

Control servo-visual aplicado a un robot Seekur en un ambiente sin obstáculos: Simulación y Experimentación

Armando Sáenz, Víctor Santibáñez, Javier Ollervides, Alejandro Dzul
Instituto Tecnológico de la Laguna

Blvd. Revolución y Calzada Cuauhtémoc, Apdo. Postal 49, Adm. 1, Torreón, Coah., 27001, México
jsaenz@itlalaguna.edu.mx, vsantiba@itlalaguna.edu.mx

Tel: +52 (871) 705 13 31 ext 125 Fax: +52 (871) 705 13 26

Resumen—Este artículo se presenta un control para un robot móvil omnidireccional no holonómico Seekur. El control se realiza por medio de visión empleando la técnica de control “servo-visual” basado en imagen con una ley de control PD exponencial. Este trabajo muestra el seguimiento de un objeto, empleando una cámara monocular montada sobre el robot y los resultados mostrados en simulación y experimentación.

Palabras clave: Robot móvil omnidireccional no holonómico, control servo-visual, control PD Exponencial, cámara monocular

I. INTRODUCCIÓN

En la actualidad una de las áreas de investigación que está llamando la atención es el área de visión. Mediante visión es posible recopilar gran información sobre un entorno, como lo son distancias, formas, colores y texturas con un solo sensor que sería una cámara. Las técnicas de visión son una buena alternativa para resolver problemas de control automático porque los dispositivos son de fácil acceso, económicos y de fácil lectura desde el punto de vista informático. Las técnicas de visión requieren de un gran costo computacional ya que se emplean cálculos con matrices de miles de elementos por lo que se requiere una unidad de procesamiento de alta velocidad.

Los trabajos realizados con “servo-visual” son muy variados (Hauck et al., 2000; Yoshida y Tsuzuki, 2006; Ferrer, 1998; Copot et al., 2010) en este artículo se presenta un control cinemático para un robot móvil Seekur cuyos trabajos previos son (Solea et al., 2010; Bogdan et al., 2011; Dumitrascu et al., 2011). El objetivo es el seguimiento de un objeto en el espacio de la imagen usando una cámara monocular montada en el robot. La técnica a utilizar es “servo-visual” basado en imagen, donde se busca la convergencia de las características de la imagen que en este caso son el centroide y el área del objeto. El control a implementar es un controlador PD exponencial.

El robot móvil Seekur es un robot móvil omnidireccional no holonómico o pseudo-omnidireccional, esta denominación es para los robots móviles de cuatro ruedas direccionables con orientación y tracción independientes. Algunos robots que entran en esta denominación es el

Care-O-bot 3 (Connette et al., 2008) y el Rob@Work 3 (Connette et al., 2012). El robot Seekur presenta una configuración cuadrangular en sus ruedas. Un robot móvil omnidireccional que posee ruedas suecas o de castor es capaz de cambiar su sentido de avance en cualquier dirección manteniendo la orientación del robot sin necesidad de reacomodar sus ruedas logrando un cambio de dirección en su desplazamiento logrando este cambio en un tiempo cero. Los robots omnidireccionales no holonómicos también pueden cambiar el sentido de avance en cualquier dirección sin modificar la orientación del robot pero es necesario cambiar la orientación de las ruedas, al hacer esta reorientación de ruedas es necesario esperar a que el robot tome una nueva postura lo que le impide hacer el cambio de movimiento instantáneamente. Los robots móviles pseudo-omnidireccionales presentan ciertas ventajas sobre los robots móviles omnidireccionales y son porque el modelo cinemático es más simple y en todo momento se conoce el punto de contacto de la rueda con el piso. En la sección II se presenta el modelo cinemático del robot, en la sección III se presenta la matriz de interacción de la imagen, el procesamiento de imagen utilizado (segmentación e interpretación) está descrito en la sección IV, el control se encuentra en la sección V, la simulación realizada con el paquete SIMULINK se encuentra en VI, el marco experimental y resultados son la sección VII y las conclusiones son la sección VIII.

