

Sistema de Clasificación Automática de Residuos para Máquinas Ecobot mediante Visión por Computador y Deep Learning

- Cano, Andrés
- García, Daniel
- Sandoval, Farid



ANÁLISIS DEL PROBLEMA

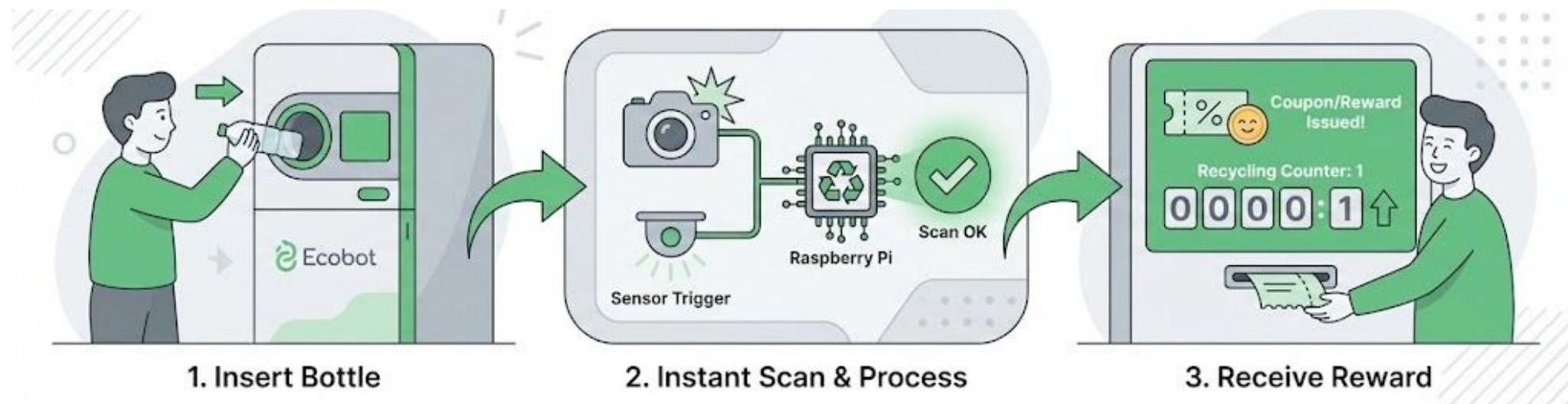
Ecobot es una empresa caleña, dedicada a la operación de máquinas expendedoras inversas (RVM) que recolectan envases reciclables como botellas, latas, PET y tetra. Dado el rol central del análisis de datos en modelos de economía circular, contar con un sistema automático de clasificación por visión computacional se vuelve crítico a la hora de ofrecer reportes de valor agregado a los aliados comerciales, marcas y entidades ambientales.



Problema Principal	Pregunta SMART
<p>La operación actual de las máquinas Ecobot no cuenta con un mecanismo automático para identificar el tipo de residuo que ingresa, lo cual:</p> <ul style="list-style-type: none">• Limita la trazabilidad por material.• Afecta la precisión de los reportes operativos.• Dificulta generar valor agregado para aliados estratégicos. 	<p>¿Qué modelo de visión por computadora ligero puede clasificar, con al menos 95% de precisión y tiempos de inferencia menores a 350 ms, los cuatro tipos de residuos que ingresan a una máquina Ecobot, utilizando una Raspberry Pi 3 y una arquitectura de captura activada por sensor e iluminación controlada?</p> 

