

Universidad Tecnológica del Perú

**Robot remoto apoyado con IA en la búsqueda de personas con sensor facial y acústico**

Curso Integrador 1: Diseño Electrónico

**Docente:**

Motta Zorrilla, Bryan

**Integrantes:**

- Romainville Barrionuevo, Camilo Jesús

- Alocén de la Cruz, Saul Junior

- Esquivel Giraldo, Hiram Renato

- Carmona Cerazo, Abram Yexuan

- Huaman Ramos, Jhonatan Rodrigo

- Carbajal Coral, Joshua

**Lima-Perú**

**2025**

1. **Planteamiento del problema**
   1. **Realidad Problemática**

En situaciones de desastres naturales, derrumbes o colapsos de edificios, uno de los mayores desafíos para los equipos de rescate es la localización rápida de las personas atrapadas entre los escombros. Las primeras horas son cruciales para poder salvar vidas, sin embargo, los métodos tradicionales de búsquedas no siempre resultan ser tan eficientes. Además, estos procedimientos pueden poner en riesgo la vida de los rescatistas.

El Perú se encuentra ubicado en una zona de interacción de placas tectónicas, lo que hace que la ocurrencia de sismos sea más frecuente en comparación a otros países. Por ello los cuerpos de rescate deben de estar preparados para actuar de manera eficaz ante este tipo de desastres. Además es importante considerar que algunas estructuras fueron mal construidas o no recibieron el mantenimiento adecuado. Un ejemplo de ello es el caso de Real Plaza Trujillo, donde, debido a la falta de mantenimiento en las canaletas, el techo del patio de comidas colapsó al superar su capacidad de carga por la acumulación de agua de lluvia.

Ante esta problemática, se plantea la necesidad de diseñar un robot remoto con inteligencia artificial, capaz de analizar señales de supervivencia mediante sensores faciales el cual es capaz de identificar rasgos humanos y sensores acústicos ya que son capaces de detectar sonidos como gritos de ayuda o golpes. Este tipo de tecnología podría optimizar las labores de rescate y reducir los riesgos para los equipos de rescate. Por ello, surge la siguiente interrogante: ¿Puede un robot remoto con inteligencia artificial, equipado con sensores faciales y acústicos, mejorar la localización de personas en situaciones de derrumbes o colapsos estructurales?

* 1. **Formulación del problema** 
     1. **Problema general**

¿De qué manera un robot remoto con inteligencia artificial, equipado con sensores faciales y acústicos, puede mejorar la localización de personas en situaciones de derrumbes o colapsos estructurales?

* + 1. **Problemas específicos**
* ¿Cómo influye el nivel de entrenamiento de la inteligencia artificial en el funcionamiento del robot durante la localización de personas en estructuras colapsadas?
* ¿Cómo influye la precisión del sensor facial en la detección de personas atrapadas en entornos con baja visibilidad o estructuras colapsadas?
* ¿De qué manera la combinación de sensores faciales y acústicos mejora la localización de personas en comparación con el uso de un solo sensor?
* ¿Cuál es la relación entre la potencia del sensor y la distancia(m) a la que se encuentra una persona de una persona?
  1. **Objetivos del proyecto**
     1. **Objetivo general:**

Reducir el tiempo de búsqueda en situaciones de rescate para brindar apoyo a los rescatistas.

* + 1. **Objetivos específicos:**
* Determinar cómo influye el nivel de entrenamiento de la IA con la capacidad de localización de personas.
* Determinar el nivel de precisión del sensor facial pese a la poca visibilidad del ambiente.
* Determinar la eficiencia de dos sensores a comparación de un sensor.
* Determinar posición por medio de la potencia y la distancia.
  1. **Justificación**

La investigación de este tema es de vital importancia, ya que contribuirá significativamente a las labores de rescate en situaciones críticas como colapsos estructurales, deslizamiento de la tierra o terremotos. Por esta razón, se propone la implementación de robots centrados en la localización de personas, con el objetivo de aumentar la efectividad de los rescates. Además, se busca reducir el riesgo al cual se enfrentan los rescatistas durante estas operaciones.

Los métodos de búsqueda tradicionales, como el empleo de herramientas manuales o perros entrenados para la localización de personas,han demostrado ser eficaces hasta cierto punto. Sin embargo, su desempeño puede verse afectado por factores externos.

Según un artículo publicado por Mendoza Talledo y Meza Capcha (2025) en infobae, el colapso del techo del patio de comidas en el Real Plaza de Trujillo provocó la muerte de ocho personas y dejó más de ochenta heridos Uno de los métodos utilizados en la búsqueda de víctimas fue el empleo de perros especializados en rescate; sin embargo, su eficacia se vio comprometida debido al alto nivel de ruido proveniente de las discotecas cercanas, lo que dificulto su capacidad de rastreo.

En contraste, la implementación de los sensores acústicos controlados por la IA permitirá filtrar los ruidos del entorno, enfocados únicamente en los sonidos relevantes generados por personas atrapadas en la zona afectada. Esto representaría una ventaja significativa frente a las limitaciones de los métodos actuales.

El dispositivo móvil será operado a control remoto y contará con la IA capaz de recopilar y enviar información relevante sobre la situación, como cantidad de personas atrapadas o información sobre zonas de riesgo, permitiendo al personal tomar decisiones más seguras y oportunas para realizar el trabajo, minimizando de esta forma las posibles bajas del personal de rescate.

Pese a que el proyecto se encuentra aún en fase de prueba, su implementación podría generar un impacto positivo en labores de localización de personas, beneficiando a organizaciones humanitarias, cuerpos de seguridad e instituciones de protección civil. Asimismo su desarrollo contribuirá al avance de la robótica orientada al bienestar social y la protección de la vida humana.

1. **Marco Teórico**
   1. **Antecedentes**

A lo largo del tiempo, los desastres naturales y los colapsos estructurales han dejado a varias personas atrapadas entre los escombros, generando unas situaciones de gran urgencia. Debido a estos acontecimientos se ha impulsado la búsqueda de soluciones tecnológicas que ayuden a los equipos de rescate a salvar vidas de forma más rápida y segura. Algunos casos como, el colapso de una parte del centro comercial del Real Plaza en Trujillo, o una tragedia más recientemente que ocurrió en República Dominicana en el cual, se derrumbó el techo de una discoteca. Estos hechos, han puesto en evidencia la necesidad de contar con herramientas más eficaces para localizar a las víctimas en estos accidentes.

En los últimos años, el panorama ha comenzado a cambiar gracias a la integración de inteligencia artificial y sensores de última generación. Tecnologías como los sensores acústicos y el reconocimiento facial están ayudando a detectar señales humanas, incluso en ambientes caóticos o con poca visibilidad. Estos avances están abriendo la puerta a una nueva generación de robots con sistemas manejados por IA, capaces de tomar decisiones en tiempo real y convertirse en verdaderos aliados en momentos de gran urgencia.

En la investigación realizada por Álvarez-Cedillo, Álvarez-Sánches, Sandoval, Sandoval-Gutiérrez y Nava-Vega (2019) titulada “*Diseño de un robot rescatista para terremotos en México*”, se desarrolló un robot enfocado en labores de rescate, mediante sensores para detectar a personas atrapadas entre los escombros del terremoto.

