



Clasificación de estrellas y galaxias a través de ensamble híbrido de deep learning.

Oscar Felipe Robayo Lozano

Monografía presentada para optar al título de Especialista en Analítica y Ciencia de Datos

Asesor

Juan Carlos Muñoz Cuartas, Doctor (PhD) en Astrofísica

Universidad de Antioquia
Facultad de Ingeniería
Especialización en Analítica y Ciencia de Datos
Medellín, Antioquia, Colombia
2023

Cita	(Robayo O, 2023)
Referencia	Robayo O. (2023). <i>Clasificación de estrellas y galaxias a través de ensamble híbrido de deep learning</i> . [Trabajo de grado especialización]. Universidad de Antioquia, Medellín, Colombia.
Estilo APA 7 (2020)	



Especialización en Analítica y Ciencia de Datos, Cohorte II.

Grupo de Investigación Física y Astrofísica Computacional (FACOM).



Repositorio Institucional: <http://bibliotecadigital.udea.edu.co>

Universidad de Antioquia - www.udea.edu.co

El contenido de esta obra corresponde al derecho de expresión de los autores y no compromete el pensamiento institucional de la Universidad de Antioquia ni desata su responsabilidad frente a terceros. Los autores asumen la responsabilidad por los derechos de autor y conexos.

Tabla de contenido

Resumen	9
Abstract	10
Introducción	11
1 Planteamiento del problema	13
2 Justificación.....	14
3 Objetivos	15
4.1 Objetivo general	15
4.2 Objetivos específicos.....	15
4 Estado del Arte	16
5 Marco Teorico	17
5.1 Radiacion de los cuerpos astrofísicos.....	17
5.1.1 Estrellas.....	17
5.1.2 Galaxias.....	17
5.2 Camaras CCD.....	18
5.2.1 Efecto fotoeléctrico.....	18
5.2.2 Reducción de imágenes CCD	19
5.2.2.1 Ruido de lectura	19
5.2.2.2 Reducción.....	20
5.3 Machine Learning	21
5.4 Deep learning	22
5.4.1 Neurona.....	23
5.4.2 Red Neuronal	24
5.4.3 Red Neuronal Convolutacional RNC	25

6 Metodología	26
6.1 Extracción, caracterización y Limpieza de la Data	28
6.3 Modelado de RNC de Detección de Objetos	31
6.3.1 Descripción del modelo	31
6.3.2 Arquitectura del modelo	31
6.3.2.1 MobileNetv2.....	32
6.3.2.2 SSD.....	34
6.3.3 Entrenamiento del modelo	35
6.3.3.1 Experimentación.....	35
6.3.3.1.1 Generación de Datasets Dummies	35
6.3.3.1.2 Resultados Experimentación	37
6.3.3.2 Entrenamiento con Dataset SDSS	39
6.4 Modelado de RNC de Clasificación de Imágenes.....	41
6.4.1 Descripción del modelo	41
6.4.2 Arquitectura del modelo	42
6.4.3 Entrenamiento del modelo	45
7.4.3.1 Experimentación.....	45
6.4.3.1.1 Resultados Experimentación	47
6.4.3.2 Entrenamiento con Dataset SDSS	49
6.5 Ensamblado de sistema híbrido.....	51
6.6 Evaluación del modelo	54
6.6.1 Métricas de Evaluación.....	55
7 Resultados	56
Conclusiones	57
Recomendaciones.....	58

Referencias	59
-------------------	----

Lista de tablas

Tabla 1 Ejemplo de metadata asociada al dataset extraído del SDSS	28
Tabla 2 Ejemplo de metadata asociada modificada para adaptarse al input del modelo de detección.....	29
Tabla 3 Ejemplo de metada generada para entrenamiento del modelo de clasificación.....	30
Tabla 4 Ejemplo de dataset de coordenadas generado	36
Tabla 5 Tabla resumen de resultados del modelo de detección dataset Dummy	38
Tabla 6 Tabla resumen de resultados del modelo de detección dataset SDSS vs dataset Dummy	39
Tabla 7 Análisis de Componentes Principales	46
Tabla 8 Resultados de modelo ResNet con dataset de prueba	47
Tabla 9 Resultados de modelo ResNet con dataset de SDSS.....	49
Tabla 10 Ejemplo formato de file CSV output del modelo de detección de objetos	52
Tabla 11 Ejemplo output de modelo de clasificación de imágenes	52
Tabla 12 Ejemplo de formato de output definitivo del ensemble completo	53
Tabla 13 Ejemplo de formato para evaluación del ensemble completo.....	54
Tabla 14 Resultados Generales del ensemble completo	56

Lista de figuras

Figura 1	Comparación entre fotografías de galaxia y estrella.....	13
Figura 2	Esquema descriptivo del efecto fotoeléctrico	18
Figura 3	Distribución de cargas acumuladas en un píxel.....	19
Figura 4	Comparación imagen cruda con imagen resultante del proceso de reducción	20
Figura 5	Machine Learning en Inteligencia Artificial.....	21
Figura 6	Deep Learning en Inteligencia Artificial	22
Figura 7	Estructura interna de una Neurona	23
Figura 8	Ejemplos de Redes Neuronales Artificiales	24
Figura 9	Ejemplo de estructura de red neuronal convolucional RNC	25
Figura 10	Diagrama de metodología general	26
Figura 11	Ejemplo de imagen para entrenamiento del modelo de detección de objetos.	28
Figura 12	Ejemplos de data inconsistente en el dataset extraído	29
Figura 13	Ejemplo de objetos recortados para entrenamiento de modelo de clasificación	30
Figura 14	Diagrama general de inputs y outputs del modelo MobileNetSSDv2.....	31
Figura 15	Arquitectura RNC MobileNetSDDv2.....	32
Figura 16	Estructura Bloque de Residuo de Cuello de Botella.....	33
Figura 17	Estructura de Bloque de Residuo de Cuello de Botella con funciones explicadas.....	33
Figura 18	Arquitectura de RNC basada en SSD	34
Figura 19	Ejemplo de datasets generados para testeo	36
Figura 20	Imágenes con Bounding Boxes construidos a partir de las predicciones	38
Figura 21	Resultados pérdida de localización y tasa de aprendizaje dataset Dummy	39
Figura 22	Resultados pérdida de localización y tasa de aprendizaje dataset SDSS.....	40

Figura 23	Diagrama general de inputs y outputs del modelo ResNet.....	41
Figura 24	Estructura de Bloque Residual en ResNet	42
Figura 25	Estructura de Bloque de Identidad.....	42
Figura 26	Estructura de un Bloque Convolutivo.....	43
Figura 27	Diagrama de estructura de RNC ResNet utilizada en el modelo.....	44
Figura 28	Ejemplo de imágenes del dataset de prueba	45
Figura 29	Ejemplo de imagen luego de pasar por PCA.....	46
Figura 30	Gráficas de métricas de evaluación de modelo ResNet con dataset de prueba, Loss Function.....	48
Figura 31	Gráficas de métricas de evaluación de modelo ResNet con dataset de prueba, Precision Function.....	48
Figura 32	Gráficas de métricas de evaluación de modelo ResNet con dataset de prueba, AUC..	48
Figura 33	Gráficas de métricas de evaluación de modelo ResNet con dataset de SDSS, Loss Function.....	50
Figura 34	Gráficas de métricas de evaluación de modelo ResNet con dataset de SDSS, Precision Function.....	50
Figura 35	Gráficas de métricas de evaluación de modelo ResNet con dataset de SDSS, AUC ...	50
Figura 36	Diagrama del ensamble para detección y clasificación de estrellas y galaxias	51

Siglas, acrónimos y abreviaturas

RNC	Red Neuronal Convolutacional
MobileNetSSDv2	Algoritmo de detección de objetos basado en MobileNetv2 y SSD
MobileNetv2	Red Neuronal Convolutacional para extracción de características.
SSD	Single Shot Multibox Detector, algoritmo de detección de objetos basado en deep learning.
AVG-16	Visual Geometry Group, red Neuronal Convolutacional para extracción de objetos utilizada en SSD.
ResNet	Residual Network, algoritmo que utiliza bloques residuales basado en deep learning.
PCA	Principal Component Analysis, algoritmo de reducción de dimensionalidad.
SDSS	Sloan Digital Sky Survey (proyecto de investigación del espacio mediante imágenes en el espectro visible y de corrimiento al rojo, realizada en un telescopio ubicado en el observatorio <i>Apache Point</i> en Nuevo México).
SVM	Support Vector Machine (algoritmos de aprendizaje supervisado para problemas de clasificación y regresión).
G/E – g/e	Galaxias/Estrellas
UdeA	Universidad de Antioquia.

Resumen

La astronomía es una ciencia que nos permite comprender la estructura y evolución del universo en el que vivimos. Una de las tareas fundamentales en la investigación astronómica es la identificación y clasificación de fuentes de luz en las imágenes capturadas por los telescopios, más específicamente entre estrellas y galaxias. Sin embargo, distinguir entre estos cuerpos celestes en una imagen puede ser un desafío, ya que ambas pueden ser similares en cuanto a forma y tamaño a partir de ciertas distancias. Por otro lado los recientes y venideros censos del cielo proveen enormes cantidades de información en la forma de imágenes (del orden de decenas de Tb por semana). Identificar y clasificar fuentes en tales conjuntos de datos se convierte en una tarea imposible desde el punto de vista práctico si se piensa en que sea una labor hecha por humanos, y se hace necesario hacer uso de herramientas computacionales para resolver el problema. En este proyecto, se presenta una solución a esta problemática mediante un framework que combina y explota dos modelos de deep learning: MobileNetSSDv2 para la detección de objetos y ResNet para clasificar los objetos detectados. El objetivo de este trabajo es demostrar la efectividad de este enfoque en la identificación precisa de fuentes de luz astronómicas en imágenes. Los modelos fueron entrenados utilizando redes neuronales convolucionales y técnicas de deep learning, que proveen mejores resultados que otros métodos basados en ingeniería de características manuales o en maquinas de vectores de soporte (SVM). Estos métodos fueron entrenados utilizando el dataset público de Sloan Digital Sky Survey (SDSS).

Palabras clave: deep learning, análisis de data, estadística, clasificación, estrellas, galaxias, detección de objetos,

Abstract

Astronomy is a science that allows us to understand the structure and evolution of the universe we live in. One of the fundamental tasks in astronomical research is the identification and classification of light sources in images captured by telescopes, specifically between stars and galaxies. However, distinguishing between these celestial bodies in an image can be challenging since both can appear similar in shape and size from certain distances. On the other hand, recent and upcoming sky surveys provide enormous amounts of information in the form of images (on the order of tens of terabytes per week). Identifying and classifying sources in such data sets becomes an impractical task if done manually by humans, and it becomes necessary to use computational tools to solve the problem. In this project, a solution to this problem is presented through a framework that combines and exploits two deep learning models: MobileNetSSDv2 for object detection and ResNet for classifying the detected objects. The objective of this work is to demonstrate the effectiveness of this approach in accurately identifying astronomical light sources in images. The models were trained using convolutional neural networks and deep learning techniques, which provide better results than other methods based on manual feature engineering or support vector machines (SVM). These methods were trained using the public dataset from the Sloan Digital Sky Survey (SDSS).

Keywords: deep learning, data analysis, statistics, classification, stars, galaxies, object detection.

Introducción

La identificación y clasificación precisa de fuentes de luz en las imágenes astronómicas es una tarea fundamental para la investigación en astrofísica, estos objetos poseen múltiples indicadores medibles que nos permiten clasificarlos y conocer características intrínsecas, procesos de formación, etc. Entre los objetos que podemos encontrar en los repositorios de cartografía del cielo visible se encuentran galaxias, estrellas, cuásares, supernovas, entre otros. Para clasificarlos se requiere la medición de los parámetros morfológicos, como color, tamaño y forma, este análisis se ha realizado tradicionalmente a través de la extracción y selección de características o mediante la inspección visual de expertos, por lo que clasificar los objetos celestes se vuelve una tarea imposible de realizar en colecciones masivas de imágenes debido al consumo de tiempo y recursos que esta requiere (Cordero-Chan R. et al 2018), adicional a esto, diferenciar mediante inspección visual se vuelve progresivamente más difícil a medida que analizamos objetos ubicados a una mayor distancia. Es por esto que en la última década se han propuesto múltiples métodos para automatizar esta tarea, pero estos todavía carecen de la precisión deseada para considerarse fiables.

Este problema se ha abordado comúnmente a través de herramientas de visión por computadora, que es un campo que busca que las máquinas puedan procesar visualmente el mundo de la misma manera que el cerebro y los ojos humanos. Anteriormente, la detección de objetos se basaba en la ingeniería de características manuales y descriptores como SIFT y HOG (Dalal y Triggs, 2005). Actualmente, hay encuestas fotométricas grandes en curso y próximas, como el Dark Energy Survey (DES) y el Large Synoptic Survey Telescope (LSST), que detectarán una gran cantidad de objetos a magnitudes tenues, por lo tanto, las observaciones fotométricas no pueden separar de manera confiable las estrellas de las galaxias no resueltas por morfología, llevando a incompletitud y contaminación en las muestras. Motivo por el cual actualmente se está abordando el problema a través de métodos de aprendizaje profundo para la extracción de características y clasificación de fuente de origen, aprovechando las mejoras en capacidad de procesamiento y disponibilidad de la data fotométrica.

En este contexto, los enfoques de aprendizaje profundo han demostrado ser efectivos en la clasificación de estrellas y galaxias en imágenes astronómicas debido a su capacidad para extraer características útiles de los datos de entrada y aprender representaciones abstractas de las mismas. Por lo tanto, el desarrollo de un framework basado en técnicas de aprendizaje profundo para la detección y clasificación de estrellas y galaxias en imágenes astronómicas puede tener un gran impacto en la investigación astrofísica, al ser más preciso, por lo tanto, inferir propiedades de los objetos de manera más confiable.

Es objetivo de esta investigación implementar un ensamble híbrido que combine dos modelos de aprendizaje profundo para la detección y clasificación entre dos categorías, estrellas y galaxias. El primer modelo se encarga de detectar las posibles fuentes de energía en las imágenes y proponer coordenadas estimadas de estos objetos. El segundo modelo se encarga clasificación, es decir determinar si el objeto es una galaxia o una estrella. En este proyecto investigativo se hará uso del sistema de detección de objetos en tiempo real MobileNetSSDv2 (MobileNet Single Shot Multibox Detector) y de la arquitectura de red neuronal ResNet (Residual Network) para abordar cada una de estas tareas respectivamente.

Para la realización de este proyecto, se utilizarán imágenes del Sloan Digital Sky Survey (SDSS), una iniciativa que tiene como objetivo realizar un mapa tridimensional del universo observable y proporcionar a la comunidad científica datos astronómicos detallados. Esta iniciativa nacida en 1998 ha producido la recopilación de imágenes digitales más grande del cielo, con más de 500 millones de objetos detectados en su base de datos y generando más de 200gb diarios de información (SDSS.org 2020). Este proyecto se hará uso de esta valiosa base de datos para entrenar y evaluar los modelos de deep learning, lo que permitirá la identificación precisa de estrellas y galaxias en las imágenes de este repositorio.

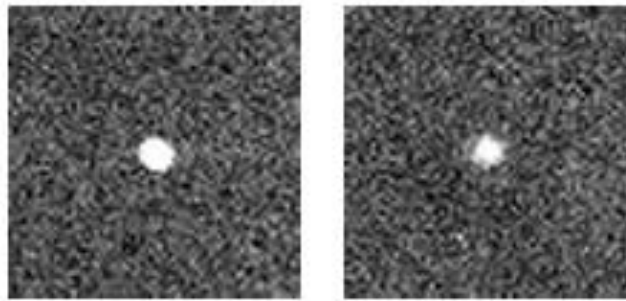
1 Planteamiento del problema

La tarea de identificación y clasificación de estrellas y galaxias en imágenes astronómicas es crucial en la investigación del universo, ya que permite comprender la composición y evolución del mismo. Sin embargo, esta tarea es altamente desafiante, ya que las estrellas y las galaxias pueden ser similares en forma y tamaño a ciertas distancias como se puede apreciar en la figura 1, lo que dificulta la distinción precisa entre ellas mediante la inspección visual, llevando a que en las observaciones fotométricas no se pueda separar de manera confiable las estrellas de las galaxias no resueltas por morfología, llevando a incompletitud y contaminación en las muestras, por lo que clasificar los objetos en la data fotométrica se vuelve una tarea imposible de realizar en colecciones masivas de imágenes debido al consumo de tiempo y recursos que requeriría esta labor.