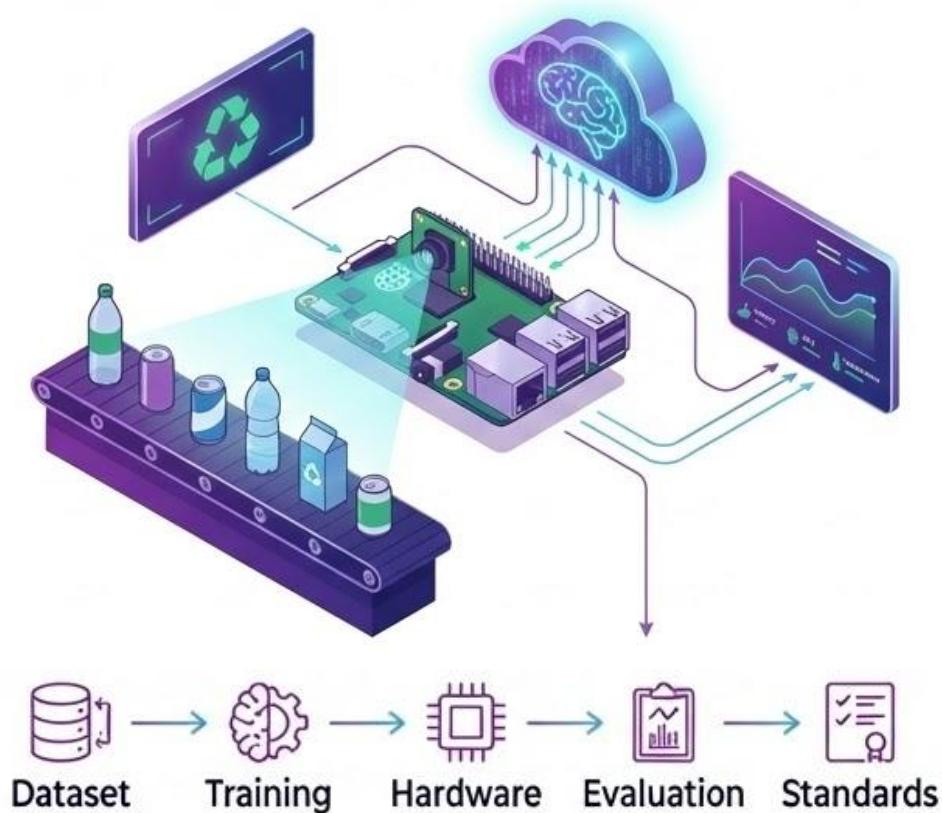
OBJETIVO GENERAL

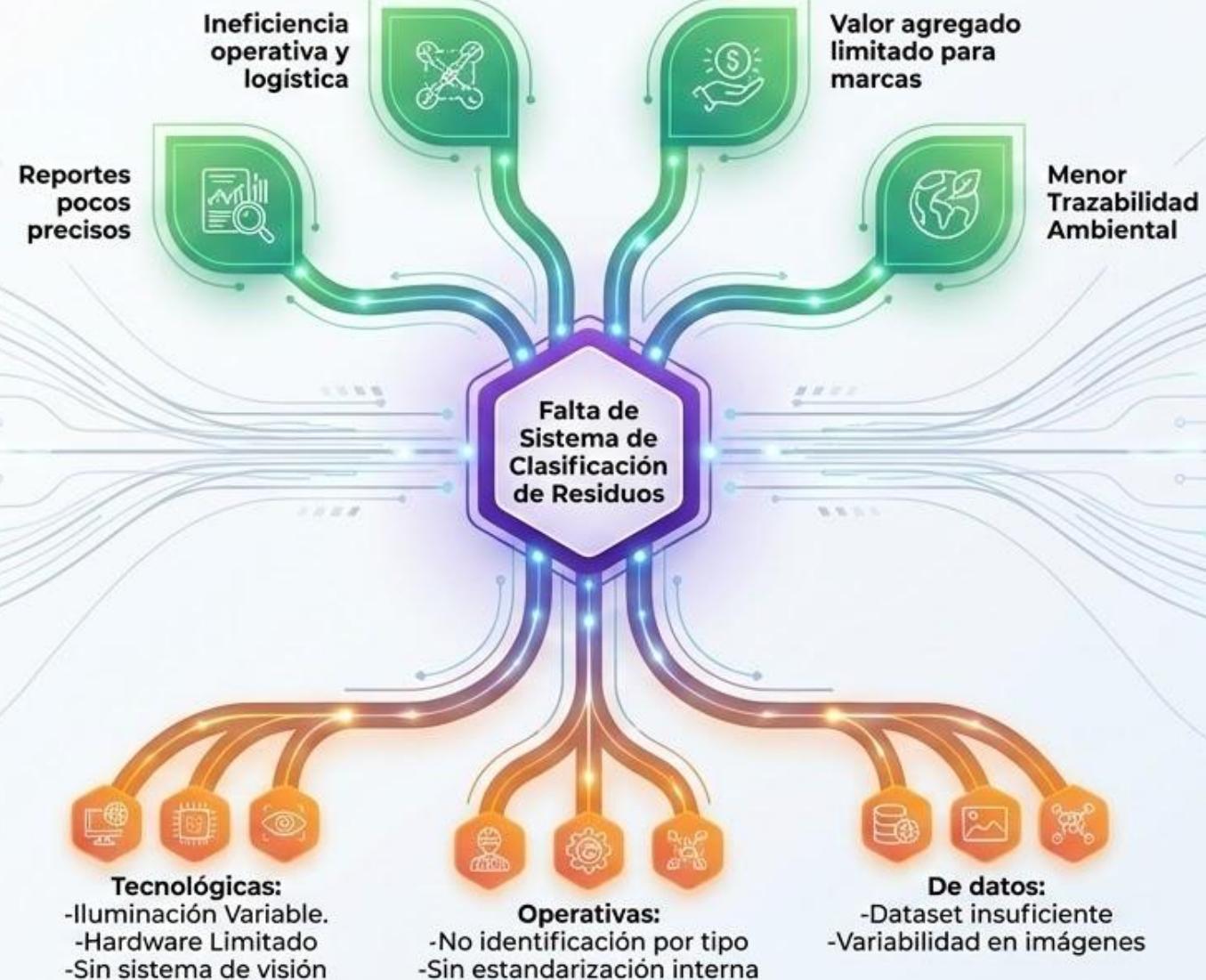
Desarrollar un sistema de clasificación automática de residuos basado en visión por computadora y modelos ligeros de deep learning, capaz de identificar botellitas, latas, PET y tetra en tiempo real, empleando una arquitectura de captura optimizada y ejecutándose en Raspberry Pi 3.