El estudio implementó un diseño que le permitiera al robot desplazarse por terrenos irregulares, alcanzando una altura de hasta 20 cm y pendientes de hasta 60 grados. Además, se utilizó un sistema de accionamiento dual para los motores, uno destinado para avance y retroceso, y otro para subir pendientes. En su fase de pruebas, trabajaron primero con un prototipo a menor escala, el cual ayudó a determinar variables como el rendimiento del sensor, posteriormente, se construyó el prototipo final a escala real.

Este antecedente es relevante para la presente investigación, ya que proporciona un diseño funcional que puede ser adaptado al desarrollo de un robot capaz de desplazarse por terrenos irregulares, esto contribuirá a reducir las posibilidades de vuelco o atasco durante su operación en escenarios de desastre.

En la investigación realizada por Gordon, Encalada, Lema, León y Chicaiza (2019) titulada “Human Rescue Based on Autonomous Robot KUKA Youbot with ROS and Object Detection”, en esta investigación se presenta un sistema que utiliza la plataforma móvil KUKA YouBot en cual realiza procesos de navegación, exploración y detección de personas, de esta forma generará un mapa con la ubicación de los individuos detectados.

El estudio empleó el software Robot Operating System (ROS) junto con las librerías de OpenCV para poder ejecutar los procesos de detección, incorporando además el algoritmo Single Shot Detector y una red neuronal previamente entrenada para identificar personas. Gracias a esta integración, el robot fue capaz de detectar individuos dentro de un edificio incluso cuando su visibilidad está parcial o totalmente bloqueada por objetos o estructuras del entorno, registrando sus coordenadas para la generación de un mapa 2D.

Este antecedente resulta ser relevante para nuestro informe, ya que proporciona ideas sobre cómo configurar una IA para la detección de personas, incluso en condiciones de visibilidad reducida debido a obstáculos estructurales o baja iluminación. Además, destaca la posibilidad de obtener las coordenadas de los individuos detectados, lo cual contribuye de manera significativa a la calidad y utilidad del informe generado por el mismo sistema.

* 1. **Base teórica**

La base de este proyecto se fundamenta en las siguientes áreas a mencionar:

* **SENSOR FACIAL**

## **Red Neuronal Convolucional (CNN) – Función de Convolución**

Permite aplicar un filtro (kernel) a una imagen de entrada para extraer patrones espaciales relevantes, como bordes y texturas.

Se emplea para identificar rostros humanos dentro de imágenes captadas por el robot en tiempo real, facilitando así su localización en entornos complejos o con poca visibilidad.

…………(1)

Variables:

* **S(i, j):** Valor de salida en la posición (i, j) del mapa de características.
* **I(m, n):** Valor de la imagen de entrada en la posición (m, n).
* **K(i - m, j - n):** Valor del kernel (filtro) en la posición correspondiente al desplazamiento.
* **∑ₘ ∑ₙ:** Sumatoria sobre los índices del filtro para aplicar convolución en toda la imagen.

## **Función de Activación ReLU**

Introduce no linealidades en la red neuronal y permite que solo pasen valores positivos, eliminando las activaciones negativas.

Se emplea porque mejora significativamente la velocidad de entrenamiento y la capacidad de generalización al detectar rostros.

…………….(2)

Variables:

* **x:** Valor de entrada a la neurona (puede ser positivo o negativo).
* **f(x):** Valor de salida activado (0 si x < 0; x si x ≥ 0).

## **Distancia Euclidiana – Comparación de rostros**

Mide la distancia entre dos vectores de características extraídas de imágenes faciales.

Se utiliza para comparar el rostro detectado con rostros previamente almacenados o de identificar si se trata de una persona nueva.

…………..(3)

Variables:

* **d(p, q):** Distancia euclidiana entre dos vectores p y q.
* **pᵢ, qᵢ:** Componentes individuales de los vectores de características de dos imágenes faciales.
* **∑:** Sumatoria sobre todas las dimensiones del vector.
* **SENSOR ACÚSTICO**

## **Transformada de Fourier – Análisis de sonido**

Convierte una señal del dominio temporal al dominio de frecuencia.

Se emplea para identificar frecuencias específicas asociadas a sonidos humanos (como gritos o golpes), facilitando la detección auditiva de personas atrapadas.

.....................(4)

Variables:

* **X(f):** Representación en frecuencia de la señal.
* **x(t):** Señal original en el dominio del tiempo.
* **f:** Frecuencia a analizar.
* **j:** Unidad imaginaria.
* **e^(-j2πft):** Núcleo de la transformada, que convierte tiempo en frecuencia.
* **MULTISENSORIAL**

## **Inferencia Bayesiana – Fusión de sensores**

Permite combinar la evidencia obtenida de sensores visuales y acústicos para calcular la probabilidad de que haya una persona en una determinada ubicación.

Mejora la fiabilidad del sistema al unir múltiples fuentes de datos con incertidumbre.

........................(5)

Variables:

* **P(H|E):** Probabilidad de la hipótesis H (por ejemplo, “hay una persona”) dado que se observó la evidencia E.
* **P(E|H):** Probabilidad de observar la evidencia E si H es verdadera.
* **P(H):** Probabilidad previa de H antes de observar E.
* **P(E):** Probabilidad total de observar E bajo todas las hipótesis.
* **INTELIGENCIA ARTIFICIAL (IA)**

## **Algoritmo de Optimización Adam**

Es un optimizador que combina el promedio móvil del gradiente (mₜ) y su cuadrado (vₜ) para actualizar los pesos de la red neuronal.

Esto acelera la convergencia y mejora la estabilidad durante el entrenamiento de los modelos que procesan datos faciales y acústicos en el robot.

.................(6)

...................(7)

........................(8)

Variables:

* **mₜ:** Promedio móvil del gradiente (primera estimación).
* **vₜ:** Promedio móvil del cuadrado del gradiente (segunda estimación).
* **gₜ:** Gradiente del error con respecto a los parámetros en el tiempo t.
* **β₁, β₂:** Términos de decaimiento exponencial (normalmente 0.9 y 0.999).
* **η (eta):** Tasa de aprendizaje.
* **ε:** Pequeño valor para evitar división por cero (ej. 10⁻⁸).
* **θₜ:** Parámetros actualizados del modelo.
* **m̂ₜ, v̂ₜ:** Estimaciones corregidas del sesgo de mₜ y vₜ.
  1. **Base conceptual**

**Sensor acústico:** Un sensor acústico es un dispositivo diseñado para detectar ondas sonoras y convertirlas en señales eléctricas. (StudySmarter, 2025)

**Sensor facial:** Dispositivo que detecta, analiza e interpreta los rasgos del rostro humano a partir de imágenes.digitales o secuencia de video (Vezzetti & Marcolin, 2012).

**Java:** Es un lenguaje de programación multiplataforma orientado a objetos, que se ejecuta en miles de millones de dispositivos gracias a la Máquina Virtual de Java (JVM). Este lenguaje es ampliamente utilizado para el desarrollo de aplicaciones de escritorio, aplicaciones móviles, software empresarial, entre otros. Una de las características principales es la portabilidad, debido a que el código escrito en Java puede ejecutarse en cualquier otro dispositivo que tenga instalado la JVM, independientemente del sistema operativo. (Oracle, 2025)

**Robótica móvil:** La robótica móvil consiste en el diseño y programación de robots capaces de desplazarse en diferentes entornos. Para aplicaciones de rescate, estos robots deben ser pequeños, resistentes, y capaces de navegar por terrenos irregulares. Su movilidad les permite acceder a zonas peligrosas sin exponerse a los rescatistas. Pero para este proyecto se llegará a diseñar un prototipo básico que nos ayude a cumplir nuestros objetivos.

