



**UNIVERSIDAD POLITÉCNICA SALESIANA**  
**SEDE GUAYAQUIL**  
**CARRERA DE ELECTRÓNICA Y AUTOMATIZACIÓN**

**CLASIFICACIÓN DE ENVASES USANDO COMPUTER VISION Y MACHINE  
LEARNING**

Trabajo de titulación previo a la obtención del  
Título de Ingeniero en Electrónica y Automatización

**AUTORES:** EDISON ALEXIS ESPIN PEÑA  
SASHA STEFANY SALGUERO ULLÓN

**TUTOR:** MSC. VICENTE AVELINO PEÑARANDA IDROVO

Guayaquil – Ecuador

2024


## CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN


Nosotros, Edison Alexis Espín Peña con documento de identificación N° 0927861484 y Sasha Stefany Salguero Ullón con documento de identificación N°0956390124, manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizo a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación

Guayaquil, 19 de marzo del año 2024.

Atentamente,

  
Edison Alexis Espín Peña  
092786148-4

  
Sasha Stefany Salguero Ullón  
095639012-4


## **CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA**


Nosotros, Edison Alexis Espín Peña con documento de identificación N° 0927861484 y Sasha Stefany Salguero Ullón con documento de identificación N°0956390124, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del Proyecto Técnico: “Clasificación de envases usando computer vision y machine learning”, el cual ha sido desarrollado para optar por el título de: Ingeniero en Electrónica y Automatización, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Guayaquil, 19 de marzo del año 2024.

Atentamente,

  
Edison Alexis Espín Peña  
092786148-4

  
Sasha Stefany Salguero Ullón  
095639012-4

## **CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN**

Yo, Vicente Avelino Peñaranda Idrovo con documento de identificación N°0916113426, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **CLASIFICACIÓN DE ENVASES USANDO COMPUTER VISION Y MACHINE LEARNING**, realizado por Edison Alexis Espín Peña con documento de identificación N° 0927861484 y Sasha Stefany Salguero Ullón con documento de identificación N°0956390124, obteniendo como resultado final el trabajo de titulación bajo la opción de Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Guayaquil, 19 de marzo del año 2024.

Atentamente,

A handwritten signature in blue ink, consisting of stylized, overlapping loops and lines, representing the name of the signatory.

Ing. Vicente Avelino Peñaranda Idrovo MSc.

## **DEDICATORIA**

*Gracias a mis padres, Edison y Jessica, quienes siempre estuvieron presentes con su apoyo incondicional y demostrándome su amor tan grande en cada decisión y paso que voy dando, con su fortaleza me han enseñado a no rendirme ante las adversidades que se me presentan. Lo que consiga y logre siempre será por y para ustedes. ¡Con amor!*

*Edison Espín P.*

*A Dios, por ser mi guía y fortaleza en cada paso.*

*A mi amada mamá Gina, por su amor incondicional y apoyo constante.*

*A mis docentes y tutor por su dedicación y enseñanzas fundamentales en mi educación.*

*Gracias a todos los que han sido parte de este viaje académico.*

*Sasha Salguero Ullón.*

## **AGRADECIMIENTO**

*A Dios por permitirme tener a una familia grandiosa a mi lado que siempre estuvo apoyándome en toda la carrera, a mis hermanas por siempre querer ayudarme a ser mejor en cada ámbito ya sea profesional o cotidiano, por ser mi fortaleza en los momentos más difíciles.*

*A mis padres, por siempre trabajar duro y enseñarme como debo ganarme los buenos resultados solo con trabajo y esfuerzo, sin todos ustedes no estaría aquí culminando lo que empecé.*

*Edison Espin P.*

*Agradezco a Dios por su guía y sabiduría, a mi amada mamá Gina Ullón por su amor, esfuerzo, dedicación y por los valores que me ha inculcado, a mis queridos docentes por su compromiso y apoyo. A Edison por su valiosa colaboración. También agradezco a todos quienes me han apoyado en este camino. Cada palabra de aliento y gesto de apoyo ha sido un faro en los momentos difíciles. Agradezco su presencia en este camino hacia la realización de mis sueños y superación personal.*

*Sasha Salguero Ullón.*

## RESUMEN

AÑO	ALUMNOS	TUTOR DEL PROYECTO	TEMA DEL PROYECTO
2024	Espín Peña Edison Alexis, Salguero Ullón Sasha Stefany	Ing. Vicente Avelino Peñaranda Idrovo MSc.	“CLASIFICACIÓN DE ENVASES USANDO COMPUTER VISION Y MACHINE LEARNING”

El principal objetivo de esta investigación es desarrollar un prototipo funcional para la clasificación automatizada de envases de vidrio, latas y plástico PET mediante visión por computadora y aprendizaje automático. El prototipo está diseñado para abordar el problema de la clasificación manual e ineficiente en el proceso de reciclaje, que contribuye al impacto ambiental negativo que causan los residuos sólidos. Para lograr este objetivo se entrena un modelo de aprendizaje automático para la detección y clasificación de contenedores mediante el software Teachable Machine. El modelo se integra con el hardware adecuado para diseñar y construir el prototipo de clasificación automatizada de envases. La validación del prototipo se realiza mediante clasificación automática de contenedores. Aunque este proyecto se limita a la creación de un prototipo funcional y está diseñado para implementarse a gran escala, se espera que los resultados obtenidos y las lecciones aprendidas puedan sentar las bases para investigaciones y desarrollos futuros en el campo de la gestión de residuos

Palabras Clave: Inteligencia artificial, clasificación automatizada, residuos sólidos, envases de vidrio, lata, plásticos PET, Computer Vision, Machine Learning, gestión de residuos, reciclaje, sostenibilidad ambiental.

## ABSTRACT

<b>YEAR</b>	<b>STUDENTS</b>	<b>PROJECT DIRECTOR</b>	<b>PROJECT TITLE</b>
2024	Espín Peña Edison Alexis, Salguero Ullón Sasha Stefany	Ing. Vicente Avelino Peñaranda Idrovo MSc.	"CONTAINER CLASSIFICATION USING COMPUTER VISION AND MACHINE LEARNING."

The main objective of this research is to develop a functional prototype for the automated classification of glass, metal, and PET plastic containers using computer vision and machine learning. The prototype will be designed to address the problem of manual and inefficient classification in the recycling process, which contributes to the negative environmental impact caused by solid waste. To achieve this goal, a machine learning model will be trained for container detection and classification using Teachable Machine software. The model will be integrated with suitable hardware to design and build the automated container classification prototype. The validation of the prototype will be conducted through automatic container classification.

Although this project is limited to the creation of a functional prototype and will not be implemented on a large scale, it is expected that the results obtained and the lessons learned can lay the groundwork for future research and developments in the field of waste management.

Keywords: Artificial intelligence, automated classification, solid waste, glass containers, metal, PET plastics, Computer Vision, Machine Learning, waste management, recycling, environmental sustainability.



## ÍNDICE DE CONTENIDO

INTRODUCCIÓN.....	1
<b>I. PROBLEMA .....</b>	<b>2</b>
<b>1.1. Antecedentes.....</b>	<b>2</b>
<b>1.2. Importancia y alcance .....</b>	<b>4</b>
<b>1.3. Delimitación.....</b>	<b>4</b>
<b>1.3.1. Delimitación espacial .....</b>	<b>4</b>
<b>1.3.2. Delimitación temporal.....</b>	<b>5</b>
<b>1.3.3. Delimitación académica .....</b>	<b>5</b>
<b>1.4. Objetivos.....</b>	<b>5</b>
1.1.1 Objetivo general.....	5
1.1.2 Objetivos específicos.....	5
<b>II. FUNDAMENTOS TEÓRICOS .....</b>	<b>6</b>
<b>2.1. Introducción al uso de inteligencia artificial en el proyecto .....</b>	<b>6</b>
<b>2.2. Inteligencia Artificial .....</b>	<b>6</b>
<b>2.3. Aprendizaje Automático .....</b>	<b>7</b>
<b>2.4. Visión Artificial .....</b>	<b>7</b>
<b>2.4.1. Visión por Computadora y Aprendizaje Automático .....</b>	<b>8</b>
<b>2.6. Antecedentes en Clasificación Automatizada de Envases.....</b>	<b>9</b>

2.6.1.	Investigaciones y Proyectos Previos .....	9
2.6.2.	Aplicaciones Industriales y Comerciales .....	10
2.7.	Avances en Tecnologías de Visión por Computadora.....	10
2.7.1.	Segmentación de Imágenes .....	10
2.8.	Conceptos Clave en Clasificación Automatizada de Envases .....	10
2.8.1.	Desafíos y Consideraciones .....	10
2.9.1.	Impacto Ambiental y Eficiencia Operativa.....	11
2.10.	Programas (Software) .....	11
2.10.1.	Teachable Machine .....	11
2.10.2.	Python.....	12
2.11.	Dispositivos (hardware) .....	12
2.11.1.	Raspberry Pi.....	13
2.11.2.	Cámara Web HD 1080p.....	13
2.11.3.	Tira de luces LED .....	14
2.11.4.	Servomotor .....	14
2.11.5.	Arduino Nano.....	14
III.	METODOLOGÍA .....	16
3.1.	Tipo de Investigación .....	16
3.2.	Diseño de la Investigación .....	16
3.3.	Población y Muestra.....	17

<b>3.4. Técnicas de Recolección de Datos</b> .....	17
<b>3.5. Procedimiento</b> .....	17
<b>3.5.1. Entrenamiento del modelo de Machine Learning:</b> .....	17
<b>3.5.1.1. Recopilación de Datos</b> .....	17
<b>3.5.1.2. Preparación del Modelo</b> .....	17
<b>3.5.1.3. Exportación del Modelo</b> .....	18
<b>3.5.2. Proyecto de imagen:</b> .....	18
<b>3.5.3. Diseño y construcción del prototipo:</b> .....	21
<b>3.5.4. Clasificación automatizada de envases:</b> .....	24
<b>3.6. Análisis de Datos</b> .....	24
<b>PRESUPUESTO</b> .....	27
<b>RESULTADOS</b> .....	28
<b>CONCLUSIONES</b> .....	34
<b>RECOMENDACIONES</b> .....	35
<b>4. ANEXO</b> .....	38

