



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

MODELADO DE UNA RED NEURONAL ARTIFICIAL PARA
RECONOCIMIENTO DE SUELOS UTILIZANDO DEEP LEARNING

FIGUEROA YAGUANA KEVIN ADRIAN
INGENIERO DE SISTEMAS

MACHALA
2022



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

MODELADO DE UNA RED NEURONAL ARTIFICIAL PARA
RECONOCIMIENTO DE SUELOS UTILIZANDO DEEP
LEARNING

FIGUEROA YAGUANA KEVIN ADRIAN
INGENIERO DE SISTEMAS

MACHALA
2022



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

TRABAJO TITULACIÓN
PROPUESTAS TECNOLÓGICAS

MODELADO DE UNA RED NEURONAL ARTIFICIAL PARA RECONOCIMIENTO
DE SUELOS UTILIZANDO DEEP LEARNING

FIGUEROA YAGUANA KEVIN ADRIAN
INGENIERO DE SISTEMAS

RIVAS ASANZA WILMER BRAULIO

MACHALA, 23 DE FEBRERO DE 2022

MACHALA
2022

tesis kevin figueroa

INFORME DE ORIGINALIDAD

5%

INDICE DE SIMILITUD

5%

FUENTES DE INTERNET

1%

PUBLICACIONES

2%

TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1 Submitted to Universidad Técnica de Machala 2%
Trabajo del estudiante

2 depositonce.tu-berlin.de 1%
Fuente de Internet

3 Submitted to Consorcio CIXUG <1%
Trabajo del estudiante

4 www.mdpi.com <1%
Fuente de Internet

5 doctorpenguin.com <1%
Fuente de Internet

6 dspace.uclv.edu.cu <1%
Fuente de Internet

7 www.risti.xyz <1%
Fuente de Internet

8 repositorio.utmachala.edu.ec <1%
Fuente de Internet

9 Jang Zern Tsai, Rong-Seng Chang, Tung-Yen Li. "Detection of Gap Mura in TFT LCDs by the <1%

CLÁUSULA DE CESIÓN DE DERECHO DE PUBLICACIÓN EN EL REPOSITORIO DIGITAL INSTITUCIONAL

El que suscribe, FIGUEROA YAGUANA KEVIN ADRIAN, en calidad de autor del siguiente trabajo escrito titulado MODELADO DE UNA RED NEURONAL ARTIFICIAL PARA RECONOCIMIENTO DE SUELOS UTILIZANDO DEEP LEARNING, otorga a la Universidad Técnica de Machala, de forma gratuita y no exclusiva, los derechos de reproducción, distribución y comunicación pública de la obra, que constituye un trabajo de autoría propia, sobre la cual tiene potestad para otorgar los derechos contenidos en esta licencia.

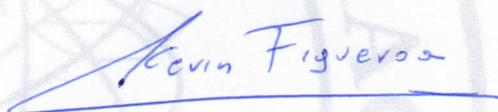
El autor declara que el contenido que se publicará es de carácter académico y se enmarca en las disposiciones definidas por la Universidad Técnica de Machala.

Se autoriza a transformar la obra, únicamente cuando sea necesario, y a realizar las adaptaciones pertinentes para permitir su preservación, distribución y publicación en el Repositorio Digital Institucional de la Universidad Técnica de Machala.

El autor como garante de la autoría de la obra y en relación a la misma, declara que la universidad se encuentra libre de todo tipo de responsabilidad sobre el contenido de la obra y que asume la responsabilidad frente a cualquier reclamo o demanda por parte de terceros de manera exclusiva.

Aceptando esta licencia, se cede a la Universidad Técnica de Machala el derecho exclusivo de archivar, reproducir, convertir, comunicar y/o distribuir la obra mundialmente en formato electrónico y digital a través de su Repositorio Digital Institucional, siempre y cuando no se lo haga para obtener beneficio económico.

Machala, 23 de febrero de 2022



FIGUEROA YAGUANA KEVIN ADRIAN
0704822444

DEDICATORIA

Con mucho corazón dedico el presente trabajo a mi familia, en particular, a mis padres, pues gracias a ellos he logrado llegar donde estoy. Han sido siempre un pilar fundamental en mi vida y mis estudios, dándome el apoyo necesario y las fuerzas de seguir adelante en cada momento que he necesitado.

También lo dedico a mi hermano porque espero que lo tome como un ejemplo de esfuerzo y perseverancia, que es posible cumplir las metas que uno se propone.

A mi abuelita Enma Dávila, pues ella ha sido como una segunda madre, me cuidó de pequeño por un tiempo y me enseñó el buen camino.

Finalmente, dedico este trabajo a mis amigos, compañeros y conocidos que, a lo largo de mi camino, han creído en mí y me han apoyado, porque a veces pequeñas palabras y buena actitud cambian tu día y tus ganas de perseverar en los sueños y objetivos.

Sr. Figueroa Yaguana Kevin Adrian

AGRADECIMIENTO

Agradezco en primer lugar a Dios por haberme permitido llegar a este momento de mi vida, culminando mi carrera universitaria con salud y fortaleza; a mis padres por el apoyo incondicional durante todas las etapas de mi vida, siempre con buenos consejos y animándome en los momentos duros.

Un gran agradecimiento también a mis amigos y compañeros, con los cuales, aunque haya compartido pocos momentos, han sido siempre parte del camino.

Finalmente, un agradecimiento muy especial a la Universidad Técnica de Machala por haber sido como mi segunda casa en mi etapa universitaria, donde reconozco haber conocido grandes personas con docentes de calidad, los cuales han sido clave en la culminación de cada materia, en particular, a mi tutor, el Ing. Wilmer Rivas, quien con su guía y conocimiento me permitió elaborar mi trabajo de titulación.

A los Ings Diego Villaseñor y Eduardo Luna, docentes de la facultad de ciencias agropecuarias de la UTMACH, que muy amablemente, dieron su apoyo y colaboración para la realización de una parte del trabajo.

Sr. Figueroa Yaguana Kevin Adrian

RESUMEN

En la actualidad, la tecnología digital se está aplicando en todo ámbito: medicina, economía, educación, agricultura, entre otros; en donde la innovación apunta a facilitar los procesos y encontrar soluciones cada vez más eficientes y autónomas, a las problemáticas o necesidades. La inteligencia artificial mediante Deep Learning es un claro ejemplo, con su aplicación que ha sido posible gracias al desarrollo de las herramientas informáticas y a la mejora continua de los equipos tecnológicos.

Esto ha permitido innovación en las técnicas tradicionales, donde las Tics son utilizadas como herramientas alternativas o complementarias en el estudio o investigación, en la búsqueda o análisis de datos; se es capaz inclusive de automatizar completamente determinadas tareas, sin la necesidad de la intervención humana (p. ej. el Internet de las cosas).

Hoy en día, también en la agricultura se está aplicando la tecnología: existen muchos estudios que revelan fuertes logros en el ámbito, aportando innovación y mejoras significativas, con las cuales se pretende llegar a una agricultura “inteligente” y de precisión para generar ventajas competitivas, ahorro de recursos y eficiencia en el campo.

Entonces, en búsqueda de innovación, se propuso el desarrollo del presente proyecto tecnológico: realizar un modelado de red neuronal artificial para el reconocimiento de suelos, que utilizará el color como criterio de reconocimiento.

El color del suelo es una propiedad física muy importante en la clasificación del suelo y permite inferir cierta información como el tipo y la naturaleza, muy útil cuando no se dispone de tiempo o herramientas para un análisis acurado en laboratorio.

Por lo tanto, el prototipo puede ser utilizado como un método alternativo para determinar el color del suelo; o, en caso de que ya exista un análisis previo, como confirmación y comparación de los resultados. Se destaca en más que, siendo una herramienta informática, se eliminaría el problema de la ambigüedad del color, el cual ocurre cuando se determina en color del suelo mediante el método tradicional de la observación directa con las tablas de Munsell.

Se desarrolló, por ende, un modelado de red neuronal artificial para el reconocimiento de suelos, que utilizará imágenes para el proceso de aprendizaje del tipo supervisado. Se optó, por consiguiente, por una arquitectura de red neuronal del tipo convolucional, siendo esta la arquitectura más usada y de mayor rendimiento general en el procesamiento de imágenes y extracción de características.

Como lenguaje de programación, se utilizó Python en conjunto a las librerías para Deep Learning de Keras y Tensorflow, con Visual Code como editor de código y entorno de ejecución.

Se finalizó el trabajo con la evaluación del prototipo a través de pruebas de reconocimiento del suelo mediante el color, las cuales fueron tabuladas en matrices de confusión, en base a las categorías de colores aprendidos; así, medir el rendimiento del modelo y determinar su nivel de confiabilidad en el reconocimiento.

Se concluye que los resultados conseguidos fueron satisfactorios y que se alcanzó una buena precisión general promedio; lo que demuestra que se logró un buen nivel de aprendizaje en el modelado, con un nivel de confiabilidad más que aceptable en el prototipo.

Palabras claves: Inteligencia artificial, aprendizaje profundo, redes neuronales convolucionales, color, suelo, Munsell.

ABSTRACT

Currently, digital technology is being applied in all areas: medicine, economy, education, agriculture, among others, where innovation aims to facilitate processes and find increasingly efficient and autonomous solutions to problems or needs. Artificial intelligence through Deep Learning is a clear example, with its application made possible thanks to the development of computer tools and the continuous improvement of technological equipment.

This has allowed innovation in traditional techniques, where ICTs are used as alternative or complementary tools in the study or research, in the search or analysis of data; it is even capable of completely automating certain tasks, without the need for human intervention (e.g., the Internet of things).

Today, technology is also being applied in agriculture: there are many studies that reveal strong achievements in the field, bringing innovation and significant improvements, with which the aim is to achieve "intelligent" and precision agriculture to generate competitive advantages, resource savings and efficiency in the field.

Therefore, in search of innovation, the development of this technological project was proposed: to develop an artificial neural network modeling for soil recognition, which will use color as a recognition criterion.

Soil color is a very important physical property in soil classification and allows to infer certain information such as type and nature, very useful when there is no time or tools available for an accurate laboratory analysis.

Thus, the prototype can be used as an alternative method to determine soil color; or, in case a previous analysis already exists, as a confirmation and comparison of the results. It is further emphasized that, being a computerized tool, it would eliminate the problem of color ambiguity, which occurs when soil color is determined by the traditional method of direct observation with Munsell tables.

Therefore, an artificial neural network modeling for soil recognition was developed, which will use images for the supervised learning process. So, a convolutional neural network architecture was chosen, as this is the most widely

used architecture with the highest overall performance in image processing and feature extraction.

As programming language, Python was used in conjunction with Keras and Tensorflow Deep Learning libraries, with Visual Code as code editor and runtime environment.

The work was completed with the evaluation of the prototype through soil recognition tests by color, which were tabulated in confusion matrices, based on the categories of colors learned; thus, measuring the performance of the model and determining its level of reliability in recognition.

It is concluded that the results obtained were satisfactory and that a good average general precision was reached; which shows that a good level of learning was achieved in the modeling, with a more than acceptable level of reliability in the prototype.

Keywords: Artificial intelligence, deep learning, convolutional neural networks, color, soil, Munsell.

CONTENIDO

DEDICATORIA	1
AGRADECIMIENTO	2
RESUMEN	3
ABSTRACT	5
INTRODUCCIÓN	11
1 CAPÍTULO I. DIAGNÓSTICO DE NECESIDADES Y REQUERIMIENTOS	
12	
1.1 Ámbito de Aplicación: descripción del contexto y hechos de interés ..	12
1.2 Establecimiento de requerimientos	13
1.3 Justificación del requerimiento a satisfacer	14
2 CAPÍTULO II. DESARROLLO DEL PROTOTIPO	15
2.1 Definición del prototipo tecnológico.....	15
2.2 Fundamentación teórica del prototipo	15
2.2.1 Marco Teórico.....	15
2.2.2 Marco Metodológico	20
2.3 Objetivos del prototipo	22
2.3.1 Objetivo General.....	22
2.3.2 Objetivos Específicos	22
2.4 Diseño del prototipo	23
2.5 Ejecución y/o ensamblaje del prototipo	24
2.5.1 Recolección de las muestras de suelo y su estandarización	24
2.5.2 Análisis color en laboratorio mediante tablas de Munsell	24
2.5.3 Toma de las fotografías	25
2.5.4 Preparación del dataset del modelo.....	27
2.5.5 Instalación entorno de trabajo IA	33
2.5.6 Generación del modelo.....	33

3	CAPÍTULO III. EVALUACIÓN DEL PROTOTIPO.....	36
3.1	Plan de evaluación.....	36
3.1.1	Métricas de evaluación	36
3.2	Resultados de la evaluación	39
3.2.1	Evaluación muestras de color Amarillo	40
3.2.2	Evaluación muestras de color Amarillo Marrón	42
3.2.3	Evaluación muestras de color Blanco	44
3.2.4	Evaluación muestras de color Marrón Amarillo	46
3.2.5	Evaluación muestras de color Marrón muy pálido.....	48
3.2.6	Evaluación muestras de color Marrón Pálido.....	50
3.2.7	Evaluación muestras de color Rojizo	52
3.2.8	Promedio de resultados obtenidos.....	53
3.2.9	Análisis global de los resultados del modelo.....	54
	CONCLUSIONES	56
	RECOMENDACIONES	57
	BIBLIOGRAFÍA	58
	ANEXOS.....	62
	Anexo I. Generar_imagenes.ipynb	62
	Anexo II. Entrenar.ipynb – Dataset	63
	Anexo III. Entrenar.ipynb – CNN.....	64
	Anexo IV. Predecir.ipynb	65

Índice de tablas

Tabla 1. Principales ámbitos de aplicación de los sistemas de IA	18
Tabla 2. Frameworks para programar con IA	20
Tabla 3. Resumen de imágenes clasificadas y generadas	33
Tabla 4. Matriz de confusión	36
Tabla 5. Métricas de evaluación.....	37
Tabla 6. Métricas de la matriz de confusión	37
Tabla 7: Cálculo y consideraciones de las métricas de desempeño	38
Tabla 8. Matriz de confusión - Amarillo	40
Tabla 9. Métricas de rendimiento - Amarillo	41
Tabla 10. Análisis métricas - Amarillo	41
Tabla 11. Matriz de confusión - Amarillo Marrón	42
Tabla 12. Métricas de rendimiento - Amarillo Marrón	43
Tabla 13. Análisis métricas - Amarillo Marrón	43
Tabla 14. Matriz de confusión - Blanco	44
Tabla 15. Métricas de rendimiento - Blanco	45
Tabla 16. Análisis métricas - Blanco	45
Tabla 17. Matriz de confusión - Marrón Amarillo	46
Tabla 18. Métricas de rendimiento - Marrón Amarillo	47
Tabla 19. Análisis métricas - Marrón Amarillo	47
Tabla 20. Matriz de confusión - Marrón muy pálido	48
Tabla 21. Métricas de rendimiento - Marrón muy pálido.....	49
Tabla 22. Análisis métricas - Marrón muy pálido	49
Tabla 23. Matriz de confusión - Marrón Pálido	50
Tabla 24. Métricas de rendimiento - Marrón Pálido	51
Tabla 25. Análisis métricas - Marrón Pálido	51
Tabla 26. Matriz de confusión - Rojizo	52
Tabla 27. Métricas de rendimiento - Rojizo	53
Tabla 28. Análisis métricas - Rojizo	53
Tabla 29. Promedios de las métricas de rendimiento	54
Tabla 30. Resumen de los resultados de las pruebas	54
Tabla 31. Análisis de las métricas de rendimiento del modelo	55

Índice de figuras

Figura 1. Arquitectura genérica de una CNN.....	19
Figura 2. Metodología utilizada	21
Figura 3: Metodología del modelo CNN	22
Figura 4: Estructura de los archivos para el modelo.....	23
Figura 5: Colocación muestras en cajas Petri de plástico	24
Figura 6: Tablas de Munsell - 10YR	25
Figura 7: Modo Pro de Xiaomi Note 9S.....	27
Figura 8: Ubicación de la data del modelo	27
Figura 9: Consultar tamaño de las fotografías tomadas a las muestras	28
Figura 10: Bulk Resize para redimensionar la data	29
Figura 11: Opciones de configuración de Bulk Resize	30
Figura 12: Proceso de Bulk Resize redimensionando las imágenes	31
Figura 13: Data redimensionada	32
Figura 14: Dataset preparado	32
Figura 15. Entrenar.ipynb – Output de la generación del modelo.....	34
Figura 16. Generación .h5 del prototipo	35
Figura 17. Formula F1-Score	39
Figura 18. Evidencia Test de color Amarillo	40
Figura 19. Evidencia Test de color Amarillo Marrón	42
Figura 20. Evidencia Test de color Blanco	44
Figura 21. Evidencia Test de color Marrón Amarillo	46
Figura 22. Evidencia Test de color Marrón muy pálido.....	48
Figura 23. Evidencia Test de color Marrón Pálido	50
Figura 24. Evidencia Test de color Rojizo	52
Figura 25. Generar_imagenes.ipynb	62
Figura 26. Entrenar.ipynb - Dataset	63
Figura 27. Entrenar.ipynb - CNN	64
Figura 28. Predecir.ipynb	65

INTRODUCCIÓN

La tecnología se aplica en todo ámbito de la sociedad y fomenta también la investigación e innovación. Es el caso de la inteligencia artificial que permite, hoy en día, resolver y facilitar muchos problemas o necesidades del ser humano, sin la necesidad de su intervención; el potencial es muy grande y cada año va creciendo gracias a que los equipos tecnológicos van mejorando y alcanzando mayores capacidades de procesamiento. Esto ha permitido que técnicas como el aprendizaje profundo sean cada vez más accesibles y puedan implementarse con mayor frecuencia.

En el presente proyecto, se tiene como propuesta aplicar la inteligencia artificial en el estudio del reconocimiento del suelo, para determinar una de sus tantas características físicas: el color. Esto con la finalidad de ayudar y complementar los estudios realizados a través de las técnicas tradicionales. Se destaca que el color es una propiedad que permite inferir desde el inicio ciertas características del suelo y muy útil cuando no se dispone de análisis inmediatos en laboratorio.

Se usará, por lo tanto, el aprendizaje profundo para crear un modelo de red neuronal artificial que sea capaz de predecir el color en muestras de suelo, generando un modelo que sea confiable y que pueda ser utilizado, por ejemplo, en una aplicación para facilitar la determinación del mismo color y apoyar de esta forma el estudio y la investigación en las ciencias agropecuarias. Se evaluará entonces el prototipo mediante pruebas de predicción que serán tabuladas y analizadas en matrices de confusión para determinar el rendimiento del modelo.