**Inteligencia Artificial (IA):** Se refiere a la capacidad de un sistema informático de realizar tareas que normalmente requieren inteligencia humana, como el reconocimiento de patrones, el aprendizaje automático, y la toma de decisiones autónoma. (Amazon Web Services [AWS], 2025)

**Machine Learning:** Es una rama de la inteligencia artificial que permite a las máquinas aprender de datos y mejorar su rendimiento sin ser programadas explícitamente. (IBM, 2024)

**Computer Vision:** Campo interdisciplinario que se ocupa de cómo las computadoras pueden obtener una comprensión de alto nivel a partir de imágenes digitales o secuencias de vídeo. (Gregersen,(Ed.),2025)

**Procesamiento de señales:** Se refiere al conjunto de técnicas que nos ayudan a analizar las señales que fueron captadas por los sensores acústicos. De esta forma podremos identificar los eventos como gritos o golpes mediante su representación en el dominio de la frecuencia.

**PROTEUS:** Software especializado en el desarrollo de de circuitos digitales de manera simulador diseñado para visualizar el desarrollo de los circuitos que quieren ser aplicados en la vida real, o si no se tiene los suficientes componentes necesarios para su ensamblaje, este software es esencial para el desarrollo de cortos electrónicos y realizar un correcto aprendizaje de la electrónica de manera mucho más eficiente y rápida.(Balladares Rocha,2020)

**Ollama:** Plataforma que permite ejecutar modelos de lenguaje de gran escala (LLMs) de manera local, optimizando su uso en sistemas embebidos o entornos desconectados. El modelo de lenguaje implementado en Ollama actúa como un cerebro cognitivo que interpreta las señales acústicas (como gritos o golpes) y patrones faciales detectados por cámaras, permitiendo tomar decisiones en tiempo real, filtrar información relevante y guiar al operador hacia posibles víctimas. (Ollama, 2025)

**MPLAB X:** este software de programación de lenguaje de bajo nivel está presente como un pieza fundamental en el desarrollo de código referente al presente componente el cual es un microcontrolador de serie, , el cual es el estándar en definir las funciones de este tipo de componentes digitales, que son mayormente microcontroladores, todo claro si se tiene en cuenta como eje la programación mediante un dispositivo Pickit 3, y el conocimiento suficiente en el lenguaje de programación c, enfocado al diseño, sin contar el reconocimiento de las salidas respectivas del componente a modificar, claro al usar la hoja de datos respectiva, e integrar a un circuito analogico respectivo a un dispositivo de naturaleza digital. (Microchip Technology Inc., 2025)

**ROBOFLOW:** ofrece un conjunto de herramientas para crear e implementar modelos de visión artificial.

**YOLO V11:** Es un algoritmo capaz de detectar objetos a simple vista, realizando la detección y la clasificación simultáneamente.

1. **Variables e hipótesis**
   1. **Variables del Problema**
      1. **Independiente**

* Sensor acústico
* Imágenes
  + 1. **Dependientes:**
* Potencia del sensor acústico(distancia).
* Sensibilidad del sensor acústico: Nivel de exactitud del sensor al detectar un sonido como gritos o golpes.
* Precisión del informe generado: Que tan exacta es la información que entrega el sistema (porcentaje de aciertos en ubicación o identificación).
* Precisión del sensor facial: Nivel de exactitud del sensor al identificar rostros.
  1. **Hipótesis del Problema**

1. Hipótesis General

El uso de un robot equipado con sensores ayudado con IA mejora la eficacia en la localización de personas, al optimizar la detección de señales humanas y reducir el tiempo de búsqueda.

1. Hipótesis Específicas

* A mayor nivel de entrenamiento de la IA, mejor será la capacidad de localización de las personas bajo estructuras colapsadas.
* El sensor facial presenta una leve disminución de precisión bajo condiciones de baja iluminación, sin embargo, mantiene un rendimiento lo suficientemente alto como para mejorar la capacidad de localización del robot en entornos colapsados.
* La integración de sensores faciales y acústicos incrementará la capacidad de localización del robot al combinar funciones visuales y auditivas, en comparación con el uso de un solo tipo de sensor.
* Existe una relación positiva entre la potencia del sensor y la distancia de detección: a mayor potencia, mejora la precisión en el cálculo de la distancia hasta la fuente del sonido, dentro de los límites establecidos por la potencia máxima del sensor.
  1. **Operacionalización de variables**

| Variables | **Dimensión** | **indicador** |
| --- | --- | --- |
| Variables Independientes |
| Sensor acústico | Detección de señales sonoras | Frecuencia (Hz), Nivel de sonido (dB) |
| imágenes | Reconocimiento facial mediante CNN | Precisión de identificación (%) |
| Variables dependientes | **Dimensión** | **Indicador** |
| potencia del sensor acústico | Alcance de detección | Voltaje (v), Distancia (m) |
| sensibilidad del sensor acústico | Capacidad de distinguir sonidos relevantes (gritos/golpes) | Decibeles (dB) |
| precisión del informe generado | Exactitud de la localización basada en sensores | Porcentaje de aciertos (%) |
| precisión del sensor facial | Reconocimiento facial en condiciones adversas | Porcentaje de detección / distancia euclidiana |

1. **Metodología**
   1. **Descripción del enfoque metodología**

La presente investigación tiene un enfoque cuantitativo, de tipo aplicado y explicativo, con un diseño metodológico experimental.

Se implementará un robot controlado remotamente, equipado con sensores acústicos e imagen, con el propósito de evaluar su capacidad para localizar personas atrapadas en entornos simulados.

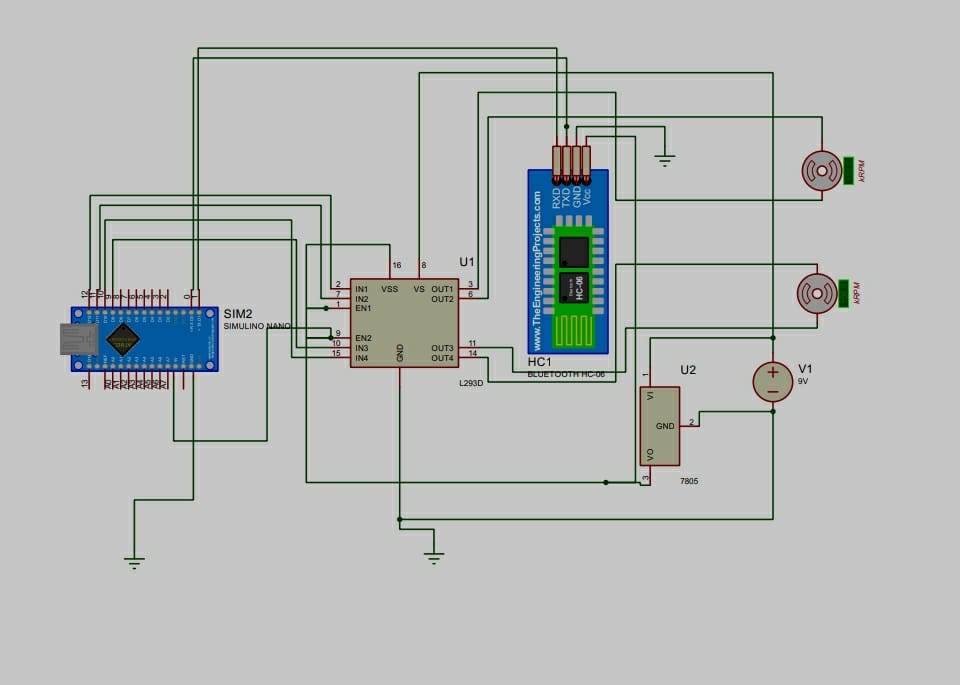
El sistema incorporará un modelo de inteligencia artificial entrenado, este se encargará de ejecutar tareas de procesamientos de datos sensoriales, como el filtrado de los ruidos ambientales y la detección de señales humanas (grito, golpes, etc.). Además, se incluirá un módulo de visión computacional el cual permitirá interpretar las imágenes captadas por el sensor facial para reconocer rostros o siluetas humanas, incluso en condiciones de baja iluminación.

