



ReCIBE. Revista electrónica de Computación, Informática,
Biomédica y Electrónica

ISSN: 2007-5448

recibe@cupei.udg.mx

Universidad de Guadalajara
México

Cruz Martínez, Claudia; Algreto Badillo, Ignacio; Arellano Pimentel, J. Jesús;
Cortés Pérez, Ernesto; Aguilar Acevedo, Francisco; Morales Rosales, Luis Alberto
Sistema de Navegación Reactiva Difusa para Giros Suaves de Plataformas Móviles Empleando el Kinect

ReCIBE. Revista electrónica de Computación, Informática,
Biomédica y Electrónica, vol. 5, núm. 3, 2016, Junio-Noviembre
Universidad de Guadalajara
México

Disponible en: <https://www.redalyc.org/articulo.oa?id=512253114004>

- ▶ Cómo citar el artículo
- ▶ Número completo
- ▶ Más información del artículo
- ▶ Página de la revista en redalyc.org



Sistema de Información Científica Redalyc
Red de Revistas Científicas de América Latina y el Caribe, España y Portugal
Proyecto académico sin fines de lucro, desarrollado bajo la iniciativa de acceso
abierto

*Recibido 4 Abr 2016
Aceptado 2 May 2016*

ReCIBE, Año 5 No. 3, Noviembre 2016

Sistema de Navegación Reactiva Difusa para Giros Suaves de Plataformas Móviles Empleando el Kinect

**Fuzzy Mobile Reactive Navigation System for Smooth Turns
by Using Kinect**

Claudia Cruz Martínez¹
ccmartinez@inaoep.mx

Ignacio Algreto Badillo²
ignacio.algreto@uptlax.edu.mx

J. Jesús Arellano Pimentel³
jjap@sandunga.unistmo.edu.mx

Ernesto Cortés Pérez³
neto_144@sandunga.unistmo.edu.mx

Francisco Aguilar Acevedo³
aguilar.afco@sandunga.unistmo.edu.mx

Luis Alberto Morales Rosales⁴
lamorales@itsm.edu.mx

¹ Instituto Nacional de Astrofísica, Óptica y
Electrónica, México

2 Universidad Politécnica de Tlaxcala, México

3 Universidad del Istmo, México

4 Instituto Tecnológico Superior de Misantla, México

Resumen: La navegación en un robot móvil es la habilidad para desplazarse de un lugar a otro dentro de un entorno evitando los obstáculos que se presenten. La autonomía de un robot móvil se basa en su sistema de navegación. La aplicación de técnicas de Inteligencia Artificial como la lógica difusa y el uso de la visión por computadora son dos herramientas empleadas para cumplir esta tarea. En algunos sistemas de navegación la seguridad y la facilidad de operación son factores muy importantes. En estos casos, la tolerancia a la incertidumbre de información, la reacción ante objetos imprevistos, y la navegación mediante giros suaves son argumentos del diseño de estos sistemas. En este artículo se presenta el desarrollo de un sistema de navegación reactiva difusa que emplea los datos de profundidad del sensor Kinect, algoritmos de visión por computadora, y lógica difusa, para generar ángulos de giro suave para la navegación de un robot móvil. En pruebas realizadas con la plataforma móvil ERA-MOBI se observaron giros suaves con un porcentaje de evasión de obstáculos del 85.7%.

Palabras clave: navegación reactiva, lógica difusa, Kinect, profundidad promedio, robot móvil.

Abstract: Navigation on a mobile robot is the ability to move from one place to another within an environment avoiding obstacles that arise. The autonomy of a mobile robot is based on your navigation system. The application of Artificial Intelligence techniques such as fuzzy logic and using computer vision are two tools used to accomplish this task. In some navigation systems the safety and ease of operation are important factors. In these cases, the tolerance for uncertainty information, the reaction to unforeseen objects, and navigation through smooth turn are design arguments of these systems. This article describes the development of a system of fuzzy logic-based reactive navigation using data depth of Kinect sensor, computer vision algorithms, and fuzzy logic, to generate smooth turn angles for navigation of a mobile robot. In tests with the platform ERA- MOBI smooth turns with a percentage of obstacle avoidance of 85.7 % were observed.

Keywords: reactive navigation, fuzzy logic, average depth, mobile robot.

1. Introducción

A diferencia de la mayoría de robots industriales con un limitado espacio de trabajo, los robots móviles pueden moverse libremente para alcanzar sus objetivos deseados (Tzafestas, 2014). Esta capacidad de movilidad hace a estos robots adecuados para un gran repertorio de aplicaciones en entornos estructurados y no estructurados. Los robots móviles que emplean ruedas para su locomoción asumen diferentes configuraciones, las cuales les confieren diferentes características y propiedades respecto a su eficiencia energética, dimensiones, cargas útiles y maniobrabilidad. Dentro estas configuraciones la diferencial es la más frecuente en robots para interiores (Ollero, 2001).

Por otra parte, un robot móvil se enfrenta a entornos locales diferentes cada vez que se mueve, teniendo la capacidad de influir y ser influido por estos (Kelly, 2013). Más importante, el mundo es un lugar peligroso, y que a menudo no puede ser diseñado para adaptarse a las limitaciones del robot, por lo que la movilidad aumenta el nivel de inteligencia necesaria. El éxito al hacer frente a las diferentes demandas y riesgos de cada lugar y cada situación es un reto importante incluso para los sistemas biológicos.

En este aspecto, la navegación es una metodología que permite guiar a un robot móvil de manera segura y estable, a través de un entorno con obstáculos con el fin de ejecutar alguna tarea en específico (Murphy, 2000). Durante las últimas décadas se han realizado importante esfuerzos en la aplicación de técnicas de Inteligencia Artificial a la navegación (Ollero, 2001). La lógica difusa es uno de los métodos empleados que proporciona una tolerancia inherente a la incertidumbre en los datos (Yerubandi, Reddy, & Kumar, 2015). Además, cuando se selecciona un sistema de inferencia difuso de forma adecuada es posible modelar eficazmente la experiencia humana en una aplicación específica (Mohan, 2005).

Desde el punto de vista de la planificación, existen diferentes arquitecturas diseñadas teniendo en cuenta especificaciones como el tiempo de respuesta del sistema y la disponibilidad de información. La solución se sitúa normalmente entre dos extremos: la planificación puramente estratégica y la puramente reactiva. En la planificación reactiva se supone que el entorno es incierto, buscándose la mayor flexibilidad posible para reaccionar en cualquier instante lo suficientemente rápido a las discrepancias entre el modelo actual y la realidad observada en el entorno (Ollero, 2001). Un ejemplo de planificación reactiva se presenta en Suárez y Loaiza (2015), con la integración de un robot móvil y un sistema de visión artificial, para detectar y evadir obstáculos presentes en la trayectoria del robot, empleando una estrategia de navegación de tipo local, basada en campos de potencial.