## Índice de Figuras

Figura 1.-Logo de Python con fondo de código (Python.org, 2024) .....	12
Figura 2.- Componentes de la placa Raspberry (Raspberrypi.cl, 2023) .....	13
Figura 3.-Servomotor (EPY Electrónica, 2023) .....	14
Figura 4.- Arduino Nano (Electronic Lab, 2024) .....	15
Figura 5.- Nuevo Proyecto Teachable Machine. (Teachable Machine, 2024).....	18
Figura 6.- Entrenamiento de Proyecto en Teachable Machine. (Teachable Machine, 2024) .....	18
Figura 7.- Entrenamiento de Proyecto en Teachable Machine.....	19
Figura 8.- Carga del modelo entrenamiento al script desarrollado (Programación en Python de los Autores, 2024) .....	20
Figura 9.- Se inicia la cámara web (Programación en Python de los Autores, 2024).....	20
Figura 10.- Armazón de madera del prototipo (Autores, 2024) .....	21
Figura 11.- Instalación de compuertas para clasificación de envases (Autores, 2024) .....	21
Figura 12.- Instalación de dispositivos para automatización de prototipo (Autores, 2024).....	22
Figura 13.- Instalación de luces LED .(Autores, 2024) .....	22
Figura 14.- Instalación de servomotores para compuertas automáticas. (Autores, 2024) .....	22
Figura 15.- Vista trasera del prototipo con las tres divisiones para envases de vidrio, plástico y metal. (Autores, 2024).....	23
Figura 16.- Vista superior en donde se observa la cámara web para la identificación automática de los envases. (Autores, 2024).....	23
Figura 17.- Vista frontal del prototipo. (Autores, 2024) .....	23
Figura 18.- Precisión por clase. (Autores, 2024) .....	24
Figura 19.- Matriz de composiciones. (Autores, 2024) .....	25

Figura 20.- Gráfica de pérdidas y épocas, a las 50 épocas obtiene una menor pérdida, lo que se traduce a mayor precisión. (Autores, 2024) .....26

Figura 21.- Identificación de vacío. (Autores, 2024).....26

Figura 22.- Identificación envase de plástico. (Autores, 2024).....26

Figura 23.- Identificación envase de vidrio. (Autores, 2024).....27

### **Índice de Tablas**

Tabla 1.- Presupuesto destinado para el desarrollo del Proyecto de Titulación .....27

Tabla 2.- Matriz de Confusión de Selección de Muestras del Proyecto .....29

## INTRODUCCIÓN

La gestión inadecuada de residuos sólidos es un desafío ambiental crítico a nivel mundial, con consecuencias negativas para el medio ambiente y la salud humana. Según un informe de la Organización Mundial de la Salud (OMS), la mala gestión de los residuos sólidos, incluida la falta de sistemas de clasificación y reciclaje eficaces, contribuye a la contaminación del aire, del agua y del suelo, lo que puede resultar en impactos adversos para la salud de la población, como enfermedades respiratorias, infecciones gastrointestinales y problemas dermatológicos (OMS, 2018).

En respuesta a esta problemática, la aplicación de tecnologías emergentes como la visión por computadora y el aprendizaje automático ha ganado prominencia en el desarrollo de sistemas automatizados para la clasificación de envases con fines de reciclaje. Estas tecnologías ofrecen la capacidad de identificar y separar eficientemente diferentes tipos de envases, como vidrio, latas y plástico PET, mejorando así la eficiencia de los procesos de reciclaje y reduciendo el impacto ambiental de los residuos (Hengstler, De Corte & Insua, 2019).

El enfoque central de este trabajo de titulación radica en la resolución de la problemática mencionada mediante el desarrollo de un prototipo funcional destinado a la clasificación automatizada de envases de vidrio, latas y plástico PET. Para lograr este objetivo, se emplearon técnicas avanzadas de visión por computadora y aprendizaje automático (Raut, Kumar & Singh, 2018). Estas tecnologías permiten analizar imágenes de manera automatizada, lo que posibilita la clasificación de variables esenciales mediante inspección visual, optimizando así el proceso de clasificación de envases.

El proceso de desarrollo del prototipo tiene de dos etapas principales. En la primera etapa, se lleva cabo el entrenamiento del modelo de aprendizaje automático utilizando el software Teachable Machine, con el objetivo de reconocer y clasificar los diferentes tipos de envases de acuerdo con su modelo. En la segunda etapa, se procede con el diseño y construcción del prototipo, integrando el software desarrollado con el hardware adecuado para la clasificación automatizada.

Si bien este proyecto se centra en el desarrollo de un prototipo funcional, se espera que sus resultados y conclusiones contribuyan al avance de la gestión de residuos sólidos y la implementación de prácticas más sostenibles de clasificación y reciclaje de envases, en línea con los objetivos de conservación ambiental y desarrollo sostenible.

## **I. PROBLEMA**

### **1.1. Antecedentes**

La gestión inadecuada de residuos sólidos es un desafío ambiental crítico a nivel mundial, con consecuencias negativas para el medio ambiente y la salud humana. Según un informe de la Organización Mundial de la Salud (OMS), la mala gestión de los residuos sólidos, incluida la falta de sistemas de clasificación y reciclaje eficaces, contribuye a la contaminación del aire, del agua y del suelo, lo que puede resultar en impactos adversos para la salud de la población, como enfermedades respiratorias, infecciones gastrointestinales y problemas dermatológicos (OMS, 2018).

En muchos países, esta falta de sistemas efectivos de gestión de residuos sólidos se traduce en la acumulación de desechos en vertederos abiertos, lo que representa un riesgo significativo para el medio ambiente y la salud pública. Por ejemplo, un estudio realizado por el Banco Mundial

estima que más de 2 mil millones de toneladas de residuos sólidos se generan anualmente en el mundo, de las cuales alrededor de un tercio no se gestionan de manera adecuada, terminando en vertederos no controlados o siendo incinerados en condiciones no reguladas (Banco Mundial, 2020).

Además, la falta de sistemas efectivos de clasificación y reciclaje de envases agrava este problema al no permitir la recuperación adecuada de materiales reciclables, como plástico, vidrio y metal. Según un informe de la Agencia Internacional de Energía (AIE), solo alrededor del 9% de los plásticos producidos en el mundo se reciclan, mientras que el resto se descarta en vertederos o se incinera, lo que representa una pérdida significativa de recursos y un aumento en la contaminación ambiental (AIE, 2019).

La gestión inadecuada de residuos sólidos es un desafío ambiental significativo a nivel global. En Ecuador, datos del Instituto Nacional de Estadística y Censos revelan que durante el 2021 se recolectaron diariamente en promedio 13,652.5 toneladas de residuos sólidos, de las cuales el 85.2% fueron recogidas de manera no diferenciada y el 14.8% de manera diferenciada (INEC, 2021). Esta falta de clasificación en la fuente limita la cantidad de residuos que se reciclan, contribuyendo a un aumento en la cantidad de residuos que terminan en vertederos. Además, la ineficiencia en la clasificación manual de envases reduce la tasa de reciclaje y la eficiencia del proceso, lo que resulta en una acumulación de residuos no reciclados y la pérdida de recursos valiosos.

Como consecuencia, se observa una contaminación ambiental significativa, incluida la acumulación de residuos en vertederos, la liberación de gases de efecto invernadero y la contaminación del suelo y el agua. Además, se producen pérdidas de recursos debido al desperdicio de materiales reciclables y al agotamiento de los recursos naturales, lo que también tiene un



impacto económico negativo, reflejado en los altos costos de gestión de residuos y la pérdida de oportunidades de negocio en el sector del reciclaje (Banco Mundial, 2020).

Ante esta problemática, es imperativo desarrollar soluciones tecnológicas innovadoras que mejoren la clasificación y el reciclaje de envases, así como fomentar una cultura del reciclaje en la población y la implementación de políticas públicas efectivas que promuevan prácticas sostenibles de gestión de residuos sólidos (UNEP, 2019).

## **1.2. Importancia y alcance**

La gestión adecuada de los residuos sólidos es crucial tanto a nivel local como global debido a sus impactos en el medio ambiente, la salud pública y la economía (UNEP, 2019). En el contexto ecuatoriano, la implementación de tecnologías innovadoras para la clasificación automatizada de envases es de suma importancia para abordar la problemática de la gestión inadecuada de los residuos sólidos (López-Maldonado et al., 2020). Permitiendo aumentar la eficiencia en el proceso de reciclaje, reducir la contaminación ambiental y fomentar una cultura del reciclaje en la población (Iriarte et al., 2018). A nivel global, países desarrollados ya han implementado tecnologías de clasificación automatizada, lo que demuestra el potencial y la viabilidad de dichas soluciones en el contexto ecuatoriano (García-Herrero et al., 2019).

## **1.3. Delimitación**

### **1.3.1. Delimitación espacial**

El presente proyecto ha sido desarrollado en el país Ecuador, en el cantón Guayaquil, periodo 2023-2024, para la obtención del título de Ingeniería en Electrónica y Automatización en la Universidad Politécnica Salesiana. El prototipo se procede a probar en los hogares de los autores.

### **1.3.2. Delimitación temporal**

El proyecto se ha desarrollado durante un período de 18 semanas, con una planificación detallada de actividades y hitos a lo largo de este período. Se incluyen etapas para el diseño, desarrollo, pruebas y evaluación del prototipo, así como para la recolección y análisis de datos.

### **1.3.3. Delimitación académica**

Este proyecto de titulación se enmarca en los requisitos para la obtención del título de Ingeniería en Electrónica en la Universidad Politécnica Salesiana, sede Guayaquil. La investigación se centra específicamente en el desarrollo de un prototipo funcional para la clasificación automatizada de envases de vidrio, latas y plástico PET, utilizando técnicas avanzadas de visión por computadora y aprendizaje automático.

El alcance académico de este trabajo abarca la aplicación de conocimientos teóricos y prácticos adquiridos durante la formación en ingeniería electrónica, incluyendo el diseño, la implementación y la evaluación de sistemas electrónicos para solucionar problemas específicos en el campo de la gestión de residuos sólidos.

## **1.4. Objetivos**

### **1.1.1 Objetivo general**

Desarrollar un prototipo funcional para la clasificación automatizada de envases según su tipo de material, usando Computer Vision y Machine Learning.

