MÉTODO AUTOMÁTICO COMPUTACIONAL PARA LA DETECCIÓN DE RESIDUOS SÓLIDOS EN PARQUES

GABRIELA RINCON RAMIREZ - 67000199

CARLOS FELIPE BAQUERO BARRETO - 67000184

PROGRAMA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN

FACULTAD DE INGENIERÍA

UNIVERSIDAD CATOLICA DE COLOMBIA

MODALIDAD TRABAJO DE INVESTIGACIÓN TECNOLÓGICA

BOGOTÁ D.C.

2022

MÉTODO AUTOMÁTICO COMPUTACIONAL PARA LA DETECCIÓN DE RESIDUOS SÓLIDOS EN PARQUES

GABRIELA RINCON RAMIREZ - 67000199
CARLOS FELIPE BAQUERO BARRETO - 67000184

ESTE TRABAJO DE GRADO ES PRESENTADO COMO REQUISITO PARA OPTAR AL TÍTULO DE: INGENIERO DE SISTEMAS Y COMPUTACIÓN

ASESOR:
JUAN CARLOS BARRERO CALIXTO

ALTERNATIVA:
TRABAJO DE INVESTIGACIÓN TECNOLÓGICA

UNIVERSIDAD CATÓLICA DE COLOMBIA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
MODALIDAD TRABAJO DE INVESTIGACIÓN TECNOLÓGICA
BOGOTÁ D.C.
2022

Atribución-NoComercial-SinDerivadas 4.0 Internacional (CC BY-NC-ND 4.0)

This is a human-readable summary of (and not a substitute for) the license. Advertencia.

Usted es libre de:

Compartir — copiar y redistribuir el material en cualquier medio o formato

La licenciante no puede revocar estas libertades en tanto usted siga los términos de la licencia

Bajo los siguientes términos:

Atribución — Usted debe dar crédito de manera adecuada, brindar un enlace a la licencia, e indicar si se han realizado cambios. Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que usted o su uso tienen el apoyo de la licenciante.

NoComercial — Usted no puede hacer uso del material con propósitos comerciales.

SinDerivadas — Si remezcla, transforma o crea a partir del material, no podrá distribuir el material modificado.

No hay restricciones adicionales — No puede aplicar términos legales ni medidas tecnológicas que restrinjan legalmente a otras a hacer cualquier uso permitido por la licencia.

https://creativecommons.org/licenses/by-nc-nd/4.0/deed.es

TABLA DE CONTENIDO

RESUMEN 12
1. INTRODUCCIÓN 15
1. GENERALIDADES 17
1.1 LÍNEA DE INVESTIGACIÓN 17
1.2 PLANTEAMIENTO DEL PROBLEMA 18
1.2.1. Descripción del problema 18
1.2.2. Formulación del problema 20
1.3.1. Objetivo general 21
1.3.2. Objetivos específicos 21
1.4. JUSTIFICACIÓN 22
1.5. DELIMITACIONES 24
1.5.1. Limitaciones 24
1.5.2. Alcance 24
2. MARCO DE REFERENCIA 25
2.1. MARCO CONCEPTUAL 25
2.1.1 Contaminación ambiental 25
2.1.2. Recolección de residuos sólidos 26
2.1.3. Separación en la fuente 26
2.1.4. Basura en parques 27
2.1.5. Reciclaje 27
2.1.6. Tipos de reciclaje 28
2.1.7. Plástico 28
2.1.8. Vidrio 29
2.1.9. Metal 29
2.1.10. Residuos orgánicos 30
2.1.11. Ciencias de la computación 30
2.1.12. Inteligencia artificial 30
2.1.12. Aprendizaje de máquina 31
2.1.13. Tipos de aprendizaje de máquina 31

2.1.14. Reconocimiento de imágenes 31
2.1.15. Conjunto de datos de entrenamiento 32
2.1.16. Medidas de desempeño 32
2.2. MARCO TEÓRICO 33
2.2.1. Teoría de aprendizaje de máquina 33
2.2.2. Preprocesamiento 33
2.2.3. Tipos de muestreo 33
2.2.4. Extracción de características 34
2.2.5. Técnicas de extracción de características: 34
2.2.6. Anaconda 34
2.2.7. Spyder 34
2.2.8. Python 35
2.2.9. TensorFlow 35
2.2.10. Librerías o frameworks de Python para cálculos numéricos y Análisis de datos 35
2.2.11. Numpy 35
2.2.12. Redes neuronales convoluciones 36
2.2.13. RELU 36
2.2.14. VGG-16: 36
2.2.15. AlexNet: 37
2.2.16. KNN: 38
2.1.17. Aprendizaje de máquina: 39
2.2.18. Recall: 39
2.2.19. MaxPooling: 40
2.3. ESTADO DEL ARTE 41
3. METODOLOGÍA 47
3.1. Obtención del dataset 47
3.2. Categorización de los datos 48
3.3. Transformación de los datos 48
3.4. Selección del método de clasificación de las imágenes 48
3.5. Entrenamiento y pruebas de algoritmos 49

3.6. Evaluación de medidas de desempeño 49
3.7. Diseño del método automático computacional 49
3.8. Rediseñar y afinar el método 49
4. DISEÑO METODOLÓGICO 50
4.1. CONSTRUCCIÓN DEL DATASET 50
4.1.1. Búsqueda de fuentes 50
4.1.2. Selección de las imágenes 51
4.1.3. Etiquetado de las imágenes seleccionadas 51
4.1.4. Organización del conjunto de datos en Google Drive 53
4.2.1. Búsqueda de algoritmos 55
4.3. PROPUESTAS DE MÉTODOS AUTOMÁTICOS COMPUTACIONALES 62
4.4.1. VGG16 66
4.4.2. VGG19 71
4.4.3. YOLOV3 71
4.5. EVALUACIÓN DE RENDIMIENTO DE LOS MÉTODOS AUTOMÁTICOS 73
4.5.1. VGG 16 73
4.5.2. VGG19 74
4.5.3. YOLOV3 75
5. RESULTADOS Y ANÁLISIS DE RESULTADOS 76
5.1. Construcción del conjunto de datos de imágenes 76
5.2. Diseño de la estrategia metodológica 78
5.3. Desarrollo del método automático 79
5.4 Evaluación de rendimiento del método automático 80
5.5 Algoritmo seleccionado 84
6. IMPACTO SOCIAL 84
7. CONCLUSIONES 85
8. RECOMENDACIONES Y TRABAJOS FUTUROS 86
9. BIBLIOGRAFÍA 86

ÍNDICE DE FIGURAS

Figura 1 Solid Waste Management in the World's Cities 23
Figura 2 Separación en la fuente 27
Figura 3 Plástico 28
Figura 4 VGG-16 37
Figura 5 AlexNet 38
Figura 6 KNN 39
Figura 7 Recall and precisión 40
Figura 8 Metodología del proyecto 47
Figura 9 Archivo de clases predefinida 52
Figura 10 LabelImg configuración YOLO 53
Figura 11 Organización de conjunto de datos en Google Drive 54
Figura 12 *Estructura del algoritmo VGG16* 56
Figura 13 Capas y convoluciones en VGG16 57
Figura 14 Arquitectura del modelo VGG19 60
Figura 15 Tratamiento de imágenes modelo YOLOV3 61
Figura 16 Parámetros del modelo 66
Figura 17 montaje almacenamiento de drive 67
Figura 18 Directorios en Google Drive 67
Figura 19 Generador de imágenes de train y valid 68
Figura 20 Generador de imágenes de train y valid 2 68
Figura 21 división de imágenes train vs samples 69
Figura 22 Parámetros del modelo a entrenar 69
Figura 23 Entrenamiento del modelo 69
Figura 24 Guardado del modelo en Google Colab 70
Figura 25 Librerías utilizadas para el desarrollo de los modelos 71
Figura 26 activación entorno virtual 72
Figura 27 Última época de ejecución VGG 16 modelo 1 73
Figura 28. Última época de ejecución VGG 19 modelo 2 74
Figura 29. Última época de ejecución YoloV3 modelo 3 75
Figura 30. Unidad en Google Drive conjunto de imágenes 77
Figura 31 Matriz de confusión VGG 16 81
Figura 32. Matriz de confusión VGG 19 82

ÍNDICE DE TABLAS

T*abla* 1 *Configuración de VGG16* 57
Tabla 2 Configuración de las redes neuronales sin las capas ReLu 59
Tabla 3 Modelos utilizados 63
Tabla 4 Modelos usados y sus librerías 64
Tabla 5 Análisis de medidas de desempeño 80

DEDICATORIA

A todas aquellas personas que fueron parte de este camino, nuestra familia, amigos y pareja que nos apoyaron siempre con su amor y palabras de aliento para no desfallecer en este camino de formación profesional.

AGRADECIMIENTOS

Agradezco a Dios por darme la fuerza para no desfallecer en ningún momento, a mi mamá Olga Patricia Ramírez Montoya por estar siempre a mi lado en este camino apoyándome de todas las maneras posibles con su gran amor. A nuestro asesor Juan Carlos Barrero por ser guía, por su paciencia al explicarnos, por brindarnos sus conocimientos y experiencia para la realización de este proyecto. A mi compañero en este proyecto Carlos Felipe Baquero, por sus conocimientos, inteligencia y esfuerzo, por la buena comunicación que tuvimos en esto, espero que podamos desarrollar más proyectos juntos.

Agradezco a mi familia por haberme dado la oportunidad de estudiar, también a mis profesores que siempre fueron pilar fundamental para esta investigación y por último, pero no menos importante a mi compañera que fue fundamental para el desarrollo de la investigación.

NOTA DE ACEPTACIÓN

_____

_____

_____

_____

Jurado

_____

Jurado

_____

Juan Carlos Barrero Calixto

Asesor

RESUMEN

La generación de grandes cantidades de residuos y el tratamiento de ellos se ha convertido en un asunto de gran preocupación global. Según el informe del Banco Mundial, se determinó que en el año 2016 se generaron 2010 millones de toneladas de basura en todo el planeta.[1]

El uso de tecnologías como el aprendizaje de máquina hace que se automaticen muchos procesos. Entre los enfoques del aprendizaje de máquina está el reconocimiento de imágenes con el que se puede de una manera más ágil por medio de imágenes identificar el tipo de residuo correspondiente además de permitir que los residuos que puedan ser reutilizables tengan otro ciclo de vida y permitir de que disminuya la contaminación medio ambiental.

La cantidad de residuos generados en lugares con gran afluencia de personas, como lo son los parques en Colombia son cantidades grandes, debido a que son lugares de atracción y ocio, muchos de estos residuos pueden tener otro ciclo de vida si se realiza una adecuada separación pueden ser reciclados y usados posteriormente para tener un nuevo uso.

La presente investigación consistió en el desarrollo de un método automático computacional implementando una técnica de aprendizaje de máquina con medidas de desempeño altas que permitió determinar el tipo de residuo en parques de la ciudad de Bogotá, por medio de reconocimiento de imágenes con enfoque de visión de computadora que clasifica la basura en categorías de residuos y que redujo el margen de error a la hora de realizar la clasificación de los residuos generados.

Dentro del presente proyecto se seleccionó el algoritmo con las mejores medidas de desempeño, se encontraron las técnicas de extracción de características y se

[1] BANCO MUNDIAL Informe del Banco Mundial: Los desechos a nivel mundial crecerán un 70 % para 2050, a menos que se adopten medidas urgentes {En línea} {15 de octubre de 2021} disponible en: https://www.bancomundial.org/es/news/press-release/2018/09/20/global-waste-to-grow-by-70-percent-by-2050-unless-urgent-action-is-taken-world-bank-report

obtuvo un método automático computacional que clasifica las imágenes de basura según su tipo utilizando un algoritmo que muestre métricas de desempeño adecuadas para la correcta clasificación de las imágenes.

Palabras clave: Reciclaje, recursos naturales, algoritmos, tecnología.

ABSTRACT

The generation of large amounts of waste and its treatment has become a matter of great global concern. According to the World Bank report, it was determined that in 2016, 2010 million tons of garbage were generated around the planet

The use of technologies such as machine learning makes many processes automated, among the approaches to machine learning is image recognition with which it is possible in a more agile way, through images, to identify the corresponding type of waste in addition to allow waste that can be reused to have another life cycle and allow environmental pollution to decrease.

The amount of waste generated in places with a large influx of people, such as parks in Colombia, because they are places of attraction and leisure, many of these wastes can have another life cycle if an adequate separation is carried out, they can be recycled. and used later to have a new life cycle.

This research dealt with the development of an automatic computational method implementing a machine learning technique with high-performance measures that allowed determining the type of waste in parks in the city of Bogotá, through image recognition with a vision approach of a computer that classifies garbage into waste categories and that reduced the margin of error when classifying the waste generated.

Within the present project, the algorithm with the best performance measures was selected, the feature extraction techniques were found, and an automatic computational method was obtained that classifies the garbage images according to their type using an algorithm that shows adequate performance metrics for correct classification of images.

Keywords: Recycling, natural resources, algorithms, technology.

1. INTRODUCCIÓN

El incremento de los residuos generados por los seres humanos interpela de una búsqueda constante de métodos, organización, estrategias que hagan realidad la correcta separación de estos, con el fin de que aquellos residuos que puedan ser reutilizados se les brinde un tratamiento correcto.

Muchos de los residuos que son generados en las ciudades se les puede dar un segundo uso, por medio del reciclaje. Es importante que se usen métodos o técnicas que permitan obtener grandes beneficios, que reduzcan el daño al medio ambiente y permitan la adecuada gestión de dichos residuos, según el contexto en el que se trate. Existen amplias técnicas enfocadas en el aprendizaje de máquina, que ayudan a la clasificación de imágenes, que servirán en este caso a la clasificación de los residuos, en sus diversas clases y categorías, ya que las personas aun suelen realizar este proceso de una manera muy manual, la cual ocasiona que se confundan o no reconozcan correctamente el recipiente de basura en la que debe dejar los residuos, en este caso provenientes de los parques en Colombia.

La recolección y tratamiento de residuos juegan un gran papel a la hora de hablar del medio ambiente y crecimiento de un país, según la Organización de las Naciones Unidas (ONU) la gestión de la basura en América Latina y el Caribe es uno de los mayores retos de sostenibilidad que tiene la región actualmente, "*Uno de los asuntos que aborda el informe es la creciente generación de residuos en la región, fruto del desarrollo y del crecimiento de la población, mientras el sector no se ha dotado de la capacidad suficiente para atender de forma adecuada a los mismos*", indica Pon. De hecho, cuarenta millones de personas carecen de acceso a la recolección de residuos.[2]

[2] NOTICIAS ONU. Basura y desarrollo en Latinoamérica {En línea} {15 de octubre 2021} Disponible en: https://news.un.org/es/story/2018/10/1443562

En la realización del presente trabajo de investigación el cual se enmarca dentro de la modalidad de investigación tecnológica, se planificó y diseñó un método automático computacional, primeramente seleccionando un dataset de imágenes acorde, extrayendo el conjunto de características de las imágenes, categorizando el conjunto de datos de imágenes, transformando los datos, posteriormente seleccionando el mejor algoritmo que se ajuste a las necesidades del contexto del proyecto y a la clasificación de imágenes, luego realizando el entrenamiento y pruebas de dicho algoritmo para luego realizar la evaluación de las medidas de desempeño.

Con la realización del método automático computacional, se van a tener impactos de distinta índole, como lo son sociales y culturales, ya que si la separación de residuos se realiza de una manera adecuada se reduce el impacto ambiental que tiene en el planeta tierra.

1. GENERALIDADES

1.1 LÍNEA DE INVESTIGACIÓN

La presente investigación para optar para el titulo de ingenieros de sistemas y computación se enmarca en la modalidad de investigación tecnológica.

