

# Vehículo Autónomo Controlado por Visión.

## Javier Vargas Occa

Investigador  
Escuela de Ingeniería Eléctrica y  
Electrónica  
Universidad del Valle

## Andrés Felipe Hurtado

Docente investigador  
Institución Universitaria Tecnológica de  
Comfacauca  
email: ahurtado@unicomfacauca.edu.co  
Sede Santander de Quilichao - Cauca

**Resumen:** este artículo presenta la descripción detallada del diseño y construcción de un vehículo con capacidad para navegar y llegar a un sitio previamente establecido, utilizando para ello información visual obtenida de una cámara con visión global del entorno. De esta manera, se integran conceptos y hardware del modelismo para conjugar estas herramientas con la visión por computador, obteniendo un sistema completo de navegación únicamente utilizando, como elemento sensor, una sencilla máquina de visión.

**Palabras clave:** visión artificial, robótica móvil, navegación, control visual.

## INTRODUCCIÓN

La expansión de la robótica en los últimos años ha sido extraordinaria, pero quizá lo más relevante haya sido la aparición, en la última década, de los robots móviles. Su aplicación en la industria, centrales nucleares, exploración del espacio y situaciones peligrosas en general, ha demostrado su importancia como herramienta de trabajo.

Dentro del sistema sensorial del que están dotados estos robots, la visión artificial juega un papel predominante, el conocimiento y desarrollo de esta tecnología es incuestionablemente significativa en la actualidad, pues su penetración en ambientes y procesos industriales ha venido creciendo de forma considerable.

En este artículo, se presenta la plataforma desarrollada que permite la implementación del control por visión de un vehículo, lo que haría factible la aplicación y desarrollo de distintas técnicas de análisis de imágenes y planeación de trayectorias.

## ROBOTS MÓVILES PROGRAMABLES

Los robots móviles programables son máquinas que utilizan como sistema central de control (cerebro) microprocesadores o microcontroladores. Este sistema central le indica al robot las tareas que debe realizar, como ordenar la revisión de la información del sistema sensorial, es decir, leer los sensores y posteriormente interpretar las lecturas para tomar las decisiones respectivas. Todas estas acciones que se han indicado anteriormente tienen que ser descargadas al sistema. Estos programas pueden ser sencillos, como ejecutar una pequeña rutina de movimiento, o llegar a un complejo sistema de navegación que le brinde completa autonomía al robot. Para la programación de este tipo de robots, se pueden tomar diferentes caminos de acuerdo con la aplicación que se desee implementar. Se puede utilizar toda la información disponible del sistema para analizar las posibles reacciones ante un evento, lo cual supone gran complejidad en el algoritmo y un tiempo de respuesta alto. También se tiene la programación utilizando prioridades, estas se asocian

de acuerdo con la importancia que tenga determinada actividad, para de igual forma reaccionar. Por último, está uno de los métodos más novedosos actualmente, se trata de algoritmos de inteligencia artificial que utilizan redes neuronales, algoritmos genéticos o lógica difusa. Un robot programado a partir de estas herramientas puede aprender y reaccionar de forma natural ante diferentes situaciones, buscando imitar las capacidades del ser humano.

### Procesamiento digital de imágenes

El procesamiento digital de imágenes vive un momento ascendente, la aparición de nuevas máquinas con grandes velocidades ha permitido su evolución hasta llegar, hoy en día, a la experimentación con imágenes cada vez más complejas. Así, el procesamiento abarca dos ideas fundamentales [1], mejorar la información de una imagen para interpretarla y procesarla y aplicar técnicas para que las imágenes sean reconocidas por sistemas computarizados (visión por computadora). Las operaciones fundamentales del procesamiento digital de imágenes se pueden observar en la figura 1 e incluyen:

- Captura de la imagen
- Procesamiento de la imagen
- Análisis de la imagen
- Comprensión de la imagen

### Modelo de color

Para realizar el procesamiento de las imágenes en colores se requiere de un modelo de color, el cual permita manipular las características de éste de forma normalizada [2].

El modelo RGB presenta cada color en sus componentes primarios espectrales, rojo, verde y azul. La representación se realiza a través del cubo unitario mostrado en la figura 2, en el que todos los valores han sido normalizados. Los colores son puntos dentro o sobre el cubo y se definen como vectores que se extienden desde el origen [3].

