

Comparativa de Controladores P, PI, PID para la Ubicación y Seguimiento de Objetos en la Plataforma Robotino®View

H. Méndez, J. Jalomo, F. Ordoñez, J. García
Instituto Tecnológico de Ciudad Guzmán
Zapotlán el Grande, Jalisco, México
hmendez97@hotmail.com

Resumen—Se describe el comportamiento de controles P, PI y PID para el seguimiento de objetos por un robot móvil. Se asigna la tarea a la plataforma Robotino de buscar objetos de colorimetría conocida, a través del giro en su propio eje y una vez localizado, su acercamiento controlando el enfoque del objeto en el centro de sensor de visión y la aproximación al objeto por medio de un sensor reflectivo. La programación del robot se realiza bajo el uso del software Robotino View de FESTO.

Palabras clave: Robótica móvil, control, visión.

I. INTRODUCCIÓN

Un robot móvil es una máquina que integra componentes mecánicos, eléctricos, comunicación y control que bajo un ambiente de programación es capaz de percibir su entorno, navegar automáticamente, así como la realización de tareas asignadas por el hombre.

Existen en la actualidad innumerables aplicaciones que se han desarrollado en robótica móvil con este fin, aplicaciones como transporte de materiales peligrosos en experimentos bioquímicos, carga de objetos pesados a edificios en construcción, navegación y autoubicación en escenarios desconocidos, hasta la exploración espacial de robots móviles teleoperados. La reciente tendencia a la automatización y control en Robótica hace necesario el uso de robots inteligentes, donde la visión y el control son herramientas fundamentales para su interacción con el medio y la realización de tareas programadas; con este fin se propone la puesta en marcha de un robot móvil de laboratorio de la marca FESTO en el Instituto Tecnológico de Cd. Guzmán con el objetivo de proveer de información al estudiante de forma que se involucren con la programación, asignación de tareas e investigación aplicada sobre robots móviles.

El presente artículo presenta una propuesta de control clásico para la ubicación y seguimiento de objetos diseñado con la instrumentación básica proporcionada por el software Robotino View de FESTO.

II. DESCRIPCIÓN DEL SISTEMA

Robotino es un robot móvil desarrollado por FESTO para fines educativos, entre las principales ventajas que ofrece es movimiento omnidireccional apoyado con sensores de proximidad, posición, colisión, así como una webcam y un

protocolo de comunicación LAN para controlar la navegación del robot de forma remota.

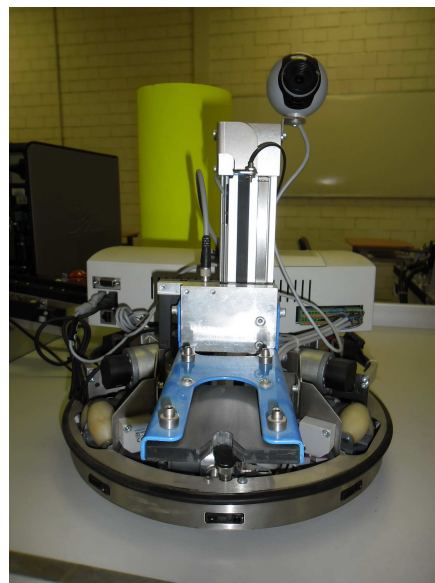


Figura 1. Plataforma Robotino de FESTO.

Robotino además de ofrecer una completa gama de sensores para navegación, también posee un sistema de control interno para velocidad, indicadores de aceleración, posición y corriente en sus 3 motores.

A. SENSORES DE DISTANCIA REFLECTIVO

Este robot móvil cuenta con nueve sensores reflectivos alrededor de él y separados uno del otro 40° (figura 2). El sensor reflectivo SHARP GP2D120 provee como salida un voltaje dependiendo de la distancia medida.

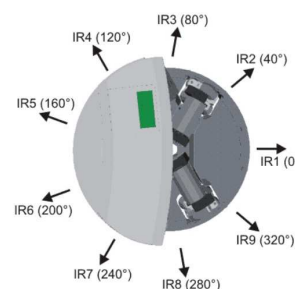


Figura 2. Sensores de Reflexión.

En la figura 3 se muestra la respuesta de este sensor con respecto a la distancia de un objeto, notando que no es recomendable su uso para distancias menores a los 4 cm (GP2D120 Datasheet).

