problemas que las investigaciones y nuevos resultados pretenden resolver en la búsqueda de una adopción más eficaz y generalizada de la energía solar. Algunos de los principales problemas asociados a los sistemas de energía solar son:

1.2.1. Intermitencia y Variabilidad

La producción de energía solar es intrínsecamente intermitente debido a factores como los ciclos día-noche, las condiciones meteorológicas y los cambios estacionales. La nubosidad, las sombras y las inclemencias del tiempo pueden reducir significativamente la disponibilidad de luz solar, afectando a la generación constante de energía solar.

1.2.2. Eficiencia y Avances Tecnológicos

Aunque la eficiencia de los paneles solares ha mejorado a lo largo de los años, aún se puede mejorar. Los investigadores trabajan continuamente en el desarrollo de nuevos

materiales y tecnologías para mejorar la eficiencia de las células solares, haciéndolas más rentables y productivas.

1.2.3. Dependencias Geográficas y Climáticas

La ubicación geográfica y el clima influyen en la generación de energía solar. Las zonas con cielos constantemente nublados o luz solar limitada pueden no beneficiarse tanto de la energía solar. Para ello se busca ampliar los lugares en los que sea factible instalar los sistemas de paneles, tomando en cuenta las condiciones geográficas y climáticas de los mismos.

1.3. Justificación

La utilización de redes neuronales para optimizar el rendimiento radiante de un sistema de energía solar se justifica por varios factores clave que muestran las capacidades y ventajas únicas que ofrece este enfoque de inteligencia artificial.

1.3.1. Optimización Basada en Datos

Las redes neuronales funcionan según un patrón basado en datos, lo que les permite aprender de datos históricos y adaptarse a condiciones dinámicas. En el contexto de la optimización de la energía solar, esta capacidad permite a la red neuronal tomar decisiones informadas basadas en datos de rendimiento del mundo real, optimizando los parámetros del sistema para obtener la máxima exposición radiante.

1.3.2. Optimización Multivariante

Los sistemas de energía solar suelen implicar la optimización de múltiples parámetros, como los ángulos de inclinación y las orientaciones de los paneles solares, para maximizar la captación de energía. Las redes neuronales pueden optimizar simultáneamente estos parámetros multivariantes, proporcionando un variado enfoque a la optimización del rendimiento radiante que va más allá de las capacidades de los algoritmos tradicionales [11].

1.3.3. Reconocimiento y Generalización de Patrones

Las redes neuronales dominan el reconocimiento de patrones y la generalización. Al aprender de los datos históricos, pueden identificar patrones asociados al rendimiento radiante óptimo en diversas condiciones. Esta adaptabilidad y capacidad de generalización permiten a la red neuronal tomar decisiones eficaces incluso en escenarios no encontrados explícitamente durante el entrenamiento.

1.3.4. Estimación Energética

Las redes neuronales contribuyen a realizar estimaciones energéticas precisas a corto y largo plazo basadas en datos históricos y entradas en tiempo real. Esta capacidad de pronóstico ayuda a anticipar mejor los cambios en los niveles de radiación solar, facilitando la optimización proactiva del rendimiento radiante.

1.4. Objetivos

1.4.1. Objetivo General

Diseñar una red neuronal convolucional y recurrente para optimizar la extracción de energía de un panel solar.

1.4.2. Objetivos Específicos

- Extraer los datos de cada imagen a analizar en la red neuronal
- Utilizar todos estos datos para entrenar a la red.
- Obtener la posición de mayor irradiancia solar.
- Realizar una transformación de posición a ángulos de azimuth y cenital para corregir la posición del panel solar.

1.5. Hipótesis

El diseño de la red neuronal convolucional y recurrente optimizará la extracción de energía de un panel solar al encontrar los ángulos óptimos del mismo.

1.6. Alcances y Limitaciones

1.6.1. Alcances

- Se optimiza la extracción de energía solar.

- El tiempo de ejecución se reduce.
- Se puede usar en cualquier modelo panel solar.
- Vésatil con cualquier tipo de imágenes del cielo.

1.6.2. Limitaciones

- Mucho tiempo de entrenamiento en su fase inicial.
- Alto consumo de recursos informáticos al entrenar la red.
- Al tener gran cantidad de imágenes se requiere más tiempo de entrenamiento.
- El PC utilizado para entrenar la red no es tan efectivo al entrenar la misma.

Capítulo II - Marco Teórico

2.1. Introducción

La Inteligencia Artificial (IA) ha desempeñado un papel cada vez más importante en la optimización de los sistemas de paneles solares y la gestión inteligente, ayudando a mitigar una de las principales debilidades de las energías renovables: la variabilidad y la eficiencia. Los paneles solares en los sistemas convencionales se montan tradicionalmente con parámetros fijos, como la inclinación y el acimut, y rara vez consideran los cambios en el entorno, por ejemplo, las posiciones cambiantes del sol, la acumulación de polvo, las sombras o la nubosidad. La optimización con IA supera esta limitación mediante el análisis continuo de las entradas en tiempo real y la anticipación de la configuración óptima para obtener la máxima producción energética. Mediante el uso de conjuntos de datos históricos y en tiempo real que contienen la irradiancia solar, la temperatura ambiente, las condiciones climáticas e incluso características geográficas como la latitud y la topografía, los algoritmos de IA, en particular los basados en aprendizaje automático y aprendizaje profundo, pueden reajustar los parámetros del sistema para mejorar el rendimiento y la fiabilidad en tiempo real.

2.2. Geometría Solar y Captura de Energía

La geometría solar se refiere al principio básico que rige la interacción de la radiación solar con la superficie de la Tierra. Define la relación espacial y temporal entre el Sol, la ubicación del observador y la inclinación de los paneles solares diseñados para maximizar la irradiancia incidente. La posición del Sol en cualquier instante depen-

de principalmente de la rotación, la inclinación axial y la órbita elíptica de la Tierra alrededor del Sol. Estos parámetros astronómicos afectan algunos ángulos solares importantes, como el ángulo de declinación, el ángulo de altitud solar, el ángulo cenital y el ángulo de acimut. La comprensión de dichas relaciones permite, a su vez, una previsión precisa de la irradiancia solar en una superficie determinada y en determinadas condiciones atmosféricas.

El ángulo de declinación solar, que cambia en el rango de aproximadamente $23,45^\circ$ debido a la inclinación axial de la Tierra, define la posición aparente del Sol al norte o al sur del ecuador celeste. Puede aproximarse mediante relaciones empíricas derivadas de observaciones astronómicas que permiten el cálculo preciso de la trayectoria diaria del Sol. El ángulo horario, determinado convencionalmente con respecto a la hora solar en el meridiano local del mediodía, es una medida de la rotación de la Tierra expresada como un ángulo y participa activamente en el movimiento aparente del Sol a través del cielo. Estos ángulos, en conjunto, determinan el ángulo cenital solar (θ_z), que influye directamente en la irradiancia recibida sobre una superficie horizontal o inclinada. Cuanto más pequeño es el ángulo cenital, más perpendiculares inciden los rayos solares sobre el panel y, por lo tanto, se maximiza la entrada de energía por unidad de área.

La irradiancia total que llega a un panel solar depende tanto de la componente directa como de la difusa de la luz solar. La irradiancia directa es esencialmente una función del coseno del ángulo cenital, mientras que la irradiancia difusa está influenciada por fenómenos de dispersión debidos a moléculas atmosféricas, aerosoles y formaciones de nubes. Por lo tanto, la modelización precisa de la geometría solar es importante para comprender la variación dinámica de la irradiancia que tiene lugar a lo largo de un día y entre estaciones. El conocimiento preciso de dichos parámetros permite obtener ángulos de inclinación y acimut óptimos, para los cuales se pueden configurar los sistemas para que funcionen en tiempo real o para alcanzar el máximo rendimiento energético anual.

La geometría solar es la base teórica de esta investigación, que vincula las imágenes del cielo con el estado físico real de la radiación solar. Las relaciones geométricas entre la posición aparente del Sol y las imágenes del cielo capturadas permiten estimar la irradiancia solar directamente a partir de parámetros derivados de las imágenes. A través de este marco geométrico, cada imagen se convierte en una instantánea de la configuración Sol-Tierra-atmósfera en un momento específico, incorporando la información necesaria para el modelado de irradiancia.

2.3. Redes Neuronales en la Extracción de Energía Solar

Las RNA ofrecen el marco computacional que puede aprender, de manera general, las complejas relaciones no lineales entre las variables ambientales y la producción de energía solar. Específicamente, las CNN están bien adaptadas al análisis basado en imágenes porque pueden extraer características tanto espaciales como radiométricas de imágenes del cielo relacionadas con la irradiancia solar y las condiciones atmosféricas. A diferencia de los enfoques tradicionales de regresión o modelado físico, las CNN aprenden estos patrones directamente de los datos, ofreciendo así una alta adaptabilidad y solidez frente a cambios impredecibles en las condiciones climáticas y de iluminación.

En este trabajo, se desarrolló una arquitectura CNN para procesar imágenes del cielo y recuperar parámetros como la intensidad de la irradiancia, la cobertura de nubes, la variación óptica y la uniformidad del brillo atmosférico. Las capas convolucionales aprenden automáticamente características localizadas relevantes, como gradientes, cambios de textura y contrastes de brillo que corresponden a las nubes, el disco solar y la dispersión en la atmósfera. Luego, la agrupación de capas reduce la redundancia espacial, centrando la atención de la red en las características de irradiancia más rele-

vantes, mientras que las capas completamente conectadas mapean estas características aprendidas en predicciones de irradiancia cuantitativas.

En el proceso de entrenamiento de la red neuronal, pasa por una fase de aprendizaje supervisado donde las características de la imagen extraídas por el proceso de preprocesamiento (en Python) se consideran como entrada, mientras que sus valores de irradiancia correspondientes, que se calculan mediante modelado físico, actúan como salidas objetivo. Este modelo minimiza el error de predicción actualizando iterativamente sus pesos internos mediante un algoritmo de retropropagación y descenso de gradiente. Con el tiempo, la CNN aprende la relación intrínseca entre la apariencia del cielo, los patrones de brillo atmosférico y los niveles reales de irradiancia.

La integración de redes neuronales en la optimización de la energía solar tiene amplias ramificaciones en la gestión de las energías renovables. A partir del aprendizaje de datos de imágenes empíricas, el modelo estima dinámicamente la irradiancia solar en diversas condiciones, como cobertura parcial de nubes y neblina atmosférica, sin instrumentación meteorológica explícita. Más importante aún, amplía esta capacidad a la optimización en tiempo real de la orientación de los paneles solares. Las predicciones de irradiancia de la red neuronal podrían proporcionar información significativa para los algoritmos de control adaptativo que operan en tiempo real para ajustar los ángulos de inclinación y acimut para obtener la máxima potencia de salida en los sistemas fotovoltaicos.

En definitiva, la fundamentación teórica de esta investigación combina la geometría solar, que representa los principios físicos y astronómicos de la radiación solar, con la inteligencia artificial que proporciona la inteligencia informática necesaria para analizar y optimizar la captura de energía solar. Crea un modelo basado en datos, que cierra las brechas entre los métodos tradicionales basados en la física y las técnicas modernas de inteligencia artificial, impulsando el desarrollo de sistemas inteligentes de energía solar

capaces de adaptarse de forma autónoma y mejorar continuamente el rendimiento.

2.4. Efectos Atmosféricos y Modelado de Irradiancia

La atmósfera es esencial para modificar tanto la cantidad como la calidad de la radiación solar que llega a la superficie terrestre. Cuando la luz solar atraviesa las capas atmosféricas, lo hace sufriendo una serie de procesos físicos, fundamentalmente absorción, dispersión y reflexión, que en conjunto determinarán la irradiancia solar efectiva incidente sobre los sistemas fotovoltaicos. Conocer estos mecanismos es crucial para desarrollar modelos precisos para las predicciones de irradiancia y también para interpretar los datos visuales adquiridos mediante técnicas de imágenes del cielo.

2.4.1. Desglose de la Irradiancia Solar

La irradiancia solar que llega al suelo consta de tres componentes principales: irradiancia directa, difusa y reflejada.

La componente directa I_b denota un haz de radiación solar que llega a la superficie siguiendo un camino recto desde el sol sin ninguna desviación. Su intensidad depende de la constante solar, el ángulo cenital y la transmitancia atmosférica.

La componente difusa I_d incluye la radiación dispersada por moléculas de aire, aerosoles y partículas de nubes. Llega a la superficie desde múltiples direcciones, reduciendo así la uniformidad general de la irradiancia.

Finalmente, el componente reflejado I_r representa la radiación reflejada desde las superficies circundantes, como el suelo, los edificios o, en general, el terreno proximal.

Por tanto, la irradiancia global total I_t viene dada por la suma de estos tres términos:

$$I_t = I_b + I_d + I_r \quad (2.1)$$

En la mayoría de los estudios de estimación de irradiancia a cielo abierto, incluido este trabajo, el término reflejado es relativamente menor y puede ignorarse o incorporarse mediante factores de corrección empíricos.

2.4.1.1. Atenuación Atmosférica y Profundidad Óptica

La caída de la intensidad solar a través de la atmósfera se describe principalmente mediante la ley de Beer-Lambert mediante la expresión de la caída exponencial de la intensidad de la radiación en función de la masa de aire y la profundidad óptica, (τ):

$$I_b = I_0 \cdot e^{-\frac{\tau}{\cos(\theta_z)}} \quad (2.2)$$

donde $I_0 = 1361W/m^2$ es la constante solar y θ_z es el ángulo cenital solar.

La profundidad óptica, τ , expresa el efecto combinado de absorción y dispersión de los componentes atmosféricos, incluidos el vapor de agua, el dióxido de carbono, el ozono y los aerosoles. Una mayor atenuación implica una mayor profundidad óptica, lo que caracteriza los días brumosos o nublados. La profundidad óptica es menor cuando la atmósfera es más clara, lo que permite una mayor transmitancia.

En este proyecto el término de atenuación exponencial

$$e^{-\frac{\tau}{\cos(\theta_z)}} \quad (2.3)$$

participa en la fórmula de estimación de la irradiancia al modelar la interacción física

entre el sol y la atmósfera. Está estrechamente relacionado con el índice de uniformidad del brillo atmosférico, deducido de imágenes del cielo, que esencialmente cuantifica visualmente los mismos fenómenos ópticos.

2.4.2. Influencia de las Nubes y la Dispersión

Entre los factores, las nubes tienen la mayor y más dramática influencia sobre las variabilidades de la radiación solar. Las nubes, según su densidad, altitud y espesor óptico, pueden reflejar o absorber una parte sustancial de la radiación solar entrante. Incluso una capa de nubes fina o parcial puede difundir la luz solar, reduciendo la irradiancia directa y al mismo tiempo aumentando la fracción difusa.

Este efecto de dispersión conduce a distribuciones de brillo espacial más suaves en las imágenes capturadas, que pueden cuantificarse con parámetros como la variación óptica y la tasa de cobertura de nubes. Estas características basadas en imágenes se utilizan como sustitutos del coeficiente de dispersión física para permitir que la red neuronal infiera la atenuación de la irradiancia a partir de los patrones de brillo visual.

La dispersión se puede dividir en dispersión de Rayleigh debida a moléculas atmosféricas, responsables del color azul del cielo despejado, y dispersión de Mie producida por partículas más grandes, como polvo y gotas de agua. Ambos efectos alteran las propiedades direccionales y espectrales de la irradiancia y, por tanto, la intensidad aparente del Sol en las imágenes del cielo. Al incluir estos patrones en los datos de entrenamiento de la red neuronal, el modelo aprende a asociar gradientes de brillo específicos y morfologías de nubes con valores de irradiancia cuantitativos.

2.4.3. Modelado Atmosférico Basado en Imágenes

En este artículo, los parámetros atmosféricos se extraen directamente de las características radiométricas de las imágenes. Consideramos que la intensidad de cada píxel es una medida localizada de la intensidad de la luz entrante modulada por la transmisión atmosférica. El análisis propuesto basado en Python aproxima la distribución de la radiación solar en el momento de la captura aislando la región circular del cielo y normalizando las intensidades de los píxeles.

Se realizaron cálculos de irradiancia media, intensidad ponderada y uniformidad del brillo atmosférico para describir el equilibrio de la iluminación total y la transparencia óptica de la atmósfera. La relación entre la irradiancia media y máxima,

$$A_{atm} = \frac{\overline{I_D}}{I_{D,MAX}} \quad (2.4)$$

el cual es una medida directa de las condiciones de nubosidad y dispersión. De manera similar, la varianza óptica, V_{opt} , cuantifica la heterogeneidad espacial en la distribución del brillo, lo que ofrece información sobre el grado de radiación difusa versus directa. Lo hace convirtiendo los fenómenos atmosféricos en descriptores numéricos, cerrando así la brecha entre el modelado físico y el análisis de visión por computadora. Estos, junto con los datos de la imagen original, se introducen en una red neuronal para predecir valores de irradiancia de alta resolución temporal con alta precisión.

2.4.4. Integración en el Marco de la Red Neuronal

Los indicadores atmosféricos derivados del análisis de imágenes son una capa de entrada esencial para la red neuronal convolucional. Cada característica, como la cobertura de nubes, la variación óptica y la uniformidad del brillo atmosférico, codifica diferen-

tes facetas de la transferencia radiativa en la atmósfera [8]. Mediante la extracción de características convolucionales, CNN aprende automáticamente cómo interpretar estos indicadores en conjunto con patrones de imágenes espaciales, replicando así de manera efectiva la física radiativa a través de representaciones aprendidas. Esta integración de la ciencia atmosférica con el aprendizaje profundo permite que el modelo capture dependencias no lineales complejas que de otro modo serían difíciles de expresar analíticamente. Por ejemplo, la cobertura parcial simultánea de nubes, las profundidades ópticas que varían en el tiempo y la dispersión dinámica se modelan correctamente mediante las capacidades de aprendizaje adaptativo de la CNN. En consecuencia, el sistema puede generar estimaciones precisas de irradiancia con diversas condiciones atmosféricas, lo que permite mejorar el rendimiento de los algoritmos de optimización y seguimiento solar.

2.5. Inclinación y Azimut

La optimización de la inclinación y el azimut en los paneles solares probablemente sea uno de los parámetros más importantes para maximizar la captación de energía solar, ya que ambos parámetros afectan directamente la cantidad de luz a la que tendrá acceso un panel a lo largo del tiempo [14]. El ángulo de inclinación es el ángulo de inclinación del panel sobre la superficie terrestre, y el ángulo azimutal especifica su orientación respecto al norte verdadero (o la dirección de orientación del panel). Los ángulos deben ajustarse correctamente para que los rayos del sol incidan sobre la superficie del panel lo más directamente posible y así maximizar la irradiancia solar incidente y, por consiguiente, la potencia eléctrica generada. Cuando un panel solar se coloca en los ángulos correctos, los rayos del sol inciden perpendicularmente sobre las células fotovoltaicas, minimizando la pérdida por reflexión y maximizando la absorción de energía durante el día y durante todo el año.

Los ángulos ideales de inclinación y azimut varían según diversos factores, principalmente la latitud geográfica, la estacionalidad y las condiciones atmosféricas locales. En el ecuador, una inclinación baja permite una exposición uniforme a la luz solar directa durante todo el año, mientras que, en latitudes superiores al ecuador, una inclinación mayor compensa la menor altitud solar. Al igual que el ángulo azimutal determina en qué momento del día el panel recibe más luz solar: el hemisferio norte y la orientación sur reciben la mayor cantidad de luz solar, o una ligera inclinación hacia el este o el oeste puede mejorar su rendimiento si el consumo de energía es mayor durante la mañana o la tarde. Sin embargo, estas configuraciones estáticas basadas en reglas no consideran el clima diario y estacional, el sombreado ni otras condiciones dinámicas que influyen en la irradiancia.