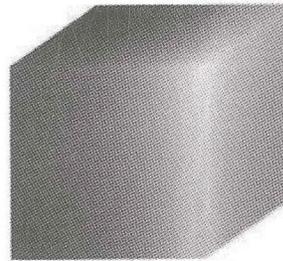
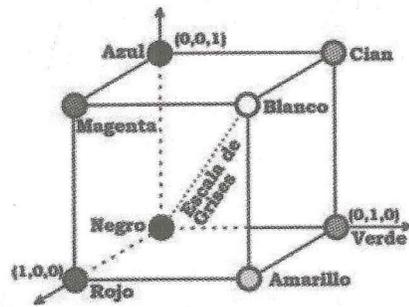


Figura 2. Cubo modelo RGB

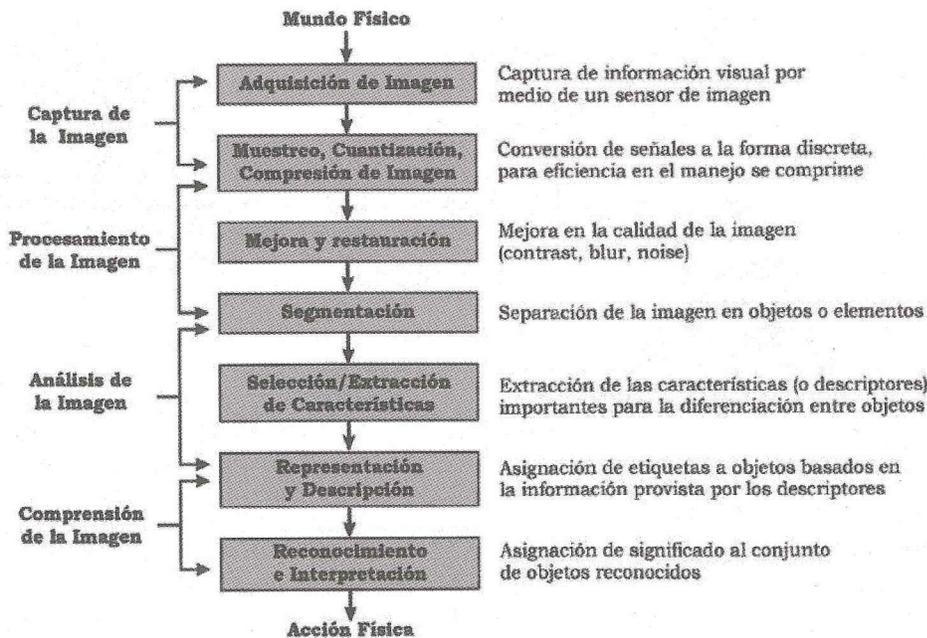


Figura 1. Operaciones del procesamiento digital de imágenes.

En el modelo RGB las imágenes están formadas por tres planos de imagen independientes (colores primarios). Debido a su gran difusión, la mayoría de las cámaras en color lo usan y es necesario trabajar inicialmente con él.

Este modelo no produce buenos resultados cuando se desea resaltar imágenes en color de rostros humanos [3] u otro tipo de situaciones, por ello no siempre es la mejor representación de color para procesamiento de imágenes, especialmente en condiciones de iluminación cambiantes y cuando se desean extraer atributos de color de los objetos en las escenas.

Por su parte, el modelo HSI brinda facilidades para el análisis y trabajo de sistemas que requieren de la percepción del color con alta calidad (visión humana). El matiz es un atributo que describe la pureza de un color (puro amarillo, naranja o rojo), mientras que la saturación da una medida del grado en el que un color puro se diluye con luz blanca. En el modelo HSI el componente de intensidad  $I$  se desacopla de la información de color en la imagen [3], los componentes de matiz ( $H$ ) y saturación ( $S$ ) simulan de forma aproximada cómo los humanos perciben el color.

En el triángulo que representa los componentes del modelo HSI (figura 3) se puede observar que para un punto  $P$ , su componente  $H$  es el ángulo formado entre el eje rojo y el vector  $P$ . Cuando  $H=0^\circ$ , el color de  $P$  es rojo; cuando  $H=60^\circ$ , el color es amarillo y si  $H=120^\circ$ , el color es verde. El componente de saturación es proporcional a la distancia de  $P$  al centro del triángulo. El componente  $I$  se mide con respecto a una línea perpendicular que pasa a través de su centro.

