




## Robot móvil para agarre de objetos por color, con visión artificial y control difuso Mamdani

### Mobile robot capable to grasp color objects using Mamdani fuzzy control and artificial vision

Dante Giovanni Sterpin Buitrago <sup>1</sup> Brayan Steven Mendivelso García <sup>2</sup> y Yeyson Alejandro Becerra Mora <sup>3</sup>

Fecha de Recepción: 25 de julio de 2024

Fecha de Aceptación: 20 de agosto de 2025

**Cómo citar:** Sterpin Buitrago., D.G., Mendivelso García., B.S., Becerra Mora., Y.A. (2025). Robot móvil para agarre de objetos por color, con visión artificial y control difuso Mamdani. *Tecnura*, 29(85), 34-62. <https://doi.org/10.14483/22487638.22525>

## Resumen


**Contexto:** este trabajo analizó la movilización autónoma de un robot diferencial simulado, por medio de un sistema de control PD, retroalimentado con una cámara a bordo. Tanto el controlador como el proceso de segmentación de imágenes en tiempo real utilizan un sistema de inferencia difusa tipo Mamdani.

**Metodología:** se dotó al robot con un sistema de percepción visual para identificar un objeto de cierto color, en la imagen capturada por su cámara, con el fin de que el controlador reaccione según la ubicación variable del objeto. Se hicieron pruebas con el objeto estático y en movimiento, además de modificar las condiciones de iluminación ambiental.


**Resultados:** se observó la efectividad de dichos sistemas difusos para movilizar el robot, ya que este logra acercarse y posicionarse para poder agarrar el objeto identificado en su campo visual. En el caso de no verlo, el robot panea hasta encontrarlo, si existe.

**Conclusiones:** los sistemas difusos empleados fueron suficientes y eficientes para el control de un robot móvil que requiere de la información cromática que provee una cámara a bordo. Su diseño, basado en términos lingüísticos que representan nociones intuitivas sobre el comportamiento necesario, permite introducir a estudiantes interesados en el campo de la robótica móvil con visión artificial. A futuro, se podría combinar con otros sistemas basados en lenguaje natural, en una sociedad artificial de robots.

**Palabras clave:** control difuso PD, lógica difusa, razonamiento difuso, visión artificial.

<sup>1</sup>Ingeniero electrónico, magíster en Ingeniería de Sistemas y Computación. Docente en la Corporación Unificada Nacional de Educación Superior (CUN) . Bogotá, Colombia. Email: [dante\\_sterpin@cun.edu.co](mailto:dante_sterpin@cun.edu.co)

<sup>2</sup>Ingeniero electrónico. Auxiliar de investigación en la Corporación Unificada Nacional de Educación Superior (CUN) . Bogotá, Colombia. Email: [brayan.mendivelso@cun.edu.co](mailto:brayan.mendivelso@cun.edu.co)

<sup>3</sup>Ingeniero en Automatización, máster en Automática y Robótica. Docente en la Corporación Unificada Nacional de Educación Superior (CUN) . Bogotá, Colombia. Email: [yeyson.becerra@cun.edu.co](mailto:yeyson.becerra@cun.edu.co)

## Abstract

**Context:** Autonomous mobilization of a simulated differential robot, by means of a PD control system with on-board camera feedback, where both the controller and the real-time image segmentation use Mamdani-type fuzzy inference.

**Methodology:** The robot was equipped with a visual perception system to identify an object of a specific color, in the image captured by its camera, so that the controller could react according to the object's changing position. Tests were carried out with the object both stationary and in motion, as well as varying ambient lightning conditions.

**Results:** The effectiveness of the fuzzy systems was observed in enabling the robot to move toward the object identified in its visual field in order to grasp it. If the object is not initially visible, the robot pans until it finds it, provided it exists.

**Conclusions:** The employed fuzzy systems were sufficient and efficient for controlling a mobile robot that relies on chromatic information provided by an onboard camera. Their design, based on linguistic terms representing intuitive notions of the required behavior, makes them suitable for introducing students interested in robotics and computer vision. In the future, it appears feasible to combine such systems with other natural language-based approaches within an artificial society of robots.

**Keywords:** artificial vision, fuzzy logic, fuzzy reasoning, PD fuzzy control.

---

## Introducción

El desplazamiento autónomo basado en percepción visual para un robot móvil es un desafío, debido a la extracción de características propias del entorno y al reconocimiento de objetos allí dentro. Garantizar el correcto funcionamiento, hace que este tipo de aplicaciones se tornen complejas (1). Una de las técnicas más empleadas en problemas de percepción es la de redes neuronales convolucionales (CNN, por su sigla en inglés), en donde básicamente se busca llevar a cabo tareas de segmentación semántica y detección de objetos. En trabajos como (2-4) entre otros, se busca un aprendizaje ambicioso que le permita los dispositivos robots segmentar e identificar un gran número de clases (p. ej.: objetos del entorno); estas redes requieren un gran esfuerzo computacional; sin embargo, una vez entrenadas pueden ser usadas para ejecutar múltiples tareas simples que requieran una o unas pocas clases para las que fue originalmente entrenada.

Debido a los múltiples elementos que interfieren en la percepción del entorno, y las actividades que el robot debe ser capaz de realizar, la navegación autónoma se convierte en una tarea realmente difícil. Con el fin de mejorar el desempeño del robot en su navegación autónoma, las redes neuronales cumplen un papel determinante. En trabajos como (5) y (6), se recurre a ellas para mejorar no solo la interpretación del entorno, sino fusiones de diversas representaciones visuales (segmentación, profundidad, distribución, clasificación de objetos/habitación) y fusiones de sensores (LiDAR y cámaras RGB), respectivamente.

Por su parte, en (7) se propone un entrenamiento conjunto de dos CNN para la estimación de la profundidad y posicionamiento de la cámara, a partir de secuencias de video no etiquetadas. En ocasiones, la iluminación del entorno puede cambiar drásticamente y generar fallos en la navegación del robot. Otros trabajos como (8) presentan un enfoque basado en aprendizaje para mejorar la odometría visual del sistema en condiciones de baja luminosidad.

Adicionalmente, métodos basados en *deep learning* han sido propuestos para estimar la posición y orientación de un objeto visto a través de una cámara RGB (9, 10). Teniendo una estimación de la localización de un objeto en particular, pueden ejecutarse tareas de manipulación con distintos tipos de sistemas robóticos. Por ejemplo, en (11) y (12) se ofrecen revisiones de manipuladores aéreos robóticos para desempeñar labores en el sector de la construcción o para realizar seguimiento o agarre de objetos en movimiento. En (13) se expone un marco de trabajo basado en visión por computador para estimar la configuración de manipuladores robóticos que no cuentan con sensores propioceptivos. Adicionalmente, aplicaciones orientadas al agarre de objetos empleando humanoides pueden verse en (14, 15) y (16).

Considerando la navegación autónoma basada en visión por computador, y que la fuente perceptiva de los datos proporciona información con cierto nivel de incertidumbre, bien sea por la calidad de los sensores usados o por condiciones del entorno que puedan afectar la percepción (17), debe recurrirse a técnicas que enfrenten el problema con buen desempeño, aun con la posible baja calidad de los datos que se perciban del entorno, en procura de un procesamiento de imagen lo más robusto posible, y que implemente la navegación autónoma, mientras se reconocen y se diferencien objetos de interés en el entorno. En (18) se plantea que este tipo de sistemas deben estar dotados de planificación de rutas, evitación, detección, reconocimiento y manipulación de objetos entre otras. En algunos casos, esos sistemas deben tener la capacidad de proporcionar constantemente información sobre la ubicación de objetos en movimiento, como lo muestran (6), quienes logran extraer información sobre sus características de desplazamiento, ubicación y dimensiones, con el fin de que la navegación autónoma sea eficiente.

Aplicar técnicas basadas en lógica difusa permite desarrollar soluciones sencillas y prácticas a problemas como el mencionado anteriormente, en donde la percepción artificial del entorno está inspirada en la percepción natural humana, además de requerirse comportamientos efectivos sobre la base de información imprecisa. En (19–22) y (23) se observa el uso de diferentes técnicas de control difuso para implementar sistemas de procesamiento de imágenes que garanticen efectividad en la diferenciación de colores, el grado de pertenencia en puntos clave que representan la clase de un objeto, y la segmentación de imágenes en regiones homogéneas, o que presenten una característica en común. La efectividad de estos sistemas depende de la calidad con la que se elijan aspectos como las variables de entrada y los parámetros de fuzzifi-

cación (20). Por otro lado, con la aplicación de sistemas de control basados en lógica difusa se da solución a problemas de navegación autónoma de vehículos, incluyendo el reconocimiento del entorno, evasión de obstáculos, planeación o seguimiento de trayectorias, para lo cual el control manual no presenta la misma estabilidad, certeza y precisión. Los desarrollos presentados en (24, 25) y (26), señalan cómo es posible plantear la navegación autónoma y obtener resultados óptimos mediante la aplicación de diferentes técnicas de control difuso.

En la actualidad, muchas labores que normalmente eran realizadas por humanos están siendo automatizadas por dispositivos robóticos, lo que implica el aumento del uso y desarrollo de nuevos sistemas que utilizan técnicas de control difuso en el ámbito industrial, en procesos como los presentados en (27) para la toma de decisiones inteligente en la selección de estrategias de navegación marítima; (28) para planificación de trayectorias en entornos dinámicos desconocidos; (29) para mantener a un vehículo de guiado automático dentro de un carril. Por ejemplo, en la selección de cultivos agrícolas se pueden aplicar técnicas de control difuso para el procesamiento de imagen, y así mejorar la eficiencia del proceso, disminuir costos y aumentar la productividad (30). Aplicaciones en el ámbito doméstico son también viables al considerar su uso en robots asistenciales que colaboren en tareas de diferenciación, reconocimiento y manipulación de objetos (18), a personas con algún tipo de discapacidad. En la mayoría de los casos, el uso de estos sistemas permite automatizar y mejorar la eficiencia en los procesos implicados, debido a la interacción en tiempo real que permiten establecer con un entorno físico.

La técnica de inferencia difusa de (31) se aplica en este trabajo para resolver el problema de navegación autónoma, a través del procesamiento digital de imágenes para la percepción visual a color con una cámara a bordo. En desarrollos de este tipo (20, 30) y (32), se observa que esta técnica brinda una amplia versatilidad en cuanto a la calibración del controlador, y ofrece un procesamiento de imagen sumamente detallado, sin incurrir en alto costo computacional ni ralentizar el control. En la actualidad, las redes neuronales convolucionales lideran el procesamiento de imagen (33), y en general, el control de robots está empleando técnicas basadas en *deep learning* (34); sin embargo, en la aplicación acá reportada se prefirió usar sistemas de lógica difusa, tipo Mamdani, con dos propósitos: aplicativo y prospectivo. Por un lado, se usaron para que un robot pueda detectar visualmente un objeto en tiempo real, con el fin de representar internamente su lejanía, y ubicación relativa, para acercarse a éste de tal manera que pueda agarrarlo. Por otro lado, al usar ese tipo de sistema lógico se tiene un algoritmo simple, capaz de vincular mecanismos numéricos y terminología lingüística, lo cual puede servir de base para que un robot logre describir verbalmente su propio comportamiento, por ejemplo, en una sociedad artificial de robots con procesos culturales de enseñanza y aprendizaje (35). De esta manera, en este artículo se reporta el diseño y los resultados de un robot controlado con visión artificial, que, además de lo ya mencionado, se quiere emplear en el contexto educativo como

un primer acercamiento a los fundamentos de la lógica difusa, y la robótica móvil basada en visión artificial, antes de usar técnicas con *deep learning*, para aplicaciones complejas.