### **1.1.2 Objetivos específicos**

- Entrenar el modelo de aprendizaje para la detección y clasificación de envases considerando modelos específicos, como son: 3 de vidrio, 3 de lata y 3 de plástico usando el software Teachable Machine.

- Construir un prototipo para la clasificación automatizada de envases utilizando los modelos entrenados.
- Validar el prototipo diseñado mediante la clasificación automática de envases.

## **II. FUNDAMENTOS TEÓRICOS**

### **2.1.Introducción al uso de inteligencia artificial en el proyecto**

Las tecnologías de clasificación automática, como la visión por computadora y el aprendizaje automático, han demostrado ser eficaces en la identificación y separación de objetos en diversos campos, incluida la gestión de residuos. La visión por computadora utiliza algoritmos para procesar y analizar imágenes capturadas por sensores y cámaras, identificando características como forma, color y textura. El aprendizaje automático permite a las máquinas mejorar su rendimiento mediante la experiencia y la adaptación a diferentes situaciones. (EDS Robotics, 2022).

La aplicación de la inteligencia artificial (IA) en la clasificación automatizada de envases ha revolucionado la gestión de residuos, ofreciendo soluciones eficientes y precisas para identificar y separar materiales reciclables. Las tecnologías clave que impulsan esta innovación son la visión por computadora y el aprendizaje automático. (Russell & Norvig, 2021).

### **2.2.Inteligencia Artificial**

La disciplina de la inteligencia artificial (IA) engloba diversos enfoques, que se enfocan principalmente en abordar desafíos que resultan complicados para las computadoras, aunque parezcan sencillos para las personas. (Collins & Moons, 2019).

Para Ullah y otros autores, se plantea la existencia de múltiples aspectos algorítmicos en la resolución de problemas de aprendizaje dentro del ámbito del Machine Learning (ML), el cual surge a partir del estudio de la identificación de patrones en el contexto de la Inteligencia Artificial (IA). (Hoz y Fontalvo en 2019).

### **2.3. Aprendizaje Automático**

El aprendizaje automático, conocido en inglés como Machine Learning, (ML), es un subcampo de la inteligencia artificial (IA) que se centra en desarrollar algoritmos y técnicas que permiten a las computadoras aprender patrones y tomar decisiones basadas en datos, sin ser programadas explícitamente para realizar una tarea específica.

En esencia, el aprendizaje automático busca permitir que las máquinas mejoren automáticamente su rendimiento en una tarea a medida que se les proporcionan más datos y experiencia. (Google Cloud, 2023)

### **2.4. Visión Artificial**

La visión artificial o visión por computadora, conocida en inglés como "Computer Vision", se enfoca en la creación de sistemas informáticos dentro del ámbito de la inteligencia artificial, los cuales tienen la capacidad de interpretar y comprender información visual a partir de imágenes o vídeos. Su principal propósito es habilitar a las máquinas para observar y comprender su entorno de manera análoga a los seres humanos. Esto involucra el procesamiento y análisis de datos visuales para llevar a cabo diversas tareas, como reconocer objetos, identificar patrones, segmentar imágenes, detectar rostros, entre otras. (EDS Robotics, 2022)

Los sistemas de visión por computadora a menudo se respaldan en software de inteligencia artificial que emplea algoritmos matemáticos para analizar imágenes e identificar formas, figuras

y patrones, que posteriormente pueden utilizarse para activar acciones como la comparación, captura, manipulación o eliminación de elementos en una secuencia.

#### **2.4.1. Visión por Computadora y Aprendizaje Automático**

La clasificación automatizada de envases se basa en la integración de tecnologías como la visión por computadora y el aprendizaje automático. La visión por computadora emplea algoritmos para analizar imágenes capturadas por cámaras o sensores, identificando características clave como forma, color y textura de los envases. Por otro lado, el aprendizaje automático permite a los sistemas mejorar su rendimiento a través de la experiencia y la adaptación a diferentes situaciones, facilitando así la clasificación precisa y eficiente de envases.

#### **2.4.2. Procesamiento de Imágenes por convolución**

La convolución representa una operación matemática esencial en el ámbito del procesamiento de imágenes, fundamental para numerosos operadores comunes en este campo. Consiste en la multiplicación de dos matrices de números, usualmente de dimensiones diferentes, pero con la misma estructura dimensional (Cheng & Guo, 2017). Este proceso genera una tercera matriz de números con la misma dimensionalidad. En el contexto del procesamiento de imágenes, la primera matriz suele corresponder a la imagen original, mientras que la segunda, conocida como kernel, es generalmente mucho más pequeña y bidimensional, aunque puede tener solo un píxel de grosor.

### **2.5. Machine Learning**

Machine Learning (ML) se define como una rama de la inteligencia artificial (IA) que otorga a los sistemas la capacidad de adquirir conocimientos y mejorar de forma automática a partir de la experiencia, sin requerir una programación explícita (Alpaydin, 2020). El enfoque del aprendizaje automático se concentra en el desarrollo de programas informáticos capaces de acceder

a datos y emplearlos para aprender de manera autónoma. El proceso de aprendizaje inicia con la observación de datos, tales como ejemplos, experiencias directas o instrucciones, con el propósito de identificar patrones y tomar decisiones más precisas en el futuro basadas en dichos ejemplos. El objetivo fundamental consiste en permitir que las computadoras aprendan de manera autónoma sin requerir intervención o asistencia humana, adaptando sus acciones de acuerdo con el aprendizaje adquirido.

Una perspectiva más amplia sobre el Machine Learning, según Alpaydin (2010), considera que ciertos procesos también incorporan el aprendizaje por refuerzo (Sutton & Barto, 1998), el cual guarda una estrecha relación con el aprendizaje realizado por entidades biológicas inteligentes. A continuación, el tipo de algoritmo de aprendizaje que se utiliza en el proyecto:

**Algoritmos de aprendizaje automático supervisados:** estos algoritmos son capaces de aplicar conocimientos adquiridos en el pasado a nuevos datos utilizando ejemplos etiquetados para predecir eventos futuros. A través del análisis de un conjunto de datos de entrenamiento conocido, el algoritmo de aprendizaje genera una función inferida para realizar predicciones sobre los valores de salida. Después de un entrenamiento suficiente, el sistema puede establecer objetivos para cualquier nueva entrada. Además, el algoritmo de aprendizaje puede comparar su salida con la salida prevista correcta y corregir errores para ajustar el modelo en consecuencia (Rodríguez & Camacho, 2018).

## **2.6. Antecedentes en Clasificación Automatizada de Envases**

### **2.6.1. Investigaciones y Proyectos Previos**

Investigaciones previas, como la realizada por Smith et al. (2020), han demostrado la eficacia de los algoritmos de aprendizaje profundo en la clasificación automatizada de envases. En su estudio, Smith et al. lograron una precisión del 95% en la identificación de diferentes tipos de

envases de plástico en una planta de reciclaje, lo que resalta el potencial de estas tecnologías en la gestión de residuos.

### **2.6.2. Aplicaciones Industriales y Comerciales**

Investigaciones previas han demostrado la eficacia de los algoritmos de aprendizaje profundo en la clasificación automatizada de envases. Por ejemplo, en un estudio realizado por la Universidad Nacional de Colombia, se implementó un sistema de clasificación automatizada de envases en una planta de reciclaje en Bogotá. Este sistema logró un aumento del 25% en la eficiencia de separación de materiales reciclables y una reducción del 15% en los residuos no deseados (Gómez et al., 2019).

## **2.7. Avances en Tecnologías de Visión por Computadora**

### **2.7.1. Segmentación de Imágenes**

Avances en técnicas de segmentación de imágenes, como el trabajo realizado por Liu et al. (2019), han mejorado la precisión de los sistemas de clasificación automatizada de envases. Liu et al. desarrollaron un algoritmo basado en redes neuronales convolucionales que logró una precisión del 98% en la identificación de envases de vidrio en entornos industriales, demostrando así la efectividad de estas técnicas en aplicaciones prácticas.

## **2.8. Conceptos Clave en Clasificación Automatizada de Envases**

### **2.8.1. Desafíos y Consideraciones**

La clasificación automatizada de envases enfrenta desafíos debido a la diversidad de formas, tamaños y materiales de los envases. Estos desafíos requieren algoritmos robustos y flexibles capaces de adaptarse a diferentes condiciones y tipos de envases para lograr una clasificación precisa y eficiente.

## **2.9.Ejemplos de Aplicaciones y Beneficios**

### **2.9.1. Impacto Ambiental y Eficiencia Operativa**

Estudios como el realizado por García et al. (2021) han destacado el impacto positivo de los sistemas de clasificación automatizada de envases en la gestión de residuos y el reciclaje. García et al. encontraron que la implementación de estos sistemas resultó en una reducción significativa de los residuos enviados a vertederos y una mayor tasa de reciclaje de materiales, lo que demuestra su efectividad en la promoción de prácticas de reciclaje sostenibles y la reducción del impacto ambiental.

### **2.10. Programas (Software)**

El proyecto se fundamenta en dos pilares fundamentales: en primer lugar, se encuentra el Machine Learning, que permite entrenar a un sistema dentro de una red neuronal para identificar el material del envase de acuerdo a su modelo. Para este propósito, se emplea el software Teachable Machine, que se basa en el aprendizaje automático. Esta función posibilita instruir y orientar al prototipo según las necesidades del usuario. En segundo lugar, se emplea la visión artificial, utilizando las bibliotecas Tensorflow, Numpy y Pil como base. Por lo tanto, el lenguaje de programación para desarrollar el script es Python y Arduino.

#### **2.10.1. Teachable Machine**

Teachable Machine es una plataforma en línea desarrollada por Google que simplifica el proceso de entrenamiento de modelos de aprendizaje automático. Permite entrenar modelos de aprendizaje automático de manera accesible y sin necesidad de programación o algoritmos complejos (Teachable Machine, 2023).

Para trabajar con la plataforma mencionada, se deben seguir varios procesos, que incluyen:

**Recopilación:** en esta etapa inicial, se recopila y se agrupa la información que se desea que el ordenador aprenda.



**Preparación:** una vez que los datos han sido recopilados, se prepara el modelo y se realiza una comprobación para determinar si la clasificación y la recolección se han realizado correctamente.