II. MODELO CINEMÁTICO DEL ROBOT

Actualmente el robot Seekur presenta de fábrica un control omnidireccional el cual recibe como señales de entrada velocidades deseadas y este calcula los ángulos de orientación de cada una de las ruedas, así como la velocidad de cada una de ellas. Este control se encarga de ingresar los torques necesarios para orientar cada una de las ruedas hacia un ICR (por sus siglas en inglés Centro de Rotación Instantáneo) como se muestra en la Figura 1 y hacerlas girar a la velocidad necesario de acuerdo a las consignas ingresadas. Cada rueda es de tracción independiente y el giro de orientación es de 360° , al no tener restricciones en

el mecanismo de orientación el ICR puede ser colocado en cualquier parte del plano. Esto le permite al robot tener un movimiento omnidireccional.

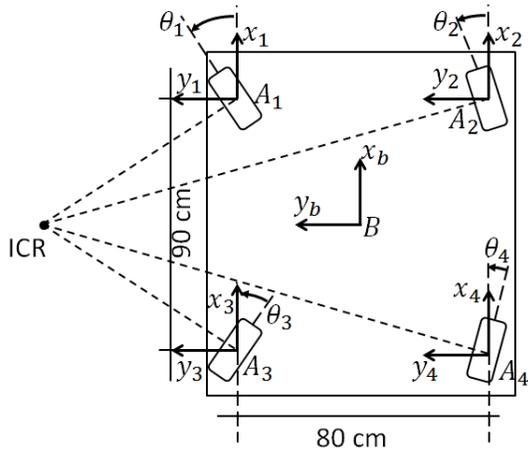


Figura 1. Orientación de las ruedas respecto a un ICR

La posición del robot se va a describir con el vector

$$\xi = [x \quad y \quad \theta]^T, \quad (1)$$

donde x y y son las coordenadas en el plano y θ es la orientación, siendo estas variables medidas respecto al marco inercial G como se muestra en la figura 2.

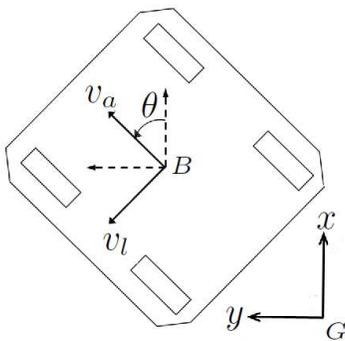


Figura 2. Diagrama de cuerpo libre del robot

Las señales de entrada para el robot son v_a para la velocidad lineal de avance en [mm/s], v_l para la velocidad lineal lateral en [mm/s] y ω para la velocidad rotacional en [grados/s] y están respecto al marco móvil del robot B . Estas velocidades se muestran en la Figura 3.

En la referencia (Canudas *et al.*, 1996) indica que el modelo cinemático de un robot con movimiento omnidireccional es

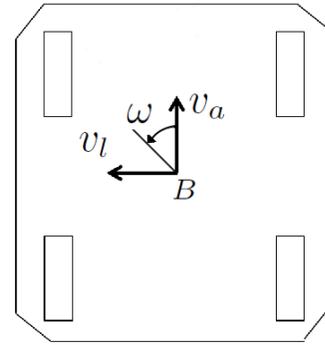


Figura 3. Velocidades que puede tener Robot Seekur

$$\begin{aligned} \dot{\xi} &= \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = R_b^g \xi_b \\ &= \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_a \\ v_l \\ \omega \end{bmatrix} \end{aligned} \quad (2)$$

Las

siendo la matriz que multiplica a ξ_b la matriz de rotación que indica la orientación del marco B respecto al marco G .

II-A. Velocidades en la cámara

Para realizar la simulación es necesario utilizar la matriz de interacción de la imagen, que se muestra en la Sección III, la cual es multiplicada por las velocidades existentes en la cámara medidas desde el marco de la cámara al cual se referirá como marco C (Véase Figura 4).

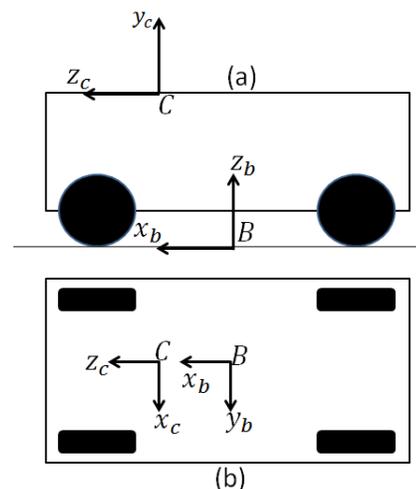


Figura 4. Marco de la cámara. (a) Vista lateral, (b) Vista superior

De la imagen anterior se puede deducir que la matriz de

rotación del marco C respecto al marco B es

$$R_c^b = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad (3)$$

al realizar el producto $R_b^g R_c^b$ se obtiene R_c^g la cual se define como

$$R_c^g = \begin{bmatrix} -\sin \theta & 0 & \cos \theta \\ \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \end{bmatrix}, \quad (4)$$