Adicional a los altos costos asociados a la labor de clasificación por medio de inspección visual, la data disponible para ser explorada y categorizada continúa creciendo a ritmos acelerados, en 2020 el Sloan Digital Sky Survey (SDSS) lanzó el análisis comprehensivo del más grande mapa tridimensional del universo hasta ahora, dentro de la data del Bayron Oscillation Spectroscopic Survey (eBOSS), podemos encontrar más de dos millones de galaxias, estrellas y cuásares cubriendo más de once mil millones de años en tiempo cósmico (SDSS.org 2020). Y se están preparando para lanzar el siguiente en 2024, por lo que no hay suficiente mano de obra experta para realizar esta labor a través de inspección visual y se vuelve necesario optar por metodologías basadas en el cómputo de imágenes para automatizar esta labor.

Figura 1

Comparación entre fotografías de galaxia y estrella



Galaxia

Estrella

Nota: galaxia identificada como IC745_H01_412_1000_3 y estrella identificada como GRB0422A. Fuente: Imágenes tomadas del telescopio 1.3M en Nainital, India.

2 Justificación

Las primeras metodologías usadas para la clasificación de objetos en astrofísica fueron basadas en ingeniería de características manuales y descriptores como SIFT y HOG, actualmente, gracias a la reciente disponibilidad de grandes volúmenes de data fotométrica y avances en el poder de procesamiento de las computadoras, los métodos deep learning se han vuelto una de las soluciones favoritas para realizar detección y clasificación de objetos. Es enfoque de este proyecto proponer un ensamble híbrido de técnicas deep learning para abordar el problema de la clasificación.

La combinación de modelos de detección de objetos SSD (Single Shot Multibox Detector) y clasificación de objetos ResNet (Residual NetWork) propuesta para el ensamble híbrido puede probar ser efectiva para la clasificación de estrellas y galaxias, ya que permite la identificación y clasificación precisa de múltiples objetos en una sola imagen del cielo. SSD no remuestrea píxeles o características para las hipótesis de cuadros delimitadores sin sacrificar precisión, por lo que es más veloz en la detección de alta precisión que otros modelos como Faster R_CNN o YOLO (Wei Liu et al 2016) y se encargará de la extracción de objetos individuales de las imágenes del cielo. Por su parte, ResNet permite la construcción de arquitecturas mucho más profundas (hasta 150 capas) lo cual está asociado con una mayor precisión de clasificación (Simonyan & Zisserman 2015). Además, el uso de un enfoque basado en detección de objetos permite una mayor eficiencia en el procesamiento de imágenes al reducir el tamaño de las imágenes a objetos individuales, lo que reduce la cantidad de datos procesados y ruido que no aporta a la clasificación.

Además, este proyecto no solo contribuye a la ciencia de la astrofísica, sino que también demuestra el potencial de los enfoques de aprendizaje profundo para resolver problemas complejos en una amplia variedad de disciplinas científicas. La detección de objetos y la clasificación de imágenes son técnicas fundamentales en muchas aplicaciones prácticas, desde la detección de objetos en imágenes médicas hasta la vigilancia de seguridad en entornos urbanos. Este proyecto demuestra la capacidad de combinar estas técnicas para resolver problemas de clasificación de imágenes en un contexto científico, lo que abre la puerta a futuras investigaciones en esta área.

3 Objetivos

4.1 Objetivo general

Desarrollar un ensamble compuesto de dos modelos: MobileNet Single Shot Multibox Detector para la detección de objetos y Residual Network para la clasificación de imágenes de estrellas y galaxias.

4.2 Objetivos específicos

1. Caracterizar la data disponible extraída de los servidores del Sloan Digital Sky Survey (SDSS).
2. Modelar arquitectura de red neuronal para la detección de objetos basada en MobileNet Single Shot Multibox Detector.
3. Modelar arquitectura de red neuronal para la clasificación de imágenes basada en Residual Network.
4. Conectar los modelos en un ensamble general.
5. Evaluar rendimiento de los modelos por separado y del ensamble completo.

4 Estado del Arte

La identificación precisa de objetos en las imágenes astronómicas ha sido objeto de numerosas investigaciones en los últimos años, en los que la mayoría se basan en técnicas avanzadas de inteligencia artificial. Algunas presentan metodologías de reducción de dimensionalidad y ensambles de técnicas de machine learning como Naive Bayes y Random Forest (Adrian & Flores), o clasificación por medio de redes neuronales convolucionales (Cordero-Chan et al 2018). Entre los enfoques más destacados, se encuentra el estudio de Kim et al. (2015), en el que se presenta un enfoque de aprendizaje híbrido para la clasificación de estrellas y galaxias utilizando un conjunto de modelos basados en árboles de decisión y redes neuronales. Estos enfoques han demostrado ser muy efectivos para la clasificación en astrofísica.

Algunas de los algoritmos basados en deep learning más utilizadas para detección de objetos son Faster R-CNN, YOLO, SSD. Estos utilizan redes neuronales convolucionales y los tres utilizan un marco de detección de objetos de dos etapas que involucra una red de propuesta de región, seguida de clasificación y localización de objetos. Faster R-CNN (Faster Region Concolutional Neural Network) es conocido por su precisión y a menudo se utiliza para tareas como la segmentación de instancias. YOLO (You Only Live Once) es conocido por su velocidad y se utiliza comúnmente en aplicaciones en tiempo real como vehículos autónomos. SSD encuentra un equilibrio entre velocidad y precisión. En cuanto a la tarea de clasificación, los algoritmos más utilizados están basados en Redes Neuronales Convolucionales (RNC) profundas, debido a que aumentar la profundidad de la red es beneficioso para la precisión de clasificación (Simonyan & Zisserman 2015). Sin embargo, las RNC de muchas capas de profundidad suelen encontrarse con el problema del desvanecimiento del gradiente, por lo que se recomienda hacer uso de técnicas como Residual Networks (ResNet) mientras se trabajen con redes de menos de 150 capas.

Actualmente existen varios datasets públicos de data fotométrica astronómica, como el Sloan Digital Sky Survey (SDSS) que se encuentra construyendo uno de los proyectos más ambiciosos de mapeo astronómico, o el Galaxy Zoo que posee un repositorio de clasificación morfológica de galaxias, vale la pena resaltar que Galaxy Zoo utiliza data del SDSS. También existen datasets privados como el Next Generation Virgo Survey (NGVS) y Next Generation Fornax Survey (NGFS).

5 Marco Teorico

5.1 Radiacion de los cuerpos astrofísicos

La astronomía y la astrofísica son ramas del conocimiento ampliamente fundamentadas en la observación, y a diferencia de otras disciplinas de las ciencias físicas, como la física de partículas, la física atómica, o la óptica, que cuentan con entornos controlados en laboratorios para manipular y entender las propiedades de los sistemas estudiados, en el campo de la astronomía esta posibilidad se desvanece. Se está limitado a la observación, más puntualmente el recuento de fotones, ya sea en grandes o pequeñas cantidades con el fin de extraer la mayor cantidad de información posible y así inferir las propiedades físicas de los cuerpos celestes bajo análisis (Muñoz J C 2019).

La radiación proveniente de las fuentes celestes se puede concebir como flujos de fotones en ciertos casos, mientras que en otros, como fuentes de onda que se propagan hacia el observador. Cuando los astrónomos captan esta radiación en la Tierra, generalmente lo hacen mediante la formación de imágenes del objeto emisor. A partir de esta información visual, se determina la cantidad de radiación detectada, habitualmente dentro de rangos específicos de longitudes de onda.

5.1.1 Estrellas

Una estrella es un objeto celeste que brilla debido a la fusión nuclear en su núcleo. Estas inmensas esferas de plasma contienen principalmente hidrógeno y helio, y emiten luz y calor a través de un proceso llamado fusión nuclear. Las estrellas varían en tamaño, temperatura y luminosidad. A través de las observaciones hechas desde telescopios, se estudian y comprenden las estrellas. Las imágenes proporcionan información sobre el tamaño, el color, la forma, entre otros, lo que permite clasificarlas en diferentes categorías, como enanas rojas, gigantes azules o estrellas de neutrones. Además, las fotografías ayudan a medir la temperatura y la luminosidad de las estrellas, lo que a su vez brinda información sobre su edad y evolución.

5.1.2 Galaxias

Una galaxia es un vasto conjunto de estrellas, planetas, gas, polvo cósmico y materia oscura que está unido gravitacionalmente. Estas estructuras cósmicas vienen en diferentes formas y tamaños, desde galaxias espirales con brazos en espiral distintivos hasta galaxias elípticas que son más redondeadas. Las galaxias albergan miles de millones o incluso billones de estrellas, y se cree que contienen agujeros negros supermasivos en sus centros. Mediante las fotografías capturadas por telescopios, podemos explorar y comprender mejor las galaxias. Las imágenes nos revelan detalles fascinantes sobre su estructura y forma, permitiéndonos clasificarlas en diferentes tipos y estudiar su evolución y dinámica. Se estima que existen más de un billón (1.000.000.000.000) de galaxias en el universo (Redacción National Geographic 2023).

5.2 Camaras CCD

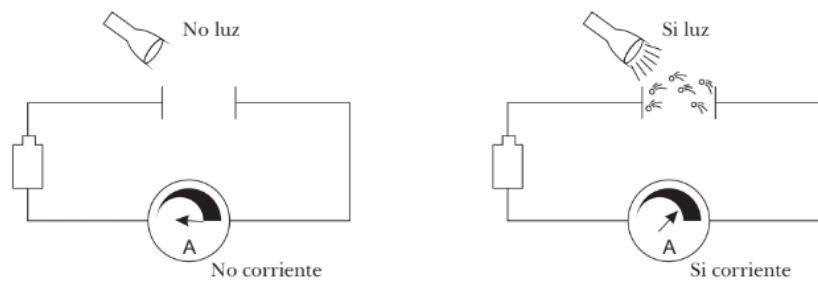
CCD significa dispositivo de carga acoplada. Un CCD es un dispositivo semiconductor que convierte la luz en una señal eléctrica a través del efecto fotoeléctrico. Los CCDs se utilizan en una variedad de aplicaciones de imagen, incluyendo cámaras digitales, videocámaras y escáneres (Definición TechEdu 2023). Actualmente casi todo dispositivo electrónico de formación de imágenes usa un dispositivo CCD, desde cámaras digitales y escáneres hasta cámaras CCD en telescopios espaciales. Su sensibilidad en diferentes longitudes de onda, su amplio ancho espectral, versatilidad y bajo ruido, han convertido a los detectores CCD en el dispositivo predilecto para el trabajo de detección de radiación en astronomía. (Muñoz J C 2019).

5.2.1 Efecto fotoeléctrico.

Cuando una fuente de luz brilla en un metal, algunos electrones pueden ser expulsados de la superficie del metal en un fenómeno conocido como el efecto fotoeléctrico. También se le suele llamar fotoemisión, y a los electrones que son expulsados del metal, fotoelectrones. En términos de su comportamiento y sus propiedades, los fotoelectrones no son diferentes de otros electrones. El prefijo foto simplemente nos indica que los electrones han sido expulsados de la superficie de un metal por la luz incidente (Khan Academy 2022).

Figura 2

Esquema descriptivo del efecto fotoeléctrico



Nota: A la izquierda, con la luz apagada, no hay fotones que incidan sobre las láminas de metal, por tanto, no se observa corriente circulando en el circuito. A la derecha, una vez se enciende la luz (con una longitud de onda adecuada), los fotones arrancan electrones del material induciendo corriente en el circuito. Fuente: “Una introducción a la astronomía práctica” de Muñoz J C 2019.

5.2.2 Reducción de imágenes CCD

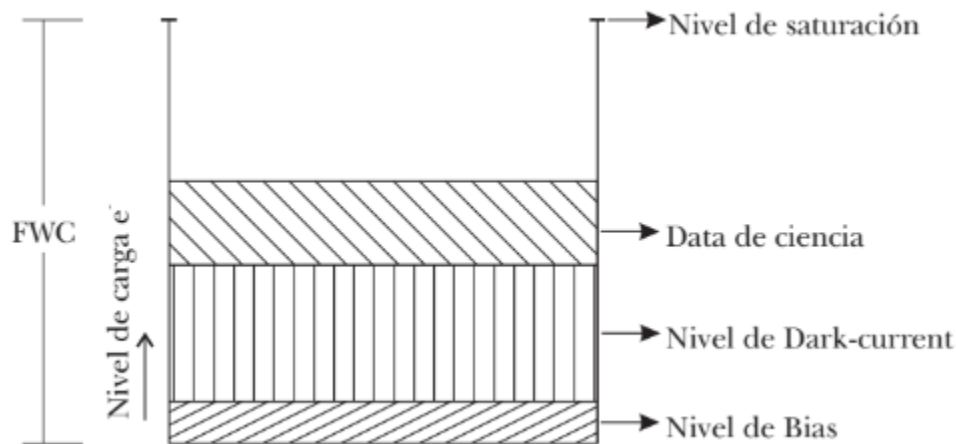
La cámara CCD no es un dispositivo perfecto y, para realizar mediciones precisas con este instrumento, es importante conocer los procedimientos que se pueden seguir para reducir los datos y prepararlos de manera que sean útiles para extraer información científica. En este contexto, se busca identificar, en la práctica, las distintas fuentes de ruido inherentes al proceso de observación con detectores CCD, así como presentar el proceso fundamental para la reducción de imágenes astronómicas capturadas con un CCD.

5.2.2.1 Ruido de lectura

Cuando se trabaja con imágenes astronómicas tomadas con detectores CCD, existen diferentes tipos de ruido que pueden afectar la calidad de las imágenes. Estos ruidos pueden surgir tanto durante la captura de la imagen como durante el proceso de lectura y amplificación de la señal del CCD. En la figura 3 se puede ver la distribución de las cargas acumuladas en un píxel individual, en este los niveles de Dark-current y Bias representan la carga asociada al ruido de la imagen.

Figura 3

Distribución de cargas acumuladas en un píxel



Nota: En la imagen vemos que adicional a la data de ciencia que es útil hay niveles asociados al Dark-current y al Bias, que están generados por ruido térmico y de lectura, respectivamente. Fuente: “Una introducción a la astronomía práctica” de Muñoz J C 2019.

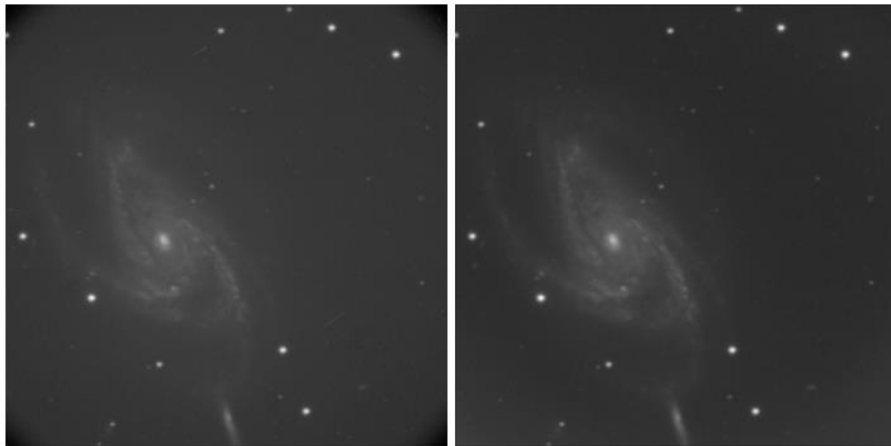
5.2.2.2 Reducción

El proceso de Reducción consiste en construir imágenes de calibración a través del aislamiento de las posibles fuentes de ruido presentes, luego se utilizan estas imágenes de calibración para operar los archivos de imágenes y obtener la data limpia como se aprecia en la figura 4.

