AI & SUSTAINABILITY IN VET EDUCATION

Nombre de los/las autores/as: Politeknika Txorierri

Título del WP: WP3 – AI-SAVES@ProjectHub

Referencia de la Actividad: Clasificador de Residuos con IA – Robot Móvil con LED y Navegación por Trayectorias

Logo:

POLITEKNIKA TXORIERRI

Cofinanciado por la Unión Europea

Este proyecto ha sido financiado con el apoyo de la Comisión Europea. Esta comunicación refleja únicamente las opiniones del autor, y la Comisión no puede ser considerada responsable del uso que pueda hacerse de la información contenida en la misma.

TABLA DE CONTENIDOS

1. Título de Proyecto: 5
2. Categoría del Proyecto: 5
3. Descripción del Proyecto: 5
4. Palabras Clave: 6
5. Público Objetivo: 6
6. Objetivos del Proyecto: 6
7. Cronograma del Proyecto: 6
8. Materiales Necesarios 7
9. Recursos Proporcionados: 8
10. Herramientas y Tecnologías Utilizadas: 8
11. Plan de implementación paso a paso: 8
12. Criterios de Evaluación: 9
13. Idiomas Disponibles: 10
14. Contribuidores/as: 10
15. Retroalimentación y mejora Continua: 10

Cofinanciado por la Unión Europea

Este proyecto ha sido financiado con el apoyo de la Comisión Europea. Esta comunicación refleja únicamente las opiniones del autor, y la Comisión no puede ser considerada responsable del uso que pueda hacerse de la información contenida en la misma.

1. **Título de Proyecto:**

Clasificador de Residuos con IA – Robot Móvil con LED y Navegación por Trayectorias

2. **Categoría del Proyecto:**

Inteligencia Artificial, Visión por Computador (Computer Vision), Robótica, Sostenibilidad.

3. **Descripción del Proyecto:**

Este proyecto utiliza una Raspberry Pi 5, una Pi Camera y un kit de robot 2WD con un controlador de motores L298N para reconocer tres tipos de materiales y mover el robot por diferentes trayectorias según el material detectado.

En lugar de utilizar un modelo de IA preentrenado, el sistema realiza una clasificación de materiales en tiempo real mediante técnicas de visión por computador (OpenCV), basándose en:

- Brillo (canal V en HSV)
- Saturación del color (canal S en HSV)
- Densidad de bordes (detección Canny)

El sistema clasifica los materiales como:

- Papel
- Plástico
- Metal

Muestra la imagen en vivo de la cámara y el nombre del material predicho en pantalla.

Enciende un LED diferente para cada material (por ejemplo: Verde, Amarillo, Rojo).

Hace que el robot ejecute un circuito de movimiento diferente:

- Material 1 (Papel)
  - Girar 90° a la izquierda y avanzar 30 cm
- Material 2 (Plástico)
  - Avanzar 30 cm
- Material 3 (Metal)
  - Girar 90° a la derecha y avanzar 30 cm

Cofinanciado por la Unión Europea

Este proyecto ha sido financiado con el apoyo de la Comisión Europea. Esta comunicación refleja únicamente las opiniones del autor, y la Comisión no puede ser considerada responsable del uso que pueda hacerse de la información contenida en la misma.

El robot utiliza un chasis 2WD con controlador de motores L298N y se alimenta con una batería de 12V y un convertidor buck 12V→5V para la Raspberry Pi.

4. **Palabras Clave:**

IA, Raspberry Pi 5, OpenCV, Visión por Computador, Robot Móvil, L298N, 2WD, LEDs, Sostenibilidad

5. **Público Objetivo:**

- Estudiantes de Formación Profesional (principiantes en Python y robótica)
- Profesorado que desee un proyecto sencillo de IA + robótica
- Talleres sobre reciclaje y sostenibilidad

6. **Objetivos del Proyecto:**

- Implementar un clasificador de materiales en tiempo real utilizando técnicas de visión por computador.
- Analizar imágenes de cámara utilizando características de color y bordes.
- Controlar LEDs y motores DC utilizando Python y lgpio.
- Hacer que el robot ejecute diferentes trayectorias según el material detectado.
- Comprender cómo técnicas de visión inspiradas en la IA pueden apoyar el reciclaje y la sostenibilidad.