Algunas aplicaciones de los robot móviles requieren de sistemas de navegación donde la seguridad y la facilidad de operación son los factores más importantes de evaluación, tal es el caso de las sillas de ruedas robotizadas (Kuno, Shimada, & Shirai, 2003). Proporcionar autonomía es una forma de mejorar ambos factores. En este tipo de aplicación resulta conveniente evitar ángulos de giro bruscos que puedan representar un riesgo de lesiones en el usuario, siendo una alternativa de solución un incremento gradual no pronunciado en el ángulo de giro al eludir un obstáculo, es decir un ángulo de giro suave. La evasión de obstáculos empleando sensores infrarrojos, ultrasónicos, de visión, y otros sensores son también motivo de estudio en estos sistemas. En este sentido uno de los sensores utilizados en aplicaciones robóticas, en particular en tareas de navegación de interiores es el sensor Kinect de Microsoft® , el cual utiliza la tecnología de una cámara RGB y una cámara infrarroja para diferenciar profundidad (El-laithy, Huang, & Yeh, 2012). En Mittal y Goyal (2014) se presenta un sistema de navegación para una silla de ruedas que emplea un sistema operativo Linux, sensores infrarrojos y una cámara Kinect, para la evasión de obstáculos, el paso en puertas y el seguimiento de paredes.

Así, en este artículo se presenta el diseño e implementación de un sistema de navegación reactiva difusa, que a partir de los datos de profundidad obtenidos con el sensor Kinect, defina ángulos de giro suave para la navegación de una plataforma/robot móvil. El sistema considera la incertidumbre de información como argumento de su diseño.

2. Sistemas de navegación reactiva difusa

Los sistemas de navegación reactiva basados en control difuso, difieren unos de otros en los algoritmos de procesamiento de información, tipo de sistema de inferencia, el conjunto de reglas, las funciones de membresía, variables de entrada y salida, o en el tipo de sensor que utiliza para la adquisición de información del ambiente.

En algunos sistemas de navegación difusa se plantea el uso de imágenes digitales 2D. Por ejemplo, en Ramírez, Gómez, Martínez, y López (2011) se emplea un sistema de inferencia Mamdani, con funciones de membresía tipo triangular y trapezoidal, los cuales permiten ángulos de giro suaves, sin embargo la detección y evasión de obstáculos en curso de colisión o imprevistos no fue considerada, limitándose únicamente a ambientes tipo pasillo lineal.

Algunos trabajos (Mester, 2010; Bueno & Rios, 2008; Villaseñor et al., 2010; Parra, Ríos, & Bueno, 2007) emplean sensores ultrasónicos y un control difuso con sistemas de inferencia tipo Mamdani; predominan las funciones de

membresía tipo triangular, algunas veces en combinación con funciones de membresía tipo gaussiana y trapezoidal; estos trabajos emplean una navegación reactiva a base de comportamientos. Por su parte Acosta, Gallardo, y Pérez (2016) presentan una arquitectura de control basada en el paradigma reactivo, para la navegación autónoma de robots móviles con capacidades de cómputo limitadas haciendo uso de la teoría de control difuso en una plataforma robótica provista de tres sensores ultrasónicos dedicada al seguimiento de paredes. Sin embargo, el reducido alcance de los sensores ultrasónicos empleados en estos trabajos impide prever colisiones futuras. Además, los giros suaves tampoco son argumentos de dichos trabajos.

Based on the previous statement, despite the vast quantity of tools to help urban modeling, there is a lack of tools devoted to simulate study-cases focusing in city operation, instead of its mere visualization; since modern tools frequently focus in 3D simulation and do not have enough power to deal with issues regarding how chaotic and erratic a real city is.

En Aguirre, Gómez, Muñoz, y Ruiz (2003); Cupertino, Giordano, Naso y Delfino (2006); y Chica, Pineda, y Esmeral (2009) se proponen arquitecturas híbridas de fusión sensorial entre sensores ultrasónicos o infrarrojos con imágenes 2D. Sin embargo, el procesamiento de las imágenes 2D y la fusión sensorial conllevan un alto costo computacional. Por su parte en Cupertino et al. (2006) y Chica et al. (2009) se utilizan funciones de membresía triangulares. En cualquiera de los casos no se hace alusión a los giros suaves como elemento de la navegación, ni se considera la prevención de colisiones futuras.

Otros trabajos proponen sistemas de navegación utilizando el Kinect como sensor principal y la lógica difusa para el procesamiento de la información. En Díaz y Romero (2012) se aplican un filtro Gaussiano a la imagen de profundidad, para posteriormente localizar el centroide de un objeto próximo. Se obtiene una distancia promedio para evadir el obstáculo y se gira a lado contrario de la posición del objeto, deteniendo motores o avanzando hacia delante. Csaba y Vámosy (2012) implementan algoritmos para planificar la ruta y evadir obstáculos, usando un sistema de inferencia Mamdani de 16 reglas, funciones de membresía trapezoidal y triangular, tres variables de entrada (profundidad del objeto más cercano de lado derecho, del lado izquierdo y al frente), y una variable difusa de salida (ángulo de giro del robot móvil). Benavidez y Jamshidi (2011) emplean las imágenes de profundidad del Kinect junto con sensores odométricos. El control difuso utiliza funciones de membresía de tipo triangular. A las imágenes de profundidad se les aplica un filtro Gradiente-Log para identificar áreas transversales (suelos). Estos trabajos basados en el Kinect tienen como característica común que buscan evadir obstáculos en un determinado rango de profundidades, sin considerar lo pronunciado de los giros ni las colisiones futuras.

Los trabajos mencionados están orientados hacia la navegación reactiva y tienen en común aplicar la lógica difusa para determinar el siguiente movimiento a ejecutar por el robot móvil. Utilizan información de profundidad o proximidad para detectar los obstáculos, evadirlos, e incluso extraer las dimensiones, las formas y centroide de los objetos.

3. Propuesta de sistema de navegación para giros suaves

En la Figura 1 se muestra un esquema del sistema de navegación propuesto. El Kinect provee los datos de profundidad los cuales son procesados en una computadora en cuatro etapas para generar finalmente el ángulo de giro suave evitando colisiones futuras e imprevistas. La plataforma móvil de prueba fue el robot con ruedas de tipo diferencial ERA-MOBI, el cual se encuentra equipado con una computadora a bordo y una antena Wifi (Videre Design, 2009). El ángulo de giro es enviado al robot vía sockets empleando el servidor Player/Stage. Player es un popular servidor genérico de código abierto (open-source) empleando en robótica para el control de sensores y actuadores (Amanda, 2010). El robot ERA-MOBI es compatible con una amplia gama de sensores, incluyendo infrarrojos, sonares, telémetros láser y de visión estereoscópica. En los siguientes apartados se describen a detalle cada una de las cuatro etapas que transforman los datos de profundidad del Kinect en un ángulo de giro suave.