Aquí es donde la optimización mediante IA se vuelve revolucionaria. Las máquinas de IA pueden utilizar datos en tiempo real y algoritmos de pronóstico para calcular y ajustar continuamente los ángulos óptimos de inclinación y azimut. Utilizando factores ambientales como la elevación solar, la intensidad de la irradiancia, la temperatura, la humedad e incluso la nubosidad, los algoritmos de aprendizaje automático pueden determinar la configuración que ofrece el máximo rendimiento energético en un momento específico. Por ejemplo, una red neuronal puede aprender de los datos históricos de irradiancia y rendimiento para comprender el efecto de diferentes combinaciones de ángulos en la generación de energía y, por lo tanto, proyectar la dirección óptima en tiempo real. Otros utilizan el aprendizaje por refuerzo, donde la IA ^{aprende sobre la marcha} aprende a refinar sus ajustes de ángulo gradualmente basándose en la retroalimentación de los datos reales de producción de energía [14].

De esta manera, el sistema podrá responder a cambios reales, como cambios en los patrones climáticos o la variación mensual en la trayectoria solar, garantizando que los paneles funcionen a su máxima capacidad en todo momento. Esta optimización adaptativa minimiza las pérdidas por ubicación no ideal, maximiza la producción energética

anual total y optimiza el retorno de la inversión de los sistemas solares. En grandes parques solares, este enfoque puede generar aumentos significativos en la producción total, mientras que en instalaciones residenciales o distribuidas, es un método económico para optimizar la producción energética de forma adaptativa. En algún momento, el ajuste muy fino de los ángulos de inclinación y acimut (específicamente con IA como fuerza impulsora) traduce los paneles solares estacionarios en dispositivos inteligentes y autoajustables con el potencial de extraer la mayor cantidad posible del sol bajo cualquier condición.

2.6. Red Neuronal Convolutional

La CNN es un tipo particular de red neuronal artificial, diseñada específicamente para procesar y examinar datos con topología de cuadrícula, principalmente imágenes, pero también puede aplicarse a series temporales, datos de sensores u otros datos de entrada correlacionados espacialmente. Las CNN son particularmente potentes porque pueden extraer de forma automática y eficiente características relevantes (patrones, formas, texturas, etc.) de datos de entrada sin procesar, sin necesidad de ingeniería de características. Su estructura refleja cómo la corteza visual humana procesa los datos visuales, desde patrones básicos hasta características cada vez más complejas.

En el núcleo de una CNN se encuentran tres tipos principales de capas: capas convolucionales, capas de agrupación y capas completamente conectadas.

2.6.1. Capas Convolucionales

Estas capas realizan la operación fundamental que da nombre a las CNN: la convolución. En esta, un pequeño conjunto de cifras, conocido como núcleo o filtro, recorre los datos de entrada (por ejemplo, una imagen) y multiplica sus valores por los valores

de los píxeles correspondientes. Estos se suman para producir un valor por ubicación, creando lo que se conoce como un mapa de características. Cada filtro está diseñado para detectar patrones específicos, como bordes, texturas o gradientes, en los datos. Durante el aprendizaje, la CNN descubre de forma independiente los valores óptimos de filtro que mejor capturan los patrones importantes para la tarea en cuestión (por ejemplo, la detección de una sombra, una forma o, en la optimización de paneles solares, la detección de la nubosidad o los patrones de orientación de los paneles a partir de imágenes).

Cada capa se somete a varios filtros, lo que permite a la CNN aprender varias características simultáneamente. Las características de bajo nivel (como bordes y líneas) suelen aprenderse en las primeras capas convolucionales, mientras que las características abstractas de alto nivel (como formas o incluso objetos completos) se aprenden en las capas más profundas.

2.6.2. Funciones de Activación

Después de cada operación de la capa convolucional, se emplea una función de activación (generalmente la Unidad Lineal Rectificada o ReLU) para proporcionar no linealidad al modelo. Esto es necesario porque la mayoría de los datos y relaciones del mundo real son no lineales. La función ReLU establece todos los valores negativos dentro del mapa de características a cero, lo que aumenta la eficiencia computacional de la red sin perder señales positivas cruciales.

2.6.3. Capas de Agrupación

Las capas de agrupación (o submuestreo) se utilizan para reducir el tamaño espacial de los mapas de características, lo que minimiza la complejidad computacional y evita

el sobreajuste de la red. La técnica de agrupación más común es la agrupación máxima, que toma el máximo elemento de regiones pequeñas (p. ej., parches de 2×2) del mapa de características. Esto ayuda a la CNN a capturar las características más destacadas sin sacrificar su información espacial, comprimiendo así la información más importante en una representación densa.

2.6.4. Capas Totalmente Conectadas

Tras una serie de capas convolucionales y de agrupación, las características de alto nivel resultantes se aplanan en un vector unidimensional y se pasan a través de una o más capas totalmente conectadas, de forma similar a una red neuronal tradicional. La decisión final la toman estas capas combinando todas las características aprendidas, por ejemplo, una clasificación de imágenes, una predicción de la irradiancia o una estimación de la salida de energía. Cada una de estas neuronas está conectada a todas las neuronas de la capa anterior para que la red pueda sumar toda la información extraída.

2.6.5. Capa de Salida

La capa final crea la salida de la red. Dependiendo de la tarea, esta puede ser un valor único (para tareas de regresión como la predicción de la generación de energía solar), valores múltiples (para la predicción de múltiples salidas) o probabilidades (para tareas de clasificación). En este punto, se emplea una función de activación adecuada, como sigmoidea, softmax o lineal.

El aprendizaje de una CNN implica introducir datos etiquetados en la red y contrastar las salidas predichas con los valores reales mediante una función de pérdida (por ejemplo, error cuadrático medio o entropía cruzada). La diferencia, o error, se propaga hacia atrás a través de la red mediante un procedimiento llamado retropropagación,

ajustando los valores de los filtros y las ponderaciones según un algoritmo de optimización (generalmente, descenso de gradiente estocástico o Adam). Tras varias ejecuciones, la CNN aprende a minimizar este error y a extraer características más precisas y útiles.

2.7. Red Neuronal Recurrente

Una Red Neuronal Recurrente (RNN, por sus siglas en inglés) es un tipo especial de red neuronal artificial, especialmente diseñada para manejar datos secuenciales; es decir, datos en los que la secuencia de entrada es significativa, por ejemplo, series temporales, voz, datos meteorológicos o lecturas de sensores. A diferencia de las redes neuronales de propagación hacia adelante tradicionales, que procesan las entradas individuales de forma aislada, una RNN posee una forma de memoria que le permite percibir dependencias temporales y relaciones entre las entradas anteriores y actuales. Por lo tanto, es especialmente adecuada cuando el contexto temporal es primordial, por ejemplo, para pronosticar patrones de irradiación solar a lo largo del día, predicciones de la producción energética o análisis del rendimiento a largo plazo de paneles solares en condiciones meteorológicas variables.

En esencia, una RNN funciona basándose en el concepto de conexiones recurrentes, ya que la salida de una neurona en un paso temporal se utilizará como parte de la entrada a la red para el siguiente paso. Este ciclo le otorga a la RNN la capacidad de retener información de entradas anteriores y utilizarla para condicionar predicciones posteriores. Es decir, las RNN tienen su propio tipo de memoria temporal para poder aprender secuencias de datos en lugar de instantáneas.

2.7.1. Desglosando Paso a Paso el Funcionamiento de una RNN

2.7.1.1. Estructura y Flujo de Información

Suponiendo un flujo de entrada $x_1, x_2, x_3, \dots, x_T$ (por ejemplo, mediciones de irradiación por hora). La RNN procesa cada entrada secuencialmente, manteniendo un estado oculto h_t , un tipo de memoria. En cada paso de tiempo t , la RNN toma la entrada actual x_t y el estado oculto previo h_{t-1} , y calcula un nuevo estado oculto mediante la siguiente fórmula:

$$h_t = f(W_{xh}x_t + W_{hh}h_{t-1} + b_h) \quad (2.5)$$

Aquí, W_{xh} y W_{hh} son las matrices de ponderación que determinan la contribución de cada uno de los estados de entrada y oculto previo al estado actual, b_h es un sesgo y f es una función de activación no lineal, como **tanh** o **ReLU**. Este estado oculto h_t registra todo lo que la red ha visto hasta el tiempo t . La RNN puede entonces generar un y_t en cada paso de tiempo, que normalmente se calcula como:

$$y_t = g(W_{hy}h_t + b_y) \quad (2.6)$$

donde W_{hy} y b_y representan los pesos y el sesgo de la capa de salida, y g una función de activación (p. ej., softmax para clasificación o lineal para regresión).

2.7.1.2. Memoria y Dependencias Temporales

La principal ventaja de las RNN es que pueden aprender a modelar patrones temporales. Dado que cada estado oculto es una función de la entrada actual y del estado oculto anterior, la red aprende a conservar información de lugares anteriores en la secuencia. Esto es apropiado para modelar patrones que cambian con el tiempo, como el cambio diurno en la irradiancia solar o los patrones climáticos que cambian lentamente y que afectan la generación solar.

2.7.1.3. Entrenamiento con Retropropagación a Través del Tiempo

El entrenamiento de las RNN consiste en optimizar una función de pérdida que captura la diferencia entre los resultados previstos y los resultados reales. Dado que las RNN manejan secuencias, el entrenamiento utiliza un algoritmo especial llamado Retropropagación a Través del Tiempo (BPTT, por sus siglas en inglés). La RNN se "despliega" a lo largo de todos los pasos de tiempo de la secuencia de entrada en BPTT, y se calculan gradientes en todos los pasos para actualizar los pesos.

Si bien es potente, esto conlleva inconvenientes: a medida que aumenta la longitud de la secuencia, los gradientes se vuelven extremadamente pequeños (problema del gradiente evanescente) o extremadamente grandes (problema del gradiente explosivo), y se vuelve cada vez más difícil para las RNN simples aprender dependencias a largo plazo.

2.7.1.4. Variantes: Memoria Larga a Corto Plazo y Unidades Recurrentes Cerradas

Estos problemas se resolvieron mediante arquitecturas de RNN más avanzadas que se desarrollaron:

- Las LSTM introducen estructuras especiales conocidas como puertas que controlan cómo se emite, se olvida y se almacena la información en cada paso de tiempo. Las puertas permiten a las LSTM olvidar información irrelevante durante largos períodos de tiempo y recordar información importante, abordando eficazmente el problema del gradiente evanescente.
- Las Unidades Recurrentes Cerradas (GRU, por sus siglas en inglés) simplifican la arquitectura de las LSTM al combinar varias puertas, lo que reduce la complejidad computacional sin sacrificar el rendimiento.

Tanto las GRU como las LSTM tienen amplias aplicaciones en problemas de series temporales, como la predicción de la irradiancia solar, la predicción energética y la predicción meteorológica.

2.7.2. Aplicaciones en la Optimización de la Energía Solar

En la optimización de paneles solares, el modelado predictivo puede ser una aplicación importante de las RNN. Por ejemplo, permiten analizar registros históricos de irradiancia, tendencias de temperatura y patrones de movimiento de las nubes para pronosticar el nivel de energía solar disponible en las próximas horas o días. Esta información se puede utilizar para modificar dinámicamente los ángulos de inclinación y azimut u optimizar los sistemas de almacenamiento de energía. Las RNN pueden combinarse con otros módulos de IA (por ejemplo, CNN-LSTM) para procesar información espacial y temporal en paralelo; por ejemplo, procesando imágenes del cielo junto con datos de irradiancia como una serie temporal.

Capítulo III - Metodología

3.1. Usando Redes Neuronales en una Base de Datos de 100,000 Imágenes

Una base de datos con 100,000 imágenes variables en el tiempo es un recurso muy valioso para entrenar y validar modelos de inteligencia artificial, como las CNN y las RNN, con el fin de estimar y predecir la irradiancia solar. Cada imagen se asocia con un instante temporal y representa visualmente atributos ambientales como la nubosidad, la intensidad solar, la turbidez atmosférica y la distribución de las sombras [10]. En conjunto, este conjunto de datos presenta variaciones espaciales y temporales que impactan directamente en la cantidad de energía solar en la superficie terrestre. El objetivo principal de utilizar este conjunto de datos es que las redes neuronales aprendan y generalicen modelos visotemporales que se correspondan con los valores de irradiancia, lo que permite una predicción precisa de la disponibilidad de energía solar basándose únicamente en imágenes del cielo.

3.1.1. Composición y Etiquetado del Conjunto de Datos

Cada imagen de la base de datos de 100,000 muestras contiene una medición de irradiancia resultante, obtenida mediante sensores solares o piranómetros calibrados en el momento de la captura. Esto permite que cada evento visual tenga una representación cuantitativa exacta, en la que se puede basar el aprendizaje supervisado. Cada observación en la base de datos incluirá:

- La imagen ambiental o del cielo tomada en tiempo real.

- La marca de tiempo, para mantener el orden cronológico y la consistencia temporal.
- La irradiancia solar global medida (en W/m^2).
- Parámetros secundarios opcionales, como la temperatura ambiente, la humedad y el ángulo de elevación solar, que podrían proporcionar contexto adicional al proceso de aprendizaje. Estos parámetros se obtendrán de la red neuronal mediante diferentes parámetros seleccionados por el usuario.

Estos datos estructurados permitirán el desarrollo de modelos robustos que puedan relacionar las características atmosféricas visuales con los niveles de irradiancia correspondientes en diferentes condiciones ambientales.

3.1.2. Preprocesamiento y Preparación de Datos

Antes del entrenamiento, el conjunto de datos de imágenes se preprocesa en una serie de pasos para que sea uniforme y compatible con el modelo. Normalización y redimensionamiento de imágenes: Todas las imágenes se redimensionarán a una resolución estándar (224 x 224 píxeles) y se normalizarán a un rango de intensidad común (0-1) [18].

- Aumento de datos: Se emplean transformaciones controladas, como pequeñas rotaciones, ajustes de brillo y normalización de contraste, para reforzar la generalización del modelo y revertir el sobreajuste, sin disminuir la relevancia física de los aspectos visuales del cielo.
- Sincronización temporal: Cada lectura de irradiancia debe sincronizarse perfectamente con el tiempo de captura de imágenes para mantener la integridad de los datos, especialmente cuando se emplea modelado temporal.

- Este proceso de preprocesamiento garantiza la consistencia en todas las muestras y prepara el conjunto de datos para un aprendizaje eficiente de redes neuronales.

3.1.3. Extracción de Características Espaciales Mediante CNN

El extractor de características visuales del sistema es la red neuronal convolucional. La CNN identifica automáticamente las características espaciales que influyen significativamente en la irradiancia solar mediante una serie de operaciones convolucionales y de agrupación. Las características podrían ser:

- Espesor óptico, forma y densidad de las nubes.
- Luminosidad y posición aparente del sol.
- Gradientes de color atmosférico debidos a efectos de dispersión.
- Distribución de sombras y oclusiones parciales sobre la región o paneles del cielo.

Cada foto se pasa a través de la CNN y se convierte en un vector de características de alta dimensión, que captura las características espaciales más informativas. Las representaciones capturan las características físicas del cielo mediante correlaciones de variabilidad de la irradiancia, lo que proporciona la base visual para el posterior análisis temporal [21].

Si no se requiere modelado temporal, las características extraídas de la CNN pueden incorporarse directamente a capas completamente conectadas para realizar una estimación de la irradiancia basada en regresión, lo que permite predecir el nivel de energía solar en un momento dado a partir de una sola imagen.

3.1.4. Modelado Temporal con RNN o LSTM

Dado que las imágenes de la base de datos no son estáticas y cambian con el tiempo, el uso de RNN, como las LSTM o GRU, permite al modelo comprender las dependencias temporales y los patrones cambiantes de la evolución de la irradiancia solar.

Con esta configuración mixta, la imagen es procesada inicialmente por la CNN para obtener características espaciales y luego se alimenta secuencialmente a la RNN. La RNN está entrenada para interpretar cómo las características evolucionan con el tiempo y, por lo tanto, representa el movimiento de las nubes, los cambios en la iluminación y otros efectos temporales sobre la irradiancia. Por ejemplo:

- Un aumento de la densidad de nubes en las imágenes posteriores puede indicar una disminución de la irradiancia.
- Por el contrario, la disipación de nubes podría indicar un aumento inminente de la intensidad solar.

Al realizar estas correlaciones de series temporales, el sistema CNN-RNN no solo puede predecir la irradiancia actual, sino también los valores futuros a corto plazo para mejorar la adaptabilidad y la precisión de los pronósticos de los sistemas de energía solar.

3.1.5. Entrenamiento y Validación del Modelo

El entrenamiento de la red neuronal se logra introduciendo secuencias o lotes de pares imagen-irradiancia en el modelo y ajustando sus parámetros mediante retropropagación. La función de pérdida, generalmente el error cuadrático medio entre la irradiancia predicha y la medida, cuantifica la calidad del modelo. El proceso de entrenamiento ajusta los pesos de la red mediante el descenso de gradiente o sus variantes (p. ej., el

optimizador Adam) para minimizar este error iterativamente.

Al contar con 100,000 imágenes, el conjunto de datos es suficiente para abarcar diversas condiciones ambientales y atmosféricas (desde días soleados y despejados hasta escenarios nublados o parcialmente nublados) sin que su procesamiento con las configuraciones de entrenamiento aceleradas por GPU sea costoso desde el punto de vista computacional. La correcta división del conjunto de datos en conjuntos de entrenamiento, validación y prueba garantiza una evaluación y generalización imparciales del modelo.

3.1.6. Implementación y Aplicaciones en Tiempo Real

El modelo CNN-RNN puede emplearse en numerosas aplicaciones útiles tras su entrenamiento:

- Estimación instantánea de la irradiancia: Utilizando fotografías del cielo en tiempo real tomadas por cámaras o drones, el sistema puede calcular la irradiancia solar instantánea sin necesidad de sensores físicos.
- Predicción de la irradiancia a corto plazo: A partir de flujos de imágenes recientes, el modelo puede predecir la irradiancia solar para el futuro cercano, lo que permite la gestión proactiva de la energía y la estabilidad de la red.
- Adaptación dinámica de la inclinación y el azimut de los paneles solares: Los valores de irradiancia previstos pueden introducirse en algoritmos de optimización que adaptan dinámicamente la inclinación y el azimut de los paneles solares para una captación óptima de energía a lo largo del día.

Estas aplicaciones resultan en una mayor eficiencia energética, control predictivo y autooperación en sistemas de energía solar.

3.1.7. Ventajas de Usar una Base de Datos de 100,000

Imágenes

Usar una base de datos de 100,000 imágenes ofrece muchas ventajas significativas:

- Amplia cobertura de diversas condiciones atmosféricas y cambios de iluminación.
- Continuidad temporal, que permite modelar cambios suaves de irradiancia.
- Gran diversidad de datos, que mejora la generalización sobre la variabilidad estacional y climática.
- Escalabilidad, que permite la futura ampliación del modelo mediante la incorporación de más imágenes o la integración con otras fuentes de datos ambientales.