El presente artículo comienza detallando, en “Metodología”, tanto el sistema de inferencia difusa para la visión artificial del robot, como los sistemas de control difuso que usan la respectiva información visual para mover sus motores. Luego, en “Resultados” se describe el comportamiento evidenciado por el robot, y en “Conclusiones”, se comenta hacia dónde se orienta esta aplicación.

## Metodología

Se empleó un robot Pioneer 3-DX dotado con una cámara abordo y un mecanismo de pinzas tipo montacarga, con lo cual es capaz de identificar en tiempo real en dónde se encuentra un objeto de cierto color, para acercarse a este de tal manera que, tras ubicarse apropiadamente, pueda agarrarlo, levantarlo y, eventualmente, llevarlo a algún otro lugar. Esta aplicación fue ejecutada con sistemas de razonamiento difuso tipo Mamdani, programados en Java sin usar librerías especializadas, en un ambiente de simulación 3D en Webots (36). En la tabla 1 se indican los parámetros de configuración de esa simulación. Allí puede notarse que en dicho escenario solo está el robot, un objeto sólido cilíndrico, y una ambientación lumínica específica, establecida con dos focos de iluminación. En la figura 1a se detalla la ubicación donde se incorporó la cámara en el robot, y en la figura 1b se muestra al robot junto con el objeto que debe agarrar. Vale la pena mencionar que el alcance de esta aplicación abarca el procesamiento de imagen para la identificación del objeto, mediante segmentación por píxeles, además del control de los motores del robot, tanto de las ruedas como de la pinza, para ubicar al robot justo en donde puede agarrar el objeto.

**Tabla 1.** Configuración de la simulación en Webots 8.4.0

| Nodo             | Parámetro        | Valor          |
|------------------|------------------|----------------|
| WorldInfo        | basicTimeStep    | 16             |
| Background       | skyColor         | [0,3 0,3 0,3]  |
| DirectionalLight | ambientIntensity | 1              |
|                  | direction        | [-0,3 -1 -0,5] |
|                  | intensity        | 0,8            |
|                  | castShadows      | TRUE           |
| DirectionalLight | ambientIntensity | 0,1            |
|                  | direction        | [-2 -1 -1]     |
|                  | intensity        | 0,5            |

|                               |                                |                          |                   |               |
|-------------------------------|--------------------------------|--------------------------|-------------------|---------------|
|                               | castShadows                    | TRUE                     |                   |               |
| Floor                         | size                           | [5 5]                    |                   |               |
| Pionner3dx                    | translation                    | [0,25 0,0975 0,25]       |                   |               |
|                               | rotation                       | [0 1 0 3,14159]          |                   |               |
| Pionner3dx<br>[extensionSlot] | Pioneer3Gripper                | translation              | [0 0,0285 -0,126] |               |
|                               | Camera                         | translation              | [0 -0,07 -0,1]    |               |
|                               |                                | fieldOfView              | 2,5               |               |
|                               |                                | width                    | 256               |               |
|                               |                                | height                   | 96                |               |
|                               |                                | noise                    | 0,02              |               |
| Solid                         | translation                    |                          | [0,5 0,06 0,75]   |               |
|                               | [children]<br>DEF Target Shape | Appearance<br>[Material] | ambientIntensity  | 0,8           |
|                               |                                | geometry<br>Cylinder     | diffuseColor      | [0,3 0,5 0,7] |
|                               |                                |                          | height            | 0,12          |
|                               |                                |                          | radius            | 0,0325        |
|                               | boundingObject                 | USE                      |                   | Target        |
|                               | Physics                        | density                  |                   | -1            |
|                               |                                | mass                     |                   | 0,055         |

Fuente: Autores.

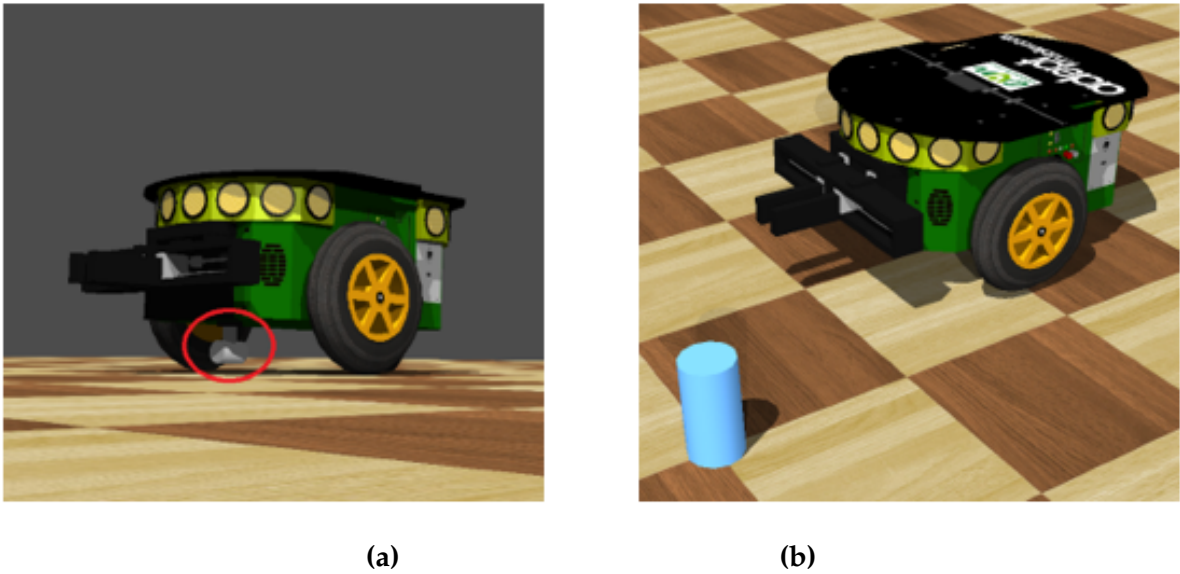


Figura 1. (a) Pioneer 3-DX con cámara abordo, y (b) objeto a agarrar, en Webots

Fuente: Autores.



A continuación se detallan los elementos propios de los sistemas difusos que se emplearon para la percepción visual y el control motor del robot en esta aplicación.

### Razonamiento difuso tipo Mamdani

Este tipo de sistema percibe, procesa y responde información empleando representaciones basadas en lógica difusa como conjuntos difusos, variables lingüísticas y reglas difusas. En principio, las *variables lingüísticas* representan los aspectos que el sistema percibe o responde, a través de conjuntos difusos para formalizar *términos lingüísticos* sobre el aspecto representado. En cada variable, cada término lingüístico tiene un conjunto difuso definido en su *universo de discurso*, que es el rango de valores cuantitativos del aspecto representado. Por su parte, las reglas *difusas* relacionan términos de percepción y acción, de manera implicativa, para representar comportamientos; por ejemplo: “Si quiero café claro pero lo veo oscuro, entonces debo agregarle más leche”. En la figura 2a se esquematiza este tipo de sistema, con  $[2 \times 2]$  reglas difusas ( $R_n$ ), en donde  $A_i$  y  $B_j$  son los términos lingüísticos de las entradas, y  $C_n$  son los *valores centro* de cada  $R_n$ , es decir el valor que se respondería al cumplirse totalmente la respectiva regla difusa. La figura 2b muestra la respectiva organización matricial de dichas reglas.

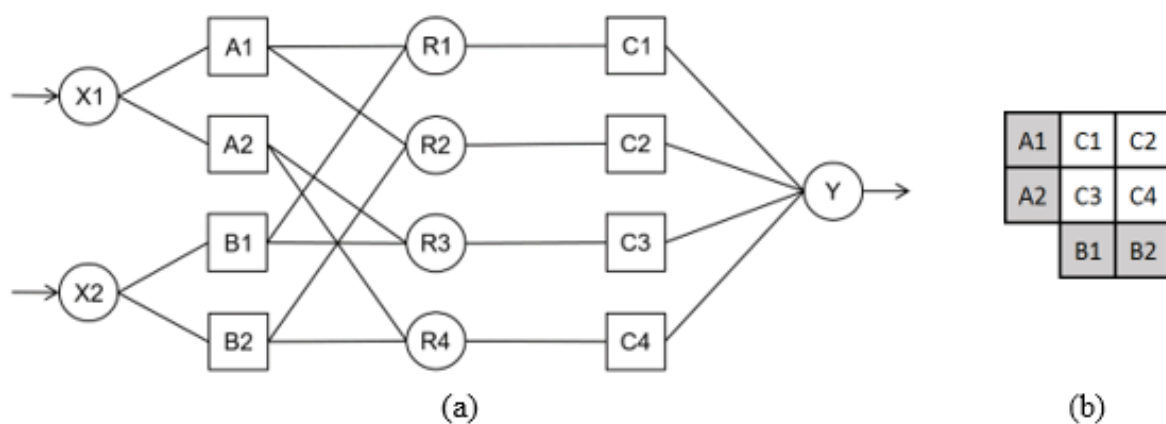


Figura 2. (a). Sistema de razonamiento difuso tipo Mamdani, y (b) matriz de centros

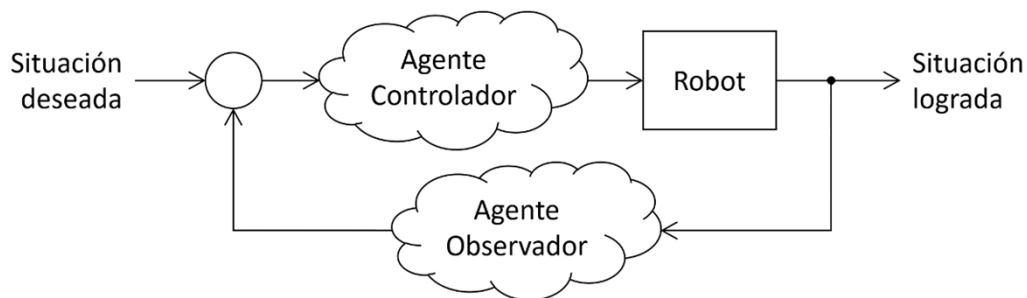
Fuente: Autores.

El razonamiento difuso consta de tres procesos fundamentales: fuzzificación, inferencia difusa y defuzzificación. Al fuzzificar, se convierten los datos percibidos en los valores difusos requeridos para el razonamiento. La inferencia emplea los valores así fuzzificados para calcular la decisión de actuación, por medio de las reglas difusas; y al defuzzificar, se obtiene un valor de respuesta acorde con la situación percibida y en pro del comportamiento deseado para el sistema. Existen diversos métodos de defuzzificación, pero acá se optó por usar el de la *media de centros*. Al usar este tipo de sistemas difusos, el diseñador debe establecer todos los valores

necesarios para que las variables lingüísticas, y las reglas difusas en cuestión, garanticen dicho comportamiento deseado. De esta manera en la figura 2a, A1 y A2 son los conjuntos difusos que fuzzifican la entrada X1, mientras que B1 y B2 son los que fuzzifican a X2. Los nodos Rn calculan la activación de la respectiva regla difusa y junto con los valores centro Cn se calcula la defuzzificación para entregar la respuesta Y.

### Composición del lazo de control

A los sistemas de razonamiento difuso acá empleados se les denominará “agentes”. Así, el robot está dotado con dos agentes, uno perceptivo y el otro de actuación, a los que se les denominó: agente observador y agente controlador, respectivamente, y en la figura 3 se ilustra cómo estos dos agentes operan en lazo cerrado. A continuación se detallan sus variables lingüísticas y las reglas difusas.