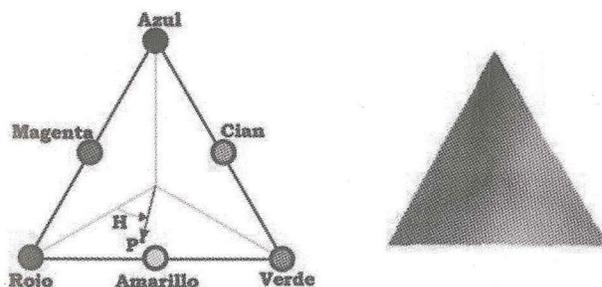


Figura 3. Componentes del modelo HSI

Para obtener los parámetros HSI a partir del modelo RGB normalizado se deben aplicar las siguientes ecuaciones:

$$I = \frac{1}{3} R + G + B \quad (1)$$

$$S = 1 - \frac{3}{R+G+B} [\min R, G, B] \quad (2)$$

$$H = \cos^{-1} \left\{ \frac{1/2 [R-G + R-B]}{\sqrt{R-G^2 + R-G G-B}} \right\} \quad (3)$$

Donde  $R$ ,  $G$  y  $B$  son los niveles de intensidad en cada plano del modelo RGB.

## DESCRIPCIÓN DEL HARDWARE DEL SISTEMA

El vehículo fue diseñado teniendo como base el máximo aprovechamiento del espacio, el tamaño del ambiente y de los obstáculos. Como la configuración de la escena incluye una cámara con visión global de ésta, se convierte en una necesidad la independencia del vehículo de cables que sirvan de alimentación o control, pues estos cables se interpondrían entre los objetos y la cámara, interfiriendo con la representación que ésta hace de la escena. Debido a esto, el vehículo se diseñó partiendo de un sistema completo de RF para automodelismo el cual consta de: un transmisor de dos canales con un potenciómetro para el control de cada canal, un receptor de dos canales y dos servomotores. Teniendo los dos servomotores con control independiente el vehículo se diseñó con tres llantas; dos llantas motrices delanteras con control independiente de tracción y una rueda trasera de pivote.

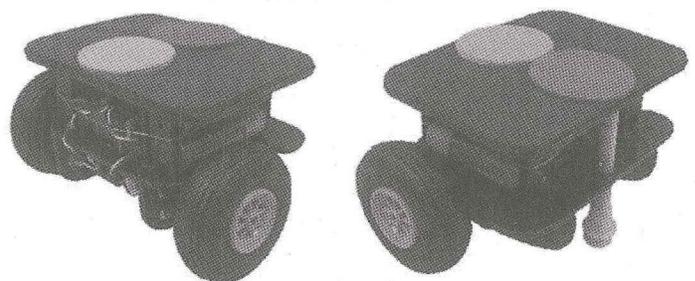


Figura 4. Robot móvil

El sistema de locomoción puede realizar tres movimientos básicos: adelante, izquierda y derecha, como se ilustra en la figura 5. Cuando el vehículo debe avanzar los servos,  $S1$  y  $S2$ , giran en igual sentido; para realizar los giros los servos se mueven en sentidos opuestos.