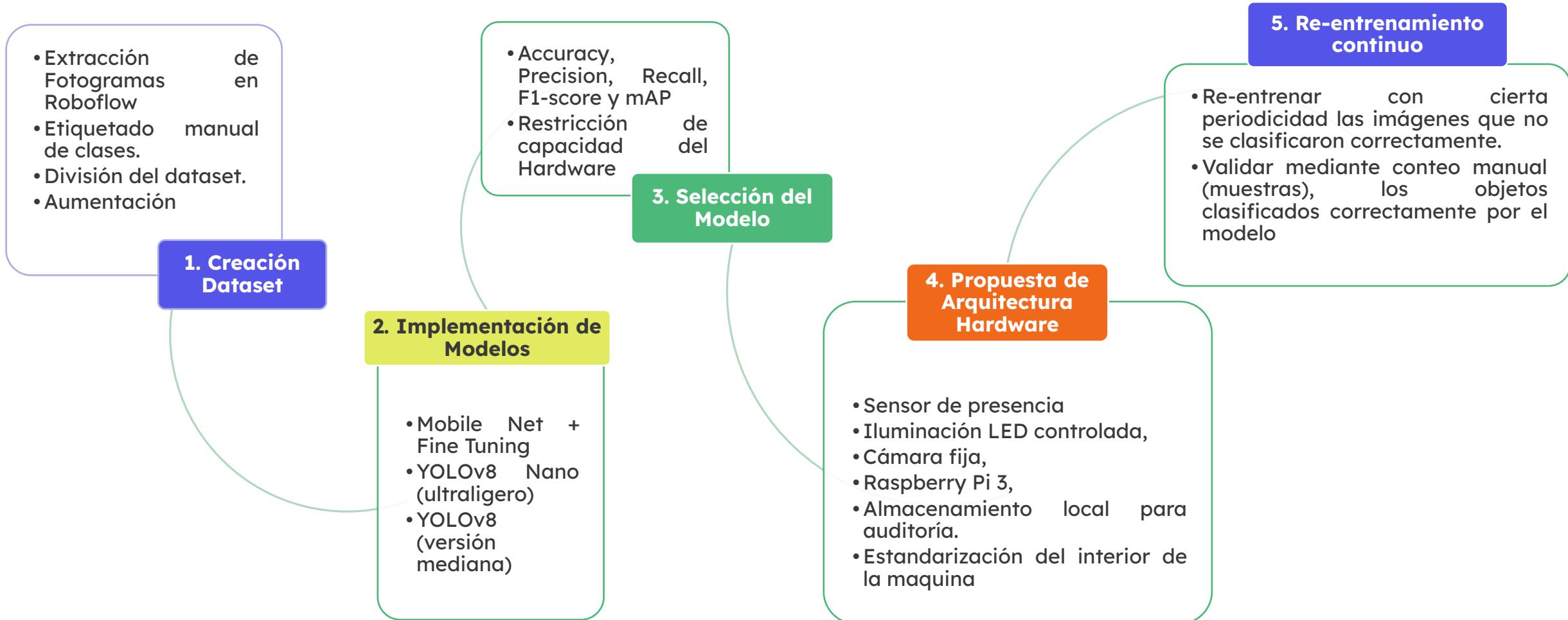


OBJETIVOS ESPECÍFICOS

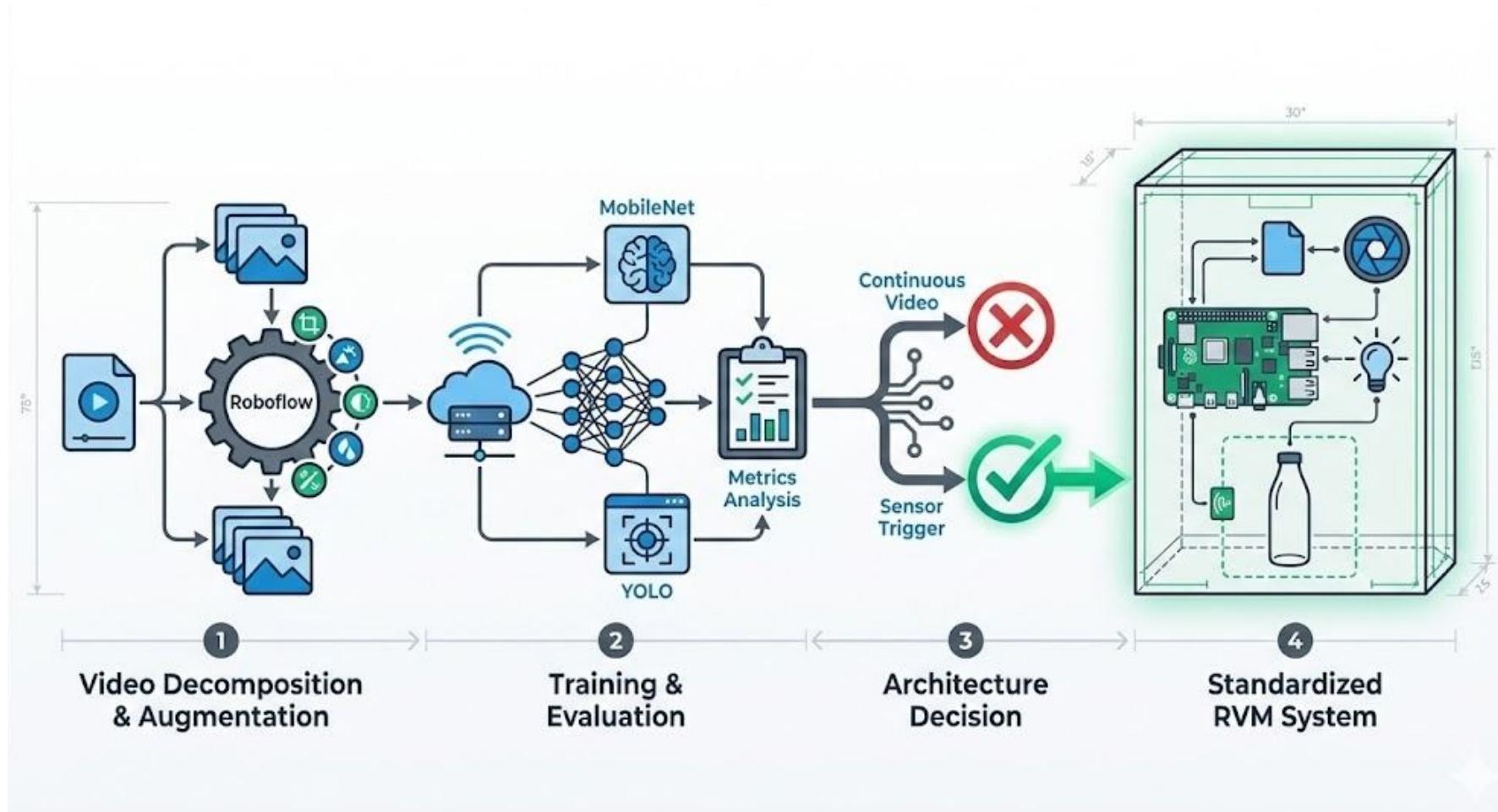
1. Construir un dataset a partir de videos reales de máquinas Ecobot.
2. Entrenar y comparar modelos ligeros en términos de precisión y velocidad.
3. Diseñar una propuesta de arquitectura operativa con sensor, iluminación controlada y captura por evento.
4. Evaluar el probable desempeño de cada modelo en Raspberry Pi 3 (latencia, uso de CPU, temperatura).
5. Proponer estándares físicos internos para mejorar la calidad y consistencia de la captura.