Como se puede ver en la figura 3 para reconstruir la información a partir de los datos crudos hay que eliminar, píxel a píxel, la contribución de las diferentes fuentes de ruido. Primero entonces será conveniente sustraer de los datos crudos las cuentas acumuladas durante el proceso de lectura (el ruido de lectura). Esto eliminará entonces la primera contribución de ruido en los datos crudos. Asumiendo que no hay que hacer sustracción de Dark Current (como ocurre en la mayoría de los casos actualmente), la contribución de los termo electrones a las cuentas en un píxel sería cero, lo que quiere decir que una vez removido el Bias las cuentas acumuladas en cada píxel son precisamente la data de ciencia.

Figura 4

Comparación imagen cruda con imagen resultante del proceso de reducción



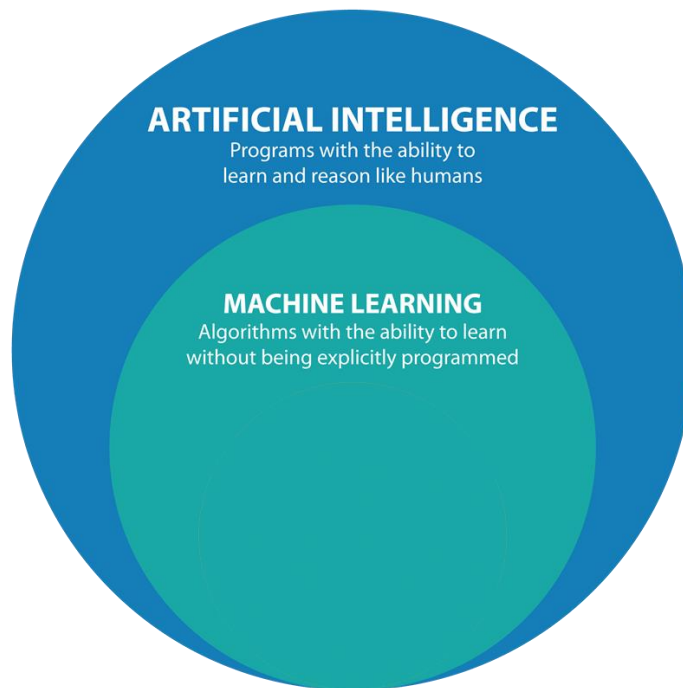
Nota: Una imagen cruda (sin reducir) de la galaxia NGC5054 acompañada de la imagen resultante del proceso de reducción. Note la evidente diferencia entre los datos crudos y los datos de ciencia. Fuente: “Una introducción a la astronomía práctica” de Muñoz J C 2019.

5.3 Machine Learning

El aprendizaje automático se enfoca en utilizar datos y algoritmos para imitar el proceso de aprendizaje humano y mejorar gradualmente su precisión. Es una disciplina que se encuentra dentro de la inteligencia artificial y la ciencia de la computación (IBM Webpage Topics 2022).

Dentro del campo de la ciencia de datos, el aprendizaje automático es fundamental. A través de métodos estadísticos, se entrenan algoritmos para realizar clasificaciones o predicciones, y descubrir ideas clave en proyectos de minería de datos. Estas ideas son utilizadas para tomar decisiones en aplicaciones y negocios, con el objetivo de impactar positivamente las métricas de crecimiento. Con el continuo crecimiento de los grandes volúmenes de datos, se espera un aumento en la demanda de científicos de datos, quienes serán necesarios para identificar las preguntas de negocio más relevantes y encontrar los datos necesarios para responderlas.

Figura 5
Machine Learning en Inteligencia Artificial



Nota: La Inteligencia Artificial se puede definir como programas que tengan la habilidad de aprender y razonar como humanos, dentro de la IA existe el Machine Learning en el que se construyen algoritmos que tienen la capacidad de encontrar patrones sin ser programados explícitamente para encontrar ese patrón en particular. Fuente: Medium – DataDrivenInvestor “Deep Learning, Brief about AI/ML/DL” by Harikrishna B 2018.

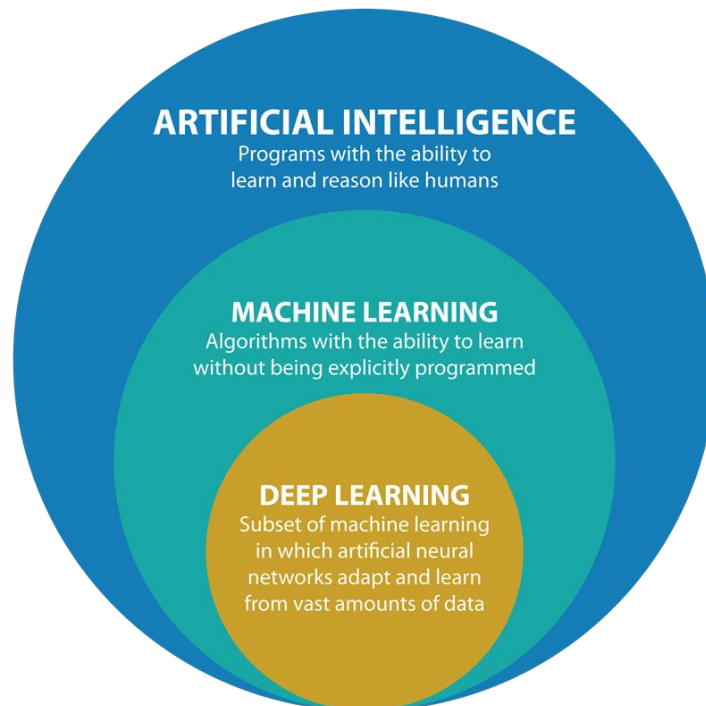
5.4 Deep learning

El aprendizaje profundo es una subcategoría del aprendizaje automático que se basa en redes neuronales con tres o más capas (ver figura 6). Estas redes neuronales intentan emular el comportamiento del cerebro humano, permitiéndoles "aprender" a partir de grandes volúmenes de datos. Mientras que una red neuronal con una sola capa aún puede hacer predicciones aproximadas, la adición de capas ocultas ayuda a optimizar y perfeccionar la precisión (IBM Webpage Topics 2022).

El aprendizaje profundo impulsa numerosas aplicaciones y servicios de inteligencia artificial que mejoran la automatización, llevando a cabo tareas analíticas y físicas sin intervención humana. La tecnología del aprendizaje profundo se encuentra detrás de productos y servicios comunes en nuestra vida diaria (como asistentes digitales, controles remotos de televisión con reconocimiento de voz y detección de fraudes en tarjetas de crédito), así como en tecnologías emergentes (como los vehículos autónomos).

Figura 6

Deep Learning en Inteligencia Artificial



Nota: El Deep Learning o aprendizaje profundo es una subcategoría del machine learning en el cual se utilizan redes neuronales que se adaptan y aprenden de grandes cantidades de data. Fuente: Medium – DataDrivenInvestor “Deep Learning, Biref about AI/ML/DL” by Harikrishna B 2018.

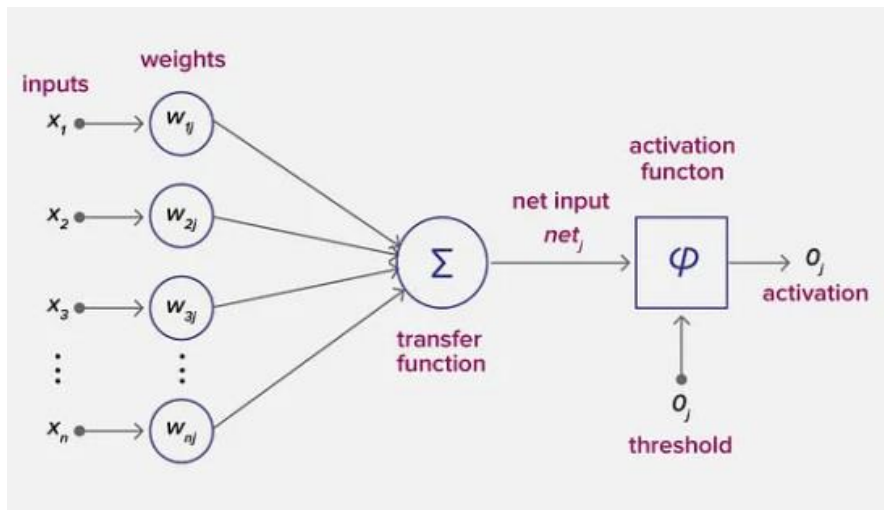
5.4.1 Neurona

Una neurona o nodo es una unidad fundamental en las redes neuronales que recibe información, realiza cálculos sencillos y la transmite a otras partes del sistema. En la figura 7 se explica el funcionamiento de una neurona.

En una red neuronal extensa con numerosos neuronas y conexiones entre ellos, las neuronas se organizan en capas. Existe una capa de entrada que recibe la información, varias capas ocultas y una capa de salida que genera resultados significativos. Cada neurona realiza transformaciones en la información de entrada.

Figura 7

Estructura interna de una Neurona



Nota: De izquierda a derecha se observan los inputs X_n que alimentarán a la neurona, estos inputs pueden ser la información inicial o outputs procesados por otras neuronas de la red. Dentro de la función de la neurona se realiza la sumatoria de la multiplicación de los inputs por un número igual de pesos W_n (Estos pesos son calibrados durante el proceso de entrenamiento), el resultado de esta operación es pasado por una función de activación que transforma la data para obtener un output. Fuente: Medium "How Do Neural Networks Really Work in the Deep Learning?" By Vijay Anandan 2021.

5.4.2 Red Neuronal

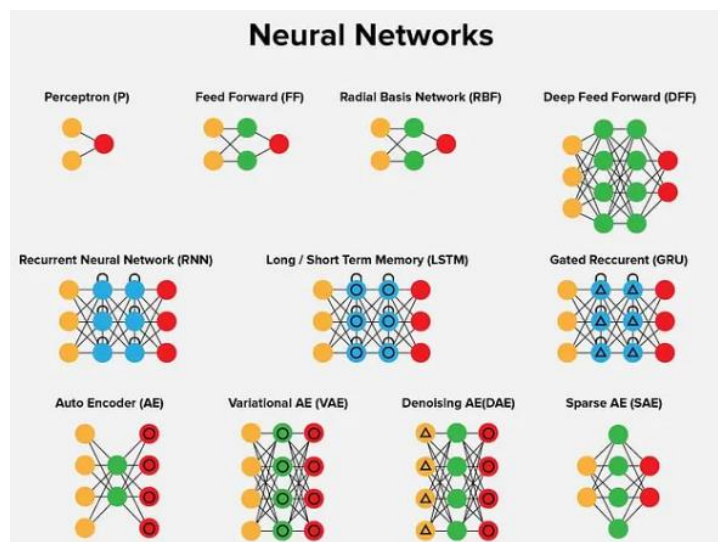
Una red neuronal artificial (RNA) es un modelo computacional que imita la estructura y el funcionamiento del cerebro humano. Está compuesta por un conjunto de neuronas interconectadas a través de sinapsis artificiales, organizadas en capas.

Las RNA son capaces de procesar y analizar grandes volúmenes de datos de manera eficiente. Cada neurona en la red recibe señales de entrada, realiza cálculos y transmite la información hacia las neuronas de la capa siguiente. Esta comunicación se realiza a través de conexiones ponderadas, conocidas como pesos sinápticos, que determinan la influencia de una neurona en otra.

A medida que la información fluye a través de la red, se aplican algoritmos de aprendizaje para ajustar los pesos sinápticos y mejorar el rendimiento de la red. Este proceso permite a las RNA aprender y adaptarse a partir de ejemplos y datos de entrenamiento, lo que les brinda la capacidad de reconocer patrones, realizar predicciones y tomar decisiones.

Las RNA pueden tener diferentes arquitecturas, como redes neuronales feedforward, redes neuronales recurrentes y redes neuronales convolucionales, cada una diseñada para abordar diferentes tipos de problemas y tareas. Estas redes se utilizan en una amplia gama de aplicaciones, incluyendo reconocimiento de imágenes, procesamiento de lenguaje natural, sistemas de recomendación, conducción autónoma y muchas otras áreas de la inteligencia artificial y la ciencia de datos.

Figura 8
Ejemplos de Redes Neuronales Artificiales



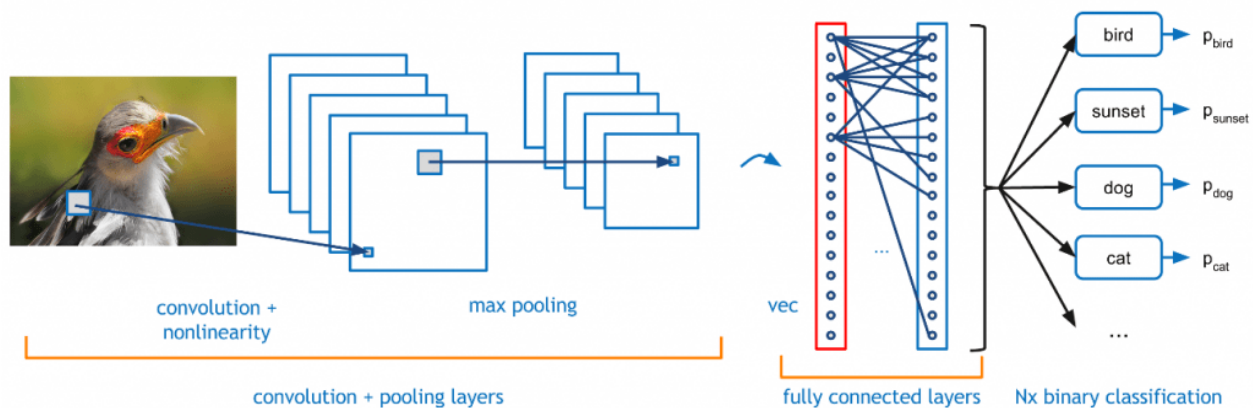
Nota: Grupo de estructuras de redes neuronales artificiales. Fuente: Medium "How Do Neural Networks Really Work in the Deep Learning?" By Vijay Anandan 2021.

5.4.3 Red Neuronal Convolucional RNC

Hay diferentes tipos de redes neuronales que se utilizan para diferentes casos de uso y tipos de datos. Por ejemplo, las redes neuronales recurrentes se usan comúnmente para el procesamiento del lenguaje natural y el reconocimiento de voz, mientras que las redes neuronales convolucionales (ConvNets o CNN) se usan comúnmente para tareas de clasificación y visión por computadora. Antes de la introducción de las CNN, se usaban métodos manuales de extracción de características que requerían mucho tiempo para identificar objetos en imágenes. Sin embargo, las redes neuronales convolucionales han llegado a ofrecer un enfoque más escalable para la clasificación de imágenes y las tareas de detección de objetos al aprovechar los principios del álgebra lineal, específicamente la multiplicación de matrices para identificar patrones en las imágenes. Sin embargo, puede ser computacionalmente intensivo y requiere una unidad de procesamiento de gráficos (GPU) para entrenar el modelo. Las redes neuronales convolucionales son el estándar actual en el aprendizaje automático profundo y se utilizan para resolver la mayoría de los problemas. Las redes neuronales convolucionales pueden ser prospectivas o recurrentes (IBM Webpage Topics 2022).