Para ello, se diseñarán diferentes escenarios de colapso estructural, variando condiciones como el nivel de ruido ambiental, la iluminación y la presencia de obstáculos.

En cada prueba se registraron datos como la distancia de detección, la precisión de identificación y el tiempo estimado para localizar a una persona.

Los datos recopilados serán analizados para validar las hipótesis específicas y determinar si existe una mejora significativa al utilizar ambos sensores en comparación de uno solo.

* 1. **Esquema detallado del circuito**

****

* 1. **procedimiento del vehículo:**

procedimiento de construcción

1. Ensamblar los dos chasis uniendolos con el cintillo.
2. Colocar la llantas a los motores DC
3. Soldar en paralelo los condensadores 104 en los polos del motor
4. Colocar los motores en la parte lateral del frontal con cintillos
5. Agregar la rueda loca al chasis y asegurarlo.
6. Colocamos el Arduino nano en la parte trasera del protoboard, el L293D en el centro, a continuación se colocará el módulo bluetooth HC-06 y luego el I7805CV de tal manera que facilite las conexiones al protoboard.
7. Colocar el soporte de baterías de litio.

Ya tenemos armado y distribuido los componentes y la estructura

1. Realizar las conexiones de los cables en la placa mediante soldadura con estaño para que sea más resistente.
2. Primero colocar dos cables uno rojo y uno negro, al positivo y negativo de los dos lados del protoboard, interconectando los lados laterales del protoboard.
3. Comenzando con los cables colocamos un cable al pin uno del L293D y lo colocamos en la placa de experimentación.
4. Colocamos un cable al pin 2 del L293D al pin 12 del Arduino.
5. Colocamos el pin 3 del L293D al motor dc del lado derecho
6. Colocar el pin 4 del L293D al lado negativo de la placa de experimentación.
7. Colocar el pin 5 del l293D al lado negativo de la placa de experimentación.
8. Colocar el pin 6 del L293D al otro cable del motor derecho
9. Colocar el pin 7 del L293D al pin 11 del Arduino
10. Colocar el pin 8 del L293D al INPUT del lm7805
11. Colocar el pin 9 del L293D al lado positivo de la placa de experimentación.
12. Colocar el pin 10 del L293D al pin 9 del Arduino
13. Colocar el pin 11 del L293D al motor izquierdo
14. Colocar el pin 12 del L293D al lado negativo de la placa de experimentación.
15. Colocar el pin 13 del L293D al lado negativo de la placa de experimentación.
16. Colocar el pin 14 del L293D al otro cable del motor izquierdo
17. Colocar el pin 15 del L293D al pin 9 del Arduino nano
18. Colocar el pin 16 del L293D al lado positivo de la placa de experimentación.
19. Colocar el ground del lm7805 al negativo de la placa de experimentación.
20. Colocar el output del lm7805 al positivo de la placa de experimentación.
21. Para alimentar el HC-06 se coloca el VCC del HC-06 al positivo de la placa de experimentación y el ground del HC-06 al negativo de la placa de experimentación..
22. El pin Tx del HC06 se coloca al pin 0 Rx del Arduino nano
23. El pin Rx del Hc06 se coloca al pin 1 Tx del Arduino nano
24. Luego en los pines power del Arduino el pin de 5v se coloca al lado positivo de la placa de experimentación y uno de los pines ground del Arduino se coloca en el lado negativo de la placa de experimentación.
25. Luego de completar las conexiones se carga el programa al Arduino, sin embargo, antes de cargar se tiene que desconectar las entradas de 5v y ground del lado de power del Arduino nano.
26. Una vez cargado el programa, debemos descargar una aplicación Bluetooth Rc controller.
27. Una vez completo todo el armado, se desconecta el cable USB del puerto USB del Arduino y se colocan de nuevo los cables power 5v y ground a su lugar.
28. Luego colocamos las dos baterías de litio de 3,7v pero al estar en serie se sumarían los 2 así que tendríamos un voltaje de 7.4 V entonces colocamos el lado negativo a de las baterías al lado negativo de la placa de experimentación y el lado positivo al INPUT del lm7805, de esa forma logramos encenderlo.
29. Luego con la ayuda de la aplicación, sincronizamos el módulo bluetooth con el celular y se prueban los motores para determinar el sentido del giro, en caso no gire de acuerdo a lo planificado se invierte los cables del motor para que pueda girar en el sentido correcto.
    1. **Funcionamiento del circuito**

Alimentación principal

* Las dos baterías de litio de 3.4v que están conectadas en serie proporcionan un voltaje de 7.4v.
* Este voltaje entra por el INPUT del LM7805 que regula a 5v para alimentar componentes más sensibles como el Arduino y el módulo HC-06.
* El OUTPUT del lm7805 va al positivo del protoboard y el gnd al negativo.

Arduino Nano

* El Arduino se alimentará del protoboard con 5v y el gnd al lado negativo de la protoboard.
* Controla las direcciones de los motores enviando señales a los pines del L293D.
* El Arduino se comunicará con el HC-06 por medio del UART,

Modulo Bluetooth HC06

* Se alimenta con los 5v y el ground de la protoboard
* El RX del HC-06 se conecta con el pin 0 TX del Arduino y el pin TX DEL HC-06 con el pin 1 Rx del Arduino.
* El módulo HC-06 recibe los comandos desde el celular y este envía los comandos al Arduino para que los interprete.

Control de motores L293D

* El l293 es un puente H doble que se usa para controlar motores
* Los pines 2,7,10,15 recibirán las señales del Arduino
* Los pines de salida del l293d 3,6,11,14 van a los motores dc que están al lado izquierdo y derecho de la estructura
* El pin 8 se conecta directamente al voltaje de 7.4v ya que su rango de voltaje de entrada es de 4.5v a 36v
* El pin 16 se conecta a los 5v para que esté alimentado la lógica del L293D

Motores DC

* Se conectan directamente a los pines de salida del L293D que son los pines 3,6,11,14.
* Tiene un condensador 104 que ayuda a filtrar el ruido eléctrico.
* El sentido de giro depende de la señal que envía el Arduino.