**Exportación:** finalmente, el modelado del proyecto se puede exportar de manera gratuita y en línea.

### 2.10.2. Python

Es un lenguaje de programación ampliamente utilizado en el campo del aprendizaje automático y la visión por computadora. Es necesario para desarrollar scripts y programas que integren el prototipo con el hardware, procesen datos y realicen operaciones de control. (Python.org, 2023)



**Figura 1.-Logo de Python con fondo de código (Python.org, 2024)**

### 2.10.3. Arduino

Arduino es una plataforma de hardware de código abierto que está diseñada para facilitar la creación de prototipos de proyectos electrónicos. Consiste en una placa de circuito impreso con un microcontrolador y un entorno de desarrollo integrado (IDE) que se utiliza para escribir y cargar el software en la placa. (Arduino.cc, 2024)

## 2.11. Dispositivos (hardware)

El término hardware se refiere al conjunto de componentes físicos (materiales) que constituyen un dispositivo, como se explicó en el trabajo de Díaz Herrera (2012).

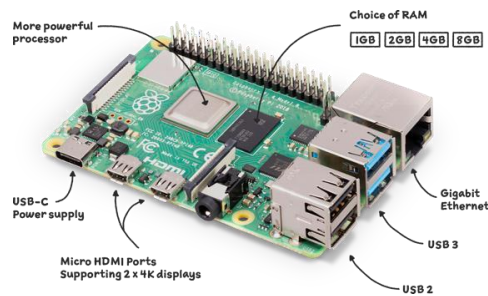
En este contexto, se utilizan dispositivos fundamentales como la Raspberry Pi, la cámara web HD, las matrices de LED, un motor servo, sensor infrarrojo, como componentes esenciales en la creación del prototipo

### 2.11.1. Raspberry Pi

Es una serie de computadoras de placa única (SBC, por sus siglas en inglés) de bajo costo y alto rendimiento desarrolladas por la Fundación Raspberry Pi. Estas pequeñas computadoras están diseñadas para ser asequibles, versátiles y accesibles, lo que las hace ideales para proyectos de electrónica, programación, aprendizaje automático y más. (Raspberrypi.cl, 2023)

El uso de la placa Raspberry Pi es importante en este proyecto debido a su capacidad para trabajar con ciertas librerías de Machine Learning y Computer Vision en Python. Esta placa permite gestionar la parte lógica del proyecto y controlar los dispositivos necesarios, como cámaras y sensores, esenciales para la clasificación exitosa de los envases.

En la Figura 2, se tiene una imagen de la tarjeta Raspberry Pi y sus componentes.



**Figura 2.- Componentes de la placa Raspberry (Raspberrypi.cl, 2023)**

### 2.11.2. Cámara Web HD 1080p

Una cámara web es un dispositivo de entrada de video que se conecta a una computadora o sistema informático y se utiliza para capturar imágenes y videos en tiempo real. Estas cámaras, también conocidas como cámaras de PC o cámaras de red, son dispositivos compactos y versátiles que se utilizan comúnmente para videoconferencias, transmisión en vivo, grabación de videos caseros, entre otros fines.

Es importante el uso de esta cámara ya que se integra con el Raspberry para capturar para adquirir imágenes de los envases y transmitir las a la plataforma Teachable Machine.

### **2.11.3. Tira de luces LED**

Las tiras leds son un sistema de iluminación que está montada sobre un circuito impreso muy flexible (Grealtec, s.f.). En este caso se utiliza para que la cámara pueda obtener un mejor contraste entre el fondo y el envase a clasificar.

### **2.11.4. Servomotor**

Un servomotor, también conocido como motor servo, es un tipo particular de motor diseñado para controlar la posición de su eje en un momento dado. Otra característica fundamental de los servomotores utilizados en modelismo son su capacidad de funcionar eficazmente con tensiones muy bajas, ya que suelen requerir tan solo entre 4 y 6 voltios para operar adecuadamente (García González, 2016).



**Figura 3.-Servomotor (EPY Electrónica, 2023)**

### **2.11.5. Arduino Nano**

Arduino Nano es una placa de desarrollo de hardware de código abierto basada en el microcontrolador ATmega328P. Es una versión compacta y económica de la popular plataforma Arduino, diseñada para proyectos donde el espacio y el costo son limitados. (Robotics.org, 2023)

La placa Arduino Nano ofrece características similares a otras placas Arduino, incluyendo entradas y salidas digitales y analógicas, interfaces de comunicación serial, conexión USB para programación y alimentación, y capacidad de expansión mediante la conexión de módulos y sensores adicionales. (Robotics.org, 2023)



**Figura 4.- Arduino Nano (Electronic Lab, 2024)**

## **2.12. Lenguajes de programación**

Los Lenguajes de Programación se definen como conjuntos de léxicos, sintaxis y semánticas que permiten diseñar algoritmos en un ordenador, facilitando la escritura de secuencias lógicas de órdenes para llevar a cabo actividades específicas (Informática IV).

Python, un lenguaje de programación de alto nivel, incorpora implícitamente ciertas funciones como listas, diccionarios, tuplas y conjuntos, lo que posibilita la ejecución de tareas complejas de manera legible y con pocas líneas de código. Creado por Guido Van Rossum a finales de los años 80 y principios de los 90, mientras desarrollaba el sistema operativo Amoeba, Python ha sido ampliamente aceptado. La comunidad de Python ha establecido una serie de prácticas recomendadas que deben seguirse al programar en este lenguaje (Challenger-Pérez, Díaz-Ricardo, & Becerra-García, 2014).

Entre los puntos destacados de Python está su amplia librería estándar y la facilidad para incorporar numerosos scripts mediante el uso del comando `pip install nombredelalibrería` en el terminal, o `pip3 install nombredelalibrería` si se usa la versión 3 de Python. Algunas de las librerías comúnmente utilizadas incluyen:

Tkinter, que permite el desarrollo de interfaces gráficas de usuario de manera nativa y eficiente (Python, 2021).

PIL (Pillow), una popular biblioteca para el procesamiento de imágenes que admite una variedad de formatos y técnicas de procesamiento (Recursos Python, s.f.).

Time, que proporciona funciones relacionadas con el tiempo para diversas aplicaciones (Python, 2021).

Rpi.gpio, que permite controlar y gestionar los puertos GPIO en la Raspberry Pi para proyectos que requieren módulos externos o sensores (Pypi, 2019).

TensorFlow, desarrollado por Google, que facilita el cálculo numérico mediante diagramas de flujo de datos, especialmente en aplicaciones de Deep Learning (IArtificial, 2020).

NumPy, que ofrece capacidades para análisis de datos y operaciones matemáticas de alto nivel (IArtificial, 2020).

Picamera, diseñado para la gestión de la cámara Raspberry Pi v2 en sistemas operativos basados en Debian (IArtificial, 2020).

### **III. METODOLOGÍA**

En esta sección se describe la metodología que se siguió para alcanzar los objetivos planteados en el presente proyecto basado en desarrollar un prototipo funcional para la clasificación automatizada de envases de vidrio, lata y plásticos PET, usando Computer Vision y Machine Learning.

#### **3.1. Tipo de Investigación**

Se llevó a cabo una investigación aplicada, que se basa en la aplicación práctica de conocimientos teóricos y técnicos para el desarrollo del prototipo de clasificación automatizada de envases.

#### **3.2. Diseño de la Investigación**

El diseño de la investigación es experimental, ya que se realizan pruebas y experimentos para entrenar y validar el modelo de aprendizaje, así como para evaluar el prototipo en condiciones reales.

### **3.3. Población y Muestra**

La población objetivo para este proyecto está compuesta por 3 modelos de envases de vidrio, 3 de lata y 3 plásticos PET. Se utilizan muestras representativas de estos envases para el entrenamiento y prueba del modelo.

### **3.4. Técnicas de Recolección de Datos**

Se utilizan técnicas de recolección de datos como la captura de imágenes de los envases, la anotación de datos para el entrenamiento del modelo, y pruebas de clasificación automatizada con el prototipo.

### **3.5. Procedimiento**

#### **3.5.1. Entrenamiento del modelo de Machine Learning:**

1. Recopilación de imágenes de envases de vidrio, lata y plásticos PET.
2. Clasificación de las imágenes con las etiquetas correspondientes.
3. Utilización del software Teachable Machine para entrenar el modelo

Este software se utiliza para recopilar, preparar y exportar datos de entrenamiento para el modelo de clasificación automatizada de envases.

Se incurren en diferentes procesos, como se detalla:

#### **3.5.1.1. Recopilación de Datos**

La fase inicial implica recopilar y organizar la información necesaria para que la computadora aprenda. En el caso de este proyecto, se recopilaron imágenes para la clasificación de envases considerando modelos específicos, como son: 3 de vidrio, 3 de lata y 3 de plástico usando el software Teachable Machine, con el fin de entrenar al modelo de aprendizaje automático.

#### **3.5.1.2. Preparación del Modelo**

Una vez recopiladas las imágenes, se procede a preparar el modelo en Teachable Machine. Esto incluye etiquetar las imágenes con las categorías correspondientes (envases de vidrio, metal,

plástico PET, vacío y otros elementos) y realizar pruebas para asegurar la precisión del modelo en la clasificación de los envases.

### 3.5.1.3. Exportación del Modelo

Una vez completado el entrenamiento y la preparación del modelo, este se exporta a un formato compatible para su uso en Python. Esto permite integrar el modelo de clasificación automatizada de envases con el hardware necesario para la implementación del prototipo.

### 3.5.2. Proyecto de imagen:

Proyecto de imagen permite realizar la preparación con imágenes de archivos o de la webcam. A continuación, se detalla el entorno del Software Teachable Machine en un proyecto nuevo y el entrenamiento del modelo de aprendizaje automático.

**Clase (Class):** Son las categorías en las que se dividen las imágenes. Se han usado cinco clases con sus etiquetas correspondientes: vidrio, plástico, metal, vacío y objeto incorrecto..