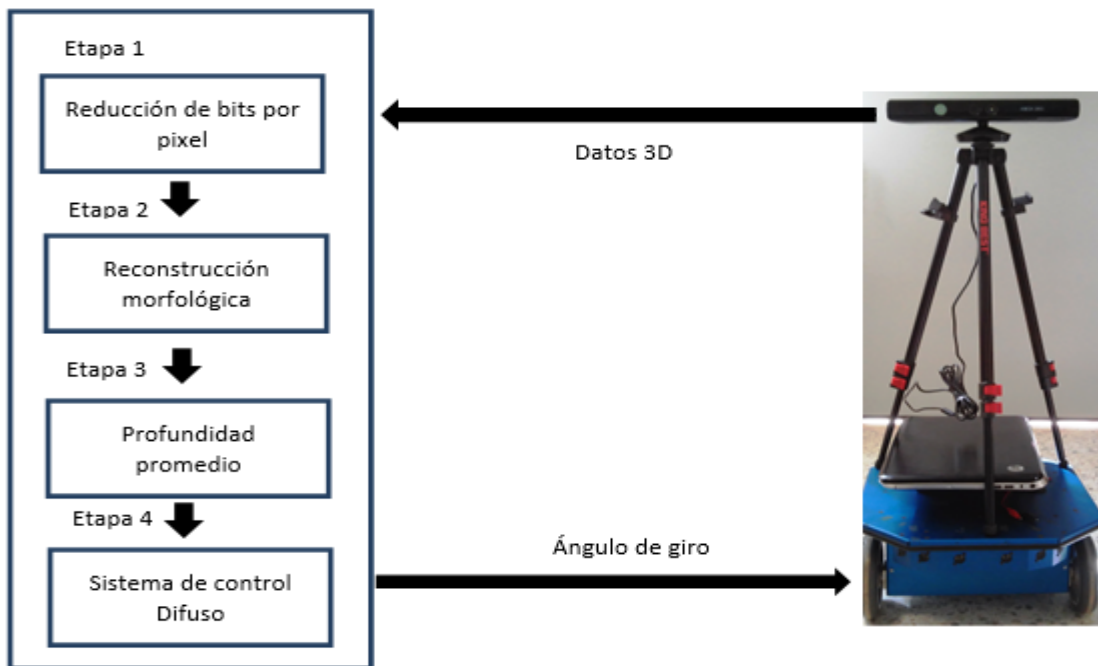


Figura 1. Esquema general del sistema de navegación reactiva difusa, etapa 1: Reducción de bits por pixel, etapa 2: Reconstrucción morfológica, etapa 3: Profundidad promedio, y etapa 4: Sistema de control Difuso.

3.1 Reducción de bits por pixel

La información de profundidad se adquiere mediante el Kinect en una imagen de 640x480, donde cada pixel indica una cantidad de profundidad en una escala de 0 milímetros (color negro) hasta 65,535 milímetros (color blanco), tal como se puede apreciar en la Figura 2a. En esta imagen la mayoría de los pixeles caen dentro del espectro más oscuro ya que superan el valor máximo de visión real del Kinect (10,000 milímetros). El proceso de reducción de bits por pixel transforma la imagen de 16 bits entregada por el Kinect en una de 8 bits (ver Figura 2b), donde el valor 0 representa objetos muy cercanos, objetos muy lejanos u objetos que no fueron identificados. Esto conduce a que el sistema de navegación busque mantener una dirección hacia lugares donde exista suficiente información de profundidad, evitando dirigirse hacia zonas donde predominen valores de 0, esta característica ayuda a evitar posibles colisiones futuras.

3.2 Reconstrucción morfológica

La morfología matemática sobre imágenes digitales permite simplificar los datos de una imagen, conservar las características esenciales y eliminar los aspectos que sean irrelevantes (Parker, 2010). En este caso el proceso de reconstrucción morfológica busca conservar las características de profundidad, y a su vez eliminar pequeñas zonas desconocidas rodeadas por zonas conocidas, brindando así una mayor definición de los objetos y sus profundidades en el ambiente. Esto se logra a través del algoritmo de erosión geodésica (Vincent, 1993), realzando los pixeles con intensidades mínimas siempre y cuando sus pixeles vecinos sean de intensidades mayores, dando como resultado una imagen de profundidad con un menor número de partículas oscuras. En la Figura 2c se muestra un ejemplo de los resultados de la reconstrucción morfológica.

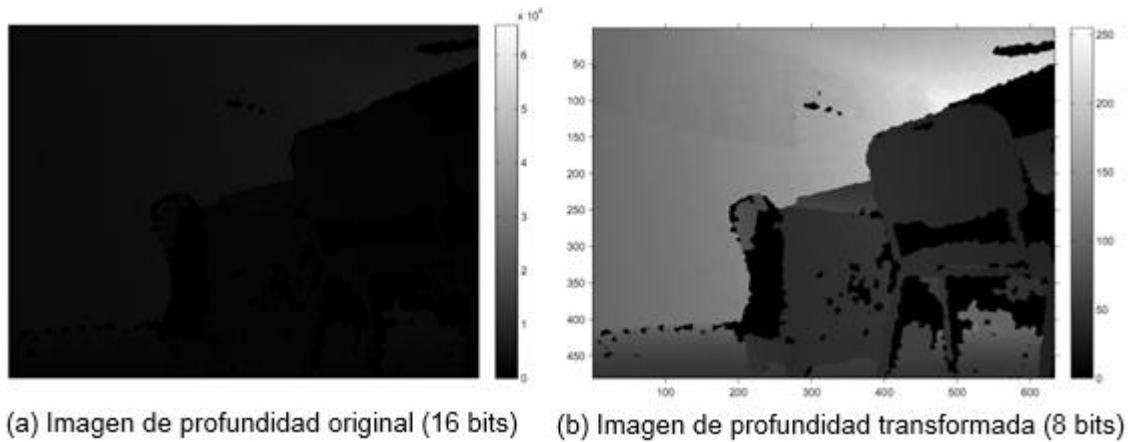


Figura 2. Reducción de bits por pixel y reconstrucción morfológica.

3.3 Profundidad promedio

Un método usado para el tratamiento de la imagen de profundidad es dividir esta en sub-imágenes (Csaba y Vámosy, 2012; Csaba, 2013). En el sistema propuesto la imagen es dividida en tres secciones de 211x480 píxeles, las cuales representan las direcciones posibles en las que la plataforma móvil puede moverse: girar a la derecha, seguir de frente y girar a la izquierda (véase Figura 3).

El sistema emplea el método planteado por Otsu (1979), para calcular el umbral que permita diferenciar las profundidades en valores mayores y menores. Para cada sub-imagen, si la cantidad de datos desconocidos (ceros) es mayor a un determinado porcentaje (67% para este caso), entonces los datos en la sub-imagen indicarán obstáculos muy cercanos, y la profundidad promedio se calculará de la media de las profundidades menores. En caso contrario, se calcula la profundidad promedio de la media de las profundidades mayores. Si en la escena no existen profundidades menores, entonces se consideran los valores de profundidad mayores.

Mediante el cálculo de la profundidad promedio en cada sub-imagen se conoce que región ofrece una menor probabilidad de colisión. A partir de esta información, el sistema de navegación reactiva debe calcular la dirección de giro, considerando los datos imprecisos, y evitando movimientos bruscos. Bajo este escenario se plantea el uso de un control difuso.