Figura 9

Ejemplo de estructura de red neuronal convolucional RNC



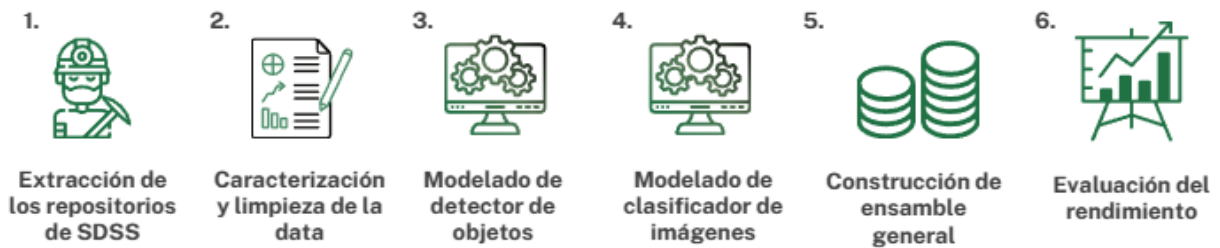
Nota: De izquierda a derecha en la estructura de la RNC se encuentran de primero las capas convolucionales y de max pooling, luego se pasan por capas neuronales conectadas, y finalmente se realiza una clasificación binaria. Fuente: LisDataSolutions, blog técnico: “Deep Learning: clasificando imágenes con redes neuronales” 2022.

6 Metodología

La metodología para la construcción de esta herramienta consta de seis etapas en las que las primeras dos tienen que ver con la preparación de la data, las siguientes abarcan la construcción y evaluación de los modelos de deep learning a utilizar, y las últimas dos incluyen el proceso de ensamblar y evaluar el sistema completo (ver figura 10).

Figura 10

Diagrama de metodología general



Nota: Diagrama se interpreta de izquierda a derecha, comprende el paso a paso general de la metodología utilizada.
Fuente: Autor.

La primera etapa es extraer los datasets a utilizar desde los servidores del Sloan Digital Sky Survey, estos tienen una plataforma para extraer la data por medio de SQL. Los datasets se componen de dos partes, la primera son las imágenes y estas pueden contener múltiples objetos, la segunda es un dataset en CSV con las coordenadas y clasificación de todos los objetos presentes.

La segunda etapa es hacer el proceso de caracterización y limpieza de la data, esta tarea es fundamental para asegurar que la información obtenida en el paso anterior sea lo más precisa y útil posible. Esto implica analizar las imágenes y eliminar cualquier información que no sea relevante para la tarea de clasificación, además de normalizar las imágenes para que tengan una resolución y calidad uniformes para ser alimentadas posteriormente al algoritmo de detección de objetos.

La tercera etapa es construir un algoritmo para detección de objetos, este está basado en MobileNetSSDv2 el cuál a su vez se compone de una RNC de extracción de características y una RNC Sigle Shot Multibox Detector (SSD) para la detección de objetos. Este modelo se encarga de detectar y localizar los objetos en las imágenes de entrada, y produce como salida una lista de coordenadas que indican la ubicación de cada objeto detectado.

El cuarto paso consiste en modelar un clasificador de imágenes que recibe imágenes de objetos individuales y devuelve una categoría en su output, es decir si el objeto detectado es una galaxia o una estrella. Esta es una RNC construida con una arquitectura ResNet, el cuál utiliza una técnica llamada conexión residual para poder usar capas de alta profundidad sin sufrir pérdidas por el desvanecimiento del gradiente. Esta arquitectura consiste en tomar el output de un bloque convolucional y sumarle el mismo input que se utilizó para alimentar el bloque, expresado en forma de función se entiende como $G(x) = F(x) + x$, donde $F(x)$ es nuestro bloque convolucional y x es a la vez el input y la conexión residual.

El quinto paso es construir el ensamble final que conecte los modelos, la idea general es que el output del detector de objetos debe ser transformado para poder usarse como input en el clasificador de imágenes, esta transformación es ejecutada por un Script de Python en el que se toma como input las coordenadas generadas en el modelo anterior para recortar todos los objetos detectados en imágenes individuales estandarizadas, que serán usadas para alimentar el modelo de clasificación.

Por último, en la sexta etapa se evalúa el rendimiento del proyecto. Para ello, se utilizan diversas métricas, tales como la precisión, la sensibilidad y la especificidad. También se generan curvas ROC y cálculo del área bajo la curva (AUC) para evaluar el desempeño general del modelo.

6.1 Extracción, caracterización y Limpieza de la Data

El conjunto de datos utilizado en este proyecto se obtuvo del Sloan Digital Sky Survey (SDSS). Las imágenes recopiladas tienen una resolución de 2048×1489 píxeles, pueden y suelen contener múltiples objetos, y están acompañadas de metadata asociada a las coordenadas de objetos astronómicos previamente identificados, ver tabla 1. La metadata sigue un formato específico que incluye campos como el nombre del archivo, la clase del objeto, las coordenadas RA (Ascensión Recta) y DEC (Declinación) que se utilizan a modo de sistema de coordenadas celestes (este debió ser convertido a coordenadas de pixel para poder ser utilizadas en el modelo), así como los valores i, r y g relacionados con las bandas de color. Para este proyecto, solo se utilizaron los campos del nombre del archivo, la clase y las coordenadas, ya que son los más relevantes para el modelo.

Tabla 1

Ejemplo de metadata asociada al dataset extraído del SDSS

file_name	class	ra	dec	i	r	g	ra_px	dec_px
frame-irg-005598-1-0027	Galaxy	346.69	5.98	21.01	21.41	23.05	1132.48	996.31
frame-irg-005598-1-0027	Galaxy	346.70	5.99	17.81	18.18	26.87	1170.19	903.77
frame-irg-005598-1-0027	Galaxy	346.70	5.99	19.49	20.25	21.50	1211.76	896.90
frame-irg-005598-1-0027	Galaxy	346.70	5.99	22.15	22.80	23.82	1221.07	968.89

Con el fin de preparar las imágenes para su posterior procesamiento, se realiza un recorte de las mismas en 24 partes de tamaño 320×320 píxeles (ver figura 11), estas imágenes serán usadas para el entrenamiento del modelo de detección. Este tamaño de recorte fue seleccionado debido a que coincide con el tamaño de entrada requerido por el modelo de detección de objetos utilizado en el ensamble de inteligencias artificiales. Al dividir las imágenes, se garantizó que cada fragmento tuviera un tamaño adecuado para su procesamiento y se conservara la información relevante de cada región.

Figura 11

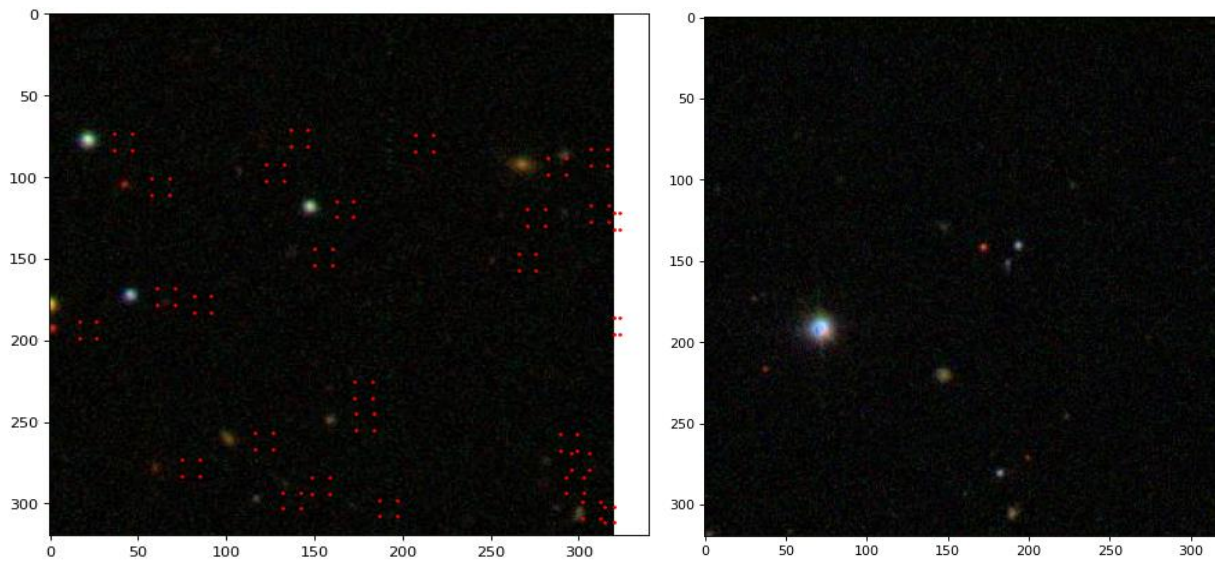
Ejemplo de imagen para entrenamiento del modelo de detección de objetos.



Durante el proceso de preparación de los datos, se identificó la presencia de algunas imágenes que carecían de coordenadas o presentaban desplazamientos en alguno de sus ejes (ver figura 12). Estas imágenes se consideraron inconsistentes y se excluyeron del conjunto de datos final. Como resultado, se obtuvo un conjunto de 1430 imágenes, cada una de las cuales está asociada con su respectiva clase y coordenadas transformadas al formato requerido por el modelo de deep learning utilizado. El nuevo formato de la metadata incluye además de la información que ya tenía el ancho y alto de la imagen, y las coordenadas en formato cuadro delimitador (bounding box) que encierra al objeto en cuestión, como se aprecia en la tabla 2.

Figura 12

Ejemplos de data inconsistente en el dataset extraído



Nota: Ejemplos de data inconsistente encontrada, de izquierda a derecha, imagen con coordenadas presentes pero desfazadas en el eje x, imagen sin ninguna coordenada asociada.

Tabla 2

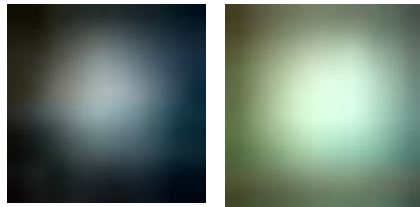
Ejemplo de metadata asociada modificada para adaptarse al input del modelo de detección.

filename	width	height	class	xmin	ymin	xmax	ymax
image_1.jpg	320	320	Star	215.0	109.7	225.0	119.7
image_2.jpg	320	320	Galaxy	165.6	289.0	175.6	299.0
image_3.jpg	320	320	Galaxy	75.5	279.3	85.5	289.3
image_14.jpg	320	320	Galaxy	270.1	292.9	280.1	302.9
image_5.jpg	320	320	Star	273.9	192.5	283.9	202.5

Para el segundo modelo, que es un clasificador de imágenes destinado a distinguir entre galaxias y estrellas, se realizó un proceso adicional utilizando el conjunto de datos anteriormente mencionado. Con el objetivo de generar ejemplos específicos para esta tarea de clasificación, se emplearon las coordenadas contenidas en la metadata de cada imagen para recortar las imágenes de entrenamiento de este modelo, el resultado se puede apreciar en la figura 13. Estos cuadros recortados representan regiones de las imágenes originales que contienen objetos astronómicos de interés, en este caso particular, estrellas y galaxias. Cada cuadro recortado se guardó como una nueva imagen redimensionada de 64×64 que son las dimensiones de entrada del modelo, y se le asignó una nueva metadata, en la tabla 3 se puede ver un ejemplo del contenido de esta. La nueva metadata asociada a los cuadros recortados consta únicamente de dos campos, el nombre del archivo y la clase. De esta manera, se generó un conjunto adicional de 10592 imágenes y metadata específicamente destinados a la tarea de entrenar el modelo de clasificación.

Figura 13

Ejemplo de objetos recortados para entrenamiento de modelo de clasificación



Nota: objetos recortados de dataset original, de izquierda a derecha, estrella y galaxia.

Tabla 3

Ejemplo de metada generada para entrenamiento del modelo de clasificación.

Object	Class
imagen_1.jpg	Star
imagen_2.jpg	Galaxy
imagen_3.jpg	Galaxy
imagen_4.jpg	Galaxy
imagen_5.jpg	Star

Contar con un proceso riguroso de recopilación y preparación de datos es de vital importancia en el desarrollo de modelos de inteligencia artificial. La calidad y la limpieza de los datos influyen directamente en la capacidad de los modelos para aprender y generalizar correctamente. Tener datos limpios garantiza la precisión y confiabilidad de los resultados obtenidos. Al aplicar cuidadosamente el recorte de cuadros basado en las coordenadas de la metadata, se ha asegurado que el conjunto de datos esté compuesto por ejemplos pertinentes y representativos, lo que permite entrenar modelos más efectivos y obtener resultados más precisos en la detección y clasificación de objetos astronómicos en las imágenes capturadas por telescopios.

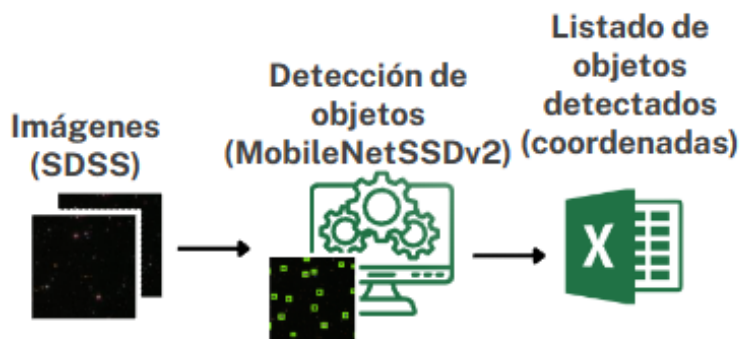
6.3 Modelado de RNC de Detección de Objetos

6.3.1 Descripción del modelo

La arquitectura elegida para el modelo fue MobileNetSSDv2, el cual está dividido en dos partes, la primera es una red neuronal convolucional MobileNetv2 que funciona como red de extracción de características. La segunda parte es un Single Shot Detector (SSD), el cual es una técnica que genera una colección de bounding boxes de tamaños fijos y un puntaje que indica la posibilidad de la presencia de un objeto, por lo que esta parte es la encargada de la detección de objetos. MobileNetSSDv2 funciona usando feature maps de alta resolución y computará un output que consta de coordenadas (expresadas como un bounding box) y una clasificación.

Figura 14

Diagrama general de inputs y outputs del modelo MobileNetSSDv2

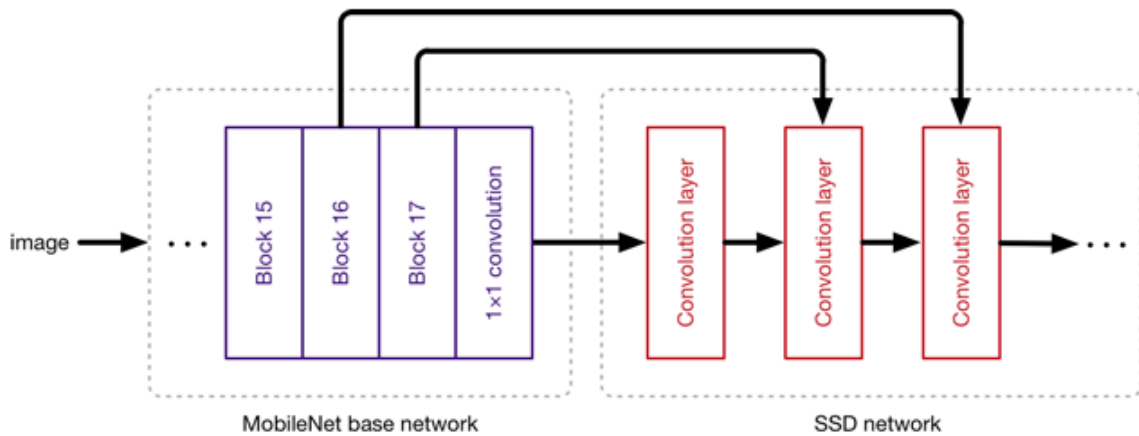


Nota: Este modelo recibirá imágenes tomadas del SDSS en las que se pueden apreciar múltiples objetos captados por el telescopio, estas pasarán a través del modelo MobileNetSSDv2 que genera un output que consta de coordenadas y una clasificación. Fuente: Autor.