El código para el control del robot por bluetooth se encuentra el la parte de anexos

EXPLICACION DE FUNCIONAMIENTO

El proyecto cuenta con múltiples componentes electrónicos entre ellos una placa Arduino nano, una cámara esp32 que nos permitirá identificar los parámetros que se desea obtener de acuerdo con la programación que se le efectuó y para lo cual está diseñado.

DESCRIPCIÓN FUNCIONAMIENTO CIRCUITO Y PRUEBAS

El circuito total funciono correctamente en la parte de la alimentación o principal para los componentes por lo que el circuito hay componentes que utilizan 5v y otros que usan un poco más que eso para ello y sacar provecho al máximo las baterías emiten 7,4v se introduzco el reductor de voltaje lm7805 que nos permite reducir el voltaje a 5v.

Y para la parte del control se introdujo un arduino nano que nos permite enviar y recibir las señales para que pueda funcionar trabajando con el L293D y el receptor bluetooth HC-06.

El receptor bluetooth se comunica con la aplicación bluetooth RC controller y esta envía los comandos al HC 06 y los transmite al arduino y este a su vez lo envía al L293D.

El L293D recibe las señales del arduino y las dirige a los motores que son alimentados con corriente DC mayor a 5v y así pueda tener un voltaje aceptable y funcione eficientemente los motores.

RESULTADOS

Se realizó los correctos cálculos para la aplicación de los voltajes correspondientes en los componente electrónicos calculando correctamente con baterías de 3.7v y como son 2 en serie son 7.4v que son suficientes.

DISEÑO DE LA TARJETA DE CONTROL PRINCIPAL

El componente que hace que todo se haga posible es el microcontrolador que es que será el que se encarga de recibir las señales que vienen del exterior y entenderlas y en base a ello envía una de las de respuesta y puede funcionar adecuadamente el circuito.

* 1. **Diseño del sistema de detección de objetos con inteligencia artificial**

El componente principal de este sistema es el modelo de red neuronal convolucional YOLOv11, el cual fue seleccionado por su alto rendimiento en tareas de detección en tiempo real. Este modelo tiene la capacidad de identificar múltiples objetos en una misma imagen con una precisión elevada y en tiempos de procesamiento reducidos, lo que lo hace ideal para aplicaciones en intersecciones urbanas.

Para el desarrollo del sistema se empleó una base de datos compuesta por imágenes del dataset Coco. La clase objetivo fue “person” y se utilizó la herramienta Roboflow para anotar manualmente las imágenes.

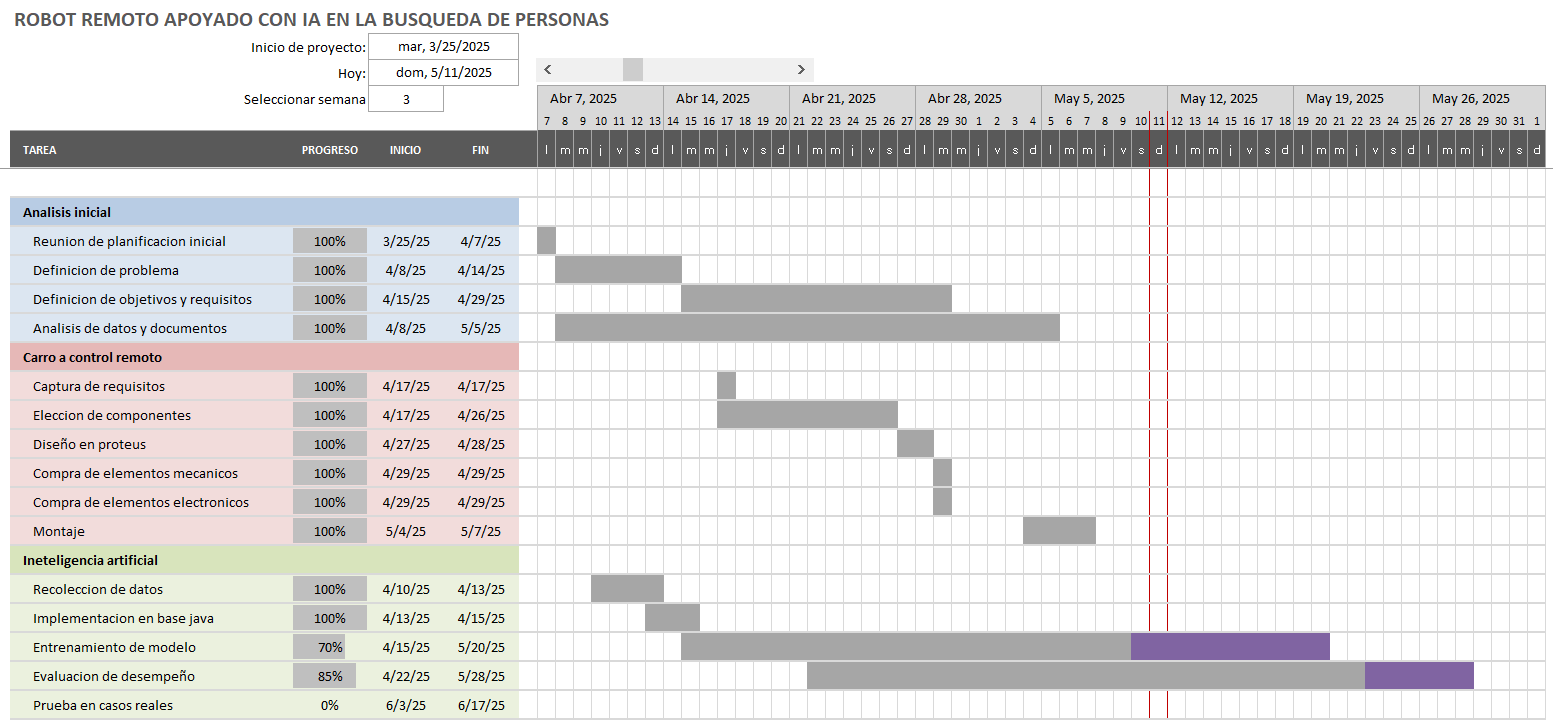
El modelo YOLOv11 fue entrenado utilizando la biblioteca PyTorch sobre una GPU NVIDIA (modelo RTX 3060). Se utilizó una resolución de entrada de 640x640 píxeles, un tamaño de lote de 10 imágenes, y una tasa de aprendizaje de 0.01 durante 200 épocas. El entrenamiento se realizó en Visual Studio Code.

Una vez completado el entrenamiento, se procedió a realizar las pruebas preliminares. Se cargó el modelo en una aplicación desarrollada en Python con OpenCV, que permitía el análisis cuadro por cuadro del video en vivo proveniente de la cámara. El sistema detectaba correctamente a las personas, generando una alerta visual en pantalla cuando una persona salía en cámara. Esto validó que el modelo funcionaba de acuerdo a los requerimientos establecidos.