1.2 PLANTEAMIENTO DEL PROBLEMA

1.2.1. Descripción del problema

En la actualidad, uno de los problemas que agobian a la población humana, es la contaminación del medio ambiente, entre otros factores, por la crisis climática. Según la ONU el problema climático que enfrenta el planeta, de diversas formas tales como incendios e inundaciones, afecta a las poblaciones más frágiles de los países, ya sean pobres o ricos, por lo cual es importante que se limite el aumento de la temperatura global al objetivo que se planteó internacionalmente. [3]

Según la revista Semana, en Colombia se generan alrededor de 12 millones de toneladas de basura al año, de esto solo se recicla un promedio de 17%. En Bogotá se producen aproximadamente 6.300 toneladas de basura al día y solo se reutilizan entre el 14% y 15%, según el Ministerio de Ambiente y Desarrollo Sostenible. En el país, además, se consumen 24 kilogramos de plástico por persona al año y de esa cifra el 56% es de un solo uso, es decir, pitillos, cubiertos, platos, tapas y envases, entre otros utensilios[4]

Se han realizado diversos esfuerzos para concientizar a las personas y los consumidores del impacto que tiene la correcta separación de los residuos sólidos en iniciativas como lo es el reciclaje. Según cifras del Ministerio de Medio Ambiente para el año 2018 el porcentaje de reciclaje era solo del 17% de las más

[3] NACIONES UNIDAS. La crisis climática - una carrera que podemos ganar {En línea} {16 de octubre} Disponible en: https://www.un.org/es/un75/climate-crisis-race-we-can-win

[4] REVISTA SEMANA. El 78% de los hogares colombianos no recicla {En línea} {16 de octubre 2021} disponible en: https://www.semana.com/medio-ambiente/articulo/el-78-de-los-hogares-colombianos-no-recicla/44231/

de 12 millones de toneladas de residuos sólidos que se producían, esto pese a los esfuerzos, iniciativas y políticas que se han realizado por parte del Gobierno.[5]

En Colombia el proceso de reciclaje se hace de manera muy manual, es decir, este proceso no está automatizado y lo realizan las personas encargadas del reciclaje. No se han aplicado tecnologías que ayuden a que este proceso sea más ágil y eficiente en cuanto a la clasificación de residuos además de ser un proceso riesgoso para aquellos individuos que separan la basura: problemas gastrointestinales de origen parasitario, bacteriano o viral. Dichos trabajadores que realizan la gestión de los residuos manualmente pueden sufrir accidentes como lo son: heridas en sus extremidades, cortes, heridas además de golpes, también tienden a tener enfermedades en la piel, ojos e infecciones de respiración. [6]

Desde enero del 2021, en Colombia se cambió el código de los colores para la separación de los residuos, John Alexander Chalarca Zapata, Magister en sostenibilidad dice que siempre ha existido una gran confusión en el hecho de saber en que recipiente deben ir determinados residuos, además que con el Ministerio de Ambiente y Desarrollo sostenible se emitió un informe con los nuevos códigos de colores según residuos.[7]

En países de Europa, se han implementado métodos que optimizan la recolección de basuras, su objetivo es optimizar los procesos para la recolección en residuos reciclables por medio de sistemas y sensores, el sistema detecta y

---

[5] EL NUEVO SIGLO. ¿Cómo vamos en Colombia con el reciclaje? {En línea} {16 de octubre de 2021} disponible en: https://www.elnuevosiglo.com.co/articulos/03-2019-como-vamos-en-colombia-con-el-reciclaje

[6] Riesgos por causas de los residuos sólidos. {En línea} {16 de octubre de 2021} disponible en: https://sites.google.com/site/residuossolidos113/11c

[7] UPB SOSTENIBLE. 5 Preguntas para entender el reciclaje en Colombia {En línea} {16 de octubre de 2021} disponible en: https://www.upb.edu.co/es/central-blogs/sostenibilidad/cinco-preguntas-para-entender-el-reciclaje

genera alarmas ante la necesidad de que se recojan residuos del bote de basura y que también tiene un sistema de localización que muestra rutas óptimas para la recogida[8]

La separación de residuos depende mucho del entorno en el que se realice, cada cultura la gestiona de manera diferente, se debe revisar un método automático, que nos ayude a realizar la identificación de imágenes de basura según el tipo.

En este proyecto es necesario considerar que no hay un data set ya construido con el cual trabajar, por ende es necesario construir o buscar un dataset que se ajuste a las necesidades del problema, tener en cuenta que si es necesario tener permisos para tomar las fotografías de la basura hay que buscar los centros comerciales para realizar este trabajo, también es necesario seleccionar los algoritmos que se acoplen a la condición de las imágenes del conjunto de datos a tratar, posteriormente implementar técnicas de extracción de características, realizar el entrenamiento del algoritmo, probarlo y realizar la evaluación de las medidas de desempeño.

1.2.2. Formulación del problema

De acuerdo con lo mencionado anteriormente en el anterior problema que se mencionó, se plantea la siguiente pregunta de investigación:
¿Qué elementos se deben tener en cuenta para construir un método automático que ayude a la clasificación de residuos sólidos por medio de algoritmos de aprendizaje de máquina de clasificación por medio del procesamiento de imágenes?

[8] Canales sectoriales. Interempresas. e-Garbage, un sistema inteligente de gestión de la recogida de residuos {En línea} {16 de octubre de 2021} disponible en: https://www.interempresas.net/Equipamiento_Municipal/Articulos/49606-e-Garbage-un-sistema-inteligente-de-gestion-de-la-recogida-de-residuos.html

1.3 OBJETIVOS

1.3.1. Objetivo general

Implementar un método automático computacional para la clasificación de residuos sólidos en parques a partir de técnicas de procesamiento de imágenes, algoritmos de aprendizaje de máquina y de aprendizaje profundo.

1.3.2. Objetivos específicos

- Construir el dataset de imágenes de residuos sólidos, a partir de repositorios públicos para el entrenamiento de los modelos seleccionados.
- Diseñar el método automático a partir de técnicas seleccionadas de extracción de características de las imágenes y de algoritmos de clasificación para lograr el ajuste de parámetros y sus respectivas medidas de desempeño.
- Desarrollar el método automático a partir del diseño realizado y del entrenamiento de los algoritmos para la clasificación de las basuras.
- Evaluar el rendimiento del método automático a partir de las métricas de desempeño seleccionadas para medir los niveles de acierto y error de este.

1.4. JUSTIFICACIÓN

Este proyecto busca elaborar un método automático para la clasificación de residuos sólidos de parques en Colombia a partir de técnicas de procesamiento de imágenes, algoritmos de aprendizaje de máquina y de aprendizaje profundo, incentivando el uso de la tecnología para la solución de problemas socio ambientales que nos afectan a todos.[9]

Es fundamental entender que si logramos que la tecnología llegue a resolver problemas de esta índole contribuiremos a un mejor futuro. Un ejemplo claro de esto es la pandemia, sin ayuda de los avances tecnológicos en la producción de vacunas probablemente nuestro futuro no sería tan prometedor. En temas ambientales pasa exactamente lo mismo, es más en casi cualquier problema la solución puede estar en la tecnología, claro está que siempre y cuando no haya intereses propios de por medio que dañe las buenas propuestas. Si hablamos del factor humano en el impacto ambiental podemos decir que ha sido el detonante de tantos problemas que vemos hoy en día, sin embargo, también podemos ser la solución con propuestas que aclaran el panorama y ponen una solución a problemas como la clasificación de residuos sólidos, que son un problema de salud pública, como lo expone la organización mundial de la salud. El no separar bien los residuos provoca la propagación de vectores de enfermedades como el paludismo, el dengue, la filariosis linfática, la enfermedad de Chagas, la oncocercosis, la leishmaniosis, la fiebre Chikunguña, enfermedad por el virus de Zika, la fiebre amarilla, la encefalitis japonesa y la esquistosomiasis.[10]

---

[9] SAS Institute Inc, ¿Que es deep learning?{En línea}{10 de mayo 2022} Disponible en:*https://www.sas.com/es_co/insights/analytics/deep-learning.html*

[10] Organización Mundial de la Salud, RESPUESTA MUNDIAL PARA EL CONTROL DE VECTORES 2017–2030{En linea}{10 de agosto 2021} Disponible en:*https://www.who.int/malaria/areas/vector_control/Draft-WHO-GVCR-2017-2030-esp.pdf*

Figura 1 Solid Waste Management in the World's Cities

World municipal solid waste	Kilograms per capita per year	Billion tonnes per year	
		2006	2025
Current estimates	310	2.0	2.4
At simple average for the reference cities	343	2.2	2.7
At average current rate for OECD	580	3.8	4.6
At current OECD maximum rate	760	4.9	5.9

Fuente: Solid Waste Management in the World's Cities : Water and Sanitation in the World's Cities 2010 {En línea} Disponible en: https://unhabitat.org/solid-waste-management-in-the-worlds-cities-water-and-sanitation-in-the-worlds-cities-2010-2

Si se actúa ahora tal vez se pueda dejarle un mejor futuro a las próximas generaciones como vemos en la figura 1, sin embargo, es un problema muy grande el cual se debe descomponer en pequeñas partes para poder resolverlo, cuando se divide este problema se llega al ser humano como individuo el cual no separa bien la basura. Exponiendo esto como una raíz del problema la cual se debe trabajar para que no siga dañando el medio ambiente. El problema de este método es que no es a corto plazo y la basura está por todo lado, si observamos detenidamente los hogares y sitios públicos son lugares donde se genera muchos desechos que son aprovechables pero que no se aprovechan particularmente en los parques, los cuales son lugares muy concurridos y que, si lográramos optimizar este proceso de reciclaje, en estos sitios logramos que muchos residuos sean aprovechados.

¿Por qué en parques? principalmente se actúa en estas zonas ya que son lugares donde por lo general no se recicla adecuadamente, la mayoría de los materiales están con restos de comida y los ciudadanos no hacen la labor de

separarlos, sino que los mezclan y además son desechos que son altamente aprovechable, es por esto por lo que es imprescindible tomar medidas al respecto.

Es imperativo que este proyecto se lleve a cabo para combatir y disminuir el impacto de las basuras en la sociedad. Según la alcaldía de Bogotá, Bogotá genera más de 7200 toneladas basura y tan solo se aprovecha 1200 toneladas de esta basura que equivale aproximadamente a un 16% con esto queremos decir que las basuras son problema para tratar, sin embargo, también la solución a otros muchos ya que utilizando los residuos que pueden ser reutilizados podemos mitigar y desmotivar la utilización de plástico por dar un ejemplo.[11]

1.5. DELIMITACIONES

1.5.1. Limitaciones

En este proyecto se realizó la clasificación de imágenes de residuos como: plástico, vidrio y metal que se encuentran en ambientes como parques de la ciudad, estas imágenes fueron recolectadas de distintos repositorios públicos conocidos para el procesamiento de imágenes, dichas imágenes se encuentran en diferentes posiciones, sin ninguna especificación de iluminación.

1.5.2. Alcance

Se realizará la búsqueda de imágenes en distintos repositorios públicos en la web para armar un conjunto de imágenes y así realizar el entrenamiento de

[11] Bogotá gov, Reciclaje, el primer paso responsable para aprovechar la basura que generamos{En linea}{20 de mayo 20022} Disponible en: https://bogota.gov.co/yo-participo/blogs/basura-en-bogota-una-responsabilidad-de-todos-los-ciudadanos#:~:text=Cada%20persona%20produce%20cerca%20de,mal%20aspecto%20visual%20del%20entorno

distintos modelos, los cuales fueron escogidos debido a su rendimiento en lo encontrado en las distintas fuentes. Posterior a esto se realizará el entrenamiento de tres modelos diferentes con el fin de evaluar y analizar el rendimiento de cada uno de estos y cuáles son los adecuados según la naturaleza del presente proyecto.

2. MARCO DE REFERENCIA

2.1. MARCO CONCEPTUAL

A continuación, se presentan conceptos importantes para la realización de la presente investigación:

2.1.1 Contaminación ambiental

Se denomina contaminación ambiental a la presencia de componentes nocivos (ya sean químicos, físicos o biológicos) en el medio ambiente (entorno natural y artificial), que supongan un perjuicio para los seres vivos que lo habitan, incluyendo a los seres humanos. La contaminación ambiental está originada principalmente por causas derivadas de la actividad humana, como la emisión a la atmósfera de gases de efecto invernadero o la explotación desmedida de los recursos naturales.

Una de las principales consecuencias de la contaminación ambiental es el calentamiento global, también conocido como cambio climático, por el cual la temperatura del planeta va aumentando de manera progresiva, tanto la temperatura atmosférica como la de mares y océanos.

La contaminación ambiental supone un riesgo para la salud de los seres vivos que habitan los ecosistemas contaminados, incluyendo a los seres humanos. Además, la tala indiscriminada, la explotación excesiva de los recursos naturales y la emisión de contaminantes al medio ambiente (gases a la atmósfera, vertidos en medios acuáticos, residuos sólidos) provoca la

destrucción de ecosistemas. De esta forma, muchas especies de animales y plantas ven cómo su hábitat natural se va reduciendo cada vez más, pudiendo llegar a provocar incluso su extinción.[12]

2.1.2. Recolección de residuos sólidos

Según la Alcaldía de la ciudad de Bogotá, la recolección de residuos consiste en la acción y efecto de retirar los residuos sólidos del lugar de la presentación provenientes de: barrido, limpieza, corte de césped, poda de árboles y los residuos seleccionados en la fuente susceptibles de reutilización o reciclaje[13]

2.1.3. Separación en la fuente

La separación en la fuente es la actividad de seleccionar y almacenar los diferentes residuos sólidos en su lugar de origen, para facilitar su posterior manejo y aprovechamiento[14]. Desde el hogar podemos hacer un buen manejo de separación de los residuos, para crear buenas prácticas y

[12]AYUNTAMIENTO DE HUELVA. ¿Qué es la contaminación ambiental? . {En línea} {18 de octubre de 2021} Disponible en: http://www.lineaverdehuelva.com/lv/consejos-ambientales/contaminantes/Que-es-la-contaminacion-ambiental.asp

[13] ALCALDÍA DE BOGOTÁ. Recolección de residuos sólidos. {En línea}{17 de octubre de 2021} Disponible en: https://bogota.gov.co/servicios/guia-de-tramites-y-servicios/recoleccion-de-residuos-solidos

[14]RECIMED COOPERATIVA MULTIACTIVA DE RECICLADORES. Reciclar en la fuente es la opción {En línea}{17 de octubre de 2021} Disponible en: https://reciclaje.com.co/blog/aprende-a-reciclar/separar-en-la-fuente-es-la-opcion/

Figura 2 Separación en la fuente

Fuente: Separar en la fuente es la opción {En línea} Disponible en: https://reciclaje.com.co/blog/aprende-a-reciclar/separar-en-la-fuente-es-la-opcion/

2.1.4. Basura en parques

La basura es un problema que siempre va en aumento, pero ¿cuál es el problema de las basuras en los parques? El problema de tener basura sin separar y regada en los parques es que mucha de esta basura se oxida y hace que se llene de moscas lo que conlleva riesgos sanitarios como: cortes, heridas o enfermedades.[15]

2.1.5. Reciclaje

El reciclaje es el proceso de recolección y transformación de materiales para convertirlos en nuevos productos, y que de otro modo serían desechados como basura. Según Ranjit Baxit, el cual es fundador de Global Recycling Foundation es importante involucrar a estos cambios a toda la población ya sean lideres,

[15]ARTEAGA, FRANCISCO. ¿Qué es un foodcourt? {En línea}{17 de octubre de 2021} Disponible en: https://ingenieriademenu.com/food-court/

empresas o personas. De ahí parte la importancia y beneficios que este presenta al planeta tierra[16]

2.1.6. Tipos de reciclaje

Existen tres tipos principales. El reciclaje primario, o de circuito cerrado, convierte los materiales en más de lo mismo, como papel en más papel o las latas de refresco en más latas de refresco. El secundario transforma un producto desechado en otros objetos, aunque fabricados con el mismo material. Y el terciario o químico descompone químicamente los materiales para producir con ellos algo muy diferente[13]

2.1.7. Plástico

Según la WWF en Colombia, el plástico es un componente compuesto por elementos orgánicos, algunas veces de materiales sintéticos o semi-sintéticos que cuando se le introduce calor y presión, puede ser maleable y que es posible su transformación sin destruir su composición.

Los plásticos son realizados con petroquímicos y elementos como lo son el gas natural, carbón y sal común. Existen también plásticos a base de ácido poli láctico, el cual viene de fuentes renovables, como el almidón de maíz, de yuca o de caña de azúcar[17].

Figura 3 Plástico

[16]área metropolitana ecología urbana. Condiciones de salud y de calidad de vida de la población ligada a los RSU {En línea} {17 de octubre de 2021} Disponible en: https://www.entrerios.gov.ar/ambiente/userfiles/files/archivos/Plan%20Girsu/A%20-%20Area%20de%20Estudio/6%20-%20Condiciones%20de%20salud/Salud.pdf

[17] WWF. Glosario ambiental: ¿Qué es el plástico? {En línea} {18 de octubre}https://www.wwf.org.co/?328912/Glosario-ambiental-Que-es-el-plastico

Fuente: Residuos en el mundo Disponible en: https://www.bbc.com/mundo/noticias-40664725

2.1.8. Vidrio

Material inorgánico duro, frágil y amorfo, según su viscosidad se obtiene aproximadamente a 1500 °C a partir de arena de sílice (SiO2), carbonato de sodio (Na2CO3) y caliza (CaCO3).” (Castells , 2013)[18].

2.1.9. Metal

Según Adalmo, los residuos metálicos se dividen en: férricos y no férricos. Los primeros están compuestos por el hierro, acero y aquellos residuos de las fundiciones de metales.[19]

[18] GOBIERNO DE ESPAÑA, ¿Qué es el vidrio? {En línea} {18 de octubre de 2021} Disponible en: https://www.miteco.gob.es/fr/calidad-y-evaluacion-ambiental/temas/prevencion-y-gestion-residuos/flujos/domesticos/fracciones/vidrio/

[19] ADALMO, Recuperación y clasificación de todo tipo de hierros y metales, {En linea}{12 de mayo de 2022} Disponible en: https://www.adalmo.es/noticias/recuperacion-y-clasificacion-de-todo-tipo-de-hierros-y-metales/#:~:text=Los%20residuos%20met%C3%A1licos%20se%20clasifican,y%20los%20residuos%20de%20fundici%C3%B3n.