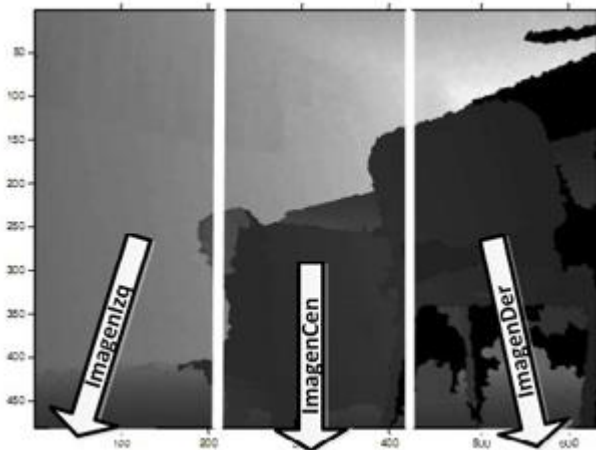


Figura 3. Sub-imágenes de profundidad.

3.4 Sistema de control difuso

Esta etapa consiste en la implementación de un control difuso para la navegación reactiva de la plataforma móvil. Se empleó la herramienta FuzzyLogicToolbox™ de Matlab®, para este desarrollo.

El sistema de inferencia difuso recibe como entrada los valores de profundidad promedio obtenidos en cada sub-imagen, generando como salida una variable en grados sexagesimales que va desde -90° a 90° , el cual representa el ángulo de giro. Se utiliza un sistema de inferencia Sugeno, el cual es computacionalmente efectivo al generar salidas numéricas que no requieren un proceso de defusificación. Los sistemas Sugeno trabajan correctamente con técnicas adaptativas y de optimización (Negnevitsky, 2005; Jang, Sun, & Mizutani, 1997). Además, como se sugiere en Kornuta & Marinelli (2013), su uso en sistemas de navegación garantiza la continuidad de la superficie de salida.

Para cada una de las variables de entrada Izquierda, Frontal y Derecha se definen 3 valores lingüísticos: Cerca representando las profundidades de los “objetos cercanos”, Medio considerando la profundidad de aquellos objetos que no están “ni muy lejos ni muy cerca”, y Lejos para indicar las profundidades de los “objetos alejados” del Kinect. Se hacen uso de funciones de membresía trapezoidales para los valores lingüísticos Cerca y Lejos, y triangular para Medio (véase Figura 4). Este tipo de funciones de membresía no representan

operaciones complejas como es el caso de la función gaussiana o la campana generalizada (Jang et al., 1997).

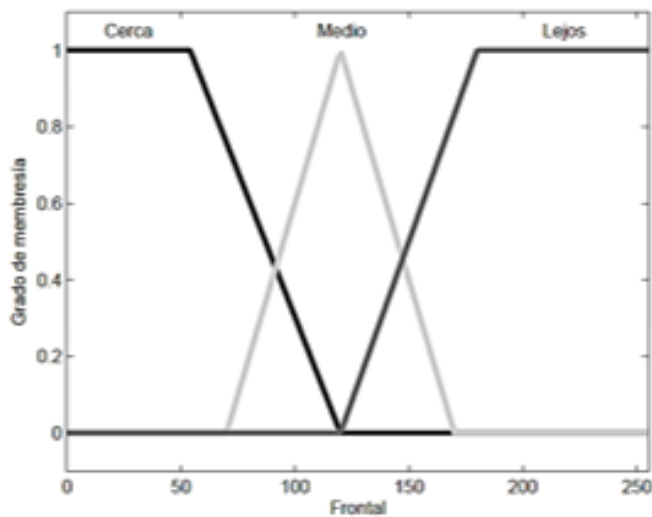


Figura 4. Variable de entrada Frontal: Cerca, Medio, Lejos.

La variable de salida Ángulo se divide en siete valores lingüísticos para interpretar qué tan pronunciado es el giro para navegar. La función para la variable lingüística de salida se define mediante la Ecuación 1.

$$S_i = A_i x + B_i y + C_i z + D_i$$

donde: • S_i : es la variable de salida para cada valor lingüístico i . • A_i, B_i, C_i, D_i : son valores constantes para cada valor lingüístico i . • x, y, z : son las variables de entrada.

Para evitar un comportamiento complejo se utilizó un sistema Sugeno de orden cero (Jang et al., 1997), por lo que la función para cada variable lingüística se define solo en términos de una constante D_i . Empleado como referencia los trabajos realizados por Csada y Vámosy (2012) y Csada (2013), para cada valor lingüístico se definieron las siguientes funciones: $D_1 = 90$ para MuyPositivo, $D_2 = 60$ para Positivo, $D_3 = 30$, para PocoPositivo, $D_4 = 0$, para Cero, $D_5 = -30$, para PocoNegativo, $D_6 = -60$, para Negativo, y $D_7 = -90$, para MuyNegativo.

Dependiendo del número de permutaciones de los valores lingüísticos (Cerca, Medio y Lejos) con respecto a las variables de entrada (Izquierda, Frontal y Derecha), se desarrolla un conjunto de reglas difusas para obtener el valor de salida que corresponde al giro de la plataforma móvil. El control difuso emplea 27 reglas divididas en cuatro grupos dependiendo de un objetivo en común: reglas para la ejecución de giros pronunciados, reglas para la predicción de colisiones, reglas para el movimiento en línea recta, y reglas para giros básicos.

En la Tabla 1 se muestran las reglas difusas planteadas. Las reglas buscan emular la reacción humana durante la navegación, ya que generan reacciones graduales que van desde prácticamente no ejecutar giro hasta los movimientos bruscos ante colisiones imprevistas o en movimiento. El conjunto de reglas tiene como principal objetivo producir giros suaves que permitan asegurar la integridad de la plataforma móvil.

Tabla 1. Reglas difusas.

		SI			ENTONCES
		<i>Izquierda</i>	<i>Frontal</i>	<i>Derecha</i>	<i>Angulo</i>
Conjunto de reglas para giros pronunciados	1	Cerca	Cerca	Cerca	<u>MuyNegativo</u>
	2	Cerca	Cerca	Medio	<u>MuyPositivo</u>
	3	Cerca	Cerca	Lejos	<u>MuyPositivo</u>
	4	Medio	Cerca	Cerca	<u>MuyNegativo</u>
	5	Medio	Cerca	Medio	<u>MuyNegativo</u>
	6	Medio	Cerca	Lejos	<u>MuyPositivo</u>
	7	Lejos	Cerca	Cerca	<u>MuyNegativo</u>
	8	Lejos	Cerca	Medio	<u>MuyNegativo</u>
	9	Lejos	Cerca	Lejos	<u>MuyNegativo</u>
Conjunto de reglas para predecir colisiones	10	Cerca	Medio	Medio	<u>PocoPositivo</u>
	11	Medio	Medio	Cerca	<u>PocoNegativo</u>
	12	Medio	Medio	Lejos	<u>PocoPositivo</u>
	13	Lejos	Medio	Medio	<u>PocoNegativo</u>
	14	Lejos	Medio	Lejos	<u>PocoNegativo</u>
	15	Cerca	Lejos	Lejos	<u>PocoPositivo</u>
	16	Medio	Lejos	Lejos	<u>PocoPositivo</u>
	17	Lejos	Lejos	Cerca	<u>PocoNegativo</u>
18	Lejos	Lejos	Medio	<u>PocoNegativo</u>	
Conjunto de reglas para el movimiento en línea recta	19	Cerca	Medio	Cerca	Cero
	20	Medio	Medio	Medio	Cero
	21	Cerca	Lejos	Cerca	Cero
	22	Cerca	Lejos	Medio	Cero
	23	Medio	Lejos	Cerca	Cero
	24	Medio	Lejos	Medio	Cero
	25	Lejos	Lejos	Lejos	Cero
Conjunto de reglas para giros básicos	26	Cerca	Medio	Lejos	Positivo
	27	Lejos	Medio	Cerca	Negativo