3.2. Dataset a Utilizar

El conjunto de datos utilizado en este trabajo corresponde a imágenes de alta resolución del cielo tomadas entre marzo y diciembre de 2017, pertenecientes a la colección Sky Images de la Universidad de Stanford. Estas imágenes se capturaron mediante un sistema de cámara terrestre calibrada, diseñado para capturar, monitorizar y registrar los cambios en la irradiancia solar, la dinámica de las nubes y las condiciones atmosféricas a lo largo del tiempo. Cada imagen ofrece una vista hemisférica de la bóveda celeste para observar las variaciones en la intensidad solar, la cobertura de nubes y la luminancia del cielo bajo diferentes condiciones temporales y meteorológicas. Los metadatos, codificados en el nombre de archivo de cada imagen, incluyen la fecha y hora de adquisición en alta resolución y son fundamentales para correlacionar parámetros geométricos solares como los ángulos cenital y azimutal. Este conjunto de datos contiene alrededor de 100,000 imágenes, que pueden utilizarse para entrenar y validar modelos

de aprendizaje profundo, especialmente redes neuronales convolucionales y recurrentes, en la estimación y predicción de los niveles de irradiancia solar. El periodo de nueve meses que abarca el conjunto de datos permite incluir casi todas las variaciones estacionales y atmosféricas y las posiciones solares, lo que proporciona una representación adecuada de las condiciones del mundo real que pueden ser necesarias para optimizar adecuadamente los sistemas de energía solar.

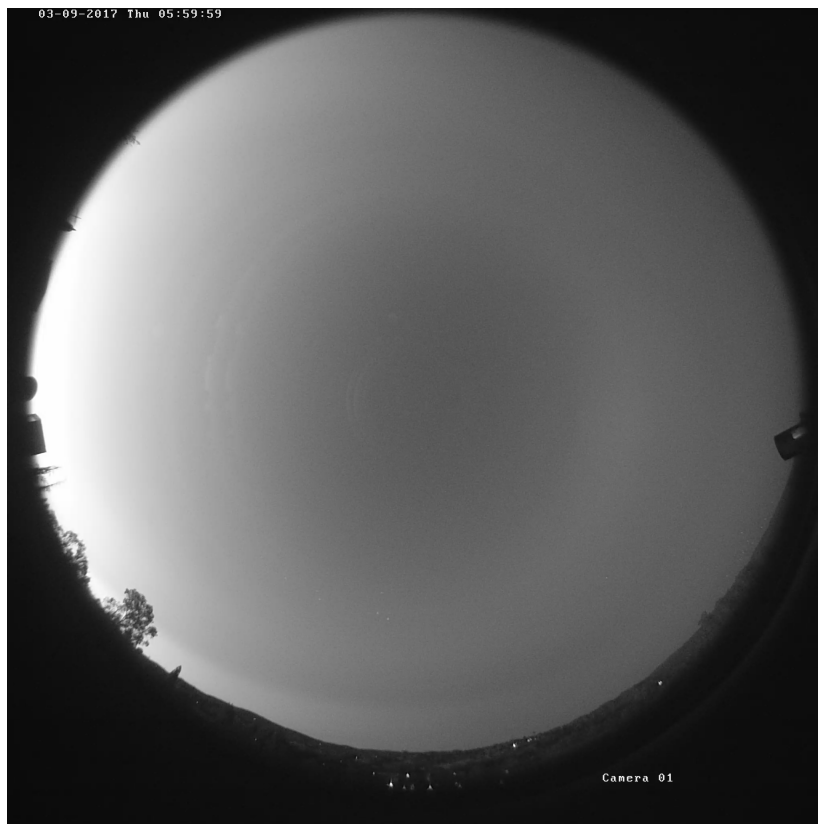


Figura 3.1: Imagen a utilizar en la base de datos de la red, todas las imágenes comparten esta apariencia

En la figura 3.1 se pueden apreciar los datos de la misma, así como un ejemplo de la cámara que se utilizó y el rango de visión que esta tiene (siendo de -90° a 90°), en ella se puede apreciar una imagen con poca intensidad solar, la fecha siendo de el 9 de marzo de 2017 y la hora siendo las 6 (en formato de 24 horas). Cada imagen necesita ser sesgada de acuerdo a la forma circular de la cámara.

3.3. Extracción de los Datos de Imagen para la Estimación de Irradiancia Solar

La extracción de datos significativos de la base de datos de imágenes constituye un paso vital en el entrenamiento y la aplicación de las redes neuronales construidas para la estimación de la irradiancia solar. Todas las imágenes de la base de datos contienen datos espaciales y temporales inherentes que deben procesarse y cuantificarse minuciosamente para extraer características representativas de las condiciones atmosféricas y solares. Las siguientes subsecciones describen cómo se extrajeron, interpretaron y formatearon los datos asociados a cada imagen para garantizar la precisión y consistencia del procesamiento posterior mediante aprendizaje automático [15].

3.3.1. Extracción de Datos Temporales

Cada archivo de imagen se nombra con la fecha exacta de captura, en un formato estándar (YYYYMMDDHHMMSS.jpg). Esta nomenclatura permite el cálculo automático de atributos temporales como año, mes, día, hora, minuto y segundo mediante técnicas de análisis de cadenas. Estas variables temporales se utilizan posteriormente para determinar los parámetros de la posición solar, como la elevación solar y los ángulos de azimut solar, mediante fórmulas astronómicas bien documentadas.

Esta información temporal es valiosa porque permite correlacionar cada imagen con la geometría solar en el momento de la adquisición. Por lo tanto, se puede analizar la variación de la irradiancia solar en función del movimiento aparente del sol y correlacionar los datos visuales con otros parámetros ambientales, como la temperatura o la humedad, si existen.

3.3.2. Extracción de Datos Radiométricos a Partir de la Intensidad de los Píxeles

La principal variable que se infiere de cada imagen es la irradiancia solar, que se estima mediante el análisis de la intensidad de los píxeles en el fotograma. Dado que las imágenes abarcan la porción visible del cielo, el brillo de cada píxel es una cuantificación directa de la radiación solar incidente sobre la superficie, modulada por las propiedades de la atmósfera (nubosidad, dispersión y contenido que opaca la luz solar).

Cada píxel se convierte en una matriz de intensidad en escala de grises, cuyo valor se equipara a la energía luminosa incidente en el sensor. A continuación, se calcula la intensidad media de los píxeles para todos los píxeles de interés, lo que proporciona una aproximación de primer orden del brillo solar total en el fotograma. Simbólicamente, esto se puede expresar como:

$$I_{mean} = \frac{1}{n} \sum_{i=1}^n I_i \quad (3.1)$$

donde I_i es la intensidad de cada píxel válido y n es el número de píxeles en el área de análisis.

La intensidad media es una buena estimación de la cantidad de irradiancia global, que suele aumentar con cielos más transparentes y una radiación solar más intensa, y disminuir con la nubosidad o la atenuación atmosférica.

3.3.3. Localización del Sol y Análisis de Distancia

Para realizar una estimación más precisa de la irradiancia, es necesario posicionar el Sol dentro de la imagen y medir el gradiente de intensidad con respecto a esta ubicación

central. El Sol se caracteriza típicamente por una región local de brillo máximo. Por lo tanto, el píxel (o pequeña región) que contiene el valor de luminancia máximo se encuentra mediante una búsqueda de máximo local de la matriz de intensidad. Una vez determinadas las coordenadas del centro del sol (x_s, y_s) , la distancia radial de cada píxel desde este centro se calcula como:

$$d_i = \sqrt{(x_i - x_s)^2 + (y_i - y_s)^2} \quad (3.2)$$

Al simular la relación entre la intensidad del píxel I_i y su distancia correspondiente d_i , es posible definir la distribución espacial de la irradiancia en la imagen. Los píxeles más cercanos al centro del sol son más intensos debido a la exposición solar directa, mientras que los píxeles más alejados del sol ven luz dispersa o atenuada. La intensidad media ponderada por distancia se puede calcular para mejorar la estimación de la irradiancia:

$$I_w = \frac{\sum_{i=1}^n I_i w_i}{\sum_{i=1}^n w_i}, \text{ where } : w_i = e^{-\alpha d_i} \quad (3.3)$$

Aquí, α es un coeficiente de atenuación ajustable que controla la influencia de los píxeles en función de su distancia al centro del sol. Este enfoque prioriza las regiones más directamente afectadas por la radiación solar y suprime las contribuciones periféricas.

3.3.4. Selección de la Región de Interés y Rechazo de Píxeles Irrelevantes

Dado que las imágenes se recortan circularmente, es necesario segmentar únicamente los píxeles que se encuentran dentro de la región de interés (RoI, por sus siglas en inglés) circular de la vista del cielo, eliminando así los bordes y los detalles no deseados. A cada

imagen se le aplica una máscara binaria que conserva únicamente el área circular válida. La máscara se da como:

$$M(x, y) \begin{cases} 1, & \text{if } (x-x_c)^2 + (y-y_c)^2 \leq r^2 \\ 0, & \text{otherwise} \end{cases} \quad (3.4)$$

donde (x_c, y_c) denota el centro de la región circular y r su radio. A continuación, la matriz de intensidad $I(x, y)$ se multiplica elemento por elemento por la máscara $M(x, y)$ para eliminar los píxeles de fondo irrelevantes más allá de la RoI.

Esto es necesario para garantizar que las regiones no celestes, como el reflejo del suelo, el encuadre de la cámara o los objetos circundantes, no sesguen la estimación de la irradiancia. Al limitar la consideración al campo de visión circular, el análisis conserva la representación completa de las características de la iluminancia atmosférica y solar.

3.3.5. Estimación de la Condición Atmosférica

Además del cálculo de la irradiancia, los datos de píxeles obtenidos pueden utilizarse para obtener condiciones atmosféricas como la nubosidad, el espesor óptico y la presencia de neblina. Se aplican diversas técnicas estadísticas y de procesamiento de imágenes a los píxeles válidos en la región de interés:

- La segmentación por umbral o la agrupación basada en gradientes de intensidad y color se aplican a la detección de nubes. Los píxeles blancos brillantes con alta varianza local uniforme simulan nubes, mientras que los píxeles azules uniformes simulan un cielo despejado. El espesor óptico se pronostica basándose en la varianza y distribución de los valores de intensidad. Una varianza alta suele ser típica de densidades de nubes variables, mientras que una varianza baja simula condiciones atmosféricas homogéneas.

- El análisis de la relación de color (en el espacio de color RGB) también puede aplicarse para discriminar entre tipos de nubes y efectos de dispersión atmosférica, especialmente con datos multiespectrales.
- Las características generadas (fracción de nubes, uniformidad de brillo e índice de contraste) también pueden emplearse como indicadores auxiliares de la transparencia atmosférica y combinarse con la estimación de la irradiancia para mejorar la precisión del modelo.

3.3.6. Integración y Estructuración de Datos

Una vez extraídas las características temporales, radiométricas, espaciales y atmosféricas, todos los parámetros se contabilizan en un registro de datos estructurado para cada imagen. Cada registro tiene los siguientes atributos:

Parámetro	Descripción
Marca de Tiempo	Extraída del Nombre del Archivo
Posición Solar	Ángulos de Azimut e Inclinación
Intensidad Media	Intensidad Promedio de Píxeles en la RoI
Intensidad Ponderada	Intensidad Corregida por Distancia
Cobertura de Nubes (%)	Oclusión Estimada del Cielo
Varianza Óptica	Variabilidad de la Intensidad de Píxeles
Coordenadas Solares	Centro de la Región de Máxima Intensidad
Indicadores Atmosféricos	Datos Ópticos y de Color

Tabla 3.1: Tabla de parámetros a utilizar en las redes neuronales

Este formato de datos garantiza que cada imagen presente un vector completo de características que miden tanto los aspectos radiométricos como físicos de la escena observada. Estas características sirven como entrada para los modelos de redes neuronales que permiten el aprendizaje y la predicción precisa de la irradiancia solar.

3.4. Geometría Solar

La geometría solar se refiere a la investigación de las relaciones temporales y espaciales entre la Tierra y el Sol, específicamente a cómo varía la ubicación del Sol a lo largo del día y de una estación a otra. Proporciona una base geométrica y matemática para calcular parámetros como el ángulo de altitud solar, el ángulo azimutal, la declinación y el ángulo horario, que definen el movimiento aparente del Sol en el cielo con respecto a una ubicación en la Tierra. Estos parámetros explican el comportamiento de la luz solar sobre paneles solares, edificios y otras superficies. La geometría solar se define por la latitud geográfica, la inclinación axial de la Tierra y la órbita alrededor del Sol, y por lo tanto es un factor fundamental para la predicción de los niveles de irradiancia solar y la optimización de la energía solar que se puede captar en cualquier lugar y momento.

La geometría solar se aplica en este proyecto como base teórica para la optimización de la orientación y la eficiencia de los paneles solares mediante inteligencia artificial. Al superponer los modelos geométricos con información en tiempo real y la estimación de la irradiancia basada en imágenes, el sistema puede determinar los ángulos óptimos de inclinación y azimut de cada configuración de panel para maximizar la absorción de energía solar. Las ecuaciones geométricas de la orientación del panel que modelan la posición del Sol se combinan con lecturas de irradiancia obtenidas de imágenes satelitales o terrestres para que la red neuronal pueda mapear la posición solar con la energía observada. Además, la geometría solar considera las condiciones atmosféricas y estacionales, lo que permite que el modelo de IA pueda generalizarse a través de estaciones y geografía. Básicamente, la geometría solar combina la comprensión física del movimiento del Sol con las capacidades computacionales de las redes neuronales para brindar asistencia precisa y adaptativa en la optimización de la energía solar en este proyecto.

3.4.1. Ecuación General de la Irradiancia Solar

La ecuación general de la irradiancia solar en función del día del año, la hora y el ángulo cenital se puede expresar como:

$$I = I_{sc}E_0\cos\theta_z \quad (3.5)$$

donde:

- I = Irradiancia solar instantánea sobre una superficie horizontal, expresada en W/m^2 .
- I_{sc} = Constante solar, aproximadamente $1361 W/m^2$, la irradiancia solar promedio recibida fuera de la atmósfera terrestre.
- E_0 = Factor de corrección de excentricidad, que representa la variación de la distancia Tierra-Sol a lo largo del año.
- θ_z = Ángulo cenital solar, el ángulo entre la vertical en un punto y la línea que une el Sol

El factor de corrección de excentricidad E_0 es:

$$E_o = 1 + 0.033\cos\left(\frac{360^\circ}{365.2425}n\right) \quad (3.6)$$

donde n es el número del día del año ($n = 1$ para el 1 de enero, $n = 365$ para el 31 de diciembre). Esta corrección se aplica para modificar la irradiancia en función de la órbita elíptica de la Tierra, ya que el planeta está ligeramente más cerca del Sol a principios de enero y más lejos a principios de julio. El ángulo cenital solar θ_z se determina a partir de la geometría solar como:

$$\cos(\theta_z) = \sin(\varphi)\sin(\delta) + \cos(\varphi)\cos(\delta)\cos(H) \quad (3.7)$$

donde:

- φ = Latitud del punto de observación (grados)
- δ = Ángulo de declinación solar, la inclinación de la Tierra con respecto al Sol, como:

$$\delta = 23.45^\circ \sin\left(\frac{360^\circ}{365.2425}(n - 81)\right) \quad (3.8)$$

- H = Ángulo horario, el movimiento aparente del Sol en el cielo, como:

$$H = 15^\circ(LST - 12) \quad (3.9)$$

El tiempo solar (también conocido como tiempo solar verdadero o tiempo solar aparente) es la hora que indica la trayectoria aparente real del Sol a través del cielo a una longitud específica; es decir, son las 12:00 cuando el Sol cruza el meridiano local (mediodía solar local). Dado que los relojes civiles tienen husos horarios estándar y el movimiento orbital de la Tierra no es perfectamente uniforme, el tiempo solar se compone de dos efectos distintos de la hora del reloj: (1) el movimiento aparente del Sol con respecto a una longitud constante (desplazamiento geométrico de longitud entre el meridiano local y el meridiano estándar del huso horario), y (2) la Ecuación del Tiempo debido a las variaciones estacionales producidas por la oblicuidad y la excentricidad orbital de la Tierra. Estos dos efectos se suman en una única corrección horaria (normalmente en minutos) que convierte la hora civil local (estándar) en hora solar.

Matemáticamente, la conversión se presenta normalmente como un pequeño conjunto de fórmulas compactas. Primero, calcula la ecuación del tiempo (en minutos) usando una aproximación analítica estándar:

$$B = \frac{360^\circ}{365.2425}(n - 81) \quad (3.10)$$

$$E_{oT} = 9.87\text{sen}(2B) - 7.53\text{cos}(B) - 1.5\text{sen}(B) \quad (3.11)$$

donde: n es el día del año (1...365; con una corrección para la duración real del año solar) y las funciones trigonométricas usan B en grados (se convierte a radianes al calcular numéricamente). La expresión indica la diferencia entre el tiempo solar aparente (el que muestra un reloj de sol) y el tiempo solar medio (el tiempo regular que marca un Sol medio imaginario que se mueve a velocidad constante) y es representado en minutos.

Ahora se calcula la corrección en la longitud con respecto a la zona horaria, el ángulo de la longitud se representa de manera positiva si se encuentra al este del Meridiano de Greenwich, y negativo si se encuentra al oeste. Dicha corrección en el tiempo (también representada en minutos) es representada por la siguiente ecuación:

$$TC = 4 * \text{rem}(\lambda, 15) + E_{oT} \quad (3.12)$$

donde:

- El factor 4 surge porque el Sol se mueve aproximadamente 1° de longitud cada 4 minutos.
- λ siendo la Longitud en grados
- $\text{rem}(x_1, x_2)$ siendo una función que devuelve el residuo que resulta al dividir x_1 y

x_2 , respetando el signo de la Longitud.

Ahora para obtener el tiempo local con la corrección en minutos de acuerdo a la posición de la tierra con respecto al sol, se utilizará la siguiente ecuación:

$$LST = h + \frac{TC}{60} \quad (3.13)$$

en dicha ecuación se convierte la corrección del tiempo (la cual es representada en minutos) a horas, y h representa la hora que marcaría el reloj en dicha ubicación (respetando el cambio de hora dado por los meridianos), esta hora está convertida del factor de tiempo a un número al convertir los minutos y segundos en decimales.

Esta ecuación aproxima colectivamente la variación de la irradiancia solar a lo largo del día, el año y la ubicación geográfica.

El coseno del ángulo cenital determina la dirección con la que la luz solar incide sobre una superficie; la irradiancia es máxima cuando el Sol está directamente sobre la superficie ($\theta_z = 0$), y disminuye cuando el Sol está cerca del horizonte debido a la mayor distancia que recorre la atmósfera y a la dispersión.

El factor de excentricidad E_0 produce pequeñas pero notables diferencias estacionales (del orden de $\pm 3\%$) resultantes de los cambios en la distancia Tierra-Sol.

La declinación y los ángulos horarios describen el movimiento diario y anual del Sol, respectivamente, y permiten modelar con precisión la irradiancia en cualquier momento y lugar. Para su proyecto, esta ecuación proporciona la referencia teórica con la que el sistema de IA puede contrastar los valores de irradiancia medidos o obtenidos a partir de imágenes. Permite a la red neuronal escalar sus predicciones en parámetros solares físicamente significativos y garantiza que la optimización de los ángulos de inclinación y azimut se ajuste al comportamiento fundamental de la radiación solar en el espacio y el tiempo.

3.4.2. Corrección de Declinación Solar

El ángulo de declinación solar (δ) cuantifica la posición angular del Sol con respecto al plano ecuatorial de la Tierra y cambia periódicamente cada año debido a la inclinación axial y el movimiento orbital de la Tierra alrededor del Sol. Es uno de los parámetros cruciales en la geometría solar, ya que influye directamente en los ángulos cenital y de elevación solar y, por lo tanto, en la irradiancia solar incidente sobre una superficie terrestre.

