

Descripción funcional y técnica del sistema

Visión general del comportamiento

El sistema implementa un comportamiento autónomo para un robot móvil basado en una **máquina de estados finitos**, cuyo objetivo es desplazarse entre distintas habitaciones, identificar información semántica situada en las paredes (números), y atravesar puertas para continuar la exploración.

La arquitectura combina:

- **Percepción geométrica** mediante sensor LiDAR.
- **Percepción visual** basada en procesamiento de imagen clásico y aprendizaje profundo.
- **Control cinemático reactivo** para navegación estable.

El diseño separa claramente las responsabilidades de percepción, decisión y control, garantizando un comportamiento robusto y fácilmente interpretable.

El ciclo general de funcionamiento en cada habitación es el siguiente:

1. Localización del centro geométrico de la habitación (LOCALISE)
2. Aproximación controlada al centro (GOTO_ROOM_CENTER)
3. Exploración visual de las paredes y lectura de números (TURN)
4. Búsqueda y aproximación a una puerta (GOTO_DOOR)
5. Alineación precisa con la puerta (ORIENT_TO_DOOR)
6. Cruce de la puerta y transición a la siguiente habitación (CROSS_DOOR)

Tras el cruce, el ciclo se reinicia en la nueva habitación.

Localización del centro de la habitación (LOCALISE)

Al entrar en una nueva habitación, el robot activa el estado **LOCALISE**, cuyo objetivo es **detectar el centro geométrico de la sala** a partir de la información proporcionada por el sensor LiDAR.

En este estado:

- Se analizan las paredes detectadas mediante percepción geométrica.
- A partir de dichas paredes se intenta estimar el centro de la habitación.
- Si el centro no puede estimarse de forma fiable:
 - El robot gira sobre sí mismo con una velocidad angular constante.
 - El giro está limitado temporalmente para evitar oscilaciones indefinidas.
 - Este comportamiento permite barrer la geometría del entorno y estabilizar la detección.

- Cuando el centro es detectado con éxito:
 - La estimación del centro se almacena internamente.
 - El sistema transita al estado **GOTO_ROOM_CENTER**.

Este estado se encarga exclusivamente de la **percepción**, sin realizar desplazamientos hacia el centro.

Aproximación al centro de la habitación (GOTO_ROOM_CENTER)

En el estado **GOTO_ROOM_CENTER**, el robot asume que el centro de la habitación ya ha sido localizado.

El comportamiento consiste en:

- Calcular la distancia y el ángulo al centro en el sistema de referencia del robot.
 - Aplicar un control desacoplado:
 - La velocidad angular es proporcional al error angular.
 - La velocidad lineal se modula mediante un freno exponencial en función del ángulo, evitando trayectorias laterales inestables.
 - Cuando la distancia al centro es inferior a un umbral predefinido:
 - El robot se considera correctamente centrado.
 - La pose del robot se ajusta al centro nominal de la habitación.
 - Se reinician las variables relacionadas con la percepción visual.
 - Se produce la transición al estado **TURN**.
-

Exploración visual y lectura de números (TURN)

En el estado **TURN**, el robot realiza una exploración visual de las paredes girando sobre sí mismo durante un tiempo limitado.

Durante este giro se ejecuta de forma periódica un pipeline de visión artificial y clasificación, cuyo objetivo es detectar y reconocer números situados en las paredes.

Pipeline de visión artificial e inteligencia artificial

La percepción visual está diseñada como un proceso jerárquico y eficiente, combinando filtrado clásico por imagen con clasificación basada en aprendizaje profundo.

1. Adquisición de imagen

- Se obtiene una imagen RGB completa desde la cámara omnidireccional.

- La imagen se convierte a escala de grises para reducir la dimensionalidad y eliminar la dependencia del color.
- Se mantiene una copia de la imagen original para depuración visual.