Se presenta, a continuación, la estructura del documento:

- Capítulo I: se describe el ámbito y la justificación del proyecto a realizarse, especificando el contexto, hechos de interés y requerimientos.
- Capítulo II: contiene el desarrollo del prototipo, iniciando con su definición y marco teórico, la metodología y los objetivos; también se define el diseño del prototipo y se relata el ensamblaje y/o ejecución.
- Capítulo III: se mencionan los resultados obtenidos en la evaluación del prototipo y se finaliza con las conclusiones y recomendaciones, según los objetivos fijados.

1 CAPÍTULO I. DIAGNÓSTICO DE NECESIDADES Y REQUERIMIENTOS

1.1 Ámbito de Aplicación: descripción del contexto y hechos de interés

En el ámbito digital, los avances tecnológicos han permitido facilitar muchas labores diarias del ser humano. La creación y el uso de las aplicaciones, por ejemplo, es una clara demostración de cómo la tecnología está interviniendo, ya sea en la vida cotidiana, laboral o social. De hecho, el uso de la tecnología digital se aplica en casi todo ámbito: en la medicina, se han desarrollado programas con inteligencia artificial para desarrollo de fármacos y predicciones de brotes para el COVID [1]; en la economía, para calcular la demanda de gas en Polonia mediante métodos de inteligencia computacional [2]; en la educación, donde se demuestra que la inteligencia artificial ya se está aplicando ampliamente en las instituciones educativas beneficiando estudiantes como docentes [3], y en la agricultura. De hecho, Smart Farming es el nuevo término utilizado en el sector, que indica la aplicación de soluciones innovadoras basadas en las Tics a las técnicas tradicionales y se afirma que estas nuevas soluciones, en Europa, generan cambios significativos en el área mediante, por ejemplo, el aprendizaje automático, procesamiento de imágenes y utilizzo de redes inalámbricas de sensores, destacando beneficios como el aumento de producción, la reducción de costos e insumos, la mejora de la calidad de los productos [4].

La inteligencia artificial está revolucionando muchas industrias al realizar tareas que normalmente requieren inteligencia humana para resolver [5]. Es un hecho entonces que la tecnología está en todos lados, que sigue avanzando en todo ámbito y permite resolver muchas necesidades y problemáticas de la actualidad.

En el presente caso, como trabajo de titulación propuesto, se busca aplicar la inteligencia artificial en la agricultura, mediante la creación de un modelo de red neuronal para predecir el color de muestras de suelo. De esta forma, facilitar el estudio del suelo, también conocido como estudio geotécnico, en el reconocimiento, aportando innovación en la investigación de este sector, muy importante para el país.

De hecho, en Ecuador, la agricultura representa un pilar fundamental de la economía, donde de los agricultores depende la alimentación saludable de la

población, parte de los productos para la exportación, además de generar fuentes de empleo en el sector rural que disminuye la pobreza en el campo [6].

Por lo tanto, fomentar la investigación del suelo mediante las Tics es mirar al futuro, a cambios de técnicas tradicionales a técnicas innovadoras, hacia una agricultura inteligente y de precisión, que aproveche la tecnología para crear ventajas competitivas en el campo, con la finalidad de aumentar la productividad y eficiencia, optimizando tiempo, costos y recursos, disminuyendo a la vez la carga del trabajo humano.

1.2 Establecimiento de requerimientos

El estudio del suelo y su reconocimiento es una tarea difícil y bastante complicada que hace parte de la ingeniería de las ciencias agropecuarias, en especial, de la edafología, disciplina que estudia la naturaleza y composición del suelo. Para crear un modelado de red neuronal que reconozca el suelo por su color, se necesitará los siguientes requerimientos:

- Un editor de código con entorno de desarrollo
- API de alto nivel para el aprendizaje profundo
- Muestras de suelo para crear un dataset de entrenamiento.
- Variedad de tonos de colores en las muestras.
- Una cámara para realizar fotografías a las muestras.
- Un ambiente adecuado para la exposición de las fotografías.

Entre los requerimientos complementarios, se destaca:

- Preparación previa para estandarizar las muestras del suelo.
- Análisis del color de las muestras mediante las tablas Munsell.

A pesar de que el modelo final será limitado al aprendizaje de solo una pequeña gama de colores del suelo, no se excluye la posibilidad de realizar nuevos entrenamientos en futuro para la predicción de los demás colores. Sin embargo, será necesario ampliar correctamente el dataset con nuevas imágenes de aprendizaje correspondientes.

1.3 Justificación del requerimiento a satisfacer

Una de las características físicas que se estudia del suelo es el color. Mediante esta característica como criterio, se puede reconocer el tipo de suelo, identificar y deducir la naturaleza de los componentes por medio de las propiedades refractivas o identificar los diferentes horizontes que componen el suelo; además de ser lo más notable y fácilmente determinado en el campo o en laboratorio [7].

Por lo tanto, a través del color, se puede tener una idea previa de la composición suelo, presentándose como fuente de información valiosa para la investigación y el reconocimiento del suelo, siendo necesaria su definición con gran precisión.

Se destaca además que las tonalidades de los colores presentes son muy amplias y, a simple vista, se pueden generalizar como blanco, marrón, rojo, amarillo y gris, entre los más característicos. Sin embargo, una correcta medición del color es mediante el uso de las tablas de Munsell a través de la observación directa de las muestras del suelo, pero esto significa subjetividad en los resultados porque cada observador puede percibir una variación de color diferente [8] . Entonces se presenta una buena justificación para realizar el actual trabajo de titulación, porque propone solucionar el problema de ambigüedad del color cuando se realiza el análisis de laboratorio mediante observación directa.

Además, se simplificará el proceso de la determinación del color y se optimizarán los tiempos de ejecución. Esto es más significativo y relevante cuando se debe reconocer y determinar el color de muchas muestras, ya que la predicción del modelo es inmediata una vez inicializado el prototipo y, aún mejor, cuando cargado en memoria y consumido por una aplicación dedicada al reconocimiento.

2 CAPÍTULO II. DESARROLLO DEL PROTOTIPO

2.1 Definición del prototipo tecnológico

El modelado de red neuronal artificial desarrollado en el presente proyecto tecnológico utiliza la estructura de las redes neuronales del tipo convolucionales (CNNs) y dará como resultado final un archivo del tipo .h5 que contendrá el modelo capaz de reconocer y entonces predecir el color de las muestras de suelo, según el grado de aprendizaje realizado.

Se destaca que la estructura de modelo elaborada permite enriquecer la data de aprendizaje a futuro y generar nuevos modelos, sin necesidad de aportar grandes cambios, siempre y cuando se considere lo siguiente:

1. Las imágenes para el aprendizaje del modelo deben estar organizadas según el color y en subcarpetas ubicadas dentro la carpeta raíz;
2. El tamaño de las imágenes está en formato de resolución 4:3, con 326 píxeles de ancho y 245 píxeles de alto.

Se recalca que el tamaño preestablecido de las imágenes requeridas puede ser modificado. Esto porque dentro del código del modelo, el tamaño está configurado en variables. Por lo tanto, se puede fácilmente aportar cambios en el tamaño de imágenes requerido; sin embargo, se evidencia la posibilidad de generar errores al intentar crear modelos con imágenes demasiado grandes, además de sobrecargar el trabajo computacional para generar el modelo.

2.2 Fundamentación teórica del prototipo

2.2.1 Marco Teórico

2.2.1.1 El estudio del suelo

El suelo es un recurso natural que, junto con el agua, se encuentra entre los recursos primordiales para la existencia y el desarrollo de la civilización humana [9]. Es crucial para el bienestar del ser humano en el planeta Tierra y posee diferentes funciones: no solo sirve para el crecimiento de las plantas, sino también como purificador de fuentes de agua, como repositorio de biodiversidad floral y faunística, como medio de protección del ambiente contra sustancias contaminantes [10].

La clasificación del suelo es hecha para múltiples propósitos: en ámbito científico, para proporcionar un lenguaje común y facilitar la comparación e intercambio de información acerca el suelo, resultados y experiencias en el tema agrícola; en ámbito práctico, para inventario y resolución de problemas aplicados a la agricultura, uso de las tierras, la ingeniería, estudios ambientales y construcción [11].

Se aclara que una clasificación es ensamblar objetos en secciones de manera lógica y secuencial, basándose en las propiedades para estudiar e identificar; de esta forma se ordena el conocimiento, se recuerdan mejor las propiedades y se aprende si hay relaciones existentes entre los objetos clasificados [12].

Las propiedades identificadas en los estudios del suelo son de diferentes indoles: se destacan propiedades físicas, químicas, mecánicas y biológicas [13]; y se enfatiza que el color es la más notable y que permite deducir características como el tipo de suelo, número de horizontes o la presencia de ciertos minerales [7].

El muestreo y el análisis visual del suelo para evaluar su estado y el potencial uso se practican ampliamente desde pequeñas parcelas a grandes escalas, donde el principal interés ha sido tradicionalmente el potencial que tiene para la producción agrícola [14].

2.2.1.2 La tecnología en el estudio de suelos

Existen diferentes investigaciones que demuestran cómo se aplica la tecnología en el estudio de los suelos con objetivos como:

- Mediante aprendizaje profundo con redes neuronales, identificar y distinguir muestras entre grava, arcilla y arena, con una exactitud perfecta en el reconocimiento de la arena [15].
- Con la generalización de un modelo, predecir la presión de expansión del suelo mediante técnicas de inteligencia computacional, teniendo en cuenta parámetros como la mineralogía, el contenido de arcilla, la densidad seca, la humedad, etc., en conjunto con factores estructurales y ambientales [16].
- Con técnicas de reconocimiento de patrones, asistidos por computadora y minería de datos, modelar el comportamiento del suelo [17].

- Mediante Extreme Learning Machine (ELM), predecir el rendimiento del café analizando las propiedades de fertilidad del suelo basadas en la materia orgánica (MOS) [18].
- Con el Internet de las Cosas (IOT), aumentar la productividad en los procesos agrícolas y ganaderos para una ecologización urbana, reduciendo la intervención humana, a través de sensores de humedad del suelo, de pH del suelo, entre otros [19].
- Con agricultura de precisión, donde la tecnología de punta impulsa la agricultura en la evaluación del suelo y el monitoreo de contaminantes por medio de sensores, destacando el método optimizado de tratamiento de datos derivado del Cloud Computing (Edge Computing) [20].
- Mediante la fotogrametría de dispositivos móviles, medir la densidad aparente del suelo, en donde se distingue que la fotogrametría es una técnica no destructiva para las muestras con los resultados obtenidos de igual precisión respecto al método del terrón y al escaneo láser tridimensional [21].

2.2.1.3 Concepto de Inteligencia Artificial

Se conoce como Inteligencia Artificial (IA) el estudio de la ciencia e ingeniería que tratan de hacer que las máquinas sean inteligentes, de modo que entiendan el lenguaje humano y aprendan, así como los seres humanos, a alcanzar problemas y metas; en otras palabras, poner la mente humana dentro la computadora donde, por inteligencia, se refiere a la parte computacional capaz de lograr los objetivos [22].

Otro concepto de IA es que son métodos con los cuales una computadora toma decisiones o hacer descubrimientos que normalmente requieren de la inteligencia humana [23]; que IA tiene por objetivo que las máquinas realicen la misma clase de cosas que la mente hace (razonar, asociar, predecir, planificar, controlar, etc.), donde “la inteligencia no es una dimensión única, sino un espacio profusamente estructurado de capacidades diversas para procesar la información” [24].

En la siguiente tabla, se presentan los principales ámbitos de aplicación de la inteligencia artificial.

Tabla 1. Principales ámbitos de aplicación de los sistemas de IA

Área	Aplicaciones
Medicina	Ayuda al diagnóstico Análisis de imágenes biomédicas Procesado de señales fisiológicas
Ingeniería	Optimización de la producción Optimización de procesos Cálculo de estructuras Planificación y logística Diagnóstico de fallos Toma de decisiones
Economía	Análisis financiero y bursátil Análisis de riesgos Estimación de precios en productos derivados Minería de datos Marketing y fidelización de clientes
Biología	Análisis de estructuras biológicas Genética médica y molecular
Informática	Procesado de lenguaje natural Criptografía Teoría de juegos Lingüística computacional
Robótica y automática	Sistemas adaptativos de rehabilitación Interfaces cerebro-computadora Sistemas de visión artificial Sistemas de navegación automática
Física y matemáticas	Demostración automática de teoremas Análisis cualitativo sistemas no-lineales Caracterización de sistemas complejos

Fuente: Raúl Benítez, Gerard Escudero, Samir - *Inteligencia artificial avanzada* [25]

La inteligencia artificial se puede dividir en varias ramas entre las cuales se encuentran el Aprendizaje Automático (Machine Learning), Visión por Computador o Visión Artificial, Lógica Difusa, Procesamiento de Lenguaje Natural, Heurística y Agentes Inteligentes [22].

2.2.1.4 Concepto de Aprendizaje Automático (Machine Learning)

El Aprendizaje Automático es una de las ramas de la IA que está basado en el desarrollo de técnicas de aprendizaje para que las máquinas puedan aprender y facilitar el procesamiento de la data de forma inteligente. De esta forma adquirir la capacidad de clasificar, predecir, descubrir o generar nuevos datos; donde la idea de Aprendizaje Automático surge después del entrenamiento de datos por medio de algoritmos, en procesos automatizados que aprenden con el ejemplo [23].

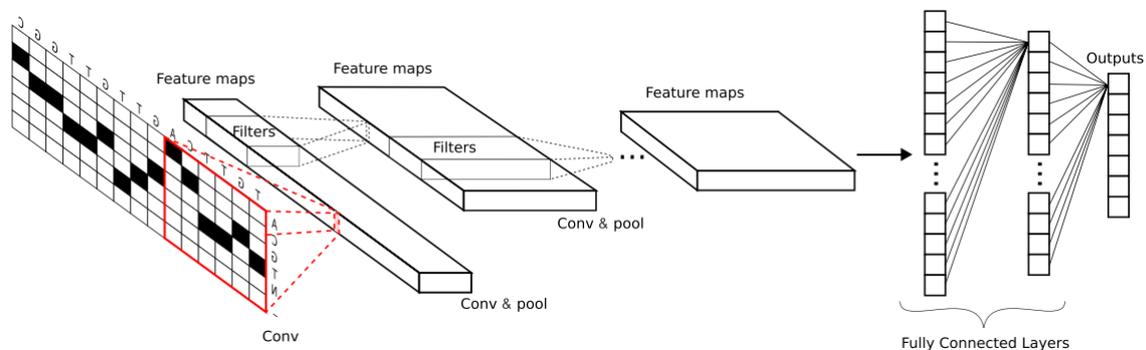
2.2.1.5 Concepto de Aprendizaje Profundo (Deep Learning)

El aprendizaje profundo es una técnica del Aprendizaje Automático que hace uso de algoritmos para el aprendizaje, con redes neuronales que simulan el cerebro humano para el aprendizaje analítico, imitando su mecanismo para interpretar datos como imágenes, textos y sonidos [26]. Los modelos generados en este tipo de aprendizaje tienen una estructura por capas: la información de salida de una capa es entrada para la siguiente capa; la mejor parte del Aprendizaje Profundo radica en que no es necesario ninguna ingeniería de funciones antes del entrenamiento de los modelos de estas redes; por lo tanto, la data de entrada pasa directamente a la red profunda y los modelos resultantes salen con un buen rendimiento en términos de precisión. [27]. Con la llegada de la Big Data, este tipo de tecnología se ha convertido en una dirección de investigación importante en el campo del Aprendizaje Automático y se ha aplicado considerablemente para el procesamiento de imágenes, procesamiento del lenguaje natural, reconocimiento de voz, publicación en línea, entre otros [28].

2.2.1.6 Redes neuronales convolucionales

Las redes neuronales convolucionales (CNN) son algoritmos típicos del Aprendizaje Profundo que han demostrado un rendimiento superior en varios dominios y que aprenden una representación de características de alto nivel con varios parámetros de optimización [29]. Se benefician de la aceleración GPU para realizar cálculos computacionalmente arduos en paralelo, en tiempos razonables y a un bajo costo [23].

Figura 1. Arquitectura genérica de una CNN



Fuente: Cruz, Murilo Horacio Pereira; Domingues, Douglas Silva; Saito, Priscila Tiemi Maeda; Paschoal, Alexandre Rossi; Bugatti, Pedro Henrique (2020) [30].

Son especialmente eficaces en el área del procesamiento de imágenes [31], de hecho, es el método principalmente utilizado [32].

2.2.1.7 Recursos para programación en IA

Afortunadamente, la mayoría de los recursos necesarios para empezar con el Aprendizaje Automático son disponibles gratuitamente [33]. En la siguiente tabla se muestra una lista de frameworks accesibles para el uso.

Tabla 2. Frameworks para programar con IA

Table 1. Deep learning frameworks with programming interfaces. Most frameworks have open source code and many support multiple programming languages.

Framework	License	Programming interfaces
Apache SINGA [420]	Apache 2.0 [421]	C++, Java, Python
BigDL [422]	Apache 2.0 [423]	Python, Scala
Caffe [424, 425]	BSD [426]	C++, MATLAB, Python
Chainer [427]	MIT [428]	Python
Deeplearning4j [429]	Apache 2.0 [430]	Clojure, Java, Kotlin, Python, Scala
Dlib [431, 432]	BSL [433]	C++
Flux [434]	MIT [435]	Julia
MATLAB Deep Learning Toolbox [436]	Proprietary [437]	MATLAB
Microsoft Cognitive Toolkit [438]	MIT [439]	BrainScript, C++, Python
Apache MXNet [440]	Apache 2.0 [441]	C++, Clojure, Go, JavaScript, Julia, Matlab, Perl, Python, R, Scala
OpenNN [442]	GNU LGPL [443]	C++
PaddlePaddle [444]	Apache 2.0 [445]	C++
PyTorch [446]	BSD [447]	C++, Python
TensorFlow [448, 449]	Apache 2.0 [450]	C++, C#, Go, Haskell, Julia, MATLAB, Python, Java, JavaScript, R, Ruby, Rust, Scala, Swift
Theano [451, 452]	BSD [453]	Python
Torch [454]	BSD [455]	C, Lua
Wolfram Mathematica [456]	Proprietary [457]	Wolfram Language

Fuente: Jeffrey M Ede [33]

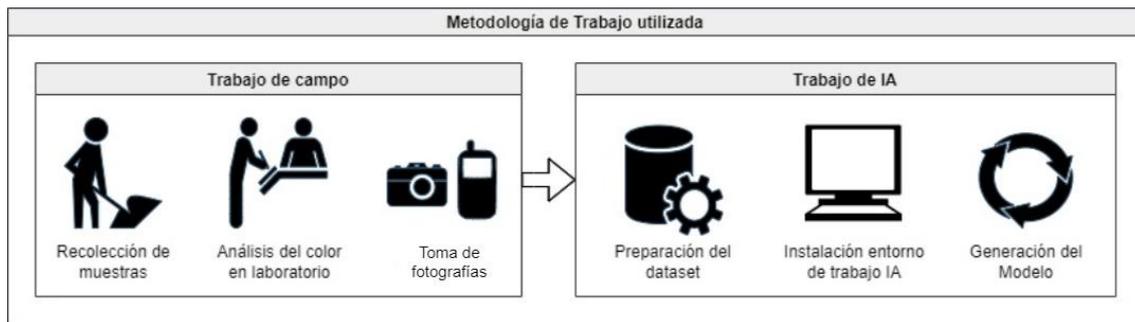
Se resalta entre ellos TensorFlow, el cual fue originalmente creado por los investigadores de Google y resulta ser el más popular entre las librerías para el Aprendizaje Profundo; además facilita y acelera en gran medida la investigación y la aplicación de modelos de redes neuronales [34].