**Figura 3.** Esquema de control empleando dos agentes de razonamiento difuso

**Fuente:** Autores.

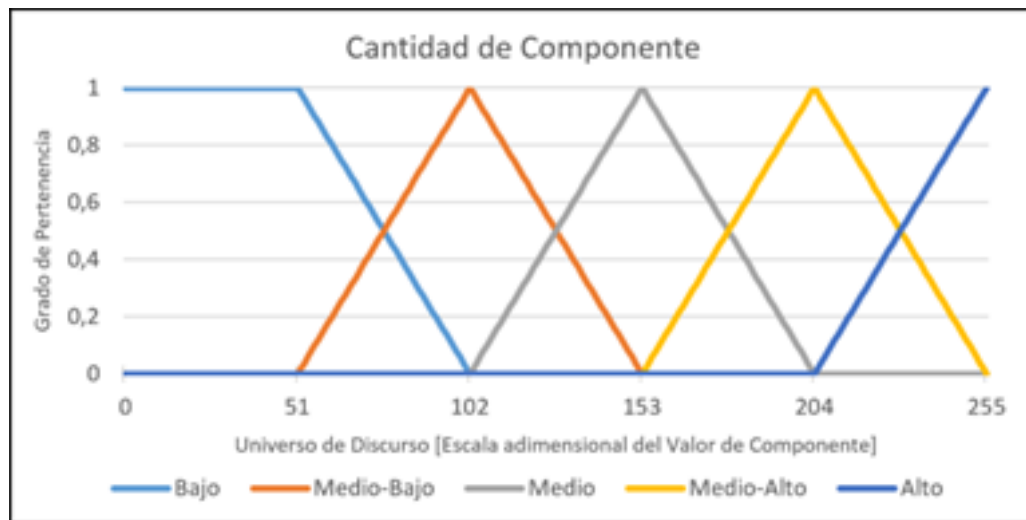
### Agente observador

Este primer agente es el encargado de procesar la información percibida del entorno mediante la cámara. Dicha información se captura en una imagen de  $[256 \times 96]$  píxeles, de la cual se obtiene la ubicación relativa del objeto, con respecto al robot, representada concretamente con dos valores: la coordenada del píxel que se encuentra en el centro del recuadro que enmarca el objeto en la imagen, y sus dimensiones en ancho, alto y área de dicho recuadro. La tarea de este agente es procesar dicha imagen, convirtiendo a gris los píxeles que no corresponden al objeto y conservando intactos los píxeles que sí corresponden; así, genera una nueva imagen, a partir de la original, en tiempo real. La decisión que discrimina dichos píxeles se escoge según el valor defuzzificado, para cada píxel de la imagen original. Así, si ese valor resulta negativo, el píxel es griseado en la imagen resultante, pero si resulta positivo, al píxel se le deja el color original en la imagen resultante.



### *Variables lingüísticas de entrada en el agente observador*

Considerando que cada pixel tiene tres componentes: {R, G, B }, se definieron tres variables lingüísticas de entrada para representarlos, usando un universo de discurso comprendido entre [0; 255], y 5 términos lingüísticos: Bajo (B), Medio-Bajo (MB), Medio (M), Medio-Alto (MA) y Alto (A), por la cantidad del respectivo componente. En la figura 4 se observan sus conjuntos difusos, centrados respectivamente en: {51, 102, 153, 204 y 255}.

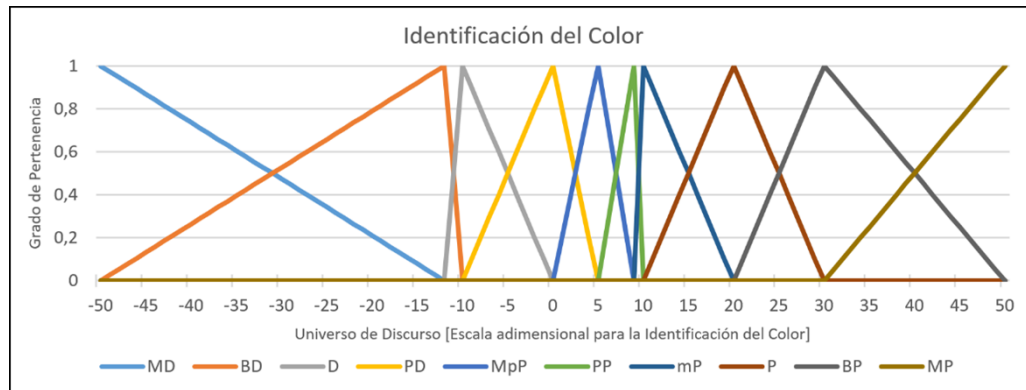


**Figura 4.** Conjuntos difusos en cada una de las tres variables de entrada

**Fuente:** Autores.

### *Variable lingüística de salida en el agente observador*

Para que la respuesta del razonamiento difuso permita discriminar los pixeles que corresponden, o no, al objeto que debe identificarse en la imagen, en la variable lingüística de salida se definieron diez términos lingüísticos que hacen referencia a cuánto se parece la combinación RGB de entrada, al color de dicho objeto, teniendo en cuenta que por la iluminación hay diversos tonos de dicho color. Si el valor de la defuzzificación es menor o igual que cero (0), indica que el color del pixel percibido es diferente y entonces se cambia a escala de gris, en la imagen resultante, mientras que para los demás valores de defuzzificación la tonalidad del pixel no será modificada. Para ello el universo de discurso se definió entre [-50 ; 50], a manera de escala adimensional para la identificación del color en cuestión, y en la figura 5 se observan los conjuntos difusos: {muy-diferente (MD), bastante-diferente (BD), diferente (D), poco-diferente (PD), muy-poco-parecido (MpP), poco-parecido (PP), medio-parecido (mP), parecido (P), bastante-parecido (BP), y muy-parecido (MP)}, centrados respectivamente en: {-50, -12, -10, 0, 5, 9, 10, 20, 30 y 50 }.



**Figura 5.** Partición de conjuntos difusos para la variable de salida del agente observador

**Fuente:** Autores.

### *Matriz de centros para la defuzzificación*

Al tener tres variables lingüísticas de entrada con cinco términos lingüísticos cada una, la matriz de centros contiene  $[5 \times 5 \times 5]$  reglas difusas, tal como se muestra en la figura 6. Allí, los términos de las variables de entrada tienen fondo rojo, verde y azul, según su respectivo componente {R, G, B}, y con los centros así establecidos, en las figuras 7b, 8c, 9c, 15c, y 16c, se evidencia que el agente observador logra identificar correctamente los pixeles que corresponden al objeto, griseando los que no corresponden.

|    | Bajo |    |    |    |    | Medio-Bajo |     |     |    |    | Medio |     |     |     |    | Medio-Alto |    |     |     |     | Alto |    |   |     |     |    |    |
|----|------|----|----|----|----|------------|-----|-----|----|----|-------|-----|-----|-----|----|------------|----|-----|-----|-----|------|----|---|-----|-----|----|----|
| A  | 0    | 0  | 0  | 0  | 0  | 0          | 0   | 0   | 0  | 0  | 0     | 0   | 0   | 0   | 0  | 0          | 0  | 0   | 0   | 0   | 0    | 0  | 0 | 0   | A   |    |    |
| MA | 0    | 0  | 0  | 0  | 10 | 0          | 0   | 0   | 0  | 50 | 0     | 0   | 0   | -12 | 50 | 0          | 0  | 0   | -12 | -10 | 0    | 0  | 0 | -10 | -10 | MA |    |
| MB | 0    | 0  | 0  | 10 | 10 | 0          | 0   | 0   | 20 | 50 | 0     | 0   | -10 | -50 | 30 | 0          | 0  | -10 | -12 | -12 | 0    | 0  | 0 | -10 | -10 | MB |    |
| MB | 0    | 10 | 10 | 10 | 9  | 0          | -10 | -10 | 20 | 9  | 0     | -10 | -10 | 0   | 0  | 0          | 0  | 0   | -10 | 0   | 0    | 0  | 0 | 0   | -10 | 0  | MB |
| B  | -10  | 5  | 0  | 0  | 0  | 0          | -10 | 5   | 0  | 0  | 0     | 0   | -10 | 0   | 0  | 0          | 0  | 0   | 0   | 0   | 0    | 0  | 0 | 0   | 0   | B  |    |
|    | B    | MB | M  | MA | A  | B          | MB  | M   | MA | A  | B     | MB  | M   | MA  | A  | B          | MB | M   | MA  | A   | B    | MB | M | MA  | A   |    |    |

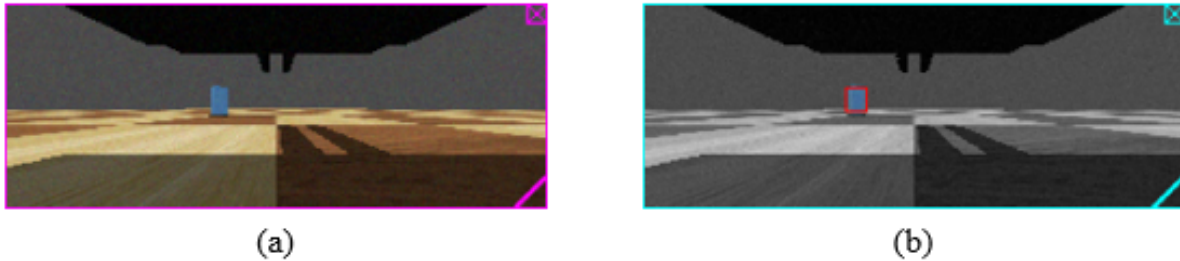
**Figura 6.** Centros en las  $[5 \times 5 \times 5]$  reglas difusas del agente observador

**Fuente:** Autores.

### *Identificación del objeto*

Tras la acción del agente observador se estima el tamaño y la ubicación del objeto en la imagen así: por una parte, para estimar el área del objeto se cuenta la cantidad de veces en que se mantuvo la tonalidad del pixel original. Por otra parte, para estimar el ancho del objeto, se cuenta fila por fila el número de veces que se mantuvo la tonalidad del píxel original, y se escoge el mayor de estos conteos. Con el valor del área y el ancho, la respectiva altura se

calcula dividiéndolos. En la figura 7b se detalla el resultado de este proceso, cuando el robot está ubicado tal como se ilustra allí. El recuadro rojo que resalta al objeto resulta centrado en una coordenada cuya fila se obtiene de sumar la fila del primer píxel que mantuvo su tonalidad original más la mitad de la altura del objeto, mientras que su columna se obtiene de sumar el valor de columna de dicho píxel más la mitad del ancho del objeto.