2.1.10. Residuos orgánicos

Son aquellos que provienen de residuos biodegradables, que son de fácil descomposición y que son provenientes de materiales de la tierra, es decir que se pueden transformar en materia[20]

2.1.11. Ciencias de la computación

Según el portal de edX, las ciencias de la computación son las disciplinas que estudian la información, la computación y las aplicaciones o ecosistemas en donde estas se desarrollan. Se trata de una ciencia que proviene de las matemáticas aplicadas y tiene 3 ramas que son bastante conocidas. Por un lado, tenemos la ingeniería de software, en segundo lugar, la teoría como tal (estudio primario de algoritmos y datos, etc.) y por último la aplicación de la amalgama de los primeros dos, que terminan resultando en disciplinas como la big data, la ciberseguridad o seguridad informática, el aprendizaje automático, la inteligencia artificial etc.[21]

2.1.12. Inteligencia artificial

Acuñado por primera vez en 1955 en una conferencia del profesor de Matemáticas de la Universidad de Dartmouth, John McCarthy, el concepto de Inteligencia Artificial denota un proceso por el que "hacer que una máquina se comporte de formas que serían llamadas inteligentes si un ser humano hiciera eso"[22]

---

[20] RSU CONSORCIO PROVINCIAL DE RESIDUOS ORGÁNICOS, Residuos orgánicos {En línea} {19 de octubre de 2021} Disponible en: https://www.consorciorsumalaga.com/5936/residuos-organicos

[21] EDX, Ciencias de la computación. {En línea} {18 de octubre de 2021} Disponible en: https://www.edx.org/es/aprende/ciencias-de-la-computacion

[22] SUMUP. Inteligencia Artificial - ¿Qué es la Inteligencia Artificial? {En línea} {18 de octubre de 2021} Disponible en: https://sumup.es/facturas/glosario/inteligencia-artificial/

2.1.12. Aprendizaje de máquina

El principal objetivo del Aprendizaje de Máquina (ML por sus siglas en inglés) es el desarrollo de sistemas que puedan cambiar su comportamiento de manera autónoma basados en su experiencia. El ML ofrece algunas de las técnicas más efectivas para el descubrimiento de conocimiento (patrones) en grandes volúmenes de datos. El ML ha jugado un rol fundamental en áreas tales como la bioinformática, la recuperación de información en la web, la inteligencia de negocios y el desarrollo de vehículos autónomos[23]

2.1.13. Tipos de aprendizaje de máquina

- **Aprendizaje supervisado:** en este tipo se necesitan datos que se encuentren etiquetados, lo cual significa que se le indica al modelo lo que se quiere que aprenda.[24]
- **Aprendizaje no supervisado:** en este tipo, por lo contrario, se trabaja con datos que no están etiquetados, no se tiene algo para predecir, se usan principalmente en tareas donde es necesario analizar los datos para extraer nuevo conocimiento o agrupar entidades por afinidad [21]

2.1.14. Reconocimiento de imágenes

La inteligencia tiene bastantes enfoques, uno de ellos y de los más interesantes es el reconocimiento de imágenes, con el cual se puede clasificar las imágenes de manera eficiente y automatizada para con eso obtener información valiosa y que pueda ser posteriormente usada[25]

---

[23] GONZÁLEZ, FABIO. Aprendizaje de Máquina Maestría en Ingeniería - Ingenieria de Sistemas y Computación. {En línea} {18 de octubre de 2021} Disponible en: https://dis.unal.edu.co/~fgonza/courses/2007-I/ml/index.html

[24]SANCHEZ, ANTONIO. ¿Cómo aprenden las máquinas? Machine Learning y sus diferentes {En línea} {18 de octubre de 2021} Disponible en: tipos https://datos.gob.es/es/blog/como-aprenden-las-máquinas-machine-learning-y-sus-diferentes-tipos

[25] ARSYS. Reconocimiento de imágenes . {En línea} {19 de octubre de 2021} Disponible en: https://www.arsys.es/blog/reconocimiento-imagenes/

- **¿Cómo funciona el reconocimiento de imágenes?**

Para realizar reconocimiento de imágenes es más fácil ya que se pueden crear sistemas que reconozcan imágenes y extraigan información de ellas. Contamos con lenguajes y librerías que tienen su enfoque en aprendizaje de máquina como Python y R. Según el blog publicado por Arsys, el funcionamiento del Machine Learning se basa en ofrecer a la máquina un conjunto, lo más amplio posible, de muestras que pueda usar para el aprendizaje. A partir de sus librerías es posible entrenar el sistema y realizar test de reconocimiento. Una vez puesto a punto el sistema seremos capaces de ejercitarlo con cualquier imagen nueva. Si hemos dispuesto de una buena cantidad de material de base, los algoritmos actuales garantizan muy buenos resultados de acierto [22]

2.1.15. Conjunto de datos de entrenamiento

Según Telefónica Tech, los datos de entrenamiento son los datos que usamos para *entrenar* un modelo. La calidad de nuestro modelo de aprendizaje automático va a ser directamente proporcional a la calidad de los datos. [26]

2.1.16. Medidas de desempeño

Son buenas formas de evaluar los modelos de clasificación para conjuntos de datos equilibrados, pero si los datos están desequilibrados y hay una disparidad de clases[27] .

[26]TELEFÓNICA TECH. Datos de entrenamiento vs datos de test. {En línea} {19 de octubre de 2021} Disponible en: https://empresas.blogthinkbig.com/datos-entrenamiento-vs-datos-de-test/
[27] NAGESH, SINGH. Métricas De Evaluación De Modelos En El Aprendizaje Automático https://www.datasource.ai/es/data-science-articles/metricas-de-evaluacion-de-modelos-en-el-aprendizaje-automatico

2.2. MARCO TEÓRICO

2.2.1. Teoría de aprendizaje de máquina

El Aprendizaje de Máquina, juega un papel importante dentro del desarrollo, como lo son minería de datos, procesador de idioma natural, reconocimiento de imagen y sistemas de expertos. Este provee posibles soluciones en todos estos dominios y más. Además, está creada para ser un pilar de nuestra civilización futura.[28]

2.2.2. Preprocesamiento

El preprocesamiento de datos es una etapa esencial del proceso de descubrimiento de información. Esta etapa se encarga de la limpieza de datos, su integración, transformación y reducción para la siguiente fase de minería de datos donde los datos empiezan a convertirse en información realmente relevante para los proyectos.[29]

2.2.3. Tipos de muestreo

Él es un conjunto de técnicas estadísticas que implican un trabajo de análisis que posteriormente podremos concluir con información extraída en forma de conclusiones, para este proceso existen diferentes técnicas sin embargo se dividen en dos categorías principales Muestreo aleatorio o probabilístico.[30]

---

[28] Nick McCrea. Aprendizaje de Máquina y sus Aplicaciones: Un Tutorial Visual con Ejemplos {En línea} {16 de octubre de 2021} disponible en:https://www.toptal.com/machine-learning/introduccion-a-la-teoria-de-aprendizaje-de-maquina-y-sus-aplicaciones-un-tutorial-visual-con-ejemplos

[29] Salvador García, Sergio Ramírez-Gallego, Julián Luengo, Francisco Herrera. Big Data: Preprocesamiento y calidad de datos{En línea} {16 de septiembre de 2021} disponible en:https://sci2s.ugr.es/sites/default/files/ficherosPublicaciones/2133_Nv237-Digital-sramirez.pdf

[30]UNIR. Tipos de muestreo: los principales y sus características {En línea} {16 de septiembre de 2021} disponible en:https://www.unir.net/ingenieria/revista/tipos-de-muestreo/

2.2.4. Extracción de características

La extracción de características (features) se trata de un proceso de reducción y codificación, donde un conjunto inicial de variables sin procesar (ej. texto en un documento) se reduce a características más manejables para su procesamiento (ej. números) y que se describa con precisión el conjunto de datos original.[31]

2.2.5. Técnicas de extracción de características:

Existen diversas técnicas de extracción de características y básicamente se clasifican en dos tipos clásicas y basadas en Deep Learning [30]

2.2.6. Anaconda

Anaconda es una de las suites mas complejas para la ciencia de datos utilizando el lenguaje de programación en Python la computación numérica que ayuda a crear un entorno virtual para el proyecto en específico contiene las librerías más usadas por los científicos de datos. Además, hace más fácil la instalación de librerías, framework que el proyecto amerite. Anaconda ofrece la posibilidad de crear varios proyectos cada uno con entornos virtuales diferentes, diferenciando hasta las versiones de Python, esto ayuda a que se puedan realizar diversas pruebas con entornos totalmente diferentes.

2.2.7. Spyder

Spyder es un marco de licencia gratuita, brindado por Anaconda. Spyder maneja características únicas a los demás entornos de programación en Python, tales como espacios para visualización de las variables del ambiente de trabajo durante la interpretación del código, esto convierte a Spyder como una opción más que viable para los programadores que ameriten estas características.

[31] ITELLIGENT. Machine Learning & NPL: cómo funciona un clasificador de documentos {En línea} {16 de septiembre de 2021} disponible en:https://itelligent.es/es/pln-clasificacion-automatica-documentos/

2.2.8. Python

Python es un lenguaje de programación de tipo interpretado, muy popular entre los programadores y sobre todo en labores relacionadas con datos por su capacidad de integración con frameworks y librerías tales como Tensor Flow, matplotlib, Basemap y Seaborn. Esto sumado a su alta eficiencia hace que sea un lenguaje de programación clave para el aprendizaje de máquina.

2.2.9. TensorFlow

Es una librería de Python que permite tener acceso a algoritmos, modelos y demás métodos numéricos para el análisis de machine learning y procesamiento de lenguaje natural.

2.2.10. Librerías o frameworks de Python para cálculos numéricos y Análisis de datos

En el aprendizaje de máquina existen varias fases para su desarrollo un de ellas es el cálculo numérico el cual consume bastante tiempo acompañado de la preparación de los datos, relevantes para el proyecto para acompañar estos procesos existen librerías como Numpy y Pandas usados para proyectos de aprendizaje de maquina.

2.2.11. Numpy

Numpy es una biblioteca de Python numérica que proporciona estructuras de datos alfanuméricos de muy buena calidad. Numpy es la librería más utilizada dada su facilidad de uso y rapidez de cálculos dado que gran parte de esta librería está hecha en el lenguaje C.[32]

[32] MARTINEZ, José, “15 Librerías de Python para Machine Learning” {En Línea} {29 de octubre 2021} Disponible en (https://www.iartificial.net/librerias-de-python-para-Machine-learning/) p.40

2.2.12. Redes neuronales convoluciones

Es un tipo de red neuronal que cuenta con aprendizaje supervisado y la cual utiliza la corteza visual del ojo con el fin de identificar respectivas características en lo que visualiza para que se puedan identificar distintos objetos.

La red utiliza capas en las cuales se realiza el procesamiento en las cuales encuentra curvas y líneas hasta que se va mejorando distintos tipos de figuras.[33]

2.2.13. RELU

Es una función que tiene como objetivo entregar una salida a partir de un valor de entrada, estos valores se encuentran entre o y 1 o -1 y 1. [34]

2.2.14. VGG-16:

VGG-16 fue propuesto por Karen Simonyan y Andrew Zisserman del Laboratorio del Grupo de Geometría Visual de la Universidad de Oxford en 2014 en el documento "REDES CONVOLUCIONALES MUY PROFUNDAS PARA EL RECONOCIMIENTO DE IMÁGENES A GRAN ESCALA". Este modelo ganó el 1er y 2do lugar en las categorías anteriores en el desafío ILSVRC 2014. [35]

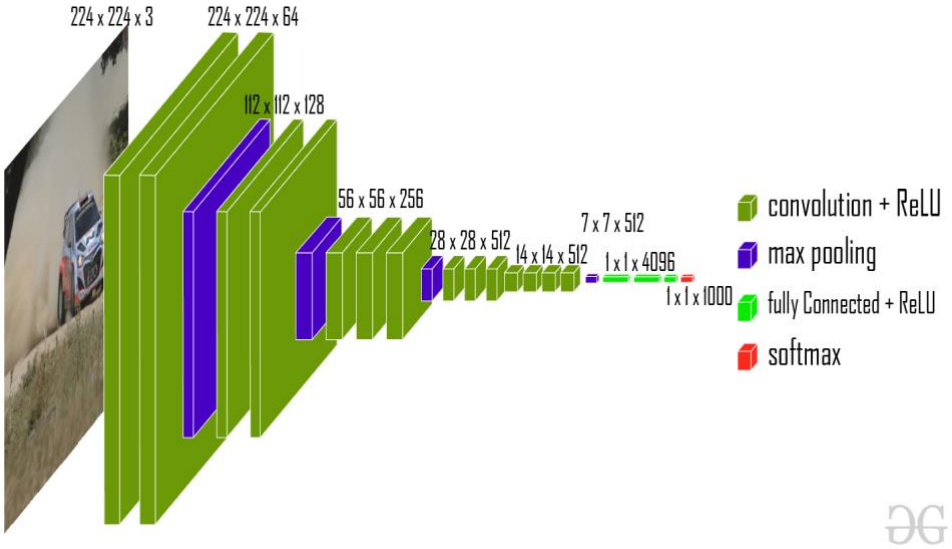
---

[33] Aprendemachinelearning. Convolutional neural network. {En linea} {12 de mayo de 2021} Disponible: https://www.aprendemachinelearning.com/como-funcionan-las-convolutional-neural-networks-vision-por-ordenador/

[34] Diego Calvo. Función de activación – Redes neuronales {En linea} {12 de mayo de 2021} Disponible en: https://www.diegocalvo.es/funcion-de-activacion-redes-neuronales/

[35] Lima Acervo, "VGG-16 | Modelo de CNN"{En Línea} {29 de octubre 2021} Disponible en (https://es.acervolima.com/2021/02/09/vgg-16-modelo-de-cnn/)

Figura 4 VGG-16



Fuente: VGG-16 | MODELO DE CNN

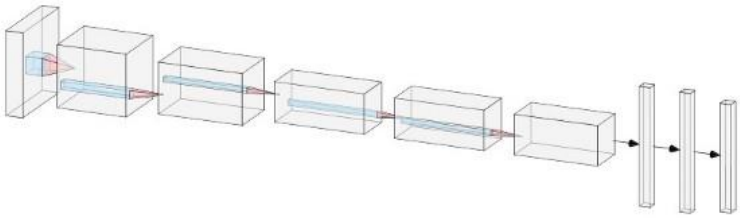
https://es.acervolima.com/2021/02/09/vgg-16-modelo-de-cnn/

Este es un modelo que tiene una precisión de prueba del 92,7% entre los primeros datos de Imagen Net con un data set bastante amplio de aproximadamente 14 millones de imágenes.

2.2.15. AlexNet:

Es una red neuronal convoluciones que fue construida con el fin de analizar conjuntos de datos más grandes de lo normal, aprender de modelos más potentes y utilizar mejores técnicas, para evitar sobre ajustes.

Figura 5 AlexNet



Fuente: ¿Cómo funcionan las Convolutional Neural Networks? Visión por Ordenador https://ichi.pro/es/alexnet-la-cnn-que-cambio-la-vision-por-computadora-169459181425709

AlexNet fue utilizado por Alex Krizhevsky, Ilya Sutskever y Geoffrey Hinton en la competencia ILSVRC 2012, bajo el nombre de equipo de SuperVision (detalles de los resultados aquí). Las imágenes dadas se redimensionan a una resolución fija de 256x256, y luego se restó la media del conjunto de datos de cada una de las imágenes.[36]

2.2.16. KNN:

El K-NN es un algoritmo de aprendizaje supervisado, es decir, que a partir de un juego de datos inicial su objetivo será el de clasificar correctamente todas las instancias nuevas. El juego de datos típico de este tipo de algoritmos está

[36] Alexander Lenail, "AlexNet: la CNN que cambió la visión por computadora" {En Línea} {29 de octubre 2021} Disponible en https://ichi.pro/es/alexnet-la-cnn-que-cambio-la-vision-por-computadora-169459181425709)

formado por varios atributos descriptivos y un solo atributo objetivo (también llamado clase).[37]

Figura 6 KNN

$$D_{m,n} = \begin{bmatrix} d_1 = & a_{1,1} & \dots & a_{1,n} & c_1 \\ d_2 = & a_{2,1} & \ddots & a_{2,n} & c_2 \\ \vdots & \vdots & \vdots & \vdots & \\ d_m = & a_{m,1} & \dots & a_{m,n} & c_m \end{bmatrix}$$

Fuente: El algoritmo KNN y su importancia en el modelado de datos {En linea} Disponible en: https://www.merkleinc.com/es/es/blog/algoritmo-knn-modelado-datos

2.1.17. Aprendizaje de máquina:

Una matriz de confusión, también llamada matriz de error, es una tabla que evalúa el rendimiento de un modelo de clasificación. [38]

2.2.18. Recall:

A diferencia de la precisión esta es una medida que determina el porcentaje de falsos positivos que hay en el modelo de aprendizaje de máquina.[39]

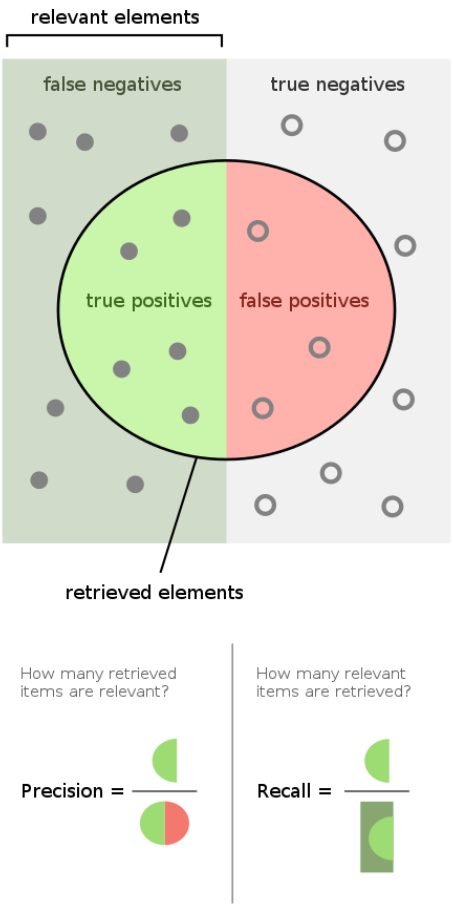
---

[37] Merkle logo, El algoritmo K-NN y su importancia en el modelado de datos {En Línea} {29 de octubre 2021} Disponible en:https://www.merkleinc.com/es/es/blog/algoritmo-knn-modelado-datos

[38] Shin Terence Comprensión de la matriz de confusión y como implementarla en Python,{En linea}{15 de noviembre 2021} disponible en:Comprensión de la Matriz de Confusión y Cómo Implementarla en Python (datasource.ai)

[39] Jose Martinez heras,"Precision, Recall, F1, Accuracy en clasificación"{En Línea}{10 de mayo 2022}

Figura 7 Recall and precisión



Fuente: Precision and recall
https://commons.wikimedia.org/wiki/File:Precisionrecall.svg#globalusage

2.2.19. MaxPooling:

Max Pooling es un método de muestro y el objetivo es disminuir las muestras de una entrada (imágenes, matriz y capas ocultas)[40]

[40] QuoraWhat is max pooling in convolutional neural networks {En Línea} {29 de octubre 2021} Disponible en: https://www.quora.com/What-is-max-pooling-in-convolutional-neural-networks

2.3. ESTADO DEL ARTE

El aprendizaje automático ha sido un campo que ha tomado mucha fuerza en últimos tiempos y que ha tomado cada vez más fuerza. Muchos países como lo son Suiza, Holanda, España etc., han ido desarrollando sistemas y técnicas eficientes para el reciclaje de residuos además de un nivel más alto culturalmente hablando y las buenas prácticas que poseen en cuanto a la clasificación de dichos residuos, en el día a día.