tomando en cuenta la propiedad $(R_c^g)^{-1} = R_g^c = (R_c^g)^T$ se obtiene

$$R_g^c = \begin{bmatrix} -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \\ \cos \theta & \sin \theta & 0 \end{bmatrix} \quad (5)$$

que es la matriz rotación del marco G respecto al marco C . Para calcular las velocidades a las que esta sometida la cámara vistas desde el marco C se realiza el siguiente producto

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{z}_c \\ \dot{\omega}_x c \\ \dot{\omega}_y c \\ \dot{\omega}_z c \end{bmatrix} = \begin{bmatrix} R_g^c & 0 \\ 0 & R_g^c \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} \quad (6)$$

siendo z_c , ω_x y ω_y igual a cero, y $\omega_z = \dot{\theta}$ se obtiene

$$\dot{\xi}_c \xi = \begin{bmatrix} \dot{x}_c \\ \dot{z}_c \\ \dot{\theta}_c \end{bmatrix} = \begin{bmatrix} \dot{y} \cos(\theta) - \dot{x} \sin(\theta) \\ \dot{x} \cos(\theta) + \dot{y} \sin(\theta) \\ \dot{\theta} \end{bmatrix} \quad (7)$$

donde \dot{x}_c es la velocidad lineal en dirección del eje x_c , \dot{z}_c es la velocidad lineal en dirección del eje z_c y $\dot{\theta}_c$ es la velocidad rotacional vista desde el eje y_c .

III. MATRIZ DE INTERACCIÓN

La matriz de interacción de una imagen permite conocer las velocidades de las características de un objeto 3D en un plano definido por el sensor de la cámara. La matriz de interacción a utilizar parte de un modelo de cámara pinhole, es un modelo lineal. La demostración de la matriz puede consultarse en (Siciliano *et al.*, 2009; Pomares *et al.*, 2002). La matriz pertenece al espacio $\mathbb{R}^{2m \times 6}$ donde m es el número de puntos a seguir. La ecuación para un solo punto es la siguiente

$$L = \begin{bmatrix} -\frac{f}{z_c} & 0 & \frac{x^s}{z_c} & \frac{x^s y^s}{f} & -\frac{f^2 + x^s}{f} & y^s \\ 0 & -\frac{f}{z_c} & \frac{y^s}{z_c} & \frac{f^2 + y^s}{f} & -\frac{x^s y^s}{f} & -x^s \end{bmatrix} \quad (8)$$

donde f es la distancia focal, z_c es la distancia entre la cámara y el objeto, x^s y y^s es la coordenada de la proyección del punto en el sensor de la cámara. Los renglones de

la matriz L deben de repetirse como tantos puntos se tengan. Para conocer el movimiento de estos puntos proyectados en el sensor se aplica la siguiente ecuación

$$\dot{s} = \begin{bmatrix} \dot{x}^s \\ \dot{y}^s \end{bmatrix} = L T^c \quad (9)$$

donde \dot{s} son las velocidades de los puntos en $\frac{m}{s}$, T^c es el vector columna de velocidades $[{}^c V \quad {}^c \Omega]^T$. Realizando las multiplicaciones se obtiene.

$$\dot{x}^s = \Delta - \frac{\dot{\theta}(f^2 + (x^s)^2)}{f} \quad (10)$$

$$\dot{y}^s = \frac{y^s}{z_c} (\dot{x} \cos \theta + \dot{y} \sin \theta) - \dot{\theta} \frac{x^s y^s}{f} \quad (11)$$

$$\Delta = \cos \theta \left(\frac{x^s \dot{x} - f \dot{y}}{z_c} \right) + \sin \theta \left(\frac{x^s \dot{y} + f \dot{x}}{z_c} \right).$$