4. Resultados

Las pruebas al sistema de navegación reactiva difusa se ejecutaron en dos entornos diferentes: doméstico y laboratorio de cómputo. En ambos casos, el Kinect se colocó a un metro de altura (con una inclinación vertical de 0°) respecto a la plataforma móvil. El sistema de control difuso trabaja con las imágenes de profundidad, empleando las imágenes RGB únicamente para una visualización más clara del ambiente. La aplicación fue ejecutada en dos equipos de cómputo diferentes, cuyas características se muestran en la Tabla 2.

Tabla 2. Características de los equipos de cómputo para pruebas.

Tipo de Equipo	Marca	Sistema Operativo	Procesador	Velocidad del Procesador	Memoria RAM
Laptop	Toshiba Satellite L855	Windows 7	Intel Core i7	2.30 GHz	8 GB
PC	Dell Vostro 420	Windows XP	Intel Core 2 Duo	3.16 GHz	3 GB

Se realizaron 1000 experimentos para calcular los tiempos de ejecución del sistema propuesto. En la Tabla 3 se observan el tiempo de captura de la imagen, los tiempos de ejecución de cada una de las cuatro etapas, y el tiempo total.

Tabla 3. Tiempos de ejecución promedio.

Tipo de Equipo	Captura de imagen	Normalización de los datos	Reconstrucción morfológica	Profundidad promedio	Control difuso	Tiempo total
Laptop	10.4 ms	3.4 ms	4.1 ms	5.4 ms	15.6 ms	38.9 ms
PC	6.4 ms	6.7 ms	4.8 ms	6.7 ms	18.9 ms	43.5 ms

El cálculo del ángulo de giro mediante el controlador difuso supera el tiempo de las demás etapas indicando un mayor consumo de recursos computacionales. Se observa que los procesos con el menor tiempo promedio varían entre equipos de cómputo. En el caso de la Laptop, el proceso de normalización de datos de profundidad consume el menor tiempo con 3.4 ms. En la PC el proceso de reconstrucción morfológica genera el menor tiempo con 4.8 ms.

El sistema de navegación reactiva difusa ejecuta sus operaciones en un rango de entre 38.9 ms y 43.5 ms. Si se considera una frecuencia de procesamiento (a nivel de hardware) del Kinect de 30 Hz (Webb & Ashley, 2012; Cruz, Lucio, & Velho, 2012; Andersen et al., 2012), lo cual representa un tiempo aproximado de 33.33 ms entre capturas, es posible valorar una adecuada sincronía entre el procesamiento del Kinect y el sistema de navegación.

Durante las primeras pruebas descritas, el Kinect no fue colocado directamente sobre el robot móvil, lo cual conllevó a que el ángulo de giro resultante fuera ejecutado un operador humano. Posteriormente se realizaron pruebas usando el robot ERA-MOBI y la computadora Laptop, realizando el montaje mostrado en la Figura 1. Se programó un cliente en la Laptop y una aplicación cliente/servidor Player en la computadora a bordo del robot para comunicar y ejecutar el ángulo de giro en el robot móvil. En las pruebas realizadas el sistema acertó en las decisiones de dirección de giro evitando obstáculos en un 85.7%.

En la Figura 5 se ilustra el comportamiento del sistema dentro de un ambiente doméstico sin movimiento de los objetos e iluminación natural moderada. En las Figuras 5a y 5b se presentan la imagen RGB y su correspondiente imagen de profundidad. En la Figura 5c se muestra una representación del sistema difuso, el cual identifica las profundidades en cada lado de la imagen otorgándoles valores lingüísticos usando las reglas difusas, para finalmente a través del método de inferencia genere el ángulo de giro en grados (véase Figura 5d). Se observó que el sistema trata de seguir el espacio con la menor probabilidad de colisión manteniéndose estable al continuar en espacios con información conocida. Dependiendo de la cantidad de datos desconocidos, el sistema infiere si debe ignorarlos o considerarlos dentro del cálculo de la profundidad promedio por cada sub-imagen. En el ejemplo de la Figura 5, los datos de profundidad desconocidos son ignorados al no abarcar un área mayor a dos terceras partes de la imagen total.

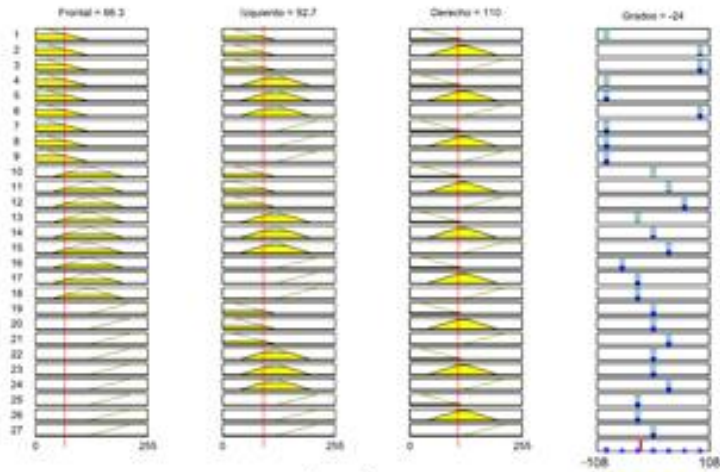
En la Figura 6 se muestra un caso donde las profundidades van desde los rangos cercanos a los lejanos en un ambiente en donde un objeto se va acercando poco a poco al sensor (leves movimientos). En la Figura 6a se presenta la imagen RGB. En la Figura 6b se muestra la imagen de profundidad, en donde se observa cómo algunos elementos toman valores desconocidos. El sensor adquiere la imagen de profundidad para que así el sistema procese la información y reaccione para evadir al objeto en movimiento (bebé en andadera en la Figura 6a). Es de notar que las partes desconocidas no son consideradas por el sistema, ya que a pesar de abarcar un área casi uniforme, no representan un objeto. El sistema define un giro hacia la izquierda de -47.3° (véase Figura 6c) al observar una profundidad promedio mayor (193) en la parte izquierda (véase Figura 6d).