Se destaca Python, un lenguaje de programación que se está convirtiendo en una herramienta prometedora para el análisis de datos en varios campos [35]; adicionalmente Keras, una interfaz Python de alto nivel para TensorFlow y otras librerías para Deep Learning [36].

2.2.2 Marco Metodológico

La metodología de trabajo utilizada para realizar el trabajo se presenta en la siguiente figura.

Figura 2. Metodología utilizada



Fuente: Elaboración propia

Consiste en un trabajo inicial de campo para recolectar las muestras de suelo estandarizadas, necesarias para generar un dataset propio de imágenes para el aprendizaje. De esta forma, proceder con el trabajo de IA para la generación del modelo, donde se realiza el entrenamiento, validación y optimización del prototipo.

2.2.2.1 Trabajo de Campo

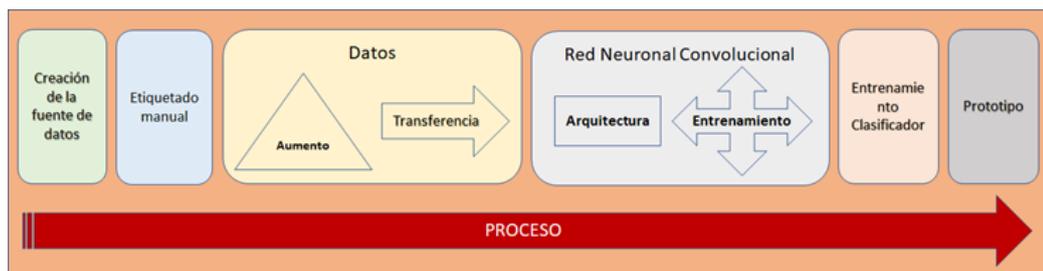
En el trabajo de campo se considera lo siguiente:

- La recolección de las muestras fue facilitada por medio de la Universidad Técnica de Machala. En resumen, para obtener la muestra, se realiza una calicata en el lugar aproximadamente de 1.20 m, se identifican los horizontes del suelo observando el color y se saca una muestra.
- Las muestras deben ser estandarizadas porque el contenido de humedad cambia el valor del color de las muestras durante el proceso de desecación [8]; entonces, en síntesis, para la estandarización de las muestras se realizó lo siguiente:
 - Secar al sol el material recabado en una bandejita por 48 horas.
 - Deshacer los terrones presentes moliendo la tierra.
 - Realizar un tamizado de la tierra con un cedazo.
 - Lo que se toma como muestra estandarizada es la parte más fina, lo que queda encima del cedazo, la parte más gruesa, se bota.
- En la toma de fotografías se consideraron diferentes aspectos importantes como la estabilidad de la cámara, condiciones de luz, la distancia entre la muestra y la cámara [37].

2.2.2.2 Trabajo de IA

- Para la preparación del dataset se tomó en cuenta la técnica del aumento de datos que ayuda a mejorar las capacidades de generalización de las redes neuronales profundas, por medio de diferentes operaciones como rotación, zoom, recorte de las imágenes originales. Se debe a que en ciertos escenarios, no es posible obtener una cantidad de datos reales suficientes y de buena calidad [38].
- Para la generación del modelo con la CNN, se siguió como referencia la siguiente metodología de desarrollo:

Figura 3: Metodología del modelo CNN



Fuente: Ángel Lerma Sánchez, Adriana Santoyo, Fabiola Villalobos, Pedro Reyes [39]

2.3 Objetivos del prototipo

2.3.1 Objetivo General

- Desarrollar un modelo de red neuronal artificial para el reconocimiento de suelos mediante el color, utilizando técnicas de Deep Learning.

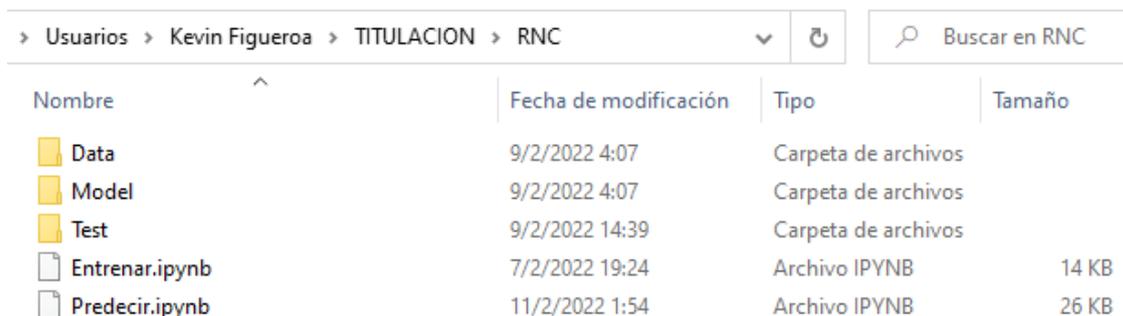
2.3.2 Objetivos Específicos

- Utilizar técnicas de aprendizaje profundo mediante redes neuronales convolucionales para la predicción del color del suelo.
- Recolectar muestras estandarizadas de suelo para un correcto análisis del color y la obtención de la data.
- Analizar en laboratorio el color de las muestras mediante el uso de las tablas de Munsell.
- Realizar una acondicionada toma fotográfica de las muestras para la generación de una data válida para el aprendizaje del modelo.
- Evaluar el rendimiento del modelo mediante pruebas de reconocimiento analizadas en diferentes matrices de confusión.

2.4 Diseño del prototipo

El prototipo del modelo de predicción tiene la siguiente estructura:

Figura 4: Estructura de los archivos para el modelo



Nombre	Fecha de modificación	Tipo	Tamaño
Data	9/2/2022 4:07	Carpeta de archivos	
Model	9/2/2022 4:07	Carpeta de archivos	
Test	9/2/2022 14:39	Carpeta de archivos	
Entrenar.ipynb	7/2/2022 19:24	Archivo IPYNB	14 KB
Predecir.ipynb	11/2/2022 1:54	Archivo IPYNB	26 KB

Fuente: Elaboración propia

Consiste en una carpeta general, en captura nombrada RNC, donde se ubica lo siguiente:

- **Data:** carpeta destinada al contenido del dataset.
- **Model:** carpeta en la cual se almacena el modelo generado.
- **Test:** carpeta donde se ubican las imágenes de prueba para la predicción.
- **Entrenar.ipynb:** archivo conteniente el código fuente para el aprendizaje del modelo.
- **Predecir.ipynb:** archivo que genera las predicciones de color según las imágenes de muestra ubicadas en la carpeta Test para la evaluación del modelo.

Se enfatiza que, para generar el modelo, es necesario cumplir con los siguientes pasos previos:

- Tener muestras de suelo estandarizadas.
- Determinar los colores de las muestras mediante las tablas de Munsell.
- Tomar fotografías válidas y adecuadas a las muestras.
- Armar el dataset con las fotografías tomadas.

A continuación, en la siguiente sección de ejecución y/o ensamblaje del prototipo, se detalla cómo se ha cumplido con los pasos previos requeridos.

2.5 Ejecución y/o ensamblaje del prototipo

2.5.1 Recolección de las muestras de suelo y su estandarización

Para el presente trabajo de titulación, las muestras estandarizadas fueron facilitadas por medio de personal docente de la Universidad Técnica de Machala de la facultad de ciencias agropecuarias; más específicamente, son muestras obtenidas por parte del trabajo de campo de estudiantes de la carrera de agronomía, realizado en varios lugares, tipo fincas y bananeras, en la provincia de El Oro.

2.5.2 Análisis color en laboratorio mediante tablas de Munsell

En esta fase del proyecto, se realizó la toma del color de las muestras estandarizadas en laboratorio mediante la observación directa. Para ello, es necesario solo una pequeña porción de tierra de cada muestra de suelo, por lo tanto, se ubicó cada muestra en una caja Petri de plástico. Esto se consideró idóneo para facilitar el trabajo, optimizar el espacio y poder luego tomar las fotografías.

Figura 5: Colocación muestras en cajas Petri de plástico



Fuente: Elaboración propia

Se procedió entonces a la determinación del color mediante observación directa, comparando el color de la muestra con las tablas de colores de Munsell.

Para este proceso, se ubica en el orificio de la tabla una pizca de muestra y se observa el color más semejante, anotando los valores del matiz (hue), claridad

(value) y pureza (chroma), que identifican el nombre del color preciso al cual pertenece la muestra [40].

Figura 6: Tablas de Munsell - 10YR



Fuente: Elaboración propia

Se indica que la tabla Munsell que se utilizó para la mayoría de las muestras estandarizadas recolectadas fue la tabla 10YR.

2.5.3 Toma de las fotografías

Esta es una fase muy importante y crítica del proyecto: de la calidad de las fotos tomadas a las muestras, dependerá la validez de la data para el aprendizaje del modelo.

Se requiere de un dispositivo que pueda tomar fotos y, además, acondicionar el lugar donde hacerlo. Esto último es fundamental porque se va a generar un modelo que va a predecir colores, por ende, el tema de una justa exposición de las fotos es determinante en la calidad final de las fotografías.

De hecho, es importante controlar la estabilidad de la cámara, las condiciones de iluminación y la distancia focal, además de tener en cuenta otros aspectos como la humedad y el tamaño de las partículas del suelo [37].

Entonces para cumplir con una adecuada toma de las fotografías de las muestras de suelo, se utilizó un smartphone de gama media y se ejecutó lo siguiente:

- La estabilidad de la cámara fue solucionada fijando el dispositivo en un soporte para mantener firme la toma y evitar fotos trepidadas.

- Las condiciones de iluminación fueron fijadas en tomas en horario nocturno con la ayuda de una lámpara led de luz blanca para tener suficiente iluminación. De esta forma se evita fotos afectadas por la luz solar que, dependiendo de su intensidad en el día, aclara u oscurece el ambiente, afectando las tomas fotográficas en correspondencia a más claras u oscuras. El horario nocturno, en vez, hará que el ambiente sea afectado por la misma intensidad de luz dada por los focos ubicados en el sitio y el uso de la lámpara es para obtener una iluminación adecuada en las muestras. Se destaca que el uso de una lámpara tipo led es mejor por su espectro luminoso que simula mejor la luz natural del día respecto a lámparas con foco de luz cálida.
- El tema de la distancia focal se acondiciona eligiendo el tipo de lente a usarse en la cámara, estableciéndose como parámetro fijo en la configuración del dispositivo.
- Los otros aspectos mencionados que afectan, tipo la humedad y el tamaño de las partículas, se solucionan mediante la estandarización de las muestras.

Además, se han tomado en cuenta aspectos adicionales:

- Que la distancia física entre la muestra del suelo y la lente de la cámara se mantenga fija en aproximadamente 5 cm. Se enfatiza que fue posible gracias a un soporte utilizado para resolver la estabilidad de la cámara.
- No utilizar el modo automático del dispositivo y configurar manualmente el modo de medición, la sensibilidad ISO, la velocidad de obturación, el zoom y el balance de blancos para tomas más exactas y profesionales.

Se detalla a continuación la configuración utilizada para la toma fotográfica del presente trabajo:

- Dispositivo: smartphone Xiaomi Note 9S
- Funciones de la cámara: modo Pro
- Tipo lente: ultra gran angular de 8 megapíxeles
- Sensibilidad ISO: 800
- Velocidad de obturación: 1/25
- Apertura focal: f/2.2

- Zoom: x1
- Modos de medición: puntual centrado

Figura 7: Modo Pro de Xiaomi Note 9S

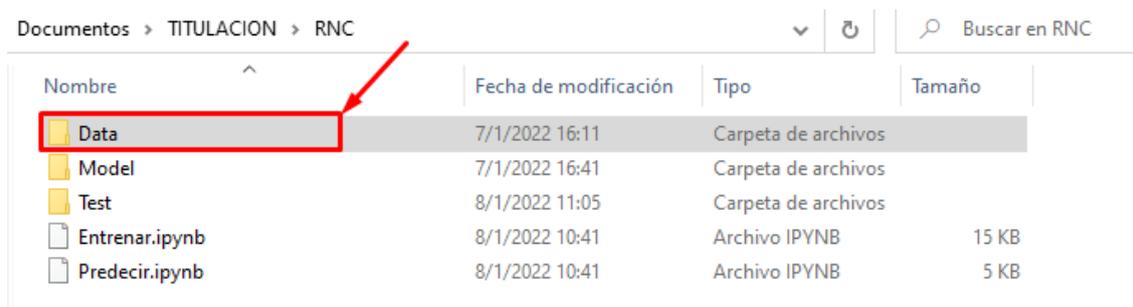


Fuente: Elaboración propia

2.5.4 Preparación del dataset del modelo

Luego de haber tomado las fotografías de las muestras, se seleccionaron las imágenes para importarlas en la carpeta raíz del modelo.

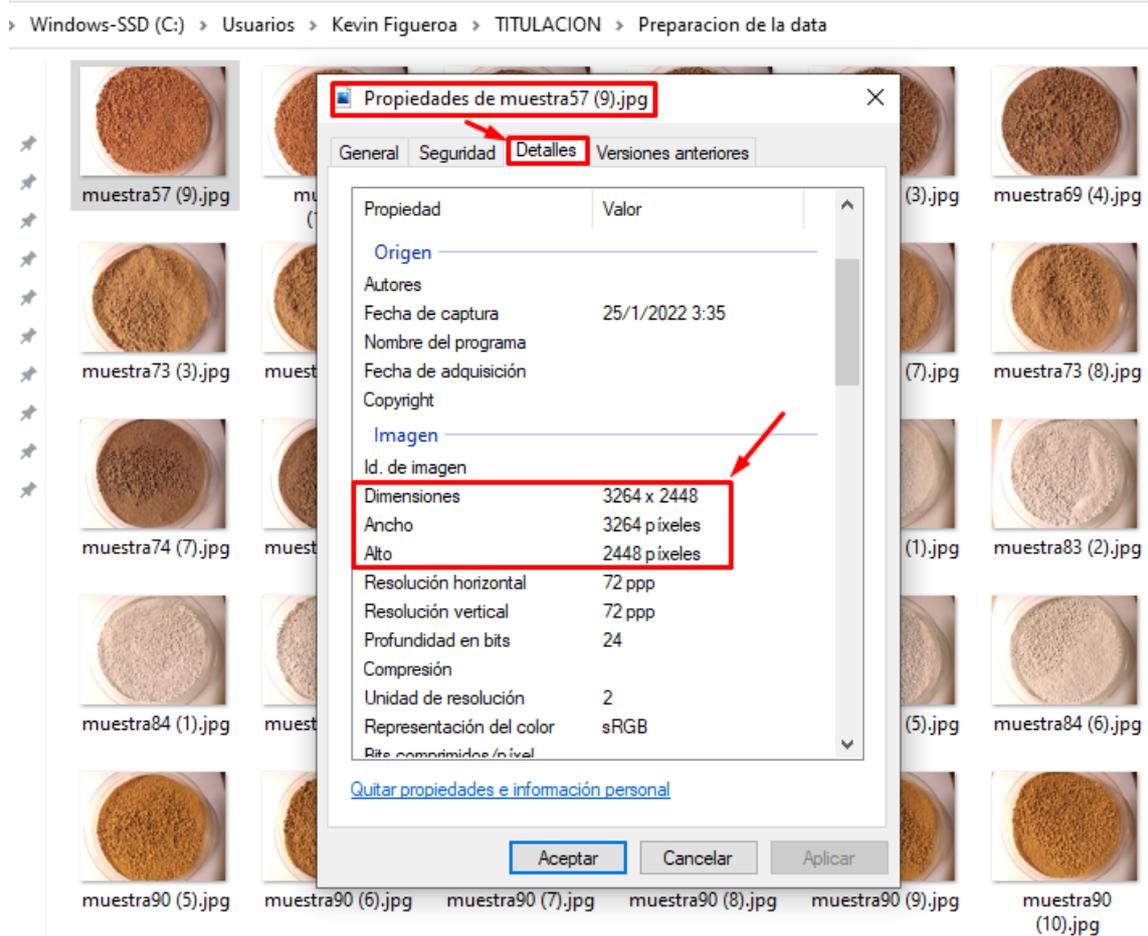
Figura 8: Ubicación de la data del modelo



Fuente: Elaboración propia

Antes de este paso, se recomienda verificar el tamaño en píxeles de las imágenes para evitar posibles errores en ejecución del código debido a un tamaño demasiado grande. Para esta operación, se usó el explorador de archivos Windows para ubicarse en las imágenes, hacer clic sobre ellas y ver en propiedades. Se abrió entonces una pestaña de diálogo donde consultar la información contenida en los metadatos de las imágenes, y en la sección de detalles, se consultó las dimensiones de alto y ancho.

Figura 9: Consultar tamaño de las fotografías tomadas a las muestras



Fuente: Elaboración propia

Como se puede observar en la figura, el tamaño supera los miles de píxeles en alto y ancho; significa que es demasiado grande. De hecho, con imágenes de tal tamaño, se ocasionan errores al momento de generar el modelo, además de requerir una mayor cantidad de recursos computacionales para la extracción de características.