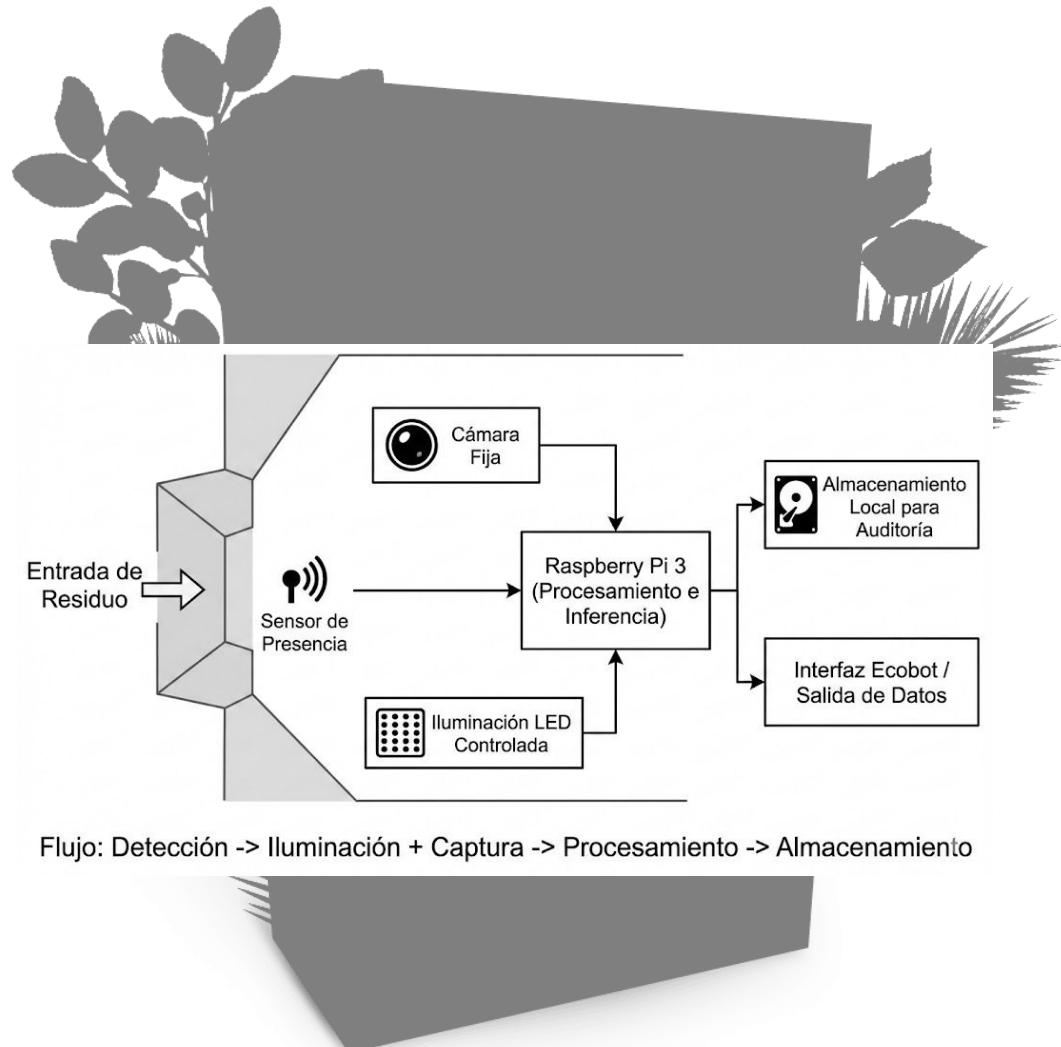




A continuación, se describen los pasos de nuestro proyecto.



A continuación, se describen la lógica de nuestro proyecto.



MobileNetV2

- Se seleccionó por su eficiencia computacional y su capacidad demostrada en dispositivos de bajo consumo (~3.4-5M)
- **Fase 1:** congelamiento del backbone y ajuste del clasificador.
- **Fase 2:** *fine-tuning* parcial.

YOLOv8 Nano (ultraligero)

- Su bajo número de parámetros (~3M) permite su ejecución eficiente en hardware ARM
- Suficiente precisión para tareas simples

YOLOv8 (versión mediana)

- Su mayor profundidad y anchura aumentan la precisión (~25M)
- Extrae características más ricas (edges → textures → shapes → context)
- Generaliza mejor objetos pequeños, traslapados, deformados o con poca luz

MobileNetV2

Accuracy en test: 94.44%

- F1-score promedio: 0.93
- Clases con mayor precisión: Lata y Tetra
- Clase más desafiante: PET (confusiones con botellita)
- Tiempo estimado por imagen: 7.337 +/- 1.603 ms

Clase	Precisión	Recall	F1-Score
Botellita	1	0.8444	0.9157
Lata	0.9672	0.9672	0.9672
Pet	0.7857	1	0.88
Tetra	0.9623	0.9808	0.9714

YOLO V8 NANO

Equilibrio adecuado entre velocidad y capacidad predictiva.