Resultados

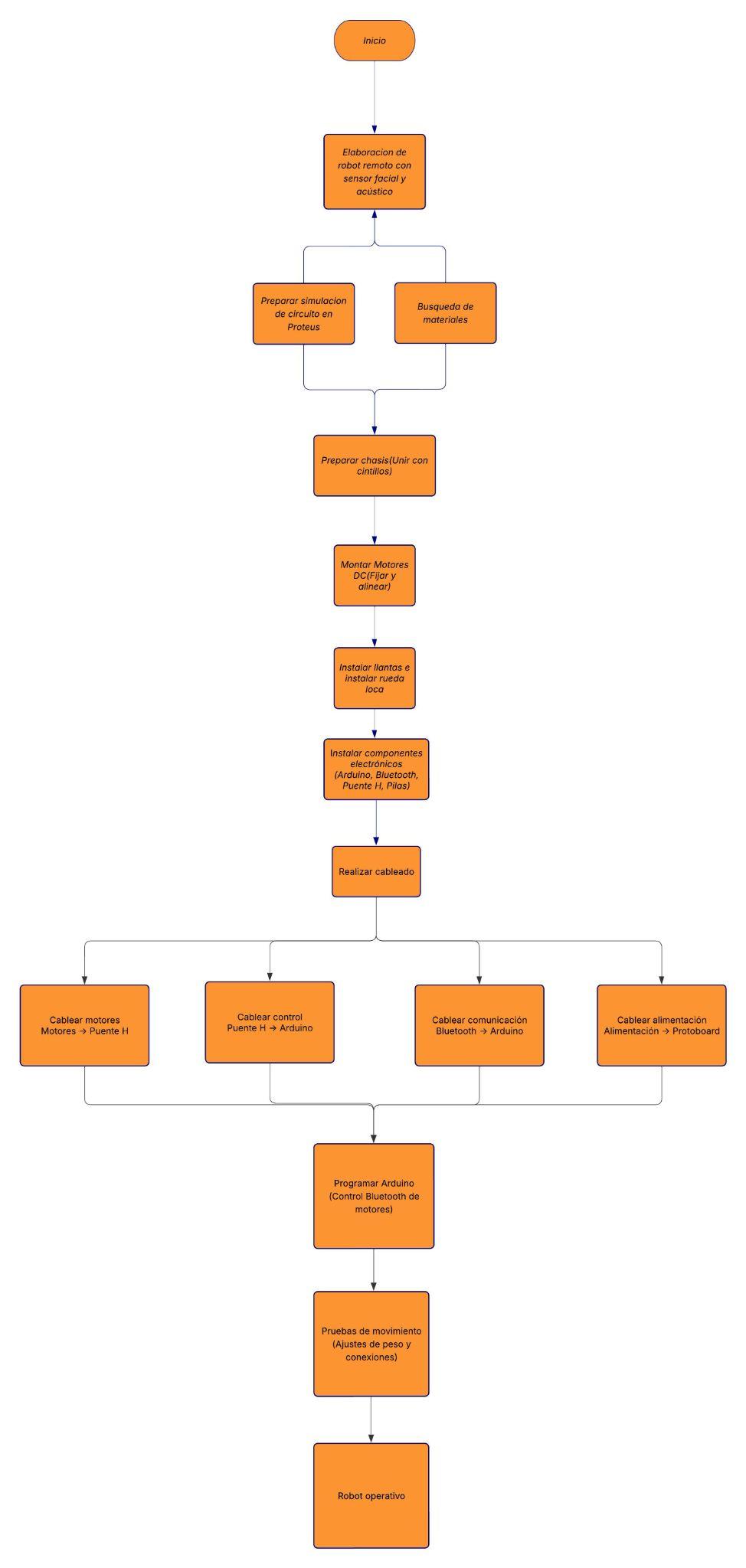
El modelo obtuvo una precisión promedio (mAP) del 88.6% y una velocidad de inferencia de 15.5 milisegundos por imagen, lo cual permite su aplicación en tiempo real. Las pruebas en entorno urbano mostraron una detección estable y precisa, incluso bajo condiciones variables de iluminación.

1. **Cronograma y disgregación de actividades**

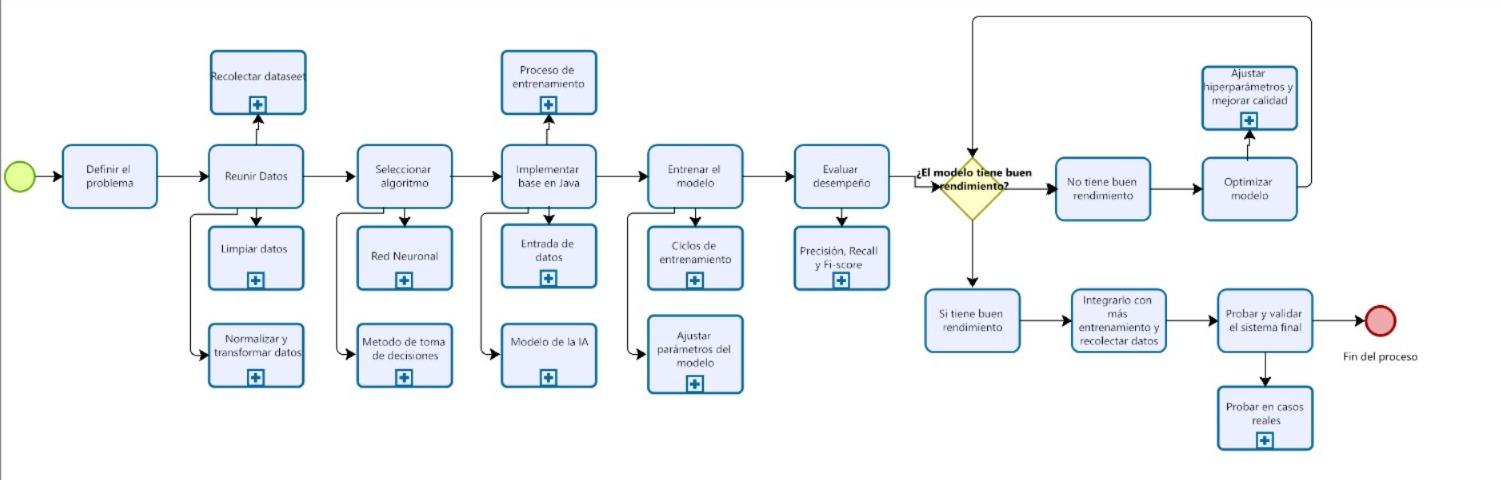
****

**DIAGRAMAS DE FLUJO:**

1. **Diagrama del vehículo:**



1. **Diagrama de la IA:**



1. **ANEXOS**
   1. **Matriz de consistencia**

Robot remoto apoyado con IA en la búsqueda de personas con sensor facial y acústico