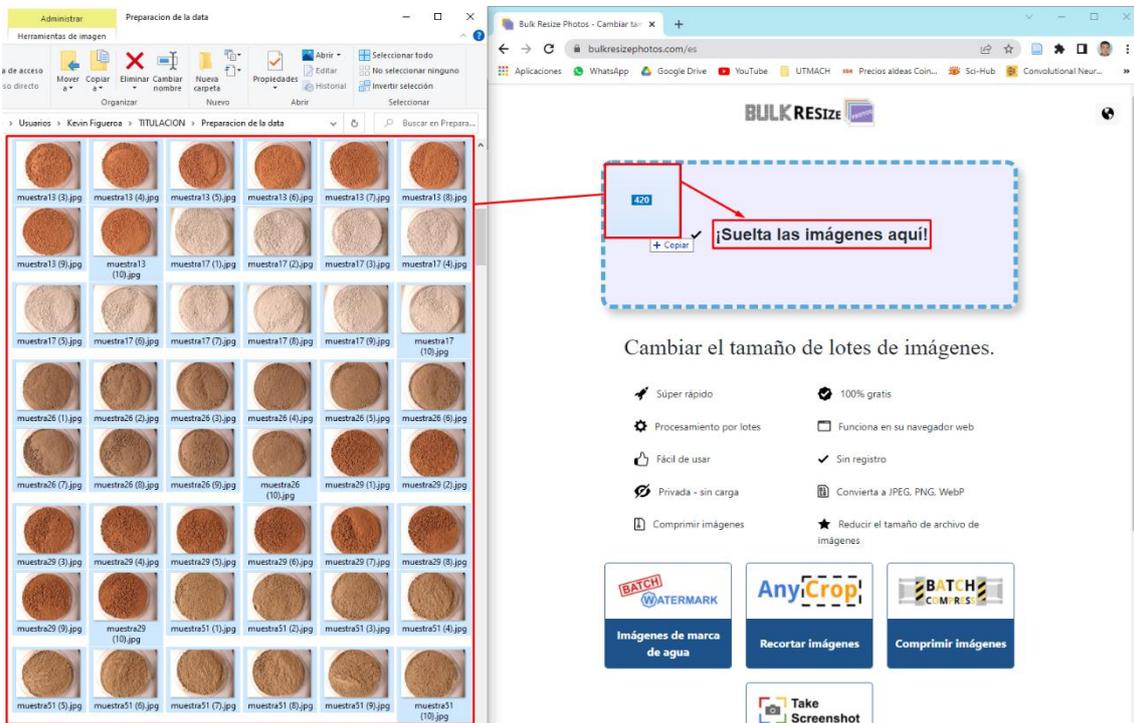
Es necesario, por lo tanto, reducir el tamaño de la imagen, aunque porque para cumplir el propósito propuesto, son suficientes imágenes de mucha menor resolución.

Una válida herramienta que se utilizó, muy eficaz para resolver el problema y automatizar el proceso, fue utilizar Bulk Resize, la cual se encontró disponible en la siguiente página web:

- <https://bulkresizephotos.com/es>

Es gratuita, rápida y permite cambiar el tamaño de lotes de imágenes sin afectar la calidad para los fines propuestos. El uso además es muy sencillo: se deben seleccionar las imágenes y arrastrarlas al área indicada en la página.

Figura 10: Bulk Resize para redimensionar la data



Fuente: Elaboración propia

Una vez soltadas las imágenes, cargará una nueva página donde se ingresa en porcentaje en cuanto escalar las imágenes, es decir, en qué porcentaje reducir el tamaño. También se podrá seleccionar un formato de la imagen, un color de fondo de imagen y la calidad en porcentaje.

Figura 11: Opciones de configuración de Bulk Resize



Fuente: Elaboración propia

Para los fines del presente trabajo de titulación, las imágenes seleccionadas han sido reducidas al 10% respecto a las dimensiones originales, el formato en JPEG y la calidad de imagen al 100%. Entonces se hace clic en el botón de comienzo para iniciar el proceso de redimensionamiento de las imágenes.

Figura 12: Proceso de Bulk Resize redimensionando las imágenes

37 / 220
muestra (40).jpg

68%
Generando archivo zip ...

Puede encontrar las imágenes de resultados en la carpeta de descargas.

Terminada

Descargar de nuevo

Antes de
961,7 MB

Después
46,3 MB

Puede encontrar las imágenes de resultados en la carpeta de descargas.

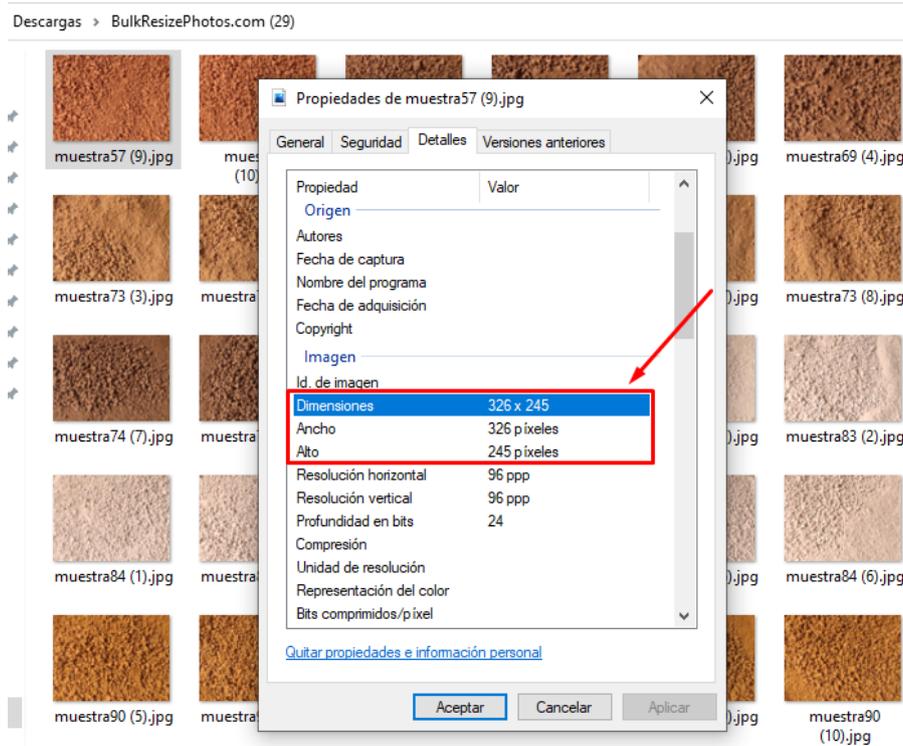
Se abre la carpeta de descarga y efectivamente se tendrá un zip con las imágenes todas redimensionadas.

Nombre	Fecha de modificación	Tipo	Tamaño
▼ Hoy (1)			
BulkResizePhotos.com (29).zip	14/2/2022 23:02	Archivo WinRAR Z...	47.500 KB
▼ La semana pasada (23)			

Fuente: Elaboración propia

En la siguiente captura, se muestran los resultados. Las imágenes mantuvieron su orden, el nombre y fueron escaladas a las dimensiones establecidas, al 10% en este caso de ejecución.

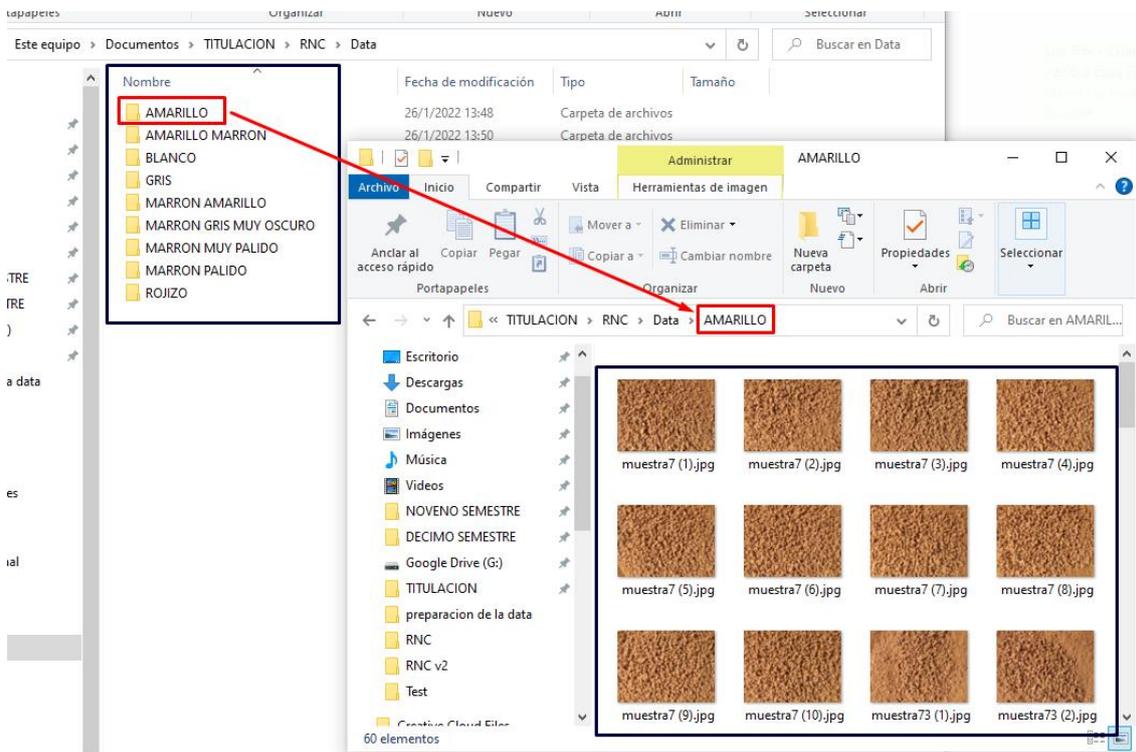
Figura 13: Data redimensionada



Fuente: Elaboración propia

Listas las imágenes con un tamaño adecuado, se las debe obligatoriamente organizar dentro la carpeta raíz, en subcarpetas organizadas por color.

Figura 14: Dataset preparado



Fuente: Elaboración propia

Para el aumento de los datos, se realizó un script en Python para crear nuevas imágenes de entrenamiento a partir de las tomas fotográficas originales.

Tabla 3. Resumen de imágenes clasificadas y generadas

Categoría	Muestras	Fotos originales	Fotos auxiliares	Fotos totales
Amarillo	6	60	600	660
Amarillo marrón	6	60	600	660
Blanco	6	60	600	660
Marrón amarillo	6	60	600	660
Marrón muy pálido	6	60	600	660
Marrón pálido	6	60	600	660
Rojizo	6	60	600	660
Total	42	420	4200	4620

Fuente: Elaboración propia

En Anexo I, una imagen del código fuente del script.

2.5.5 Instalación entorno de trabajo IA

Esto debe hacerse para poder programar y ejecutar los archivos de generación del modelo. Se utilizó principalmente el entorno de Visual Code como editor de código, Python como lenguaje de programación y las librerías de TensorFlow y Keras para la IA; en un equipo portátil i7-8750H de 2.20GHz con 16 GB de RAM y tarjeta gráfica GeForce GTX 1060 con 6GB de memoria dedicada.

A continuación, se detalla las versiones y librerías adicionales:

- Editor código: Microsoft Visual Studio Code 1.64.2
- Lenguaje programación: Python 3.7.4
- Librerías:
 - Tensorflow: 2.6.0
 - Keras: 2.6.0
 - Keras-Preprocessing: 1.1.2
 - numpy: 1.19.5
 - matplotlib: 3.1.1

2.5.6 Generación del modelo

Después de haber instalado las librerías necesarias para el entorno de trabajo, se procedió con el desarrollo del prototipo, directamente en la parte de la red neuronal convolucional.

Esto porque en el proceso de preparación del dataset, ya se realizó todo lo inherente a los datos y etiquetado.

Durante el desarrollo de la arquitectura de la red, fue importante establecer:

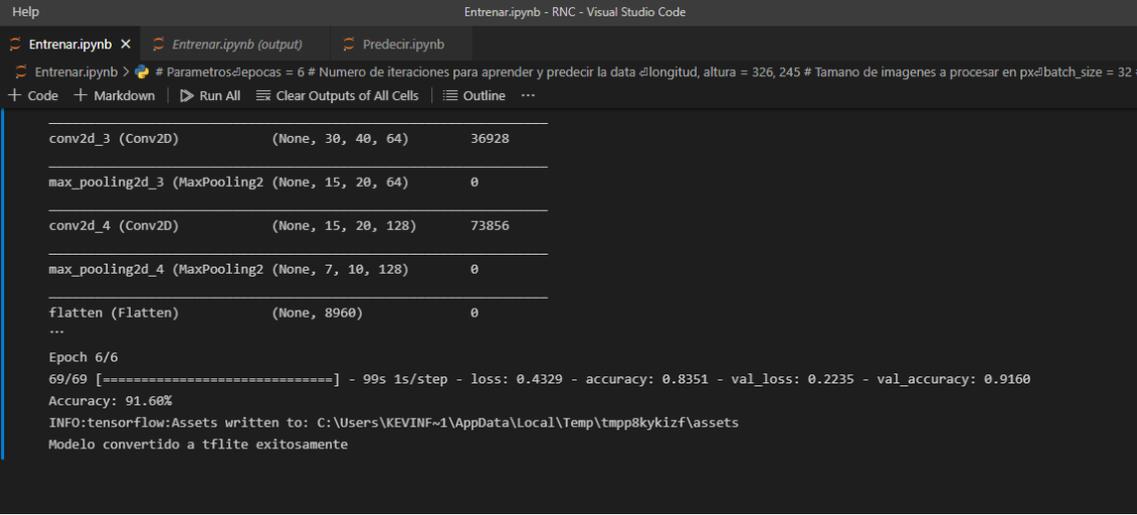
- Lectura y carga de todas las imágenes del dataset.
- Etiquetado y clases del dataset (ver Anexo II)
- El procesamiento de las imágenes para entrenamiento y validación.
- Valor de los parámetros de aprendizaje (p. ej. n épocas, batch_size)
- Estructura del modelo CNN (convoluciones, filtros y max-pooling; ver Anexo III).
- Compilación del modelo.

Codificado el prototipo, se procedió a la ejecución del mismo para el aprendizaje y la generación del modelo.

Esta fase de aprendizaje y generación se realizó muchas veces hasta conseguir el mejor modelo de predicción, cambiando parámetros donde era necesario.

A continuación, una captura del output del mejor modelo conseguido.

Figura 15. Entrenar.ipynb – Output de la generación del modelo

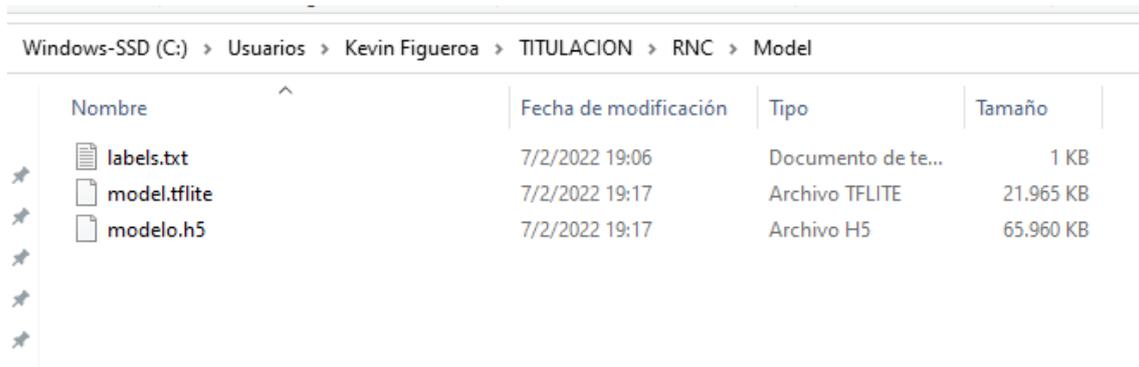


```
conv2d_3 (Conv2D) (None, 30, 40, 64) 36928
-----
max_pooling2d_3 (MaxPooling2D) (None, 15, 20, 64) 0
-----
conv2d_4 (Conv2D) (None, 15, 20, 128) 73856
-----
max_pooling2d_4 (MaxPooling2D) (None, 7, 10, 128) 0
-----
flatten (Flatten) (None, 8960) 0
...
Epoch 6/6
69/69 [=====] - 99s 1s/step - loss: 0.4329 - accuracy: 0.8351 - val_loss: 0.2235 - val_accuracy: 0.9160
Accuracy: 91.60%
INFO:tensorflow:Assets written to: C:\Users\KEVINF~1\AppData\Local\Temp\tmp8kykizf\assets
Modelo convertido a tflite exitosamente
```

Fuente: Elaboración propia

Se guardó el modelo del prototipo en la carpeta correspondiente.

Figura 16. Generación .h5 del prototipo



The image shows a Windows File Explorer window with the address bar displaying the path: Windows-SSD (C:) > Usuarios > Kevin Figueroa > TITULACION > RNC > Model. The main area shows a list of files with the following columns: Nombre, Fecha de modificación, Tipo, and Tamaño. There are three files listed: labels.txt (1 KB), model.tflite (21.965 KB), and modelo.h5 (65.960 KB). The file icons on the left are partially visible.

Nombre	Fecha de modificación	Tipo	Tamaño
labels.txt	7/2/2022 19:06	Documento de te...	1 KB
model.tflite	7/2/2022 19:17	Archivo TFLITE	21.965 KB
modelo.h5	7/2/2022 19:17	Archivo H5	65.960 KB

Fuente: Elaboración propia

Nota: se ha codificado como adicional que el modelo del prototipo se genere en su versión lite. Esto para que el prototipo de predicción pueda también utilizarse en aplicaciones del tipo móvil.

3 CAPÍTULO III. EVALUACIÓN DEL PROTOTIPO

3.1 Plan de evaluación

Como plan de evaluación del prototipo, se hará uso de una matriz de confusión para evaluar el rendimiento del modelo.

En general, la matriz de confusión es una tabulación cruzada entre clases que proporciona una base para determinar precisión general, caracterizar errores específicos y obtener otras medidas que se derivan de la misma [41].

Para aplicar la matriz, se utilizará un conjunto de imágenes de prueba; para cada una de ellas, se tiene una clasificación por etiqueta y el modelo debe reconocer el color, acertando esa etiqueta. Entonces, analizando cuántos resultados correctos y cuántos resultados erróneos se obtienen, se puede dar una medida del rendimiento alcanzado por el modelo, entendido básicamente como un porcentaje de acierto o errores, sobre el conjunto de imágenes de prueba.

3.1.1 Métricas de evaluación

Como herramienta de evaluación y obtener las métricas respectivas, se aplicará exactamente una matriz de confusión del tipo binario, como la que se presenta a continuación.

Tabla 4. Matriz de confusión

Table 4. Conceptualization of a binary classification confusion matrix. TP = True Positive, TN = True Negative, FP = False Positive, and FN = False Negative.

		Reference Data	
		Positive	Negative
Classification Result	Positive	TP	FP
	Negative	FN	TN

Fuente: Maxwell, A. E., Warner, T. A., & Guillén, L. A. (2021) [42]

Tiene cuatro entradas: el número de muestras verdadero positivos (TP) y verdaderos negativos (TN) que son los resultados correctamente clasificados como positivos y negativos; y las dos categorías de errores, los falsos positivos (FP) y falsos negativos (FN), que son el número de negativos clasificados incorrectamente como positivos y viceversa [42].