**Figura 7.** (a) Imagen percibida, y (b) objeto identificado en la imagen resultante

**Fuente:** Autores.

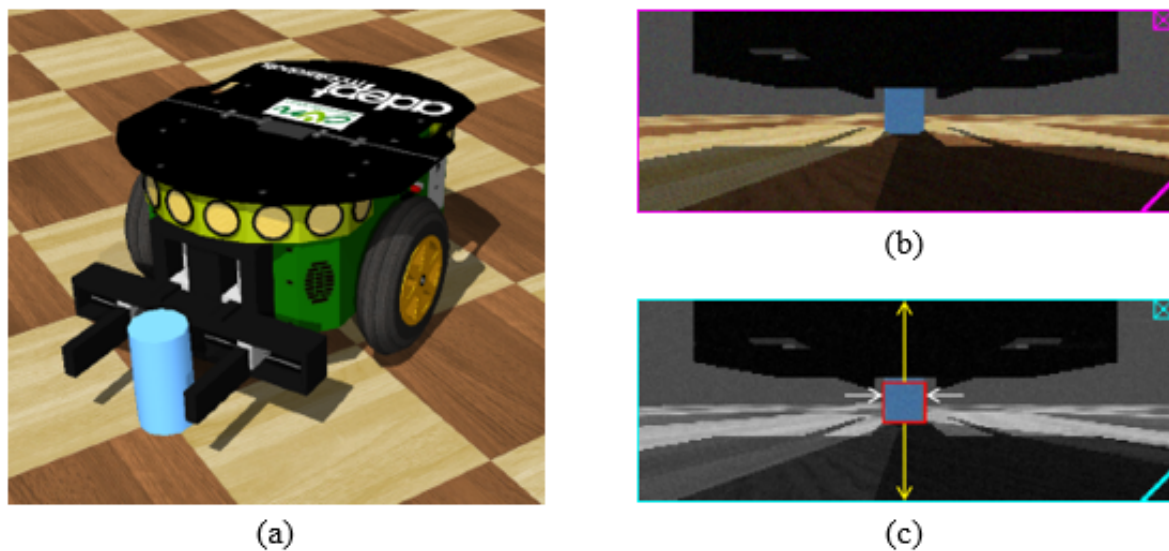
### Agente controlador

Este otro agente se encarga de mover el robot, tanto con los motores de sus ruedas como con los de la pinza, y está compuesto por dos subsistemas de control difuso, bajo el paradigma de control PD (proporcional derivativo). A uno se le denominó *razonamiento basado en el ancho observado* (RAO), y al otro *razonamiento basado en la posición horizontal* (RPH). Por su parte, el RAO permite que el robot retroceda o avance hacia el objeto, además de extender o retraer la pinza, según se requiera, en función del ancho del objeto percibido. Por otra parte, el RPH permite que el robot gire hacia la derecha o a la izquierda, para alinearse con el objeto según su posición horizontal en la imagen. Con un mecanismo de posprocesado, se combina la respuesta de estos controladores para definir las velocidades de las ruedas del robot, y la posición de los “dedos” de la pinza.

En la figura 8a puede observarse que, cuando el objeto está ubicado entre los “dedos” de la pinza, el robot puede agarrarlo, y esta es la situación deseada que se menciona en la figura 3. La figura 8b es la imagen percibida, y la figura 8c es la que entrega el agente observador, en dicha situación. En la figura 8c, las flechas blancas indican el ancho ideal que debe tener el objeto en la imagen, es decir 20 píxeles, mientras que las flechas amarillas indican su posición horizontal ideal, en la columna central de imagen, es decir 129.

### *Variables lingüísticas de entrada para el RAO*

Aplicando el paradigma de control PD difuso, y como el RAO se basa en el ancho percibido del objeto, se definió un error de ancho (EA) y la derivada del EA (DA). Por su parte, el EA resulta de restar el ancho ideal y el ancho percibido, y así el robot concibe qué tan lejos o cerca está del objeto. Cuando el objeto está lejos su ancho en la imagen es menor que su valor ideal, y el EA es positivo. En contraste, el EA es negativo cuando el objeto está muy cerca, incluso fuera del alcance de la pinza, como se observa en la figura 9.

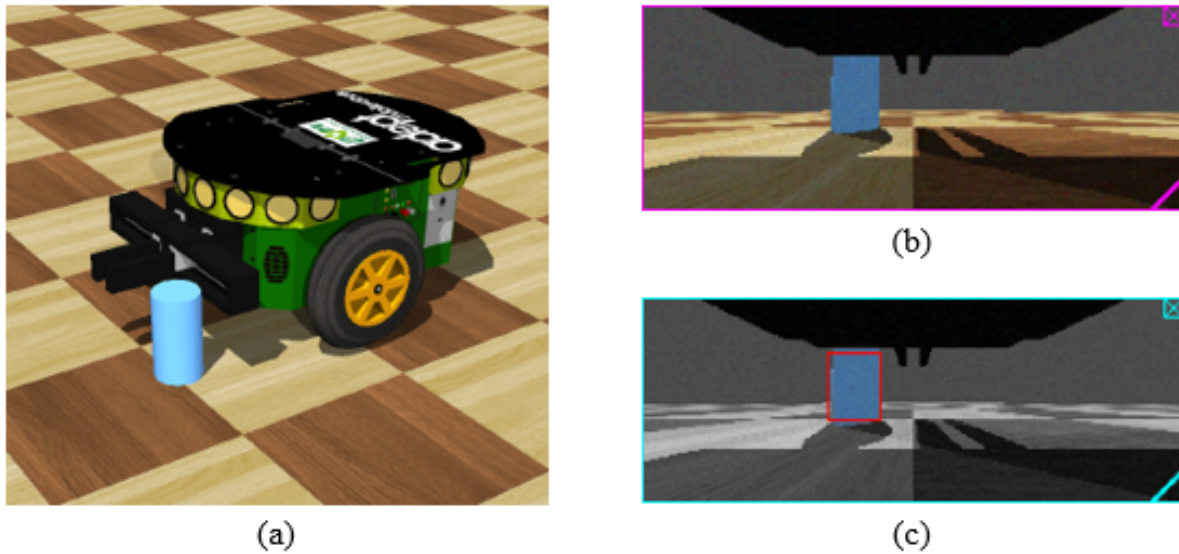


**Figura 8.** (a) Situación deseada, (b) imagen percibida, y (c) objeto identificado

**Fuente:** Autores.

El universo de discurso del EA se definió entre  $[-50 ; 20]$ . Allí, el extremo positivo ocurre cuando no hay objeto en el espacio visible, mientras que el extremo negativo sucede cuando se observa el mayor ancho posible del objeto en la imagen, que es 70. Sus términos son: {muy-negativo (MN), poco-negativo (PN), cero (Z), poco-positivo (PP), y muy-positivo (MP)}, y sus respectivos conjuntos difusos se centraron en:  $\{-9, -3, 0, 3, 9\}$ .

Por otra parte, la DA es la resta entre el EA en la iteración presente y el de la inmediatamente anterior. Así, el robot concibe la variación del ancho percibido según como se mueva, es decir, si al moverse de alguna manera se está acercando o alejando del objeto. Esta variable tiene un universo de discurso comprendido entre  $[-70 ; 70]$ , sus términos son: {MN, PN, Z, PP, y MP}, y sus respectivos valores centro están en:  $\{-3, -1, 0, 1, 3\}$ .



**Figura 9.** Posible situación en la que el objeto no está al alcance de la pinza

**Fuente:** Autores.

### ***Variable lingüística de salida en el RAO***

El RAO está enfocado, particularmente, en las situaciones donde el robot está alineado con el objeto, es decir, cuando este último se percibe en la región central de la imagen. De esta manera, el RAO debe responder un mayor valor, cuando la necesidad de avanzar para acercarse al objeto sea mayor. En contraste, dicho valor debe reducirse en la medida que el robot esté más cerca del objeto, hasta detenerse cuando ya lo pueda agarrar. Por ello, esta variable de salida fue denominada *necesidad de avanzar* (NA). Su universo de discurso es una escala adimensional entre  $[-50 ; 50]$ , sus términos son: {rápido-hacia-atrás, medio-rápido-hacia-atrás, medio-lento-hacia-atrás, lento-hacia-atrás, no-es-necesario, lento-hacia-delante, medio-lento-hacia-delante, medio-rápido-hacia-delante, y rápido-hacia-delante}, y sus respectivos conjuntos difusos están centrados en:  $\{-50, -37, -25, -12, 0, 12, 25, 37, 50\}$ .

### ***Variables lingüísticas de entrada para el RPH***

Considerando ahora el valor de columna deseable para la posición horizontal del objeto en la imagen, se definió un *error de posición horizontal* (EP) y la *derivada del EP* (DP). En este caso el EP es la diferencia entre la columna deseada y la observada, y sirve para que al robot conciba qué tan desalineado está el objeto, y hacia cuál de sus dos lados: derecho o izquierdo. Su universo de discurso está comprendido entre  $[-128 ; 128]$ , cuyo extremo negativo es cuando no hay objeto en el campo visual, mientras que el extremo positivo es cuando se alcanza a identificar ligeramente al objeto en la columna más a la izquierda de la imagen, es decir la 256va.

Sus términos lingüísticos también son: {MN, PN, Z, PP, y MP}, pero sus conjuntos difusos están centrados en: {-115, -38, 0, 38, 115}, respectivamente.

Por su parte, la DP representa la variación que tiene la posición observada del objeto, entre la iteración presente y la inmediatamente anterior, para que el robot conciba si al moverse, con cierta tendencia lateral, se está alineando con el objeto, o no. Para calcular esta variable se resta el EP en la iteración presente y el EP en la iteración pasada. Esta variable tiene un universo de discurso comprendido entre [-256 ; 256]. Sus términos son: {MN, PN, Z, PP, y MP}, y sus respectivos centros están en: {-38, -13, 0, 13, 38}.

### ***Variable lingüística de salida en el RPH***

La respuesta del RPH le permite al robot determinar cómo debe girar según la ubicación horizontal del objeto, pues si dicho valor es mayor que 0,6 el objeto está hacia la izquierda y es necesario girar hacia ese lado para alinearse con él. En contraste, si dicho valor es menor que -0,6, el objeto se encuentra hacia la derecha, y de igual manera el robot concibe que es necesario girar en ese sentido. Por ello, a esta variable se le denominó *necesidad de girar* (NG). Su universo de discurso también es una escala de valores adimensionales entre [-50 ; 50], sus términos son: {rápido-a-la-derecha, medio-rápido-a-la-derecha, medio-lento-a-la-derecha, lento-a-la-derecha, no-es-necesario, lento-a-la-izquierda, medio-lento-a-la-izquierda, medio-rápido-a-la-izquierda, y rápido-a-la-izquierda}, y sus conjuntos difusos están centrados en {-50, -37, -25, -12, 0, 12, 25, 37, 50}, respectivamente.

### ***Matrices de centros para el agente controlador***

Considerando que las dos variables de entrada en el RAO se definieron con cinco términos cada una, la matriz de centros es de  $[5 \times 5]$  reglas difusas, como se muestra en la figura 10a. Los valores centro allí definidos representan cuán necesario es avanzar para el robot, según la respectiva combinación  $[EA \times DA]$  en cada regla difusa. Los valores positivos indican que el avance es hacia delante, mientras que los negativos indican lo contrario.

Por su parte, el RPH también tiene dos variables de entrada con cinco términos, y su matriz de centros también es de  $[5 \times 5]$  reglas difusas, como se muestra en la figura 10b, y allí los centros representan cuán necesario es girar para el robot, con tendencia a la izquierda o a la derecha, según la respectiva combinación  $[EP \times DP]$  en cada regla difusa.