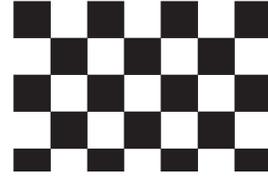


Figura 5. Diagrama de cuerpo libre del robot

Las unidades de salida son en $\left[\frac{m}{s}\right]$ pero las medidas que utiliza una imagen es en pixeles, así que requiere una conversión. Un pixel no es una medida definida en el mundo real, una misma imagen puede cambiar su resolución en pixeles y seguir midiendo lo mismo. Como se está tomando un modelo lineal y también se conoce la resolución de la cámara RVision PTZ(Pan Tilt Zoom) que es de 620 x 480 pixeles, se tomaron fotos del patron que se emplea para calibrar imagenes que se muestra en la Figura 5 del cual se conocen sus dimensiones y aplicando las siguientes igualdades

$$x^s = \frac{f}{z_c} x_o \quad (12)$$

$$y^s = \frac{f}{z_c} y_o \quad (13)$$

donde x_o y y_o son las coordenadas del objeto con respecto al marco del mundo. Estas igualdades fueron obtenidas de (Siciliano *et al.*, 2009). Para conocer la proyección del punto en el sensor de la cámara en *pixeles* se obtiene un modelo lineal

$$c_x = \frac{z_c}{x_o f} x_p \quad (14)$$

$$c_y = \frac{z_c}{y_o f} y_p \quad (15)$$

donde c_x es la unidad para convertir de *metros* a *pixeles* en el eje de las abscisas, c_y es la unidad para convertir de

metros a *pixeles* en el eje de las ordenadas y x_p, y_p son las coordenadas en pixeles. Los valores para $c_x = 21,886$ y para $c_y = 21,111$. Una vez obtenidos estos valores unicamente es necesario realizar la multiplicación de los vectores $[x^s \ y^s][c_x \ c_y]^T$ para conocer el movimiento de la imagen en pixeles.

Tomando en cuenta las siguientes transformaciones

$$\begin{aligned} x_1 &= c_x x_s \\ x_2 &= -c_y y_s \\ x_3 &= \theta \\ u_1 &= V_a \cos x_3 - V_l \sin x_3 \\ u_2 &= V_a \sin x_3 + V_l \cos x_3 \\ u_3 &= \omega \end{aligned} \quad (16)$$

donde $x_1 = x_p, x_2 = y_p$ el modelo cinemático en espacio de estados se escribe

$$\dot{x}_1 = c_x \Delta + c_x \frac{u_3(f^2 + (x_1)^2)}{f} \quad (17)$$

$$\begin{aligned} \dot{x}_2 &= -c_y \frac{x_2}{z_c} (u_1 \cos x_3 + u_2 \sin x_3) \\ &\quad + c_y u_3 \frac{x_1 x_2}{f} \end{aligned} \quad (18)$$

$$\dot{x}_3 = u_3 \quad (19)$$

$$\Delta = \cos x_3 \left(\frac{x_1 u_1 - f u_2}{z_c} \right) + \sin x_3 \left(\frac{x_1 u_2 + f u_1}{z_c} \right).$$

Siendo este modelo cinemático el usado tanto en simulación como en experimentación del controlador cinemático propuesto.

IV. PROCESAMIENTO DE IMAGENES

La técnica de control a implementar es conocida o nombrada en la literatura como control “servo-visual” basado en imagen. El objetivo de esta técnica es la convergencia de las características obtenidas de la imagen. Las características a tomar son el centroide y el área. El objeto a seguir es una esfera, esto porque la esfera es completamente simétrica así que no importa la posición desde donde sea observada, sus características son las mismas. En la referencia (Siciliano *et al.*, 2009) menciona que el tratamiento de la imagen previo a la realimentación consta de dos etapas segmentación e interpretación. Para realizar el procesamiento de la imagen se va a hacer uso del software OpenCV.



Figura 6. Resultados de la segmentación

Para eliminar la información contenida en la imagen y no es necesaria existen numerosos métodos, en este caso

se utiliza un filtro de colores HSV (Hue Saturation Value - Matiz Saturación Valor). Los resultados del filtrado se muestran en la Figura 6. Después del filtrado se calcula la posición del centroide y el tamaño del área lo cual se realiza obteniendo los momentos de la imagen.

V. CONTROL

El lazo de realimentación se muestra en la siguiente Figura 7.