| **Problemas** | **Objetivos** | **Hipótesis** | **Variables** | **Metodología** |
| --- | --- | --- | --- | --- |
| **Problema general:**  ¿De qué manera un robot remoto con inteligencia artificial, equipado con sensores faciales y acústicos, puede mejorar la localización de personas en situaciones de derrumbes o colapsos estructurales? | **Objetivo general:**  Reducir el tiempo de búsqueda en situaciones de rescate para brindar apoyo a los rescatistas. | **Hipótesis general:**  El uso de un robot equipado con sensores ayudado con IA mejora la eficacia en la localización de personas, al optimizar la detección de señales humanas y reducir el tiempo de búsqueda. | **Independientes:**  Sensor acústico  Sensor facial  Imágenes | La presente investigación tiene un enfoque cuantitativo, de tipo aplicado y explicativo, con un diseño metodológico experimental.  Se implementará un robot controlado remotamente, equipado con sensores faciales y acústicos, con el propósito de evaluar su capacidad para localizar personas atrapadas en entornos simulados.  El sistema incorporará un modelo de inteligencia artificial entrenado, este se encargará de ejecutar tareas de procesamientos de datos sensoriales, como el filtrado de los ruidos ambientales y la detección de señales humanas (grito, golpes, etc.). Además, se incluirá un módulo de visión computacional el cual permitirá interpretar las imágenes captadas por el sensor facial para reconocer rostros o siluetas humanas, incluso en condiciones de baja iluminación.  Para ello, se diseñarán diferentes escenarios de colapso estructural, variando condiciones como el nivel de ruido ambiental, la iluminación y la presencia de obstáculos.  En cada prueba se registraron datos como la distancia de detección, la precisión de identificación y el tiempo estimado para localizar a una persona.  Los datos recopilados serán analizados para validar las hipótesis específicas y determinar si existe una mejora significativa al utilizar ambos sensores en comparación de uno solo. |
| **Problemas específicos:**  ¿Cómo influye el nivel de entrenamiento de la inteligencia artificial en el funcionamiento del robot durante la localización de personas en estructuras colapsadas?  ¿Cómo influye la precisión del sensor facial en la detección de personas atrapadas en entornos con baja visibilidad o estructuras colapsadas?  ¿De qué manera la combinación de sensores faciales y acústicos mejora la localización de personas en comparación con el uso de un solo sensor?  ¿Cuál es la relación entre la potencia del sensor y la distancia(m) a la que se encuentra una persona de una persona? | **Objetivos específicos:**  Determinar cómo influye el nivel de entrenamiento de la IA con la capacidad de localización de personas.  Determinar el nivel de precisión del sensor facial pese a la poca visibilidad del ambiente.  Determinar la eficiencia de dos sensores a comparación de un sensor.  Determinar posición por medio de la potencia y la distancia. | **Hipótesis específicas:**  A mayor nivel de entrenamiento de la IA, mejor será la capacidad de localización de las personas bajo estructuras colapsadas.  El sensor facial presenta una leve disminución de precisión bajo condiciones de baja iluminación, sin embargo, mantiene un rendimiento lo suficientemente alto como para mejorar la capacidad de localización del robot en entornos colapsados.  La integración de sensores faciales y acústicos incrementará la capacidad de localización del robot al combinar funciones visuales y auditivas, en comparación con el uso de un solo tipo de sensor.  Existe una relación positiva entre la potencia del sensor y la distancia de detección: a mayor potencia, mejora la precisión en el cálculo de la distancia hasta la fuente del sonido, dentro de los límites establecidos por la potencia máxima del sensor. | **Dependientes:**  Potencia del sensor acústico(distancia)  Sensibilidad del sensor acústico: Nivel de exactitud del sensor al detectar un sonido como gritos o golpes  Precisión del informe generado: Que tan exacta es la información que entrega el sistema (porcentaje de aciertos en ubicación o identificación)  Precisión del sensor facial: Nivel de exactitud del sensor al identificar rostros |

* 1. **Presupuesto y componentes**

**Tabla de presupuesto asignado al proyecto:**

| **MATERIALES** | **PRECIO** |
| --- | --- |
| Chasis de plástico para soporte | S/ 25.00 |
| Cintillos de sujeción x25u | S/ 3.6 |
| Arduino Nano | S/ 30.00 |
| 2 motores DC | S/ 12.00 |
| 2 ruedas de plástico para los motores DC | S/ 4.00 |
| Receptor bluetooth | S/ 20.00 |
| L293B | S/ 3.50 |
| Cables puente o conjunto de cables x40 | S/ 4.50 |
| Regulador de voltaje (LM7805) | S/ 1.50 |
| Protoboard | S/ 10.00 |
| Rueda de soporte (rueda loca) | S/ 3.00 |
| Cámara ESP32 CAM | S/ 55.00 |
| Sensor de sonido KY-038 | S/ 7.00 |
| Soporte de baterías de 3,7v | S/ 3.00 |
| 2 pilas de litio de 3.7v | S/ 16.00 |
| **TOTAL:** | **S/ 198.10** |

**Componentes:**

* Chasis del carro o vehículo de plástico:

Sirve como la estructura física del Robot, proporcionando una base estable sobre la cual se montaran los componentes.

* Cintillos de sujeción:

Servirán para mantener sujetos algunos componentes, incluso durante la fase de prueba para la movilidad del robot. Una vez culminado esa parte se podría a cambiar por soldadura o silicona

* Arduino Nano

Es un microcontrolador que actuará de cerebro del robot, el cual será capaz de leer sensores, procesar información y/o controlar los motores dependiendo del programa cargado.

* Microcontrolador (ESP32-WROOM-32E) / Como segunda opción:

Procesará la información de todos los sensores, controlará los actuadores (motores, brazos, etc.), gestionará la comunicación inalámbrica y ejecutará los algoritmos de búsqueda y rescate.

* 2 motores DC: (TT 6V)

Serán responsables del movimiento del robot.

* 2 ruedas de plástico para los motores DC (D65mm)
* condensadores tipo lenteja 104
* Receptor bluetooth: (CH-06)

Este dispositivo permite la comunicación de manera unidireccional, el cual se encargará de recibir las señales que se le enviaran a partir de otro dispositivo como puede ser por una laptop o un celular.

* Controlador de motor DC (L293D):

El circuito puente H nos permite controlar la dirección y velocidad del motor de corriente continua, este cambiará la polaridad de la corriente que pasa por el motor para determinar en qué dirección debe de girar.

* Cables puente o conjunto de cables:

permite realizar las conecciones de los dispositivos en el protoboard

* Protoboard

Se utiliza para montar y probar el circuito sin la necesidad de soldar, permitiendo realizar cambios durante las pruebas, una vez cumpla con su propósito se optara por cambiarlo para mejorar el rendimiento del auto.

* Rueda de soporte: rueda loca (D25mm)

será utilizada como un soporte auxiliar para mejorar la estabilidad del robot, debido a que este se mueve en todas las direcciones sin restricción.

* Cámara ESP32 CAM
* Sensor de sonido KY-038
* Regulador de voltaje L7805CV
* Soporte de baterías 3.7v

Lugar en donde se colocará una cantidad determinada de pilas para el funcionamiento del dispositivo.

* 2 baterías de litio de 3.7v

Son la fuente de alimentación para el sistema.