Una expresión simplificada para la declinación, que suele ser adecuada para cálculos generales de la posición solar, viene dada por la ecuación 3.8. Si bien esta ecuación proporciona una buena aproximación, se basa en la suposición de una órbita circular y no tiene en cuenta la elipticidad de la órbita terrestre ni la excentricidad orbital con los cambios en el perihelio-afelio. Estos factores generan errores pequeños pero detectables en el cálculo de la posición solar, que pueden acumularse y reducir la precisión de un modelo de irradiancia, especialmente cuando se analizan conjuntos de datos de alta resolución, como imágenes del cielo, con una frecuencia subhoraria.

Estas limitaciones se corrigen mediante una formulación más precisa del ángulo de declinación solar. La siguiente expresión para el ángulo de declinación tiene en cuenta la excentricidad orbital de la Tierra y las perturbaciones en su movimiento orbital:

$$\begin{aligned}\delta = & 0.006918 - 0.399912\cos(G) + 0.070257\sin(G) - \\ & 0.006758\cos(2G) + 0.000907\sin(2G) - \\ & 0.002697\cos(3G) + 0.001480\sin(3G)\end{aligned}\tag{3.14}$$

donde:

$$G = \frac{2\pi(n-1)}{365.2425} \quad (3.15)$$

Esta aproximación extendida del ángulo de declinación solar mediante series de Fourier se basa en datos astronómicos empíricos y reproduce con mayor fidelidad las variaciones periódicas del movimiento aparente del Sol [23]. Cada término de la serie refina la estimación al modelar las sutiles variaciones en la latitud aparente del Sol causadas por la precesión axial y la excentricidad orbital de la Tierra.

La adición de componentes sinusoidales de orden superior permite que el modelo simule la declinación real con un margen de error inferior a 0.01° , en comparación con una desviación máxima de hasta 0.5° en su contraparte simplificada. En combinación con un modelado preciso de la irradiancia solar, esta exactitud garantizará que, en un instante específico, el nivel de irradiancia predicho refleje correctamente las condiciones geométricas solares reales, especialmente en sistemas basados en IA que aprenden a partir de imágenes del cielo con resolución temporal. Con esta fórmula mejorada del ángulo de declinación, la estimación de la posición aparente del Sol a lo largo del año debería ser más fiable y, por consiguiente, más precisa para los modelos de predicción de irradiancia, lo que permitirá una mejor optimización de la orientación de los paneles solares y la producción de energía.

3.4.3. Computando las Ecuaciones y su Muestreo en Gráficas

Para poder visualizar con mayor facilidad los resultados que se obtienen al evaluar las ecuaciones anteriores, se utilizó el programa MatLAB para generar las gráficas siguientes en varias iteraciones del día, y a su vez en cada día del año. Se utilizarán las coordenadas 0,0 de la Tierra para mayor facilidad a la hora de mostrar los cálculos.

El código para computar las anteriores ecuaciones es el siguiente:

```

abs_n = zeros(2400,365);
abs_hour = zeros(2400,365);

for hour = 1:2400
    for n = 1:365
        abs_n(hour,n) = n + hour/2400 - 1;
        abs_hour(hour,n) = hour/100;
    end
end

B = (360/365.2425)*(abs_n - 81);
G = (2*pi*(abs_n - 1))/365.2425;
EoT = 9.87*sin(2*deg2rad(B)) - 7.53*cos(deg2rad(B)) - 1.5*
    sin(deg2rad(B));

Latitude = 0;
Longitude = 0;

%Sol_Dec = 23.45*sin(deg2rad(B)); Less Accurate Formula
Sol_Dec_Acc = rad2deg(0.006918 - 0.399912*cos(G) +
    0.070257*sin(G) - 0.006758*cos(2*G) + 0.000907*sin(2*G)
    - 0.002697*cos(3*G) + 0.001480*sin(3*G));

TC = 4*rem(Longitude,15) + EoT;
LST = abs_hour + (TC/60);
H = 15*(LST-12);

SubAns = sin(deg2rad(Latitude)).*sin(deg2rad(Sol_Dec_Acc))
    + cos(deg2rad(Latitude)).*cos(deg2rad(Sol_Dec_Acc)).*
    cos(deg2rad(H));
Elevation = rad2deg(asin(SubAns));
Elevation(Elevation <= 0) = 0;
Zenith = 90 - Elevation;

```

Código 3.1: Generando el Ángulo de Elevación

Y al utilizar comandos de generación de gráficas con los datos descritos en las mismas, genera las siguientes:

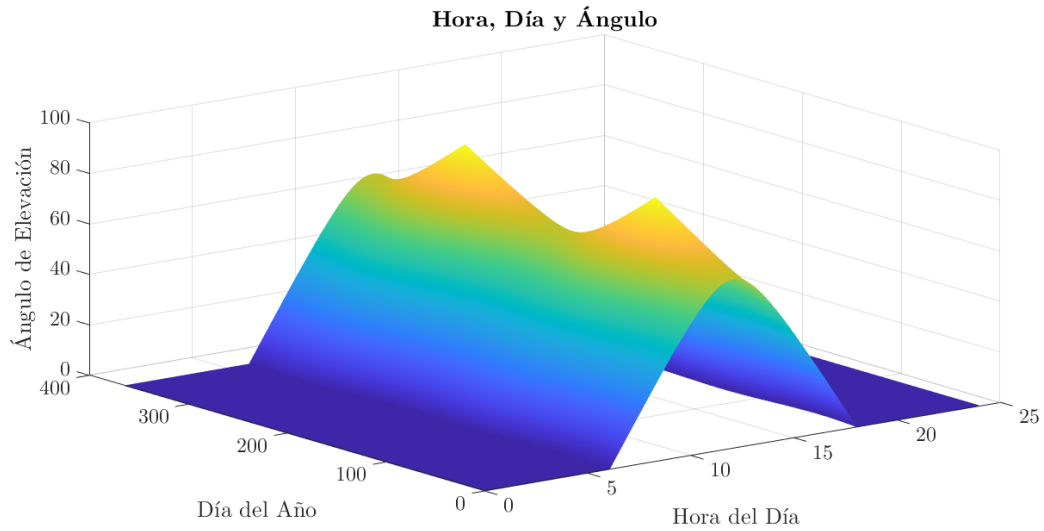


Figura 3.2: Elevación solar durante todo el día, en todo el año

En la imagen 3.2 se puede apreciar que tanto varía el ángulo de elevación en relación al día y a la hora, en ella se puede observar un fenómeno único que ocurre cercano a la línea ecuatorial, y es que dicha franja tiene 2 días largos ubicados en junio y diciembre, y la relación de su día más largo y el más corto es de apenas unos segundos o minutos entre más se alejen del Ecuador.

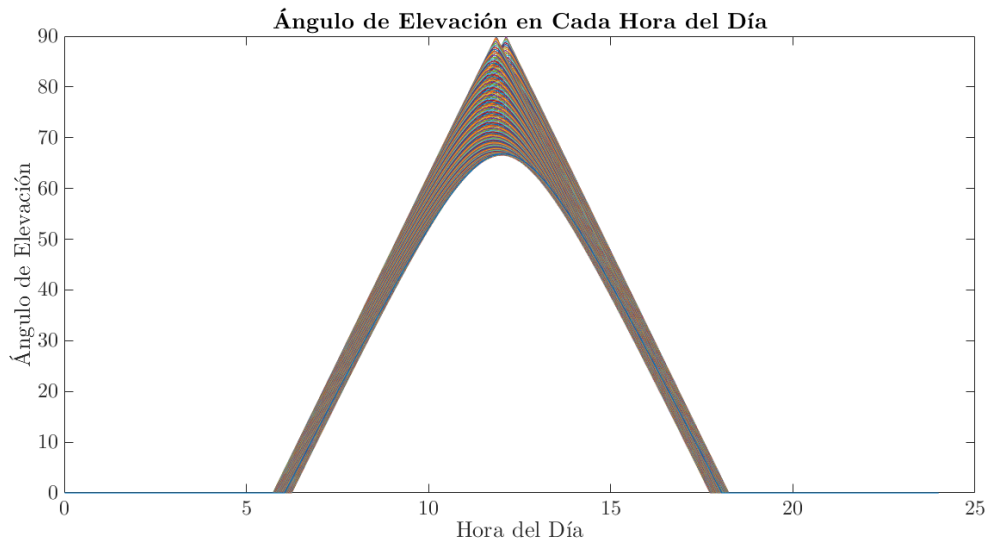


Figura 3.3: Patrón de elevación en un día

En la imagen 3.3 se observa una parte de la gráfica 3.2 pero vista desde el eje de "Hora del Día", en ella se observa que tanto varía la elevación solar a lo largo del día y su grosor representa que tanto cambia a en todo el año, a su vez, se observa la hora del amanecer y la puesta del sol aproximadas y su poca variación en el año.

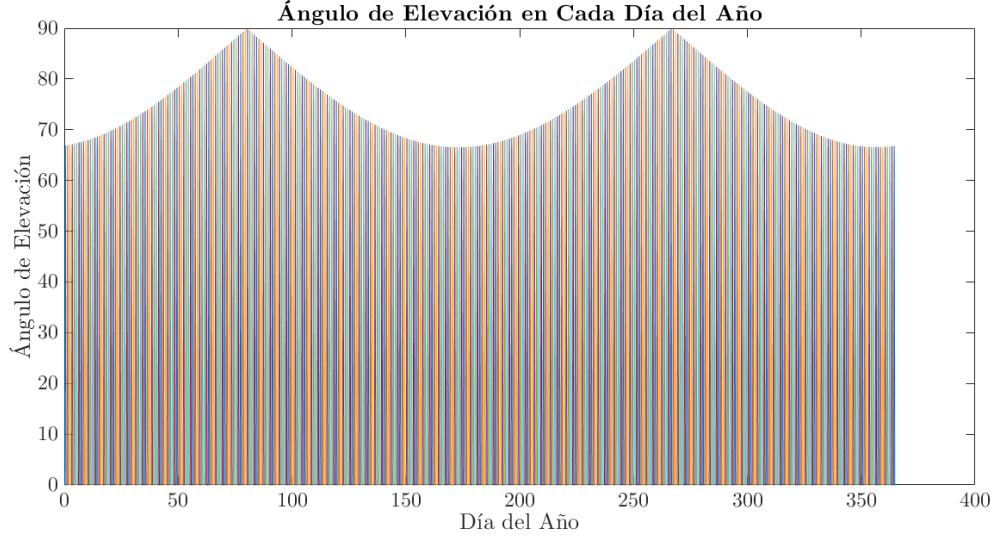


Figura 3.4: Puntos de elevación máxima en el día, durante todo el año

En cambio, en la imagen 3.4 se muestra que tanto varía la elevación en todo el año, ya que la elevación se encuentra en constante cambio en un día, entonces en la gráfica se podrá apreciar la máxima elevación solar que tiene ese día y su registro a lo largo del año.

3.4.4. Ecuación General de la Irradiancia Solar

La fórmula

$$I_D = 1361W/m^2 \left(1 + 0.033 \cos \left(\frac{360^\circ}{365.2425} n \right) \right) \cos(\theta_z) e^{\frac{-\tau}{\cos(\theta_z)}} \quad (3.16)$$

es un modelo matemático general utilizado para estimar la irradiancia solar directa

I_D , es decir, la cantidad total de radiación solar que llega a la superficie terrestre en línea recta desde el Sol, sin ser desviada por la atmósfera. Esta fórmula incorpora tres elementos fundamentales de la modelización de la energía solar: la variación de la distancia Tierra-Sol a lo largo del año, la relación geométrica entre el Sol y el punto de observación, y la atenuación de la luz solar al atravesar la atmósfera. Cada término de la fórmula corresponde a uno de estos procesos físicos fundamentales, por lo que la fórmula es concisa pero expresiva al describir el comportamiento de la radiación solar.

La primera constante, $1361W/m^2$, se conoce como constante solar, la irradiancia solar media sobre una superficie perpendicular a los rayos del Sol fuera de la atmósfera terrestre. Sin embargo, la órbita terrestre no es circular, sino ligeramente elíptica, por lo que la distancia Tierra-Sol varía a lo largo del año. Esto se ajusta mediante el factor de corrección de eccentricidad mostrado en la ecuación 3.6, donde n es el día del año (entre 1 y 365.2425, para tener en cuenta los años bisiestos). Este término aumenta progresivamente la irradiancia cuando la Tierra está más cerca del Sol (a principios de enero) y la reduce cuando la Tierra está más lejos (a principios de julio), lo que introduce una variación estacional aproximada de $\pm 3.3\%$ en la cantidad de radiación solar extraterrestre.

El siguiente componente $\cos(\theta_z)$ es la proyección geométrica de la luz solar sobre una superficie horizontal, donde $\cos(\theta_z)$ es el ángulo cenital solar: el ángulo entre los rayos del Sol y la vertical en un punto y momento. Cuando el Sol está verticalmente sobre la superficie ($\theta_z = 0$, $\cos(\theta_z) = 1$), y la superficie recibe la máxima irradiancia, a medida que el Sol se oculta en el cielo hacia el amanecer o el atardecer, θ_z incrementa y $\cos(\theta_z)$ decrementa, lo que refleja la reducción de la cantidad de luz solar por unidad de área debido a la incidencia oblicua de la luz solar. Este componente geométrico explica la tendencia diurna de la energía solar, un determinante clave de la producción energética diaria en sistemas solares fotovoltaicos.

Finalmente, el término exponencial en la ecuación 2.3 modela el efecto de atenuación atmosférica o transmitancia; es decir, cuánta luz solar es absorbida o dispersada por gases, aerosoles y partículas en la atmósfera antes de llegar a la superficie. El símbolo τ representa la profundidad óptica, una medida de la opacidad de la atmósfera a la radiación solar. Un valor de τ más alto representa condiciones de niebla, contaminación o humedad que atenúan la irradiancia directa. La expresión 2.3 es directamente proporcional a la masa de aire, que es la longitud relativa del recorrido de la luz solar a través de la atmósfera. Cuando el Sol está más cerca del horizonte, la longitud del recorrido aumenta significativamente; es decir, la luz solar debe atravesar más aire y, por lo tanto, está sujeta a una mayor atenuación. El término exponencial se deriva de la ley de Beer-Lambert, la reducción exponencial de la intensidad de la luz al atravesar un medio absorbente.

El coeficiente de atenuación atmosférica (τ) mide la cantidad de radiación solar que se pierde por absorción y dispersión cuando la luz solar atraviesa la atmósfera. Depende de la masa del aire, definida como la longitud relativa del recorrido de la luz solar a través de la atmósfera en comparación con la situación en la que el Sol se encuentra en la parte superior. La masa de aire se calcula aproximadamente mediante la ecuación de Kasten y Young:

$$AM = \frac{1}{\cos(\theta_z) + 0.50572(96.07995 - \theta_z)^{-1.6364}} \quad (3.17)$$

cuya transmitancia atmosférica se modela mediante como:

$$\tau = e^{-kAM} \quad (3.18)$$

donde k es el coeficiente de atenuación atmosférica, y depende de las condiciones atmosféricas, tales como la humedad, contaminación del aire, nubosidad, etc. Los datos

tipicamente se distribuyen (más no se limitan) desde 0.1 (para cielos más despejados) hasta llegar a 0.4 (para días nublados).

Esta ecuación resume la interacción entre la geometría, la mecánica orbital y la física atmosférica para calcular la irradiancia solar directa disponible en cualquier punto de la Tierra y en cualquier época del año. Proporciona un método basado en la física para describir cómo varía la energía solar con parámetros astronómicos (día del año, ángulo solar) y ambientales (transparencia atmosférica). Como modelo de referencia, proporciona un estándar con el que se pueden calibrar métodos basados en datos en las áreas de aplicación de la optimización de la energía solar y el modelado de inteligencia artificial. Las redes neuronales entrenadas con imágenes solares, por ejemplo, pueden aprender a aproximar o corregir las predicciones de este modelo teórico incorporando variabilidad del mundo real, como la nubosidad, la carga de aerosoles y los patrones climáticos locales, que son difíciles de parametrizar directamente.

3.4.5. Computando la Irradiancia Solar

Computando el código en las coordenadas de la colección de imágenes (Latitud 37.43°, Longitud -122.17°), y ahora incluyendo en el código la fórmula para calcular la irradiancia solar en condiciones atmosféricas ideales, se obtiene la figura 3.5.

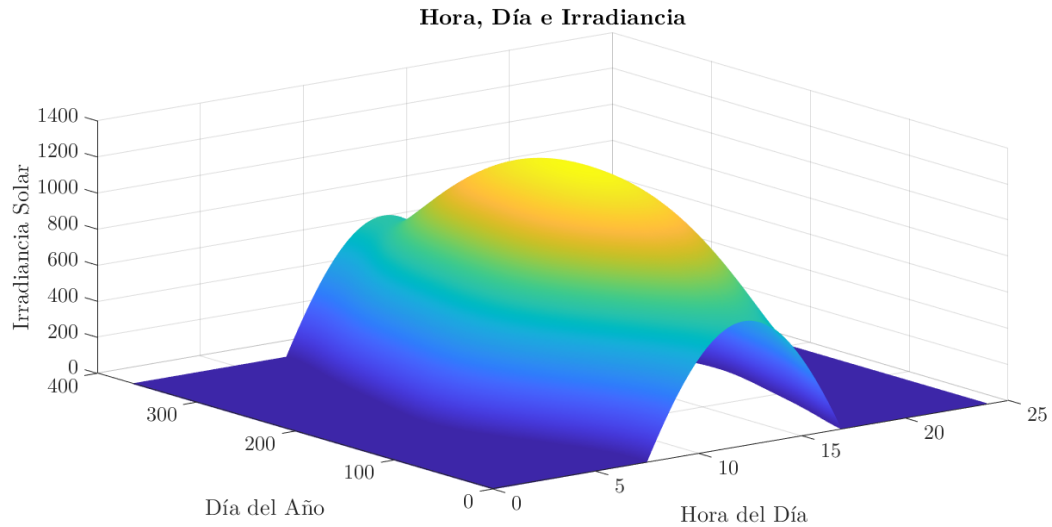


Figura 3.5: Presencia de irradiancia en cada minuto del día durante todo el año

Donde se puede observar el periodo de mayor irradiancia, y asimismo los que tienen menor irradiancia. Esto se logra añadiendo las siguientes líneas de código al programa antes propuesto:

```
Tau = 0; %Perfect Conditions
Irradiance = 1361*(1 + 0.033.*cos(deg2rad((360/365.2425).*
    abs_n))).*cos(deg2rad(Zenith)).*exp(-Tau./cos(deg2rad(
    Zenith)));
```

Código 3.2: Añadiendo la Ecuación de Irradiancia

Igualmente, al escribir las instrucciones para generar una figura, junto con los datos de los días, la hora, y la irradiancia, se obtiene una gráfica tal como 3.5.

3.5. Determinando las Condiciones Atmosféricas Usando Inteligencia Artificial

La IA se ha convertido en una herramienta clave para la estimación y predicción más precisa y rápida de las condiciones atmosféricas que los métodos tradicionales. Gracias a su capacidad para procesar grandes volúmenes de datos históricos y en tiempo real, los algoritmos de IA pueden detectar patrones ocultos en la evolución del clima, la distribución de aerosoles, la nubosidad y la humedad atmosférica, entre otros parámetros que influyen directamente en la radiación solar.