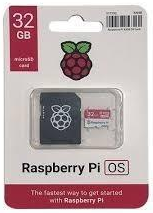
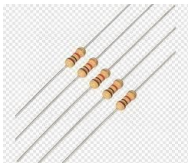
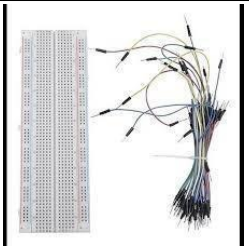
7. **Cronograma del Proyecto:**

Tarea	**Horas estimadas**	**Descripción**
Introducción e IA + reciclaje	1 hora	Explicar idea y objetivos
Fundamentos de visión por computador	2 horas	Color, brillo, bordes
Configuración Raspberry Pi	2 horas	Sistema operativo, librerías, prueba de cámara
Cableado LEDs y L298N	3 horas	Conectar LEDs, motores, probar scripts
Clasificación basada en visión	4 horas	HSV + detección de bordes
Integración IA + robot	5 horas	Clasificación + LEDs + trayectorias
Pruebas y ajustes (distancia/giro)	3 horas	Ajustar tiempos para 30 cm y 90°
Total	**20 horas**	

Cofinanciado por la Unión Europea

Este proyecto ha sido financiado con el apoyo de la Comisión Europea. Esta comunicación refleja únicamente las opiniones del autor, y la Comisión no puede ser considerada responsable del uso que pueda hacerse de la información contenida en la misma.

8. Materiales Necesarios

Raspberry Pi 5 (4GB/8GB)	
Raspberry Pi Camera (V3)	
MicroSD 32 GB con Raspberry Pi OS	
3 LEDs (Rojo, Verde, Amarillo) + 3× resistencias de 220–330 Ω	 
Protoboard + cables jumper	
Chasis robot 2WD con: ● 2 motores DC ● Ruedas ● Soporte/base para batería	
Controlador de motores L298N (doble puente H)	

Cofinanciado por la Unión Europea

Este proyecto ha sido financiado con el apoyo de la Comisión Europea. Esta comunicación refleja únicamente las opiniones del autor, y la Comisión no puede ser considerada responsable del uso que pueda hacerse de la información contenida en la misma.

Convertidor buck 12V → 5V (mín. 3A, recomendable 5A)	

9. Recursos Proporcionados:

- Conjuntos de datos de ejemplo y plantillas de Teachable Machine para clasificación de objetos.
- Scripts Python para control de motores y procesamiento de imágenes.
- Diagramas de cableado y guías de configuración de Raspberry Pi.
- Rúbrica de evaluación y plantilla de informe.

10. Herramientas y Tecnologías Utilizadas:

- Hardware: Raspberry Pi 5, Pi Camera, L298N, 2 motores DC, chasis 2WD, LEDs, resistencias, batería 12V, convertidor buck.
- Software: Raspberry Pi OS, Python 3, OpenCV, Picamera2, lgpio, Flask (para transmisión de vídeo), NumPy.
- Herramienta online: Google Teachable Machine (Proyecto de Imágenes).

11. Plan de implementación paso a paso:

1) **Fase A – Detección basada en visión**
   a) Capturar imágenes en vivo usando Picamera2.
   b) Convertir imágenes al espacio de color HSV.
   c) Analizar:
      i) Brillo medio (canal V)
      ii) Saturación media (canal S)
      iii) Densidad de bordes (Canny)
   d) Definir umbrales para clasificar:
      i) Papel
      ii) Plástico
      iii) Metal

2) **Fase B – Cableado del hardware**
   a) **Cámara**: conectar la Pi Camera al puerto CSI y probar con camera_test.py.
   b) **LEDs**:
      i) Conectar el ánodo de cada LED a un GPIO mediante una resistencia.
      ii) Ejemplo:
         (1) Rojo → GPIO 16
         (2) Verde → GPIO 19
         (3) Amarillo → GPIO 26

Cofinanciado por la Unión Europea

Este proyecto ha sido financiado con el apoyo de la Comisión Europea. Esta comunicación refleja únicamente las opiniones del autor, y la Comisión no puede ser considerada responsable del uso que pueda hacerse de la información contenida en la misma.

iii) Conectar todos los cátodos a GND.

c) **L298N + Motores (cableado sencillo):**
   i) Conectar ambos motores a los terminales OUT1/OUT2 y OUT3/OUT4.
   ii) Conectar la batería de 12V a L298N +12V y GND.
   iii) Colocar los jumpers ENA y ENB → siempre habilitados (velocidad máxima).
   iv) Conectar las entradas del L298N a los GPIO de la Raspberry Pi (BCM):
      (1) IN1 → GPIO 20 (motor izquierdo, adelante)
      (2) IN2 → GPIO 21 (motor izquierdo, atrás)
      (3) IN3 → GPIO 12 (motor derecho, adelante)
      (4) IN4 → GPIO 13 (motor derecho, atrás)
   v) **MUY IMPORTANTE**: Conectar el GND del L298N al GND de la Raspberry Pi.

d) **Alimentación**:
   i) Batería 12V → L298N +12V (motores)
   ii) Batería 12V → convertidor buck → 5V → USB-C a Raspberry Pi 5