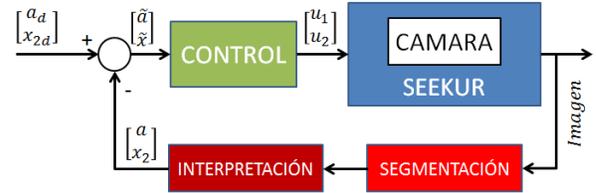


Figura 7. Diagrama a bloques del sistema

La ley de control no lineal a utilizar es de tipo PD exponencial. En esta ley de control la acción tanto proporcional como derivativa son acotadas por una constante P_M para la acción proporcional y una constante D_M para la acción derivativa. Ambas constantes (P_M y P_D) son valores de los cuales no va a pasar la señal de entrada y siendo la suma de estas dos ($P_M + P_D$) la máxima señal de entrada al robot. La expresión a utilizar es la siguiente

$$\begin{aligned} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} &= \begin{bmatrix} P_{M1} & 0 \\ 0 & P_{M2} \end{bmatrix} \begin{bmatrix} \text{Sign}(\tilde{a})(1 - e^{-k_{p1}|\tilde{a}|}) \\ \text{Sign}(\tilde{x})(1 - e^{-k_{p2}|\tilde{x}|}) \end{bmatrix} \\ &\quad + \begin{bmatrix} 0 \\ P_D \text{Sign}(-\dot{x}_2)(1 - e^{-k_{p2}|\dot{x}_2|}) \end{bmatrix} \\ \tilde{a} &= a_d - a \\ \tilde{x} &= x_{2d} - x_2 \end{aligned} \quad (20)$$

donde \tilde{x} es el error proporcional en $[pixel]$, $\dot{\tilde{x}}$ es el error derivativo en $[\frac{pixel}{segundo}]$, a es el área obtenida por en OpenCV y la cual puede calcularse con la ecuación 21, k_p es la ganancia proporcional en $[\frac{metros}{pixelsegundo}]$ y k_d es la ganancia derivativa en $[\frac{metros}{pixel}]$. El perfil de las leyes de control se muestra en las Figuras 8 y 9.

$$a = \frac{(\pi r^2)}{z_c} c_x \quad (21)$$

donde r es el radio del objeto. Se emplea la ecuación 21 porque se usó un objeto de forma esférica como se menciona en la sección IV.

VI. SIMULACIONES

Las simulaciones se realizaron en SIMULINK. La señal de entrada que se escogió es una onda senoidal de una amplitud de 100 pixeles y una frecuencia de 0.1 Hz. Se tomó en cuenta 1 punto característico correspondiente a la coordenada del centroide en el eje de las abscisas. El tiempo

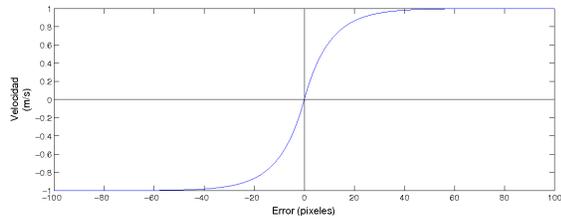


Figura 8. Respuesta de la ecuación 20 con un $P_{M2} = 1$ y un $k_{p2} = 0.1$

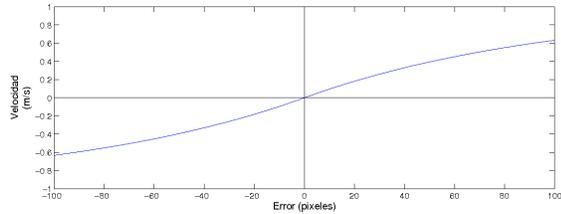


Figura 9. Respuesta de la ecuación 20 con un $P_{M2} = 1$ y un $k_{p2} = 0.01$

de simulación es de 50 segundos. El objeto a seguir se sitúa a una distancia de 6 metros y el centroide comienza en las coordenadas $[x_s \ y_s] = [200 \ 0]$ *pixeles*.

No se controla la velocidad rotacional porque no se requiere para seguir un objeto por lo que $u_3 = 0$. Los resultados de la simulación se muestran en las Figuras 10, 11 y 12. En la figura 10 se aprecia la señal de entrada (línea continua) y la señal de salida (línea punteada). En la figura 11 se presenta un error el cual va disminuyendo conforme avanza el tiempo. Por último en la figura 12 se observa la gráfica de las velocidades del carro en donde la velocidad máxima se tiene al inicio a $40 \left[\frac{cm}{s} \right]$ (La resolución del carro permite ingresar velocidades de $1 \left[\frac{mm}{s} \right]$).