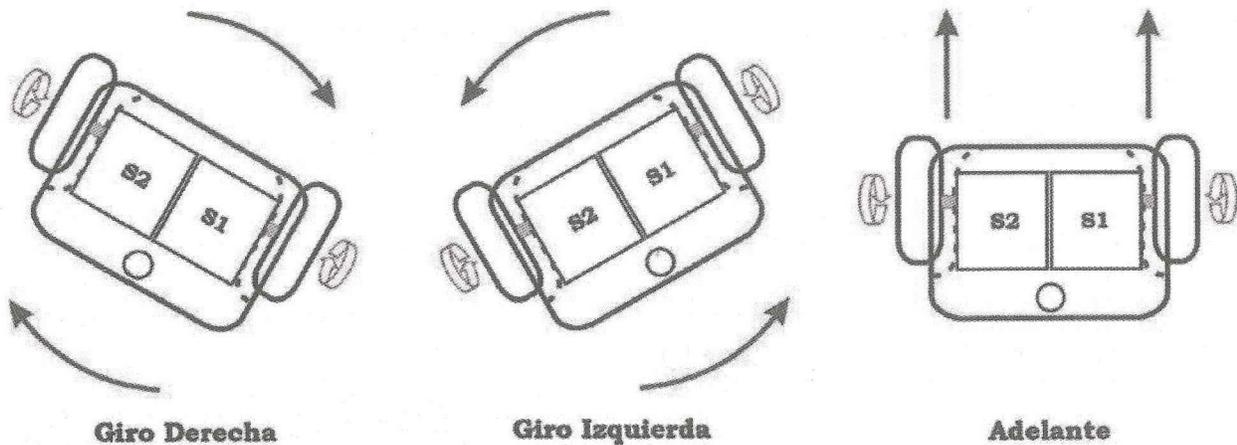


Figura 5. Movimientos del vehículo

El control central del sistema se encuentra en el computador, éste se encarga de hacer todo el procesamiento para realizar los movimientos del vehículo, estos movimientos son el efecto de las órdenes creadas inteligentemente para evitar obstáculos y llegar a un destino de la forma más eficiente posible. Las órdenes son enviadas remotamente, para ello se diseñó una tarjeta que permite la comunicación entre el PC (puerto paralelo) y el vehículo. La tarjeta se conecta a un módulo emisor Futaba FP-T2NL, el cual envía la información al receptor ubicado en el robot móvil.

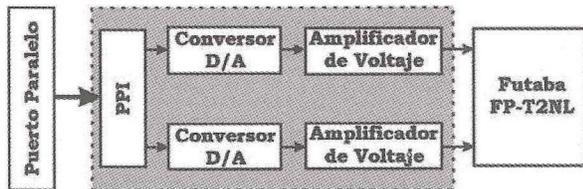


Figura 6. Diagrama de bloques tarjeta de comunicación

El sistema central de control (computador) recibe la información y se realimenta a través de una cámara, la cual se ubica estáticamente para obtener una vista superior del ambiente en el cual actúa el robot móvil.

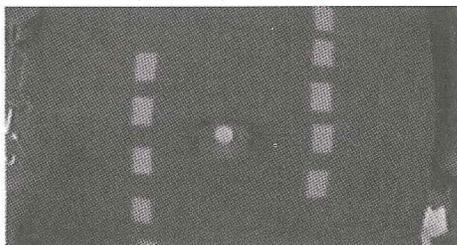


Figura 7. Perspectiva de la cámara

Se empleó una pequeña cámara inalámbrica XCam2, con una frecuencia de 2.4 GHz que transmite video a color. El sistema de adquisición de imagen cuenta con la cámara XCam2, el receptor modelo VR36A y el convertor de RCA a USB modelo VA11A [4]. Las especificaciones técnicas de la cámara se presentan en la tabla 1.

Tabla 1. Especificaciones XCam2

Área de sensado	1/3" Ovos
Distancia focal	3.0 mm±5%
Distancia focal trasera	5.84 mm±5%
Extensión del enfoque	20 cm hasta infinito
Iris	Fixed
Rangote temperatura	-20°C a +60°C
Limites de humedad	0 a 95%
Angulo (diagonal)	60 deg

## DESCRIPCIÓN DEL SOFTWARE DEL SISTEMA

El componente de software de este proyecto puede dividirse básicamente en 5 partes, como se muestra en la siguiente figura:

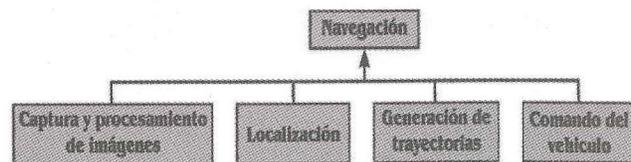


Figura 8. Estructura del software del sistema

**Captura y procesamiento de imágenes:** incluye el manejo del hardware de adquisición, la corrección de la distorsión de la cámara, la conversión al modelo de

color HSI y la segmentación de la imagen.

**Localización:** se encarga de la extracción de información útil de la imagen segmentada, tal como la posición de los obstáculos y la posición y la orientación del vehículo.

**Generación de la trayectoria:** con base en la información obtenida de la localización (de obstáculos y vehículo), la imagen planea una trayectoria a un destino determinado por el usuario, logrando evadir los obstáculos.