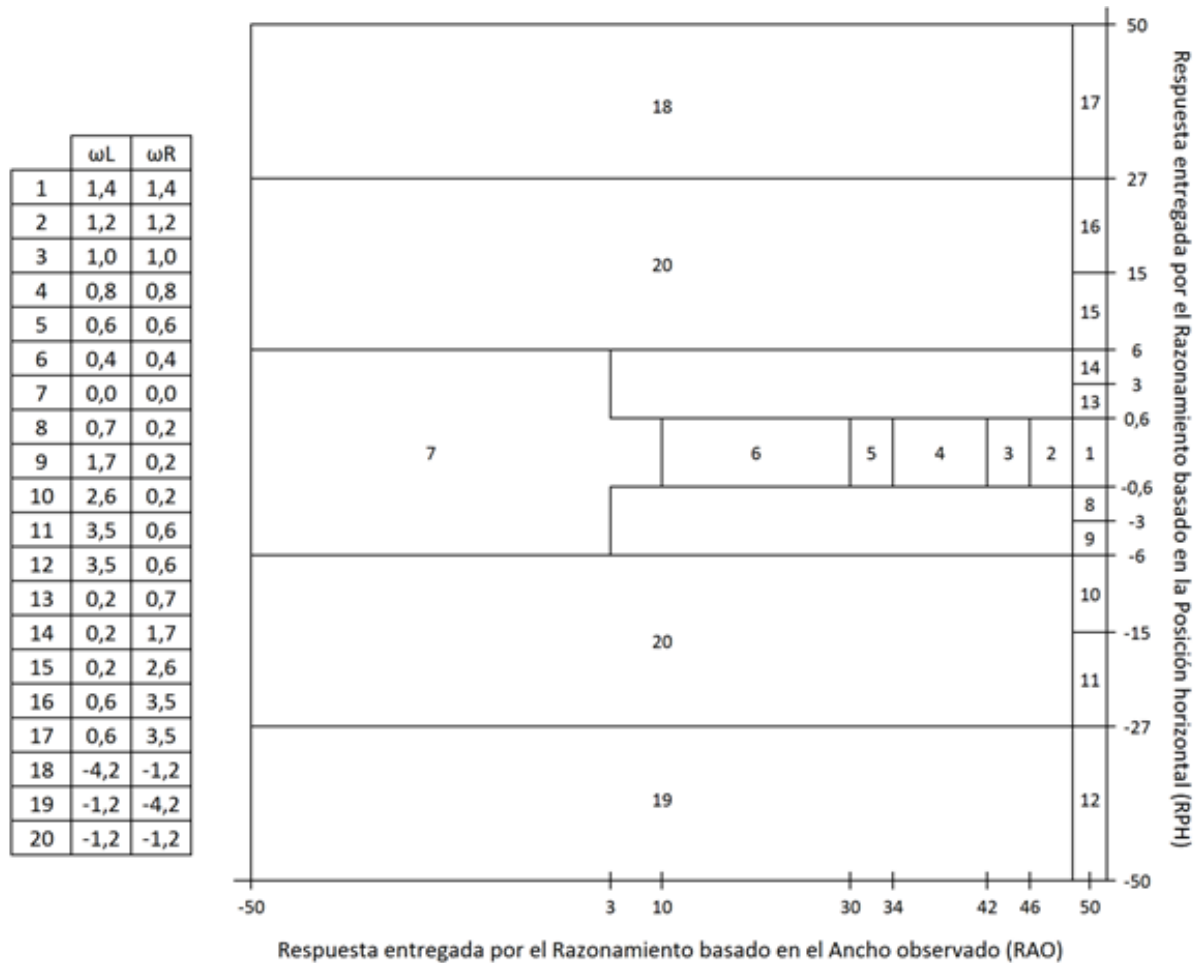
### ***Combinando las respuestas del RAO y del RPH***

Como el desplazamiento del robot debe combinar su necesidad de avanzar (NA) y la de girar (NG), en la figura 11 se muestra cómo se cruzan las respuestas del RAO y del RPH, para









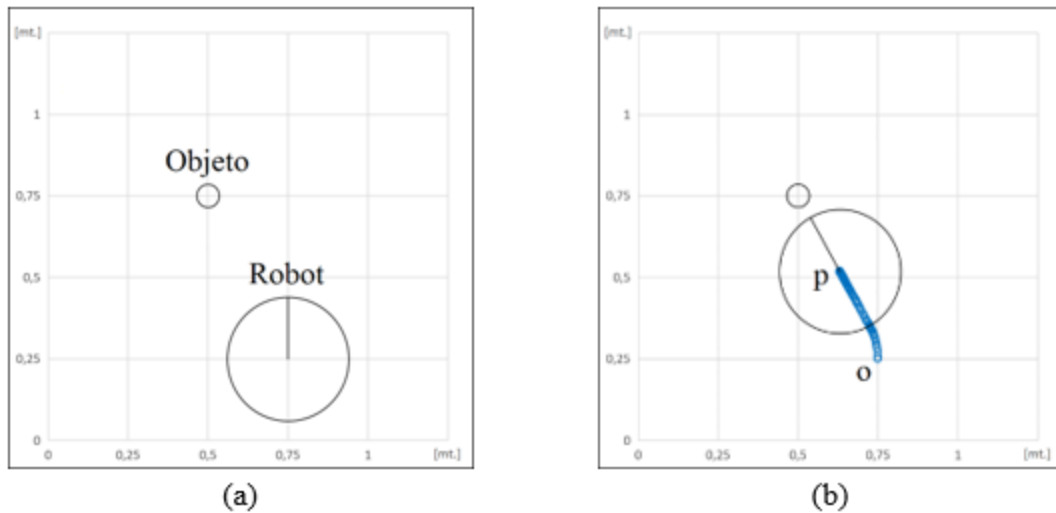
**Figura 11.** Definición de velocidades angulares para las ruedas del robot

**Fuente:** Autores.

## Resultados

El funcionamiento de los agentes descritos se validó en la versión 8.4.0 de *Webots*, con la configuración de la tabla 1. La identificación del objeto se observó en las figuras 7, 8 y 9, y a continuación se detalla el comportamiento evidenciado por el robot, particularmente, en dos situaciones experimentales.

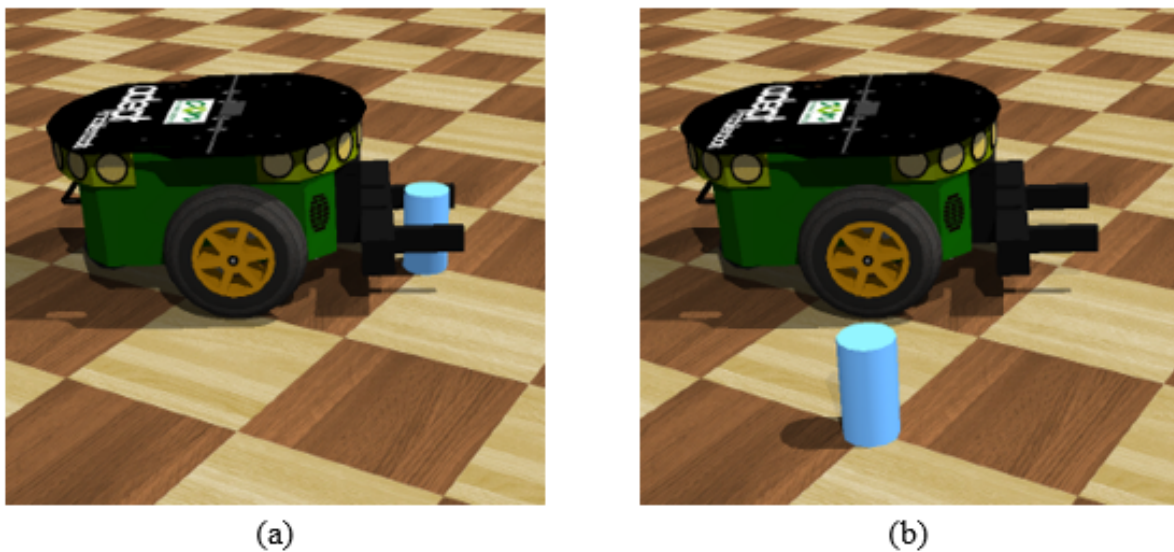
La primera de esas situaciones es la que se detalló en la figura 1b, cuya representación en plano cartesiano se muestra en la figura 12a. A partir de ese punto inicial “o”, la trayectoria ejecutada por el robot se muestra en la figura 12b, cuyo punto final “p” corresponde a la situación de la figura 13a, en donde el robot puede agarrar el objeto en cuestión.



**Figura 12.** (a). Situación inicial de la figura 1b, y (b) trayectoria de acercamiento

**Fuente:** Autores.

La segunda situación comienza cuando se quita repentinamente el objeto del espacio visual del robot y se coloca en una nueva ubicación, como se muestra en la figura 13b.

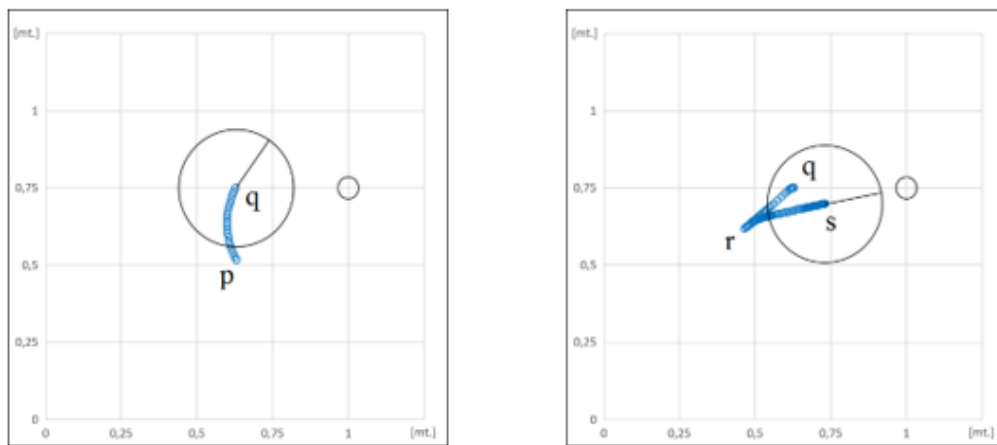


**Figura 13.** Situación correspondiente al punto “p” en las figuras: (a) 12b, y (b) 14a

**Fuente:** Autores.

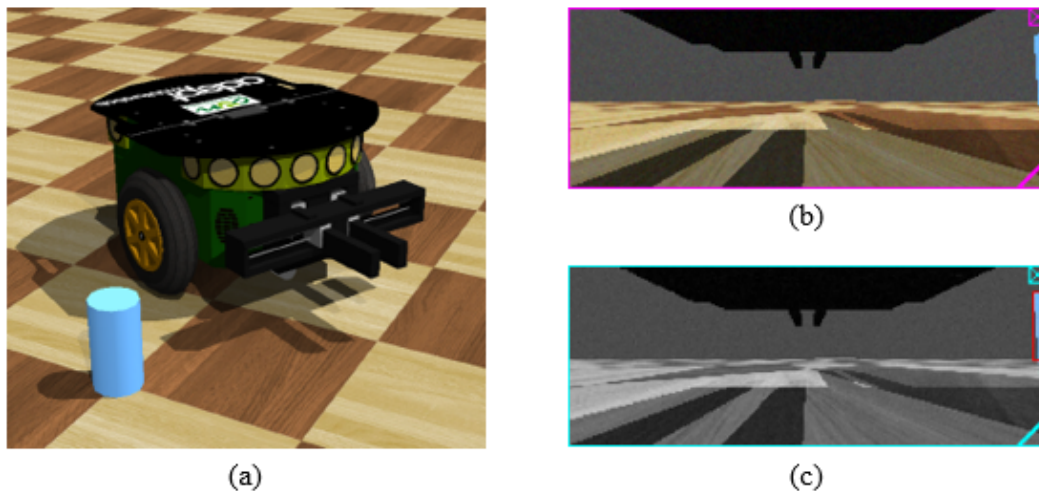
A partir de allí el robot panea hasta detectar ligeramente el objeto a la derecha, en el punto “q” de la figura 14a, que representa la situación de la figura 15a. Allí, el robot retrocede hasta el punto “r” de la figura 14b, tomando suficiente distancia del objeto, como se detalla en la

figura 16a, y a partir de ese momento el robot logra de nuevo su objetivo, en el punto “s” de la figura 14b. Vale la pena mencionar que el paneo siempre se ejecuta automáticamente, girando por la derecha, y en la figura 17 se detallan las gráficas de convergencia para las dos variables controladas, teniendo allí marcados los momentos {o, p, q, r, s}.