A partir de esas métricas de la matriz de confusión, es posible calcular las métricas tradicionales de evaluación de precisión del modelo:

Tabla 5. Métricas de evaluación

Table 3. Binary assessment metrics and relationships to traditional accuracy assessment metrics.

Metric	Equation	Relation to Traditional RS Measures
Overall Accuracy (OA)	$\frac{TP+TN}{TP+TN+FP+FN}$	Overall Accuracy
Recall	$\frac{TP}{TP+FN}$	PA for positives
Precision	$\frac{TP}{TP+FP}$	UA for positives
Specificity	$\frac{TN}{TN+FP}$	PA for negatives
Negative Predictive Value (NPV)	$\frac{TN}{TN+FN}$	UA for negatives

Fuente: Maxwell, A. E., Warner, T. A., & Guillén, L. A. (2021) [43]

A continuación, se presenta la interpretación de las métricas de la matriz de confusión.

Tabla 6. Métricas de la matriz de confusión

Término		Interpretación
• En inglés	• En español	• ¿A qué pregunta responde?
True Positive (TP)	Verdadero positivo	Responde a la pregunta de cuántas veces el modelo clasificó correctamente una muestra positiva como positiva.
False Negative (FN)	Falso negativo	Responde a la pregunta de cuántas veces el modelo clasificó incorrectamente una muestra positiva como negativa.
False Positive (FP)	Falso positivo	Responde a la pregunta de cuántas veces el modelo clasificó incorrectamente una muestra negativa como positiva.
True Negative (TN)	Verdadero negativo	Responde a la pregunta de cuántas veces el modelo clasificó correctamente una muestra negativa como negativa.

Fuente: elaboración propia

En la siguiente tabla, se detalla cómo calcular cada métrica de rendimiento y algunas consideraciones.

Tabla 7: Cálculo y consideraciones de las métricas de desempeño

Métrica • En inglés	• En español	Cálculo	Consideraciones
Overall Accuracy (OA)	Exactitud o Precisión general	Se calcula como la suma entre los verdaderos positivos (TP) y los verdaderos negativos (TN) sobre el total de casos clasificados (TP, TN, FP y FN).	Es la cantidad de predicciones de muestras positivas correctas. Es de útil importancia cuando todas las clases tienen la misma importancia ya que describe el rendimiento general del modelo. Caso contrario, la precisión puede ser engañosa. Entre más alto es el valor, mayor es el desempeño del modelo.
Recall	Exhaustividad o Sensibilidad	Se determina entre el número de casos verdaderos positivos (TP) y la suma de los verdaderos positivos (TP) y los falsos negativos (FN).	Es la tasa de verdaderos positivos. Mide el desempeño del modelo en detectar las muestras positivas. Este valor es 1 cuando todas las muestras positivas son correctamente detectadas; de otro lado, si es 0, significa que el modelo es incapaz de detectar muestras positivas. Entre mayor es el recall, más confiable es el modelo en detectar muestras positivas.
Precision	Precisión	Se calcula como la relación entre las muestras verdaderas positivas (TP) sobre el total de casos clasificados como verdaderos positivos (TP) y falsos positivos (FP)	Es la tasa de casos positivos detectados. Mide la exactitud del modelo al clasificar una muestra positiva.
Specificity	Especificidad	Se calcula como el valor dado entre los verdaderos negativos (TN) y la suma de verdaderos negativos (TN) y falsos positivos (FP)	Mide la exactitud del modelado al clasificar una muestra negativa. Es 1 cuando todas las muestras negativas son correctamente detectadas; es 0 cuando el modelo no es capaz de detectar muestras negativas. Es la tasa de verdaderos negativos.

Negative Predictive Value (NPV)	Valor predictivo negativo	Se calcula como la relación entre los verdaderos negativos (TN) y la suma de los verdaderos negativos (TN) y los falsos negativos (FN)	Es la tasa de casos negativos detectados. Es la exactitud del modelo en clasificar una muestra negativa.
---------------------------------	---------------------------	--	---

Fuente: Elaboración propia

Existe también otra métrica muy comúnmente utilizada, F1 Score, que se describe como la media armónica entre la precisión y la sensibilidad y es adecuada con conjuntos de datos desequilibrados [44].

Figura 17. Formula F1-Score

$$\text{F1 Score} = \frac{2 \times \text{TP}}{2 \times \text{TP} + \text{FN} + \text{FP}}$$

Fuente: Maxwell, A. E., Warner, T. A., & Guillén, L. A. (2021) [43]

Se resalta que para una clasificación multiclase, la precisión se evalúa por clase y, luego, se determina una precisión promedio [45]. Entonces, para aplicar la matriz de confusión y determinar las respectivas métricas, las pruebas se dividirán en las siete categorías de colores aprendidas. De esta forma, la matriz de confusión permitirá visualizar el nivel de confusión de predicción por color y, luego, se calculará la precisión promedio del modelo.

3.2 Resultados de la evaluación

El modelo fue evaluado en base a las siete categorías aprendidas en el entrenamiento y validación, mediante 140 imágenes de prueba. Dado que cada clase de cada categoría tiene el mismo valor, se realizó de la siguiente forma:

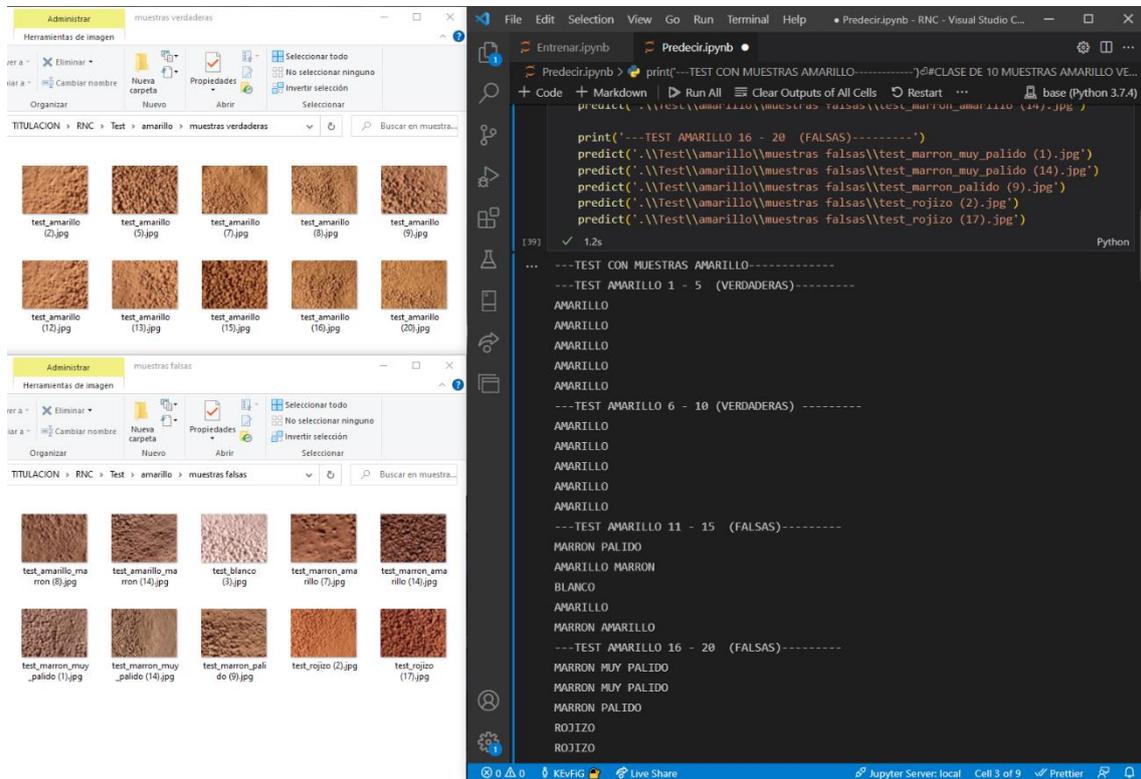
- Evaluar 10 predicciones de muestras verdaderas (el color real de las muestras sí es de la categoría correspondiente)
- Evaluar 10 predicciones de muestras falsas elegidas aleatoriamente (el color real de las muestras es de las demás categorías)

Por lo tanto, por cada categoría, se hicieron 20 pruebas de predicción.

3.2.1 Evaluación muestras de color Amarillo

En figura, se muestran las imágenes de test utilizadas y el resultado de las predicciones del modelo.

Figura 18. Evidencia Test de color Amarillo



Fuente: Elaboración propia

Se tabularon los resultados obtenidos en la matriz.

Tabla 8. Matriz de confusión - Amarillo

		Predicción	
		Positivo	Negativo
Realidad	Verdadero	10	0
	Falso	1	9

Fuente: Elaboración propia

Se calculan las métricas de rendimiento a partir de los valores de la matriz.

Tabla 9. Métricas de rendimiento - Amarillo

Métrica	Fórmula	Cálculo	Resultado	%
Overall Accuracy (Exactitud)	$\frac{TP + TN}{TP + TN + FP + FN}$	$\frac{10 + 9}{10 + 9 + 0 + 1} = \frac{19}{20}$	0,95	95
Recall (Sensibilidad)	$\frac{TP}{TP + FN}$	$\frac{10}{10 + 1} = \frac{10}{11}$	0,91	91
Precision (Precisión)	$\frac{TP}{TP + FP}$	$\frac{10}{10 + 0} = \frac{10}{10}$	1	100
Specificity (Especificidad)	$\frac{TN}{TN + FP}$	$\frac{9}{9 + 0} = \frac{9}{9}$	1	100
NPV	$\frac{TN}{TN + FN}$	$\frac{9}{9 + 1} = \frac{9}{10}$	0,90	90
F1-Score	$\frac{2 * TP}{2 * TP + FN + FP}$	$\frac{2 * 10}{2 * 10 + 1 + 0} = \frac{20}{21}$	0,95	95

Fuente: Elaboración propia

Análisis rápido de los resultados:

Tabla 10. Análisis métricas - Amarillo

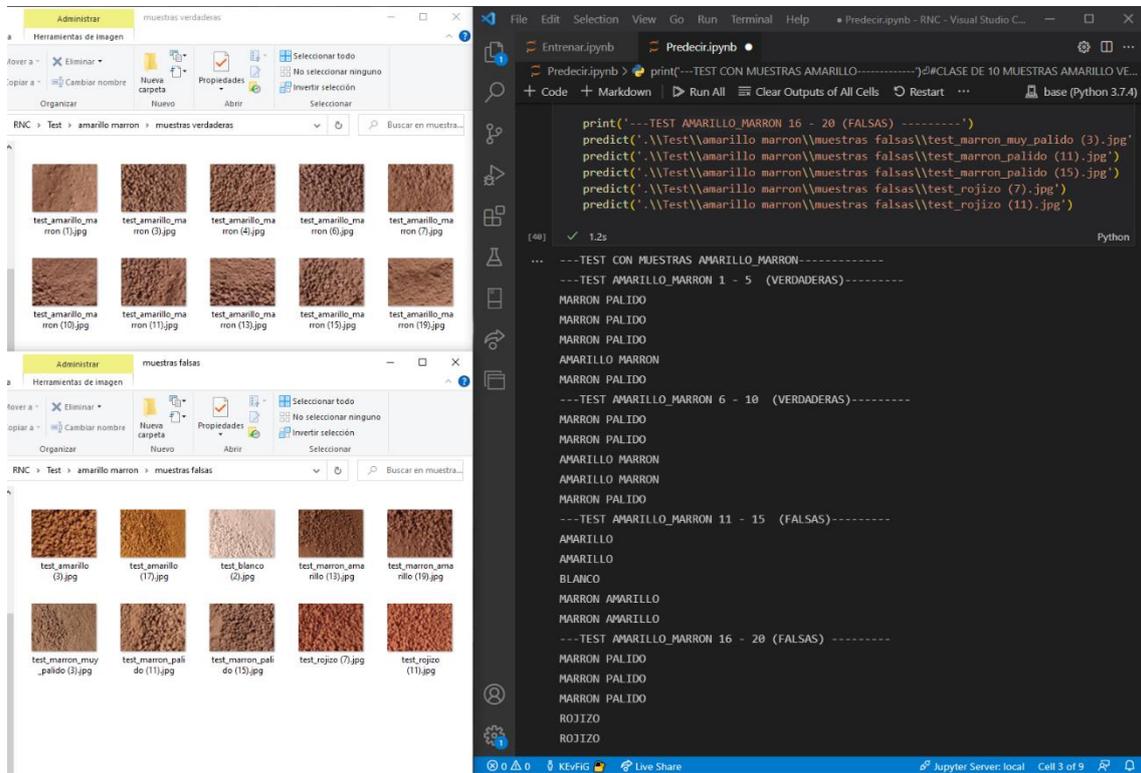
Métrica	Valor (%)	Análisis
Exactitud	95	El modelo tiene una precisión general del 95%, por lo tanto, es confiable en predecir las muestras amarillas.
Sensibilidad	91	Las muestras predichas como color amarillo son confiables al 91%.
Precisión	100	Las muestras realmente amarillas son acertadas en su predicción al 100%.

Fuente: Elaboración propia

3.2.2 Evaluación muestras de color Amarillo Marrón

En figura, se visualiza las imágenes de test utilizadas y los resultados de la predicción del modelo.

Figura 19. Evidencia Test de color Amarillo Marrón



Fuente: Elaboración propia

Tabulación de los resultados de la matriz.

Tabla 11. Matriz de confusión - Amarillo Marrón

		Predicción	
		Positivo	Negativo
Realidad	Verdadero	3	7
	Falso	0	10

Fuente: Elaboración propia

Se calculan las métricas de rendimiento.

Tabla 12. Métricas de rendimiento - Amarillo Marrón

Métrica	Fórmula	Cálculo	Resultado	%
Overall Accuracy (Exactitud)	$\frac{TP + TN}{TP + TN + FP + FN}$	$\frac{3 + 10}{3 + 10 + 7 + 0} = \frac{13}{20}$	0,65	65
Recall (Sensibilidad)	$\frac{TP}{TP + FN}$	$\frac{3}{3 + 0} = \frac{3}{3}$	1	100
Precision (Precisión)	$\frac{TP}{TP + FP}$	$\frac{3}{3 + 7} = \frac{3}{10}$	0,30	30
Specificity (Especificidad)	$\frac{TN}{TN + FP}$	$\frac{10}{10 + 7} = \frac{10}{17}$	0,59	59
NPV	$\frac{TN}{TN + FN}$	$\frac{10}{10 + 0} = \frac{10}{10}$	1	100
F1-Score	$\frac{2 * TP}{2 * TP + FN + FP}$	$\frac{2 * 3}{2 * 3 + 0 + 7} = \frac{6}{13}$	0,46	46

Fuente: Elaboración propia

Análisis rápido de los resultados:

Tabla 13. Análisis métricas - Amarillo Marrón

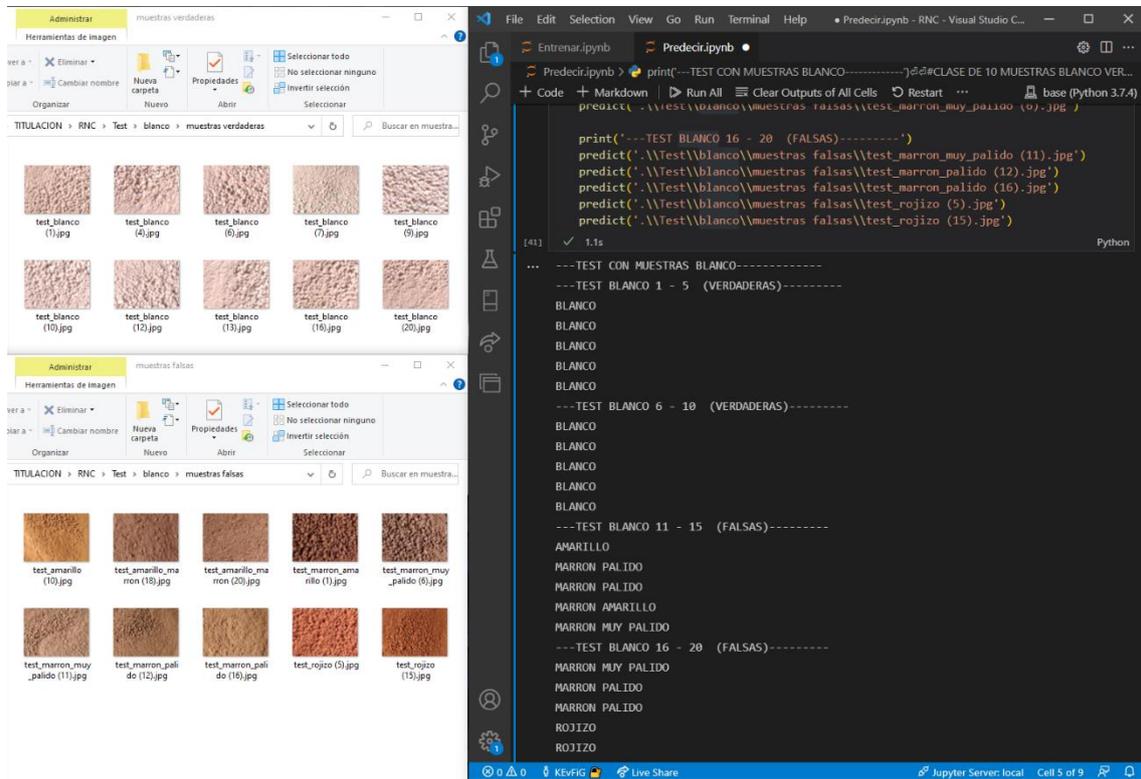
Métrica	Valor (%)	Análisis
Exactitud	65	La precisión general del modelo es del 65%, por ende, no es muy exacto en predecir las muestras amarillo marrón.
Sensibilidad	100	El modelo dificulta en esta categoría en general, sin embargo, las muestras predichas como amarillo marrón son confiables al 100%.
Precisión	30	Las muestras realmente de color amarillo marrón son acertadas solo al 30% en la predicción.

Fuente: Elaboración propia

3.2.3 Evaluación muestras de color Blanco

En figura, se encuentran las imágenes para el test y los resultados de predicción del modelo.

Figura 20. Evidencia Test de color Blanco



Fuente: Elaboración propia

Tabulación de los resultados.

Tabla 14. Matriz de confusión - Blanco

		Predicción	
		Positivo	Negativo
Realidad	Verdadero	10	0
	Falso	0	10

Fuente: Elaboración propia

Cálculo de las métricas de rendimiento para el color Blanco.