**Comando del vehículo:** transforma la información obtenida, al calcular la trayectoria, en datos adecuados para que la interfaz de radiocontrol opere el vehículo.

**Navegación:** combina las cuatro partes anteriores, con el fin de lograr el control del vehículo mediante la realimentación visual.

El vehículo responde a comandos con información de ángulos (orientación) y distancia. Teniendo la trayectoria en estos términos, se evalúa si está constituida sólo por dos puntos, que sería el caso de una línea recta. Además, se pregunta si la distancia del vehículo al destino final es menor que 10 píxeles, que es la desviación promedio calculada para el comando de avance del vehículo. En caso de que ambas sentencias sean afirmativas se da por terminada la navegación y se asume que el vehículo llegó al destino. Si alguna de las sentencias anteriores no se cumple, se orienta al vehículo en dirección del próximo punto y se avanza un tramo de distancia igual a 40 píxeles. Si la distancia a éste es menor que 40 píxeles se transmite su valor. Este valor de avance es configurable y puede ser modificado desde la aplicación. Se puede notar que la trayectoria es recalculada en su totalidad cada vez que el vehículo hace una aproximación, éstas no dejan de hacerse mientras el punto destino esté por fuera de la circunferencia que contiene al vehículo. Lo anterior proporciona al vehículo la capacidad de reaccionar ante un cambio en la configuración de la escena.

## PRUEBAS SOBRE LA EJECUCIÓN DE TRAYECTORIAS.

Para probar el funcionamiento general del sistema se dispusieron obstáculos en la escena, en configuraciones variadas que pudieran mostrar las capacidades del vehículo y medir su desempeño. Las pruebas implementadas se denominan:

- Trayectoria tipo U – obstáculos estáticos

- Trayectoria tipo pasillo – obstáculos estáticos
- Trayectoria con obstáculos dinámicos

Los obstáculos se acomodan haciendo una U invertida en la escena, de forma tal que el vehículo debe rodearlos para alcanzar el punto final (figura 9). El diseño del programa permite que el vehículo realice una trayectoria cercana a los obstáculos, es decir, el vehículo rodea los obstáculos guardando un margen para no chocar con ellos.

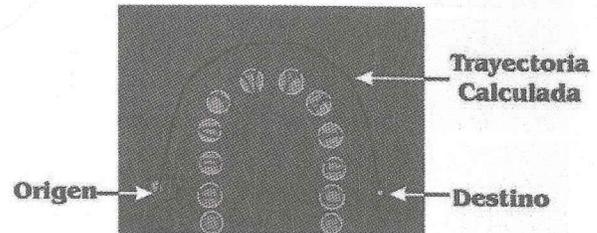


Figura 9. Distribución de obstáculos para trayectoria tipo U

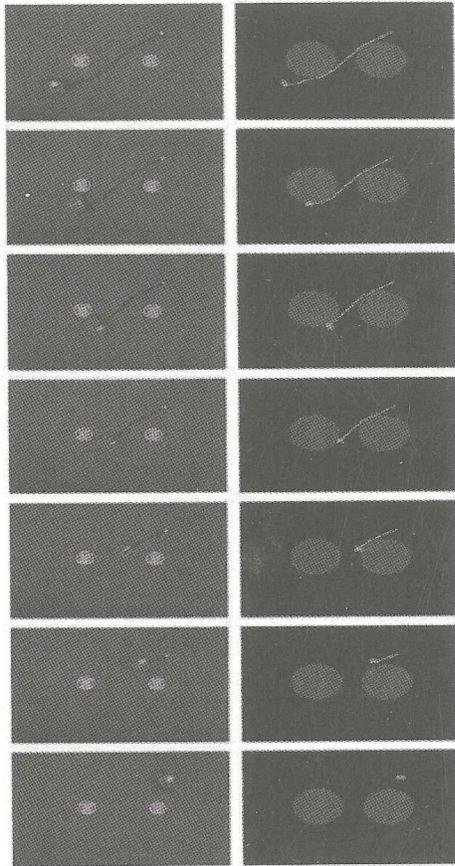
Para analizar el comportamiento del sistema, ante este tipo de trayectorias, se llevaron a cabo cuatro ensayos, en los cuales se recogió información como el número de tramos recorridos, distancia recorrida, tiempo, etc., los cuales permiten cualificar el desempeño del vehículo para este tipo de trayectoria.