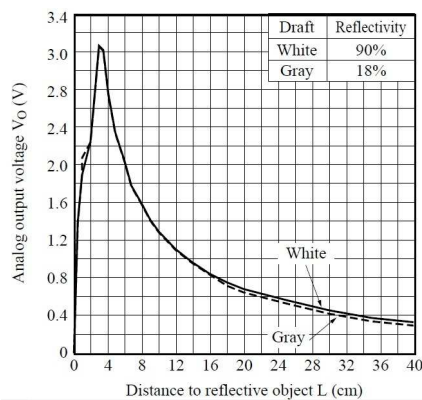


Figura 3. Respuesta del sensor SHARP GP2D120

B. ORIENTACIÓN Y UBICACIÓN DE MOTORES

El movimiento de Robotino se basa en un eje coordenado 2-D y su rotación se da con respecto a Omega (figura 4), y su movimiento lo podemos controlar por medio del bloque de omnidireccionalidad que proporciona la plataforma.

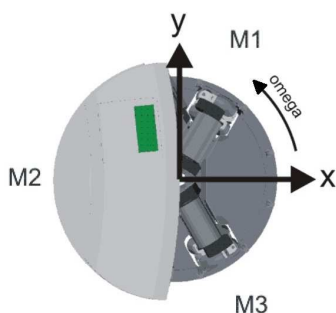


Figura 4. Orientación del Robotino.

C. SEGMENTACIÓN POR COLORIMETRÍA Y UBICACIÓN DE OBJETOS

Esta plataforma usa como sensor de imagen una webcam convencional y la ubicación de objetos se realiza por medio de un bloque Segmentador y un Extractor de Segmentos. El Segmentador nos permite almacenar el rango de colores R, G y B máximo y mínimo en el cual se encuentra uno o varios objetos, tal que detecta y provee el centroide del objeto que se le programe (figura 5). El extractor de segmentos nos permite la selección individual de cada uno de los objetos almacenados por el segmentador, arrojando como salida la imagen del objeto detectado, su centroide y un indicador de presencia del objeto en pantalla.

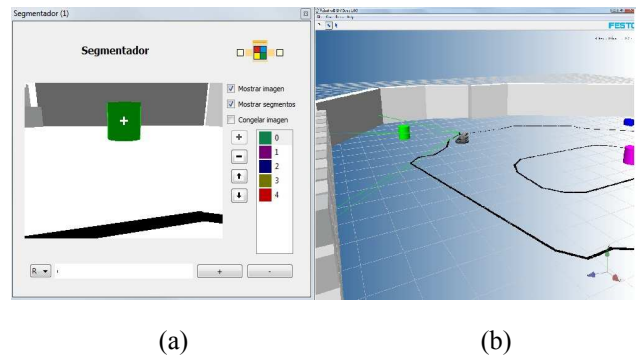


Figura 5. Robotino® SIM, (a) Segmentación y ubicación de objetos, (b) Ambiente de simulación.

III. DISEÑO Y LÓGICA DE FUNCIONAMIENTO EN BASE A LA LOCALIZACIÓN DE OBJETOS.

La plataforma de FESTO permite la programación del robot con un lenguaje gráfico controlado por diagrama de flujo, con la opción de modificar libremente la secuencia lógica de un programa. En seguida se describe la lógica de operación usando control PID para la localización y seguimiento de objetos.

A. FLUJO DEL PROGRAMA Y LÓGICA DE CONTROL.

El programa asignado al Robotino además de la inicialización de variables consiste en tres pasos, *Girar*, *Acercar* y *SiguienteC*. Iniciar establece los valores iniciales de la búsqueda (figura 6).

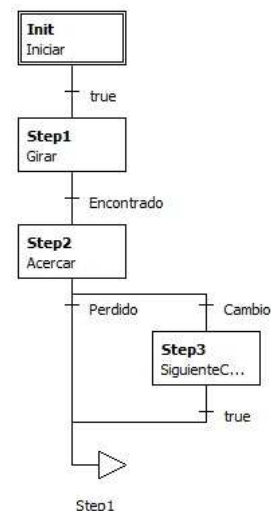


Figura 6. Flujo del Programa.

Girar pasa el control de Omega a un controlador proporcional, tal que el Robotino gire sobre su eje hasta localizar el primer objeto, una vez que se ha localizado pasa a la siguiente etapa.

Acercar establece un modo de control para el acercamiento hacia el objeto, un control PID para mantener el objeto enfocado en la mira del Robotino y un control

Proporcional que permite el avance hacia este. El controlador PID controla el eje de giro del Robotino (omega) permitiendo que conforme avance el robot no pierda su objetivo de vista. Este controlador fue implementado con las herramientas básicas que provee el software de FESTO (figura 7) y conforme a la ecuación de diferencias de un PID digital mostrada en la ec. (1) (Ogata, 1996), tal que se encuentra conformado con los elementos más básicos como registros de corrimiento, de muestreo y retención, sumadores, multiplicadores, etc.

$$m(k) = (K_p + K_i + K_d)e(k) + (-K_p - 2K_d)e(k-1) + (K_d)e(k-2) + m(k-1) \quad \text{Ec. (1)}$$