Los autores Bernardo S. Costa, Aiko CS Bernardes, Julia VA Pereira, Vitoria H. Zampa, Vitoria A. Pereira, Guilherme F. Matos, Eduardo A. Soares, Claiton L. Soares, Alexandre F. Silva del Departamento de Computación - Instituto Federal del Sur de Minas Gerais (IFSuldeMinas) realizaron una investigación respectiva a la inteligencia artificial en la clasificación automatizada en el reciclaje de basura en cual utilizan una visión de computadora que clasifique la basura en categorías de reciclar, el cual optimiza de una manera muy eficiente la forma en que se realiza el tratamiento de los residuos, para ello utilizaron una base de datos de imágenes de basura que contenía aproximadamente unas 400 imágenes de distintas clases. Se utilizaron los modelos VGG-16, AlexNet, Support Vector Machine (SVM), K-Nearest Neighbor (KNN) y Random Forest (RF). Los experimentos que ellos realizaron les arrojaron una precisión de aproximadamente del 93%[41].

[41] FIENO, ALEXANDRE. SOARES, EDUARDO. Artificial Intelligence in Automated Sorting in Trash Recycling {En Línea} {29 de octubre 2021} Disponible en: https://www.researchgate.net/publication/326994757_Artificial_Intelligence_in_Automated_Sorting_in_Trash_Recycling

Hablando en la línea de detección de imágenes, los autores Chofugaoka, Chofu-shi, Tokio Japón han realizado una investigación respectiva a detección de imágenes en comidas por medio de un sistema automático, en el cual usan Multiple Kernel Learning (MKL) con el fin de integrar distintos tipos de características en la imagen como lo es el color, la textura y SIFT de forma adaptativa. MKL permite estimar pesos óptimos para combinar las características de imagen para sus categorías[42]

En el 2018, los autores Rahul Chauhan, Kamal Kumar Ghanshala, RC Joshi realizaron un proyecto de investigación de Redes Neuronales Convolucionales (CNN) para detección y reconocimiento de imágenes. Nos mencionan en su artículo que las redes neuronales convoluciones son algoritmos de aprendizaje profundo que pueden entrenar grandes conjuntos de datos con bastantes parámetros, en forma de imágenes 2D como entrada y convolucionadas con filtros para producir los resultados esperados. En el artículo usan modelos de redes neuronales que sirven para evaluar el desempeño en conjuntos de datos de detección y reconocimiento de imágenes. Se utilizaron los algoritmos de MNIST y CIFAR-10 y se evalúa su rendimiento, los cuales dieron una precisión del 99,6%[43].

En el presente artículo de investigación de maestría del autor Mandar Satvilkar, del año 2018 titulada “Clasificación de basura basada en imágenes mediante algoritmos de aprendizaje automático para el estado de reciclabilidad” el cual tiene como objetivo cerrar la brecha existente entre la clasificación de imágenes

[42] YANAI, KEIJI. A food image recognition system with Multiple Kernel Learning. {En Línea} {29 de octubre 2021} Disponible en: https://www.researchgate.net/publication/326994757_Artificial_Intelligence_in_Automated_Sorting_in_Trash_Recycling

[43] Rahul Chauhan; Kamal Kumar Ghanshala; R.C Joshi. Convolutional Neural Network (CNN) for Image Detection and Recognition. {En Línea} {30 de octubre 2021} Disponible en: https://ieeexplore.ieee.org/abstract/document/8703316

y la clasificación de basura y garantizar mediante aprendizaje automático que se detecte el tipo de material correcto. En este caso utilizan analítica para realizar la clasificación de las imágenes en cada tipo, además se tiene el objetivo de tomar fotografías, segregar dichos elementos de las fotografías y clasificarlas en: cartón, vidrio, metal, papel y plástico. Utilizaron un conjunto de datos de 400 a 500 imágenes de las clases mencionadas, a las cuales acceden de manera gratuita a través de GitHub. El trabajo se realizó con Redes Neuronales Convoluciones (CNN) ya que tienen bastantes capacidades en cuanto al aprendizaje de máquina y aprendizaje automático en el campo de la clasificación de imágenes[44]

Los autores Olugboja Adedeji, Zenghui Wang en la realización de la 2da Conferencia sobre Procesamiento y Fabricación de Materiales Sostenibles en el año 2019 en su artículo llamado “Intelligent Waste Classification System Using Deep Learning” en el cual plantean un proceso para la separación de residuos avanzado por medio de un sistema que los gestione de manera eficiente, dicho sistema lo desarrollaron utilizando ResNet-50 la cual es un modelo de red neuronal convolucionar, por medio de aprendizaje automático que sirve como extractor y máquina de soporte vectorial (SVM) que lo utilizaron para clasificar los desechos en distintos grupos. Utilizaron un conjunto de datos de imágenes de basura que fue desarrollado por Gary Thung y Mindy Yang que en su ejecución logró una métrica de precisión de 87%, lo cual demostró que el sistema es más eficaz que la recolección manual[45].

---

[44] SATVILKAR, MANDAR. Image Based Trash Classification using Machine Learning Algorithms for Recyclability Status. {En Línea} {29 de octubre 2021} Disponible en: http://norma.ncirl.ie/3422/

[45]ADEDEJI, OLUGBOJA. WANG, ZENGHUI.Intelligent Waste Classification System Using Deep Learning Convolutional Neural Network. {En Línea} {30 de octubre 2021} Disponible en: https://www.sciencedirect.com/science/article/pii/S2351978919307231

En la escuela de computación del Colegio Nacional de Irlanda, en la investigación realizada por el estudiante Himanshu Gupta en el año 2020 titulada como “Trash Image Classification System using Machine Learning and Deep Learning” traducido en español como “Sistema de clasificación de imágenes de basura usando Aprendizaje automático y aprendizaje profundo” en el cual nos menciona que en muchas de las ciudades del mundo, con el creciente aumento de la tecnología se ha dejado de lado la correcta separación de residuos y que ha ido afectado la salud de los seres humanos y del medio ambiente. El objetivo de dicho estudio fue por medio de un sistema que se clasificaron dichas imágenes en las categorías con ayuda de las técnicas correspondientes de aprendizaje automático y profundo, además de ello para la ejecución de su proyecto tomaron el data set de TACO en el cual hay 1500 imágenes de basura que se encuentran ya etiquetadas de manera correspondiente. Los algoritmos que utilizaron para el desarrollo del sistema fueron: Keras, Resnet-50 y VGG-19 y el modo XGBoost con el fin de extraer características[46].

Por otra parte, también enfocada a la detección de imágenes, Faisal Y. Alzyyou, Waleed Maqableh y Faiz Al Shrouf en su artículo de investigación titulado “Un enfoque adaptativo semi inteligente para la clasificación de basura” proponen un enfoque con algo que llama “Semi Smart Trash Separator” con el fin de separar la basura utilizando una técnica que reciclado el cual contiene un código QR según el material el cual haga que se realice la separación de los residuos según el código que se encuentre en él, como vemos, esto es un enfoque más enfocado al hardware ya que contiene un separador magnético que hace que los residuos se clasifiquen según su dureza, si en dicho proceso hay mucha ambigüedad utilizan redes neuronales AlexNet y GoogLeNet, en sus resultados obtuvieron

[46]GUPTA, HIMANSHU. Trash Image Classification System using Machine Learning and Deep Learning Algorithms. {En Línea} {30 de octubre 2021} Disponible en: http://norma.ncirl.ie/4396/

que la tasa de precisión de reconocimiento del tipo de residuo con AlexNet y GoogLeNet es del 75% y del 83%[47].

Como se evidencia, se tiene en común que, en la mayoría de los artículos presentados, se utilizan algoritmos de reconocimiento de imágenes, independientemente del enfoque que se les dé, es importante seleccionar aquellas técnicas que arrojen unas mejores medidas de desempeño.

También existe otro enfoque, en el cual se utiliza hardware, en este artículo de Miko Pamintuan, Shiela Mae Mantequilla, Hillary Reyes, Mary Jane Samonte nos muestran que desarrollan un contenedor de basura con sensores que separa de manera inteligente los residuos que se gestionan en el utilizando aprendizaje de máquina, en el entrenaron más de 2000 muestras de tipo residuo biodegradable y no biodegradable. Los resultados mostraron que se puede clasificar la basura de manera eficiente. [48]

"A Novel Framework for Trash Classification Using Deep Transfer Learning" en el presente articulado realizado por los investigadores Anh H. Vo; Le Hoang Son; Minh Thanh Vo, Tuong Lee, se enfoca en la manera en que la sociedad y población ha crecido y con eso también el aumento de los residuos que se producen causando un impacto negativo, con ello plantean la elaboración de un sistema que clasifica la basura por medio de visión de computadora e integración de sensores. En el estudio proponen un modelo con redes neuronales profundas que clasifique basura y para ello primero recopilaron el dataset de VN-trash el cual tiene 5904 imágenes de diferentes clases como son orgánicos, inorgánicos

[47] Alzyoud, Faisal; Maqableh, Waleed; Faiz Al Shrouf. A Semi Smart Adaptive Approach for Trash Classification {En Línea} {30 de octubre 2021} Disponible en: https://www.proquest.com/openview/98c370e8f0f1e762a54e72c4ba32a150/1?pq-origsite=gscholar&cbl=5045567

[48]Miko Pamintuan; Shiela Mae Mantiquilla; Hillary Reyes; Mary Jane Samonte. i-BIN: An Intelligent Trash Bin for Automatic Waste Segregation and Monitoring System. {En Línea} {30 de octubre 2021} Disponible en: https://ieeexplore.ieee.org/abstract/document/9072787

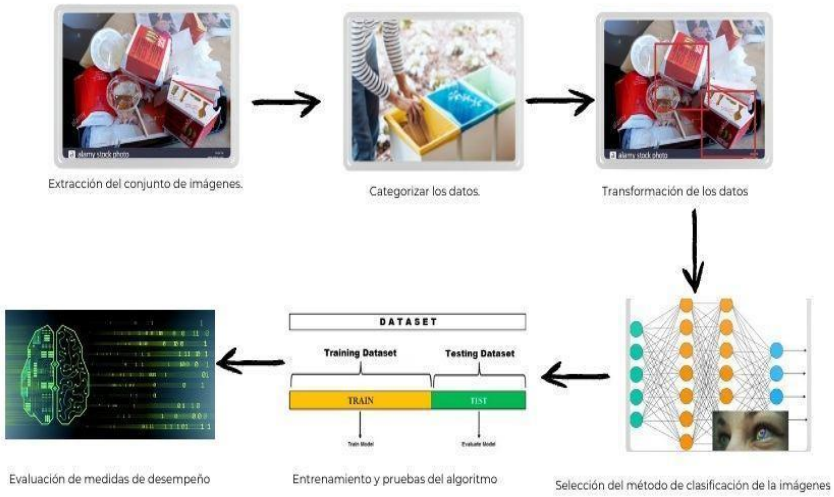
y médicos. En la red neuronal profunda utilizaron DNN-TC el cual rinde un 94% y 98% en términos de precisión
para los conjuntos de datos Trashnet y VN-trash respectivamente y, por lo tanto, supera los métodos de vanguardia para basura clasificación en ambos conjuntos de datos experimentales[49].

Con los artículos previamente mencionados, ampliamos nuestro panorama del conocimiento en el área y nos podemos basar para ver cuales algoritmos han sido los que cuentan con medidas de desempeño más precisas

[49]Anh H. Vo; Le Hoang Son; Minh Thanh Vo, Tuong Lee. A Novel Framework for Trash Classification Using Deep Transfer Learning.{En Línea} {30 de octubre 2021} Disponible en: https://ieeexplore.ieee.org/abstract/document/8930948

3. METODOLOGÍA

Figura 8 Metodología del proyecto



Fuente: Los autores

Para la planeación e implementación del presente trabajo de investigación, del método automático computacional, es necesario conocer cuál es el proceso para llevar a cabo cada una de las fases que permitirán que se cumplan satisfactoriamente los objetivos que se plantean en el proyecto, los cuales están basados en etapas del aprendizaje de máquina combinados con estrategias de ejecución de proyectos, también basados en otros trabajos que han sido elaborados recientemente. Se presentan las siguientes fases o etapas con su respectiva explicación:

3.1. Obtención del dataset

Previamente a la realización de la clasificación de las imágenes de basuras de distintos tipos, es importante realizar una búsqueda de datasets que aporten a la ejecución de la solución planteada, que tengan cantidad de imágenes suficientes

para que no exista un sobre overfitting[50] lo cual hace que la máquina entrena solo entienda ciertos casos que se le enseñan y no nuevos datos de entrada.

3.2. Categorización de los datos

En la presente fase se extraen las características de un conjunto de variables sin procesar y se reduce a características que sean más manejables para su futuro procesamiento y se describa con el conjunto de datos inicial[51].

3.3. Transformación de los datos

Este proceso se realiza posteriormente a la limpieza de los datos, luego de aplicar distintas técnicas de limpieza con un dataset que ya esté validado, estructurado y sin problemas, es importante utilizar herramientas de transformación de datos con el fin de mejorar el aprendizaje y también que se lleve a cabo[52].

3.4. Selección del método de clasificación de las imágenes

A partir del análisis de distintos métodos y de las pruebas de desempeño de los algoritmos se escoge el método de clasificación más acorde según la naturaleza de los datos del conjunto de datos seleccionado.

---

[50] NA8. Qué es overfitting y underfitting y cómo solucionarlo..{En Línea} {1 Noviembre 2021} Disponible en: https://www.aprendemachinelearning.com/que-es-overfitting-y-underfitting-y-como-solucionarlo/#:~:text=Overfitting%20en%20Machine%20Learning,reconocer%20nuevos%20datos%20de%20entrada.

[51]Intelligent, Machine Learning & NPL: cómo funciona un clasificador de documentos{En linea}{2 de noviembre 2021} disponible en: https://itelligent.es/es/pln-clasificacion-automatica-documentos/

[52] XERIDIA. Transformación de datos en Inteligencia Artificial. {En línea} {2 de noviembre de 2021} Disponible en: https://www.xeridia.com/blog/transformacion-de-datos-en-inteligencia-artificial

3.5. Entrenamiento y pruebas de algoritmos

Es importante dividir los datos que se van a trabajar en dos conjuntos, uno de entrenamiento y otro de pruebas con el fin de obtener un modelo de datos de entrenamiento y posteriormente verificar la eficiencia con los datos que se guardaron para test.

En aprendizaje de máquina, tiene gran impacto la manera en que se tratan los datos de los cuales han sido generados, es importante que se cuiden los datos y el equilibrio entre variables.

3.6. Evaluación de medidas de desempeño

En esta fase se realiza el análisis comparativo con las medidas de desempeño.

3.7. Diseño del método automático computacional

A continuación, se debe realizar el diseño del método automático según la naturaleza de las variables del dataset escogido para la problemática a tratar y con base en los artículos e investigaciones encontradas recientemente.

3.8. Rediseñar y afinar el método

En esta fase mejoramos según las retroalimentaciones recibidas del proyecto, teniendo en cuenta cada una de las fases anteriores y donde se puedan presentar posibles errores en algún proceso, debido a que si son encontrados de manera previa no cuestan tanto como si se encuentran en una fase más avanzada del proyecto.

4. DISEÑO METODOLÓGICO

4.1. CONSTRUCCIÓN DEL DATASET

En el presente capítulo se mostrará cada uno de los pasos que se tuvieron en cuenta a la hora de la elaboración del dataset, el cual es uno de los insumos principales para el proyecto con el cual se realizó el entrenamiento del algoritmo

4.1.1. Búsqueda de fuentes

Para empezar la elaboración del dataset se tuvo en cuenta ciertas fuentes que se encontraban en la web. Existen variedad de páginas web o repositorios en los cuales puedes encontrar conjuntos de imágenes de diferentes tipos y clases.

Al realizar la búsqueda se encontró información de datasets de imágenes de residuos de distintos tipos, entre ellos:

4.1.1.1. Trashnet

Dataset de imágenes alojado en un repositorio alojado en Github, este repositorio contiene imágenes de 6 tipos de clases de residuos: vidrio, papel, cartón, plástico, metal y basura. Este dataset fue elaborado por Gary Thung y Mindy Yang de la Universidad de Stanford.