La secuencia de imágenes del primer ensayo se muestra en la figura 10. Luego de escoger un punto final, ubicado al otro extremo de la U invertida, se ejecuta la trayectoria que calcula el sistema.

En la secuencia de imágenes se puede observar como el vehículo alcanza el punto destino a través de desplazamientos cortos sucesivos.

Para realizar un desplazamiento se calcula la trayectoria completa entre el punto actual y el destino, posteriormente se transmite al vehículo la información necesaria para recorrer un tramo. De esta forma, se mantiene el control del vehículo por medio de la realimentación visual en ambientes estáticos o dinámicos.

En la tabla 2 se consignan los datos obtenidos al realizar cuatro ensayos, para la misma configuración de trayectoria tipo U, con un nivel de intensidad de luz suministrado por las dos lámparas encendidas, más la iluminación del espacio de trabajo.



Posición del vehículo (172,71), Orientación 171°  
 Distancia al punto próximo 59 píxeles, Orientación 108°  
**Rotar Vehículo 63° derecha, Avanzar 40 píxeles.**

Posición del vehículo (158,99), Orientación 102°  
 Distancia al punto próximo 26 píxeles, Orientación 98°  
**Rotar Vehículo 4° derecha, Avanzar 26 píxeles.**

Posición del vehículo (154,127), Orientación 90°  
 Distancia al punto próximo 22 píxeles, Orientación 131°  
**Rotar Vehículo 41° izquierda, Avanzar 22 píxeles.**

Posición del vehículo (138,143), Orientación 138°  
 Distancia al punto próximo 61 píxeles, Orientación 140°  
**Rotar Vehículo 2° izquierda, Avanzar 40 píxeles.**

Posición del vehículo (105,168), Orientación 140°  
 Distancia al punto próximo 31 píxeles, Orientación 133°  
**Rotar Vehículo 7° derecha, Avanzar 31 píxeles.**

Posición del vehículo (78,189), Orientación 140°  
 Distancia al punto próximo 44 píxeles, Orientación 133°  
**Rotar Vehículo 7° derecha, Avanzar 40 píxeles.**

Posición del vehículo (59,226), Orientación 108°  
 Distancia al punto próximo 4 píxeles, Orientación 75°  
**Se asume que el vehículo llega a destino**

Figura 10. Ejemplo navegación a un punto destino

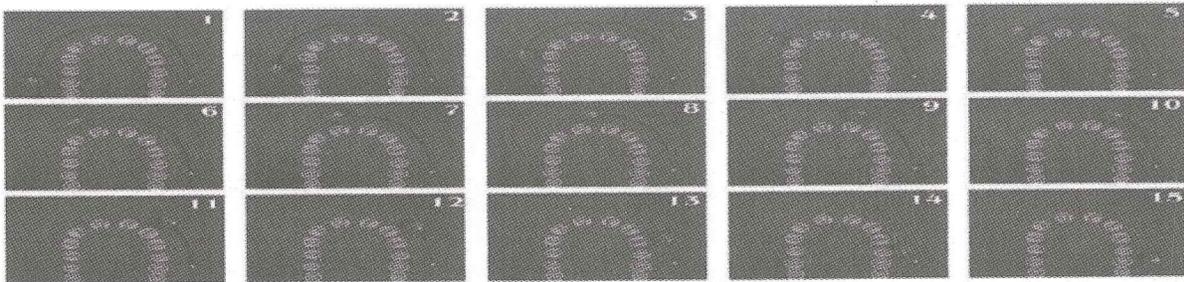


Figura 11. Recorrido de trayectoria tipo U

Tabla 2. Resultados trayectoria tipo U

Datos Obtenidos	Prueba1	Prueba2	Prueba3	Prueba4
Número de Obstáculos	12	12	12	12
Número tramos Recorridos	14	14	13	13
Destino fijado	196,274	194,282	194,276	190,274
Punto de llegada	193,269	197,278	191,268	186,270
Error (píxeles)	5,83	5,00	8,54	6,65
Distancia Recorrida (píxeles)	439	450	433	441
Tiempo empleado (segundos)	78	92	82	80
Velocidad Promedio (pix/seg)	5,63	4,89	5,28	5,51
Obstáculos no evadidos	0	1	0	0