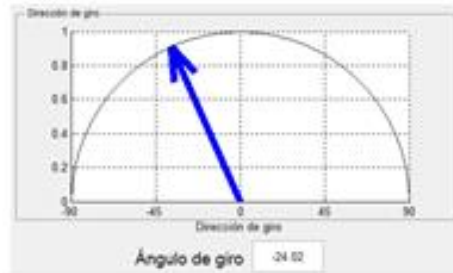
(a) Imagen RGB



(b) Imagen de profundidad



(c) Representación gráfica del sistema difuso



(d) Gráfica de dirección de giro

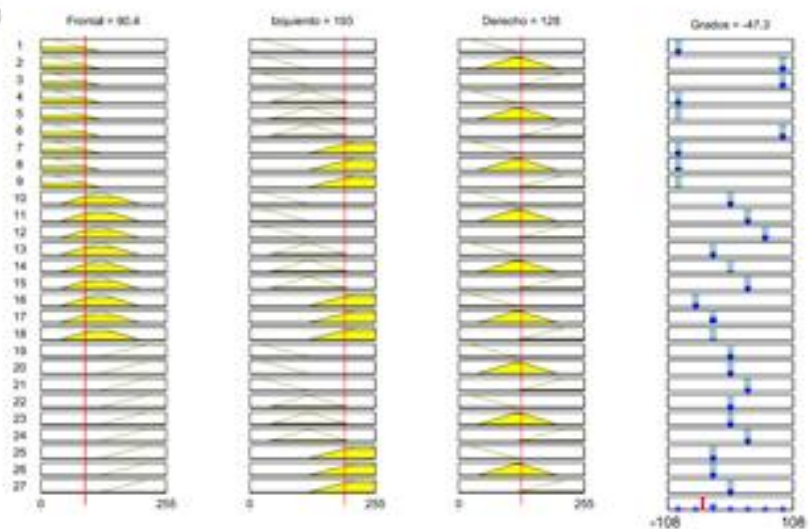
Figura 5. Comportamiento del sistema en ambientes domésticos, sin movimiento de los objetos



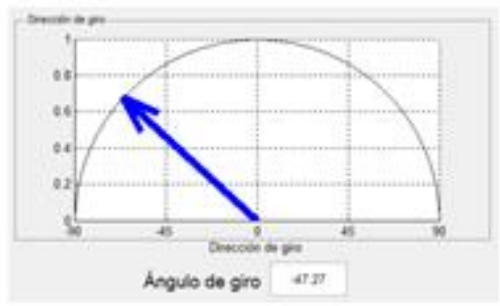
Objeto en movimiento [bebé con andadora]

(a) Imagen RGB

(b) Imagen de profundidad



(c) Representación gráfica del sistema difuso



(d) Gráfica de dirección de giro

Figura 6. Comportamiento del sistema en ambientes domésticos, con ligeros movimientos

El comportamiento de cada variable en función de los datos de profundidad puede ser visualizado en un gráfico 3D. Donde el eje z es el ángulo de giro de la

plataforma móvil, el eje x y y son los valores de profundidad promedio acorde a dos variables de entrada. De esta forma es posible examinar el comportamiento de la variable de salida (ángulo de giro) utilizando combinaciones entre dos entradas que varían como producto de cambios en el entorno (obstáculos o ruido) y una entrada constante, generando los siguientes escenarios:

- Comportamiento con cambios en las sub-imágenes de la parte frontal y el lado derecho, y sin cambios al lado izquierdo (véase Figura 7).
- Comportamiento con cambios en las sub-imágenes de la parte frontal y el lado izquierdo, y sin cambios al lado derecho (véase Figura 8).
- Comportamiento con cambios en las sub-imágenes del lado izquierdo y el lado derecho, y sin cambios en la parte frontal (véase Figura 9).

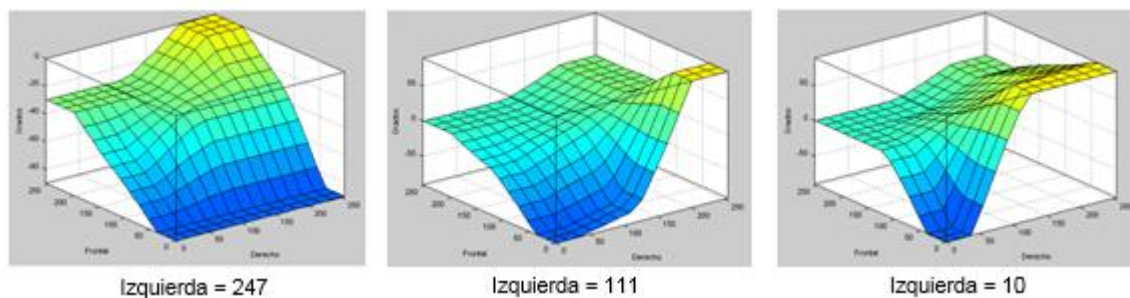


Figura 7. Representación de la relación entre las variables de entrada Frontal y Derecho con respecto a la variable de salida Ángulo.

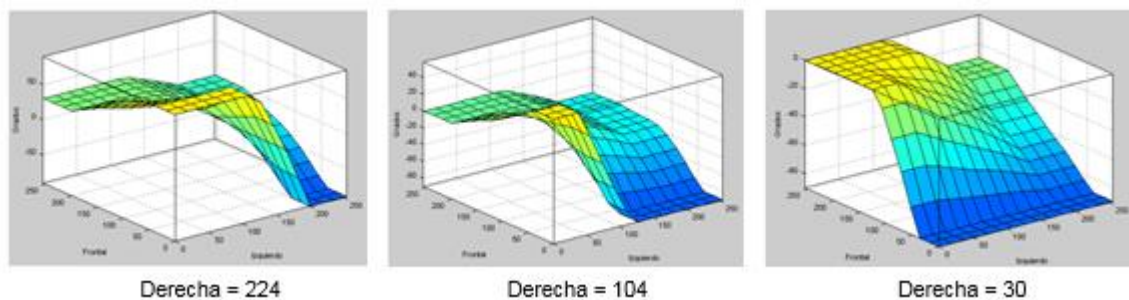


Figura 8. Representación de la relación entre las variables de entrada Frontal e Izquierda con respecto a la variable de salida Ángulo

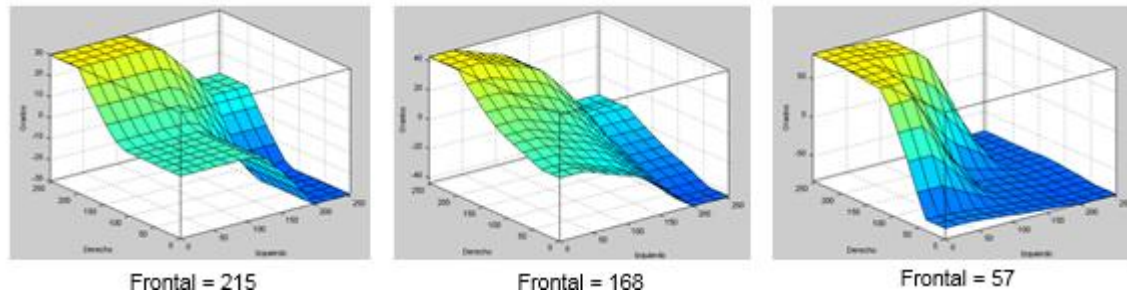


Figura 9. Representación de la relación entre las variables de entrada Derecha e Izquierda con respecto a la variable de salida Ángulo