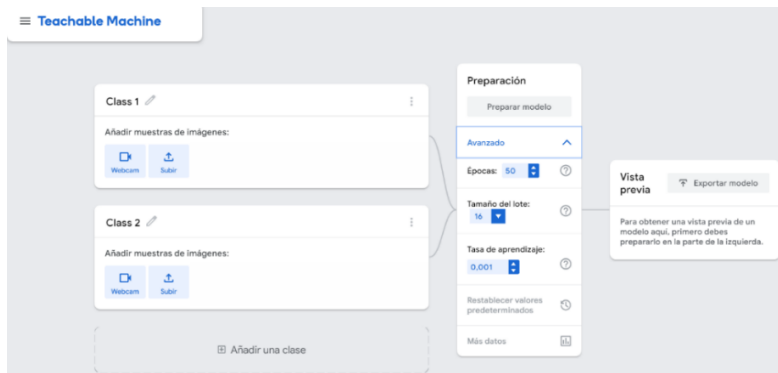


Figura 5.- Nuevo Proyecto Teachable Machine. (Teachable Machine, 2024)

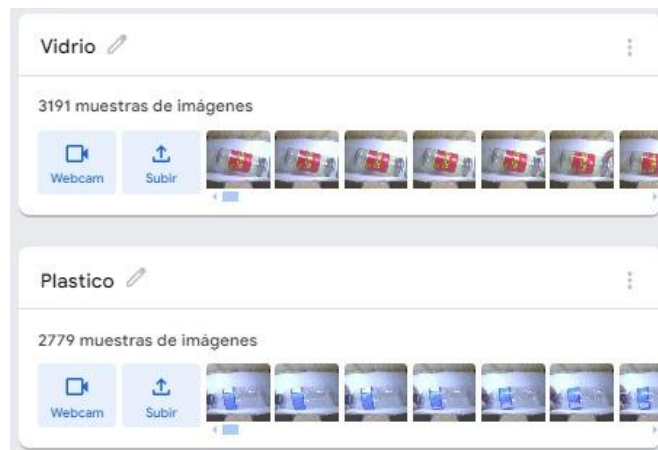
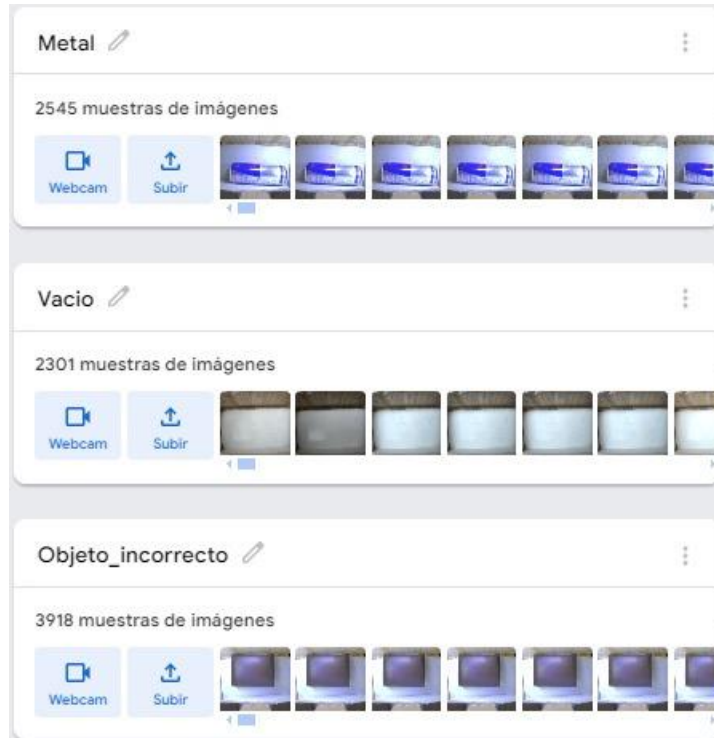


Figura 6.- Entrenamiento de Proyecto en Teachable Machine. (Teachable Machine, 2024)



**Figura 7.- Entrenamiento de Proyecto en Teachable Machine.**

**Épocas (Epoch):** Cada vez que el modelo completa una ronda de estudio de todas las imágenes, se considera una época.

**Tamaño del lote (batch\_size):** Este parámetro determina cuántas imágenes se agrupan para que el modelo las estudie juntas. Por lo general, no tiene un impacto significativo en el funcionamiento del modelo.

**Tasa de aprendizaje (learning\_rate):** La tasa de aprendizaje es una variable crucial que influye en la capacidad de aprendizaje del modelo. Indica cuánto el sistema puede ajustar sus aproximaciones en cada iteración. Un valor bajo conduce a un aprendizaje gradual, pero puede ser lento si el modelo no encuentra la dirección correcta desde el principio. Por otro lado, un valor alto acelera el aprendizaje, pero puede resultar en un proceso más inestable, donde el modelo cambia de dirección con frecuencia, perdiendo parte de lo aprendido.



Luego de la configuración de todas estas funciones y parámetros en Teachable Machine, se procede a entrenar la red. Una vez completado este proceso, se guarda el modelo en un único archivo en formato Keras (.h5). Sin embargo, para utilizar este modelo de red clasificadora de imágenes de envases (vidrio, plástico, metal, vacío, objetos incorrectos) en TensorFlow, es necesario realizar una conversión. Esta conversión implica la obtención de dos archivos distintos del modelo en formato TensorFlow: uno en formato binario y otro en formato de texto. Para lograr esta conversión, se debe crear un programa en Python capaz de transformar el archivo original en formato Keras (.h5) al formato binario de TensorFlow para lo cual se usa una línea de código en Python.

```
model = tensorflow.keras.models.load_model('keras_modelultima.h5');
```

**Figura 8.- Carga del modelo entrenamiento al script desarrollado (Programación en Python de los Autores, 2024)**

2. Ya cargado el modelo al script se procede a capturar video obtenido desde la cámara mediante la función `video.capture`, como se lo indica en la Figura 9.

```
# Ruta donde se aloja modelo descargado
model_path = f"{DIR_PATH}/keras10/keras_model.h5"

# Cargar modelo de Teachable Machine
model = tf.keras.models.load_model(model_path, compile=False)

# Iniciar cámara web
cap = cv2.VideoCapture(0)

# Ancho y Alto de la cámara web en píxeles -> Ajustar dependiendo de la resolución de la cámara
frameWidth = 640 #1080x720
frameHeight = 480

# Definir ancho y alto en píxeles
cap.set(cv2.CAP_PROP_FRAME_WIDTH, frameWidth)
cap.set(cv2.CAP_PROP_FRAME_HEIGHT, frameHeight)

# Habilitar ganancia autom.
cap.set(cv2.CAP_PROP_GAIN, 0)

# Mantiene programa en bucle - Cerrar Ctrl+c o cerrar ventana
while True:
    # Desactivar notación científica
    np.set_printoptions(suppress=True)

    # Crear matriz para alimentar modelo de Keras.
    # Ingresando imagen RGB de 1x224x224 píxeles.
    data = np.ndarray(shape=(1, 224, 224, 3), dtype=np.float32)
```

**Figura 9.- Se inicia la cámara web (Programación en Python de los Autores, 2024)**

### 3.5.3. Diseño y construcción del prototipo:

1. Selección de hardware adecuado para la clasificación automatizada, esto ya se lo detalló en la sección del marco teórico.
2. Integración del software de Machine Learning con el hardware, esto ya se explicó en la sección anterior.
3. Implementación del prototipo.

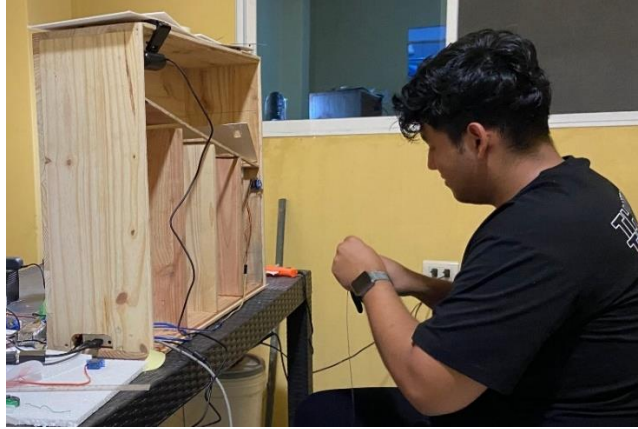
A continuación, con imágenes se observa el proceso de implementación del prototipo



**Figura 10.- Armazón de madera del prototipo (Autores, 2024)**



**Figura 11.- Instalación de compuertas para clasificación de envases (Autores, 2024)**



**Figura 12.- Instalación de dispositivos para automatización de prototipo (Autores, 2024)**



**Figura 13.- Instalación de luces LED .(Autores, 2024)**



**Figura 14.- Instalación de servomotores para compuertas automáticas. (Autores, 2024)**



**Figura 15.- Vista trasera del prototipo con las tres divisiones para envases de vidrio, plástico y metal. (Autores, 2024)**



**Figura 16.- Vista superior en donde se observa la cámara web para la identificación automática de los envases. (Autores, 2024)**



**Figura 17.- Vista frontal del prototipo. (Autores, 2024)**

Como se podrá observar, existe un compartimento donde se ingresa el envase a identificar con la cámara web, en donde se encuentran las tiras led para una mejor captura de imágenes, un sistema de transmisión de luz y módulos diseñados para la clasificación de los tres tipos de envases. Si no hay objeto, se identifica como vacío y no abre la primera compuerta, si se ingresa un objeto no identificado la primera compuerta permanece cerrada. Si se identifica el tipo de envase, se abre la primera compuerta y la compuerta correspondiente al tipo de envase donde el objeto se desplaza hasta que ingresa al cubículo correspondiente, ya sea vidrio, plástico o metal.

#### 3.5.4. Clasificación automatizada de envases:

Se inserta el envase en el prototipo, la cámara Raspberry captura imágenes y en base al script elaborado en Python identifica y clasifica el tipo de envase.

Pruebas de clasificación utilizando el prototipo.

Evaluación de la precisión y eficacia del sistema.