Tabla 15. Métricas de rendimiento - Blanco

Métrica	Fórmula	Cálculo	Resultado	%
Overall Accuracy (Exactitud)	$\frac{TP + TN}{TP + TN + FP + FN}$	$\frac{10 + 10}{10 + 10 + 0 + 0} = \frac{20}{20}$	1	100
Recall (Sensibilidad)	$\frac{TP}{TP + FN}$	$\frac{10}{10 + 0} = \frac{10}{10}$	1	100
Precision (Precisión)	$\frac{TP}{TP + FP}$	$\frac{10}{10 + 0} = \frac{10}{10}$	1	100
Specificity (Especificidad)	$\frac{TN}{TN + FP}$	$\frac{10}{10 + 0} = \frac{10}{10}$	1	100
NPV	$\frac{TN}{TN + FN}$	$\frac{10}{10 + 0} = \frac{10}{10}$	1	100
F1-Score	$\frac{2 * TP}{2 * TP + FN + FP}$	$\frac{2 * 10}{2 * 10 + 0 + 0} = \frac{20}{20}$	1	100

Fuente: Elaboración propia

Análisis rápido de los resultados:

Tabla 16. Análisis métricas - Blanco

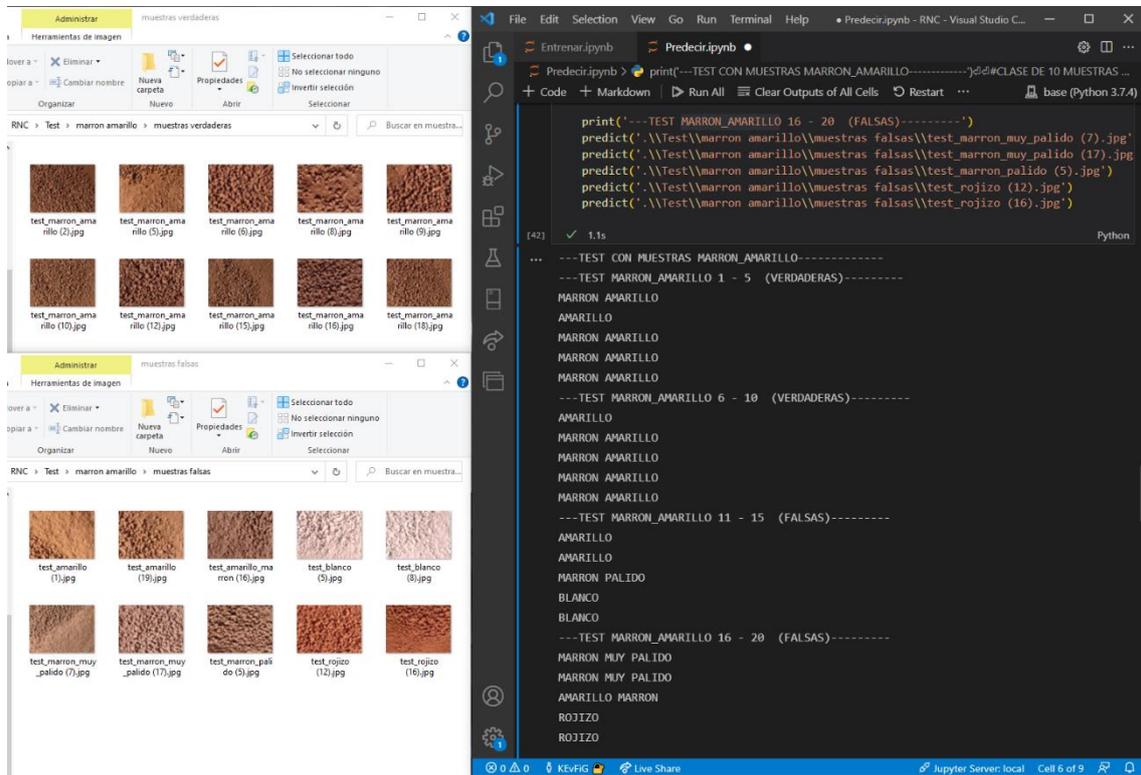
Métrica	Valor (%)	Análisis
Exactitud	100	El modelo tiene una precisión general del 100% en predicción de las muestras blancas, así que, es totalmente confiable.
Sensibilidad	100	Las muestras predichas como blancas son exactas al 100%.
Precisión	100	Las muestras realmente de color blanco son acertadas al 100% por el modelo.

Fuente: Elaboración propia

3.2.4 Evaluación muestras de color Marrón Amarillo

En figura, se ubican las imágenes de testeo y los resultados del modelo en la predicción.

Figura 21. Evidencia Test de color Marrón Amarillo



Fuente: Elaboración propia

Se tabularon los resultados obtenidos.

Tabla 17. Matriz de confusión - Marrón Amarillo

		Predicción	
		Positivo	Negativo
Realidad	Verdadero	8	2
	Falso	0	10

Fuente: Elaboración propia

Se realiza el cálculo de las métricas de rendimiento a partir de los valores de la matriz de confusión.

Tabla 18. Métricas de rendimiento - Marrón Amarillo

Métrica	Fórmula	Cálculo	Resultado	%
Overall Accuracy (Exactitud)	$\frac{TP + TN}{TP + TN + FP + FN}$	$\frac{8 + 10}{8 + 10 + 2 + 0} = \frac{18}{20}$	0,90	90
Recall (Sensibilidad)	$\frac{TP}{TP + FN}$	$\frac{8}{8 + 0} = \frac{8}{8}$	1	100
Precision (Precisión)	$\frac{TP}{TP + FP}$	$\frac{8}{8 + 2} = \frac{8}{10}$	0,80	80
Specificity (Especificidad)	$\frac{TN}{TN + FP}$	$\frac{10}{10 + 2} = \frac{10}{12}$	0,83	83
NPV	$\frac{TN}{TN + FN}$	$\frac{10}{10 + 0} = \frac{10}{10}$	1	100
F1-Score	$\frac{2 * TP}{2 * TP + FN + FP}$	$\frac{2 * 8}{2 * 8 + 0 + 2} = \frac{16}{18}$	0,89	89

Fuente: Elaboración propia

Análisis rápido de los resultados:

Tabla 19. Análisis métricas - Marrón Amarillo

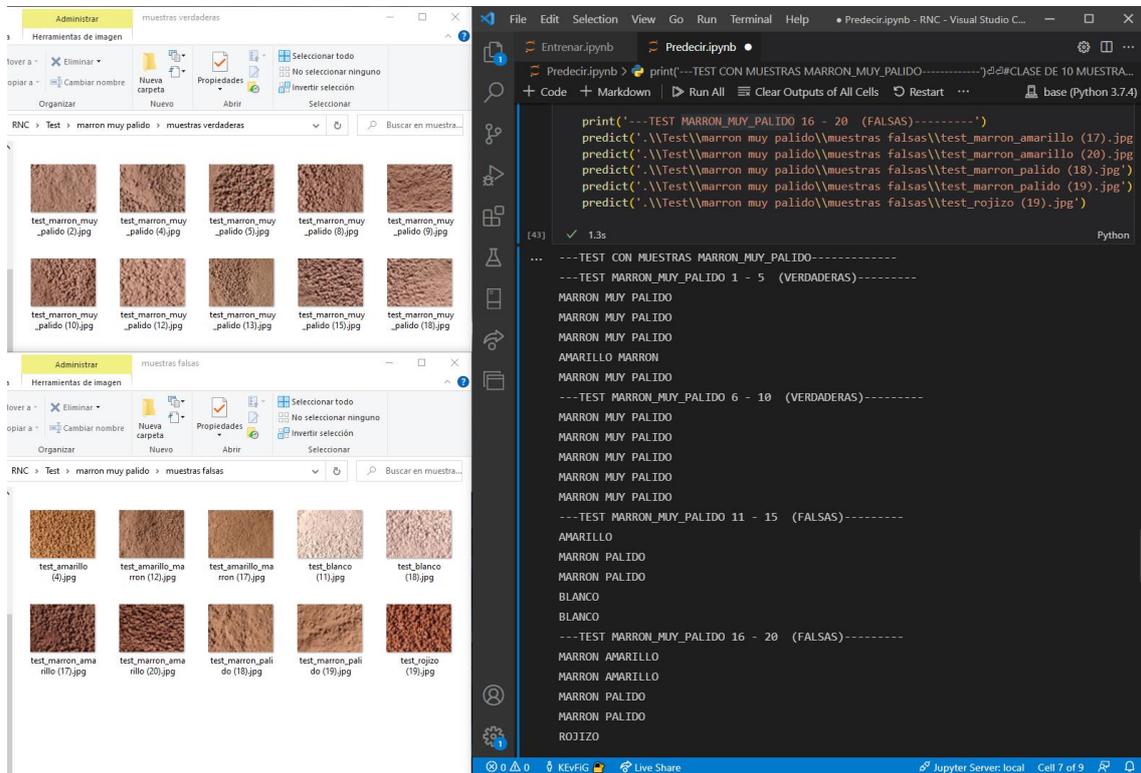
Métrica	Valor (%)	Análisis
Exactitud	90	El modelo tiene una precisión general del 90%, por lo tanto, es muy confiable en la predicción de muestras marrón amarillo.
Sensibilidad	100	Las muestras predichas como marrón amarillo son exactas al 100%.
Precisión	80	Las muestras realmente marrón amarillo son acertadas en la predicción al 80%.

Fuente: Elaboración propia

3.2.5 Evaluación muestras de color Marrón muy pálido

En figura, se muestran las imágenes de test utilizadas y el resultado de predicción del modelo.

Figura 22. Evidencia Test de color Marrón muy pálido



Fuente: Elaboración propia

Se tabula los resultados conseguidos en la matriz de confusión.

Tabla 20. Matriz de confusión - Marrón muy pálido

		Predicción	
		Positivo	Negativo
Realidad	Verdadero	9	1
	Falso	0	10

Fuente: Elaboración propia

Se calcula el valor de las métricas de rendimiento en base a los valores de la matriz.

Tabla 21. Métricas de rendimiento - Marrón muy pálido

Métrica	Fórmula	Cálculo	Resultado	%
Overall Accuracy (Exactitud)	$\frac{TP + TN}{TP + TN + FP + FN}$	$\frac{9 + 10}{9 + 10 + 1 + 0} = \frac{19}{20}$	0,95	95
Recall (Sensibilidad)	$\frac{TP}{TP + FN}$	$\frac{9}{9 + 0} = \frac{9}{9}$	1	100
Precision (Precisión)	$\frac{TP}{TP + FP}$	$\frac{9}{9 + 1} = \frac{9}{10}$	0,90	90
Specificity (Especificidad)	$\frac{TN}{TN + FP}$	$\frac{10}{10 + 1} = \frac{10}{11}$	0,91	91
NPV	$\frac{TN}{TN + FN}$	$\frac{10}{10 + 0} = \frac{10}{10}$	1	100
F1-Score	$\frac{2 * TP}{2 * TP + FN + FP}$	$\frac{2 * 9}{2 * 9 + 0 + 1} = \frac{18}{19}$	0,95	95

Fuente: Elaboración propia

Análisis rápido de los resultados:

Tabla 22. Análisis métricas - Marrón muy pálido

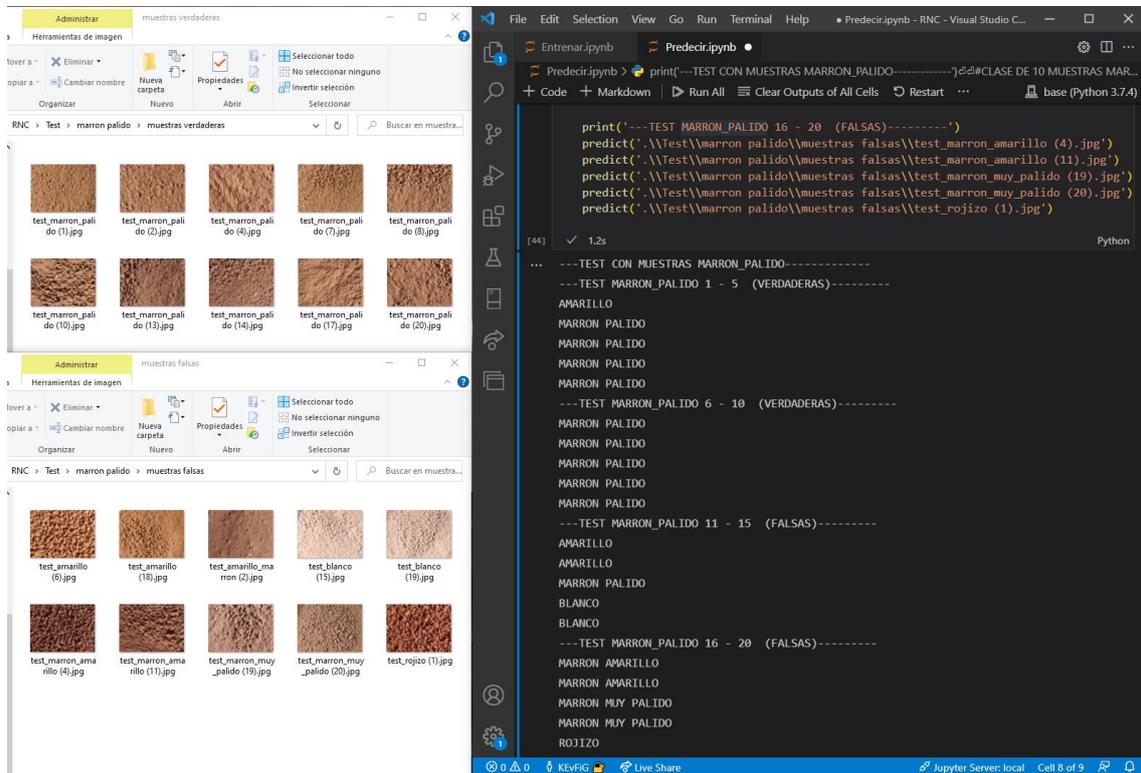
Métrica	Valor (%)	Análisis
Exactitud	95	La precisión general del modelo resulta ser del 95%, así que es muy exacto en predecir las muestras de color marrón muy pálido.
Sensibilidad	100	Las muestras predichas como marrón muy pálido son justas al 100%.
Precisión	90	En las predicciones, las muestras realmente de color marrón muy pálido son acertadas al 90%.

Fuente: Elaboración propia

3.2.6 Evaluación muestras de color Marrón Pálido

En figura, se ubican las imágenes de testeo utilizadas y el resultado de las predicciones del modelo.

Figura 23. Evidencia Test de color Marrón Pálido



Fuente: Elaboración propia

Se tabula los resultados obtenidos en la matriz.

Tabla 23. Matriz de confusión - Marrón Pálido

		Predicción	
		Positivo	Negativo
Realidad	Verdadero	9	1
	Falso	1	9

Fuente: Elaboración propia

Se calculan las métricas de rendimiento a partir de los valores de la matriz.

Tabla 24. Métricas de rendimiento - Marrón Pálido

Métrica	Fórmula	Cálculo	Resultado	%
Overall Accuracy (Exactitud)	$\frac{TP + TN}{TP + TN + FP + FN}$	$\frac{9 + 9}{9 + 9 + 1 + 1} = \frac{18}{20}$	0,90	90
Recall (Sensibilidad)	$\frac{TP}{TP + FN}$	$\frac{9}{9 + 1} = \frac{9}{10}$	0,90	90
Precision (Precisión)	$\frac{TP}{TP + FP}$	$\frac{9}{9 + 1} = \frac{9}{10}$	0,90	90
Specificity (Especificidad)	$\frac{TN}{TN + FP}$	$\frac{9}{9 + 1} = \frac{9}{10}$	90	90
NPV	$\frac{TN}{TN + FN}$	$\frac{9}{9 + 1} = \frac{9}{10}$	0,90	90
F1-Score	$\frac{2 * TP}{2 * TP + FN + FP}$	$\frac{2 * 9}{2 * 9 + 1 + 1} = \frac{18}{20}$	0,90	90

Fuente: Elaboración propia

Análisis rápido de los resultados:

Tabla 25. Análisis métricas - Marrón Pálido

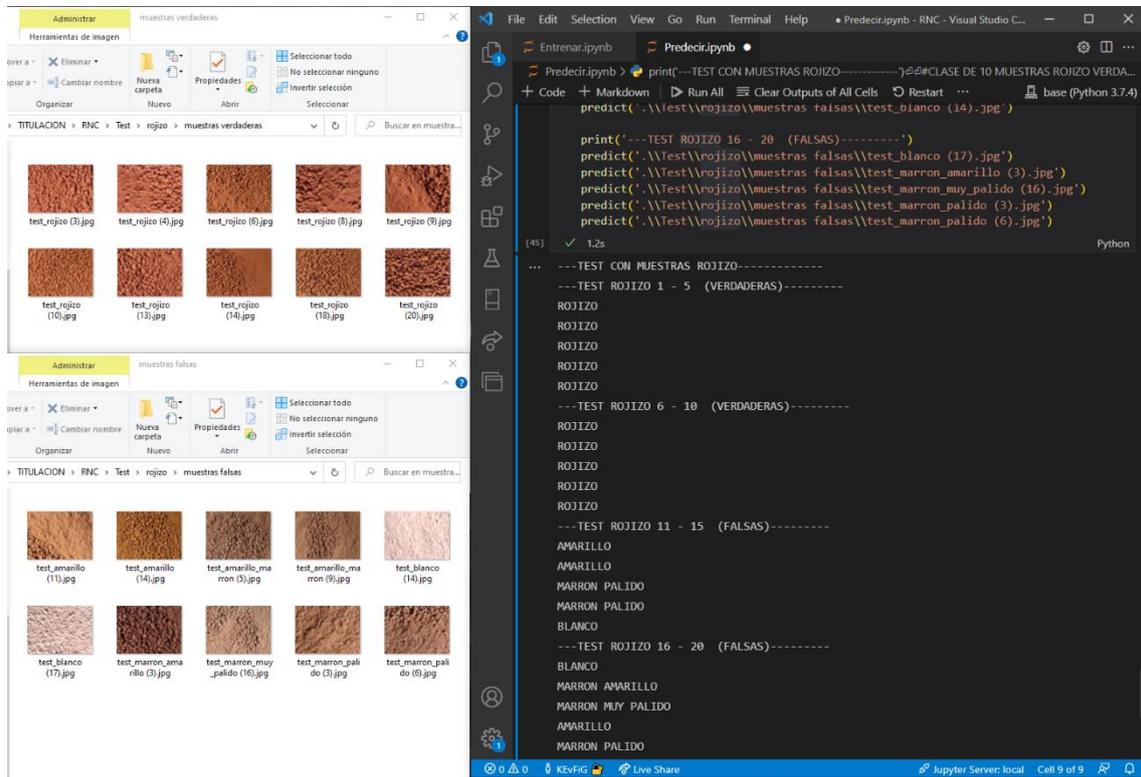
Métrica	Valor (%)	Análisis
Exactitud	90	La precisión general es del 90%, por lo tanto, la confiabilidad en predecir las muestras marrón pálido es alta.
Sensibilidad	90	Las muestras predichas como marrón pálido son exactas al 90%.
Precisión	90	Las muestras realmente de color marrón pálido se aciertan al 90%.