5. Conclusiones

En este trabajo se presentó un sistema de navegación reactiva difusa para generar giros suaves y anticiparse a futuras colisiones utilizando imágenes de profundidad adquiridas mediante el Kinect. El sistema propuesto cuenta con un conjunto de 27 reglas predictivas que buscan evitar: colisiones futuras considerando las profundidades más alejadas, colisiones con objetos muy cercanos u objetos imprevistos, y giros pronunciados que puedan poner en riesgo la integridad de la plataforma móvil.

La reconstrucción morfológica aplicada permitió reducir la cantidad de información desconocida en la imagen de profundidad, y obtener una mayor definición de los objetos inmersos en la escena. El proceso de reducción de bits por pixel realizado sobre la imagen de profundidad original llevó a reducir el costo computacional de 16 bits a 8 bits, haciendo más factible su implementación en hardware reconfigurable o sistemas basados en microcontroladores. La elección de un sistema de inferencia Sugeno también contribuye a reducir el procesamiento de información.

Las pruebas realizadas corroboraron la viabilidad del sistema para la obtención de giros suaves de plataformas móviles. El sistema de navegación reactiva difusa resulta conveniente en aplicaciones que requieran tolerancia a la incertidumbre de información y/o mecanismos de seguridad ante posibles colisiones.

Una aplicación propuesta para el sistema de navegación reactiva difusa es su incorporación como elemento de control de una silla de ruedas robotizada, considerando que en estos dispositivos el cuidado de la integridad de usuario es la prioridad, siendo la reacción ante objetos imprevistos y la navegación mediante giros suaves dos elementos de utilidad. La integración de algoritmos para discriminar los objetos inmersos en la escena o el suelo como tal, es una línea de trabajo a seguir.

Referencias

Acosta, G., Gallardo, J., & Pérez, R. (2016). Arquitectura de control reactiva para la navegación autónoma de robots móviles. *Ingeniare*, 24 (1), 173-181.

Aguirre, E., Gómez, M., Muñoz, R., & Ruiz, C. (2003). Un sistema multi-agente que emplea visión activa y ultrasonidos aplicado a navegación con comportamientos difusos. En *IV Workshop en Agentes Físicos* (pp. 63-74). Alicante, España: Universidad de Alicante.

Amanda, W. (2010). *Programming Mobile Robots with Aria and Player*. London: Springer

Andersen, M.R., Jensen, T., Lisouski, P., Mortensen, A.K., Hansen, M.K., Gregersen T., & Ahrendt, P. (2012). Kinect Depth Sensor Evaluation for Computer Vision Applications (Technical report ECE-TR-6, Aarhus University, Denmark). Recuperado de http://eng.au.dk/fileadmin/DJF/ENG/PDF-filer/Tekniske_rapporter/Technical_Report_ECE-TR-6-samlet.pdf

Benavidez, P. & Jamshidi M. (2011). Mobile robot navigation and target tracking system. In *6th International Conference on System of Systems Engineering*, (pp. 299-304). Albuquerque, New Mexico, USA: IEEE Reliability Society & IEEE Systems, Man, and Cybernetics Society. doi: 10.1109/SYSOSE.2011.5966614

Bueno, M. & Rios, L. H. (2008). Implementación de comportamientos para navegación inteligente de robots móviles. *Tecnura*, 11 (22), 40-52.

Chica, M. T., Pineda, A. L., & Esmeral, M. A. (2009). Sistema de navegación para robots móviles utilizando fusión sensorial. *Tecnura*, 13 (25), 128-135.

Cruz, L., Lucio, D., & Velho, L. (2012). Kinect and RGBD Images: Challenges and Applications. In *25th Conference on Graphics, Patterns and Images Tutoriais*, (pp. 36-49). Ouro Preto, Brazil: IEEE Computer Society & Brazilian Computer Society. doi: 10.1109/SIBGRAPI-T.2012.13

Csaba, G., (2013). Improvement of an adaptive fuzzy-based obstacle avoidance algorithm using virtual and real kinect sensors. In *9th International Conference on Computational Cybernetics* (pp. 113-120).Tihany, Hungary: IEEE Hungary Section & IEEE Systems, Man, and Cybernetics Society. doi: 10.1109/ICCCyb.2013.6617572

Csaba, G. & Vámosy Z. (2012). Fuzzy based obstacle avoidance for mobil robots with Kinect sensor. In *4th IEEE International Symposium on Logistics and Industrial Informatics* (pp. 135-144). Smolenice, Eslovaquia: IEEE Hungary Section. doi: 10.1109/LINDI.2012.631947

Díaz, C. A. & Romero, C. A. (2012). Navegación de robot móvil usando Kinect, OpenCV y Arduino. *Prospectiva*, 10 (1), 71–78. doi: 10.15665/rp.v10i1.398

El-laithy, R. A., Huang, J.; & Yeh, M. (2012). Study on the use of Microsoft Kinect for robotics applications. In *Position Location and Navigation Symposium* (pp. 1280–1288). Myrtle Beach, South Carolina, USA: IEEE Aerospace and Electronic Systems Society & Institute of Navigation. doi: 10.1109/PLANS.2012.6236985

Jang, J-S.R, Sun, C-T., & Mizutani, E. (1997). *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*. New Jersey, USA: Prentice Hall.

Kelly, A. (2013). *Mobile Robotics: Mathematics, Models, and Methods*. New York, USA: Cambridge University Press.

Kornuta, C. & Marinelli, M. (2013). Estudio comparativo de la eficiencia entre controladores difusos del tipo Mandani y Sugeno. En *XV Workshop de Investigadores en Ciencias de la Computación*. (pp. 926–929). Paraná, Argentina: Universidad Autónoma de Entre Ríos.

Kuno, Y., Shimada N., & Shirai, Y. (2003). A Robotic Wheelchair Based on the Integration of Human and Environmental Observations. *IEEE Robotics & Automation Magazine*, 10 (1), 26-34.

Mester, G., (2010). Intelligent mobile robot motion control in unstructured environments. *Acta Polytechnica Hungarica*, 7 (4), 153–165.

Mittal, R., Goyal, D. (2014). Autonomous Navigation of Smart Wheelchair using Kinect Camera. *International Journal of Engineering and Technical Research*. 2 (4), 357-358.