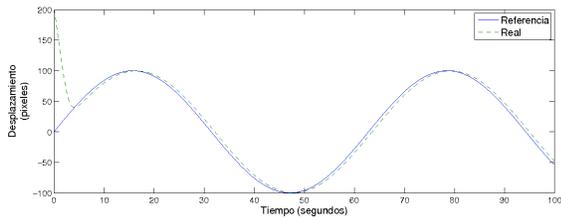


Figura 10. Gráfica de referencia (línea seguida) y salida (línea punteada)

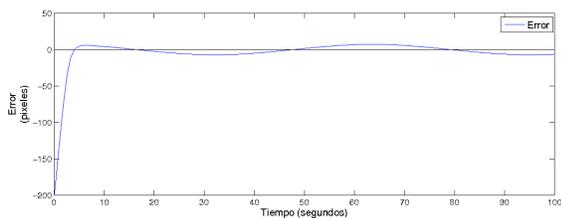


Figura 11. Gráfica del error medida en pixeles

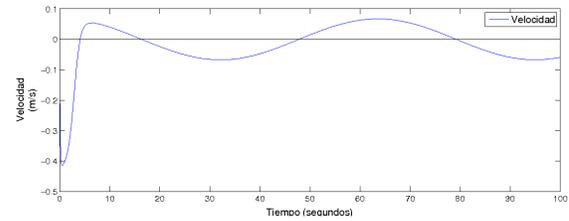


Figura 12. Velocidad lateral del carro ($[m/s]$)

VII. MARCO EXPERIMENTAL Y RESULTADOS

Para esta investigación se utilizó un robot móvil seekur para exteriores (véase la Figura 13). Este robot es fabricado por la empresa Mobile Robots el cual tiene el propósito de ser utilizado para trabajos de investigación. Es un robot móvil omnidireccional de 4 ruedas, cada una de las ruedas tiene tracción y orientación independiente. Cuenta con 2 modos de operación omnidireccional y diferencial. La forma de comunicación del robot es por medio de un protocolo de comunicación Wifi. Los accesorios con los que cuenta el robot son LRF (Laser Range Finder) de una distancia de 10 [metros] y 270 grados, un GPS marca Trimble con 2 metros de error, una cámara monocular PTZ (Pan-Tilt Zoom) marca RVision la cual tiene conectada un “Frame Grabber Sensoray” modelo 2253 para digitalizar las imágenes, una cámara estero MobileRanger C3D de la empresa Focus Robotics, una PC a bordo para cargar rutinas al robot o bien de servidor para una comunicación remota y por último un centro de acelerómetros, codificadores ópticos incrementables y un giroscopio de eje único para realizar la estimación de desplazamiento por el método de odometría.



Figura 13. La fotografía del Robot Móvil Seekur

Los experimentos se realizaron dentro del instituto en un lugar abierto con suelo firme y sin obstáculos dentro del área de movimiento del robot. El objeto a seguir se muestra en la Figura 14 del cual primero se toma una foto para determinar los valores HSV para la segmentación e interpretación. El círculo blanco que se observa en la Figura 14 es introducido para identificar el objeto que se está reconociendo.

Se utilizó el programa “Aria” que es de la empresa “MobileRobots” para programar el controlador. Este programa



Figura 14. Objeto utilizado para el experimento

contiene librerías para el lenguaje de programación C/C++ que permite establecer la comunicación con el robot de manera local o remota. Para la obtención de imágenes se emplearon las librerías del V4L2 y el tratamiento se realizó con las librerías de OpenCV 3.2.1. Los resultados se muestran en las Figuras 15 y 16. El experimento consiste en desplazar el carro lo necesario para que el centroide del objeto observado coincida con el eje central de la imagen, centrada la imagen el objeto es desplazado esperando de reubicación Figura 15 porque son con razón de 2 o 3 pixeles, las cuales tienen una repercusión en la señal de control que se ingresa al robot. El tiempo del experimento fue superior a los tres minutos.