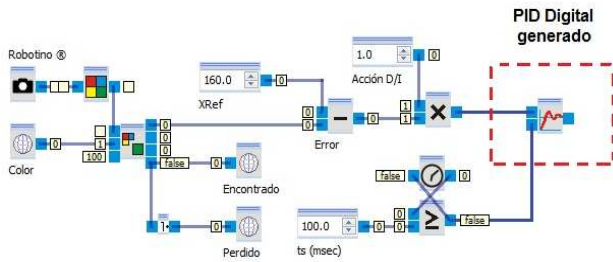


Figura 7. Representación del control PID digital construido bajo las herramientas de software de FESTO.

El avance hacia el objeto es controlado proporcionalmente (figura 8) y conforme a la medición de un sensor reflectivo que se encuentra a 0° (IR1), tomando control sobre el eje X del Robotino.

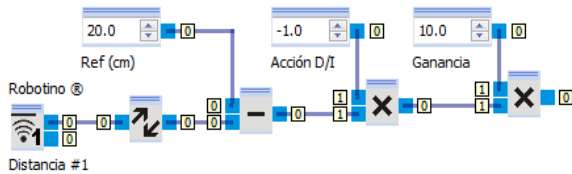


Figura 8. Control Proporcional con respecto a la distancia de un objeto.

Una vez que el Robotino se ha acercado a una distancia establecida, se inicia un conteo de 5 segundos para cambiar de objetivo y reiniciar la secuencia.

IV. RESULTADOS

Una vez construido el controlador digital, se establecieron pruebas de control partiendo del Proporcional con $k_p=1$ para establecer un marco de referencia, posteriormente se hicieron cambios en K_i y K_d , tal que para el PI se establecieron $K_p=1$, $K_i=0.1$ y para el PID $K_p=1$, $K_i=0.15$ y $K_d=0.01$.

Los parámetros del controlador PID fueron obtenidos siguiendo el criterio de Ziegler-Nichols (Ogata, 1998), sin embargo se obtuvieron oscilaciones indeseables en la estabilización del sistema, por lo que se actuó de manera intuitiva para el mejoramiento de la respuesta.

En la figura 9 se muestra una comparativa entre el control P y PI para seguimiento de objetos dada una referencia

establecida por un punto en el eje X de la pantalla con una resolución de 320x240.

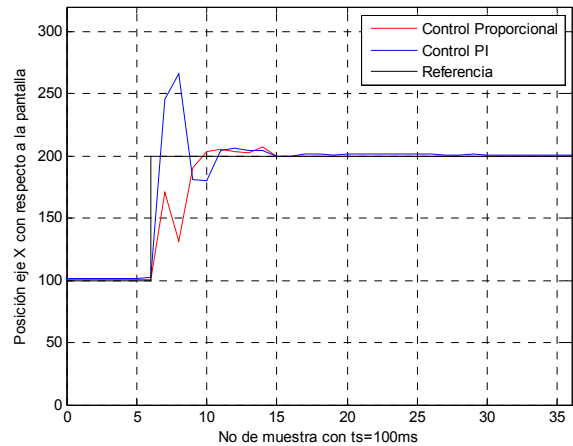


Figura 9. Seguimiento de la referencia en el eje X con control P y PI.

Por otra parte, la figura 10 muestra el desempeño de un controlador PID en comparativa con el PI, apreciándose una oscilación permanente del Robotino tratando de enfocar el objeto en su referencia y mostrando como con los parámetros inadecuados, el sistema pierde su objeto de vista.

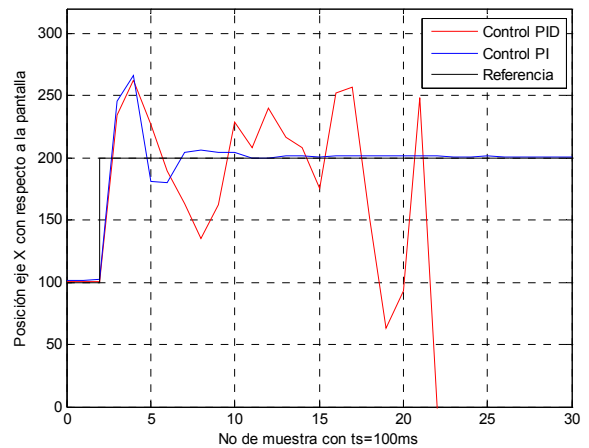


Figura 10. Seguimiento de la referencia en el eje X con control PI y PID.