4.1.1.2. TACO

Es un dataset de imágenes de residuos en la naturaleza, en él se encuentran fotografías tomadas en distintos ambientes desde parques hasta en calles de ciudades además de que se encuentran divididas las imágenes super categoría y categorías.

4.1.1.3. Detect-waste

Es un dataset que se encuentra en Github, en este se encuentran diversas imágenes de residuos en una oficina, dichos residuos pueden ser plástico, papel, cartón y basura. El creador de este dataset aparece con nombre de usuario en Github como: wimlds-trojmiasto.

4.1.1.4. Trashbox

Es un dataset de imágenes con objetos de basura para la clasificación de residuos, este se encuentra en un repositorio de Github y fue elaborado por Nikhil Venkat Kumsetty.

4.1.2. Selección de las imágenes

Se escogió finalmente las imágenes de los datasets: Trashnet y TACO puesto que estos se encuentran con imágenes asociadas a entornos como lo son parques.

Para la selección de cada una de las imágenes, primero se realizó la descarga de la carpeta de imágenes del repositorio de Trashnet, la cual venía dividida por varias carpetas con los diferentes tipos de residuos, en este caso se tomó la carpeta de vidrio, plástico y metal y de ahí se tomaron las imágenes que eran acordes y se guardaron en la carpeta en el equipo local, todas nombradas con el nombre del material y números consecutivos.

4.1.3. Etiquetado de las imágenes seleccionadas

En la clasificación de imágenes usando aprendizaje de máquina es importante que antes del entrenamiento del algoritmo, las imágenes seleccionadas se encuentren etiquetadas y que estén etiquetadas de manera correcta con el fin de lograr una buena precisión.

El etiquetar imágenes significa seleccionar el objeto a detectar por medio de un rectángulo con un software especial, en este caso se utilizó *LabelImg;* el cual nos permite realizar el etiquetado de grandes grupos de imágenes.

Esta herramienta de etiquetado de imágenes fue elaborada con Python y para mostrar la interfaz gráfica usa el framework de *Qt* que se encuentra, descargarla y empezar a usarla resulta fácil puesto que es una herramienta gratuita.

https://blog.roboflow.com/labelimg/

Para descargar esta herramienta, se debe hacer desde el repositorio de GitHub en el siguiente enlace: https://github.com/tzutalin/labelImg**,** en este repositorio nos indican los pasos a seguir para la instalación según el sistema operativo con el que se encuentre el equipo en el cual se va a realizar el etiquetado.

Una vez que se tuvo instalado *LabelImg* en la máquina local, se procedió a realizar la configuración para el etiquetado de las imágenes ya previamente seleccionadas y almacenadas en su carpeta correspondiente según su tipo: *plástico, metal y vidrio.*

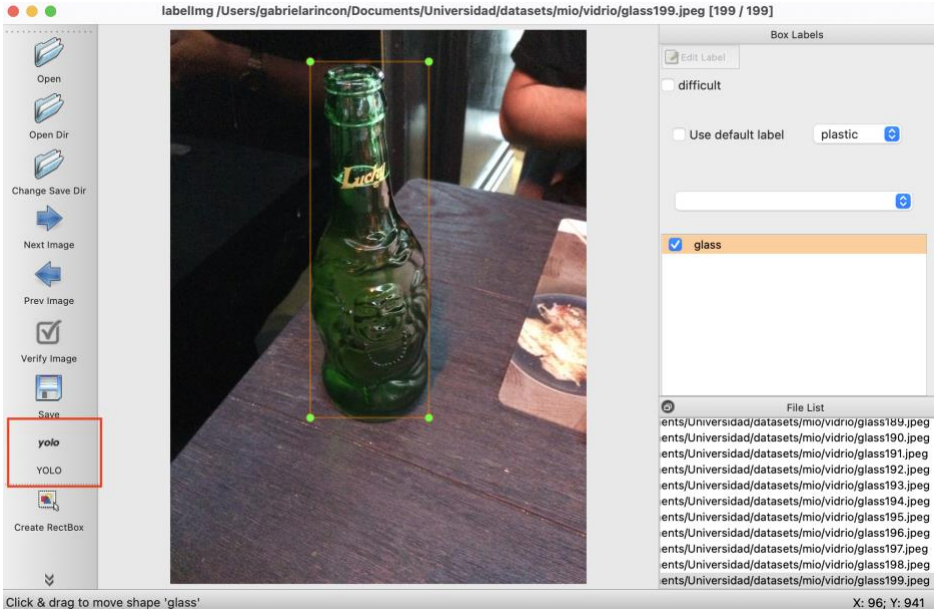
En la carpeta en la cual se almacenó la herramienta, debemos ingresar y cambiar el archivo llamado *predefined_classes.txt* el cual se encuentra dentro de la carpeta data, en este archivo deben encontrarse las clases que se clasificaron con el algoritmo que se escogió, divididas por salto de línea, de la siguiente manera:

Figura 9 Archivo de clases predefinida

Fuente: Los autores

Previamente a la etiquetación de imágenes, se debe seleccionar la carpeta base donde están las imágenes y la carpeta en la que queremos que se almacenen los archivos de las imágenes etiquetadas, en este caso se realizó la configuración para que se almacenarán las imágenes etiquetadas en archivos con formato .txt este formato se llama YOLO, el cual es un método de detección de objetos con una red neuronal. En la figura 11 podemos encontrar la configuración de Labelimg.

Figura 10 LabelImg configuración YOLO



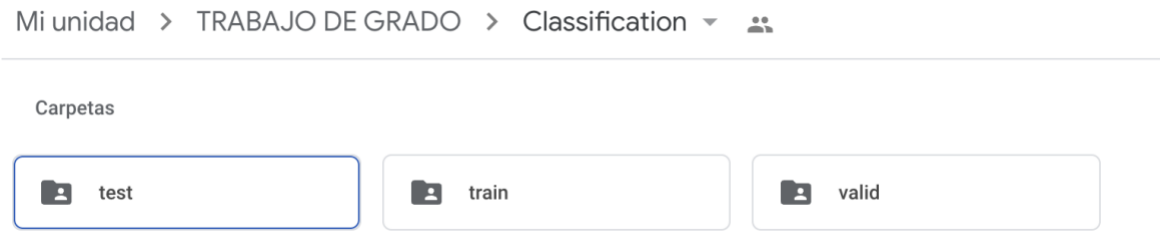
Fuente: Los autores

En la carpeta con los archivos .txt encontraremos las imágenes etiquetadas en un formato, el cual indica en una columna el número de la clase y en las demás las coordenadas en la imagen en donde se realizó el cuadrado o rectángulo de la etiqueta.

4.1.4. Organización del conjunto de datos en Google Drive

En el caso de los algoritmos no supervisados, no se necesitaron las imágenes etiquetadas, por ende, lo que se realizó fue la organización de las imágenes del dataset en carpetas en Google Drive, con el fin de utilizarlas cuando se realice el entrenamiento en pasos posteriores en Google Colab Pro. La organización de las carpetas se puede evidenciar en la figura 11.

Figura 11 Organización de conjunto de datos en Google Drive



Fuente: Los autores

Se creó una carpeta llamada “TRABAJO DE GRADO”, dentro de dicha carpeta se encontraba una carpeta llamada “Classification” y a su vez dentro de estas tres carpetas llamadas “test”, “train” y “valid” las cuales fueron utilizadas para el entrenamiento, pruebas y validación de los modelos entrenados.

Dentro de las características del conjunto de imágenes armado, se encontró que dichas imágenes no tenían una iluminación específica, las imágenes también se encontraban con ruido de fondo, las imágenes se encontraban por partes, es decir, podía mostrarse solo una parte de dicho residuo, esto se realizó de esta manera con el fin de ver el comportamiento de los modelos ante todas estas características mencionadas.

Las imágenes quedaron a color, pues la naturaleza de estos modelos no exige que haya un cambio en el color de las imágenes para su procesamiento. En cuanto a tamaños de las imágenes fue indiferente, pues las redes neuronales convolucionales de los modelos podían modificar el tamaño de entrada de las imágenes al ingresar en ellos.

4.2. DISEÑO DEL MÉTODO AUTOMÁTICO

4.2.1. Búsqueda de algoritmos

A continuación, en esta etapa se realiza la búsqueda y selección de algoritmos, la cual se realiza dentro distintos tipos de literaturas conocidas en el campo. Los algoritmos fueron:

4.2.1.1. VGG16

Esta es una red neuronal convolucional que fue presentada en la Universidad de Oxford por las personas K Simonyan y A. Zisserma, la cual ganó un gran reconocimiento en el año 2014 por el premio en la competencia llamada "ILSVRC ImageNet" en este concurso alcanzó un accuracy o precisión de 92,7%. Imagenet es un gran banco de imágenes con más de 14 millones de imágenes etiquetadas de distintas clases. [53]

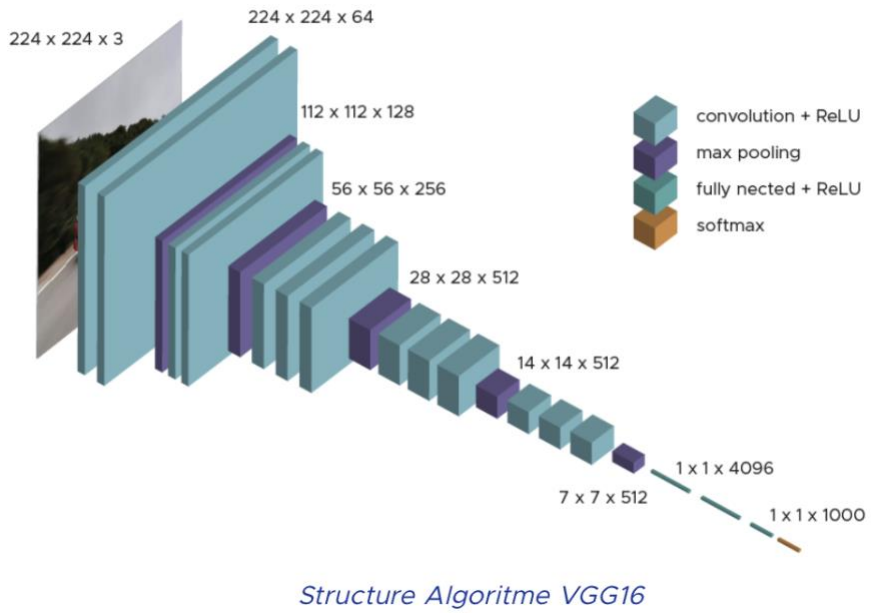
Esta red neuronal tiene una gran mejora y es que posee núcleos más pequeños, de 3x3 en sus capas, estos filtros compensan otros campos receptivos que traían otros tipos de redes neuronales, lo cual permite que el área receptiva sea de mayor tamaño; el área receptiva o campo receptivo es el espacio en que la neurona responde ante estímulos que se le dan. [54]

---

[53] Great Learning Team. Introduction to VGG16 | What is VGG16? {En línea} {10 de mayo de 2022} Disponible en:https://www.mygreatlearning.com/blog/introduction-to-vgg16/#VGG%2016%20Architecture

[54]Upf Edu. Campo receptivo {En línea} {10 de mayo de 2022} https://www.upf.edu/es/web/focus/noticies/-/asset_publisher/gOocsyZZDGHL/content/id/237800839/maximized#:~:text=Se%20denomina%20campo%20receptivo%20(RF,la%20respuesta%20de%20dicha%20neurona

Figura 12 *Estructura del algoritmo VGG16*



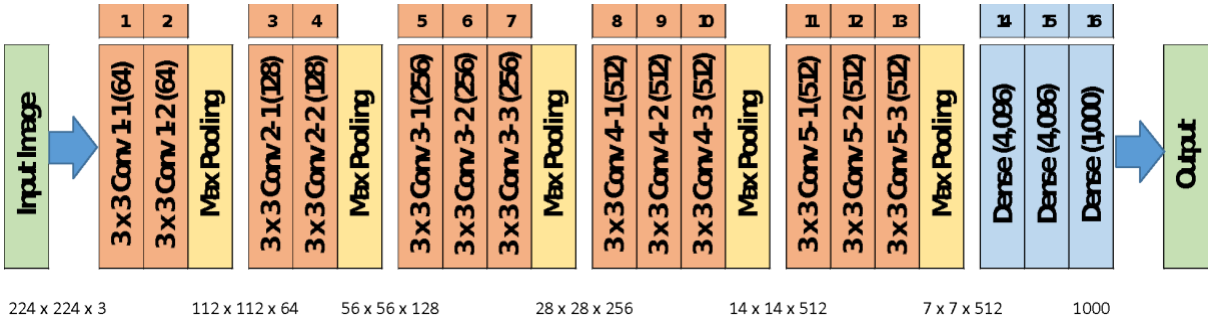
Fuente: VGG 16 Architecture {En línea} Disponible en: https://www.mygreatlearning.com/blog/introduction-to-vgg16/#VGG%2016%20Architecture

En este tipo de red neuronal las configuraciones manejan imágenes de un tamaño de 224 x 224 las cuales están divididas en R, G y B los cuales representan los colores de los pixeles de las imágenes a procesar, esta parte es el procesamiento que reciben las imágenes por pixel.

El proceso inicial es que la imagen pasa por un grupo de capas con un campo receptivo de 3x3 como se menciona anteriormente, posteriormente cuenta con activaciones por medio de ReLu, dentro de estas capas se encuentran 64 filtros, cuando se realiza la convolución la hace teniendo en cuenta un pixel y el relleno también es de 1 pixel, todo esto conservando la resolución del espacio y el mapa que realiza la activación de la salida cuenta con las mismas dimensiones que contaba la imagen del input. Posterior a esto, los mapas de activación pasan por

un max pooling de 2 x 2 con pasos de 2, lo cual hace que el tamaño de las activaciones se reduzca. Las activaciones finales en la primera pila serían de 112 x 112 x 64.

Figura 13 Capas y convoluciones en VGG16



Fuente: My Great Learning. VGG architecture. {En linea} Disponible en: https://www.mygreatlearning.com/blog/introduction-to-vgg16/#VGG%2016%20Architecture

Como se puede observar en la figura, después de este proceso, las siguientes convoluciones pasan por un proceso similar, con 128 filtros. Lo cual hace que el tamaño posterior a esta pila se convierta en 56 x 56 x 128, luego en la tercera capa cuenta con 3 convoluciones y un max pooling con filtro de 256 y una salida de 28 x 28 x 256, posteriormente aparecen tres capas de convolución las cuales traen 512 filtros, estas dan como resultado una salida de 7 x 7 x 512.[55]

T*abla* 1 *Configuración de VGG16*

[55]My Great Learning. VGG 16 Architecture {En línea} {11 de mayo de 2022} Disponible en: https://www.mygreatlearning.com/blog/introduction-to-vgg16/#VGG%2016%20Architecture

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Fuente: My Great Learning. VGG Configurations {En linea} Disponible en: https://www.mygreatlearning.com/blog/introduction-to-vgg16/#VGG%2016%20Architecture

4.2.1.2. VGG19

Este es un modelo de Keras, una variante como tal de VGG el cual cuenta con 19 capas como lo son: 16 capas de convolución, 3 conectadas totalmente, 3 capas de max pool y una capa soft.

Es una red neuronal convolucional profunda que su función es la clasificación de imágenes, en la presente figura, en la columna E se puede observar la arquitectura de este modelo y sus configuraciones. [56]

Tabla 2 Configuración de las redes neuronales sin las capas ReLu

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

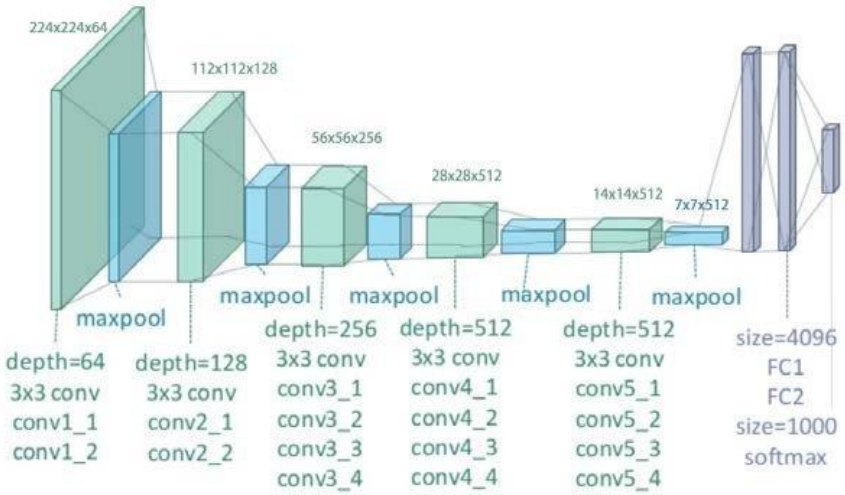
Fuente: OpenGenus IQ: Computing Expertise & Legacy. Understanding the VGG19 Architecture {En línea} {11 de mayo de 2022} Disponible en: https://iq.opengenus.org/vgg19-architecture/#:~:text=VGG19%20is%20a%20variant%20of,VGG19%20has%2019.6%20billion%20FLOPs.

[56] OpenGenus IQ: Computing Expertise & Legacy. Understanding the VGG19 Architecture {El linea} {11 de mayo de 2022} Disponible en: https://iq.opengenus.org/vgg19-architecture/#:~:text=VGG19%20is%20a%20variant%20of,VGG19%20has%2019.6%20billion%20FLOPs.