Uno de los principales enfoques es el entrenamiento de modelos de aprendizaje automático, como redes neuronales profundas, máquinas de vectores de soporte o bosques aleatorios, utilizando datos de estaciones meteorológicas, satélites, teledetección y simulaciones atmosféricas. Los algoritmos aprenden las complejas relaciones no lineales entre las variables climáticas y, posteriormente, pueden predecir parámetros como la transmitancia atmosférica, la opacidad del cielo o la probabilidad de nubosidad en un futuro determinado.

En los sistemas solares, por ejemplo, la IA puede predecir las variaciones diarias de la irradiancia basándose no solo en la geometría solar, sino también en la atenuación debida a las condiciones atmosféricas locales. Esto permite optimizar dinámicamente los sistemas solares para aumentar su eficiencia o planificar la producción de energía en las redes eléctricas según las fuentes renovables. Además, en regiones donde los datos meteorológicos son inexistentes o de baja calidad, la IA puede predecir y extrapolar datos, completando la información faltante con predicciones que mantienen altos índices de precisión.

El uso de la IA también está revolucionando los grandes modelos climáticos. Los

pronósticos basados en aprendizaje profundo, incorporados a estos modelos, permiten una mejor simulación del comportamiento atmosférico a corto y mediano plazo, lo que repercute positivamente en sectores como la agricultura, el transporte aéreo y la gestión de desastres naturales. Los sistemas inteligentes integran cada vez más los modelos físicos tradicionales con programas de predicción automatizados, generando así herramientas híbridas que aprovechan las ventajas de ambos enfoques: la base teórica y la capacidad de la IA para aprender y adaptarse a nuevos datos o comportamientos atmosféricos no predecibles.

Por lo tanto, la IA no solo acelera la predicción de las condiciones atmosféricas, sino que también amplía la escala y la variedad de aplicaciones que dependen del conocimiento del comportamiento solar y climático.

3.6. Ubicación Ideal de Granjas de Paneles Solares

De acuerdo a la información anterior, la ubicación ideal de las redes de paneles solares depende de una combinación de condiciones geográficas, climáticas y ambientales que inciden directamente en la cantidad y uniformidad de la irradiación solar recibida durante el año. En teoría, las ubicaciones óptimas se encuentran cerca de las regiones ecuatoriales de la Tierra, típicamente entre las latitudes 15° Norte y 15° Sur, ya que estas reciben la radiación solar más directa y uniforme durante todo el año debido al ángulo de incidencia solar casi perpendicular. La trayectoria solar se mantiene alta en el cielo en estos lugares, con menos pérdidas por dispersión atmosférica y reflexión. Esto se traduce en una mayor irradiación promedio y, en consecuencia, en un mayor rendimiento energético potencial por metro cuadrado de paneles instalados. Además, estas áreas suelen experimentar una variación estacional mínima en el ángulo solar, lo que reduce la necesidad de sistemas de seguimiento complejos para lograr la máxima inclinación.

Sin embargo, la selección del emplazamiento de un parque solar no se limita a la latitud. Los desiertos montañosos y las regiones áridas, como el desierto de Atacama en Chile, el desierto del Sahara en África o partes del suroeste de Estados Unidos, también son deseables. Estos emplazamientos presentan una alta irradiancia y una baja nubosidad y humedad atmosférica, lo que minimiza la dispersión y absorción de la radiación solar. La falta de vegetación y precipitaciones minimiza aún más el costo de mantenimiento debido al polvo, la corrosión y el sombreado de la vegetación cercana. Además, estas regiones suelen tener patrones climáticos constantes que hacen que la producción de energía sea más predecible y la conexión a la red eléctrica más fiable.

La orografía local y la claridad atmosférica también son factores importantes. Las mayores altitudes pueden ofrecer un aire más limpio y una atmósfera más delgada, lo que mejora la irradiancia efectiva sobre los paneles. Las regiones con baja concentración de aerosoles y baja contaminación atmosférica presentan una radiación difusa menor y una irradiancia normal directa mayor, lo que beneficia a los sistemas de CSP en general. Por otro lado, para los sistemas solares fotovoltaicos, incluso una combinación de luz solar directa y difusa puede producir mucha energía, y las ubicaciones con poca nubosidad, por ejemplo, el Mediterráneo o las regiones subtropicales, pueden seguir funcionando si las plantas solares se adaptan óptimamente al entorno.

Además de los factores ambientales, la infraestructura y la distribución de la energía son factores importantes para el éxito de la implementación. La proximidad a líneas de transmisión accesibles, centros urbanos o zonas industriales puede reducir las pérdidas de energía en el transporte y el costo de instalación. Las zonas con un buen apoyo político a las energías renovables, terrenos adecuados y bajas presiones competitivas en el uso del suelo (como desiertos o tierras recuperadas del mar) aumentan aún más la viabilidad. En definitiva, el despliegue ideal de las redes solares reside en la combinación de la intensa irradiación solar, la claridad atmosférica, la estabilidad geográfica y el acceso logístico, lo que convierte a las regiones ecuatoriales, los desiertos montañosos y

las llanuras secas y soleadas en, en teoría, los lugares más óptimos para la generación de energía solar a gran escala.

Capítulo IV - Extracción de Datos Previos al Entrenamiento

4.1. Extracción de Datos de las Imágenes Usando Python

Para extraer valores de irradiancia solar y otras variables atmosféricas del conjunto de datos Sky Images de la Universidad de Stanford (2017), se ha diseñado un sistema de procesamiento de datos que utiliza las bibliotecas Python, OpenCV, NumPy y Pandas. El objetivo principal de este sistema de procesamiento de datos es proporcionar imágenes precisas del cielo, que podrían transformarse en funciones para entrenar un sistema de aprendizaje automático para modelar los valores de irradiancia solar. Las imágenes del cielo son esencialmente vistas de cúpulas celestes hemisféricas en un momento determinado, de las cuales se podrían extraer características para identificar las condiciones radiativas y geométricas de la atmósfera que afectan la irradiancia solar.

4.1.1. Preprocesamiento de Imágenes

Primero, cada imagen se pasa para su análisis usando OpenCV, luego se convierte a una imagen en escala de grises debido a la captura del cielo circular contenida en un marco rectangular, donde los píxeles más allá de la región circular de interés debían ocultarse para evitar anomalías en los valores de intensidad en esos píxeles. Se aplica una máscara circular binaria a una imagen en el centro

$$(x_c, y_c)$$

con un radio de r de la siguiente manera:

$$M(x, y) = \begin{cases} 1, & \text{si } (x - x_c)^2 + (y - y_c)^2 \leq r^2 \\ 0, & \text{de lo contrario} \end{cases} \quad (4.1)$$

El mapa del cielo elimina los píxeles de datos del cielo no válidos, eliminando así cualquier sesgo potencial de los datos del cielo que en realidad son datos de fondo.

4.1.2. Estimación de la Irradiancia Solar

La imagen en escala de grises $I(x, y)$ se normaliza a un valor entre $[0, 1]$ y se multiplica por la constante solar $I_o = 1361 \text{ W/m}^2$, que es un valor promedio de la irradiancia solar extraterrestre:

$$I_D(x, y) = I_o \cdot I(x, y) \quad (4.2)$$

Una estimación de la irradiancia solar total en el momento de esta imagen se determina tomando la irradiancia media de todos los píxeles que caen bajo la máscara circular [7] [15]:

$$\overline{I_D} = \frac{1}{N} \sum_{(x,y) \in M} I_D(x, y) \quad (4.3)$$

donde N es el número total de píxeles del cielo válidos.

4.1.3. Intensidad Ponderada

Para comprender mejor la distribución local del brillo en función de la ubicación del Sol, se calculó un valor de irradiancia media ponderada, que dio mayor prioridad a los píxeles alrededor del disco solar. Los pesos $w(x, y)$ disminuyen en función de la distancia d desde la ubicación central del Sol (x_s, y_s) :

$$w(x, y) = e^{-\frac{d^2}{2\sigma^2}}, \quad d = \sqrt{(x - x_s)^2 + (y - y_s)^2} \quad (4.4)$$

Entonces, la intensidad ponderada se definiría como:

$$I_W = \frac{\sum_{(x,y) \in M} w(x, y) \cdot I_D(x, y)}{\sum_{(x,y) \in M} w(x, y)} \quad (4.5)$$

Esto se debe a que se concentra más en la irradiancia de las regiones cercanas al centro solar, que es más resistente a la luz solar dispersa y a las obstrucciones de las nubes.

4.1.4. Coordenadas del Centro Solar

La ubicación aparente del Sol en una imagen se determina encontrando el conjunto de píxeles más brillantes dentro de esta región enmascarada, lo que se logra encontrando el 2,5 % superior de los píxeles por intensidad y luego tomando el centro de esos píxeles:

$$x_s = \frac{\sum_i x_i \cdot I(x_i, y_i)}{\sum_i I(x_i, y_i)}, \quad y_s = \frac{\sum_i y_i \cdot I(x_i, y_i)}{\sum_i I(x_i, y_i)} \quad (4.6)$$

El motivo es garantizar que ni los píxeles brillantes atípicos, como los reflejos o la iluminación artificial, tengan mucho peso ni comprometan la detección precisa de la

ubicación del centro del disco solar.

4.1.5. Estimación de la Cobertura de Nubes

La cantidad de cobertura de nubes se calculó a partir de variaciones de intensidad dentro de la región de la máscara del cielo [12]. Las áreas de intensidad alta y constante generalmente correspondían a regiones de cielo despejado, mientras que las grandes variaciones indicaban nubes. El porcentaje de cobertura de nubes C se estimó mediante la fórmula:

$$C = 1 - \frac{\sigma_I}{\sigma_{I,\max}} \quad (4.7)$$

donde σ_I es la desviación estándar de la intensidad en la región válida y $\sigma_{I,\max}$ es la varianza máxima en condiciones claras. Esto produce valores entre 0 (cielo despejado) y 1 (nublado).

4.1.6. Varianza Óptica

La varianza óptica mide la anisotropía de la luz que se dispersa en la cúpula del cielo, que refleja la turbidez atmosférica. El valor de la varianza óptica se calcula a partir de la siguiente fórmula:

$$V_{\text{opt}} = \frac{\sum_{(x,y) \in M} (I_D(x,y) - \overline{I_D})^2}{N \cdot \overline{I_D}^2} \quad (4.8)$$

Los valores más grandes indican sistemas de nubes más compactos o una mayor carga de aerosoles, los cuales influyen en la irradiancia directa.

4.1.7. Índice de Uniformidad del Brillo Atmosférico

Para evaluar la claridad y la propiedad de dispersión de la atmósfera a partir de imágenes del cielo, se ha formulado un índice denominado Índice de uniformidad de brillo atmosférico (A_{atm}). Este índice calcula, como resultado, la relación entre la irradiancia total de todos los píxeles del cielo y la irradiancia máxima presente en esta imagen del cielo tomada por un satélite. A_{atm} está formulado de la siguiente manera de acuerdo a la ecuación 2.4, donde I_D representa la irradiancia media en la máscara circular del cielo, $I_{D,Max}$ generalmente está vinculado a valores como la irradiancia del disco solar, que suele estar en su máxima intensidad, presumiblemente en el lugar.

Este valor es un parámetro cuantitativo adimensional de la transparencia atmosférica y la distribución de la irradiancia difusa. En tiempo despejado, el brillo del disco solar es mucho mayor que el del cielo circundante, por lo que A_{atm} se acerca a 0, lo que indica valores de irradiancia más altos pero baja difusión, mientras que en tiempo nublado, el cielo circundante irradia más brillante, lo que hace que I_D sea más cercano a $I_{D,Max}$, que muestra valores de A_{atm} más altos (cerca de 1) para valores más altos de nubes en la atmósfera, más irradiancia, así como difusión debido a que la atmósfera de la Tierra dispersa los rayos del sol.

Como resultado, A_{atm} funciona como un componente atmosférico compacto y eficiente basado en imágenes del cielo, que refleja la fracción relativa de la irradiancia solar total de las contribuciones directas y difusas del cielo. Esto contribuye aún más a aumentar las capacidades de los algoritmos de predicción de la irradiancia relacionados con las condiciones del cielo, lo cual es imperativo en los procesos de optimización de la irradiancia.

4.1.8. Extracción de Metadatos Temporales

Los valores de marca de tiempo para cada nombre de archivo de imagen aparecen en el formato YYYYMMDDHHMMSS en todos los casos. El script extrae esta información como variables numéricas para año, mes, día, hora, minuto y segundo, que desempeñan un papel fundamental al relacionar características extraídas ópticamente con fórmulas relacionadas con la geometría solar [22], incluida la declinación, el ángulo horario y los valores del ángulo cenital. Estos parámetros serán útiles para calcular la irradiancia por medio de la fórmula 3.16 en condiciones ideales para poder entrenar la red neuronal. Dicho código para extraer esos datos es el siguiente:

```
import os
import cv2
import numpy as np
import pandas as pd
from scipy.stats import skew

def extract_image_features(image_path, circle_center=None,
                           circle_radius=None):

    image = cv2.imread(image_path)
    if image is None:
        raise ValueError("Image not found or could not be
                           loaded.")

    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
    h, w = gray.shape

    if circle_center is None:
        circle_center = (w // 2, h // 2)
    if circle_radius is None:
        circle_radius = (min(h, w) // 2) - 102

    Y, X = np.ogrid[:h, :w]
    dist_from_center = np.sqrt((X - circle_center[0])**2 +
                                (Y - circle_center[1] + 102)**2)
    mask = dist_from_center <= circle_radius

    normalized = gray / 255.0
```

```

irradiance_map = normalized * 1361
masked = np.where(mask, irradiance_map, np.nan)

intensity_img = np.nan_to_num(masked, nan=0.0)
pct = 97.5
try:
    thresh_val = np.nanpercentile(intensity_img[mask],
                                   pct)
except ValueError:
    thresh_val = np.nanmax(intensity_img)

bright_mask = (intensity_img >= thresh_val) & mask
bright_uint8 = (bright_mask.astype(np.uint8) * 255)
num_labels, labels = cv2.connectedComponents(
    bright_uint8)

if num_labels <= 1:
    _, _, _, maxLoc = cv2.minMaxLoc(gray)
    sun_x, sun_y = maxLoc
else:
    max_label = max(range(1, num_labels), key=lambda
        lbl: np.sum(labels == lbl))
    ys, xs = np.where(labels == max_label)

    if xs.size == 0:
        _, _, _, maxLoc = cv2.minMaxLoc(gray)
        sun_x, sun_y = maxLoc
    else:
        comp_intensities = intensity_img[ys, xs]
        total_int = np.sum(comp_intensities)
        if total_int == 0:
            sun_x = int(np.mean(xs))
            sun_y = int(np.mean(ys))
        else:
            sun_x = float(np.sum(xs * comp_intensities
                                   ) / total_int)
            sun_y = float(np.sum(ys * comp_intensities
                                   ) / total_int)

valid_pixels = masked[mask & ~np.isnan(masked)]

mean_irradiance = np.nanmean(valid_pixels)
weighted_intensity = np.nanmean(valid_pixels**2) / np.
    nanmean(valid_pixels)
cloud_coverage = np.sum(valid_pixels < (0.6 * np.
    nanmax(valid_pixels))) / np.sum(mask)
optical_variance = np.nanvar(valid_pixels)

```

```

atmospheric_indicator = np.nanmean(valid_pixels) / np.
    nanmax(valid_pixels)

return {
    "mean_irradiance": mean_irradiance,
    "weighted_intensity": weighted_intensity,
    "cloud_coverage": cloud_coverage,
    "optical_variance": optical_variance,
    "sun_X": sun_x,
    "sun_Y": sun_y,
    "atmospheric_indicator": atmospheric_indicator
}

def extract_timestamp_from_filename(filename):
    base = os.path.splitext(filename)[0]
    return {
        "Filename": filename,
        "Year": int(base[0:4]),
        "Month": int(base[4:6]),
        "Day": int(base[6:8]),
        "Hour": int(base[8:10]),
        "Minute": int(base[10:12]),
        "Second": int(base[12:14])
    }

def process_image_folder(folder_dir, output_base_name):
    all_records = []
    file_count = 96
    chunk_size = 1000

    for i, filename in enumerate(os.listdir(folder_dir)):
        if filename.lower().endswith(".jpg"):
            image_path = os.path.join(folder_dir, filename)
            try:
                time_data =
                    extract_timestamp_from_filename(
                        filename)
                features = extract_image_features(
                    image_path)

                record = {**time_data, **features}
                all_records.append(record)

                if len(all_records) == chunk_size:
                    file_count += 1
                    output_csv = f"{output_base_name}_{

```

```

        file_count }.csv"
    df = pd.DataFrame(all_records)
    df.to_csv(output_csv, index=False)
    print(f"Saved {len(all_records)}
          records to {output_csv}")
    all_records = [] # <- RESET

except Exception as e:
    print(f"Error processing {filename}: {e}")

if all_records:
    file_count += 1
    output_csv = f"{output_base_name}_{file_count}.csv"

    df = pd.DataFrame(all_records)
    df.to_csv(output_csv, index=False)
    print(f"Saved remaining {len(all_records)} records
          to {output_csv}")

print(f"All data saved into {file_count} CSV files.")

folder_dir = "PATH"
output_base_name = "irradiance_features"
process_image_folder(folder_dir, output_base_name)

```

Código 4.1: Extracción de Datos

4.1.9. Estructuración y Almacenamiento de Datos

Todos los valores calculados, como el tiempo, la irradiancia media, la intensidad ponderada, la cobertura de nubes, la varianza óptica, las coordenadas del Sol y las variables atmosféricas, se organizaron en un DataFrame organizado utilizando la biblioteca Pandas para fines de gestión eficiente de datos, ya que todos los datos se almacenan en varios archivos .csv que constan de 1000 características para cada imagen, lo que es útil para una fácil preparación para ingresar a una red neuronal.

Todos esos conjuntos de datos finalmente se fusionan en un archivo .csv para eli-

minar la llamada a todos los archivos individualmente, lo que aumenta el tiempo de procesamiento. Dicho código para fusionar los archivos en uno solo delimitado por comas es el siguiente:

```
import pandas as pd
import glob
import os

joined_files = os.path.join("PATH", "irradiance_features_*.csv")

joined_list = glob.glob(joined_files)

df = pd.concat(map(pd.read_csv, joined_list), ignore_index=True)
df.to_csv("MergedData.csv", index=False)
```

Código 4.2: Uniendo los Archivos

4.2. Estimación de la Posición Solar

El cálculo de la posición aparente del Sol en el cielo se puede realizar mediante un método vectorial que describe su dirección utilizando un vector unitario en un espacio tridimensional. Este formalismo matemático garantiza transformaciones de coordenadas precisas y, por lo tanto, sirve de base para obtener los puntos de proyección en un sistema de coordenadas cartesianas bidimensional, como el plano de imagen de una lente ojo de pez o la representación gráfica de un mapa solar.

4.2.1. Posición del Sol como un Conjunto de Vectores

Unitarios 3D

La posición aparente del Sol se describe generalmente mediante el acimut y la elevación solar. Estos dos ángulos se definen por la dirección de la luz solar incidente con respecto a un punto de observación en la Tierra:

- Ángulo de elevación solar: visibilidad del Sol desde el horizonte (de 0° = horizonte a 90° = cenit).
- Ángulo de acimut solar: ángulo medido en sentido horario desde el norte geográfico (u otra dirección de referencia, según la convención).

Para convertir estas coordenadas angulares en un vector unitario, se puede emplear el método estándar para convertir coordenadas esféricas a cartesianas. Este vector resultante describe la dirección hacia el Sol en un sistema de coordenadas tridimensional [1] [20], donde:

- El eje x apunta al Este.
- El eje y apunta al Norte.
- El eje z es positivo y apunta hacia el cenit.