En todos los ensayos, y durante todas la planeaciones de trayectorias, el sistema detectó el mismo número de obstáculos, esto muestra la confiabilidad de los algoritmos empleados y la repetibilidad de sus resultados. La trayectoria de los dos primeros ensayos se cumple en 14 aproximaciones, mientras que la de los dos siguientes en 13. Esto se debe a la imprecisión en los movimientos del vehículo, que deben ser corregidos cada vez que se planea una trayectoria y se realiza una aproximación, haciendo variable el número de aproximaciones necesarias para llegar al destino. El error del punto de llegada, con respecto al punto deseado, se mantiene dentro del margen establecido de 10 píxeles y la velocidad promedio de los 4 intentos es 5.32 píxeles/segundo.

## CONCLUSIONES

Se presentan las fases de diseño del sistema, que combina conocimientos de robótica con visión artificial, en una aplicación confiable y robusta de control por visión. Los procesos de adquisición y procesamiento de imagen permiten obtener información clara de la escena, haciendo más livianas las etapas superiores de procesamiento.

El proceso de localización permite la detección de obstáculos y su diferenciación del vehículo, para luego planear una trayectoria de recorrido mínimo que alcance el destino final. Todos estos procesos, unidos al sistema de control por RF del vehículo, hacen posible que éste navegue bajo el control que posee la cámara.

El diseño del vehículo, de dimensiones reducidas, cuenta con gran movilidad, puede desplazarse hacia adelante, izquierda o derecha. Su diseño le permite realizar giros sobre su centroide, característica que le facilita el desplazamiento en espacios pequeños y disminuir los choques con obstáculos que puedan estar a su paso.

El sistema posee gran robustez ante diferentes cambios en la iluminación, gracias al empleo del modelo de color HSI, el cual permite explotar técnicas para el procesamiento de imágenes con características del sistema de visión humano, en particular, frente a la separación del componente de intensidad de la información de color.

## REFERENCIAS

- [1] R. Bonifaz Solórzano. *Robots de Exploración*. Puebla, Universidad de las Américas, Departamento de Ingeniería en Sistemas Computacionales, Escuela de Ingeniería, 2003.
- [2] R. González, C. Woods, E. Richard. *Tratamiento Digital de Imágenes*. Wilmington Delaware, Addison-Wesley, 1996.
- [3] H. Loaiza. *Introducción a los sistemas de visión en colores. Energía y Computación*, 1998, pp. 12-19.
- [4] *XCam2 Wireless Video Camera*. Disponible en <http://www.x10.com>
- [5] P. Arboleda y L. González. *Diseño de un robot móvil que evade obstáculos controlado con lógica difusa*. Cali, Universidad del Valle, Escuela de Ingeniería Eléctrica y Electrónica, 1999.
- [6] A. Casals. *Sensor devices and systems for robotics*. New York: Springer-Verlag, 1989.
- [7] *Futaba RC*. Disponible en <http://www.futaba-rc.com/servos/index.html>
- [8] B. K. Ghosh, Ning Xi y T. J. Tam. *Control in Robotics and Automation: Sensor-based Integration*. San Diego, Academic Press, 1999.
- [9] D. R. Malcolm. *Robotics: an Introduction*, Boston: PWS-KENT Pub. Co., Segunda ed., 1988.
- [10] *Microsoft Development Network*, Disponible en [http://msdn.microsoft.com/library/default.asp?url=/library/en-us/multimed/htm/\\_win32\\_video\\_for\\_windows.asp](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/multimed/htm/_win32_video_for_windows.asp)
- [11] J. A. Rehg. *Introduction to robotics: a systems approach, Englewood Cliffs*. N.J., Prentice-Hall, 1985.
- [12] *Seattle Robotics Society*, Disponible en: <http://www.seattlerobotics.org/guide/servohack.html>
- [13] R. Swain y M. Devy. *Navegación de un Robot Móvil por Medio de Control Visual en Ambiente Estructurado*. Primer encuentro de computación, 1997.
- [14] R. Y. Tsai. *A Versatile Camera Calibration Technique for High-accuracy 3D Machine Vision Metrology Using off the Shelf TV Cameras and Lenses*. IEEE Journal of Robotics and Automation, Vol. 3, no. 4, 1987, pp. 323-344.
- [15] J. A. Villalobos. *Diseño y Manejo Estructuras de Datos en C*. Colombia, 1996.