2. Prefiltrado rápido (early exit)

Para evitar computación innecesaria:

- Se genera una versión reducida de la imagen.
- Se aplica un umbral de intensidad para detectar regiones oscuras.
- Si no se detectan contornos negros significativos, el frame se descarta inmediatamente.

Este paso permite que el sistema sea reactivo y eficiente en tiempo real, incluso cuando no hay información relevante en la escena.

3. Detección del marco negro

Cuando el prefiltrado es positivo:

- Se aplica un suavizado gaussiano para reducir ruido.
- Se realiza una segmentación por intensidad para detectar regiones negras.
- Se aplican operaciones morfológicas de cierre para consolidar regiones fragmentadas.
- Se extraen contornos y se filtran según:
 - Área mínima relativa a la imagen.
 - Relación de aspecto compatible con un marco aproximadamente cuadrado.

Se selecciona el contorno más probable como marco candidato.

4. Extracción de la región interior (ROI)

- Se define una región de interés interior al marco detectado, aplicando un margen para evitar los bordes.
- Se descartan regiones con bajo contraste (desviación estándar baja), evitando clasificar ruido o sombras.

Esta etapa garantiza que solo se procesan regiones con información visual suficiente.

5. Segmentación del dígito

Dentro de la ROI:

- Se normaliza dinámicamente el contraste.
- Se aplica una umbralización adaptativa gaussiana para separar el dígito del fondo.
- Se detectan contornos y se selecciona el contorno dominante:
 - Área significativa respecto a la ROI.
 - Sin contacto con los bordes (para evitar confusión con el marco).

El área seleccionada se considera el dígito.

6. Normalización para la red neuronal

- El dígito segmentado se reescala a 28×28 píxeles.
- Se mantiene una única canalización en escala de grises.
- La imagen se normaliza a valores en el rango $[0,1]$.

Esto permite una compatibilidad directa con el dataset MNIST.

7. Clasificación mediante red neuronal convolucional

- Se utiliza una red neuronal convolucional entrenada previamente con MNIST.
- La red recibe un tensor de forma (1, 1, 28, 28).
- La inferencia se ejecuta en CPU sin cálculo de gradientes.
- La salida se transforma en probabilidades mediante softmax.
- Se selecciona el dígito con mayor probabilidad y su confianza asociada.

Solo se acepta la clasificación si la confianza supera un umbral definido.

8. Gestión temporal de la información semántica

- El último dígito detectado se almacena junto con su confianza y un timestamp.
- La interfaz de acceso a la información devuelve el valor únicamente si:
 - El dígito es válido.
 - La detección no ha caducado.
- Esto evita reutilizar detecciones antiguas o falsas.

Además, el sistema garantiza que **solo se lee un número por habitación**.

Al finalizar el tiempo máximo de exploración, o tras una detección válida, el sistema transita a **GOTO_DOOR**.

Búsqueda y aproximación a puertas (GOTO_DOOR)

En este estado, el robot busca puertas utilizando percepción LiDAR.

- Si no se detectan puertas:
 - El robot gira para ampliar su campo perceptivo.
- Si se detectan puertas:
 - Se selecciona la puerta más alineada con el eje frontal del robot.
 - Se calcula un punto previo a la puerta como objetivo intermedio.
 - El robot se aproxima a dicho punto mediante un controlador cinemático suave.

Cuando el punto previo a la puerta es alcanzado, se transita a **ORIENT_TO_DOOR**.

Alineación y cruce de la puerta (ORIENT_TO_DOOR y CROSS_DOOR)

En **ORIENT_TO_DOOR**, el robot ajusta su orientación hasta que el error angular con la puerta es inferior a un umbral.

Una vez alineado:

- Se transita al estado **CROSS_DOOR**.
- El robot avanza recto durante un intervalo de tiempo fijo, sin reevaluación perceptiva.

Tras finalizar el cruce:

- Se actualiza la habitación actual.
- Se reinician las variables internas.
- Se descarta la información del centro previo.
- El sistema vuelve al estado **LOCALISE**, reiniciando el ciclo en la nueva habitación.