El vector unitario S se puede escribir como:

$$S = (S_x, S_y, S_z) \tag{4.9}$$

con componentes:

$$\begin{aligned}
S_x &= \cos(\phi_s) \sin(\lambda_s - \lambda_o) \\
S_y &= \cos(\phi_o) \sin(\phi_s) - \sin(\phi_o) \cos(\phi_s) \cos(\lambda_s - \lambda_o) \\
S_z &= \sin(\phi_o) \sin(\phi_s) - \cos(\phi_o) \cos(\phi_s) \cos(\lambda_s - \lambda_o)
\end{aligned} \tag{4.10}$$

donde:

- ϕ_s es la latitud del punto subsolar, equivalente a la declinación solar.
- λ_s es la longitud del punto subsolar, equivalente a -H dado por la ecuación 3.9.
- ϕ_o es la latitud del observador.
- λ_o es la longitud del observador.

Este vector tiene magnitud unitaria:

$$\sqrt{S_x^2 + S_y^2 + S_z^2} = 1 \tag{4.11}$$

Esto garantiza que indique únicamente la dirección hacia el Sol y no su intensidad.

4.2.2. Interpretación de las Componentes del Vector Unitario

El significado de cada componente de este vector unitario es evidente:

- S_x es la dirección del Sol en el eje Este-Oeste.
- S_y es la dirección en el eje Norte-Sur.
- S_z representa el componente vertical, que describe la altura del Sol sobre el horizonte.

Dado que S_x , S_y y S_z dependen de funciones trigonométricas que involucran azimut y elevación, esto garantiza que dichas funciones se comporten de manera continua y

uniforme a lo largo del día y del año gracias a la representación mediante vectores unitarios. Este método resulta muy útil para implementar algoritmos de seguimiento solar, radiación y proyección geométrica, ya que la manipulación de vectores es más conveniente y menos propensa a errores en comparación con las evaluaciones que implica este método.

4.2.3. Corrección de la Proyección Solar

En este punto, solo queda un paso por realizar: adaptar la proyección solar a la trayectoria del sol a lo largo de dichas coordenadas. Esto se logra mediante la rotación de las matrices sobre los ejes del vector unitario.

Primero, se rota el eje x según el negativo de la latitud, utilizando las siguientes fórmulas:

$$\begin{aligned} S'_y &= S_y \cos(-\phi_o) - S_z \sin(-\phi_o) \\ S'_z &= S_y \sin(-\phi_o) + S_z \cos(-\phi_o) \end{aligned} \tag{4.12}$$

Luego, se rota el eje y según la longitud:

$$\begin{aligned} S'_x &= S_x \cos(\lambda_o) + S_z \sin(\lambda_o) \\ S'_z &= -S_x \sin(\lambda_o) + S_z \cos(\lambda_o) \end{aligned} \tag{4.13}$$

Finalmente, se rota el eje x según la latitud:

$$\begin{aligned} S'_y &= S_y \cos(\phi_o) - S_z \sin(\phi_o) \\ S'_z &= S_y \sin(\phi_o) + S_z \cos(\phi_o) \end{aligned} \tag{4.14}$$

Si se ejecutan las gráficas de los datos obtenidos en un intervalo de 1 hora por 24

horas y cada día durante todo el año se obtiene la figura 4.1

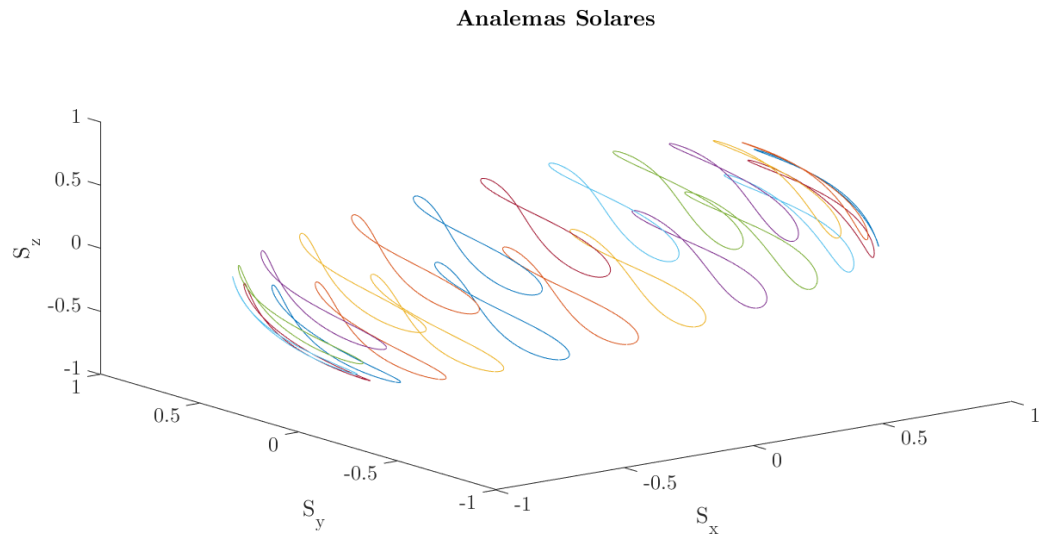


Figura 4.1: Analemas solares en un punto, por intervalos de 1 hora los 365 días del año

Y al momento de sesgar los datos en los cuales la elevación es menor a 0° se obtiene la figura 4.2

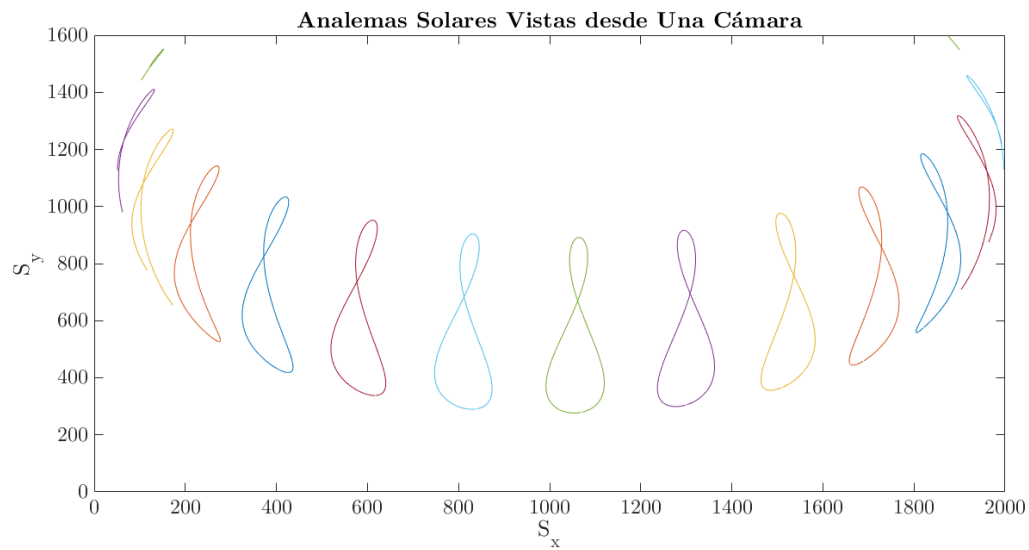


Figura 4.2: Analemas solares vistas desde un punto elevado, con la limpieza de datos carentes de luz solar

4.2.4. Proyección Sobre un Plano 2D

Ahora que la posición solar se puede describir mediante un vector 3D, es posible proyectar esta información sobre una superficie bidimensional. Esto es fundamental para relacionar la posición del Sol con el plano de imagen de un sistema de cámara con lente ojo de pez.

Para una proyección cartesiana simple, se omite la parte vertical S_z , dejando solo las partes horizontales:

$$x = S_x, y = S_y \quad (4.15)$$

4.2.5. Eliminación de Datos Innecesarios

El ángulo α especifica la altura del Sol sobre el horizonte. Un valor positivo indica que el Sol está sobre el horizonte y, por lo tanto, es visible. Un valor negativo indica que el Sol está bajo el horizonte y, por lo tanto, no contribuye a la radiación detectable y no es visible. Los conjuntos de datos que emplean simulación de irradiancia solar, evaluación de enmascaramiento de nubes o redes neuronales artificiales ejecutan esta función de filtrado:

$$\begin{aligned} \alpha > 0^\circ, (S_x, S_y) &= S_x, S_y \\ \alpha < 0^\circ, (S_x, S_y) &= NaN, NaN \end{aligned} \quad (4.16)$$

La ecuación del vector unitario permite obtener un método preciso y optimizado para calcular la posición aparente del Sol en el espacio. Utilizando los valores de acimut y elevación solar para obtener un vector de dirección y proyectar sus componentes horizontales sobre un plano cartesiano, es posible analizar la representación bidimensional

de la trayectoria solar con un alto grado de estabilidad numérica y precisión geométrica. Este método es muy popular en la tecnología de seguimiento solar, las simulaciones ambientales y el procesamiento de imágenes del cielo [1].

4.3. Integración de los Resultados de la Ecuación al Dataset

Para integrar todos los parámetros obtenidos en base a las ecuaciones anteriores se utilizará el siguiente código:

```
clc
clear
close all

noImgs = 131382;
t = 1:noImgs;

Latitude = 37.43;
Longitude = -122.17;

extract = readtable('MergedData.csv');
SortExtract = sortrows(extract);
main = table2array(removevars(SortExtract, 'Filename'));

Year = main(:,1);
Month = main(:,2);
Day = main(:,3);
Hour = main(:,4);
Minute = main(:,5);
Second = main(:,6);

SunX = main(:,11);
SunY = main(:,12);

abs_n = zeros(noImgs,1);
for j = 1:noImgs
    switch Month(j)
        case 1
            abs_n(j) = 0;
```

```

        case 2
            abs_n(j) = 31;
        case 3
            abs_n(j) = 59;
        case 4
            abs_n(j) = 90;
        case 5
            abs_n(j) = 120;
        case 6
            abs_n(j) = 151;
        case 7
            abs_n(j) = 181;
        case 8
            abs_n(j) = 212;
        case 9
            abs_n(j) = 243;
        case 10
            abs_n(j) = 273;
        case 11
            abs_n(j) = 304;
        case 12
            abs_n(j) = 334;
    end
    if (Year(j) == 1) && (Month(j) > 2)
        abs_n(j) = abs_n(j) + 1;
    end
end

abs_hour = Hour + Minute/60 + Second/3600;
abs_n = abs_n + Day - 1 + abs_hour/24;

B = (360/365.2425)*(abs_n - 81);
EoT = 9.87*sin(2*deg2rad(B)) - 7.53*cos(deg2rad(B)) - 1.5*
    sin(deg2rad(B));
TC = 4*rem(Longitude,15) + EoT;
LST = abs_hour + (TC/60);
H = 15*(LST-12);
LonSubP = deg2rad(-H);

G = (2*pi*(abs_n - 1))/365.2425;
LatSubP = (0.006918 - 0.399912*cos(G) + 0.070257*sin(G) -
    0.006758*cos(2*G) + 0.000907*sin(2*G) - 0.002697*cos(3*
    G) + 0.001480*sin(3*G));

CommonLon = LonSubP - deg2rad(Longitude);

Sx = cos(LatSubP).*sin(CommonLon);

```

```

Sy = cos(deg2rad(Latitude)).*sin(LatSubP) - sin(deg2rad(
    Latitude)).*cos(LatSubP).*cos(CommonLon);
Sz = sin(deg2rad(Latitude)).*sin(LatSubP) + cos(deg2rad(
    Latitude)).*cos(LatSubP).*cos(CommonLon);

Sol_Dec = rad2deg(LatSubP);
SubElevationAns = sin(deg2rad(Latitude)).*sin(LatSubP) +
    cos(deg2rad(Latitude)).*cos(LatSubP).*cos(deg2rad(H));
Elevation = rad2deg(asin(SubElevationAns));
Elevation(Elevation <= 0) = 0;
Zenith = 90 - Elevation;

Tau = 0; %Perfect Conditions
IrradianceEq = 1361*(1 + 0.033.*cos(deg2rad((360/365.2425)
    .*abs_n))).*cos(deg2rad(Zenith)).*exp(-Tau./cos(deg2rad
    (Zenith)));

%Rotate by x Axis
RotX = deg2rad(-Latitude);
SyN = Sy*cos(RotX) - Sz*sin(RotX);
SzN = Sy*sin(RotX) + Sz*cos(RotX);
Sy = SyN;
Sz = SzN;

%Rotate by y Axis
RotY = deg2rad(Longitude);
SxN = Sx*cos(RotY) + Sz*sin(RotY);
SzN = -Sx*sin(RotY) + Sz*cos(RotY);
Sx = SxN;
Sz = SzN;

%Rotate by x Axis
RotX = deg2rad(Latitude);
SyN = Sy*cos(RotX) - Sz*sin(RotX);
SzN = Sy*sin(RotX) + Sz*cos(RotX);
Sy = SyN;
Sz = SzN;

Resolution = [2048,2048];
RadReduction = 102;
Offset = 102;

RotateFactor = deg2rad(209.1); % Final Angle of 209.1
Temp_Sx = Sx.*cos(RotateFactor) - Sy.*sin(RotateFactor);
Temp_Sy = Sx.*sin(RotateFactor) + Sy.*cos(RotateFactor);

Temp_Sx = Temp_Sx*((Resolution(1) - RadReduction)/2) +

```

```

    Resolution(1)/2;
Temp_Sy = Temp_Sy*((Resolution(2) - RadReduction)/2) +
    Resolution(2)/2 + Offset;

Sx = Temp_Sx;
Sy = Temp_Sy;

SortExtract.IrradianceEq = IrradianceEq;
SortExtract.Sx = Sx;
SortExtract.Sy = Sy;

writetable(SortExtract, 'PATH\FinalTrainData.csv')

```

Código 4.3: Aumentando la Precisión de la Red Incluyendo Datos Computados

Código cuyo propósito tiene de analizar el archivo .csv obtenido del programa de Python de extracción de datos, extraer los datos de tiempo, y computarlos en las ecuaciones antes descritas para añadir un dato preciso en base a una ecuación, lo cual hará que los datos de predicción puedan ser más precisos [1].

Capítulo V - Elaboración de las Redes Neuronales

5.1. Desglose de la Red Neuronal Convolutacional

Para elaborar la red neuronal convolutacional, se utilizó el software de Python utilizando las librerías de TensorFlow, ya que esta librería facilita la elaboración de la red neuronal [26]. Se toman en cuenta los siguientes pasos para elaborarla:

1. Establecer todas las librerías a utilizar, tales como manipulación de matrices, de directorios de archivos, de los propios archivos, así como la librería de la red neuronal.
2. Escribir los directorios del DataSet, los valores del mismo, y donde se escribirán los archivos resultantes junto con su nombre (en este último se usó el directorio por defecto).
3. Establecer los parámetros modificables por el usuario.
4. Poner las columnas objetivo.
5. Ingresar opciones de precisión de la red.
6. Verificar si existen los directorios.
7. Leer el archivo de la data de las imágenes.
8. Verificar que exista el archivo de imagen en el archivo de datos.
9. Establecer los datos de entrenamiento, los de prueba, y los archivos pertenecientes a ellos.

10. Establecer funciones para estandarizar datos.
11. Establecer función de preprocesamiento de imágenes cambiándolas de tamaño y convirtiéndolas en matrices.
12. Establecer una función para obtener los datasets.
13. Ejecutar dichas funciones al llamar a hacer los datasets.
14. Construir la red neuronal de acuerdo a los parámetros de TensorFlow, con la arquitectura propuesta por el usuario.
15. Establecer parámetros de optimización de recursos, ya que la PC de entrenamiento requiere de un menor uso de RAM y de GPU.
16. Llamar el modelo de la red neuronal con el número de épocas de 200, junto con parámetros dados por TensorFlow.
17. Guardar el registro de error por época.
18. Guardar el archivo del modelo de la red neuronal, junto con los pesos de las neuronas.
19. Escribir las predicciones en una matriz llamando a la red neuronal y el archivo recién generado de .keras.
20. Guardar finalmente el archivo de las predicciones en un .csv para su posterior muestreo de gráficas [13].

Y el código elaborado resultó en el siguiente:

```
import os
import math
import csv
import argparse
import numpy as np
import pandas as pd
```

```

import tensorflow as tf
from tensorflow.keras import layers, models, callbacks,
    optimizers, losses, metrics
from tensorflow.keras.applications import EfficientNetB0

IMAGE_DIR = "D:/CNNImagenesComprimidas/"
LABELS_CSV = "FinalTrainData.csv"
MODEL_SAVE_PATH = "cnn_solar_efficientnetFINAL.keras"
PRED_CSV = "cnn_predicted_parametersFINAL.csv"

IMAGE_SIZE = (224,224)
BATCH_SIZE = 32
EPOCHS = 200
VALIDATION_SPLIT = 0.20
AUTOTUNE = tf.data.AUTOTUNE

TARGET_COLS = [
    "mean_irradiance", "weighted_intensity", "
        cloud_coverage",
    "optical_variance", "sun_X", "sun_Y",
    "atmospheric_indicator"
]

USE_MIXED_PRECISION = True
FREEZE_BACKBONE = True

if USE_MIXED_PRECISION:
    from tensorflow.keras import mixed_precision
    mixed_precision.set_global_policy("mixed_float16")
    print("Mixed precision enabled: global policy =",
        mixed_precision.global_policy())

assert os.path.isdir(IMAGE_DIR), f"Image dir not found: {
    IMAGE_DIR}"
assert os.path.exists(LABELS_CSV), f"Labels CSV not found:
    {LABELS_CSV}"

labels_df = pd.read_csv(LABELS_CSV)

if "Filename" not in labels_df.columns:
    required = {"Year", "Month", "Day", "Hour", "Minute", "
        Second"}
    if required.issubset(set(labels_df.columns)):
        labels_df["Filename"] = labels_df.apply(
            lambda r: f"{int(r.Year):04d}{int(r.Month):02d}
                {int(r.Day):02d}{int(r.Hour):02d}{int(r.
                    Minute):02d}{int(r.Second):02d}.jpg",

```

```

        axis=1
    )
    else:
        raise ValueError("No 'Filename' column found and
                           cannot construct it from timestamps.")

full_paths = labels_df["Filename"].apply(lambda f: os.path
                                           .join(IMAGE_DIR, f))
exists_mask = full_paths.apply(os.path.exists)
if not exists_mask.all():
    missing = (~exists_mask).sum()
    print(f"Warning: {missing} images referenced in CSV
           were not found on disk. Those rows will be skipped.
           ")
labels_df = labels_df[exists_mask].reset_index(drop=True)

for c in TARGET_COLS:
    if c not in labels_df.columns:
        raise ValueError(f"Target column missing in CSV: {
                           c}")

filenames = labels_df["Filename"].astype(str).to_numpy()
targets = labels_df[TARGET_COLS].to_numpy(dtype=np.float32
)

num_samples = len(filenames)
indices = np.arange(num_samples)
np.random.seed(42)
np.random.shuffle(indices)
val_count = int(math.ceil(VALIDATION_SPLIT * num_samples))
val_idx = indices[:val_count]
train_idx = indices[val_count:]

train_files = filenames[train_idx]
train_targets = targets[train_idx]
val_files = filenames[val_idx]
val_targets = targets[val_idx]

print(f"Total samples: {num_samples}, train: {len(
    train_files)}, val: {len(val_files)}")

target_mean = train_targets.mean(axis=0)
target_std = train_targets.std(axis=0)

target_std[target_std == 0] = 1.0

print("Target mean:", target_mean)