**Figura 14.** (a) Trayectoria de paneo, y (b) trayectoria de alejamiento-acercamiento

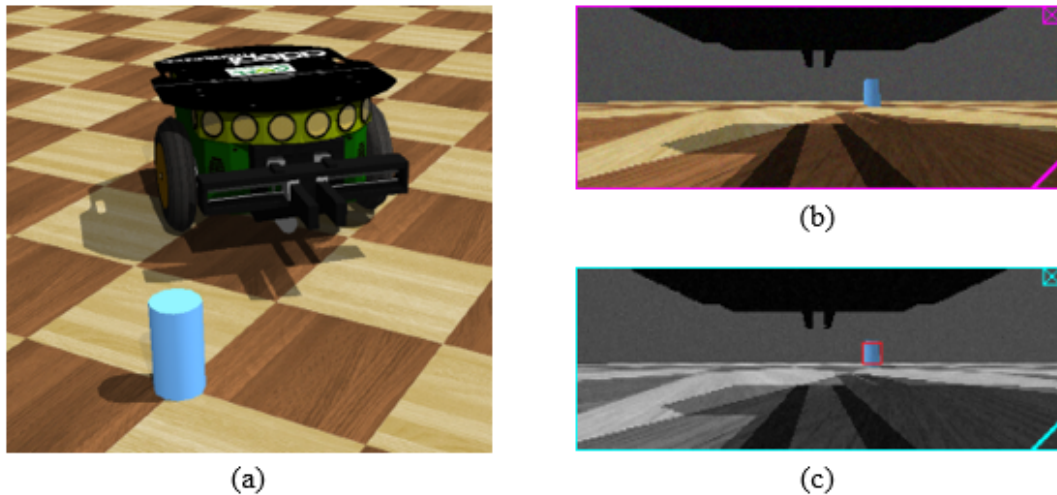
**Fuente:** Autores.



**Figura 15.** (Situación correspondiente al punto “q”, indicado en la figura 14a)

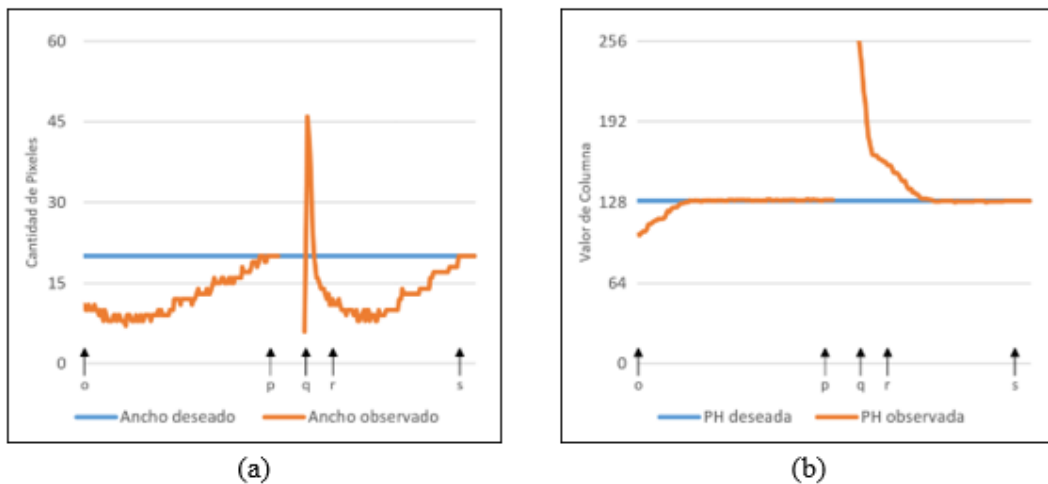
**Fuente:** Autores.

En la figura 18 se muestra el escenario con el que se realizaron pruebas adicionales, ubicando inicialmente al objeto en la coordenada (1,0 ; 1,0), y moviéndolo en una trayectoria cuadrada de 1,5 m de lado, en torno a la ubicación inicial del robot, en (0,25 ; 0,25). Dicho movimiento se



**Figura 16.** (Situación correspondiente al punto "r", indicado en la figura 14b)

**Fuente:** Autores.

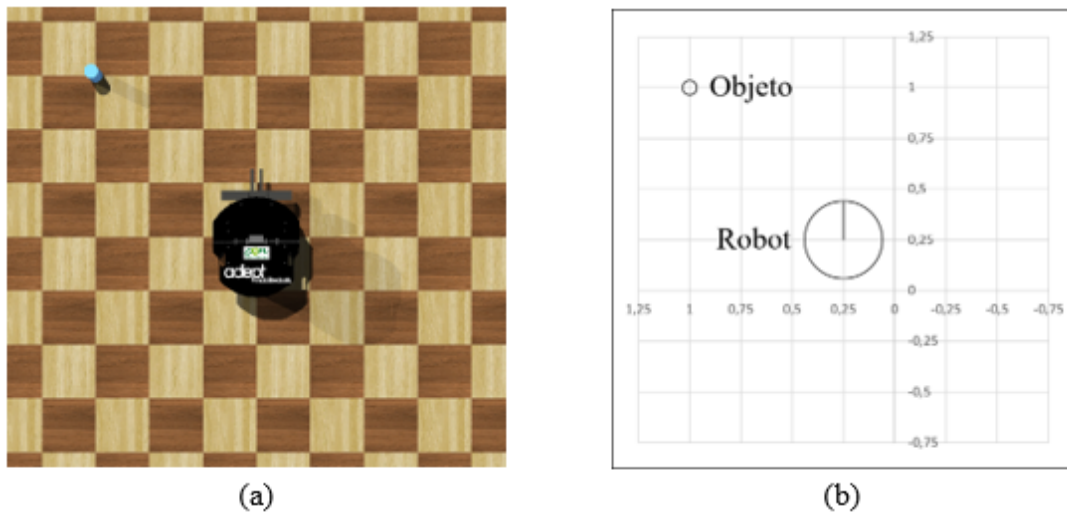


**Figura 17.** (a) Ancho, y (b) posición horizontal (PH), del objeto en la imagen.

**Fuente:** Autores.

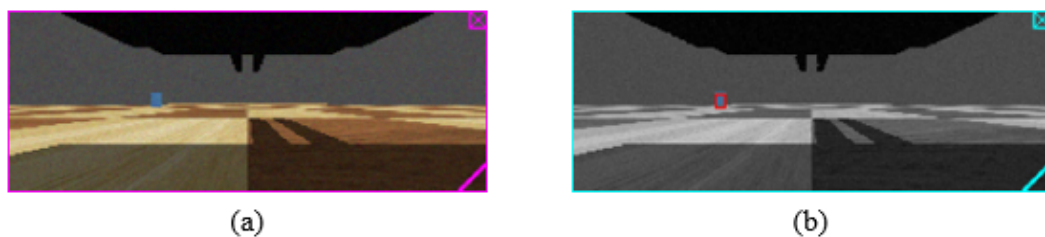
realizó tanto en sentido horario, como antihorario, y al final en ambos casos, se detuvo el objeto en cierta coordenada:  $(1,0 ; 0,76)$ , cuando se movió en sentido horario, o  $(0,76 ; 1,0)$  cuando se movió en el otro sentido. En la figura 19 puede detallarse que, al inicio de dicho movimiento, el objeto está en el campo visual del robot.

En las figuras 20 a 22, la trayectoria en color gris es la del objeto, mientras que la de color azul es el comportamiento realizado por el robot. En la figura 20a, el objeto se mueve en sentido



**Figura 18.** (a) Escenario de pruebas adicionales, y (b) coordenadas iniciales

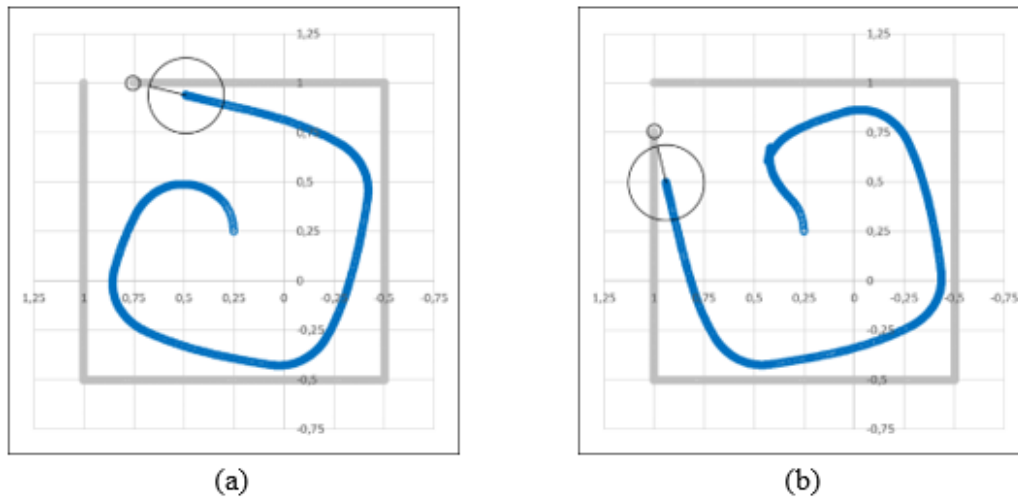
**Fuente:** Autores.



**Figura 19.** (a) Imagen percibida, y (b) objeto identificado en la situación inicial

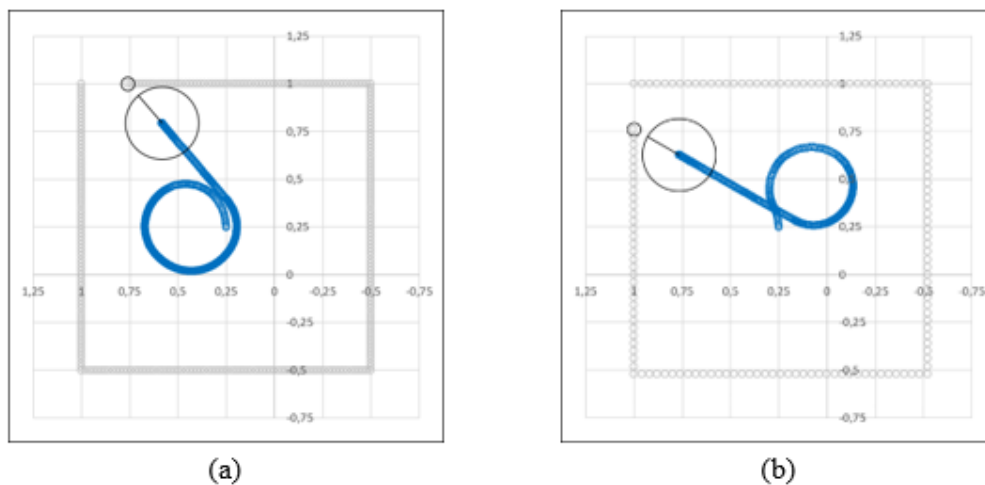
**Fuente:** Autores.

antihorario, a 10,84 cm/s, mientras que en la figura 20b, se mueve en sentido horario con esa misma velocidad. En ambas situaciones es evidente que el robot tiende a perseguir al objeto, y al final, cuando el objeto se detiene, el robot se posiciona correctamente para poder agarrarlo. En la figura 21a, el objeto se mueve en sentido antihorario, a 30,78 cm/s, mientras que en la figura 21b, se mueve en sentido horario a 61,53 cm/s. En estas situaciones, donde el objeto se mueve más rápido, el robot lo sigue distante, tan solo con la “mirada”, pero al final el robot se posiciona correctamente para poder agarrarlo. En la figura 22a, el objeto se mueve en sentido antihorario, a 1,15 m/s, y en la figura 22b, se mueve en sentido horario con esa misma velocidad. En estas situaciones el robot pierde de vista al objeto y, cuando la trayectoria se ve allí de color rojo, el robot está paneando hacia la derecha en procura de volver a percibir el objeto. Al final de estas situaciones, el robot también logra posicionarse correctamente para poder agarrar al objeto.