Fuente: Elaboración propia

3.2.7 Evaluación muestras de color Rojizo

En figura, se muestran las imágenes de test utilizadas y el resultado de predicción del modelo.

Figura 24. Evidencia Test de color Rojizo



Fuente: Elaboración propia

Se tabula los resultados conseguidos en la matriz de confusión.

Tabla 26. Matriz de confusión - Rojizo

		Predicción	
		Positivo	Negativo
Realidad	Verdadero	10	0
	Falso	0	10

Fuente: Elaboración propia

Se calculan las métricas de rendimiento para el color Rojizo, a partir de los valores en la matriz.

Tabla 27. Métricas de rendimiento - Rojizo

Métrica	Fórmula	Cálculo	Resultado	%
Overall Accuracy (Exactitud)	$\frac{TP + TN}{TP + TN + FP + FN}$	$\frac{10 + 10}{10 + 10 + 0 + 0} = \frac{20}{20}$	1	100
Recall (Sensibilidad)	$\frac{TP}{TP + FN}$	$\frac{10}{10 + 0} = \frac{10}{10}$	1	100
Precision (Precisión)	$\frac{TP}{TP + FP}$	$\frac{10}{10 + 0} = \frac{10}{10}$	1	100
Specificity (Especificidad)	$\frac{TN}{TN + FP}$	$\frac{10}{10 + 0} = \frac{10}{10}$	1	100
NPV	$\frac{TN}{TN + FN}$	$\frac{10}{10 + 0} = \frac{10}{10}$	1	100
F1-Score	$\frac{2 * TP}{2 * TP + FN + FP}$	$\frac{2 * 10}{2 * 10 + 0 + 0} = \frac{20}{20}$	1	100

Fuente: Elaboración propia

Análisis rápido de los resultados:

Tabla 28. Análisis métricas - Rojizo

Métrica	Valor (%)	Análisis
Exactitud	100	El modelo tiene una precisión general del 100% en el color rojizo, por lo tanto, la predicción es completamente confiable.
Sensibilidad	100	Las muestras predichas como color rojizo son atinadas al 100%.
Precisión	100	Las muestras realmente rojizas son acertadas al 100% en las predicciones.

Fuente: Elaboración propia

3.2.8 Promedio de resultados obtenidos

A continuación, se presenta una tabla con el cálculo de los promedios de rendimiento, obtenidos en las pruebas hechas por categoría.

Tabla 29. Promedios de las métricas de rendimiento

Métrica Color	Overall Accuracy (Exactitud)	Recall (Sensibilidad)	Precision (Precisión)	Specificity (Especificidad)	NPV	F1-Score
Amarillo	0,95	0,91	1	1	0,90	0,95
Amarillo Marrón	0,65	1	0,30	0,59	1	0,46
Blanco	1	1	1	1	1	1
Marrón amarillo	0,90	1	0,80	0,83	1	0,89
Marrón muy pálido	0,95	1	0,90	0,91	1	0,95
Marrón pálido	0,90	0,90	0,90	0,90	0,90	0,90
Rojizo	1	1	1	1	1	1
PROMEDIO	0,91	0,97	0,84	0,89	0,97	0,88
En (%)	91	97	84	89	97	88

Fuente: Elaboración propia

3.2.9 Análisis global de los resultados del modelo

Para facilitar el análisis, se presenta la siguiente tabla de resumen.

Tabla 30. Resumen de los resultados de las pruebas

Color	Número de Imágenes de Test	Imágenes Correctamente clasificadas
Amarillo	20	20
Amarillo Marrón	20	4
Blanco	20	20
Marrón amarillo	20	17
Marrón muy pálido	20	18
Marrón pálido	20	17
Rojizo	20	20
Total	140	116
Overall Accuracy (Exactitud)		0,91
Precision (Precisión)		0,97
Recall (Sensibilidad)		0,84

Fuente: Elaboración propia

Se recalca que los resultados obtenidos son el reflejo del desempeño de un modelado de red neuronal convolucional que ha sido entrenado para demostrar predicciones de colores en muestras estandarizadas de suelo.

NOTA: El dataset utilizado para el aprendizaje del modelo tiene un total de 4480 imágenes (70% para entrenamiento – 30% para validación) y consta de siete categorías diferentes igualadas, cada una representando un color.

A continuación, en la última tabla de análisis:

Tabla 31. Análisis de las métricas de rendimiento del modelo

Métrica	Valor (%)	Análisis
Exactitud	91	El modelo resultante tiene una exactitud del 91%; por ende, la confiabilidad de las predicciones es alta.
Sensibilidad Precisión	97 84	Se resalta que el modelo obtenido tiene una precisión alta del 97% y una sensibilidad considerable del 84%.

Fuente: Elaboración propia

Se puede decir que el modelo CNN maneja bien las clases aprendidas en línea general, porque se evaluaron todas las categorías por iguales y se obtuvo 116 aciertos sobre 140 pruebas; por lo tanto, una predicción acertada real del 82.86%.

Se observa eso sí que el reconocimiento real en las pruebas realizadas (82.86%), es más bajo del reconocimiento esperado (tipo un 90%). Esto se debe a que el amarillo marrón no fue aprendido como debido. De hecho, de ese color, apenas 4 muestras de 20 fueron reconocidas correctamente; y las métricas correspondientes de rendimiento son la evidencia del mal aprendizaje.

CONCLUSIONES

En base al trabajo realizado para la generación de un modelo de predicción de colores del suelo, se concluye lo siguiente:

- Se desarrolló un modelo de red neuronal para el reconocimiento del suelo mediante el color, aplicando técnicas de aprendizaje profundo con Python y librerías de Keras y Tensorflow.
- Se utilizó el aprendizaje profundo para reconocer los colores de muestras de suelo mediante redes neuronales convolucionales. Esto demuestra que es posible fomentar el estudio del suelo y aportar a las investigaciones de las ciencias agropecuarias mediante la aplicación de técnicas de la inteligencia artificial. Por lo tanto, no se descarta que es posible determinar ulteriores características del suelo, en particular, aquellas físicas; por ende, el presente trabajo de titulación abre puertas a posibles futuras investigaciones sobre el ámbito.
- Se realizó el trabajo de campo en la provincia de El Oro para recolectar las muestras de suelo. Estas luego fueron elaboradas y estandarizadas para adecuarlas al análisis correcto del color.
- Se analizó el color de las muestras en laboratorio mediante las tablas de Munsell para entrenar el modelo de predicción y comprobar luego los resultados otorgados. Se destaca que es un proceso sencillo de realizar, pero muy sujeto a posibles errores humanos: ciertas muestras pueden presentar colores muy parecidos que confunden fácilmente la vista, entonces dar resultados ambiguos cuando las tonalidades del color son semejantes.
- Se tomaron fotografías a las muestras del suelo recolectadas para crear el dataset de aprendizaje del modelo. Se resalta que es uno de los procesos críticos del trabajo: se tuvieron que considerar diferentes aspectos delicados como la configuración de la cámara, la estabilidad y la posición, las condiciones de luz y exposición.
- Se evaluó el rendimiento del modelo con 140 pruebas de reconocimiento, utilizando matrices de confusión para analizar los resultados de predicción del modelo, donde se puede destacar un alto rendimiento general del prototipo y un porcentaje de reconocimiento real del 82.86%.

RECOMENDACIONES

- Se recomienda analizar el color de las muestras con las tablas de Munsell las veces necesarias para asegurar los resultados. Los colores pueden ser muy parecidos y su correcta identificación es fundamental para categorizar bien las imágenes en carpetas cuando se arma el dataset de entrenamiento y aprendizaje.
- Considerar muy atentamente el lugar y la posición donde tomar las fotos de las muestras de suelo porque las condiciones de luz y exposición influyen mucho en la calidad de las fotos realizadas.
- Se propone también realizar una toma fotográfica previa a un reducido número de muestras para verificar la calidad de las imágenes. Malas configuraciones de la cámara, como la sensibilidad de ISO y el tipo de lente, pueden afectar gravemente las tomas y reflejar en la fotografía colores de suelo demasiado alterados.
- Trabajar con un equipo de buena potencia de cálculo y con imágenes de tamaño reducido no superior a los 500 píxeles de ancho y alto.

BIBLIOGRAFÍA

- [1] M. Supriya y V. K. Chattu, «A Review of Artificial Intelligence, Big Data, and Blockchain Technology Applications in Medicine and Global Health», *Big Data Cogn. Comput.*, vol. 5, n.º 3, p. 41, sep. 2021, doi: 10.3390/bdcc5030041.
- [2] A. Palinski, «Forecasting gas demand using artificial intelligence methods», *Naft.-Gaz*, n.º 2, pp. 111-117, feb. 2019, doi: 10.18668/NG.2019.02.07.
- [3] L. Chen, P. Chen, y Z. Lin, «Artificial Intelligence in Education: A Review», *IEEE Access*, vol. 8, pp. 75264-75278, 2020, doi: 10.1109/ACCESS.2020.2988510.
- [4] V. Moysiadis, P. Sarigiannidis, V. Vitsas, y A. Khelifi, «Smart Farming in Europe», *Comput. Sci. Rev.*, vol. 39, p. 100345, feb. 2021, doi: 10.1016/j.cosrev.2020.100345.
- [5] N. Muthukrishnan, F. Maleki, K. Ovens, C. Reinhold, B. Forghani, y R. Forghani, «Brief History of Artificial Intelligence», *Neuroimaging Clinics of North America*, vol. 30, n.º 4, pp. 393-399, nov. 2020, doi: 10.1016/j.nic.2020.07.004.
- [6] «Agricultura, la base de la economía y la alimentación – Ministerio de Agricultura y Ganadería». <https://www.agricultura.gob.ec/agricultura-la-base-de-la-economia-y-la-alimentacion/> (accedido 6 de diciembre de 2021).
- [7] R. R. Poppiel *et al.*, «Soil Color and Mineralogy Mapping Using Proximal and Remote Sensing in Midwest Brazil», *Remote Sens.*, vol. 12, n.º 7, p. 1197, abr. 2020, doi: 10.3390/rs12071197.
- [8] N. P. Kirillova, J. Grauer-Gray, A. E. Hartemink, T. M. Sileova, Z. S. Artemyeva, y E. K. Burova, «New perspectives to use Munsell color charts with electronic devices», *Comput. Electron. Agric.*, vol. 155, pp. 378-385, dic. 2018, doi: 10.1016/j.compag.2018.10.028.
- [9] R. Bujnovsky, J. Vilcek, M. Lorincova, y M. Kudla, «Agricultural Soil and Freshwater Ecosystem Services in Slovakia - Opportunities and Challenges for Their Practical Application», *Folia Geogr.*, vol. 63, n.º 1, pp. 110-122, 2021.
- [10] M. J. Wilson, «The importance of parent material in soil classification: A review in a historical context», *Catena*, vol. 182, p. 104131, nov. 2019, doi: 10.1016/j.catena.2019.104131.
- [11] A. A. Nikiforova, «Soil Classification», *Knowl. Organ.*, vol. 46, n.º 6, pp. 467-488, 2019, doi: 10.5771/0943-7444-2019-6-467.
- [12] P. Srivastava, A. Shukla, y A. Bansal, «A comprehensive review on soil classification using deep learning and computer vision techniques», *Multimed. Tools Appl.*, vol. 80, n.º 10, pp. 14887-14914, abr. 2021, doi: 10.1007/s11042-021-10544-5.
- [13] B. Sun *et al.*, «Effects of freeze-thaw on soil properties and water erosion», *Soil Water Res.*, vol. 16, n.º 4, pp. 205-216, 2021, doi: 10.17221/143/2020-SWR.

- [14] E. K. Bunemann *et al.*, «Soil quality - A critical review», *Soil Biol. Biochem.*, vol. 120, pp. 105-125, may 2018, doi: 10.1016/j.soilbio.2018.01.030.
- [15] S. Inazumi, S. Intui, A. Jotisankasa, S. Chairakaikeow, y K. Kojima, «Artificial intelligence system for supporting soil classification», *Results Eng.*, vol. 8, p. 100188, dic. 2020, doi: 10.1016/j.rineng.2020.100188.
- [16] S. K. Das, P. Samui, A. K. Sabat, y T. G. Sitharam, «Prediction of swelling pressure of soil using artificial intelligence techniques», *Environ. Earth Sci.*, vol. 61, n.º 2, pp. 393-403, jul. 2010, doi: 10.1007/s12665-009-0352-6.
- [17] A. A. Javadi y M. Rezaia, «Applications of artificial intelligence and data mining techniques in soil modeling», *Geomech. Eng.*, vol. 1, n.º 1, pp. 53-74, mar. 2009, doi: 10.12989/gae.2009.1.1.053.
- [18] L. Kouadio, R. C. Deo, V. Byrareddy, J. F. Adamowski, S. Mushtaq, y V. P. Nguyen, «Artificial intelligence approach for the prediction of Robusta coffee yield using soil fertility properties», *Comput. Electron. Agric.*, vol. 155, pp. 324-338, dic. 2018, doi: 10.1016/j.compag.2018.10.014.
- [19] A. A. R. Madushanki, M. N. Halgamuge, W. A. H. S. Wirasagoda, y A. Syed, «Adoption of the Internet of Things (IoT) in Agriculture and Smart Farming towards Urban Greening: A Review», *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, n.º 4, pp. 11-28, abr. 2019, doi: 10.14569/IJACSA.2019.0100402.
- [20] M. N. Akhtar, A. J. Shaikh, A. Khan, H. Awais, E. A. Bakar, y A. R. Othman, «Smart Sensing with Edge Computing in Precision Agriculture for Soil Assessment and Heavy Metal Monitoring: A Review», *Agriculture-Basel*, vol. 11, n.º 6, p. 475, jun. 2021, doi: 10.3390/agriculture11060475.
- [21] M. Whiting, S. W. Salley, D. K. James, J. W. Karl, y C. W. Brungard, «Rapid bulk density measurement using mobile device photogrammetry», *Soil Sci. Soc. Am. J.*, vol. 84, n.º 3, pp. 811-817, jun. 2020, doi: 10.1002/saj2.20063.
- [22] C. Gonzalez Garcia, E. Rolando Nunez-Valdez, V. Garcia-Diaz, B. C. Pelayo G-Bustelo, y J. M. Cueva Lovelle, «A Review of Artificial Intelligence in the Internet of Things», *Int. J. Interact. Multimed. Artif. Intell.*, vol. 5, n.º 4, pp. 9-20, mar. 2019, doi: 10.9781/ijimai.2018.03.004.
- [23] C. J. Fluke y C. Jacobs, «Surveying the reach and maturity of machine learning and artificial intelligence in astronomy», *Wiley Interdiscip. Rev.-Data Mining Knowl. Discov.*, vol. 10, n.º 2, p. e1349, mar. 2020, doi: 10.1002/widm.1349.
- [24] M. A. Boden, *Inteligencia Artificial*. Turner, 2017.
- [25] R. Benítez, G. Escudero, S. Kanaan, y D. M. Rodó, *Inteligencia artificial avanzada*. Editorial UOC, 2014.
- [26] J. Martinez Torres, C. Iglesias Comesana, y P. J. Garcia-Nieto, «Review: machine learning techniques applied to cybersecurity», *Int. J. Mach. Learn. Cybern.*, vol. 10, n.º 10, pp. 2823-2836, oct. 2019, doi: 10.1007/s13042-018-00906-1.
- [27] M. Rashid, H. Singh, y V. Goyal, «The use of machine learning and deep learning algorithms in functional magnetic resonance imaging-A systematic