```

```

print("Target std:", target_std)

def standardize_targets(y):
    return (y - target_mean) / target_std

def destandardize_targets(y_scaled):
    return y_scaled * target_std + target_mean

def preprocess_image(filename):
    img_path = tf.strings.join([IMAGE_DIR, filename],
                               separator=os.sep)
    img = tf.io.read_file(img_path)
    img = tf.io.decode_jpeg(img, channels=3)
    img = tf.image.convert_image_dtype(img, tf.float32)
    img = tf.image.resize(img, IMAGE_SIZE, method='
        bilinear')

    return img

def make_dataset(file_list, target_array, batch_size,
                 shuffle=True):
    ds_files = tf.data.Dataset.from_tensor_slices(
        file_list)
    ds_targets = tf.data.Dataset.from_tensor_slices(
        target_array)
    ds = tf.data.Dataset.zip((ds_files, ds_targets))
    if shuffle:
        ds = ds.shuffle(buffer_size=4096)

    def _map_fn(filename, target):
        img = preprocess_image(filename)
        t = (target - target_mean) / target_std
        return img, t
    ds = ds.map(_map_fn, num_parallel_calls=AUTOTUNE)
    ds = ds.batch(batch_size, drop_remainder=False)
    ds = ds.prefetch(AUTOTUNE)
    return ds

train_ds = make_dataset(train_files, train_targets,
                        batch_size=BATCH_SIZE, shuffle=True)
val_ds = make_dataset(val_files, val_targets, batch_size=
    BATCH_SIZE, shuffle=False)

def build_model(input_shape=(None, None, 3), num_outputs=
    len(TARGET_COLS), freeze_backbone=FREEZE_BACKBONE):
    base = EfficientNetB0(include_top=False, weights='
        imagenet', input_shape=(*IMAGE_SIZE, 3))

```

```

base.trainable = not freeze_backbone
inputs = layers.Input(shape=(*IMAGE_SIZE, 3))
x = base(inputs, training=False)
x = layers.GlobalAveragePooling2D()(x)
x = layers.Dense(256, activation='relu')(x)
x = layers.Dropout(0.3)(x)
x = layers.Dense(128, activation='relu')(x)
x = layers.Dropout(0.2)(x)
outputs = layers.Dense(num_outputs, dtype='float32',
                        activation='linear')(x)
model = models.Model(inputs, outputs)
return model

strategy = tf.distribute.MirroredStrategy() if len(tf.
    config.list_physical_devices('GPU')) > 1 else None
if strategy:
    print("Using MirroredStrategy on", strategy.
        num_replicas_in_sync, "GPUs")
    with strategy.scope():
        model = build_model()
        opt = optimizers.Adam(learning_rate=1e-4)
        model.compile(optimizer=opt, loss=losses.
            MeanSquaredError(), metrics=[metrics.
                MeanAbsoluteError()])
else:
    model = build_model()
    opt = optimizers.Adam(learning_rate=1e-4)
    model.compile(optimizer=opt, loss=losses.
        MeanSquaredError(), metrics=[metrics.
            MeanAbsoluteError()])

model.summary()

cb_checkpoint = callbacks.ModelCheckpoint(MODEL_SAVE_PATH,
    save_best_only=True, monitor='val_loss', mode='min',
    save_weights_only=False)
cb_early = callbacks.EarlyStopping(monitor='val_loss',
    patience=8, restore_best_weights=True)
cb_reduce = callbacks.ReduceLROnPlateau(monitor='val_loss',
    factor=0.5, patience=3, verbose=1)

history = model.fit(
    train_ds,
    validation_data=val_ds,
    epochs=EPOCHS,
    callbacks=[cb_checkpoint, cb_early, cb_reduce],
    verbose=1

```

```

)

hist_df = pd.DataFrame(history.history)
hist_df.to_csv("training_history.csv", index=False)

val_eval = model.evaluate(val_ds, verbose=1)
print("Validation evaluation (loss, mae):", val_eval)

if os.path.exists(MODEL_SAVE_PATH):
    model = tf.keras.models.load_model(MODEL_SAVE_PATH)

import math, csv

true_targets_array = labels_df[TARGET_COLS].values
def predict_to_csv(model, file_list, true_targets_array,
    out_csv, batch_size=BATCH_SIZE):
    ds_files = tf.data.Dataset.from_tensor_slices(
        file_list).batch(batch_size)
    ds_imgs = make_dataset(file_list, true_targets_array,
        batch_size=batch_size, shuffle=False)

    header = ["Filename"] + [f"Pred_{c}" for c in
        TARGET_COLS] + [f"True_{c}" for c in TARGET_COLS]
    with open(out_csv, mode='w', newline='') as f:
        writer = csv.writer(f)
        writer.writerow(header)

        for batch_files, (batch_imgs, batch_targets_scaled
            ) in zip(ds_files, ds_imgs):
            batch_fnames = [fn.numpy().decode("utf-8") if
                isinstance(fn.numpy(), (bytes, bytearray))
                else str(fn.numpy()) for fn in batch_files]

            preds_scaled = model.predict_on_batch(
                batch_imgs)
            preds_scaled = np.asarray(preds_scaled)
            batch_targets_scaled = np.asarray(
                batch_targets_scaled)

            preds_unscaled = destandardize_targets(
                preds_scaled)
            true_unscaled = destandardize_targets(
                batch_targets_scaled)

            for fname, pred_arr, true_arr in zip(
                batch_fnames, preds_unscaled, true_unscaled
            ):

```

```

        pred_list = np.array(pred_arr).reshape(-1)
        .tolist()
        true_list = np.array(true_arr).reshape(-1)
        .tolist()
        writer.writerow([fname] + pred_list +
            true_list)

    print(f"Predictions saved to: {out_csv}")

predict_file_list = labels_df["Filename"].tolist()
predict_to_csv(model, predict_file_list,
    true_targets_array, PRED_CSV, batch_size=BATCH_SIZE)

print("Done")

```

Código 5.1: Red Neuronal Convolutacional

Y de acuerdo a este código, la gráfica (elaborada en Matlab) de las métricas de error se muestran en la figura 5.1.

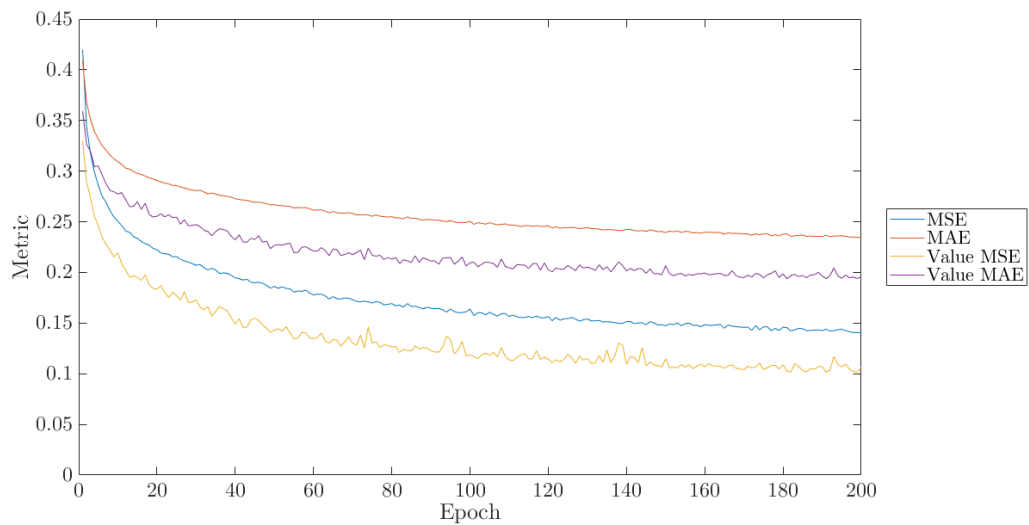


Figura 5.1: Métricas de error de la CNN

5.2. Desglose de la Red Neuronal Recurrente

Para elaborar la red neuronal recurrente, se utilizó igualmente el software de Python utilizando las librerías de TensorFlow, ya que dichas librerías facilitan la elaboración de la RNN de tipo LSTM. Se toman en cuenta los siguientes pasos para elaborarla:

1. Se introducen las librerías a utilizar, incluyendo la más importante en este código, la librería TensorFlow.
2. Se establecen los archivos a utilizar y los archivos a generar, estos pueden cambiarse de acuerdo al usuario.
3. Se lee el archivo delimitado por comas a utilizar y se normalizan los datos en un arreglo por python.
4. Se establece la secuencia a utilizar y el número de datos que corresponda a esta secuencia.
5. Las matrices se separan en entrenamiento y prueba en una relación 80/20 en cuanto a porcentaje.
6. Se propone el modelo, el número de neuronas a utilizar y las capas ocultas junto con el método de activación.
7. El modelo se compila con las métricas de error establecidas.
8. Se entrena la red en base al modelo, y se guarda la métrica de error histórica en un archivo delimitado por comas separado del principal.
9. Los valores escalados se convierten nuevamente en los valores originales del archivo usado como base de datos.
10. Por último, se exportan los datos en los archivos debidos, cuyos son la predicción de los datos y el peso de las neuronas [24].

Y el código elaborado resultó significativamente más corto que el de la CNN ya que este solo se ejecuta en base a una dataset dada por el usuario, y el mismo se muestra a continuación:

```
import numpy as np
import pandas as pd
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import \acrshort{LSTM}, Dense
    , Dropout
from sklearn.preprocessing import StandardScaler

CSV_FILE = "FinalTrainData.csv"
MODEL_SAVE = "rnn_solar_model.keras"
PRED_CSV = "rnn_predictions.csv"

df = pd.read_csv(CSV_FILE)

df_numeric = df.select_dtypes(include=[np.number])
data = df_numeric.values

scaler = StandardScaler()
data_scaled = scaler.fit_transform(data)

SEQ_LEN = 20

def create_sequences(data, seq_len):
    X, y = [], []
    for i in range(len(data) - seq_len):
        X.append(data[i:i+seq_len])
        y.append(data[i+seq_len])
    return np.array(X), np.array(y)

X, y = create_sequences(data_scaled, SEQ_LEN)

print("X shape:", X.shape)
print("y shape:", y.shape)

split = int(0.8 * len(X))
X_train, X_val = X[:split], X[split:]
y_train, y_val = y[:split], y[split:]

model = Sequential([
```

```

\acrshort{LSTM}(128, return_sequences=True,
    input_shape=(SEQ_LEN, X.shape[2])),
Dropout(0.2),

\acrshort{LSTM}(64, return_sequences=False),
Dropout(0.2),

Dense(64, activation='relu'),
Dense(y.shape[1], activation='linear')
])

model.compile(optimizer='adam', loss='mse', metrics=['mae',
])
model.summary()

history = model.fit(
    X_train, y_train,
    validation_data=(X_val, y_val),
    epochs=30,
    batch_size=32,
    verbose=1
)

hist_df = pd.DataFrame(history.history)
hist_df.to_csv("historyRNN.csv", index=False)

model.save(MODEL_SAVE)
print(f"Model saved as {MODEL_SAVE}")

pred_scaled = model.predict(X_val)

y_val_unscaled = scaler.inverse_transform(y_val)
pred_unscaled = scaler.inverse_transform(pred_scaled)

columns = df_numeric.columns

out_df = pd.DataFrame(
    np.hstack([y_val_unscaled, pred_unscaled]),
    columns=[f"True_{c}" for c in columns] + [f"Pred_{c}"
        for c in columns]
)

out_df.to_csv(PRED_CSV, index=False)
print(f"Predictions saved to {PRED_CSV}")

```

Código 5.2: Red Neuronal Recurrente modo LSTM

De acuerdo al código anterior, las métricas de error al ejecutar la red neuronal se muestran en la figura 5.2.

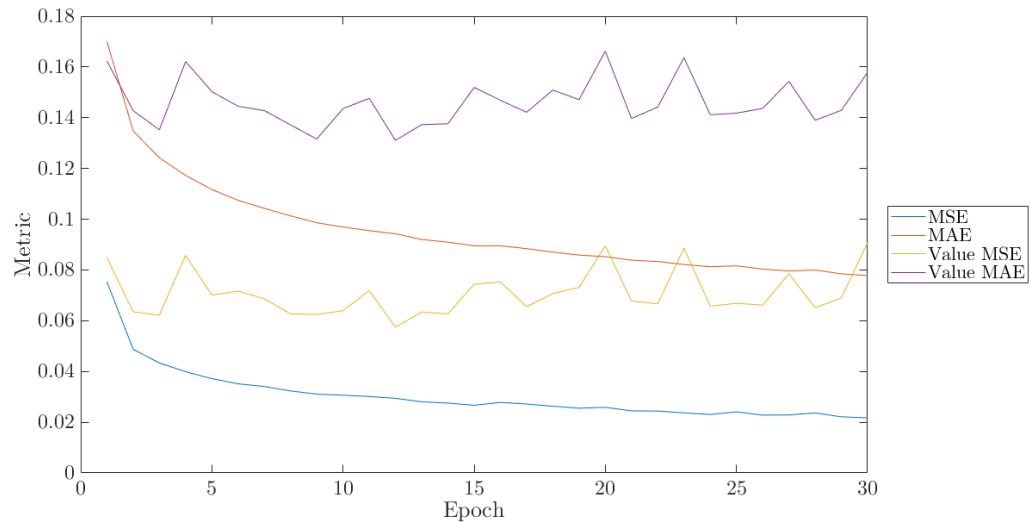


Figura 5.2: Métricas de error de la RNN

Capítulo VI - Análisis y Resultados

6.1. Resultados Obtenidos de Ambas Redes

Debido a la gran cantidad de resultados, se realizará un muestreo de solamente los primeros 1000 de los resultados de predicción de la red neuronal.

El análisis de resultados obtenido mediante el funcionamiento de la red neuronal proporciona diversas tareas críticas que podrían ayudar a validar eficazmente la capacidad del sistema para predecir las variables físicas que definen la variabilidad de los recursos solares. La predicción de la irradiancia solar es una de las tareas más críticas, ya que abarca una serie de variables que influyen en la actividad solar del entorno. Al comparar los resultados del sistema CNN-RNN con el cálculo teórico de la ecuación 3.16 de referencia de la irradiancia, podría ser posible determinar si el sistema puede predecir la dependencia del cenit solar, la reducción relacionada con la masa de aire o la profundidad óptica del entorno, basándose únicamente en las características visuales de las imágenes. Esto podría ayudar a identificar los sesgos críticos del sistema ante diversos patrones de nubes y predecir si el sistema puede identificar eficazmente las condiciones de cielo despejado y los patrones estocásticos con reducción de nubes.

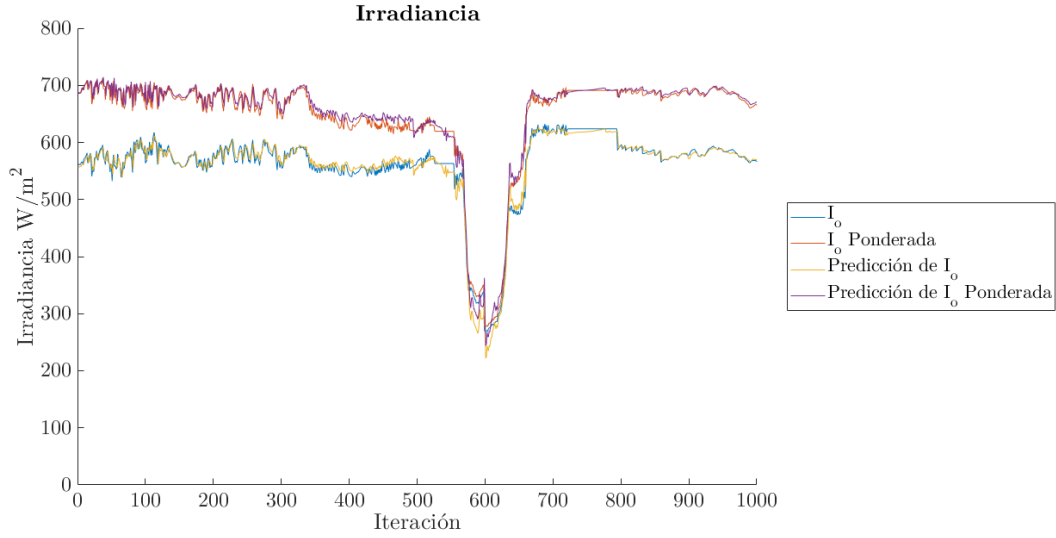


Figura 6.1: Gráfica de la irradiancia, comparando datos reales y datos generados

En la figura 6.1 se muestra la comparativa de las irradiancias reales y ponderadas junto con las predicciones de ambos parámetros antes dichos, se puede observar que se mantienen muy cercanos ambos valores con una mínima de error.

El análisis de los resultados de las coordenadas predichas del sol en el vector unitario o la proyección 2D tiene una función complementaria, ya que garantiza que la red haya aprendido con precisión la relación geométrica entre los patrones del cielo y la ubicación del disco solar. La determinación precisa de la posición del sol es fundamental, ya que la elevación y el azimut solar afectan directamente la ecuación de irradiancia, así como el algoritmo de orientación del seguidor solar de un sistema de seguimiento solar. Mediante la determinación del margen de error resultante del cálculo de la posición predicha del sol a partir de las ecuaciones de mecánica celeste del sistema solar, el estudio confirma si el razonamiento geométrico de la red se corresponde con la realidad astronómica, dado que la posición del sol puede estar cubierta o no de nubes.

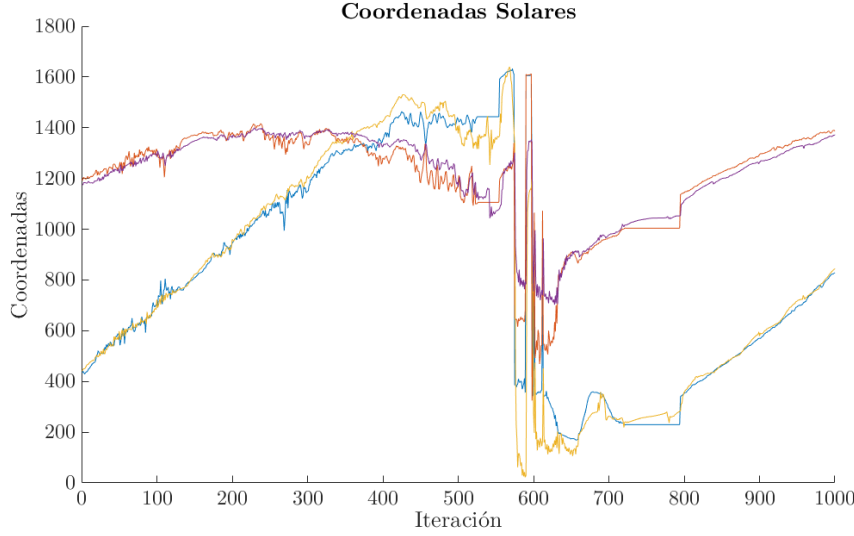


Figura 6.2: Gráfica de coordenadas solares, comparando datos reales y datos generados

Así como con la imagen 6.1 las muestras de datos se encuentran muy cercanas entre si, respectivamente las coordenadas (x, y) del sol junto con sus predicciones, tal y como se muestra en la figura 6.2.