**Figura 20.** Pruebas con el objeto moviéndose en (a) sentido antihorario, y (b) sentido horario, ambos a 10,84 cm/s

**Fuente:** Autores.

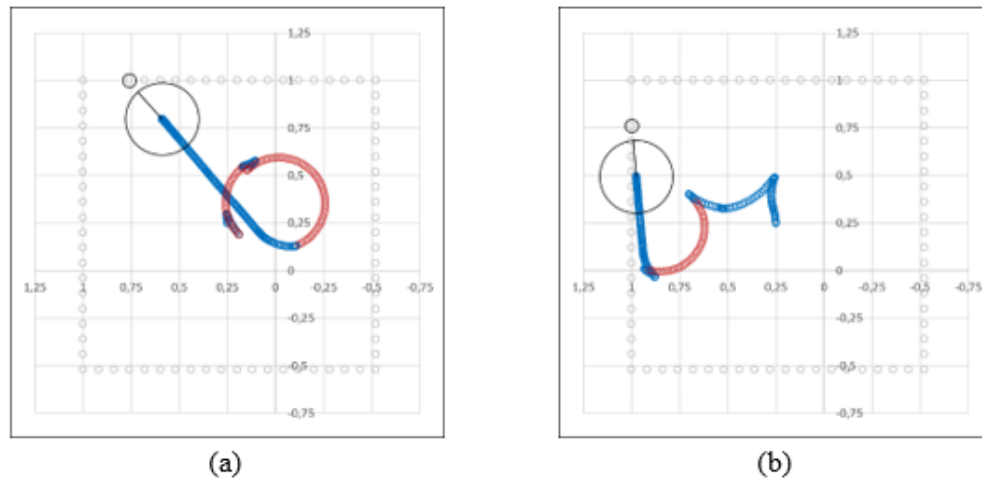


**Figura 21.** Pruebas con el objeto moviéndose en: (a) sentido antihorario (a 30,78 cm/s), y (b) sentido horario (a 61,53 cm/s)

**Fuente:** Autores.

Finalmente, se efectuaron pruebas semejantes a las anteriores, pero reduciendo los parámetros lumínicos al 80 % de los originales. Para tal fin se reconfiguró la simulación, modificando los valores que se especifican en la tabla 2. Debido a dicha variación, el objeto se percibe más oscuro que en todas las pruebas anteriores, tal como puede apreciarse en la figura 23a; sin embargo, en la figura 23b se evidencia que el agente observador sigue identificando correc-

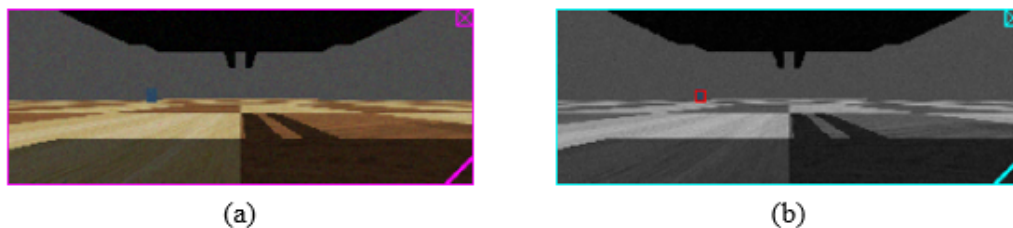




**Figura 22.** Pruebas con el objeto moviéndose en: (a) sentido antihorario, y (b) sentido horario, ambos a 1,15 m/s

**Fuente:** Autores.

tamente al objeto. En las figuras 24 a 26, se observa que el robot evidencia, prácticamente, el mismo comportamiento observado en las figuras 20 a 22, debido a que el objeto es movilizado, respectivamente, de la misma manera como en aquellas circunstancias. De esta manera, el sistema de control evidenció ser tolerante a la variación lumínica en este experimento; sin embargo, es notable es que el robot se detiene un poco más cerca del objeto, tal como se especifica en la tabla 3, aunque el robot no chocó ni derribó al objeto, en ninguna de las pruebas realizadas.



**Figura 23.** (a) Imagen percibida, y (b) objeto identificado en la situación inicial

**Fuente:** Autores.

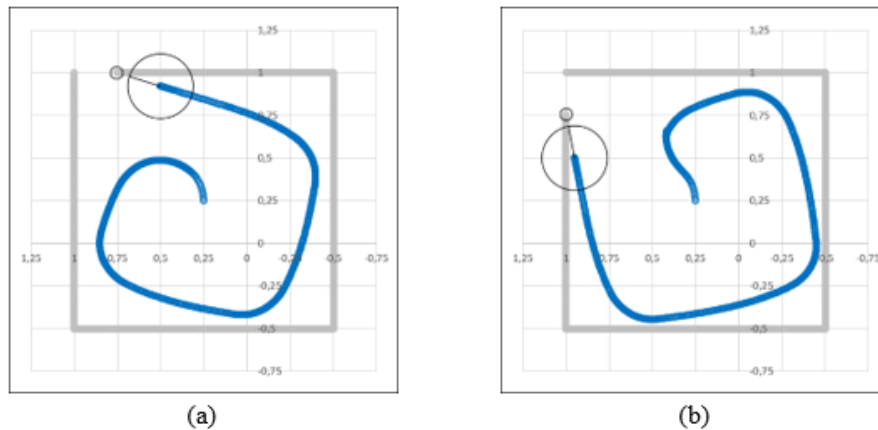
**Tabla 2.** Modificación de los parámetros lumínicos, al 80 %

| Nodo             | Parámetro        | Valor |
|------------------|------------------|-------|
| DirectionalLight | ambientIntensity | 0,8   |
|                  | Intensity        | 0,64  |
| DirectionalLight | ambientIntensity | 0,08  |



|       |                                |                          |                  |      |
|-------|--------------------------------|--------------------------|------------------|------|
|       | Intensity                      |                          |                  | 0,4  |
| Solid | [children]<br>DEF Target Shape | Appearance<br>[Material] | ambientIntensity | 0,64 |

**Fuente:** Autores.



**Figura 24.** Pruebas con el objeto moviéndose en: (a) sentido antihorario, y (b) sentido horario, ambos a 10,84 cm/s

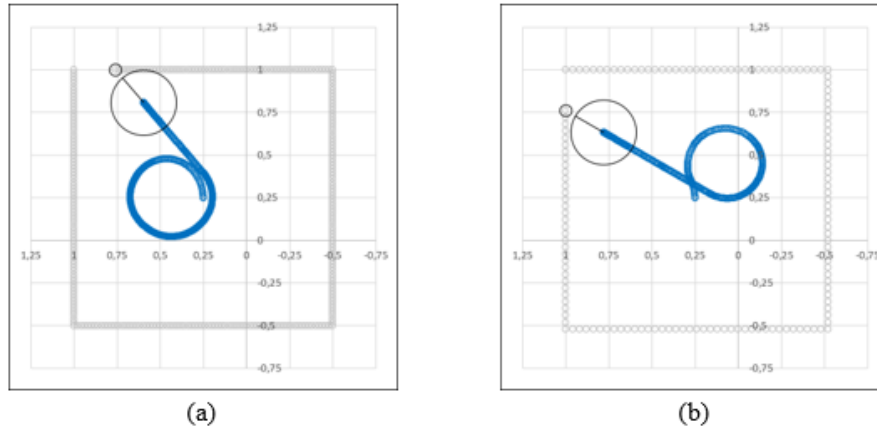
**Fuente:** Autores.

**Tabla 3.** Distancias entre el robot y el objeto, al final de cada prueba

| Sentido                        | Iluminación | 10,84 cm/s | 30,78 cm/s  | 1,15 m/s   |
|--------------------------------|-------------|------------|-------------|------------|
| Antihorario                    | 100 %       | 27,1995 cm | 27,16298 cm | 26,5528 cm |
| Antihorario                    | 80 %        | 26,5061 cm | 25,32125 cm | 25,0096 cm |
| Diferencia (más cerca en 80 %) |             | 0,6934 cm  | 1,84173 cm  | 1,5432 cm  |
| Sentido                        | Iluminación | 10,84 cm/s | 61,53 cm/s  | 1,15 m/s   |
| Horario                        | 100 %       | 26,6165 cm | 26,9508 cm  | 26,7725 cm |
| Horario                        | 80 %        | 26,0464 cm | 25,3975 cm  | 25,4235 cm |
| Diferencia (más cerca en 80 %) |             | 0,5701 cm  | 1,5533 cm   | 1,349 cm   |

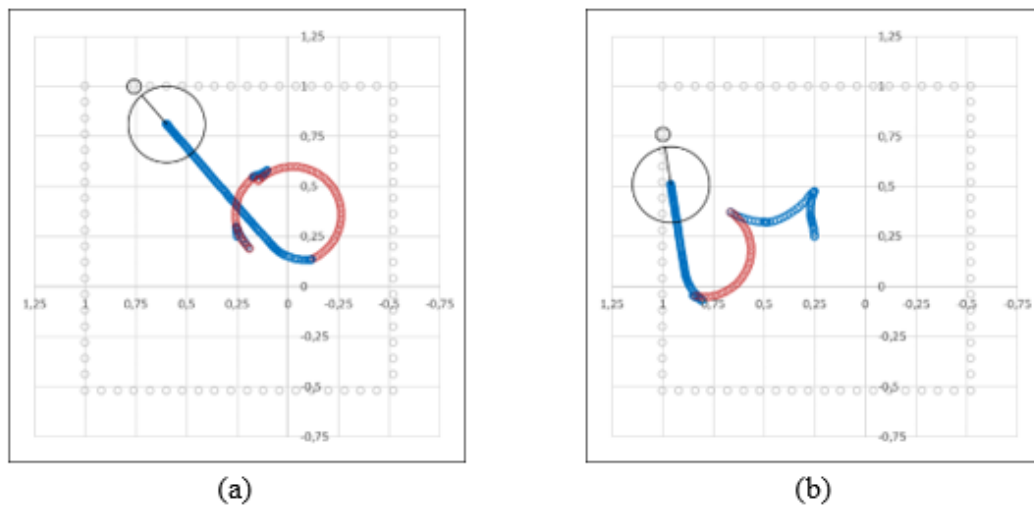
## Conclusiones

En este artículo se reporta la validación del diseño de una estructura de control para un robot móvil, basada en visión artificial con cámara abordo empleando sistemas de lógica difusa, tipo Mamdani. Con dicho diseño se puede expresar el comportamiento del robot en lenguaje



**Figura 25.** Pruebas con el objeto moviéndose en: (a) sentido antihorario (a 30,78 cm/s), y (b) sentido horario (a 61,53 cm/s)

**Fuente:** Autores.



**Figura 26.** Pruebas con el objeto moviéndose en: a). Sentido antihorario, y (b) sentido horario, ambos a 1,15 m/s

**Fuente:** Autores.

natural, mediante las reglas difusas. Por ejemplo: “si el EA es PP, y la DA es PN, entonces *necesito avanzar* medio lento hacia delante”, y, “como el EP es PN, y la DP es PP, entonces *necesito girar* medio lento a la derecha”, lo cual, expresado así en primera persona involucra una especie de “yo”, con la intención de satisfacer cierta “necesidad personal”, considerando que el deseo de agarrar el objeto pudiese provenir de algún sistema adicional, por ejemplo, si el uso de este sistema formara parte de alguna planeación comportamental, en la cual, agarrar ese objeto sirviera para realizar una tarea más sofisticada.