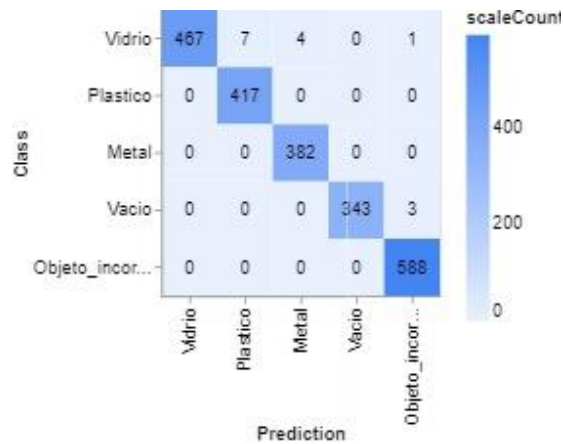
#### 3.6. Análisis de Datos

Se analizan los resultados obtenidos durante las pruebas de clasificación automatizada para determinar la precisión del modelo y la efectividad del prototipo.



CLASS	ACCURACY	# SAMPLES
Vidrio	0.97	479
Plastico	1.00	417
Metal	1.00	382
Vacio	0.99	346
Objeto_incorrecto	1.00	588

Figura 18.- Precisión por clase. (Autores, 2024)



**Figura 19.- Matriz de composiciones. (Autores, 2024)**

Para corroborar lo obtenido de los indicadores de predicción y perdida se verifica la matriz de confusiones se aplicó esta herramienta que permite la visualización del desempeño del entrenamiento que se emplea en el aprendizaje supervisado del modelo, la Figura 18 presenta dicha matriz, donde se evidencia que el modelo no confunde ninguna de las clases plateadas (vidrio, plástico, metal, vacío, objeto incorrecto) esto significa que el modelo es un buen clasificador de envases de acuerdo a su tipo.

Además de la precisión como validador para el entrenamiento del modelo, se tiene las pérdidas por época, este indicador mide el nivel de aprendizaje del modelo propuesto a la hora de predecir la clasificación correcta de los envases según su tipo y modelo, por ende, la Figura 20 presenta el nivel de pérdida del modelo el cual arroja un valor 0.0235 siendo un valor no representativo y cercano a cero.



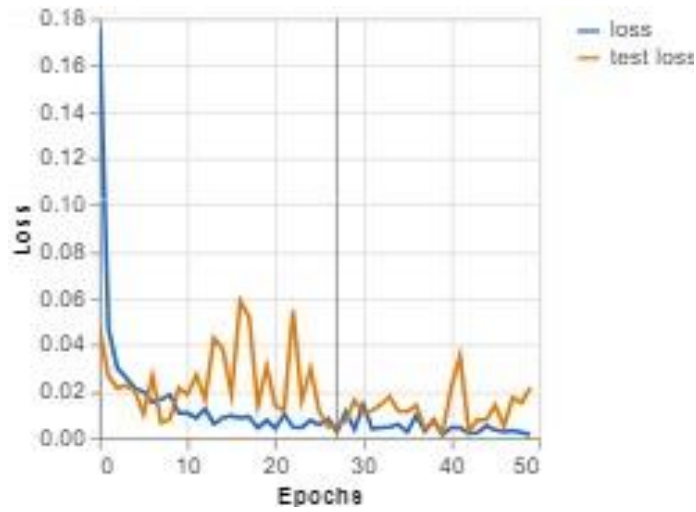


Figura 20.- Gráfica de pérdidas y épocas, a las 50 épocas obtiene una menor pérdida, lo que se traduce a mayor precisión. (Autores, 2024)

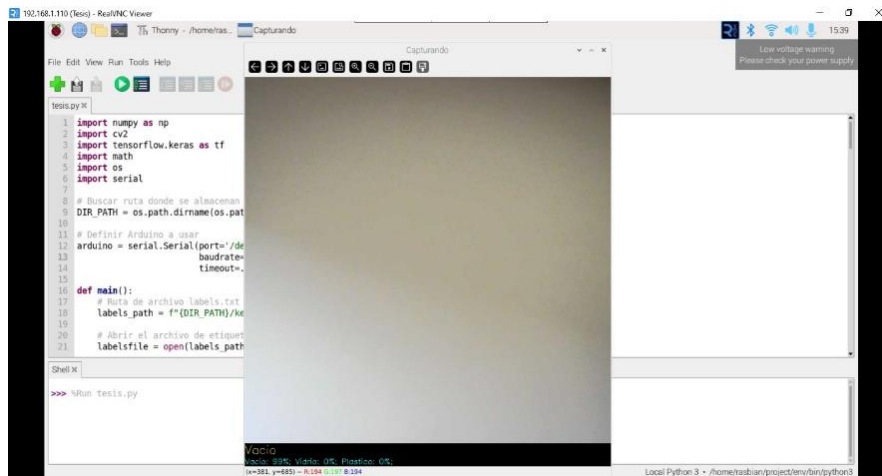


Figura 21.- Identificación de vacío. (Autores, 2024)

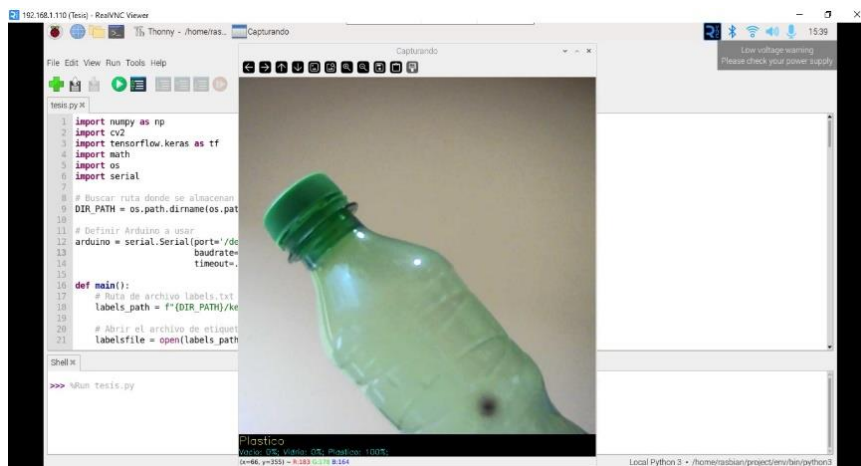
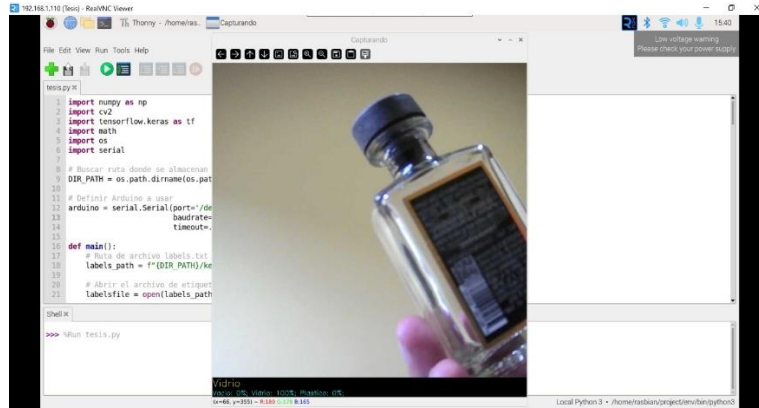


Figura 22.- Identificación envase de plástico. (Autores, 2024)



**Figura 23.- Identificación envase de vidrio. (Autores, 2024)**

## PRESUPUESTO

Considerando que en el presente proyecto se realizó un prototipo, dentro del presupuesto se ha considerado las horas de trabajo de ingeniería invertidas en el desarrollo de esta investigación, así como útiles de trabajo requerido para la culminación del proyecto, y los materiales utilizados, valores que se detallan en la Tabla 1, sumando un valor de \$ 761,50.

**Tabla 1.- Presupuesto destinado para el desarrollo del Proyecto de Titulación**

Cant.	Detalle	Valor unit	Valor total
150	Horas ingeniería	de \$ 3.00	\$450.00
1	Material trabajo	de \$ 10.00	\$ 10.00
1	Raspberry Pi	\$156.00	\$156.00
3	Servomotores	\$ 5.00	\$ 15.00
1	Cámara web HD	\$ 16.50	\$ 16.50
4	Jumpers tiras	\$ 1.00	\$ 4.00
1	Maqueta	\$ 60.00	\$ 60.00
1	Articulos varios	\$ 50.00	\$ 50.00
	<b>Valor total</b>		<b>\$761.50</b>



## RESULTADOS

Como prueba de la precisión del modelo clasificador automático de envases se empleó la matriz de confusión. A continuación, se explica qué es, para que sirve la matriz de confusión y por qué se la emplea en el presente proyecto.

La matriz de confusión es una herramienta esencial en la evaluación de modelos de clasificación (Fawcett, 2006). En el caso del modelo clasificador automático de envases, su utilidad radica en proporcionar una visión detallada del rendimiento del modelo al clasificar muestras en diferentes clases.

Esta matriz permite identificar y contar los Verdaderos Positivos (VP), Falsos Positivos (FP), Verdaderos Negativos (VN) y Falsos Negativos (FN) para cada clase, lo que facilita el análisis del desempeño del modelo (Provost & Fawcett, 2013). Los VP indican cuántas muestras fueron correctamente clasificadas como pertenecientes a una clase específica, los VN indican cuántas muestras fueron correctamente clasificadas como no pertenecientes a esa clase, mientras que los FP y FN revelan los errores de clasificación del modelo.

Con esta información detallada, los evaluadores pueden analizar la precisión del modelo, su capacidad para identificar correctamente muestras positivas y negativas, y su habilidad para evitar clasificar erróneamente muestras negativas (Sokolova & Lapalme, 2009). Esto permite tomar decisiones informadas sobre la efectividad del modelo y posibles mejoras.

Para evaluar la efectividad del modelo clasificador propuesto, se llevaron a cabo pruebas específicas utilizando tres tipos de modelos distintos para cada tipo de material. Esto incluyó tres modelos de vidrio, tres de lata y tres de plástico. Cada uno de estos modelos fue sometido a pruebas con un total de 10 muestras individuales, lo que suma un total de 30 muestras por cada tipo de

material. Además de estas pruebas, se incluyeron cinco muestras adicionales de objetos no relacionados.

A continuación, en la Tabla 2 se observa la matriz de confusión basada en pruebas realizadas.