6.3.2 Arquitectura del modelo

Para entender la arquitectura de este modelo tenemos que abordarlo en sus dos partes, la primera parte funciona como red de extracción de imágenes (MobileNetv2) y la segunda se encarga de realizar la tarea de identificación de objetos, como se encuentra representado en la figura 15.

Figura 15
Arquitectura RNC MobileNetSDDv2



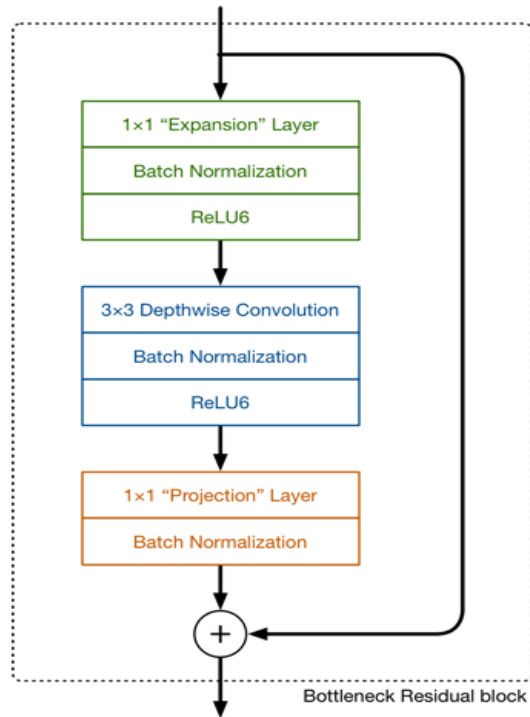
Nota: Estructura de una red neuronal convolucional MobilNetSDDv2 en la que se aprecia la división entre las partes del modelo, a la izquierda se encuentra la red basada en MobileNetv2 para extracción de características y a la derecha la red basada en SSD para detección de objetos y clasificación. La red MobileNet alimenta sus outputs finales y los de múltiples capas anteriores al modelo de detección de objetos SSD, es precisamente es por esto que funciona como extractor de características Fuente: “MobileNet version 2” by Matthijs Hollemans, 2018.

También cabe resaltar que otra característica importante de este tipo de modelo es que no solo tomamos el output generado por el modelo MobileNetv2 para alimentar el modelo de detección SSD, sino que también se usan los outputs de varias capas anteriores del MobileNetv2, el trabajo de estas capas es convertir los pixeles del input en características que describan los contenidos de la imagen y pasarlos a las siguientes capas, actuando entonces como un extractor de características para una segunda red neuronal.

6.3.2.1 MobileNetv2

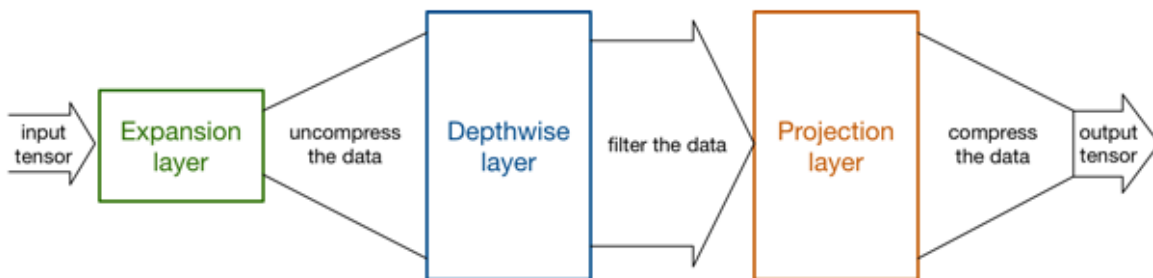
MobileNetv2 consta de 17 bloques de residuo de cuello de botella que a su vez se componen de 3 capas convolucionales con sus respectivos Normalization Batches y funciones de activación Relu como se puede apreciar en la figura 16. La primera capa de expansión funciona como un descompresor que restaura la data (input tensor) a su forma completa y lo realiza mediante una convolución 1×1 (Esto cambia en el primero de todos los bloques, en el que utiliza una convolución 3×3 con 32 canales en lugar de la capa de expansión). Luego la segunda ejecuta el filtrado que sea necesario para la etapa de la red en dónde se encuentre y consta de una convolución 3×3 . Finalmente, la capa de proyección comprime la data de vuelta a un tamaño menor a través de una convolución 1×1 generando el output tensor, tal como se muestra en la figura 17.

Figura 16
Estructura Bloque de Residuo de Cuello de Botella.



Nota: Estructura de un bloque de residuo de cuello de botella utilizado en MobilNetv2, se compone de una capa de expansión, una capa Depthwise, una capa de proyección y una conexión residual que funciona de forma similar a los modelos ResNet. Fuente: “MobileNet version 2” by Matthijs Hollemans, 2018.

Figura 17
Estructura de Bloque de Residuo de Cuello de Botella con funciones explicadas



Nota: Estructura de un bloque de residuo de cuello de botella utilizado en MobilNetv2 con explicaciones fuente: “MobileNet version 2” by Matthijs Hollemans, 2018.

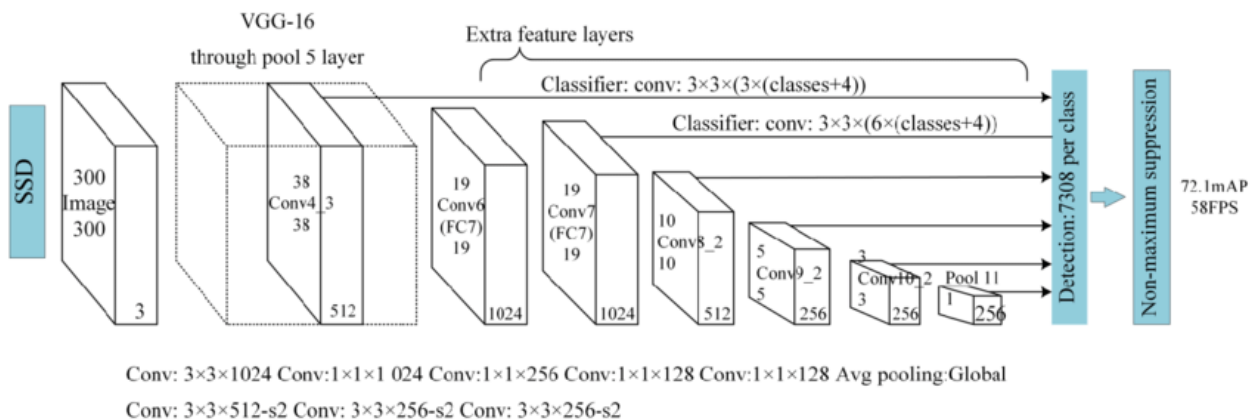
6.3.2.2 SSD

SSD Consta de una RNC Visual Geometry Group (VGG-16) que extrae características y de varias capas de detección que predicen las coordenadas y las clases de los objetos. Luego, las capas de detección se agregan a diferentes escalas de la salida del backbone para detectar objetos a diferentes tamaños (Jianying Cao et al 2020), se puede observar la arquitectura de esta estructura en la figura 18. Cada capa de detección consta de una capa de convolución para generar un conjunto de mapas de características y dos capas adicionales de convolución que predicen las coordenadas de los cuadros delimitadores (bounding boxes) y las clases de los objetos.

El algoritmo consiste entonces de 16 capas conectadas (VGG-16), más una estructura de 15 capas convolucionales que disminuyen de tamaño gradualmente, una capa de Pooling y finalmente se encuentra un non-maximum-supression (NMS) para eliminar detecciones redundantes y generar la lista final de objetos detectados. Este algoritmo funciona a través de crear una cuadrícula sobre la imagen input. Cada cuadrícula tiene una serie de cuadros de tamaño fijo llamados Bounding Boxes, en el caso particular de este proyecto se utilizará $K=2$ Bounding Boxes, cada uno de los cuales predice $C=2$ categorías objetivo (Galaxia o Estrella).

Figura 18

Arquitectura de RNC basada en SSD



Nota: De izquierda a derecha se observa el input: una imagen 300x300, siguiente se encuentra la RNC VGG-16 para extracción de características, siguiente tenemos 5 capas de detección compuestas cada una por 3 capas convolucionales, siguiente a la derecha vemos una capa de Pooling y finalmente se encuentra un non-maximum-supression (NMS) para eliminar detecciones redundantes y generar la lista final de objetos detectados. Fuente: “Target detection algorithm based on improved multi-scale SSD” by Jianying Cae et al 2020.

Entonces el algoritmo de detección de objetos MobileNetSSDv2 es la combinación de los modelos MobileNetv2 y SSD, el primero funciona como un extractor de características para mejorar el rendimiento del segundo que se encarga de detectar y clasificar los objetos presentes en la imagen, este modelo fue pre-entrenado con el dataset `ssd_mobilenet_v2_fpnlite_320x320_coco17_tpu-8`.

6.3.3 Entrenamiento del modelo

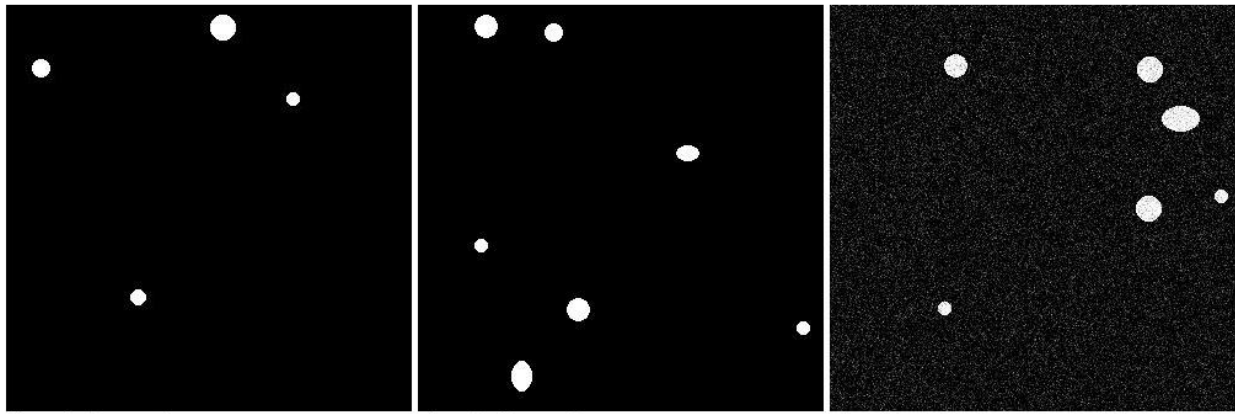
6.3.3.1 Experimentación

Antes de entrenar el modelo de detección de objetos basado en MobileNetSSDv2, se llevó a cabo un proceso de experimentación utilizando datasets "dummy" diseñados específicamente para simular de manera aproximada las características de la data real. El propósito de esta etapa inicial fue evaluar la idoneidad del modelo seleccionado para la tarea de detección de fuentes de luz en imágenes astronómicas. Para ello, se crearon varios datasets con diferentes características, a fin de abordar una variedad de escenarios posibles (ver figura 19). Los datasets "dummy" utilizados en este proceso de experimentación incluyeron: 1) un fondo negro con múltiples círculos blancos distribuidos aleatoriamente y de diversos tamaños, 2) un fondo negro con múltiples círculos y elipses blancos distribuidos aleatoriamente y de diversos tamaños, y 3) un fondo negro con ruido gaussiano junto con múltiples círculos y elipses blancos distribuidos aleatoriamente y de diversos tamaños.

Esta fase de experimentación resultó esencial para evaluar la capacidad del modelo MobileNetSSDv2 para detectar correctamente las fuentes de luz en imágenes astronómicas. Al utilizar datasets "dummy" que imitaban de manera controlada las características esperadas en las imágenes reales, fue posible analizar el rendimiento del modelo en diferentes condiciones y determinar su eficacia en la tarea propuesta, este enfoque de experimentación ayudó a confirmar la idoneidad del modelo MobileNetSSDv2 para la tarea de detección de fuentes de luz en imágenes astronómicas y proporcionó una base sólida para continuar con el proceso de entrenamiento utilizando el conjunto de datos real.

6.3.3.1.1 Generación de Datasets Dummies

Se generaron 3 Datasets de prueba, utilizando las librerías PIL, Numpy, Random, y os. El primer dataset consta de imágenes de fondo negro con entre 3 y 7 círculos de color blanco de distintos tamaños y distribuidos aleatoriamente dentro de la imagen. El segundo dataset es similar al primero, pero este además de círculos, también tiene un número aleatorio de elipses puestas para simular la morfología de una galaxia. Finalmente, el tercer dataset es en esencia igual al segundo ya que trae círculos y elipses sobre un fondo negro, pero se le añade ruido gaussiano al fondo para simular el ruido presente en las imágenes tomadas por telescopios. Todos los datasets constaron de 500 imágenes y fueron usados para entrenar el modelo de MobileNetSSDv2. Adicionalmente a las imágenes, se genera también un archivo CSV con las coordenadas de los objetos generados, sus clases, y el tamaño y nombre de la imagen (ver Tabla 4).

Figura 19*Ejemplo de datasets generados para testeo*Ejemplo Dataset 1:
CírculosEjemplo Dataset 2:
Círculos y ElipsesEjemplo Dataset 3:
Círculos y Elipses + Ruido

Nota: De izquierda a derecha se encuentra una imagen generada para el dataset de prueba “Círculos”, a la derecha se encuentra una imagen generada para el dataset de prueba “Círculos y Elipses”, y por último una imagen generada para el dataset de prueba con ruido gaussiano. Fuente: Autor.

Tabla 4*Ejemplo de dataset de coordenadas generado*

Data Imagen			Coordenadas Objeto				
Filename	Width	Height	Class	xmin	ymin	xmax	ymax
image_1.jpg	330	330	Circulo	109	281	123	295
image_1.jpg	330	330	Circulo	42	260	60	278
image_1.jpg	330	330	Circulo	258	253	276	271
image_1.jpg	330	330	Circulo	240	246	256	262
image_1.jpg	330	330	Circulo	198	215	218	235
image_1.jpg	330	330	Circulo	164	-4	180	12
image_1.jpg	330	330	Elipse	210.5	45	231.5	59
image_1.jpg	330	330	Elipse	20	202.5	38	229.5

Nota: En esta tabla se puede ver la información generada junto con las imágenes, entiéndase cada fila de la tabla como un objeto independiente que se encuentra dentro una imagen. En esta tabla se encuentran los objetos generados en el archivo image_1.jpg, de izquierda a derecha vemos las columnas correspondientes al nombre del archivo, ancho y alto de la imagen, clase del objeto, y las coordenadas x_{min} , x_{max} , y_{min} , y_{max} , donde se encuentra el objeto en la imagen.

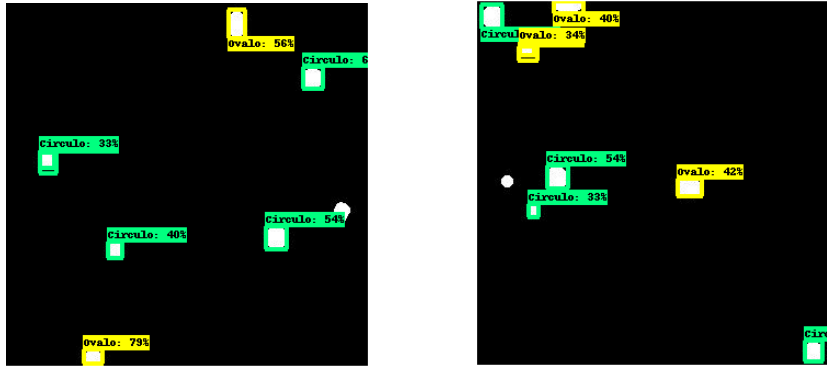
6.3.3.1.2 Resultados Experimentación

Para la evaluación de los resultados de la experimentación del modelo de detección de objetos, se hará uso de varias métricas para medir su desempeño. Estas métricas incluyen pérdida de localización, tasa de aprendizaje, precisión, recuerdo y el índice de intersección sobre unión (IoU). Cada una de estas métricas proporciona información importante sobre diferentes aspectos del modelo y su capacidad para detectar correctamente las fuentes de luz en las imágenes astronómicas. En la figura 20 se presentan ejemplos de las predicciones generadas por el modelo para el dataset generado artificialmente.