3) **Fase C – Configuración del software**
   a) Instalar las librerías necesarias (muchas pueden estar ya instaladas):

```
sudo apt install python3-opencv python3-picamera2

pip install tflite-runtime gpiozero lgpio numpy
```

   b) Probar la cámara: python3 camera_test.py
   c) Probar los LEDs: python3 led_test.py
   d) Probar los motores: python3 motor_test.py

4) **Fase D – Integración completa (IA + LEDs + Robot)**
   a) Eejcutar: classify_robot.py:
      i) Imprime y dibuja el nombre del material detectado.
      ii) Enciende el LED correspondiente.
      iii) Cada vez que se detecta un material (con una pequeña pausa), el robot:
         (1) Para papel: gira ~90° a la izquierda + avanza ~30 cm
         (2) Para plástico: avanza ~30 cm
         (3) Para metal: gira ~90° a la derecha + avanza ~30 cm

   b) Las distancias y ángulos se controlan mediante constantes de tiempo (segundos). El alumnado puede ajustar estos tiempos por ensayo y error en su propio robot.
   c) El sistema espera varios fotogramas consecutivos con la misma clasificación antes de tomar una decisión (control de estabilidad).
   d) Cada material activa:
      i) Un LED específico
      ii) Una única secuencia de movimiento

Cofinanciado por la Unión Europea

Este proyecto ha sido financiado con el apoyo de la Comisión Europea. Esta comunicación refleja únicamente las opiniones del autor, y la Comisión no puede ser considerada responsable del uso que pueda hacerse de la información contenida en la misma.

e) El robot proporciona transmisión de vídeo en vivo vía navegador: http://IP:8080/video

12. Criterios de Evaluación:

- Calidad del dataset (25%): Variedad, iluminación, número de muestras.
- Montaje del hardware (20%): Cableado GPIO correcto y seguro.
- Integración de IA (30%): Detección estable en tiempo real y reacción LED.
- Creatividad y presentación (15%): Explicación, diagramas, sostenibilidad.
- Comprensión de la lógica de visión (10%): Explicación correcta de cómo se usan color y bordes para clasificar materiales.

13. Idiomas Disponibles:

- inglés y español

14. Contribuidores/as:

- **Miren Arrien (responsable del proyecto) –** Docente principal responsable del desarrollo y la coordinación del proyecto.
- **Enaitz Elejalde (Experto técnico) –** Aportó experiencia en IoT.
- **Aitor Rey (Experto técnico) –** Aportó experiencia en IoT.
- **Jokin Goioaga (Experto técnico)** – Aportó experiencia en IoT.

15. Retroalimentación y mejora Continua:

Después de cada fase del proyecto, estudiantes y profesorado deben revisar brevemente tanto el sistema de IA como el comportamiento del robot.

- **Revisión por fases**

Entrenamiento del modelo: comprobar si la IA identifica correctamente papel, plástico y metal.

  - Cableado: verificar que LEDs, motores y el controlador L298N están conectados de forma segura y funcionan correctamente.
  - Pruebas de movimiento: confirmar que el robot gira cerca de 90° y se mueve aproximadamente 30 cm.
  - Integración completa: asegurar que el robot reacciona correctamente a cada material (LED + movimiento).
  -

- **Reflexión final**

El alumnado evalúa:

Cofinanciado por la Unión Europea

Este proyecto ha sido financiado con el apoyo de la Comisión Europea. Esta comunicación refleja únicamente las opiniones del autor, y la Comisión no puede ser considerada responsable del uso que pueda hacerse de la información contenida en la misma.

- ¿Qué tan preciso fue el modelo de IA?
- ¿Qué tan fiable fue el movimiento del robot?
- ¿Qué desafíos surgieron durante la integración?

- **Propuestas de mejora**

El alumnado propone mejoras sencillas como:

- Ajustar umbrales de visión (brillo, saturación, densidad de bordes).
- Mejorar las condiciones de iluminación para una detección más estable.
- Ajustar tiempos de movimiento (mejor giro de 90° / desplazamiento de 30 cm).
- Añadir sensores para detección de obstáculos.
- Combinar reglas de visión con modelos sencillos de aprendizaje automático.
- Añadir pantalla, retroalimentación sonora o monitorización web.

- **Uso futuro**

Los resultados y sugerencias ayudan a mejorar futuras clases que combinen IA, robótica y sostenibilidad.

Cofinanciado por la Unión Europea

Este proyecto ha sido financiado con el apoyo de la Comisión Europea. Esta comunicación refleja únicamente las opiniones del autor, y la Comisión no puede ser considerada responsable del uso que pueda hacerse de la información contenida en la misma.