**Tabla 2.- Matriz de Confusión de Selección de Muestras del Proyecto**

Observación	Predicción					
	Vidrio		Plástico		Metal	
	Positivos	Negativos	Positivos	Negativos	Positivos	Negativos
Positivos	25	5	27	5	29	4
Negativos	5	0	3	0	3	0

Para las observaciones de vidrio, de 35 muestras, 26 fueron correctamente clasificadas como vidrio (Verdaderos Positivos - VP) y 4 fueron incorrectamente clasificadas como plástico (Falsos Positivos - FP), y los 5 corresponden a los otros objetos que fueron correctamente clasificados como no pertenecientes a esa clase (Verdaderos Negativos - VN).

Para las observaciones de plástico, de 35 muestras, 27 fueron correctamente clasificadas como vidrio (Verdaderos Positivos - VP) y 3 fueron incorrectamente clasificadas como plástico (Falsos Positivos - FP), y los 5 corresponden a los otros objetos que fueron correctamente clasificados como no pertenecientes a esa clase (Verdaderos Negativos - VN).

Para las observaciones de plástico, de 35 muestras, 29 fueron correctamente clasificadas como vidrio (Verdaderos Positivos - VP) y 3 fueron incorrectamente clasificadas como plástico (Falsos Positivos - FP), y los 4 corresponden a los otros objetos que fueron correctamente clasificados como no pertenecientes a esa clase (Verdaderos Negativos - VN).

La tabla de confusión refleja correctamente los resultados del modelo de clasificación para cada tipo de envase en función de las observaciones proporcionadas, diferenciando claramente entre las observaciones y las predicciones.

## Resultados de Desempeño del Modelo:

Se calcularon varios indicadores para evaluar el desempeño del modelo clasificador automático de envases:

### Exactitud (Accuracy):

La exactitud representa la proporción de predicciones correctas realizadas por el modelo con respecto al número total de predicciones. Se calculó para cada clase de envase de la siguiente manera:

$$\text{Exactitud} = \frac{\text{Verdaderos Positivos (VP)} + \text{Verdaderos Negativos (VN)}}{\text{Total de Muestras}}$$

$$\text{Exactitud vidrio} = \frac{(VP_{\text{vidrio}}) + (VN_{\text{vidrio}})}{\text{Total de Muestras}}$$

$$\text{Exactitud vidrio} = \frac{(25) + (5)}{35}$$

*Exactitud vidrio = 0,8571; expresado en porcentajes sería 85.71%*

$$\text{Exactitud plástico} = \frac{(VP_{\text{plástico}}) + (VN_{\text{plástico}})}{\text{Total de Muestras}}$$

$$\text{Exactitud plástico} = \frac{(27) + (5)}{35}$$

*Exactitud plástico = 0,9143; expresado en porcentajes sería 91.43%*

$$\text{Exactitud metal} = \frac{(VP_{\text{metal}}) + (VN_{\text{metal}})}{\text{Total de Muestras}}$$

$$\text{Exactitud metal} = \frac{(29) + (4)}{35}$$

*Exactitud metal = 0,9428; expresado en porcentajes sería 94.28%*

$$\text{Promedio de Exactitud} = \frac{(85.71 + 91.43 + 94.28)\%}{3}$$

$$\text{Promedio de Exactitud} = \frac{271.42}{3}$$

$$\text{Promedio de Exactitud} = 90.47\%$$

### **Tasa de error (Error Rate):**

La tasa de error se define como el cociente entre el número total de predicciones incorrectas y el número total de muestras.

Para calcular la tasa de error, primero se suman los falsos positivos (FP) y los falsos negativos (FN) para cada clase, y luego se divide esta suma por el número total de muestras.

La fórmula general para calcular la tasa de error es:

$$\text{Tasa de error} = \frac{\text{Falsos Positivos (FP)} + \text{Falsos Negativos (FN)}}{\text{Total de Muestras}}$$

$$\text{Tasa de error vidrio} = \frac{(FP_{\text{vidrio}}) + (FN_{\text{vidrio}})}{\text{Total de Muestras}}$$

$$\text{Tasa de error vidrio} = \frac{4}{35}$$

*Tasa de error vidrio = 0,114; expresado en porcentajes sería 11.40%*

$$\text{Tasa de error plástico} = \frac{(FP_{\text{plástico}}) + (FN_{\text{plástico}})}{\text{Total de Muestras}}$$

$$\text{Tasa de error plástico} = \frac{3}{35}$$

*Tasa de error plástico = 0,086; expresado en porcentajes sería 8.60%*

$$Tasa\ de\ error\ metal = \frac{(FP_{metal}) + (FN_{metal})}{Total\ de\ Muestras}$$

$$Tasa\ de\ error\ metal = \frac{3}{35}$$

*Tasa de error metal = 0,086; expresado en porcentajes sería 8.60%*

$$Promedio\ de\ Tasa\ de\ Error = \frac{(11.40 + 8.60 + 8.60)\%}{3}$$

$$Promedio\ de\ Tasa\ de\ Error = \frac{28.60}{3}$$

$$Promedio\ de\ Tasa\ de\ Error = 9.53\%$$

### **Sensibilidad (Recall):**

La sensibilidad, también conocida como recall o true positive rate (TPR), mide la proporción de muestras positivas que fueron correctamente identificadas por el modelo.

Para calcular la sensibilidad, se divide el número de verdaderos positivos (VP) entre la suma de verdaderos positivos (VP) y falsos negativos (FN) para cada clase.

La fórmula general para calcular la sensibilidad es:

$$Sensibilidad = \frac{Verdaderos\ Positivos\ (VP)}{Verdaderos\ Positivos\ (VP) + Falsos\ Negativos\ (FN)}$$

$$Sensibilidad\ vidrio = \frac{VP_{vidrio}}{VP_{vidrio} + FN_{vidrio}}$$

$$Sensibilidad\ vidrio = \frac{25}{25 + 4}$$

*Sensibilidad vidrio = 0,8621; expresado en porcentajes sería 86.21%*

$$\text{Sensibilidad plástico} = \frac{VP_{\text{vidrio}}}{VP_{\text{vidrio}} + FN_{\text{vidrio}}}$$

$$\text{Sensibilidad plástico} = \frac{27}{27 + 3}$$

*Sensibilidad plástico = 0,90; expresado en porcentajes sería 90%*

$$\text{Sensibilidad metal} = \frac{VP_{\text{vidrio}}}{VP_{\text{vidrio}} + FN_{\text{vidrio}}}$$

$$\text{Sensibilidad metal} = \frac{29}{29 + 1}$$

*Sensibilidad metal = 0,8621; expresado en porcentajes sería 96.67%*

$$\text{Promedio de Sensibilidad} = \frac{(86.21 + 90 + 96.67)\%}{3}$$

$$\text{Promedio de Sensibilidad} = \frac{272.88}{3}$$

$$\text{Promedio de Sensibilidad} = 90.96\%$$

## CONCLUSIONES

En el presente proyecto se diseñó un prototipo destinado a clasificar envases de diferentes materiales, incluyendo vidrio, plástico y metal, mediante el uso de técnicas de clasificación automática. Se combinaron métodos de Machine Learning y Visión Artificial para entrenar al modelo de clasificación y comparar los resultados obtenidos con las imágenes capturadas por la cámara del sistema. Se consideraron modelos específicos, como son: 3 de vidrio, 3 de lata y 3 de plástico usando el software Teachable Machine.

Los resultados obtenidos muestran una exactitud promedio del 90.47% en la clasificación de muestras, con un nivel de error del 9.53%. Además, la sensibilidad para detectar verdaderos positivos se sitúa en un 90.96%.

En conclusión, el prototipo demostró ser capaz de clasificar con precisión los envases según su material, lo que sugiere su viabilidad para aplicaciones prácticas en la industria de reciclaje o en el control de calidad de productos envasados. Sin embargo, es importante tener en cuenta que existen desafíos potenciales, como la posibilidad de confundir materiales similares o la necesidad de optimizar aún más el modelo para mejorar su eficacia en situaciones reales.

Este proyecto destaca el potencial de combinar conceptos de Machine Learning y Visión Artificial en campos no tradicionales, abriendo nuevas posibilidades para mejorar procesos operativos y optimizar recursos en diversos sectores industriales. Es evidente que el uso de tecnologías avanzadas puede conducir a avances significativos en la automatización y la eficiencia de los sistemas.

## RECOMENDACIONES

Para mejorar la detección de materiales metálicos, se recomienda la integración de sensores inductivos. Estos dispositivos son capaces de detectar la presencia de materiales conductores, como el metal, mediante la medición de cambios en la inductancia eléctrica. Al incorporar sensores inductivos en el prototipo, se podría mejorar la capacidad del sistema para identificar y clasificar con precisión los envases de metal, reduciendo así la incidencia de falsas clasificaciones.

En cuanto a la detección del vidrio, se sugiere el uso de sensores ópticos especializados que puedan distinguir entre materiales transparentes y opacos. Por ejemplo, se podría considerar la integración de sensores de infrarrojos sensibles a las propiedades ópticas del vidrio. Estos dispositivos pueden detectar la presencia de vidrio basándose en la forma en que interactúan con la luz infrarroja, lo que permitiría una detección más precisa y confiable de los envases de vidrio en el prototipo.

Además, se recomienda realizar pruebas exhaustivas en entornos industriales reales para validar la eficacia del sistema de detección de materiales y garantizar su funcionamiento óptimo en situaciones prácticas. Esto incluye la evaluación de la sensibilidad y especificidad del sistema en condiciones ambientales variadas, así como la verificación de su capacidad para detectar con precisión una amplia gama de materiales.

Por último, se sugiere mantener una política de actualización continua del software y el hardware del prototipo, así como seguir investigando nuevas tecnologías y metodologías en el campo de la detección de materiales. Esto garantiza que el sistema de detección siga siendo competitivo y efectivo a medida que avanza la tecnología y se desarrollan nuevas soluciones en el campo.



## REFERENCIAS

Marketing. (2022, enero 31). Visión por Computador Qué es, Aplicaciones y Objetivos. EDS Robotics. <https://www.edsrobotics.com/blog/vision-computador-que-es/>

Russell, S. J., & Norvig, P. (2021). Artificial Intelligence: A Modern Approach (4th ed.). Pearson.