4.2.1.2.1. Arquitectura del modelo

Cuenta con un tamaño de 224 x 224 de RGB en su input, es decir que su matriz tenía la forma de 224 x 224 x 3, para el preprocesamiento de esto se realiza la resta del RGB un píxel sobre todo el set de datos del conjunto de entrenamiento del algoritmo. Se usan núcleos de 3 x 3 con saltos de 1 pixel con el fin de lograr el cubrimiento de toda la imagen. Además, se cuenta con una función de activación lineal (ReLu) con el fin de involucrar la no linealidad para que la clasificación se realice mejor y los tiempos de cómputos mejores que otros modelos anteriores.[57]

Figura 14 Arquitectura del modelo VGG19



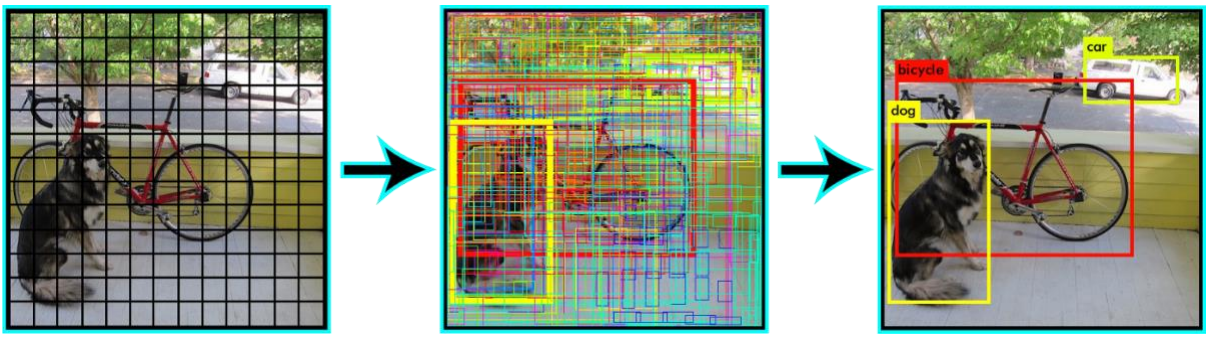
Fuente: Researchgate. Ilustration of the network VGG19 model {En linea} Disponible en: https://www.researchgate.net/figure/Ilustration-of-the-network-

[57] OpenGenus IQ: Computing Expertise & Legacy. Understanding the VGG19 Architecture {El linea} {11 de mayo de 2022} Disponible en: https://iq.opengenus.org/vgg19-architecture/#:~:text=VGG19%20is%20a%20variant%20of,VGG19%20has%2019.6%20billion%20FLOPs.

architecture-of-VGG-19-model-conv-means-convolution-FC-means_fig2_325137356

4.2.1.3. YOLOV3 (You Only Look Once)

Es una arquitectura que detección de objetos sencilla, construida con el fin de agilizar métodos de detección de objetos en video y en tiempo real por medio de una red neuronal convolucional delimitando entre cuadros los objetos que identifique, adicionalmente el modelo lleva con si la ventaja de no requerir un pipeline profundo con solo ejecutar la red neuronal sobre una imagen a predecir basta, es tolerante a errores ya que utiliza toda la imagen y no solo un sector.
En sí el funcionamiento del modelo se basa en una red neuronal convolucional, en la que se busca una etiqueta previamente hecha, esto hace que se identifique los objetos que hay en una imagen, como tal el algoritmo lo que hace es dividir la imagen unas secciones de un tamaño específico lo cual hace que si el objetivo cae en una de estas secciones esta sección va a ser la que detecte el objeto.
El siguiente paso es que dentro de estas etiquetas va dando puntaje de confianza también llamado score, estas puntuaciones son el reflejo de la calidad del modelo y que tan fiable puede ser, en el caso que no haya un objeto en esta sección el score será cero.
Figura 15 Tratamiento de imágenes modelo YOLOV3



Fuente: Detección de objetos con YOLO: implementaciones y cómo usarlas {En línea} Disponible en: https://medium.com/@enriqueav/detecci%C3%B3n-de-objetos-con-yolo-implementaciones-y-como-usarlas-c73ca2489246

En la figura 16 podemos ver el resultado del tratamiento de la arquitectura YOLOV3 para identificar un objeto de una imagen.[58]

4.3. PROPUESTAS DE MÉTODOS AUTOMÁTICOS COMPUTACIONALES

Dentro de la investigación se realizaron varias incursiones en diferentes algoritmos que se ajustaran al dataset previamente construido, estas incursiones consistían en 3 pasos a grandes rasgos, cada uno de estos pasos indican el diseño del método automático.

El primer paso o etapa es entender el algoritmo la arquitectura de este, sus pros y sus contras lo que da como resultado una subtarea la cual era descartar los algoritmos que no se ajustaban a las necesidades y/o requerimientos.

[58] Raquel Miranda Pérez. Jaffette Solano Arias. Abel Méndez Porras. Introducción al Aprendizaje Automático con YOLO {En línea}{10 de mayo de 2022} Disponible en: https://revistas.ulatina.ac.cr/index.php/tecnologiavital/article/download/250/260/547#:~:text=Es%20un%20detector%20de%20objetos,real%20muy%20r%C3%A1pidos%20y%20precisos.

Una vez los algoritmos se ajustarán a las necesidades se procedió a ver sus requisitos, tales como librerías y demás, dado que cada método automático es diferente en algunos aspectos tales como las versiones de Python o de Tensorflow, así mismo fue imperativo en los casos de aquellos algoritmos que no se ejecutaban en Google Colab contar con una buena tarjeta gráfica.

Para el siguiente trabajo de investigación se definieron los siguientes modelos de aprendizaje de máquina para el desarrollo de lo que se planteó en los objetivos, en la tabla 3 podemos encontrar los modelos.

Tabla 3 Modelos utilizados

Modelo	**Nombre**	**Ambiente**
1	VGG16	Google Colab
2	VGG19	Google Colab
3	YOLOv3	Google Colab, Conda

Fuente: Los autores

Posteriormente se continuó con el desarrollo en los ambientes de programación con el dataset construido y se analizaron los resultados que arrojaban, como medida estándar un mínimo del 50% de precisión, si no arrojaba al menos 50% de precisión se descartaron parcialmente, en casos específicos se cambiaron parámetros de la estructura de la red neuronal, como el muestreo.

4.4. DESARROLLO DE LOS MÉTODOS AUTOMÁTICOS

De acuerdo a lo realizado en la fase anterior se continuó con el desarrollo de los métodos automáticos según los modelos seleccionados para los experimentos, como se mencionaba anteriormente, estos modelos fueron desarrollados en distintos ambientes, algunos en Google Colab Pro y otros de manera híbrida, es decir en Google Colab y de manera local con Anaconda que permitió la creación de un entorno propio para las dependencias y con ello poder realizar pruebas del entrenamiento de las redes neuronales ya que fue más fácil acceder a archivos locales como lo son los archivos del dataset.

Las librerías que se utilizaron para el reconocimiento de imágenes se dividen principalmente en dos la relacionadas en la primera imagen que son parte de tensor flow principalmente y también de Sklearn las cuales nos ayudan a identificar medidas que posteriormente nos indicarán por ejemplo qué tan preciso es nuestro modelo, adicionalmente tenemos matplotlib la cual nos ayuda a graficar medidas.

En la siguiente tabla podemos se puede observar los modelos utilizados y sus correspondientes librerías utilizadas

Tabla 4 Modelos usados y sus librerías

Modelo	Librerías

YOLOV3	pytorch=1.1.0, torchvision=0.3.0, cudatoolkit=10.0, pytorch, opencv, numpy matplotlib, tensorboard, terminaltables, pillow tqdm.

Fuente: Los autores

Basado en los diseños de los métodos anteriores, el paso siguiente es poner en práctica y desarrollar los métodos automáticos, teniendo en cuenta los algoritmos preseleccionados. Los modelos se desarrollaron en local y en Google Colab Pro dado que en los dos entornos es importante tener manejo de estos dos entornos por las ventajas que cada uno de estos proporciona, sin embargo, cabe aclarar que el 90% de los modelos corrieron primero en Google Colab dada la facilidad que tienen este para la instalación de paquetes y manejo de versiones de esta.

Posteriormente a esto se realiza el muestreo del conjunto de imágenes setenta porciento (70%) entrenamiento (train), veinte por ciento (20%) de prueba (test) y un diez por ciento (10%) de validación (valid) del algoritmo en cuestión sea cual sea de los algoritmos seleccionados, cabe resaltar que dicha distribución está basado en la literatura y se parte de este para ajustar el que va ser ideal para el método en cuestión dado que estos parámetros deben ser monitoreados para no hacer un sobre ajuste del método. Dicho muestreo se realizó de manera manual armando cada una de las carpetas con la cantidad de imágenes correspondientes por clase según los porcentajes.
Para cada uno de los modelos se realizó un tratamiento similar dejando cada uno de estos en celdas separadas para que a la hora de medir las medidas de desempeño se realizará de forma independiente para cada uno.

4.4.1. VGG16

Lo primero que se realizó fue la importación de librerías, las cuales se pueden evidenciar en la tabla 4 para el presente modelo. Posteriormente en la siguiente línea se realiza la configuración de los parámetros para el entrenamiento, como se ve en la figura 16.

Figura 16 Parámetros del modelo

```
[2] width_shape = 224
    height_shape = 224
    num_classes = 3
    epochs = 50
    batch_size = 16
```

Fuente: Los autores

Estos parámetros fueron fundamentales ya que ellos fueron aquellos fueron estimados cuando se realizó el entrenamiento de los modelos junto al conjunto de imágenes almacenado, estos fueron necesarios para que el modelo aprenda de los datos del conjunto de datos y además fueron necesarios para que el modelo pudiera realizar predicciones.

Para llegar a estos parámetros no existe una formula mágica, se tuvo que hacer prueba y error probando con distintos parámetros, algunos elementos dieron indicios, esto depende de la experiencia que se tenia ejecutando modelos anteriores con parámetros similares y que no daban buenos resultados, debido a ello se tuvo que cambiar.

El batch_size anteriormente estaba en x y se disminuyo a x, pues es recomendable que no sea tan grande dicho parámetro cuando el dataset no es tan grande, cuando hay limitaciones de memoria o recursos computacionales, en este caso puntual el dataset no fue muy grande.

Además, se cambio el optimizador de Adam a Adadelta porque era uno de los posibles origines de que no converja bien.

Para saber que parámetros se deben usar en cada caso existen técnicas como AutoML en donde se programa el sistema para que pruebe muchos parámetros y varíe los hiper-parametros, incluso la arquitectura de las redes y el sistema utiliza la mejor

El siguiente paso fue montar el Drive en donde se almacenó el conjunto de imágenes el cual se distribuyó en las distintas categorías mencionadas anteriormente y el cual además contó con un muestreo de 70-20-10. En la figura 18 podemos observar la manera en que se realiza este procedimiento

Figura 17 montaje almacenamiento de drive

```
from google.colab import drive
drive.mount('/content/drive')
```

Fuente: Los autores

Se agregó cada uno de los paths de las carpetas del Google drive en donde se encontraban las imágenes, como se muestra en la figura 19

Figura 18 Directorios en Google Drive

```
[50] !cp -r "/content/drive/MyDrive/TRABAJO DE GRADO/Classification/train" "/content/"
     !cp -r "/content/drive/MyDrive/TRABAJO DE GRADO/Classification/valid" "/content/"
     train_data_dir = '/content/train'
     validation_data_dir = '/content/valid'

     !cp -r "/content/drive/MyDrive/TRABAJO DE GRADO/Classification/test" "/content/"
```

Fuente: Los autores

Posteriormente a esto se realiza con los algoritmos la generación de las imágenes para entrenamiento y validación, esto lo podemos evidenciar en las figuras 19 y 20.

Figura 19 Generador de imágenes de train y valid

```
train_datagen = ImageDataGenerator(
    rotation_range=20,
    zoom_range=0.2,
    width_shift_range=0.1,
    height_shift_range=0.1,
    horizontal_flip=True,
    vertical_flip=True,
    preprocessing_function=preprocess_input)

valid_datagen = ImageDataGenerator(
    rotation_range=20,
    zoom_range=0.2,
    width_shift_range=0.1,
    height_shift_range=0.1,
    horizontal_flip=True,
    vertical_flip=True,
    preprocessing_function=preprocess_input)
```

Fuente: Los autores

Figura 20 Generador de imágenes de train y valid 2

```
train_generator = train_datagen.flow_from_directory(
    train_data_dir,
    target_size=(width_shape, height_shape),
    batch_size=batch_size,
    #save_to_dir='',
    class_mode='categorical')

validation_generator = valid_datagen.flow_from_directory(
    validation_data_dir,
    target_size=(width_shape, height_shape),
    batch_size=batch_size,
    #save_to_dir='',
    class_mode='categorical')

Found 404 images belonging to 3 classes.
Found 111 images belonging to 3 classes.
```

Fuente: Los autores

En la siguiente línea de código se procede a poner cuantas imágenes hay en las carpetas de train y cuantas en valid, esto con base en las figuras anteriores, como se puede observar en la figura 22 a continuación:

Figura 21 división de imágenes train vs samples

```
nb_train_samples = 404
nb_validation_samples = 111
```

Fuente: Los autores

Posteriormente a estas configuraciones, se corrió el modelo mencionado, en el cual se establecen los parámetros por capas de este modelo, luego de correr esto se evidenció que al final se congelan todas las capas menos la capa dense. En la figura 22 podemos observar la cantidad de parámetros total del modelo, los parámetros entrenables y los que no son entrenables.

Figura 22 Parámetros del modelo a entrenar

```
=================================================================
Total params: 17,942,979
Trainable params: 3,228,291
Non-trainable params: 14,714,688
_________________________________________________________________
```

Fuente: Los autores

En la figura 23, podemos observar el siguiente paso, en el cual se ejecuta el entrenamiento del modelo, en el entrenamiento lo que se hizo fue que con los datos almacenados para entrenamiento en la carpeta train el modelo aprendiera para poder clasificar las imágenes que se le proporcionarán.

Figura 23 Entrenamiento del modelo

```
# Entrenamos el modelo
model_history = custom_model.fit(
    train_generator,
    epochs=epochs,
    validation_data=validation_generator,
    callbacks=[tensorboard_callback])
```

Fuente: Los autores

El entrenamiento normalmente se tarda en realizar, pero en este caso como se realizó con Google Colab Pro y sus máquinas virtuales con GPUs especiales para esto el entrenamiento no tardó mucho tiempo, además que no todos los miembros del equipo contaban con un computador con tarjeta gráfica, además otros factores que afectan el tiempo de este entrenamiento también se encontró la velocidad del internet con el que se realizó. En la figura 24 se observa la manera en que se guarda el modelo, el cual se va a encontrar en la nube en la carpeta de archivos.

Figura 24 Guardado del modelo en Google Colab

```
#Guardar el modelo que se entrenó
custom_Model= load_model("VGG16.h5")
```

Fuente: Los autores

Luego de que se tuviera el modelo entrenado, se procedió a ejecutar las líneas de código que arrojaron las medidas de desempeño del modelo entrenado y también la matriz de confusión, estas herramientas son aquellas que permitieron ver cómo le fue al modelo con la clasificación de las imágenes.

4.4.2. VGG19

Para este modelo se realizaron las mismas configuraciones y pasos mencionados anteriormente en el modelo 1, con los mismos parámetros y conjunto de imágenes.

Al correr este modelo y entrenarlo difirió en la cantidad de capas que se ejecutaron en el entrenamiento de este, con el mismo conjunto de imágenes, VGG19 trabaja con 19 capas convolucionales y VGG16 con 16 capas convolucionales.

Luego del entrenamiento del modelo, se mostraron las medidas de desempeño y la matriz de confusión, las cuales en el próximo capítulo se evidenciará los resultados que se obtuvieron de estas.

4.4.3. YOLOV3

Como primer paso para el entrenamiento elaboramos o creamos un entorno virtual como vemos en la figura 26 y una vez ya teniendo las imágenes entrenadas, procedemos a crear nuestro entorno virtual para nuestro modelo, el entorno que vamos a crear está en py == 3,6 ya que las librerías que utilizamos son para esta versión de py y son más estables que para posteriores versiones.

Figura 25 Librerías utilizadas para el desarrollo de los modelos

```
Downloading and Extracting Packages
pytables-3.6.1        | 2.2 MB    | ##################################################################### | 100%
spyder-4.1.4          | 5.7 MB    | ##################################################################### | 100%
pytz-2020.1           | 184 KB    | ##################################################################### | 100%
zope.event-4.4        | 10 KB     | ##################################################################### | 100%
mkl_random-1.0.4      | 289 KB    | ##################################################################### | 100%
tblib-1.6.0           | 16 KB     | ##################################################################### | 100%
mpir-3.0.0            | 1.3 MB    | ##################################################################### | 100%
pycodestyle-2.6.0     | 43 KB     | ##################################################################### | 100%
certifi-2020.6.20     | 156 KB    | ##################################################################### | 100%
pickleshare-0.7.5     | 13 KB     | ##################################################################### | 100%
tqdm-4.47.0           | 62 KB     | ##################################################################### | 100%
wheel-0.34.2          | 66 KB     | ##################################################################### | 100%
heapdict-1.0.1        | 9 KB      | ##################################################################### | 100%
ujson-1.35            | 24 KB     | ##################################################################### | 100%
gmpy2-2.0.8           | 139 KB    | ##################################################################### | 100%
greenlet-0.4.16       | 26 KB     | ##################################################################### | 100%
parso-0.7.0           | 72 KB     | ##################################################################### | 100%
mccabe-0.6.1          | 14 KB     | ##################################################################### | 100%
jupyterlab_server-1.  | 25 KB     | ##################################################################### | 100%
freetype-2.10.2       | 470 KB    | ##################################################################### | 100%
python-3.6.10         | 15.9 MB   | ##################################################################### | 100%
wincertstore-0.2      | 14 KB     | ##################################################################### | 100%
sphinx-3.1.2          | 1.1 MB    | #                                                                     |   1%
```

Fuente: Los autores

Una vez ejecutamos el comando para crear nuestro entorno nos descarga las paqueterías necesarias para su ejecución como lo podemos ver en la figura 26.