**Código para el movimiento del auto:**

**int mo1ad=10; //motor 1 - adelante**

**int mo1at=9; // motor 1 - atrás**

**int mo2ad=12; // motor 2 - adelante**

**int mo2at=11; //motor 2 - atrás**

**char val='S'; //variable para almacenar comando inicial**

**void setup(){ // configurar los pines como salidas**

**pinMode(mo1ad,OUTPUT);**

**pinMode(mo1at,OUTPUT);**

**pinMode(mo2ad,OUTPUT);**

**pinMode(mo2at,OUTPUT);**

**Serial. Begin(9600); //se inicia la comunicación a 9600 baudios**

**}**

**void loop(){ // verifica si hay datos disponibles en el puerto Serial**

**if (Serial.available()>0){**

**val =Serial.read(); // leera el caracter recibido**

**Serial.println(val);**

**}**

**if(val=='F'){ // adelante Forward**

**digitalWrite(mo1ad,HIGH);**

**digitalWrite(mo1at,LOW);**

**digitalWrite(mo2ad,HIGH);**

**digitalWrite(mo2at,LOW);**

**}**

**else if(val=='B'){ // atras Backward**

**digitalWrite(mo1ad,LOW);**

**digitalWrite(mo1at,HIGH);**

**digitalWrite(mo2ad,LOW );**

**digitalWrite(mo2at,HIGH);**

**}**

**else if (val=='L'){ // girar a la izquierda Left**

**digitalWrite(mo1ad,LOW);**

**digitalWrite(mo1at,LOW);**

**digitalWrite(mo2ad,HIGH );**

**digitalWrite(mo2at,LOW);**

**}**

**else if(val=='R'){ //girar a la derecha Right**

**digitalWrite(mo1ad,HIGH);**

**digitalWrite(mo1at,LOW);**

**digitalWrite(mo2ad,LOW );**

**digitalWrite(mo2at,LOW);**

**}**

**else if (val=='S'){ // detenerse Stop**

**digitalWrite(mo1ad,LOW);**

**digitalWrite(mo1at,LOW);**

**digitalWrite(mo2ad,LOW );**

**digitalWrite(mo2at,LOW);**

**}**

**else if(val=='I'){ // diagonal derecha adelante**

**digitalWrite(mo1ad,HIGH);**

**digitalWrite(mo1at,LOW);**

**digitalWrite(mo2ad,LOW );**

**digitalWrite(mo2at,LOW);**

**}**

**else if (val=='G'){ // diagonal izquierda adelante**

**digitalWrite(mo1ad,LOW);**

**digitalWrite(mo1at,LOW);**

**digitalWrite(mo2ad,HIGH );**

**digitalWrite(mo2at,LOW);**

**}**

**else if(val=='J'){ // diagonal para atras derecha**

**digitalWrite(mo1ad,LOW);**

**digitalWrite(mo1at,HIGH);**

**digitalWrite(mo2ad,LOW );**

**digitalWrite(mo2at,LOW);**

**}**

**else if (val=='H'){ // diagonal atras izquierda**

**digitalWrite(mo1ad,LOW);**

**digitalWrite(mo1at,LOW);**

**digitalWrite(mo2ad,LOW );**

**digitalWrite(mo2at,HIGH);**

**}**

**}**

**Referencias**

* Álvarez-Cedillo, J., Álvarez-Sánchez, T., Sandoval, R., Sandoval-Gutierrez, J., Nava-Vega, A., (2019). *Diseño de un robot rescatista para terremotos en México*. Research in Computing Science, 148(2), 31-41. <https://doi.org/10.13053/rcs-148-2-2>
* Gordon, C., Encalada, P., Lema, L., León, D., Chicaiza, D., (2020). Human Rescue Based on Autonomous Robot KUKA Youbot with ROS and Object Detection. *In: Bi, Y., Bhatia, R., Kapoor, S. (eds) Intelligent Systems and Applications. IntelliSys 2019*. Advances in Intelligent Systems and computing, vol 1037. Springer, Cham. <https://doi.org/10.1007/978-3-030-29516-5_58>
* García-Sanmartín, JF, Cruz Ulloa, C., Cerro, J., Barrientos, A., (2024). Active robotic search for victims using ensemble deep learning techniques. *Machine learning: science and technology*. 5(2). <https://doi.org/10.1088/2632-2153/ad33df>
* Mendoza Talledo, V., Meza Capcha, E., (2025, 23 febrero). *Tragedia en Real Plaza Trujillo por caída de techo del patio de comidas: Bomberos elevaron a ocho la cifra de fallecidos*. Infobae. <https://www.infobae.com/peru/2025/02/22/panico-en-real-plaza-trujillo-por-caida-de-techo-del-patio-de-comidas-hay-heridos-y-no-se-descartan-personas-atrapadas/>
* Silva, R. (2025, 24 febrero). Tragedia en el Real Plaza Trujillo: Falla en el sistema de drenaje, mantenimiento y supervisión serian principales causas. Infobae.<https://www.infobae.com/peru/2025/02/24/tragedia-en-el-real-plaza-trujillo-falla-en-el-sistema-de-drenaje-mantenimiento-y-supervision-serian-principales-causas/>
* Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press. <https://mitpress.mit.edu/author/ian-goodfellow-22586/>
* Nair, V., & Hinton, G. E. (2010). Rectified Linear Units Improve Restricted Boltzmann Machines. *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 807–814. <https://www.cs.toronto.edu/~fritz/absps/reluICML.pdf>
* Duda, R. O., Hart, P. E., & Stork, D. G. (2000). *Pattern Classification* (2nd ed.). Wiley-Interscience.  
  <https://www.wiley.com/en-us/Pattern%2BClassification%2C%2B2nd%2BEdition-p-9780471056690>
* Oppenheim, A. V., Willsky, A. S., & Nawab, S. H. (1996). *Signals and Systems* (2nd ed.). Prentice Hall. <https://www.amazon.com/Signals-Systems-2nd-Alan-Oppenheim/dp/0138147574>
* Kingma, D. P., & Ba, J. (2014). Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*. <https://arxiv.org/pdf/1412.6980>
* Thrun, S., Burgard, W., & Fox, D. (2005). *Probabilistic Robotics*. MIT Press. <https://mitpress.mit.edu/9780262201629/probabilistic-robotics/>
* StudySmarter. (2025). *Acoustic Sensors: Definition & Applications*. Recuperado de <https://www.studysmarter.co.uk/explanations/engineering/mechanical-engineering/acoustic-sensors/>
* Vezzetti, E., & Marcolin, F. (2012). Face recognition - an overview. *ScienceDirect*. Recuperado de<https://www.sciencedirect.com/topics/computer-science/face-recognition>
* Oracle. (2025). *Java*. Recuperado de<https://www.java.com>
* Amazon Web Services. (2025). *¿Qué es la inteligencia artificial?* Recuperado de<https://aws.amazon.com/es/what-is/artificial-intelligence/>
* IBM. (2024). *¿Qué es el aprendizaje automático?* Recuperado de <https://www.ibm.com/es-es/topics/machine-learning>
* Gregersen, E. (Ed.). (2025). *Computer vision*. Encyclopædia Britannica. Recuperado de<https://www.britannica.com/technology/computer-vision>[Encyclopedia Britannica](https://www.britannica.com/technology/computer-vision?utm_source=chatgpt.com)
* Balladares Rocha, M. G. (2020). *Proteus: Aplicaciones en circuitos*. Recuperado de <https://es.slideshare.net/slideshow/7-proteus-aplicaciones-en-circuitospdf/252543873>
* Ollama. (2025). *Ollama - Ejecuta Modelos de Lenguaje Localmente*. Recuperado de<https://ollama.org/es/>
* Microchip Technology Inc. (2025). *MPLAB® X IDE*. Recuperado de <https://www.microchip.com/en-us/tools-resources/develop/mplab-x-ide>
* Ivan Espinoza.(2021, 13 de julio). Carrito a control remoto con Arduino / Bluetooth / Android, en Protoboard [video], Youtube. <https://youtu.be/hqNz4wcLMy8?si=JBWpqfcUEq4IVhJP>