Mohan, V. (2005). Fuzzy logic controller for an autonomous mobile robot (Master's thesis, Cleveland State University, Ohio). Recuperado de <http://www.eng.auburn.edu/~tropical/courses/5530%202011C%20Robots%20Fall%2011/literature/Mohan%20Thesis%20fuzzy%20logic%20wall%20follow.pdf>

Murphy, R. R. (2000). *An introduction to AI robotics*. Cambridge, Massachusetts, USA: The MIT Press.

Negnevitsky, M. (2005). *Artificial Intelligence: A Guide to Intelligent Systems* (2nd ed.). England: Addison-Wesley.

Ollero A. (2001). *Robótica: manipuladores y robots móviles*. Barcelona, España: Marcombo Alfaomega.

Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9 (1), 62–66. doi:10.1109/TSMC.1979.4310076

Parker, J. (2010). *Algorithms for image processing and computer vision*. 2a Ed. USA: Wiley Publishing, Inc.

Parra, H., Ríos L. H., & Bueno M. (2007). Navegación de robots móviles mediante comportamientos utilizando lógica difusa. *Scientia Et Technica*, 13 (34), 79–83.

Ramírez, J. M., Gómez, P., Martínez, J., & López, F.(2011). A LabVIEW-based autonomous vehicle navigation system using robot vision and fuzzy control. *Ingeniería Investigación y Tecnología*, 12(2), 129–136.

Suárez, A. F. & Loaiza, H. (2015). Implementación de un esquema de navegación reactiva con sensores RGB-D. *UIS Ingenierías*, 14 (1), 7-19.

Tzafestas, S. G. (2014). *Introduction to Mobile Robot Control*. Elsevier

Videre Design (2009). *ERA Mobile Robot User's Manual (Rev H, October 2009)*. Videre Design LLC.

Villaseñor, U. G., González, M. A., Sotomayor, A., Gorrostieta, E., Pedraza, J. C., Vargas, J. E. & Tovar S. (2010). Desarrollo de un sistema de navegación para robots móviles mediante diferentes patrones de comportamientos. En VIII Congreso Internacional sobre Innovación y Desarrollo Tecnológico (pp. 407-412). Cuernavaca, Morelos, México: EEE sección Morelos y AMIME sede Morelos.

Vincent, L. (1993). Morphological gray scale reconstruction in image analysis: applications and efficient algorithms. *IEEE Transactions on Image Processing*, 2 (2), 176–201.

Webb, J.& Ashley, J. (2012). *Beginning Kinect Programming with the Microsoft Kinect SDK*. New York, USA: Apress.

Yerubandi, V., Reddy, Y. M., & Kumar, M. V. (2015). Navigation system for an autonomous robot using fuzzy logic. *International Journal of Scientific and Research Publications*. 5 (2). Recuperado de <http://www.ijsrp.org/research-paper-0215/ijsrp-p3807.pdf>

Notas biográficas:



Claudia Cruz Martínez nació en Oaxaca, México en 1989. Recibió el grado de Ingeniero en Computación por la Universidad del Istmo en 2014. Fue becaria del Cuerpo Académico de Ingeniería en Computación. Actualmente se encuentra estudiando la Maestría en Ciencias Computacionales en el Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE). Su área de interés se centra en el uso de la visión 3D, cómputo de alto desempeño y aplicaciones de tiempo real enfocados al SLAM visual.



Ignacio Alfredo Badillo . recibió los grados de Ingeniero en Electrónica y de Maestro y Doctor en Ciencias Computacionales por parte del Instituto Tecnológico de Puebla (ITP) y del Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE), respectivamente. Él se desempeña como Profesor de Tiempo Completo en la Universidad Politécnica de Tlaxcala. Él es miembro del SNI desde el 2011, tiene más de 70 artículos científicos, es revisor en varios comités nacionales e internacionales, liderea varios proyectos y sus líneas de investigación son enfocadas en el diseño y desarrollo de sistemas digitales, arquitecturas reconfigurables, plataformas basadas en el concepto radio software, seguridad, prototipos didácticos, sistemas criptográficos, implementaciones en FPGA, sistemas basados en microcontroladores y microprocesadores y aceleración en hardware para aplicaciones específicas.



J. Jesús Arellano Pimentel es ingeniero en Sistemas Computacionales por el Instituto Tecnológico de Morelia en 2002 y Maestro en Ciencias en Ingeniería Eléctrica con opción en Sistemas Computacionales por la Universidad Michoacana de San Nicolás de Hidalgo en 2005. Actualmente se desempeña como Profesor-Investigador de Tiempo Completo adscrito a la carrera de Ingeniería en Computación en la Universidad del Istmo campus Tehuantepec, Oaxaca, México. También es miembro activo del Cuerpo Académico de Ingeniería en Computación de la UNISTMO. Además ha obtenido el Reconocimiento a Perfil Deseable PRODEP durante los periodos 2013-2016 y 2016-2019. Sus áreas de interés incluyen: software educativo, realidad virtual, prototipos didácticos, compiladores, y robótica móvil.



Francisco Aguilar Acevedo es ingeniero en electrónica por la Universidad Tecnológica de la Mixteca. Tiene una maestría en ingeniería mecatrónica por el Centro Nacional de Investigación y Desarrollo Tecnológico. Es Profesor-Investigador de Tiempo Completo en la carrera de Ingeniería en Computación en la Universidad del Istmo Campus Tehuantepec, Oaxaca, México.

También es miembro activo del Cuerpo Académico de Ingeniería en Computación de la UNISTMO. Sus áreas de interés son la dirección de proyectos de ingeniería, los sistemas embebidos, la instrumentación electrónica y la automatización de procesos.



Ernesto Cortés Pérez recibió el grado de M. C. en Ciencias de la Computación por ITA-LITI (Laboratorio de Investigación en Tecnologías Inteligentes) en el Instituto Tecnológico de Apizaco, Tlaxcala, México. Desde 2007 ha sido Profesor-Investigador de Tiempo Completo en la Universidad del Istmo UNISTMO Campus Tehuantepec, Oaxaca, México. También es miembro activo del

Cuerpo Académico de Ingeniería en Computación de la UNISTMO. Su actual área de interés incluye: sistemas inteligentes, lógica difusa, clasificación de patrones, redes neuronales artificiales, algoritmos bio-inspirados y visión artificial.



Luis Alberto Morales Rosales recibió el grado de ingeniero en Sistemas Computacionales del Instituto Tecnológico de Colima. Los grados de maestría y doctorado del Instituto Nacional de Astrofísica Óptica y Electrónica (INAOE). Ha sido Profesor-Investigador de Tiempo Completo en la Universidad Politécnica de Puebla (2007-2010). Coordinó y fundó el programa de

posgrado en sistemas computacionales del Instituto Tecnológico Superior de Misantla (2010-2016), cuenta con reconocimiento al perfil deseable por parte de PRODEP desde 2011. Actualmente es catedrático CONACYT asignado a la Universidad Michoacana de San Nicolás de Hidalgo. Sus líneas de investigación son enfocadas al cómputo móvil, sistemas distribuidos, desarrollo de algoritmos bio-inspirados, computación inteligente, y sistemas de transporte inteligente.



Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 2.5 México.