1. Pérdida de localización: Es una métrica que mide qué tan precisa es la estimación del modelo de los cuadros delimitadores (bounding boxes) alrededor de los objetos detectados. Se calcula utilizando una función de pérdida específica, como la pérdida de error cuadrático medio (Mean Squared Error, MSE), que compara las coordenadas del cuadro delimitador predicho por el modelo con las coordenadas de referencia o ground truth.
2. Tasa de aprendizaje: Es un hiperparámetro que determina la magnitud de los ajustes realizados en los pesos del modelo durante el entrenamiento. Especifica la velocidad a la que el modelo aprende y se actualiza en cada paso de entrenamiento.
3. Precisión: Es una métrica que evalúa la proporción de detecciones correctas realizadas por el modelo en comparación con todas las detecciones realizadas. Se calcula dividiendo el número de verdaderos positivos (detecciones correctas) entre el número total de detecciones realizadas (*verdaderos positivos + falsos positivos*).
4. Recuerdo: El recuerdo, también conocido como sensibilidad o tasa de verdaderos positivos, mide la capacidad del modelo para recuperar de manera efectiva todas las detecciones positivas verdaderas. Se calcula dividiendo el número de verdaderos positivos entre el número total de positivos verdaderos (*verdaderos positivos + falsos negativos*).
5. Índice de intersección sobre unión (IoU): El IoU, también conocido como Jaccard Index, mide la superposición entre los cuadros delimitadores generados por el modelo y las anotaciones de referencia o ground truth. Se calcula dividiendo el área de intersección entre los cuadros delimitadores por el área de unión de los mismos.

Figura 20

Imágenes con Bounding Boxes construidos a partir de las predicciones



Nota: Imágenes generadas a partir de predicciones del modelo entrenado con el dataset de prueba. Fuente: Autor.

Al observar los resultados obtenidos al final del entrenamiento, se obtuvieron los resultados expuestos en la tabla 5 y en la figura 21. La pérdida de localización es indicador de que el modelo tiene una buena capacidad para estimar con precisión las coordenadas de las fuentes de luz en las imágenes, los valores expuestos en la figura 21 sugieren que el modelo ha aprendido a ajustar correctamente los cuadros delimitadores (bounding boxes) alrededor de las fuentes de luz. La precisión de 0.98 representa la proporción de detecciones correctas realizadas por el modelo en comparación con todas las predicciones realizadas, esto indica que el modelo tiene una capacidad notable para identificar con precisión las fuentes de luz en las imágenes astronómicas. El recuerdo de 0.9 indica la capacidad del modelo para detectar la mayor parte los objetos presentes, es decir en promedio solo el 10% de los objetos no son detectados por el modelo (falsos negativos). Finalmente, el IoU de 0.73 mide la superposición entre los cuadros delimitadores generados por el modelo y las anotaciones de referencia. Un IoU alto indica que los cuadros delimitadores generados por el modelo se superponen en gran medida con las regiones correctas en las imágenes. En conclusión, el modelo muestra alto potencial para detectar de manera efectiva las fuentes de luz en las imágenes astronómicas, lo que respalda su idoneidad para la tarea propuesta, por lo que se toma la decisión de implementarse en el ensamble completo final.

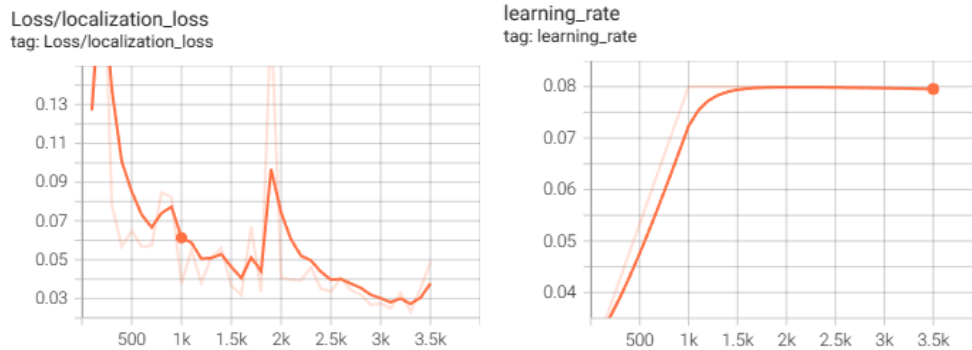
Tabla 5

Tabla resumen de resultados del modelo de detección dataset Dummy

Dataset: Circulos y Elipses con ruido Gaussiano	
Métrica	Valor
Precisión	0.9814
Recuerdo	0.9017
IoU Promedio	0.7324

Figura 21

Resultados pérdida de localización y tasa de aprendizaje dataset Dummy



Nota: A la izquierda está la gráfica que retrata la pérdida de localización del modelo durante el entrenamiento, a la derecha la gráfica que retrata la tasa de aprendizaje del modelo durante el entrenamiento.

6.3.3.2 Entrenamiento con Dataset SDSS

Debido al rendimiento prometedor obtenido con el dataset de prueba, se toma la decisión de proceder a entrenar el modelo utilizando la data del SDSS. Al utilizar este dataset se busca evaluar el rendimiento del modelo en un conjunto de datos más realista y representativo de las imágenes astronómicas reales.

Al analizar los resultados del entrenamiento con la data del SDSS, se observa en la tabla 6 una precisión de 0.9729, un recuerdo de 0.7036 y un índice de intersección sobre unión (IoU) promedio de 0.6031. Si bien la precisión se ha mantenido alta, se observa una disminución en el recuerdo y el IoU en comparación con los resultados anteriores, el localization loss y el learning rate se mantuvieron muy similares a los del modelo entrenado con la data de prueba, esto se puede apreciar en la figura 22.

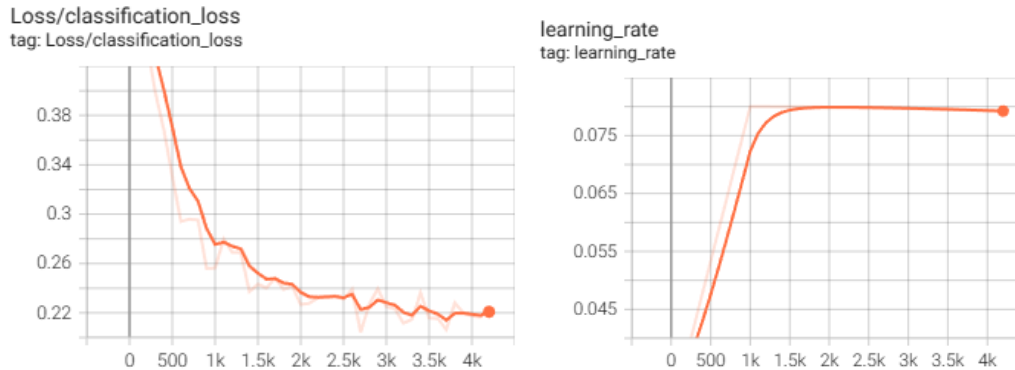
Tabla 6

Tabla resumen de resultados del modelo de detección dataset SDSS vs dataset Dummy

Evaluación Modelo Deteccion		
Métrica	Dummy	SDSS
Precisión	0.9814	0.9729
Recuerdo	0.9017	0.7036
IoU Promedio	0.7324	0.6031

Figura 22

Resultados pérdida de localización y tasa de aprendizaje dataset SDSS



Nota: A la izquierda está la gráfica que retrata la pérdida de localización del modelo durante el entrenamiento, a la derecha la gráfica que retrata la tasa de aprendizaje del modelo durante el entrenamiento.

Varias teorías podrían explicar esta disminución en el recuerdo y el IoU. Una posible causa podría ser la mayor complejidad y variabilidad de las imágenes reales del SDSS en comparación con los conjuntos de datos dummy utilizados previamente. Es posible que las características y patrones de las fuentes de luz en las imágenes reales sean más diversas y complejas, lo que podría afectar la capacidad del modelo para detectarlas con precisión. Otra posibilidad es que el modelo pueda estar sufriendo de sobreajuste a los datos de entrenamiento del SDSS. El sobreajuste ocurre cuando el modelo se vuelve demasiado específico para los datos de entrenamiento y no generaliza bien a nuevas imágenes. Esto podría llevar a una disminución en estas métricas ya que el modelo puede tener dificultades para reconocer y detectar correctamente los objetos que difieren significativamente de las del conjunto de entrenamiento.

Dado que el modelo presenta un rendimiento mixto en las métricas evaluadas, se podría considerar que el modelo tiene potencial, pero también podría requerir mejoras y ajustes adicionales para lograr un rendimiento óptimo. Sería recomendable investigar y experimentar con diferentes arquitecturas de modelos, técnicas de preprocesamiento de imágenes y estrategias de entrenamiento para mejorar la capacidad del modelo para detectar y clasificar con precisión las estrellas y galaxias en las imágenes astronómicas.

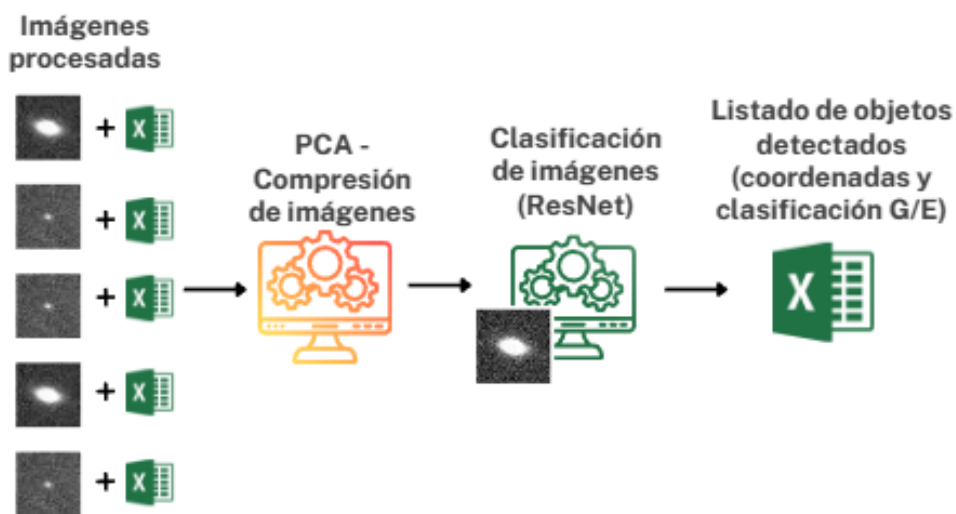
6.4 Modelado de RNC de Clasificación de Imágenes

6.4.1 Descripción del modelo

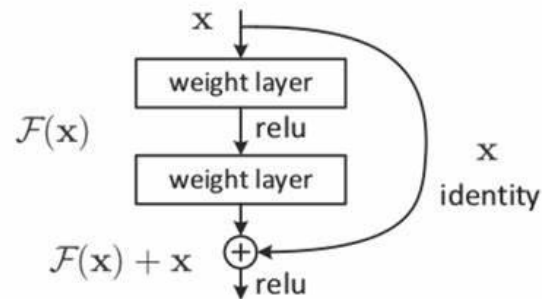
Para este modelo se usa una red neuronal convolucional basada en una arquitectura ResNet, este modelo va a estar encargado de clasificar las imágenes, recibirá de input una imagen de un objeto individual a la que se le ha aplicado previamente reducción de dimensionalidad a través de Principal Component Analysis y le asignará una de dos categorías: galaxias y estrellas, en la figura 23 podemos observar los inputs y outputs del modelo.

Figura 23

Diagrama general de inputs y outputs del modelo ResNet



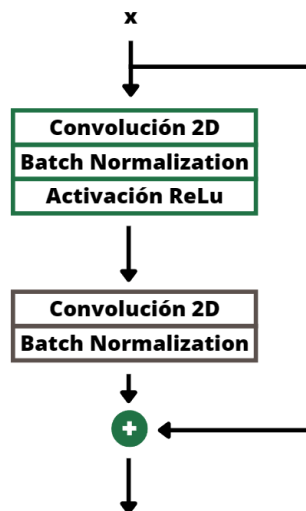
La principal característica distintiva de ResNet es la introducción de bloques residuales en la arquitectura. Estos bloques residuales permiten que la información de la entrada original de una capa pueda ser pasada directamente a una capa posterior, lo que se conoce como "conexión de salto" o "skip connection" (ver figura 24). Esto significa que, en lugar de tratar de aprender la función de mapeo entre las capas, la red puede simplemente aprender a ajustar la salida de una capa para que se parezca a la entrada original. En otras palabras, la capa aprende a identificar la diferencia entre la entrada original y la salida esperada, en lugar de tratar de aprender la salida esperada directamente. Esta conexión de salto ayuda a prevenir el problema de la desaparición del gradiente, que se produce cuando la retropropagación del error a través de muchas capas hace que el gradiente se vuelva cada vez más pequeño. En lugar de depender únicamente del gradiente para actualizar los pesos de la red, ResNet permite que la información fluya directamente a través de las capas, lo que facilita el entrenamiento de redes neuronales profundas.

Figura 24*Estructura de Bloque Residual en ResNet*

Nota: El valor de entrada x pasa a través de las capas convolucionales y además es sumado al output de esas operaciones. Fuente: *Residual Neural Network (ResNet)*, OpenGenus IQ: *Computing Expertise & Legacy Article*, Kandpal 2020.

6.4.2 Arquitectura del modelo

La arquitectura de este modelo está compuesta por dos tipos de bloques que contienen capas convolucionales: bloque de identidad y bloque convolucional. El primero, aparte de aprender la función correspondiente al dominio del problema, ejecuta la conexión de salto que se ve en la figura 25, siendo este el principal componente de una arquitectura ResNet.

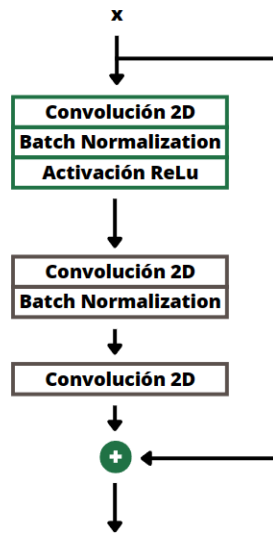
Figura 25*Estructura de Bloque de Identidad*

Nota: Estructura de un bloque de identidad, nótese la conexión de salto sumando al output de las convoluciones. Fuente: Autor.

El segundo bloque del que se apoya la arquitectura es el bloque convolucional y este se encarga de aumentar o disminuir las dimensiones de entrada y salida para que coincidan y se pueda modelar la función de identidad correctamente (Martínez 2019), su estructura la podemos ver en la figura 26.