Cabe mencionar que la inclusión del control derivativo puede producir una perturbación adicional al sistema si el cambio en el error es frecuente, debido a que este reacciona de manera proporcional a la rapidez de cambio del error (Bolton, 2010). Además se tiene que tomar en cuenta que el segmentador dado por la plataforma Robotino procesa la imagen en el espacio de color RGB, siendo constante el cambio de tono en la ubicación de un objeto conforme el robot avanza y la iluminación cambia, produciendo una distorsión geométrica en el objeto a detectar (Russ, 2007). Estos cambios producen una oscilación constante en la ubicación del objeto, teniendo un error variante y afectando en la estabilidad del sistema.

En la figura 11 se muestra la evolución de la ubicación y seguimiento de un objeto bajo el diagrama de flujo propuesto en la sección III y demostrando que a pesar de mover la referencia donde se quiere enfocar el objeto, el control permite la ubicación rápida del objeto.

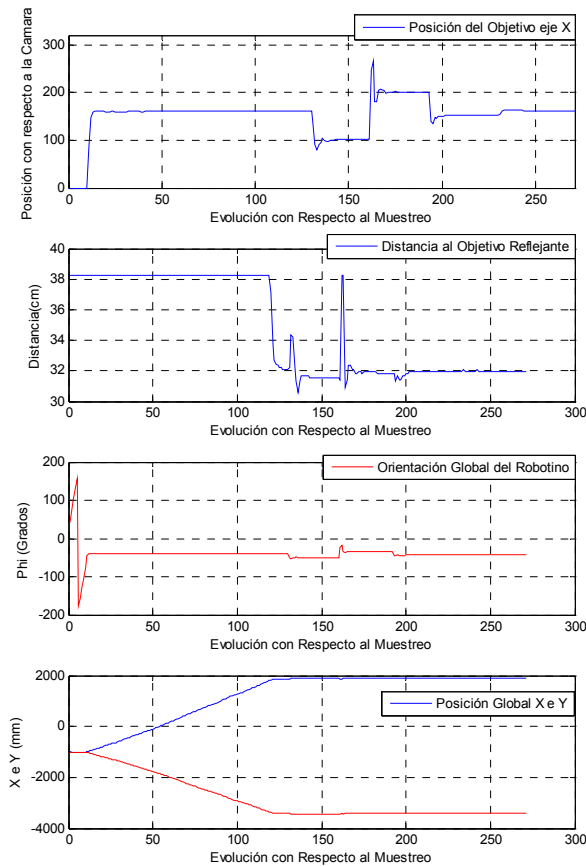


Figura 11. Ejemplo de búsqueda y seguimiento de objetos por control PI.

El sistema propuesto en este trabajo también fue probado en la plataforma para Robotino instalada en el Instituto Tecnológico de Cd. Guzmán mostrando buena respuesta en el seguimiento continuo de objetos (figura 12).

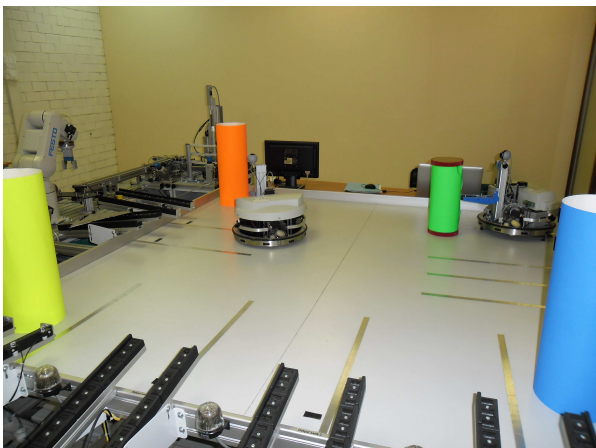


Figura 12. Puesta en marcha del sistema en Instituto Tecnológico de Cd. Guzmán.

V. CONCLUSIONES

Tras la realización de este trabajo se aprecia la importancia del procesamiento digital de imagen para la ubicación de objetos, ya que uno de los principales inconvenientes que se presentaron en la implementación física fue el cambio constante de luminancia en el objeto conforme el movimiento del Robotino. Además si el control no está optimizado la oscilación producida por el control en conjunto con los errores de la captación del objeto lleva rápidamente a inestabilidad al sistema.

REFERENCIAS

- Bolton, W. (2010). *Mecatrónica – Sistemas de Control Electrónico en Ingeniería Mecánica y Eléctrica*. Alfaomega, México.
- Ogata, K. (1996). *Sistemas de Control en Tiempo Discreto*. Prentice Hall, Edo. de México, México.
- Ogata, K. (1998). *Ingeniería de control moderna*. Prentice Hall, Edo. de México, México.
- Russ, J. (2007). *The Image Processing Handbook*. CRC Press, USA.
- FESTO. *Manual RobotinoView2*.
- SHARP. *GP2D120 Datasheet*.