Figura 26 activación entorno virtual

```
(base) C:\Users\user>conda activate YoloCustom

(YoloCustom) C:\Users\user>
```

Fuente: los autores

Una vez tengamos el entorno instalado lo activamos como vemos en la figura 26 y dentro instalamos librerías adicionales para la ejecución del proyecto estas librerías son: opencv-python, numpy, matplotlib, tensorboard, terminaltables, pillow pytorch==1.1.0, torchvision==0.3.0, cudatoolkit=10.0, pytorch
y tqdm las cuales son fundamentales para el entrenamiento del modelo Yolov3 y listo tenemos nuestro entorno virtual preparado.

Una vez preparamos nuestro entorno en la máquina local procedemos al montaje en Google colab donde se realizó el entrenamiento, ya que es una plataforma

óptima para este tipo de procesos, sin embargo, la prueba del algoritmo se realizó en el computador local donde se comprobó el funcionamiento de este.

4.5. EVALUACIÓN DE RENDIMIENTO DE LOS MÉTODOS AUTOMÁTICOS

En este apartado se evaluaron las medidas de desempeño de los modelos, cada uno con características precisas como el tipo de arquitectura.

4.5.1. VGG 16

Figura 27 Última época de ejecución VGG 16 modelo 1

	precision	recall	f1-score	support
0	1.0000	0.1500	0.2609	20
1	0.7083	1.0000	0.8293	17
2	0.6129	0.9048	0.7308	21
accuracy			0.6724	58
macro avg	0.7737	0.6849	0.6070	58
weighted avg	0.7744	0.6724	0.5976	58

Fuente los autores

En el Modelo 1 como se muestra en la figura 27 se tiene como resultado unas medidas de desempeño bastante buenas recalcando el Recall del layer 0 que nos da medidas muy bajas esto nos indica que al momento de identificar esta clase el modelo se confunde, sin embargo, dado que la precisión es muy alta se puede inferir que el problema puede ser ajustar el data set y precisamente en la clase de metal.

También la figura 27 existe una medida importante como lo es el F1-score la cual indica un promedio entre Recall y precisión esto es importante ya que hace más

fácil poder comparar el Recall y la precisión en un solo número y vuelve a reforzar el problema que se tuvo en la clase 0.

4.5.2. VGG19

Figura 28. Última época de ejecución VGG 19 modelo 2

	precision	recall	f1-score	support
0	0.2500	0.1500	0.1875	20
1	0.3226	0.5882	0.4167	17
2	0.4667	0.3333	0.3889	21
accuracy			0.3448	58
macro avg	0.3464	0.3572	0.3310	58
weighted avg	0.3497	0.3448	0.3276	58

Fuente: Los autores

En la figura 29 podemos observar que las medidas que tenemos son muy bajas esto como consecuencia de su arquitectura la cual cuenta con 19 capas y 16 son convolucionales lo que cambia completamente la función de activación.

4.5.3. YOLOV3

Figura 29. Última época de ejecución YoloV3 modelo 3

```
---- [Epoch 199/200, Batch 132/133] ----
+------------+--------------+--------------+--------------+
| Metrics    | YOLO Layer 0 | YOLO Layer 1 | YOLO Layer 2 |
+------------+--------------+--------------+--------------+
| grid_size  | 12           | 24           | 48           |
| loss       | 0.741776     | 0.396764     | 0.489162     |
| x          | 0.008884     | 0.008953     | 0.032346     |
| y          | 0.018885     | 0.043027     | 0.149009     |
| w          | 0.003110     | 0.004018     | 0.000892     |
| h          | 0.011221     | 0.001494     | 0.028007     |
| conf       | 0.322207     | 0.331003     | 0.273057     |
| cls        | 0.377468     | 0.008269     | 0.005849     |
| cls_acc    | 66.67%       | 100.00%      | 100.00%      |
| recall50   | 0.666667     | 1.000000     | 1.000000     |
| recall75   | 0.666667     | 1.000000     | 1.000000     |
| precision  | 0.500000     | 0.500000     | 0.157895     |
| conf_obj   | 0.977595     | 0.938290     | 0.944640     |
| conf_noobj | 0.000794     | 0.000611     | 0.000729     |
+------------+--------------+--------------+--------------+
Total loss 1.627701997756958
---- ETA 0:00:00
```

Fuente: Los autores

Como podemos observar en la figura 29 tenemos medidas de desempeño, las más importantes y con las que vamos a analizar los modelos son Recall y precisión.

Empezamos con el Recall la cual en el modelo 3 identificamos que tenemos porcentajes altos indicando que el modelo 3 es capaz de identificar con claridad las clases que le pasamos (Layer cero es metal, layer uno es plástico y layer 2 es vidrio) sin embargo cuando pasamos a las medidas de precisión nos damos cuenta que no tiene una calidad alta sino que por el contrario cuando intenta identificar el layer 0 y layer 1, tiene un 50% de probabilidad de equivocarse en la clase, lo cual lo hace un modelo impreciso al momento de identificar la clase, en conclusión el modelo 3 identifica a la perfección donde hay un un objeto de los que queremos identificar pero no sabe muy bien qué tipo de objeto es.

Cuando se observa con detenimiento las medidas del layer 1 y 2 se infiere que tiende a confundirse por lo cual es necesario cambiar de arquitectura, argumentando que siendo el modelo 5 un modelo basado en la detección de objetos y estos dos objetos plástico y vidrio tienen características similares que a la hora de compararlos no tienen diferencias, se recomienda que en trabajos futuros se enfoquen más en estas dos clases y en arquitecturas basadas en índices de refracción.

5. RESULTADOS Y ANÁLISIS DE RESULTADOS

En el presente capítulo se menciona cada uno de los resultados obtenidos en la presente investigación de trabajo de grado, además el análisis respectivo que se hizo a cada una de las etapas del desarrollo metodológico.

5.1. Construcción del conjunto de datos de imágenes

En cuanto al resultado de la creación del conjunto de datos de imágenes de residuos el cual fue utilizado para los experimentos realizados con los modelos, se utilizaron 206 imágenes del tipo plástico, 199 imágenes de tipo vidrio y 194 imágenes del tipo metal. Esto para un total de 599 imágenes en todas las clases.

Tabla 4: Resultado construcción del conjunto de imágenes

Tipo de residuo	Cantidad de imágenes
Plástico	206
Vidrio	199
Metal	174

Fuente: Los autores

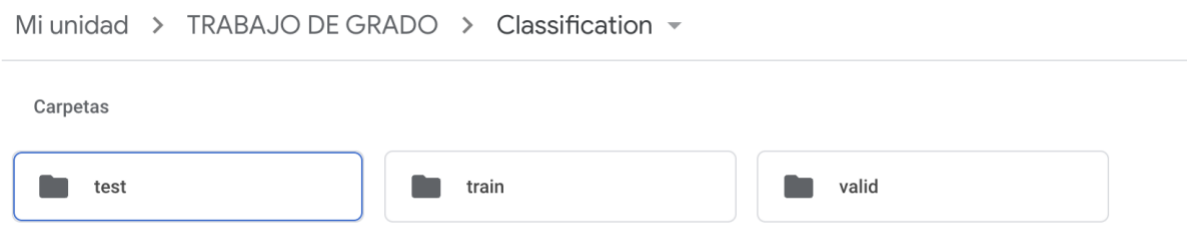
Las imágenes del conjunto de datos provinieron de diferentes fuentes como data sets conocidos en la industria y de distintos repositorios, estas imágenes venían algunas en formato PNG, JPG y JPEG las cuales fueron convertidas todas a .JPG el cual es menos pesado que los otros formatos mencionados anteriormente.

Para el tratamiento de las imágenes con algunos algoritmos se realizó el etiquetado de las imágenes, esto con el fin de que el algoritmo sepa bien en toda la imagen donde se encuentra el residuo y además sepa identificar correctamente los diferentes tipos de residuo en la imagen, este etiquetado se realizó utilizando *LabelImg*, un software realizado con Python el cual permite que el algoritmo realice la detección de una mejor manera. En cuanto al muestreo de las imágenes se utilizó una parte de imágenes para entrenamiento y pruebas, para el entrenamiento un 70% y test para 30%.

En algunos casos, en el data set se realizó la carga de los archivos a Google Drive, por la naturaleza de algunos algoritmos, en los cuales se se hizo el desarrollo en Google Colab el cual ofrece máquinas virtuales con grandes virtudes en cuanto a GPU y CPU, las cuales son muy útiles a la hora de realizar entrenamiento de algoritmos con grandes conjuntos de datos y que requieren de gran capacidad de procesamiento de imágenes.

Fue importante que las imágenes se encontrarán recortadas, con el fin de que el algoritmo no confunda el objeto con lo demás que se encontrara al fondo, esto en el caso que no requieren un etiquetado de las imágenes.

Figura 30. Unidad en Google Drive conjunto de imágenes



Fuente: Los autores

Se creó una carpeta en Google Drive donde internamente se encontraban otras tres carpetas: test, train y valid. En la carpeta *train* se encontraban aquellas imágenes que se usaron para el entrenamiento del algoritmo, que en este caso fueron del 70%, en la carpeta de *test* se encontraban las imágenes con las cuales se iba aprobar el entrenamiento realizado por el algoritmo, estas constituían el 30% de las imágenes.

5.2. Diseño de la estrategia metodológica

Como resultado de la investigación del estado del arte, información relevante en el tema, estudios realizados y demás, se diseñó una estrategia metodológica compuesta por seis etapas, esta fue aplicada para la clasificación de tipos de residuos en parques en Colombia en donde se utilizaron 579 imágenes armadas con base en varias fuentes de información.

La segunda fase fue la categorización de los datos, la cual se puede observar en la imagen B de la figura 31, la cual consistió en tomar las imágenes del conjunto de datos y dividirlas según la clase que fueran, es decir, ya sean plástico, vidrio y metal, esto con el fin de que el algoritmo supiera las imágenes a qué clase pertenecen

En la fase de transformación de los datos, mostrada en la imagen C, consistió en tomar las imágenes que se encontraban en el conjunto de datos y etiquetarlas, este proceso se realizó con el fin de entrenar los algoritmos de tipo supervisado, con el fin de que el algoritmo se entrene con el historial de los datos y aprenda y ofreciera una etiqueta de salida, lo que hace el predecir el output. En los casos de los algoritmos no supervisados que se revisaron para el siguiente proyecto de investigación, no fue necesario la transformación de estos datos, solo la agrupación de los tipos de clases de imágenes en diferentes carpetas en Google Drive

En la cuarta fase, de selección del método de clasificación de imágenes, visualizada en la imagen D, se realizó la investigación de varios métodos de clasificación de imágenes por medio de aprendizaje de máquina, como lo son: *VGG16, VGG19, YoloV3* A partir de los resultados que se mostraron en cada uno de los algoritmos, se escogió el algoritmo *VGG16* por sus resultados y su capacidad de reconocimiento de las imágenes

En la fase de entrenamiento y pruebas de algoritmos mostrada en el inciso E, se realizó el entrenamiento del algoritmo seleccionado, dividiendo el entrenamiento y pruebas en dos partes, un 70% de las imágenes enfocadas para el entrenamiento de las imágenes y un 30% para las pruebas luego del entrenamiento, esto con el fin de ver si el algoritmo fue capaz de realizar la clasificación de imágenes, sin importar de qué conjunto de datos provenga.

5.3. Desarrollo del método automático

En el desarrollo del método automático seleccionado, se pudo evidenciar varias etapas en su realización, el algoritmo seleccionado fue VGG16, una red neuronal convolucional que por su robusta arquitectura se ha posicionado como una de

las mejores en cuanto a visión por computador, además de ser una red que cuenta con

16 capas cada una con determinados pesos. Es una red neuronal bastante grande y con gran cantidad de parámetros.

Una vez con el conjunto de datos almacenados en Google Drive, en sus respectivas carpetas se procede a realizar las siguientes importaciones en Google Colab.

5.4 Evaluación de rendimiento del método automático

Tabla 5 Análisis de medidas de desempeño

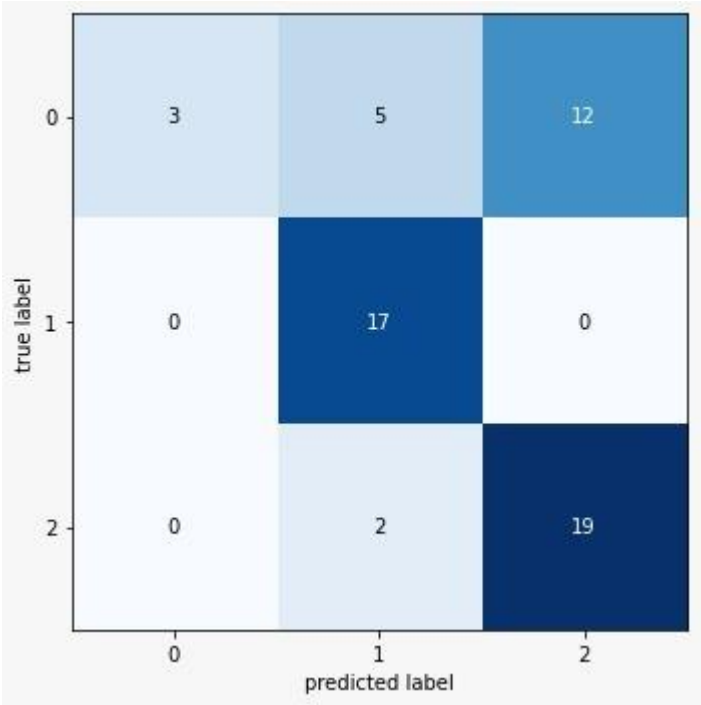
Modelo	Medida	Análisis
VGG16	F1-SCORE f1-score 0.2609 0.8293 0.7308	Como se estaba mencionando en capítulos anteriores, esta es una de las medidas mas importantes porque son un conjunto de resultados entre el recall y la precisión, en este caso en particular estas medidas fueron bastante altas en el label 1,2 lo que muestra que el modelo es capaz de conocer estas medidas en mas del 70% de las veces que aparecen
VGG19	F1 Score	En este caso en particular se puede apreciar unas medidas bajas ya que la arquitectura, no favorece el tipo de data set que

	f1-score 0.1875 0.4167 0.3889	se construyó además que al parecer la función de activación tendía a confundirse entre las tres clases llegando hasta a no identificar ninguna de las clases.
YoloV3	Recall y precisión recall50 | 0.666667 | 1.000000 | 1.000000 recall75 | 0.666667 | 1.000000 | 1.000000 precision | 0.500000 | 0.500000 | 0.157895	En el presente modelo podemos apreciar como las medidas de desempeño varían entre la precisión y el Recall, lo cual significa que, aunque el modelo es capas de reconocer las imágenes y cuando las reconoce es un verdadero positivo, se le dificulta reconocerlas, lo que con variación de parámetros puede ser mas exacto sin embargo dado los objetivos del trabajo y el alcance de este no se abordó más en el tema

Fuente: los autores

Como se puede apreciar en la tabla 5 las medidas de desempeño de los modelos dejan ver que tan precisos son estos teniendo en cuenta cuantos falsos positivos puede llegar a dar (Recall) o que tan preciso es a la hora de categorizar los objetos en una imagen (precisión), esto importante ya que el funcionamiento del modelo está directamente relacionado con esto.

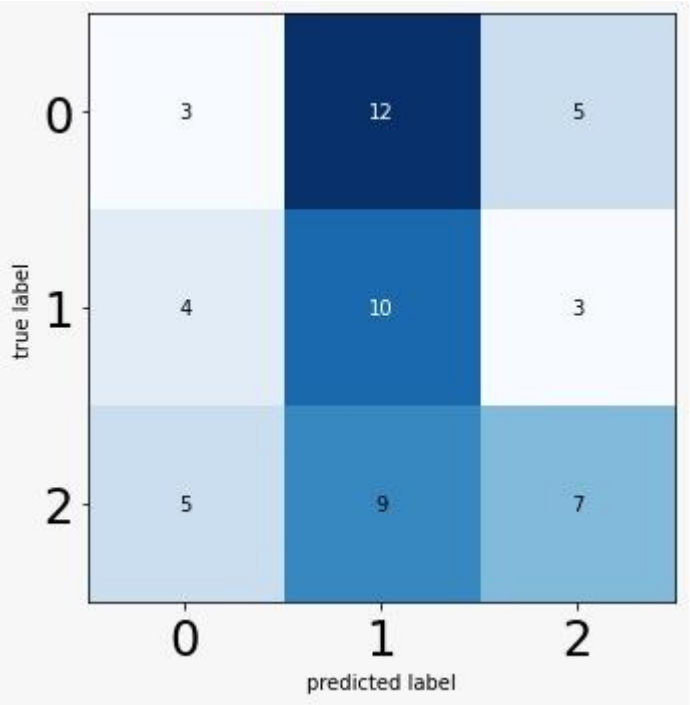
Figura 31 Matriz de confusión VGG 16



Fuente: Los autores

En la figura 31 se puede ver los labels (clases) que se analizaron donde 0 es vidrio, 1 es metal y 2 es plástico. Las filas significan el valor real de la matriz y las columnas lo que identifico el modelo es decir en este caso se puede inferir que de el label 0 solo detecto 3 imágenes de las 20 en total que eran de vidrio en cambio el label 1 detecto exactamente 17 de las 17 que eran de metal, al igual que el label 2 que detecto 19 de las 21 que eran de plástico, gracias a esta matriz de confusión podemos saber que hay que hacer un tratamiento diferente para vidrio ya que la mayoría de imágenes de vidrio las detecto como si fuesen plástico.