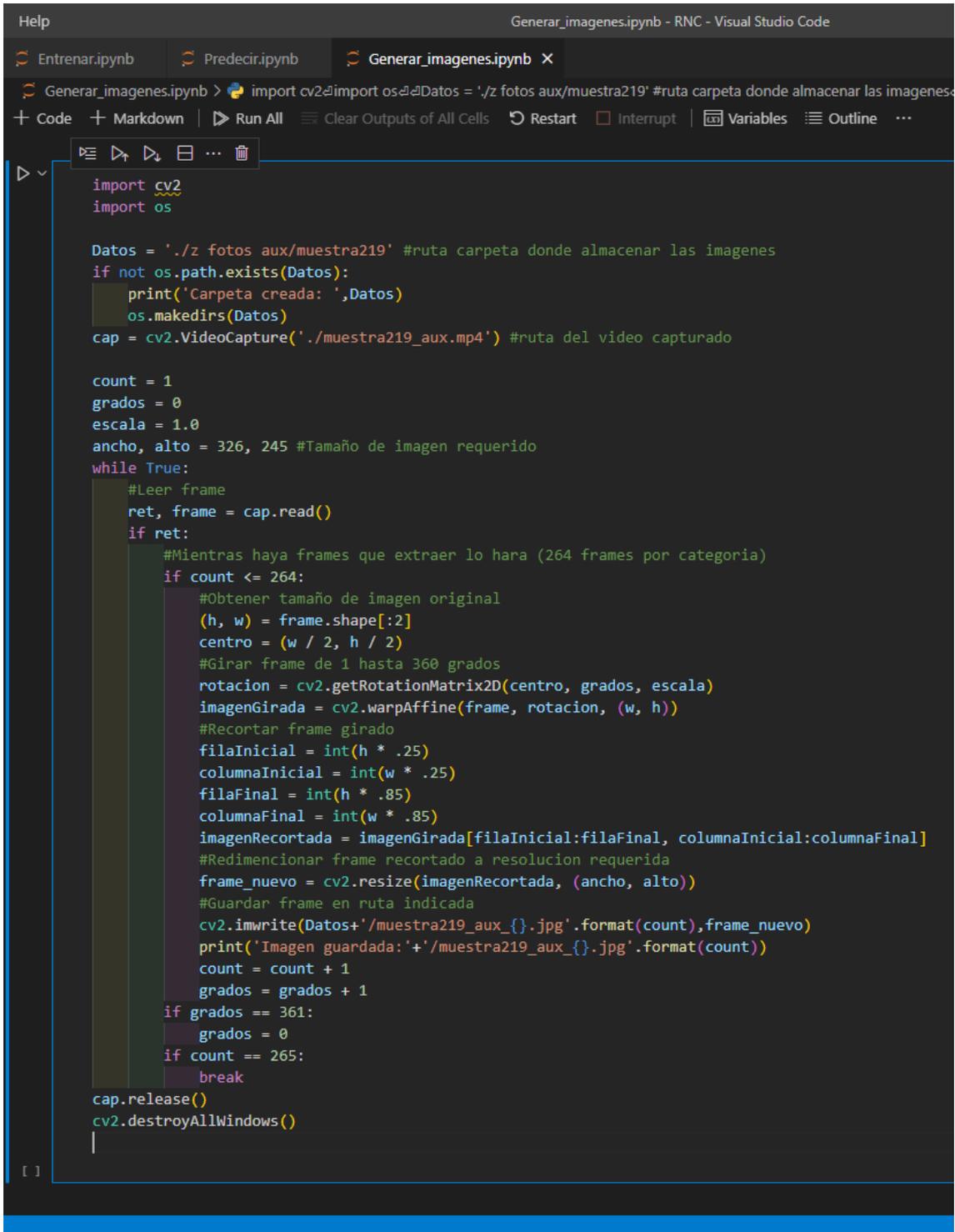
- review», *Expert Syst.*, vol. 37, n.º 6, p. e12644, dic. 2020, doi: 10.1111/exsy.12644.
- [28] R. Mu y X. Zeng, «A Review of Deep Learning Research», *KSII Trans. Internet Inf. Syst.*, vol. 13, n.º 4, pp. 1738-1764, abr. 2019, doi: 10.3837/tiis.2019.04.001.
- [29] X. Yan, S. Hu, Y. Mao, Y. Ye, y H. Yu, «Deep multi-view learning methods: A review», *Neurocomputing*, vol. 448, pp. 106-129, ago. 2021, doi: 10.1016/j.neucom.2021.03.090.
- [30] M. H. Pereira da Cruz, D. S. Domingues, P. T. Maeda Saito, A. R. Paschoal, y P. H. Bugatti, «TERL: classification of transposable elements by convolutional neural networks», *Brief. Bioinform.*, vol. 22, n.º 3, p. bbaa185, may 2021, doi: 10.1093/bib/bbaa185.
- [31] D. Goularas y S. Kamis, «Evaluation of Deep Learning Techniques in Sentiment Analysis from Twitter Data», en *2019 International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML)*, ago. 2019, pp. 12-17. doi: 10.1109/Deep-ML.2019.00011.
- [32] S. Ghaffarian, J. Valente, M. van der Voort, y B. Tekinerdogan, «Effect of Attention Mechanism in Deep Learning-Based Remote Sensing Image Processing: A Systematic Literature Review», *Remote Sens.*, vol. 13, n.º 15, p. 2965, ago. 2021, doi: 10.3390/rs13152965.
- [33] J. M. Ede, «Deep learning in electron microscopy», *Mach. Learn.: Sci. Technol.*, vol. 2, n.º 1, p. 011004, mar. 2021, doi: 10.1088/2632-2153/abd614.
- [34] B. Pang, E. Nijkamp, y Y. N. Wu, «Deep Learning With TensorFlow: A Review», *J. Educ. Behav. Stat.*, vol. 45, n.º 2, pp. 227-248, abr. 2020, doi: 10.3102/1076998619872761.
- [35] S. Morita, «Chemometrics and Related Fields in Python», *Anal. Sci.*, vol. 36, n.º 1, pp. 107-112, ene. 2020, doi: 10.2116/analsci.19R006.
- [36] M. Perez-Enciso y L. M. Zingaretti, «A Guide on Deep Learning for Complex Trait Genomic Prediction», *Genes*, vol. 10, n.º 7, p. 553, jul. 2019, doi: 10.3390/genes10070553.
- [37] S. A. Schmidt y C. Ahn, «A Comparative Review of Methods of Using Soil Colors and their Patterns for Wetland Ecology and Management», *Commun. Soil Sci. Plant Anal.*, vol. 50, n.º 11, pp. 1293-1309, jun. 2019, doi: 10.1080/00103624.2019.1604737.
- [38] J. Nalepa, M. Marcinkiewicz, y M. Kawulok, «Data Augmentation for Brain-Tumor Segmentation: A Review», *Front. Comput. Neurosci.*, vol. 13, p. 83, dic. 2019, doi: 10.3389/fncom.2019.00083.
- [39] Á. M. Lerma Sánchez *et al.*, «Clasificación automática de anastomosis mediante redes neuronales convolucionales en video fetoscópico», *RIDE. Revista Iberoamericana para la Investigación y el Desarrollo Educativo*, vol. 11, n.º 22, jun. 2021, doi: 10.23913/ride.v11i22.856.
- [40] J. M. Domínguez Soto, A. D. Román Gutiérrez, F. Prieto García, y O. Acevedo Sandoval, «Sistema de Notación Munsell y CIELab como

- herramienta para evaluación de color en suelos», *Revista mexicana de ciencias agrícolas*, vol. 3, n.º 1, pp. 141-155, feb. 2012.
- [41] S. Ye, R. G. Pontius, y R. Rakshit, «A review of accuracy assessment for object-based image analysis: From per-pixel to per-polygon approaches», *ISPRS-J. Photogramm. Remote Sens.*, vol. 141, pp. 137-147, jul. 2018, doi: 10.1016/j.isprsjprs.2018.04.002.
- [42] A. E. Maxwell, T. A. Warner, y L. A. Guillen, «Accuracy Assessment in Convolutional Neural Network-Based Deep Learning Remote Sensing Studies-Part 1: Literature Review», *Remote Sens.*, vol. 13, n.º 13, p. 2450, jul. 2021, doi: 10.3390/rs13132450.
- [43] A. E. Maxwell, T. A. Warner, y L. A. Guillen, «Accuracy Assessment in Convolutional Neural Network-Based Deep Learning Remote Sensing Studies-Part 2: Recommendations and Best Practices», *Remote Sens.*, vol. 13, n.º 13, p. 2591, jul. 2021, doi: 10.3390/rs13132591.
- [44] D. Sisodia y D. S. Sisodia, «Data Sampling Strategies for Click Fraud Detection Using Imbalanced User Click Data of Online Advertising: An Empirical Review», *IETE Tech. Rev.*, doi: 10.1080/02564602.2021.1915892.
- [45] E. Lieskovska, M. Jakubec, R. Jarina, y M. Chmulik, «A Review on Speech Emotion Recognition Using Deep Learning and Attention Mechanism», *Electronics*, vol. 10, n.º 10, p. 1163, may 2021, doi: 10.3390/electronics10101163.

ANEXOS

Anexo I. Generar_imagenes.ipynb

Figura 25. Generar_imagenes.ipynb



```
Help Generar_imagenes.ipynb - RNC - Visual Studio Code
Entrenar.ipynb Predecir.ipynb Generar_imagenes.ipynb X
Generar_imagenes.ipynb > import cv2 import os Datos = './z fotos aux/muestra219' #ruta carpeta donde almacenar las imagenes
+ Code + Markdown Run All Clear Outputs of All Cells Restart Interrupt Variables Outline ...
import cv2
import os

Datos = './z fotos aux/muestra219' #ruta carpeta donde almacenar las imagenes
if not os.path.exists(Datos):
    print('Carpeta creada: ',Datos)
    os.makedirs(Datos)
cap = cv2.VideoCapture('./muestra219_aux.mp4') #ruta del video capturado

count = 1
grados = 0
escala = 1.0
ancho, alto = 326, 245 #Tamaño de imagen requerido
while True:
    #Leer frame
    ret, frame = cap.read()
    if ret:
        #Mientras haya frames que extraer lo hara (264 frames por categoria)
        if count <= 264:
            #Obtener tamaño de imagen original
            (h, w) = frame.shape[:2]
            centro = (w / 2, h / 2)
            #Girar frame de 1 hasta 360 grados
            rotacion = cv2.getRotationMatrix2D(centro, grados, escala)
            imagenGirada = cv2.warpAffine(frame, rotacion, (w, h))
            #Recortar frame girado
            filaInicial = int(h * .25)
            columnaInicial = int(w * .25)
            filaFinal = int(h * .85)
            columnaFinal = int(w * .85)
            imagenRecortada = imagenGirada[filaInicial:filaFinal, columnaInicial:columnaFinal]
            #Redimensionar frame recortado a resolucioin requerida
            frame_nuevo = cv2.resize(imagenRecortada, (ancho, alto))
            #Guardar frame en ruta indicada
            cv2.imwrite(Datos+'muestra219_aux_{}.jpg'.format(count),frame_nuevo)
            print('Imagen guardada: '+'/muestra219_aux_{}.jpg'.format(count))
            count = count + 1
            grados = grados + 1
        if grados == 361:
            grados = 0
        if count == 265:
            break
    cap.release()
    cv2.destroyAllWindows()
```

Fuente: Elaboración propia

Anexo II. Entrenar.ipynb – Dataset

Figura 26. Entrenar.ipynb - Dataset



```
Help Entrenar.ipynb - RNC - Visual Studio Code
Entrenar.ipynb X Entrenar.ipynb (output) Predecir.ipynb
Entrenar.ipynb > # Parametros: epocas = 6 # Numero de iteraciones para aprender y predecir la data longitud,
+ Code + Markdown | ▶ Run All | Clear Outputs of All Cells | Outline ...
Suma Total de imagenes en subdirs: 4480

labels=[]
indice=0
for cantidad in dircount:
    for i in range(cantidad):
        labels.append(indice)
        indice=indice+1
print("Cantidad etiquetas creadas: ",len(labels))

deportes=[]
indice=0
archivoTxt = open(".\Model\labels.txt", "w")
for directorio in directories:
    name = directorio.split(os.sep)
    print(indice , name[len(name)-1])
    deportes.append(name[len(name)-1])
    # Crear archivo TXT de etiquetas
    archivoTxt.write(name[len(name)-1])
    archivoTxt.write("\n")
    indice=indice+1
archivoTxt.close()

y = np.array(labels)
X = np.array(images, dtype=np.uint8) # convierto de lista a numpy

# Find the unique numbers from the train labels
clases = np.unique(y)
nClases = len(clases)
print('Total number of outputs : ', nClases)
print('Output classes : ', clases)

... Cantidad etiquetas creadas: 4480
0 AMARILLO
1 AMARILLO MARRON
2 BLANCO
3 MARRON AMARILLO
4 MARRON MUY PALIDO
5 MARRON PALIDO
6 ROJIZO
Total number of outputs : 7
Output classes : [0 1 2 3 4 5 6]

# Mezclar todo y crear los grupos de entrenamiento y testing
```

Fuente: Elaboración propia

Anexo III. Entrenar.ipynb – CNN

Figura 27. Entrenar.ipynb - CNN

```
Help Entrenar.ipynb - RNC - Visual Studio Code
Entrenar.ipynb x Entrenar.ipynb (output) Predecir.ipynb
Entrenar.ipynb > # Parametros: epocas = 6 # Numero de iteraciones para aprender y predecir la data
longitud, altura = 326, 245 # Tamano de imagenes a procesar en px
+ Code + Markdown ▶ Run All ☰ Clear Outputs of All Cells | ☰ Outline ...

# Parametros
epocas = 6 # Numero de iteraciones para aprender y predecir la data
longitud, altura = 326, 245 # Tamano de imagenes a procesar en px
batch_size = 32 # Rango de imagenes a analizar por cada paso (entrenamiento y validacion)
tamano_pool = (2, 2) # Tamano de matriz de caracteristicas especificas
lr = 0.0001 # Tasa de aprendizaje

cnn = Sequential()
# Extraccion de caracteristicas #1
cnn.add(Convolution2D(32, (3, 3), padding="same", activation="relu", input_shape=(altura, longitud, 3)))
cnn.add(MaxPooling2D(pool_size=tamano_pool))
# Extraccion de caracteristicas #2
cnn.add(Convolution2D(64, (3, 3), padding="same", activation="relu"))
cnn.add(MaxPooling2D(pool_size=tamano_pool))
# Extraccion de caracteristicas #3
cnn.add(Convolution2D(64, (3, 3), padding="same", activation="relu"))
cnn.add(MaxPooling2D(pool_size=tamano_pool))
# Extraccion de caracteristicas #4
cnn.add(Convolution2D(64, (3, 3), padding="same", activation="relu"))
cnn.add(MaxPooling2D(pool_size=tamano_pool))
# Extraccion de caracteristicas #5
cnn.add(Convolution2D(128, (3, 3), padding="same", activation="relu"))
cnn.add(MaxPooling2D(pool_size=tamano_pool))
# Capa bidimensional o aplanadora
cnn.add(Flatten())
cnn.add(Dropout(0.5))
# Capa full conectada #1
cnn.add(Dense(600, activation="relu"))
cnn.add(Dropout(0.5))
# Capa full conectada #2
cnn.add(Dense(128, activation="relu"))
cnn.add(Dropout(0.5))
# Capa full conectada final
cnn.add(Dense(nClases, activation='softmax'))

cnn.summary() # Ver resumen de la estructura de capas

# Indicaciones de perdida, ganancia, optimizacion y entrenamiento del modelo
cnn.compile(loss='categorical_crossentropy',
            optimizer=optimizers.Adam(learning_rate=lr),
            metrics=['accuracy'])

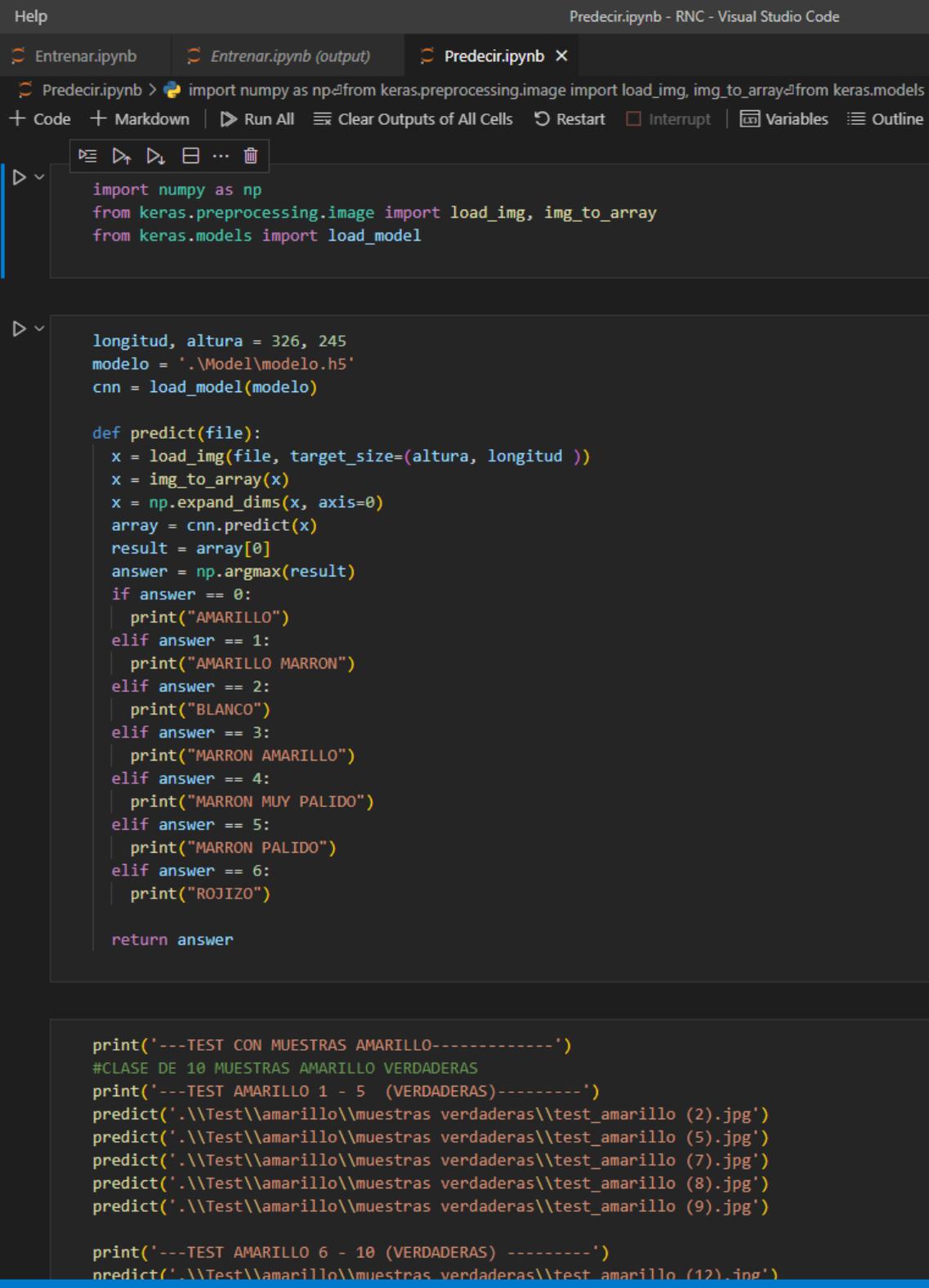
salida = cnn.fit(train_X, train_label, batch_size=batch_size, epochs=epocas, verbose=1, validation_data=(valid_X, valid_label))

# Evaluacion del modelo
scores = cnn.evaluate(valid_X, valid_label, verbose=0)
print("Accuracy: %.2f%%" % (scores[1]*100))
```

Fuente: Elaboración propia

Anexo IV. Predecir.ipynb

Figura 28. Predecir.ipynb



```
Help Predecir.ipynb - RNC - Visual Studio Code
Entrenar.ipynb Entrenar.ipynb (output) Predecir.ipynb X
Predecir.ipynb > import numpy as np
from keras.preprocessing.image import load_img, img_to_array
from keras.models import load_model

longitud, altura = 326, 245
modelo = './Model\modelo.h5'
cnn = load_model(modelo)

def predict(file):
    x = load_img(file, target_size=(altura, longitud))
    x = img_to_array(x)
    x = np.expand_dims(x, axis=0)
    array = cnn.predict(x)
    result = array[0]
    answer = np.argmax(result)
    if answer == 0:
        print("AMARILLO")
    elif answer == 1:
        print("AMARILLO MARRON")
    elif answer == 2:
        print("BLANCO")
    elif answer == 3:
        print("MARRON AMARILLO")
    elif answer == 4:
        print("MARRON MUY PALIDO")
    elif answer == 5:
        print("MARRON PALIDO")
    elif answer == 6:
        print("ROJIZO")

    return answer

print('---TEST CON MUESTRAS AMARILLO-----')
#CLASE DE 10 MUESTRAS AMARILLO VERDADERAS
print('---TEST AMARILLO 1 - 5 (VERDADERAS)-----')
predict('./Test\amarillo\muestras verdaderas\test_amarillo (2).jpg')
predict('./Test\amarillo\muestras verdaderas\test_amarillo (5).jpg')
predict('./Test\amarillo\muestras verdaderas\test_amarillo (7).jpg')
predict('./Test\amarillo\muestras verdaderas\test_amarillo (8).jpg')
predict('./Test\amarillo\muestras verdaderas\test_amarillo (9).jpg')

print('---TEST AMARILLO 6 - 10 (VERDADERAS) -----')
predict('./Test\amarillo\muestras verdaderas\test_amarillo (12).jpg')
```

Fuente: Elaboración propia