- mAP50: 0.9886
- Precision global: 0.99
- Recall global: 0.95
- Tiempo estimado por imagen (en GPU T4): ~9.5 ms

Resultados por clase (mAP50):

- Botellita: 0.97
- Lata: 0.995
- PET: 0.995
- Tetra: 0.995

YOLO V8 MEDIUM

El modelo YOLOv8 de tamaño mediano fue evaluado como referencia de máximo desempeño. Los resultados alcanzados fueron los mejores entre los modelos analizados:

- mAP50: 0.992
- Precision global: 0.974
- Recall global: 0.985
- Tiempo por imagen en GPU: ~15.7 ms

Resultados por clase (mAP50):

- Botellita: 0.986
- Lata: 0.99
- PET: 0.995
- Tetra: 0.995

A continuación se presenta un estimado del presupuesto que requiere cada arquitectura.

MobileNetV2

Título: MobileNetV2

Hardware: "Minichips" con acelerador dedicado (NPU) de bajo costo. Específicamente el Kendryte K210 (Sipeed Maix).

Detalles Técnicos:

- Ejecuta modelos ligeros como YOLOv2-tiny.
- Usa cámaras tipo DVP (interfaz antigua y básica).

Costo: 30 USD.

YOLO V8 NANO

Título: YOLO V8 NANO (Optimizado RPI 3)

Hardware: Kit Raspberry Pi 3 (usado/refurbished), Cámara PiCam V2, Sensor IR (Trigger), Iluminación LED controlada básica, MicroSD Industrial 32GB y Fuente 5V/3A.

Detalles Técnicos:

- Ejecuta YOLOv8 Nano en CPU ARM sin hardware adicional.
- Bajo consumo, compatible con arquitectura existente.

Costo: 130 USD.

YOLO V8 MEDIUM

Título 1: YOLO V8 MEDIUM (Optimizado RPi 3)

Hardware: Kit Raspberry Pi 3, Sensor IR (Trigger), Cámara (PiCam/USB), Iluminación LED controlada y MicroSD Industrial.

Costo: 130 USD

Título 2: YOLO V8 MEDIUM

Hardware: Kit NVIDIA Jetson Orin, Cámara de Alta Velocidad (Global Shutter), Iluminación LED, Fuente de poder industrial y almacenamiento SSD NVMe.

Costo: 1350 USD.

Reducción Drástica del CAPEX (Costo de Hardware)

Gracias al desempeño del modelo en la infraestructura existente, se eliminó la necesidad de adquirir nuevo hardware, generando un ahorro significativo en CAPEX y evitando un sobreendeudamiento innecesario.

Precisión Operativa (Evitar Pérdidas)

Recall PET 100%: Garantía de captación de todo el material de alto valor. No se pierden botellas válidas.

Filtro de Fraude: El sistema diferencia automáticamente una "Botellita de Amor" (plásticos flexibles) de una botella PET, evitando pagar incentivos por material no trazado en esta campaña.

Monetización de Datos (Nuevo Ingreso)

Trazabilidad Premium: Capacidad de cobrar un cuota adicional a aliados por reportes detallados (Ej: "¿Cuántas latas vs. botellas ingresaron en la máquina del Centro Comercial X?").

Auditoría Real: Eliminación del error humano en el conteo manual post-recolección.

CONCLUSIONES

- Factores como la iluminación, el movimiento del objeto y la variabilidad en el entorno físico pueden tener un impacto significativo en la precisión del sistema, esto resalta la necesidad de estandarizar el interior de la máquina.
- La implementación de un sistema automático de clasificación de residuos en máquinas Ecobot mediante modelos ligeros de deep learning es técnicamente factible. Los experimentos realizados demuestran que es posible alcanzar altos niveles de precisión incluso en hardware embebido como Raspberry Pi 3.

RECOMENDACIONES A FUTURO

- Integración IoT: Conectar la salida del modelo con la base de datos central de Ecobot para dashboards en vivo.
- Estandarización: Definir protocolos de iluminación interna para asegurar consistencia en la captura.