Figura 32. Matriz de confusión VGG 19



Fuente: Los autores

En la figura 32 se puede ver los labels (clases) que se analizaron donde 0 es vidrio, 1 es metal y 2 es plástico. En este caso se puede ver un patrón similar que el de la figura 31 sin embargo la figura 32 deja en claro que el modelo no logro identificar ninguna clase de forma precisa, pero sigue dando como resultado que la clase de metal es la más precisa y que la de vidrio la menos precisa.

En conclusión, tenemos que el modelo que acertó más fue el VGG 16 y que el modelo de Yolo v3 se debe reajustar y parametrizar de forma diferente para llegar a una conclusión más precisa de este, sin embargo, se puede trabajar dado que las medidas de Yolo son buenas, pero les falta más trabajo a profundidad, es importante que en trabajos futuros se tenga muy en cuenta esto.

5.5 Algoritmo seleccionado

Durante el desarrollo de los algoritmos se realizó un proceso de depuración para focalizar el desarrollo en el algoritmo con mejores medidas de desempeño y la arquitectura que más se adaptara a las necesidades propuestas por los autores. Por consiguiente, se tomó la decisión de continuar con el algoritmo de VGG16, dado que este algoritmo fue el que mejores medidas de desempeño tuvo basado en las métricas de desempeño que fueron Precisión, Recall y F1 score las cuales las podemos evidenciar en la figura 27. Esto muestra, que la arquitectura VGG 16 al ser una arquitectura pre entrenada en la base de Keras facilita los cálculos de los pesos de las imágenes reconociendo de una forma más precisa el conjunto de imágenes con el cual se trabajó, sobre todo, porque no requiere un cómputo avanzado para su funcionamiento, es decir es una arquitectura sencilla y confiable.

Como si fuera poco es una red que tiene bastante aplicabilidad en muchos de los problemas que agobian la sociedad y que en este particular caso podemos decir que funciono de una forma concluyente para el data set utilizado.

6. IMPACTO SOCIAL

En los resultados de la presente investigación se evidenció que la realización de un método de selección de imágenes por medio de aprendizaje automático ayuda a la problemática planteada del cambio climático ya que los residuos pueden ser identificados fácilmente, pero es importante saber las clases que se van a clasificar, ya que cuando las imágenes a clasificar de residuos presentan similitudes muy grandes, al algoritmo se le dificulta decir que residuo es cada uno.

Esto conlleva a que la clasificación de basuras sea más precisa y se aprovechen residuos evitando que lleguen a rellenos y haya contaminación y propagación de

enfermedades, sin embargo, para trabajos futuros es importante hacer una integración de arquitecturas mixtas para la identificación de residuos y que materiales como vidrio y plástico sean identificados de manera más precisa.

7. CONCLUSIONES

- Es claro lo que elementos se deben tener en cuenta para la construcción del método automático para la clasificación de residuos sólidos, como lo podemos observar en la investigación, lo primero que se debe tener en cuenta como elemento fundamental es un data set robusto el cual pueda suplir necesidades como la clasificación entre vidrio y plástico que son dos clases que deben estar muy bien alimentadas de imágenes de nuestras clases.
- Es fundamental entender el tipo de arquitectura del modelo que se va a montar ya que el método influirá de forma directa en la precisión de nuestro método automático, gracias a la investigación que se realizó concluimos que arquitecturas basadas en reconocimiento de imágenes y no en reconocimiento de objetos, tuvo un mejor desempeño.
- Como consecuencia del desarrollo de la investigación, se identificó que si bien este es un avance significativo para proyectos venideros es importante incursionar en otras infraestructuras de métodos como lo pueden ser las arquitecturas para el reconocimiento de texturas o también modificar funciones de activación.
- Otro de los aspectos fundamentales que son necesarios para la construcción del método automático es el algoritmo que se escoja para el entrenamiento, la infraestructura de este y la división de imágenes para pruebas y entrenamiento, esto último es fundamental para un buen desempeño de cualquiera de los algoritmos que se escojan, en nuestro

caso por la naturaleza de data set y la literatura que se siguió el mejor desempeño lo tuvimos con VGG16 puesto que es una arquitectura que se adapta de forma perfecta a data set pequeños o medianos, además que es más rápida que otros modelos.

8. RECOMENDACIONES Y TRABAJOS FUTUROS

- Construir un data set más robusto tomando imágenes de diferentes proyectos incluyendo este.
- Utilizar herramientas como Google Colab para aumentar el rendimiento de los procesos ejecutados por gráfica, combinándolo con pruebas del algoritmo entrenado en local, para tener mayor facilidad a la hora de interactuar con conjuntos de archivos grandes en pruebas.
- No descartar algoritmos ni infraestructuras descartadas en otras investigaciones ya que estas son totalmente diferentes y con herramientas como Google Colab podemos probar muchas en diferentes celdas.

9. BIBLIOGRAFÍA

[1] BANCO MUNDIAL Informe del Banco Mundial: Los desechos a nivel mundial crecerán un 70 % para 2050, a menos que se adopten medidas urgentes {En línea} {15 de octubre de 2021} disponible en: https://www.bancomundial.org/es/news/press-release/2018/09/20/global-waste-to-grow-by-70-percent-by-2050-unless-urgent-action-is-taken-world-bank-report

[2] NOTICIAS ONU. Como la basura afecta el desarrollo de América Latina {En línea} {15 de octubre 2021} Disponible en: https://news.un.org/es/story/2018/10/144356

[3] Organización Mundial de la Salud, RESPUESTA MUNDIAL PARA EL CONTROL DE VECTORES 2017–2030{En línea}{10 de agosto 2021} Disponible en:https://www.who.int/malaria/areas/vector_control/Draft-WHO-GVCR-2017-2030-esp.pdf

[4] EL TIEMPO, Enseñan a reciclar en los centros comerciales de Bogotá {En línea}{16 de octubre 2021} Disponible en:https://www.eltiempo.com/bogota/como-reciclar-en-los-centros-comerciales-de-bogota-136186

[5] NACIONES UNIDAS. La crisis climática - una carrera que podemos ganar {En línea} {16 de octubre} Disponible en: https://www.un.org/es/un75/climate-crisis-race-we-can-win

[6] REVISTA SEMANA. El 78% de los hogares colombianos no recicla {En línea} {16 de octubre 2021} disponible en: https://www.semana.com/medio-ambiente/articulo/el-78-de-los-hogares-colombianos-no-recicla/44231/

[7] EL NUEVO SIGLO. ¿Cómo vamos en Colombia con el reciclaje? {En línea} {16 de octubre de 2021} disponible en: https://www.elnuevosiglo.com.co/articulos/03-2019-como-vamos-en-colombia-con-el-reciclaje

[8] Riesgos por causas de los residuos sólidos {En línea} {16 de octubre de 2021} disponible en: https://sites.google.com/site/residuossolidos113/11c

[9] UPB SOSTENIBLE. 5 Preguntas para entender el reciclaje en Colombia {En línea} {16 de octubre de 2021} disponible en: https://www.upb.edu.co/es/central-blogs/sostenibilidad/cinco-preguntas-para-entender-el-reciclaje

[10] Canales sectoriales Interempresas. e-Garbage, un sistema inteligente de gestión de la recogida de residuos {En línea} {16 de octubre de 2021} disponible en: https://www.interempresas.net/Equipamiento_Municipal/Articulos/49606-e-Garbage-un-sistema-inteligente-de-gestion-de-la-recogida-de-residuos.html

[11] AYUNTAMIENTO DE HUELVA. ¿Qué es la contaminación ambiental? . {En línea} {18 de octubre de 2021} Disponible en: http://www.lineaverdehuelva.com/lv/consejos-ambientales/contaminantes/Que-es-la-contaminacion-ambiental.asp

[12] ALCALDÍA DE BOGOTÁ. Recolección de residuos sólidos. {En línea}{17 de octubre de 2021} Disponible en:

https://bogota.gov.co/servicios/guia-de-tramites-y-servicios/recoleccion-de-residuos-solidos

[13] RECIMED COOPERATIVA MULTIACTIVA DE RECICLADORES. Reciclar en la fuente es la opción {En línea}{17 de octubre de 2021} Disponible en: https://reciclaje.com.co/blog/aprende-a-reciclar/separar-en-la-fuente-es-la-opcion/

[14] ARTEAGA, FRANCISCO. ¿Qué es un foodcourt? {En línea}{17 de octubre de 2021} Disponible en: https://ingenieriademenu.com/food-court/

[15] BBVA. ¿Qué es el reciclaje y por qué es importante reciclar? {En línea} {17 de octubre de 2021} Disponible en: https://www.bbva.com/es/sostenibilidad/que-es-el-reciclaje-y-por-que-es-importante-reciclar/

[16] WWF. Glosario ambiental: ¿Qué es el plástico? {En linea} {18 de octubre}https://www.wwf.org.co/?328912/Glosario-ambiental-Que-es-el-plastico

[17] GOBIERNO ESPAÑA, Papel y carton {En linea} {18 de octubre de 2021} Disponible en: https://www.miteco.gob.es/es/calidad-y-evaluacion-ambiental/temas/prevencion-y-gestion-residuos/flujos/domesticos/fracciones/papel-y-carton/Como-se-trata.aspx

[18] GOBIERNO DE ESPAÑA, ¿Qué es el vidrio? {En línea} {18 de octubre de 2021} Disponible en: https://www.miteco.gob.es/fr/calidad-y-evaluacion-ambiental/temas/prevencion-y-gestion-residuos/flujos/domesticos/fracciones/vidrio/

[19] RSU CONSORCIO PROVINCIAL DE RESIDUOS ORGÁNICOS, Residuos orgánicos {En línea} {19 de octubre de 2021} Disponible en: https://www.consorciorsumalaga.com/5936/residuos-organicos

[20] EDX, Ciencias de la computación. {En línea} {18 de octubre de 2021} Disponible en: https://www.edx.org/es/aprende/ciencias-de-la-computacion

[21] SUMUP. Inteligencia Artificial - ¿Qué es la Inteligencia Artificial? {En línea} {18 de octubre de 2021} Disponible en:

https://sumup.es/facturas/glosario/inteligencia-artificial/

[22] GONZÁLEZ, FABIO. Aprendizaje de Máquina Maestria en Ingenieria - Ingenieria de Sistemas y Computación. {En línea} {18 de octubre de 2021} Disponible en:
https://dis.unal.edu.co/~fgonza/courses/2007-I/ml/index.html

[23] SANCHEZ, ANTONIO. ¿Cómo aprenden las máquinas? Machine Learning y sus diferentes {En línea} {18 de octubre de 2021} Disponible en: tipos https://datos.gob.es/es/blog/como-aprenden-las-máquinas-machine-learning-y-sus-diferentes-tipos

[24] ARSYS. Reconocimiento de imágenes . {En línea} {19 de octubre de 2021} Disponible en: https://www.arsys.es/blog/reconocimiento-imagenes/

[25] TELEFÓNICA TECH. Datos de entrenamiento vs datos de test. {En línea} {19 de octubre de 2021} Disponible en: https://empresas.blogthinkbig.com/datos-entrenamiento-vs-datos-de-test/

[26] NAGESH, SINGH. Métricas De Evaluación De Modelos En El Aprendizaje Automático
https://www.datasource.ai/es/data-science-articles/metricas-de-evaluacion-de-modelos-en-el-aprendizaje-automatico

[27] Nick McCrea. Introducción a la Teoría de Aprendizaje de Máquina y sus Aplicaciones: Un Tutorial Visual con Ejemplos {En línea} {16 de octubre de 2021} disponible en:https://www.toptal.com/machine-learning/introduccion-a-la-teoria-de-apren

[28] Salvador García, Sergio Ramírez-Gallego, Julián Luengo, Francisco Herrera. Big Data: Preprocesamiento y calidad de datos{En línea} {16 de septiembre de 2021} disponible

en:https://sci2s.ugr.es/sites/default/files/ficherosPublicaciones/2133_Nv237-Digital-sramirez.pdf
[29] UNIR. Tipos de muestreo: los principales y sus características {En línea} {16 de septiembre de 2021} disponible en:https://www.unir.net/ingenieria/revista/tipos-de-muestreo/
[30] ITELLIGENT. Machine Learning & NPL: cómo funciona un clasificador de documentos{En línea} {16 de septiembre de 2021} disponible en:https://itelligent.es/es/pln-clasificacion-automatica-documentos/
[31] Daniel Ortego Delgado el 15 de Septiembre de 2017" {En Línea} {29 de octubre 2021} Disponible en: ¿Qué es Tensorflow? | OpenWebinars
[32] MARTINEZ, José, "15 Librerías de Python para Machine Learning" {En Línea} {29 de octubre 2021} Disponible en (https://www.iartificial.net/librerias-de-python-para-Machine-learning/) p.40

[33] Lima Acervo, "VGG-16 | Modelo de CNN"{En Línea} {29 de octubre 2021} Disponible en (https://es.acervolima.com/2021/02/09/vgg-16-modelo-de-cnn/)
[34] Alexander Lenail, "AlexNet: la CNN que cambió la visión por computadora" {En Línea} {29 de octubre 2021} Disponible en https://ichi.pro/es/alexnet-la-cnn-que-cambio-la-vision-por-computadora-169459181425709)
[35] Merkle logo, El algoritmo K-NN y su importancia en el modelado de datos{En Línea} {29 de octubre 2021} Disponible en:https://www.merkleinc.com/es/es/blog/algoritmo-knn-modelado-datos
[36] Shin Terence. Comprensión de la matriz de confusión y como implementarla en python,{En línea}{15 de noviembre 2021} disponible en:Comprensión de la Matriz de Confusión y Cómo Implementarla en Python (datasource.ai)

[37] FIENO, ALEXANDRE. SOARES, EDUARDO. Artificial Intelligence in Automated Sorting in Trash Recycling {En Línea} {29 de octubre 2021} Disponible en: https://www.researchgate.net/publication/326994757_Artificial_Intelligence_in_Automated_Sorting_in_Trash_Recycling

[38] YANAI, KEIJI. A food image recognition system with Multiple Kernel Learning. {En Línea} {29 de octubre 2021} Disponible en: https://www.researchgate.net/publication/326994757_Artificial_Intelligence_in_Automated_Sorting_in_Trash_Recycling

[39] Rahul Chauhan; Kamal Kumar Ghanshala; R.C Joshi. Convolutional Neural Network (CNN) for Image Detection and Recognition. {En Línea} {30 de octubre 2021} Disponible en: https://ieeexplore.ieee.org/abstract/document/8703316

[40] SATVILKAR, MANDAR. Image Based Trash Classification using Machine Learning Algorithms for Recyclability Status. {En Línea} {29 de octubre 2021} Disponible en: http://norma.ncirl.ie/3422/

[41] ADEDEJI, OLUGBOJA. WANG, ZENGHUI.Intelligent Waste Classification System Using Deep Learning Convolutional Neural Network. {En Línea} {30 de octubre 2021} Disponible en: https://www.sciencedirect.com/science/article/pii/S2351978919307231

[42] GUPTA, HIMANSHU. Trash Image Classification System using Machine Learning and Deep Learning Algorithms. {En Línea} {30 de octubre 2021} Disponible en: http://norma.ncirl.ie/4396/

[43] Alzyoud, Faisal; Maqableh, Waleed; Faiz Al Shrouf. A Semi Smart Adaptive Approach for Trash Classification {En Línea} {30 de octubre 2021} Disponible en: https://www.proquest.com/openview/98c370e8f0f1e762a54e72c4ba32a150/1?pq-origsite=gscholar&cbl=5045567

[44] Miko Pamintuan; Shiela Mae Mantiquilla; Hillary Reyes; Mary Jane Samonte. i-BIN: An Intelligent Trash Bin for Automatic Waste Segregation and Monitoring System. {En Línea} {30 de octubre 2021} Disponible en: https://ieeexplore.ieee.org/abstract/document/9072787

[45] Anh H. Vo; Le Hoang Son; Minh Thanh Vo, Tuong Lee. A Novel Framework for Trash Classification Using Deep Transfer Learning.{En Línea} {30 de octubre 2021} Disponible en: https://ieeexplore.ieee.org/abstract/document/8930948

[46] NA8. Qué es overfitting y underfitting y cómo solucionarlo..{En Línea} {1 Noviembre 2021} Disponible en: https://www.aprendemachinelearning.com/que-es-overfitting-y-underfitting-y-como-solucionarlo/#:~:text=Overfitting%20en%20Machine%20Learning,reconocer%20nuevos%20datos%20de%20entrada

[47] Intelligent, Machine Learning & NPL: cómo funciona un clasificador de documentos{En linea}{2 de noviembre 2021} disponible en: https://itelligent.es/es/pln-clasificacion-automatica-documentos/

[48] XERIDIA. Transformación de datos en Inteligencia Artificial. {En línea} {2 de noviembre de 2021} Disponible en: https://www.xeridia.com/blog/transformacion-de-datos-en-inteligencia-artificial