Figura 26

Estructura de un Bloque Convolucional

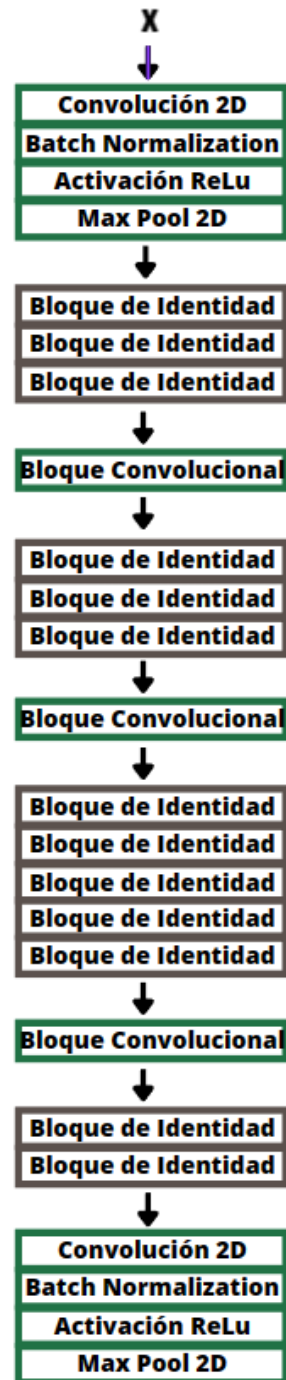


Nota: Funciona de manera similar al bloque de identidad, pero este añade una capa convolucional más para extracción de características. Fuente: Autor

La arquitectura que se utilizó para este proyecto consta de una primera capa base con Batch Normalization, función de activación Relu, y un Max Pool 2d, luego de la capa base se ve 13 bloques de identidad y 3 convolucionales repartidos según se ve en la Figura 23, y finalmente un bloque con Average Pooling 2d, Flatten, Una capa densa con relu y un Soft Max. Para una representación gráfica de la estructura de la arquitectura, ver figura 27

Figura 27

Diagrama de estructura de RNC ResNet utilizada en el modelo



Nota: Diagrama de la estructura utilizada en el modelo clasificador de imágenes basado en ResNet. Fuente: Autor.

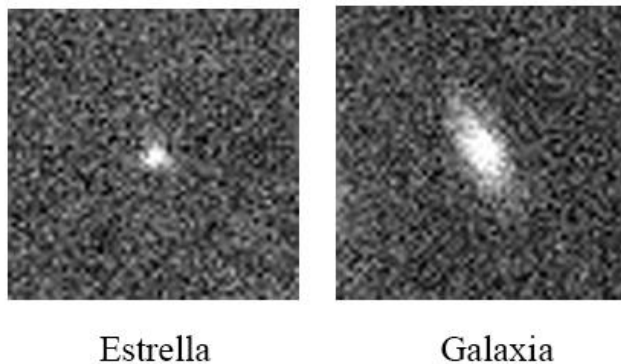
6.4.3 Entrenamiento del modelo

7.4.3.1 Experimentación

Con este modelo también se ejecutó un experimento previo a su testeo con la data definitiva, en este se entrenó con una base de datos tomada del reto de Kaggle: Star-Galaxy Classification Data (ver figura 28), así que es razonable esperar un resultado similar con la data definitiva, ya que el dataset proviene de un desafío específico dedicado a la clasificación de estrellas y galaxias. Esto significa que las imágenes incluidas en el dataset están etiquetadas y se ha realizado una clasificación precisa de las instancias de estrellas y galaxias presentes en ellas. Además, ha sido utilizado por la comunidad científica y los investigadores interesados en el campo de la astronomía y la clasificación de objetos celestes. Esto significa que se ha validado y evaluado previamente, lo que brinda una base sólida para comparar los resultados obtenidos en la experimentación. Se entrenó por 25 epochs y consta de 4000 imágenes con dimensiones 64×64 que fueron pre procesadas a través de un Análisis de Componentes Principales (PCA).

Figura 28

Ejemplo de imágenes del dataset de prueba



Nota: Imágenes ejemplo tomadas del dataset de la competencia de Kaggle: Star-Galaxy Classification Data. Fuente: Imágenes tomadas desde el telescopio 1.3M en Nainital, India.

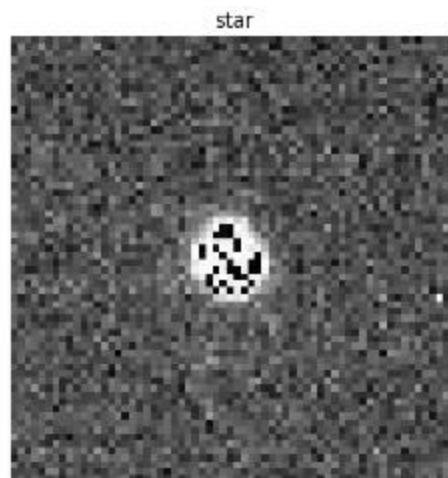
Los resultados del análisis de componentes que se puede apreciar en la tabla 7 indican que con un poco menos de la mitad de componentes se representa el 90% de la varianza acumulada. La capacidad de describir un porcentaje tan alto de la varianza acumulada con un número relativamente pequeño de componentes sugiere que se puede lograr una reducción significativa en la dimensionalidad de los datos sin perder una cantidad significativa de información relevante. Esto resulta positivo ya que puede ayudar a mitigar el ruido o la información redundante en los datos. Al eliminar estos componentes de baja varianza se filtran características menos informativas y se enfoca en aquellas que contribuyen de manera significativa a la variabilidad de los datos (ver figura 29).

Tabla 7
Análisis de Componentes Principales

Principal Components	Varianza Acumulada
20	0.81
21	0.83
22	0.84
23	0.85
24	0.86
25	0.87
26	0.88
27	0.89
28	0.9
29	0.91
30	0.92
...	...
65	1

Nota: Se eligen los primeros 23 componentes principales ya que estos poseen el 85% de la varianza acumulada y una reducción del 64% de los componentes. Fuente: Autor.

Figura 29
Ejemplo de imagen luego de pasar por PCA



Nota: Nótese que se elimina información del centro de la estrella, esto se puede deber a que la parte interna de las estrellas y las galaxias hace son pixeles que mantienen baja varianza y por lo tanto se dejaron de tener en cuenta luego del PCA.

6.4.3.1.1 Resultados Experimentación

Los resultados de la experimentación indican que el modelo tiene buen desempeño en la tarea de clasificación, las métricas utilizadas para evaluar el modelo fueron Validation Loss, Validation Precision y validation AUC, en la Tabla 2 se pueden ver estas métricas con los resultados del modelo.

1. El validation loss es una métrica que representa la cantidad de error o discrepancia entre las predicciones del modelo y los valores reales en el conjunto de datos de validación.
2. La validation precision es una métrica que mide la proporción de instancias clasificadas correctamente como positivas sobre el total de instancias clasificadas como positivas según el modelo en el conjunto de validación.
3. El AUC (Area Under the Curve) es una métrica utilizada para evaluar el rendimiento de modelos de clasificación binaria. Es una medida de la capacidad de discriminación de un modelo para distinguir entre las dos clases de interés. En este caso, las clases serían las estrellas y las galaxias.

Tabla 8

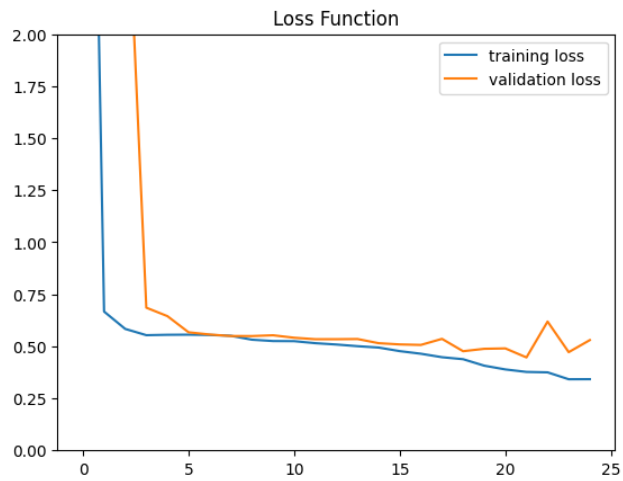
Resultados de modelo ResNet con dataset de prueba

Métrica	Valor
Validation Loss	0.34
Validation Precision	0.84
Validation AUC	0.93

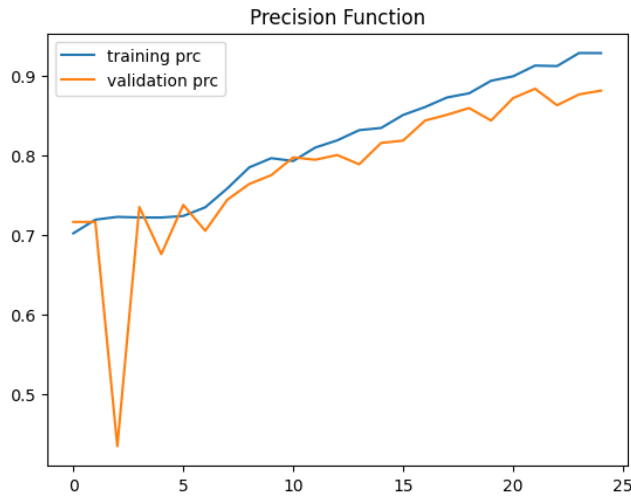
La Validation Loss indica la magnitud del error del modelo en la fase de validación, y un valor bajo de loss indica que el modelo está haciendo predicciones precisas. En este caso, una Validation Loss de 0.34 indica que el modelo está cometiendo relativamente pocos errores. La Validation Precision mide la proporción de predicciones positivas que son correctas, lo que significa que una precisión de 0.84 indica que el modelo está acertando en la mayoría de las predicciones positivas que hace. Por último, la Validation AUC es una medida de la capacidad del modelo para distinguir entre las clases positivas y negativas, y un valor de 0.93 indica que el modelo está clasificando bien las muestras en función de su clase. Las gráficas de estas métricas se encuentran en las figuras 30, 31, y 32.

Figura 30

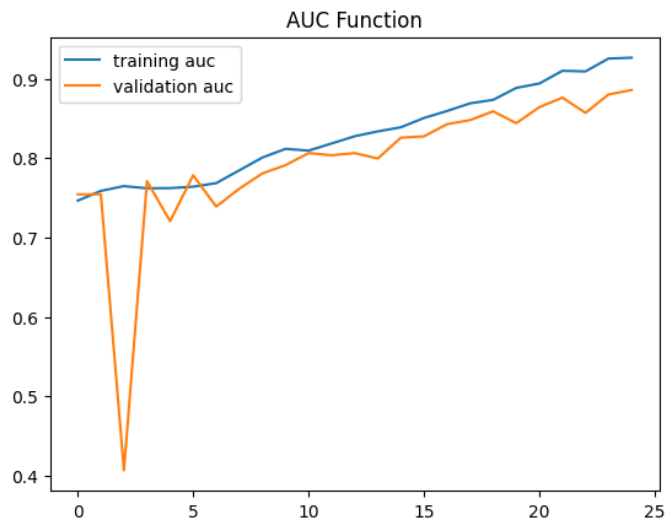
Gráficas de métricas de evaluación de modelo ResNet con dataset de prueba, Loss Function

**Figura 31**

Gráficas de métricas de evaluación de modelo ResNet con dataset de prueba, Precision Function

**Figura 32**

Gráficas de métricas de evaluación de modelo ResNet con dataset de prueba, AUC



En base a estos resultados, es una buena opción proceder a entrenar el modelo con el dataset del SDSS. Los resultados obtenidos en la validación muestran un buen rendimiento y sugieren que el modelo tiene el potencial de generalizar y realizar predicciones precisas en un conjunto de datos más amplio y diverso. Al entrenar con el dataset real, se espera que el modelo tenga un comportamiento similar al presentado en los resultados del experimento.

6.4.3.2 Entrenamiento con Dataset SDSS

Dado el éxito de los resultados obtenidos en la experimentación previa, se procede a entrenar el modelo ResNet con la data extraída y modificada del dataset del SDSS (ver figura 13 y tabla 3)

Los resultados del entrenamiento con el dataset del Sloan Digital Sky Survey (se encuentran en la tabla 9 y en las figuras 33, 34 y 35) muestran un validation loss de 0.45, una validation precision de 0.75 y una validation AUC de 0.86. Estos valores reflejan un rendimiento satisfactorio y positivo del modelo al realizar la tarea de clasificación, aunque podemos observar que ha habido un incremento en el valor del validation loss y una disminución en la validation precision y validation AUC. El aumento en el validation loss puede deberse a diversos factores. Podría indicar que el modelo está teniendo dificultades para ajustarse a los datos del SDSS, posiblemente debido a la complejidad de las imágenes astronómicas reales y las variaciones presentes en ellas. Además, podría implicar que el modelo está experimentando cierto grado de sobreajuste. La disminución en la validation accuracy y la validation AUC da a entender que el modelo está cometiendo más errores en la clasificación de estrellas y galaxias en comparación con los resultados anteriores. Esto puede ser causado por varios factores, como la presencia de imágenes más desafiantes en el dataset del SDSS, variaciones en la iluminación o el ruido en las imágenes.

Es importante tener en cuenta que estos cambios en las métricas no necesariamente indican un empeoramiento general del modelo. En cambio, pueden reflejar una mayor complejidad en los datos reales del SDSS y ofrecer oportunidades para ajustar y mejorar aún más el modelo. Es crucial analizar en detalle los errores cometidos por el modelo y considerar ajustes en la arquitectura o los hiperparámetros del modelo para mejorar su rendimiento.

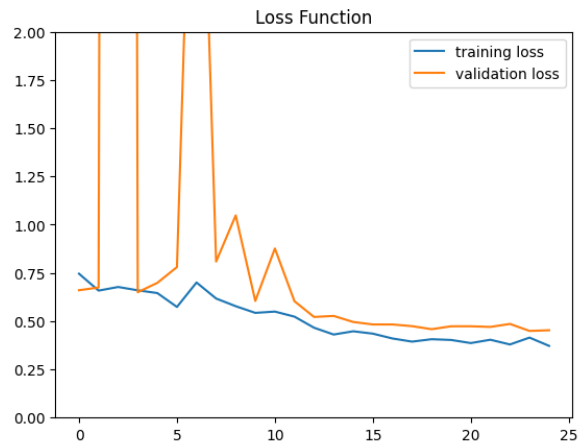
Tabla 9

Resultados de modelo ResNet con dataset de SDSS

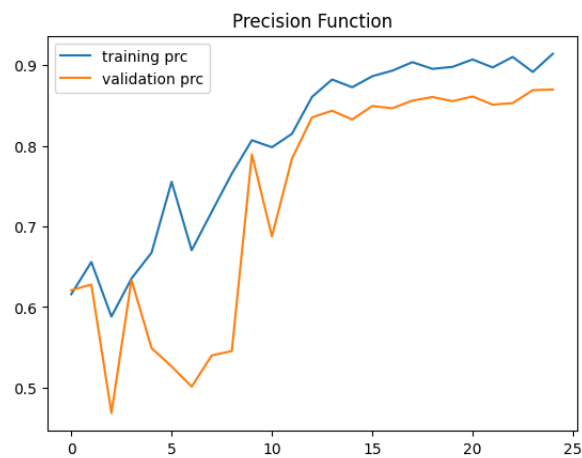
Métrica	Valor
Validation Loss	0.4529
Validation Precision	0.7547
Validation AUC	0.8673

Figura 33

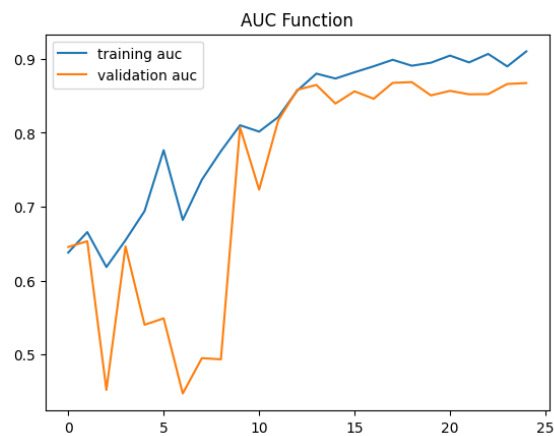
Gráficas de métricas de evaluación de modelo ResNet con dataset de SDSS, Loss Function

**Figura 34**

Gráficas de métricas de evaluación de modelo ResNet con dataset de SDSS, Precision Function

**Figura 35**

Gráficas de métricas de evaluación de modelo ResNet con dataset de SDSS, AUC

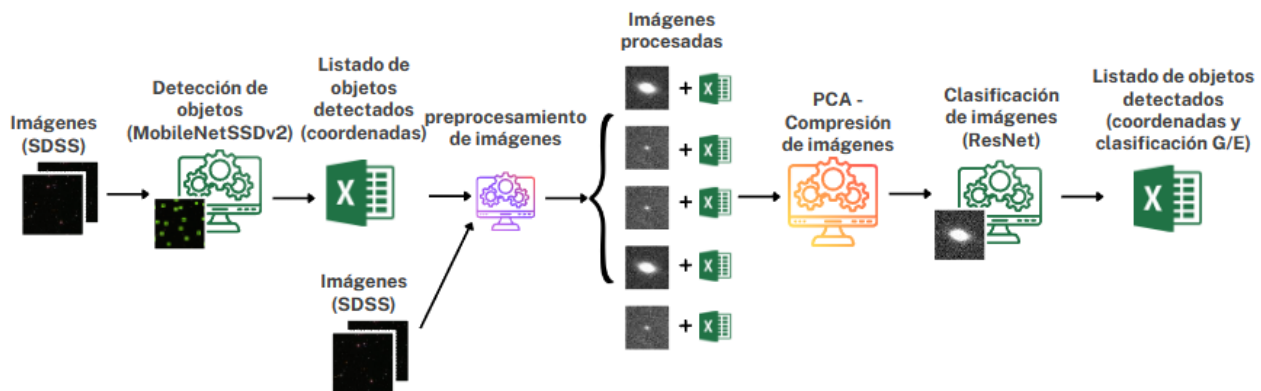


6.5 Ensamblado de sistema híbrido

Ahora como ambos modelos de deep learning se encuentran entrenados, se tienen construidas por separado todas las piezas del sistema, en esta etapa de la metodología se conectan los modelos a través de Notebooks en Colab y se prepara para que sea ejecutable y se le pueda alimentar imágenes del SDSS.