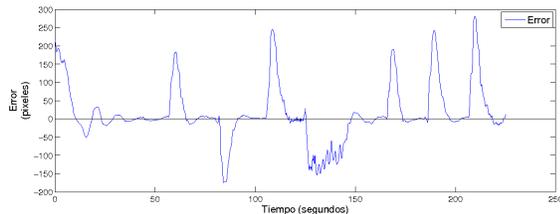


Figura 15. Error obtenido en el experimento

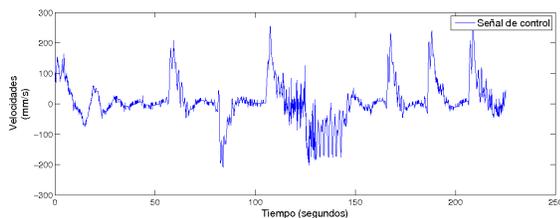


Figura 16. Velocidad lateral del robot durante el experimento

VIII. CONCLUSIONES.

En resumen se presenta un control de seguimiento para un robot móvil omnidireccional. La técnica a utilizar es control “servo-visual” basado en imagen, donde la coordenada en eje de las abscisas se utiliza para el movimiento lateral. Los resultados presentados aunque razonables, todavía pueden presentar mejoría agregando una acción integral. Otro punto a considerar para mejorar el seguimiento es la estimación de la distancia a la que se encuentra el objeto, ya que entre mas alejado se encuentre el objeto del robot, el movimiento

lateral presentará cambios más pequeños debido a que la proyección en el sensor de la cámara tiene un movimiento más lento. Una ventaja que tiene el robot es el hecho de no presentar restricciones no holonómicas ya que la trayectoria deseada por más complicado que sea puede ser realizada, aunque en el presente trabajo únicamente se toma en cuenta un grado de libertad del robot, por lo que se planea usar el el avance y orientación del robot para realizar evasión de obstáculos.

IX. AGRADECIMIENTOS.

Este trabajo ha sido parcialmente apoyado por los proyectos CONACYT 134534 y DGEST.

REFERENCIAS

- Bogdan, Dumitrascu, Filipescu Adrian, Minzu Viorel, Voda Alina y Minca Eugenia (2011). Discrete-Time Sliding-Mode Control of Four Driving-Steering Wheels Autonomous Vehicle. En: *30th Chinese Control Conference*. Yantai, China. pp. 3620–3625.
- Canudas, Carlos, Bruno Siciliano y Georges Bastin (1996). *Theory of Robot Control*. Springer.
- Connette, Christian, Martin Hagele y Alexander Verl (2012). Singularity-free state-space representation for non-holonomic, omnidirectional undercarriages by means of coordinate switching. En: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Ieee. Vilamoura, Algarve, Portugal. pp. 4959–4965.
- Connette, Christian P., Andreas Pott, Martin Hagele y Alexander Verl (2008). Control of an pseudo-omnidirectional, non-holonomic, mobile robot based on an ICM representation in spherical coordinates. En: *2008 47th IEEE Conference on Decision and Control*. Ieee. Cancún, México. pp. 4976–4983.
- Copot, Cosmin, Adrian Burlacu y Corneliu Lazar (2010). Visual control architecture of servoing systems based on image moments. En: *2010 12th International Conference on Optimization of Electrical and Electronic Equipment*. Ieee. pp. 801–806.
- Dumitrascu, Bogdan, Adrian Filipescu, Cristian Vasilache, Eugenia Minca y Adriana Filipescu Jr (2011). Discrete-Time Sliding-Mode Control of Four Driving / Steering Wheels Mobile Platform. En: *19th Mediterranean Conference on Control and Automation*. Corfu, Greece. pp. 1076–1081.
- Ferrier, N J (1998). Performance of Visual Tracking Systems : Implications for Visual Controlled Motion. En: *IEEE Conference on Decision & Control*. número December. Tampa, Florida. pp. 3725–3730.
- Hauck, Alexa, Georg Passigt, Thomas Schenk, Michael Sorgt y Georg Farber (2000). On the Performance of a Biologically Motivated Visual Control Strategy for Robotic Hand-Eye Coordination. En: *IEEE/RSJ International Conference on Intelligent Robots and Systems*. pp. 1626–1632.
- Pomares, J, F Torres y P Gil (2002). Seguimiento de trayectorias 3d mediante control visual basado en imagen. En: *XXIII Jornadas de Automática*.
- Siciliano, Bruno, Lorenzo Sciavicco, Luigi Villani y Giuseppe Oriolo (2009). *Robotics Modelling, Planning and Control*. Springer.
- Solea, R, A Filipescu y D Cernega (2010). Lateral Motion Control of Four-Wheels Steering Vehicle Using a Sliding-Mode Controller. En: *29th Chinese Control Conference*. Beijing, China. pp. 3699–3703.
- Yoshida, Y. y K. Tsuzuki (2006). Visual tracking and control of a moving overhead crane load. En: *9th IEEE International Workshop on Advanced Motion Control, 2006.*. Ieee. pp. 630–635.