Se notó que el comportamiento de paneo, realizado entre “p” y “q”, en la figura 14a, tiene una gran desventaja al ejecutarse siempre por el lado derecho, de manera automática, y desinformada. Dicho comportamiento tiene la finalidad de hacer girar al robot para ver si, eventualmente, encuentra al objeto en el campo visual. Sin embargo, puede suceder que el objeto haya “desaparecido” del campo visual, en caso de que algo externo lo haya movido lateralmente, a ras del suelo. En esta eventualidad, el robot debería ser capaz de notar que el objeto “salió” del campo visual, por alguno de los dos lados laterales de la imagen, y en tal caso, el robot debería considerar que es mejor panear por el lado respectivo, con el fin de volver a ver el objeto. Esto último mejoraría el desempeño acá presentado, pues, cuando el objeto “sale” del campo visual por el lado izquierdo, el robot se detiene y panea hacia la derecha, haciéndolo incongruente en este sentido.

El diseño acá propuesto usa términos lingüísticos intuitivos para denominar las necesidades de movimiento del robot, es decir, la de avanzar (NA) y la de girar (NG), en lugar de usar sus respectivos términos formales: “velocidad lineal” y “velocidad angular”, aun sabiendo que estos últimos son los términos que normalmente se usan para calcular las velocidades angulares de las ruedas, mediante el modelo cinemático del robot móvil. En la aplicación acá reportada se quiso evitar a propósito el uso de dicho formalismo porque no es necesaria, particularmente en el contexto donde se pretende utilizar a futuro. Para comprender esto, se puede notar que los seres humanos no necesitan conocer el modelo cinemático de su propio cuerpo, para lograr acercarse a un objeto que requieran agarrar, y adicionalmente, pueden describir su comportamiento, empleando narrativas coloquiales, es decir, sin necesidad de usar tecnicismos como “velocidad lineal” y “velocidad angular”. Así, a manera de trabajo futuro, el sistema de razonamiento difuso acá reportado se usará en una sociedad artificial de robots, en donde algunos de ellos deban describir verbalmente su comportamiento, en un proceso cultural de enseñanza y aprendizaje, siguiendo el método propuesto por (35) (Quiroga, 2023), pues de esta manera, la interpretabilidad del sistema difuso acá presentado ayudaría a “verbalizar” el comportamiento de sistemas ANFIS, como los allí referenciados.

## Financiamiento

Este artículo derivó del desarrollo del proyecto de investigación titulado: “Pythonisa (fase I) - Módulo de inferencia visual para la propagación cultural basada en replicación neurométrica artificial”, el cual fue avalado y financiado por la Corporación Unificada Nacional de Educación Superior (CUN), Bogotá (Colombia). Los autores agradecen particularmente al ingeniero Eddy Santiago Casallas Triana, LITE de Ingeniería Electrónica en la CUN, por su apoyo en la consolidación de este artículo.

## Referencias

- [1] M. S. Güzel, "Autonomous vehicle navigation using vision and mapless strategies: a survey", *Adv. Mech. Eng.*, vol. 5, pp. 234747, 2013.
- [2] A. Krizhevsky, I. Sutskever, y G. E. Hinton, "Imagenet classification with deep convolutional neural networks", *Adv. Neural Inf. Process. Syst.*, vol. 25, 2012.
- [3] R. Girshick, J. Donahue, T. Darrell *et al.*, "Rich feature hierarchies for accurate object detection and semantic segmentation", en *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580-587.
- [4] J. Redmon, S. Divvala, R. Girshick *et al.*, "You only look once: unified, real-time object detection", en *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 779-788.
- [5] W. B. Shen, D. Xu, Y. Zhu *et al.*, "Situational fusion of visual representation for visual navigation", en *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 2881-2890.
- [6] H. Rashed, M. Ramzy, V. Vaquero *et al.*, "Fusemodnet: real-time camera and lidar based moving object detection for robust low-light autonomous driving", en *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019.
- [7] T. Zhou, M. Brown, N. Snavely *et al.*, "Unsupervised learning of depth and ego-motion from video", en *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1851-1858.
- [8] E. Jung, N. Yang, y D. Cremers, "Multi-frame GAN: image enhancement for stereo visual odometry in low light", en *Conference on Robot Learning*, PMLR, 2020, pp. 651-660.
- [9] J. Cheng, P. Liu, Q. Zhang *et al.*, "Real-time and efficient 6-D pose estimation from a single RGB image", *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1-14, 2021.
- [10] W. Chen, J. Duan, H. Basevi *et al.*, "PointPoseNet: point pose network for robust 6D object pose estimation", en *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2020, pp. 2824-2833.
- [11] Y. Becerra y D. G. Sterpin Buitrago, "Actividades de construcción sencillas desarrolladas por sistemas aéreos no tripulados", *Tecnura*, vol. 27, n.º 76, pp. 12-30, jul. 2023, DOI: <https://doi.org/10.14483/22487638.18612>

- [12] Y. A. Becerra-Mora y S. Soto-Gaona, "Seguimiento y agarre de objetos en movimiento empleando manipuladores aéreos robóticos: un breve resumen", *Rev. UIS Ing.*, vol. 22, n.º 4, pp. 115-128, nov. 2023, DOI: <https://doi.org/10.18273/revuin.v22n4-2023011>
- [13] V. Ortenzi, N. Marturi, R. Stolkin *et al.*, "Vision-guided state estimation and control of robotic manipulators which lack proprioceptive sensors", en *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2016, pp. 3567-3574.
- [14] P. Vicente, L. Jamone, y A. Bernardino, "Towards markerless visual servoing of grasping tasks for humanoid robots", en *2017 IEEE International Conference on Robotics and Automation (ICRA)*, jun. 2017, pp. 3811-3816, DOI: <https://doi.org/10.1109/ICRA.2017.7989441>
- [15] Y. Becerra, J. León, S. Orjuela *et al.*, "Smart manipulation approach for assistant robot", en *AETA 2018 - Recent Advances in Electrical Engineering and Related Sciences: Theory and Application*, I. Zelinka, P. Brandstetter, T. Trong Dao *et al.*, eds., Cham: Springer International Publishing, 2020, pp. 904-913.
- [16] W. -J. Baek, C. Pohl, P. Pelcz *et al.*, "Improving humanoid grasp success rate based on uncertainty-aware metrics and sensitivity optimization", en *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*, nov. 2022, pp. 786-793, DOI: <https://doi.org/10.1109/Humanoids53995.2022.10000206>
- [17] I. Emmanuel, "Fuzzy logic-based control for autonomous vehicle: a survey", *Int. J. Educ. Manag. Eng.*, vol. 7, n.º 2, p. 41, 2017.
- [18] E. Martínez-Martin y A. P. del Pobil, "Object detection and recognition for assistive robots: experimentation and implementation", *IEEE Robot. Autom. Mag.*, vol. 24, n.º 3, pp. 123-138, 2017.
- [19] T.-L. Nguyen, R. Boukezzoula, D. Coquin *et al.*, "Combination of Sugeno fuzzy system and evidence theory for NAO robot in colors recognition", en *2015 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, IEEE, 2015, pp. 1-8.
- [20] A. Celestine y J. D. Peter, "Investigations on adaptive connectivity and shape prior based fuzzy graph-cut colour image segmentation", *Expert Syst.*, vol. 37, n.º 5, p. e12554, oct. 2020. DOI: <https://doi.org/10.1111/exsy.12554>
- [21] Y. I. Daradkeh, I. Tvoroshenko, V. Gorokhovatskyi *et al.*, "Development of effective methods for structural image recognition using the principles of data granulation and apparatus of fuzzy logic", *IEEE Access*, vol. 9, pp. 13417-13428, 2021.
- [22] P. Melo-Pinto, P. Couto, H. Bustince *et al.*, "Image segmentation using Atanassov's intuitionistic fuzzy sets", *Expert Syst. Appl.*, vol. 40, n.º 1, pp. 15-26, 2013.

- [23] M. N. Khan, M. Al Hasan, y S. Anwar, "Improving the robustness of object detection through a multi-camera-based fusion algorithm using fuzzy logic", *Front. Artif. Intell.*, vol. 4, p. 638951, 2021.
- [24] N. Delavarpour, S. Eshkabilov, T. Bon *et al.*, "The tractor-cart system controller with fuzzy logic rules", *Appl. Sci.*, vol. 10, n.º 15, p. 5223, 2020.
- [25] F. del P. González, J. G. Guarnizo, y G. Benavides, "Emulation system for a distribution center using mobile robot, controlled by artificial vision and fuzzy logic", *IEEE Lat. Am. Trans.*, vol. 12, n.º 4, pp. 557-563, 2014.
- [26] J. de J. Zavala Zavala, H. D. Vázquez Delgado, y A. de J. Ley Clemente, "Robot diferencial con lógica difusa para evadir obstáculos frontales", *Rev. Alta Tecnol. Soc.*, vol. 11, n.º 1, 2019.
- [27] B. Wu, T. Cheng, T. L. Yip *et al.*, "Fuzzy logic based dynamic decision-making system for intelligent navigation strategy within inland traffic separation schemes", *Ocean Eng.*, vol. 197, p. 106909, febr. 2020, DOI: <https://doi.org/10.1016/j.oceaneng.2019.106909>
- [28] F. Kamil y M. Y. Moghrabiah, "Multilayer decision-based fuzzy logic model to navigate mobile robot in unknown dynamic environments", *Fuzzy Inf. Eng.*, vol. 14, n.º 1, pp. 51-73, 2022.
- [29] M. Munadi, B. Radityo, M. Ariyanto *et al.*, "Automated guided vehicle (AGV) lane-keeping assist based on computer vision, and fuzzy logic control under varying light intensity", *Results Eng.*, vol. 21, p. 101678, 2024.
- [30] M. T. Ahmad, M. Greenspan, M. Asif *et al.*, "Robust apple segmentation using fuzzy logic", en *2018 5th International Multi-Topic ICT Conference (IMTIC)*, abr. 2018, pp. 1-5, DOI: <https://doi.org/10.1109/IMTIC.2018.8467275>
- [31] J. M. Soto-Hidalgo, J. M. Alonso, G. Acampora *et al.*, "JFML: a Java library to design fuzzy logic systems according to the IEEE Std 1855-2016", *IEEE Access*, vol. 6, pp. 54952-54964, 2018, DOI: <https://doi.org/10.1109/ACCESS.2018.2872777>
- [32] N. A. Nasruddin y F. S. Ismail, "Color segmentation and recognition using fuzzy logic human based perception approach for color vision deficiency", 2016.
- [33] Z. Li, F. Liu, W. Yang *et al.*, "A survey of convolutional neural networks: analysis, applications, and prospects", *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, n.º 12, pp. 6999-7019, 2021.
- [34] C. Urrea, "Artificial intelligence-driven and bio-inspired control strategies for industrial robotics: a systematic review of trends, challenges, and sustainable innovations

toward industry 5.0", *Machines*, vol. 13, n.º 8, 2025, DOI: <https://doi.org/10.3390/machines13080666>

- [35] B. L. Quiroga Gavilán, *Módulo de procesamiento de lenguaje natural para el proceso de enseñanza y aprendizaje en una sociedad de robots*, trabajo final de especialización, Facultad de Ingeniería, Universidad de Buenos Aires, 2023. [En línea]. Disponible en <https://lse-posgrados-files.fi.uba.ar/tesis/LSE-FIUBA-Trabajo-Final-CEIA-Leandro-Quiroga-2023.pdf>
- [36] *Webots documentation: webots user guide*, Cyberbotics, Gollion, Suiza, 2024. Disponible en <https://cyberbotics.com/doc/guide/index>