Figura 36

Diagrama del ensamble para detección y clasificación de estrellas y galaxias



Nota: En este diagrama se retrata el flujo de la data a través del sistema y de los modelos hasta convertirse en predicciones de categoría para los objetos en la imagen. Fuente: Autor.

En la Figura 36 se ve el viaje que hace la data de input a través de todo el sistema y de ambos modelos hasta convertirse en predicciones para los posibles objetos en las fotos, inicialmente se le alimenta de data fotométrica del espacio, imágenes de 320×320 transformadas en tensores que pasa por el algoritmo detector de objetos basado en MobileNetSSDv2 y este genera una lista de hasta 100 objetos detectados, parte de los valores más relevantes del output son:

- File name: Nombre original del archivo alimentado al modelo
- Detection Score: es una medida que indica la confianza o certeza con la que el modelo predice la presencia de un objeto en una determinada región de la imagen. Cada detección realizada por el modelo tiene asociado un detection score que refleja la probabilidad de que la detección sea correcta.
- Coordenadas del bounding box (Xmin, Ymin, Xmax, Ymax)

Esta información es tomada del output del primer modelo, se le añade una etiqueta única a cada objeto detectado y se organiza en un Data Frame que es exportado en formato csv, la estructura de este documento se puede observar en la tabla 10, este será utilizado al momento de evaluar el ensamble completo y también para recortar de las imágenes originales los objetos detectados por el modelo, se utilizan las coordenadas junto con la librería PIL para recortar cada uno de los objetos detectados por el algoritmo de detección.

Tabla 10

Ejemplo formato de file CSV output del modelo de detección de objetos

original_filename	object_name	xmin	ymin	xmax	ymin	detection_score
image_1001.jpg	objeto_1.jpg	294.22	218.92	304.05	229.29	0.69
image_1001.jpg	objeto_2.jpg	19.91	246.37	29.52	256.39	0.58
image_1001.jpg	objeto_3.jpg	0.75	87.62	11.05	98.22	0.55
image_1001.jpg	objeto_4.jpg	196.63	314.51	206.51	319.81	0.53
image_1001.jpg	objeto_5.jpg	114.89	174.97	124.43	185.01	0.43

Luego a estas imágenes se les hace reducción de dimensionalidad a través del método Principal Components Analysis, para ser alimentadas al modelo clasificador de imágenes basado en ResNet, este procesa las imágenes y genera un output que consta de una tupla de dos números donde cada uno representa la probabilidad de pertenecer a cada clase disponible (estrella o galaxia), el formato de este output se puede observar en la tabla 11.

Tabla 11

Ejemplo output de modelo de clasificación de imágenes

Classification
(0.8342 , 0.1658)
(0.7725 , 0.2275)
(0.3284 , 0.6716)
(0.4452 , 0.5548)
(0.9231 , 0.0769)

Esta información es suficiente para concluir sobre el modelo, pero antes debe ser procesada y conectada con la información generada por el modelo de detección, es decir que en un solo Data Frame se tenga la información sobre la clasificación y ubicación en píxeles de cada objeto, de esta forma se puede realizar una validación del ensamble de forma general a través de un solo Data Frame que consolide toda la información de un objeto particular. El formato del output definitivo del modelo se puede observar en la tabla 12.

Tabla 12*Ejemplo de formato de output definitivo del ensamble completo*

object name	original filename	xmin	ymin	xmax	ymin	detection score	class
objeto_1.jpg	image_1001.jpg	294.22	218.92	304.05	229.29	0.69	Galaxy
objeto_2.jpg	image_1001.jpg	19.91	246.37	29.52	256.39	0.58	Galaxy
objeto_3.jpg	image_1001.jpg	0.75	87.62	11.05	98.22	0.55	Star
objeto_4.jpg	image_1001.jpg	196.63	314.51	206.51	319.81	0.53	Galaxy
objeto_5.jpg	image_1001.jpg	114.89	174.97	124.43	185.01	0.43	Galaxy

Teniendo esta información ya, es posible evaluar el ensamble completo comparando con la data de referencia que se extrajo junto con las imágenes del dataset del SDSS (ver tabla 2) y así determinar qué clase de rendimiento está presentando y qué tan eficaz es para realizar la tarea en cuestión.

6.6 Evaluación del modelo

Para poder comparar las predicciones del ensamble es necesario relacionar cada línea del output con un objeto de referencia, el que se encuentre más cerca. Para realizar esto se hace uso de las coordenadas para calcular el índice de intersección sobre unión (IoU) de cada predicción con el objeto de referencia más cercano, y se define como un verdadero positivo si el área de intersección es significativa (mayor al 40%) y si la clase asignada por el modelo coincide con el del objeto de referencia, esta relación se ve representada en la tabla 13. Se ejecuta el ensamble con una muestra de 240 imágenes nuevas en las que existen aproximadamente 5700 fuentes de luz para clasificar.

Tabla 13

Ejemplo de formato para evaluación del ensamble completo

Columna	Ejemplo
object_name	objeto_1.jpg
original_filename	image_1001.jpg
xmin_pred	294.22
ymin_pred	218.92
xmax_pred	304.05
ymax_pred	229.29
detect_score	0.69
class_pred	Galaxy
xmin_gt	292.77
ymin_gt	220.98
xmax_gt	302.77
ymax_gt	230.98
class_gt	Star
iou	0.5429
eval_detect	Verdadero Positivo
eval_classification	Falso Positivo
eval_model	Falso Positivo

Nota: la terminación pred indica que la columna tiene valores de las predicciones del modelo, la terminación gt indica que la columna tiene valores de los objetos de referencia (ground truth). Eval detect es verdadero positivo si el IoU es mayor a 0.4. Eval classification es verdadero positivo si class_pred es igual a class_gt. Eval model es verdadero positivo si y solo si eval detect y eval classification son ambos verdaderos positivos.

6.6.1 Métricas de Evaluación

En este proyecto, se han seleccionado varias métricas clave para evaluar el desempeño del ensamble, por tareas y en conjunto completo:

1. **Precisión de detección:** mide la proporción de detecciones positivas que son realmente correctas, lo que nos indica la confiabilidad del modelo a la hora de detectar fuentes de luz en las imágenes proporcionadas.
2. **Recuerdo de detección:** mide la capacidad del modelo para detectar todos los objetos relevantes presentes en los datos. Cuanto mayor sea el recuerdo, menor será la probabilidad de pasar por alto objetos importantes.
3. **IoU Promedio:** el índice de intersección sobre unión promedio nos da una medida de la superposición entre las cajas predichas y las cajas de referencia. Un IoU promedio alto indica una mayor precisión en la delimitación de los objetos.
4. **Precisión de clasificación:** es una métrica que evalúa la capacidad del modelo para clasificar correctamente los objetos detectados. Especifica la proporción de predicciones positivas de clasificación que son verdaderamente correctas.
5. **Precisión del ensamble completo:** es una métrica especial que se aplica cuando se evalúa la detección y clasificación simultáneas de los objetos. Esta métrica solo considera una predicción como un verdadero positivo si se clasifica correctamente y se detecta correctamente al mismo tiempo. Por lo tanto, esta métrica proporciona una medida más estricta y precisa del rendimiento del modelo en términos de detectar y clasificar correctamente los objetos.

7 Resultados

En la etapa de evaluación utilizando las imágenes de prueba, se obtuvieron los resultados presentados en la tabla 14, Los valores altos de precisión de detección indican que el modelo logró clasificar correctamente la gran mayoría de los objetos detectados. Sin embargo, el recuerdo fue de aproximadamente 70%, lo que sugiere que el modelo no pudo detectar todos los objetos presentes en las imágenes de prueba. El valor promedio de IoU fue de 60%, lo que indica una superposición moderada entre las predicciones y las ubicaciones reales de los objetos.

En cuanto a la precisión del ensamble completo, se obtuvo un valor de 52%. Esta métrica considera tanto la detección como la clasificación simultánea de los objetos. La disminución significativa en la precisión de detección al trabajar con el ensamble completo puede ser atribuida a la dificultad de lograr una alta precisión tanto en la detección como en la clasificación de los objetos al mismo tiempo. También es muy posible que se deba a sobreajuste y haya que implementar técnicas para mitigarlo.

Tabla 14

Resultados Generales del ensamble completo

Ensamble Completo	
Precisión de detección	0.9729
Recuerdo	0.7036
IoU Promedio	0.6031
Precisión de clasificación	0.5521
Precisión del ensamble completo	0.524

Conclusiones

Basándonos en los resultados obtenidos, podemos concluir que el modelo diseñado para la tarea de detección y clasificación de objetos ha demostrado cierta eficacia. La precisión de detección individual fue alta, lo que indica que el modelo es capaz de detectar objetos con precisión en las imágenes de prueba. Sin embargo, la precisión del ensamble completo, que considera tanto la detección como la clasificación, fue más baja, que sugiere que existen desafíos en la capacidad del modelo para realizar ambas tareas simultáneamente con altos niveles de precisión. Aunque el modelo muestra cierta utilidad en la tarea de detección de objetos, es evidente que se requieren mejoras para mejorar su rendimiento en la clasificación y en el ensamble completo. La discrepancia en los resultados entre los modelos cuando fueron testeados individualmente y el ensamble completo indica que hay margen para optimizar la integración de ambos componentes y mejorar la precisión global del modelo. Es importante considerar los resultados actuales como un punto de partida y utilizarlos para iterar y mejorar el modelo en futuros proyectos. Con las mejoras adecuadas, el modelo podría volverse más efectivo y funcional para la tarea específica para la cual fue diseñado.

Recomendaciones

Para mejorar la precisión de soluciones que intenten acercamientos similares al presentado en este proyecto, se recomienda considerar las siguientes estrategias. Primero, podría ser útil revisar y ajustar los criterios utilizados para la clasificación de los objetos detectados, asegurando una mayor precisión en la asignación de etiquetas. Además, se podría explorar el uso de técnicas de refinamiento de detección y clasificación, como el uso de modelos de post-procesamiento o la incorporación de información contextual adicional. También se recomendaría aumentar el tamaño y la diversidad del conjunto de datos de entrenamiento para capturar una mayor variabilidad de objetos y escenarios. Esto podría ayudar al modelo a aprender patrones más robustos y generalizables y evitar el posible sobreajuste presentado.

Referencias

Cordero-Chan, R., Orozco-del-Castillo, M. G., Moreno-Sabido, M. R., Hernández-Gómez, J. J., Cetzal-Balam, G., & Couder-Castaneda, C. (2018). Clasificación de galaxias utilizando procesamiento digital de imágenes y redes neuronales artificiales. *Res. Comput. Sci.*, 147(7), 295-307.

Elías, H. A. G., & Conde, O. F. Clasificación Morfológica de Galaxias Mediante Métodos de Reducción de Dimensionalidad y Ensamble de Clasificadores. : Jianying Cao et al 2020 *J. Phys.: Conf. Ser.* 1570 012014

Dalal, N., Triggs, B., 2005. Histograms of oriented gradients for human detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), Vol. 1. pp. 886–893 vol. 1. doi:10.1109/CVPR.2005.177.

Muñoz J.C. (2019). Una Introducción a la Astronomía Práctica. Instituto de física de la Universidad de Antioquia. Medellín – Colombia.

Sloan Digital Sky Survey Publications (2020). Publicaciones de Surveys de mapeo del cielo. <https://www.sdss.org/science/publications/>

Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu1, Alexander C. Berg (2016). SSD: Single Shot Multibox Detector. arXiv:1512.02325v5

Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: NIPS. (2015).

Edward J. Kim, Robert J. Brunner, and Matias Carrasco Kind (2015). A hybrid ensemble learning approach to star–galaxy classification. *MNRAS* 453, 507–521 (2015).

Jianying Cao, Yan Kong, Xinlu Zhang, Yongjia Li and Xiaofeng Xie (2020). Target detection algorithm based on improved multi-scale SSD. *J. Phys.: Conf. Ser.* 1570 012014

Kandpal Apoorva. (2020). Residual Neural Network (ResNet). OpenGenus IQ. <https://iq.opengenus.org/residual-neural-networks/>

Holleman, M. (2018). MobileNet Version 2. Machinethink.net [MobileNet version 2 \(machinethink.net\)](https://www.machinethink.net/mobilenet-version-2/)

Martinez, J. (2019). Toma un atajo con ResNets. DataSmarts.net
[Toma un Atajo con ResNets - DataSmarts Español](#)

Mehta, Vidish. (2021). Object Detection using SSD MobilNetv2. Vidishmeta204.medium.com.
[Object Detection using SSD Mobilenet V2 | by Vidish Mehta | Medium](#)

Redacción National Geographic (13 Abril 2023). ¿Cuántas galaxias conocidas hay en el universo?
<https://www.nationalgeographicla.com/espacio/2023/03/cuántas-galaxias-conocidas-hay-en-el-universo>

Redacción IBM Topics (2022). What is Machine Learning?
<https://www.ibm.com/topics/machine-learning>

Redacción IBM Topics (2022). What is Deep Learning?
<https://www.ibm.com/topics/deep-learning>

Harikrishna B. (2018). Brief about AI/ML/DL.
[Deep Learning. Brief about AI / ML / DL | by Harikrishna B | DataDrivenInvestor](#)

Vijay Anandan (2021). How Do Neural Networks Really Work in the Deep Learning?
<https://medium.com/analytics-vidhya/how-do-neural-networks-really-work-in-the-deep-learning-72f0e8c4c419>

LisDataSolutions, blog técnico (2022). Deep Learning: clasificando imágenes con redes neuronales.
<https://www.lisdatasolutions.com/es/blog/deep-learning-clasificando-imagenes-con-redes-neuronales/>

Khan Academy, Artículo (2022). Efecto Fotoeléctrico.
<https://es.khanacademy.org/science/ap-chemistry/electronic-structure-of-atoms-ap/bohr-model-hydrogen-ap/a/photoelectric-effect>

TechEdu (2023). Definición de CCD.
<https://techlib.net/techedu/ccd/>