Ya que el panel solar no podrá interpretar corrección por coordenadas exactamente, se tendrán que transformar para corregir el ángulo de azimuth y de elevación, que se relacionan directamente con la coordenada x y y respectivamente. Y la transformación se da por estas fórmulas:

$$\begin{aligned}
 Off_x &= (max(x) + min(x))/2 \\
 Off_y &= (max(y) + min(y))/2 \\
 r &= \sqrt{(x - Off_x)^2 + (y - Off_y)^2} \\
 \theta_e &= 90^\circ(1 - \sin(r(pi/2)/((max(x) - min(x))/2))) \\
 \theta_a &= atan2((y - Off_y)/(x - Off_x))
 \end{aligned} \tag{6.1}$$

donde:

- Off_x es el desplazamiento de x en caso de no encontrarse su centro en el centro

de la imagen.

- Off_y es el desplazamiento de y en caso de no encontrarse su centro en el centro de la imagen.
- θ_e es la corrección del ángulo de elevación, de requerirse, se puede directamente multiplicar por 90° sin aplicar ninguna resta dependiendo de donde se encuentre el ángulo de origen.
- θ_a es la corrección del ángulo de azimut, se le puede colocar un desplazamiento dependiendo de donde se encuentre el ángulo 0° .

De esta manera el panel puede interpretar la corrección en ambos de sus ángulos y poder proyectar el sol de manera que la cámara apunte a las coordenadas $(0,0)$ de ese momento en caso de ser una cámara dinámica. En caso de ser estática se calculará la corrección de acuerdo a los ángulos del panel solar y se aplicará la debida actualización de ambos ángulos al analizar y evaluar la siguiente foto.

El indicador atmosférico, el cual condensa la información proporcionada por las observaciones de claridad del cielo en una única métrica basada en estadísticas de intensidad de píxeles, es un resultado esencial que aclara la capacidad de la red para traducir las características del cielo que afectan la irradiancia. Dado que esta variable se corresponde directamente con el factor atenuante en la expresión 2.3 de la ecuación de irradiancia, sus resultados de predicción permiten estimar si la red traduce eficazmente el difusor óptico de las nubes y las características de dispersión difusa del cielo en efectos relacionados con la masa de aire. La coherencia de los resultados de predicción de las variables atmosféricas con los cálculos teóricos del coeficiente de atenuación y los resultados de irradiancia confirma que la red captura eficazmente los fenómenos atmosféricos reflejados por la fórmula.

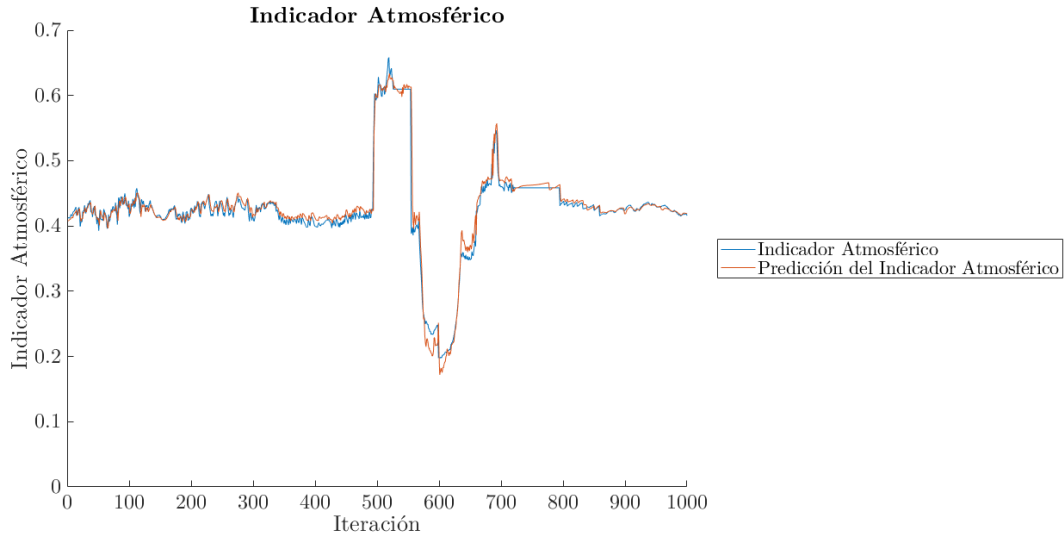


Figura 6.3: Gráfica de indicador atmosférico, comparando datos reales y datos generados

En la figura 6.3 se muestra la comparación del indicador atmosférico verdadero y el que obtuvieron las redes neuronales, estos datos se utilizarán en la ecuación de irradiancia 3.16 ahora utilizando el valor de indicador atmosférico y no un ideal como primeramente se utilizó, y se obtiene la gráfica 6.4.

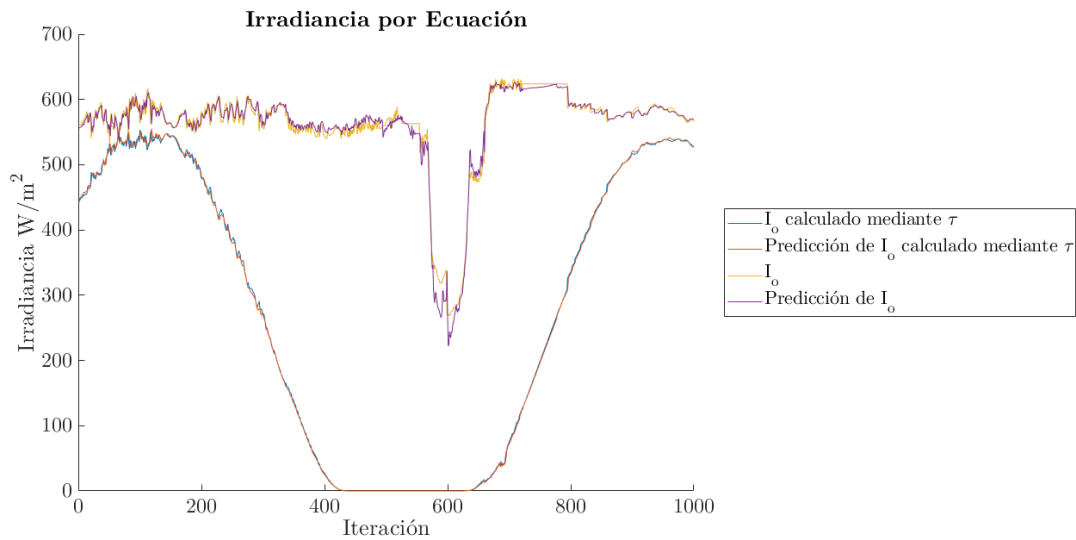


Figura 6.4: Gráfica de irradiancia mediante la ecuación, comparando datos reales, datos generados y datos de irradiancia de tanto la red como los datos reales

6.2. Trabajos a Futuro con Estos Resultados

Los resultados de la red neuronal sientan las bases para una amplia gama de trabajos futuros sobre la predicción de la energía solar, la evaluación de los recursos solares y el control inteligente de sistemas fotovoltaicos. Dado que el modelo extrae indicadores atmosféricos, métricas de nubes, estimaciones de irradiancia y proyecciones de la posición solar a partir de las propias imágenes del cielo, el desarrollo futuro de sistemas de energía solar más autónomos y adaptativos puede perfeccionar aún más estos resultados. Una posible vía reside en la integración de la irradiancia y los parámetros atmosféricos pronosticados en algoritmos sofisticados de seguimiento solar. En lugar de utilizar únicamente fórmulas astronómicas o dispositivos de posicionamiento con horarios fijos, los seguidores del futuro podrían incorporar predicciones basadas en redes neuronales para lograr la optimización en tiempo real de los ángulos de azimuth e inclinación. De esta manera, las instalaciones fotovoltaicas podrían compensar las condiciones transitorias de las nubes y las variaciones atmosféricas, mejorando así la captación de energía durante los periodos en que los seguidores tradicionales no pueden responder eficazmente.

Otra línea de investigación relevante para el futuro consiste en ampliar los horizontes de predicción. Aunque el modelo actual se centra principalmente en la estimación instantánea, en el futuro se podría trabajar en la predicción de la irradiancia a corto plazo combinando las salidas de la red neuronal con modelos temporales como LSTM, transformadores o redes neuronales híbridas basadas en la física. Estos modelos aprenderían la evolución temporal de las formaciones nubosas y el comportamiento atmosférico directamente a partir de secuencias de imágenes, prediciendo así la irradiancia con varios minutos o incluso horas de antelación. Esto resultaría muy útil para los sistemas fotovoltaicos conectados a la red, que requieren predicciones precisas a corto plazo para

mantener la estabilidad, implementar estrategias de respuesta a la demanda y reducir la dependencia de los mecanismos de equilibrio basados en combustibles fósiles.

Además, los resultados de la red neuronal pueden utilizarse para cuantificar mejor las incertidumbres en la modelización de la energía solar. Dado que los indicadores atmosféricos y las predicciones de cobertura nubosa son inherentemente variables, un trabajo posterior podría consistir en el uso de técnicas de predicción probabilística, como redes neuronales bayesianas o métodos de Monte Carlo con dropout, que permiten determinar un intervalo de confianza para cada parámetro de salida. Estas estimaciones de incertidumbre pueden, por lo tanto, ser utilizadas por los sistemas de gestión de energía para tomar decisiones más robustas, especialmente en microrredes o sistemas autónomos aislados, donde los impactos operativos debidos a errores de predicción son significativos.

Otra área de investigación consiste en transferir el modelo entrenado a diferentes climas, estaciones o sistemas de imagen mediante técnicas de adaptación de dominio. Dado que el conjunto de datos existente se limita a una ubicación y período de tiempo específicos, este estudio determinará, mediante aprendizaje por transferencia, generación de cielo sintético o calibración interregional, cómo la red neuronal se generaliza a las condiciones atmosféricas globales. Estos esfuerzos contribuirían en gran medida a establecer si el modelo podría escalarse para su implementación en redes de monitorización solar a gran escala y proporcionaría evaluaciones de irradiancia consistentes y fiables en diversas condiciones ambientales. Finalmente, los resultados del modelo podrían incorporarse a marcos de optimización de energías renovables más amplios. Por ejemplo, las predicciones de la red neuronal podrían integrarse en algoritmos de despacho económico, controladores de sistemas renovables híbridos o sistemas de mantenimiento predictivo capaces de identificar patrones atmosféricos anómalos que afectan al rendimiento fotovoltaico. Los datos extraídos sobre nubes e irradiancia podrían utilizarse como variables de entrada para una simulación de producción energética a largo

plazo, superando con creces la dependencia de los modelos tradicionales de cielo despejado y permitiendo evaluaciones mucho más realistas de la viabilidad de la energía fotovoltaica en futuras instalaciones. De este modo, la red neuronal desarrollada aquí constituye no solo una herramienta predictiva, sino también un componente modular que puede extenderse a investigaciones y aplicaciones de energía solar cada vez más sofisticadas.

Capítulo VII - Conclusiones

El entrenamiento y desarrollo de una red neuronal convolucional para la estimación de parámetros solares fue un proceso computacionalmente muy intensivo. Es evidente que las redes neuronales, especialmente aquellas con arquitecturas convolucionales profundas como EfficientNet, requieren una gran capacidad de procesamiento al entrenarse con numerosos conjuntos de datos a gran escala que comprenden miles de imágenes de alta resolución del cielo. A lo largo de este trabajo, se hizo patente que un ordenador de última generación equipado con una GPU potente, amplia memoria RAM y capacidad de almacenamiento optimizada es indispensable tanto para el entrenamiento como para la validación. Sin estos recursos de hardware, los tiempos de entrenamiento se prolongan considerablemente, lo que podría afectar negativamente a la eficiencia de convergencia.

El entrenamiento del modelo también requiere un gran número de épocas para garantizar el refinamiento progresivo de los pesos junto con la minimización de los valores de pérdida. Sin embargo, debido a las limitaciones prácticas de capacidad computacional y disponibilidad de tiempo, el número de épocas utilizadas en esta tesis tuvo que ser limitado. Por lo tanto, existía un límite natural en cuanto al grado de mejora posible en el rendimiento. Idealmente, los modelos de aprendizaje profundo se benefician de sesiones de entrenamiento prolongadas, ajuste iterativo de hiperparámetros y múltiples rondas de reentrenamiento con diferentes arquitecturas. Sin embargo, la reducida ventana computacional disponible impidió realizar más de unas pocas ejecuciones de épocas durante el desarrollo de esta tesis, lo que pudo haber limitado la capacidad de aprendizaje óptima del modelo.

Además, los largos tiempos de entrenamiento para cada época incrementaron significativamente este desafío. Para conjuntos de datos exigentes, como la monitorización visual continua del cielo, la detección de nubes y la estimación de irradiancia, cada

época puede tardar desde varios minutos hasta incluso horas, dependiendo del tamaño del lote, la resolución de entrada y la capacidad de la GPU. Esto es lo que dificulta la experimentación prolongada, especialmente cuando se trabaja con estrictas limitaciones de tiempo. Por lo tanto, esta tesis se centró principalmente en las configuraciones más importantes del modelo, en lugar de explorar exhaustivamente todo el espacio de búsqueda de hiperparámetros o arquitecturas. Si bien esto sugiere tendencias prometedoras en el rendimiento de este modelo, indica que, incluso con un tiempo de entrenamiento reducido, la red neuronal es capaz de aprender relaciones significativas entre las imágenes del cielo y los parámetros solares. Es razonable esperar que, con mayores recursos computacionales, ciclos de entrenamiento más largos y una experimentación más exhaustiva, la precisión predictiva y la capacidad de generalización mejoren significativamente. Esto refuerza la importancia de una infraestructura de hardware adecuada en la investigación moderna de aprendizaje automático y señala el potencial de futuras mejoras una vez que se superen las limitaciones computacionales.

Referencias Bibliográficas

- [1] K. Ahmed and A. Solanki. Calibration and sun-path extraction from hemispherical sky images. In *IEEE Photovoltaic Specialists Conference*, 2020. Methods for extracting solar trajectories from sky-camera imagery.
- [2] Mohammed Al-Lahham and Others. Short-term solar irradiance forecasting using sky-imager features and machine learning. *Energies*, 13(18):1–20, 2020. Feature-based irradiance estimation from all-sky images.
- [3] Faris A Almalki, Amani A Albraikan, Ben Othman Soufiene, and Obaid Ali. Utilizing artificial intelligence and lotus effect in an emerging intelligent drone for persevering solar panel efficiency. *Wireless Communications and Mobile Computing*, 2022(1):7741535, 2022.
- [4] Anonymous and Others. Benchmarking deep learning architectures for solar irradiance nowcasting: A 2025 evaluation, 2025. Compares CNN, CNN-LSTM, TCN, transformer-based models across multiple datasets.
- [5] Gustavo Arencibia-Carballo. La importancia del uso de paneles solares en la generación de energía eléctrica. *REDVET. Revista Electrónica de Veterinaria*, 17(9):1–4, 2016.
- [6] Sergio Carballo and Others. Reinforcement learning for optimal heliostat aiming in concentrated solar power systems. *Solar Energy*, 260:1200–1215, 2025. RL controllers for heliostat aiming with cloud and atmosphere uncertainty.
- [7] A. Chowdhury and R. Patel. Clstm-based spatiotemporal model for solar irradiance nowcasting. *Energy and AI*, 2023. ConvLSTM trained on sequences of sky images; captures cloud motion implicitly.

- [8] S. Chu, M. Nguyen, and Y. Fu. Sky image-based solar irradiance estimation using convolutional neural networks. *Renewable Energy*, 2019. Deep CNN trained on all-sky images for direct irradiance regression; improves short-term forecasting accuracy.
- [9] Ali El Alani and Others. Hybrid cnn-mlp architectures for short-term solar irradiance forecasting. *Solar Energy*, 218:345–358, 2021. Demonstrates gains from hybrid CNN-MLP and CNN-LSTM models.
- [10] J. Fernandez, B. Aranda, and M. Gómez. All-sky image-based solar forecasting: A review of deep learning methods. *Renewable and Sustainable Energy Reviews*, 2024. Comprehensive review covering CNNs, Transformers, optical flow, and hybrid models (2015–2024).
- [11] Soteris Kalogirou and Arzu Sencan. Artificial intelligence techniques in solar energy applications. *Solar collectors and panels, theory and applications*, 15:315–340, 2010.
- [12] A. Kazantzidis, P. Tzoumanikas, and P. Kosmopoulos. Cloud-motion based forecasting of solar irradiance using all-sky cameras and optical flow. *Solar Energy*, 2020. Combines sky imagery with optical-flow cloud tracking; benchmark work in short-term nowcasting.
- [13] S. Kim and D. Park. Hybrid cnn-transformer architecture for sky-image-based solar irradiance prediction. *Solar Energy*, 2024. CNN encoder + Transformer temporal module for multi-step forecasting. State-of-the-art performance.
- [14] H. Lee and J. Kim. Learning-based control of pv tilt angle using real-time sky imagery. *Renewable Energy*, 2023. Uses camera-based neural estimators to dynamically adjust PV tilt.
- [15] J. Liu and R. Chen. Deep learning for short-term global horizontal irradiance forecasting using sky cameras. *Applied Energy*, 2021. Multi-hour ahead predictions

using deep CNNs; explores architectures and ablation studies.

- [16] National Renewable Energy Laboratory (NREL). Advances in solar forecasting: A comprehensive review. Technical report, NREL, 2023. Review of deep learning, sky-imager forecasting, and physics-guided methods.
- [17] National Renewable Energy Laboratory (NREL). Case studies on image-based solar forecasting and control systems. Technical report, NREL, 2024. Implementation lessons, benchmark systems, uncertainty treatment.
- [18] E. Ntavelis, I. Papachristou, and D. Karamanis. Solar irradiance forecasting with cnn-based sky feature extraction and lagged radiometric data. *Energy AI*, 2022. Uses CNN image encoders feeding into regression layers; shows superiority over pure radiometric models.
- [19] Jimmy Rajagukguk and Others. Deep learning approaches for solar irradiance estimation using sky cameras. *Sensors*, 21(9):1–22, 2021. Includes preprocessing, cloud coverage estimation, and irradiance prediction.
- [20] L. Ramírez and J. Ortega. Fisheye lens geometric projection models for solar position estimation. *Journal of Atmospheric and Solar–Terrestrial Physics*, 2022. Camera calibration and mapping of celestial coordinates to fisheye images.
- [21] R. Rathore and N. Verma. Physics-informed neural networks for solar irradiance prediction. *Applied Solar Energy*, 2021. Shows integration of solar geometry equations inside neural architectures.
- [22] J. Sirignano and K. Chen. Fusion of cnn features and numerical weather data for solar power prediction. In *IEEE Conference on Energy Forecasting and Analytics*, 2020. Hybrid physical–ML model that merges sky images with meteorological variables.

- [23] JW Spencer. A comparison of methods for estimating hourly diffuse solar radiation from global solar radiation. *Solar Energy*, 29(1):19–32, 1982.
- [24] Y. Tanaka and S. Mori. Neural estimation of effective irradiance for real-time pv efficiency maximization. In *International Conference on Smart Energy Systems*, 2023. Combines irradiance estimation with power flow optimization.
- [25] Masato Tsuchida and Others. Deep reinforcement learning for solar pv tracking under variable cloudiness. *Energies*, 15(22):1–18, 2022. Actor–critic control strategies for optimal panel tilt.
- [26] L. Wang and X. Zhou. Deep learning-based sun-tracking optimization for photovoltaic panels. *IEEE Transactions on Sustainable Energy*, 2021. CNN-based estimations of optimal panel angles; improves energy capture.
- [27] Wei Xu and Others. Multi-step solar irradiance forecasting using lstm–informer hybrid networks. *Energy AI*, 6:100150, 2024. Advanced temporal models for minute-level multi-step forecasting.