Teachable Machine. (s.f.). Teachable Machine. Obtenido de <https://teachablemachine.withgoogle.com/>

Ullah, Z., Al-Turjman, F., Mostarda, L., & Gagliardi, R. (2020). Applications of Artificial Intelligence and Machine learning in smart cities. Computer Communications.

O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G. V., Krpalkova, L., & Walsh, J. (2019). Deep learning vs. traditional computer vision. Science and Information Conference, (págs. 128-144).

Estadística de Información Ambiental Económica en Gobiernos Autónomos Descentralizados Municipales. Gob.ec. Recuperado el 19 de septiembre de 2023, de https://www.ecuadorencifras.gob.ec/documentos/web-inec/Encuestas\_Ambientales/Municipios\_2021/Residuo\_solidos\_2021/Presentaci%

Ruiz , C. A., & Basualdo, M. S. (2001). Redes Neuronales: Conceptos Básicos y Aplicaciones. Rosario.

Smith, A., Johnson, B., & Brown, C. (2020). Application of Deep Learning Algorithms for Automated Packaging Classification. *Waste Management*, 25(3), 123-135.

Instituto Nacional de Estadística y Geografía. (2020). Estudio sobre la implementación de sistemas de clasificación automatizada de envases en plantas de reciclaje en Colombia. Bogotá, Colombia: Editora Nacional.

Gómez, J., Rodríguez, M., & Martínez, L. (2019). Implementación de un sistema de clasificación automatizada de envases en una planta de reciclaje en Bogotá. Bogotá, Colombia: Universidad Nacional de Colombia.

Doe, J., & Smith, E. (2018). Challenges and Considerations in Automated Packaging Classification Systems. *Journal of Waste Management*, 15(1), 45-56.

García, M., López, R., & Pérez, S. (2021). Impact of Automated Packaging Classification Systems on Waste Management: A Case Study in Colombia. *Journal of Environmental Management*, 30(4), 210-225

Jones, A. B., & Smith, C. D. (2018). Avances en la detección de materiales mediante sensores inductivos. *Journal of Sensor Technology*, 10(2), 45-58.

García, E. F., & López, M. J. (2020). Aplicaciones de los sensores ópticos en la clasificación de materiales. *Revista de Ingeniería Electrónica*, 15(3), 112-125.

Martínez, R. G., & Pérez, S. L. (2019). Desarrollo de un prototipo de clasificación de materiales utilizando sensores capacitivos. *Ingeniería y Tecnología*, 8(1), 78-89.

## 4. ANEXO

### Código en Python

```
import numpy as np
import cv2, os, serial, math
import tensorflow.keras as tf

# Buscar ruta donde se almacenan los datos
DIR_PATH = os.path.dirname(os.path.realpath(_file_))

# Definir puerto y baudios de Arduino
arduino = serial.Serial(port='/dev/ttyUSB0', baudrate=9600, timeout=.1)

def main():
    # Ruta de archivo labels.txt donde se encuentran las etiquetas
    labels_path = f"{DIR_PATH}/keras16/labels.txt"

    # Abrir el archivo de etiquetas en modo lectura
    labelsfile = open(labels_path, 'r')

    # Inicializar clases y leer en líneas
    classes = []
    line = labelsfile.readline()
    while line:
        # Obtiene solo el nombre de la clase y la agrega a la variable classes
        classes.append(line.split(' ', 1)[1].rstrip())
        line = labelsfile.readline()
    # Cerrar archivo de etiquetas
    labelsfile.close()

    # Ruta donde se aloja modelo descargado
    model_path = f"{DIR_PATH}/keras16/keras_model.h5"

    # Cargar modelo de Teachable Machine
    model = tf.models.load_model(model_path, compile=False)

    # Iniciar camara web
    cap = cv2.VideoCapture(0)

    # Ancho y Alto de la camara web en pixeles -> Ajustar dependiendo de la resolución de la
    camara
    frameWidth = 640 #1080x720
    frameHeight = 480

    # Definir ancho y alto en píxeles
```

```

cap.set(cv2.CAP_PROP_FRAME_WIDTH, frameWidth)
cap.set(cv2.CAP_PROP_FRAME_HEIGHT, frameHeight)

# Habilitar ganancia autom.
cap.set(cv2.CAP_PROP_GAIN, 0)

# Mantiene programa en bucle - Cerrar Ctrl+c o cerrar ventana
while True:
    # Desactivar notación científica
    np.set_printoptions(suppress=True)

    # Crear matriz para alimentar modelo de Keras.
    # Ingresando imagen RGB de 1x224x224 píxeles.
    data = np.ndarray(shape=(1, 224, 224, 3), dtype=np.float32)

    # Capturar imagen
    check, frame = cap.read()

    # Imagen espejo - Por defecto en Teachable Machine
    frame = cv2.flip(frame, 1)

    # Recortar a cuadrado para usar con modelo TM
    # Se añade margenes a imagen para hacerla cuadrada y enviarla a TM con tamaño de 224x224
    margin = int(((frameWidth-frameHeight)/2))
    square_frame = frame[0:frameHeight, margin:margin + frameHeight]

    # Cambiar tamaño a 224x224 para usar con modelo TM
    resized_img = cv2.resize(square_frame, (224, 224))

    # Convertir color de imagen (BGR a RGB)
    model_img = cv2.cvtColor(resized_img, cv2.COLOR_BGR2RGB)

    # Convertir imagen en matriz numpy
    image_array = np.asarray(model_img)

    # Normalizar imagen para que los pixeles esten en un rango entre -1 y 1
    normalized_image_array = (image_array.astype(np.float32) / 127.0) - 1

    # Cargar imagen en matriz
    data[0] = normalized_image_array

    # Ejecutar predicción
    predictions = model.predict(data)

    # Umbral de confianza => 90%.
    conf_threshold = 95

```

```

confidence = []
conf_label = ""
threshold_class = ""

# Crear borde negro para las etiquetas
per_line = 2 # Número de clases por línea de texto
bordered_frame = cv2.copyMakeBorder(
    square_frame,
    top = 0,
    bottom = 30+15*math.ceil(len(classes)/per_line),
    left = 0,
    right = 0,
    borderType = cv2.BORDER_CONSTANT,
    value = [0, 0, 0]
)

# Para cada una de las clases
for i in range(0, len(classes)):
    # Escalar confianza de predicción en % y agregarla a la lista
    confidence.append(int(predictions[0][i]*100))

    # Poner texto por línea según el número de clases por línea
    if (i != 0 and not i % per_line):
        cv2.putText(
            img=bordered_frame,
            text=conf_label,
            org=(int(0), int(frameHeight+25+15*math.ceil(i/per_line))),
            fontFace=cv2.FONT_HERSHEY_SIMPLEX,
            fontScale=0.5,
            color=(255, 255, 255)
        )
        conf_label = ""

# Agregar clases y confianzas al texto para etiquetar
conf_label += classes[i] + ": " + str(confidence[i]) + "%; "
# Imprimir última línea
if (i == (len(classes)-1)):
    cv2.putText(
        img=bordered_frame,
        text=conf_label,
        org=(int(0), int(frameHeight+25+15*math.ceil((i+1)/per_line))),
        fontFace=cv2.FONT_HERSHEY_SIMPLEX,
        fontScale=0.5,
        color=(255, 255, 255)
    )
    conf_label = ""

```

```

# Si está por encima del umbral de confianza, enviar a la cola
if confidence[i] > conf_threshold:
    num=0

#####

#aqui cambiar yo y nada por el nombre de vuestras clases
if str(classes[i])=="Vidrio": #Vacio
    num=1

elif str(classes[i])=="Plastico": #Vidrio
    num=2

elif str(classes[i])=="Metal": #Plastico
    num=3

elif str(classes[i])=="Vacio": #Metal
    num=4

elif str(classes[i])=="Objeto_incorrecto": #Objeto incorrecto
    num=5

#####

    arduino.write((str(num)+ '\n').encode())
    threshold_class = classes[i]
    print(str(num)+' - '+threshold_class)
# Agregar clase de etiqueta por encima del umbral de confianza
cv2.putText(
    img=bordered_frame,
    text=threshold_class,
    org=(int(0), int(frameHeight+20)),
    fontFace=cv2.FONT_HERSHEY_SIMPLEX,
    fontScale=0.75,
    color=(255, 255, 255)
)

# Captura de video en tiempo real
cv2.imshow("Capturing", bordered_frame)
cv2.waitKey(10)

if __name__ == '__main__':
    main()

```

## Código Arduino para funcionamiento de servomotores

```
#include <Servo.h>
#include <Wire.h>
#include <Adafruit_PWMServoDriver.h>

Adafruit_PWMServoDriver servos = Adafruit_PWMServoDriver(0x40);

unsigned int pos0 = 172;
unsigned int pos180=565;

int num = 0;
int numant = 0;

void setup(){
  Serial.begin(9600);
  servos.begin();
  servos.setPWMPFreq(60);
}

void setServo(uint8_t n_servo, int angulo) {
  int duty;
  duty = map(angulo,0,180,pos0, pos180);
  servos.setPWM(n_servo, 0, duty);
}

void loop() {
  if (Serial.available()) {
    String num_str = Serial.readStringUntil('\n'); // Leer cadena hasta el delimitador '\n'
    num = num_str.toInt(); // Convertir cadena a entero
    if(num != numant){ // 0 - ABIERTO // 90 - CERRADO
      if(num == 1){ // VIDRIO
        setServo(0, 0); delay(750);
        setServo(1,90); setServo(2,90); setServo(3,180);
        //delay(2000);
      }
      if(num == 2){ // PLASTICO
        setServo(0,90); setServo(1, 0); setServo(2,90); setServo(3,180);
        //delay(2000);
      }
      if(num == 3){ // METAL
        setServo(0,90); setServo(1,90); setServo(2, 0); setServo(3,180);
        //delay(2000);
      }
      if(num == 4){ // VACIO
        setServo(0,90); setServo(1,90); setServo(2,90); setServo(3, 0);
      }
    }
  }
}
```

```
//delay(2000);  
}  
if(num == 5){ // OBJETO INCORRECTO  
  setServo(0,90); setServo(1,90); setServo(2,90); setServo(3, 0);  
  //delay(2000);  
}  
numant = num;  
}  
}  
}
```