

Implementación de un controlador PID al sistema de visión artificial de un brazo robótico para el seguimiento de objetos de color

Implementation of a PID controller to the machine vision system of a robotic arm for the color object tracking

Implementação de um controlador PID para o sistema de visão de máquina de um braço robótico para monitoramento o objeto de cor

Luis E. Beltrán Casana¹

Resumen

El objetivo del estudio fue implementar un controlador PID (proporcional, integral, derivado) al sistema de visión artificial de un brazo robótico para el seguimiento de objetos de color. Se diseñó y construyó un brazo robótico con dos servos motores en cada articulación de tipo SG 5010. El control del hardware se realizó con la placa electrónica Arduino Uno, usando el software de Matlab. Se diseñó un programa que decodifica las imágenes captadas por la cámara, el cual brinda las coordenadas del objeto a seguir; luego los datos obtenidos se procesaron mediante el controlador PID y se enviaron al Arduino, el cual envía una señal eléctrica para que el brazo pueda moverse de manera que siga el objeto indicado. La aplicación de un controlador PID a un brazo robot podría usarse en el control de robots humanoides; asimismo, el estudio permitió desarrollar algoritmos de procesamiento de imágenes flexibles.

Palabras clave: Controlador PID, Robótica, visión artificial.

Abstract

The objective of the study was to implementation a PID (proportional, integral, derivative) controller to the machine vision system of a robotic arm for the color object tracking. It was designed and built a robotic arm with two servo motors at each joint of type SG5010. Hardware control was carried out with the electronic board Arduino Uno, using Matlab software. We designed a program that decodes the images captured by the camera, which provides the coordinates of the object to follow; then the data obtained were processed using the PID controller and sent to the Arduino, which sends an electrical signal to the arm can move so you follow the given object. The application of a PID controller to a robot arm could be used in the control of humanoid robots; in addition, the study allowed to develop flexible image processing algorithms.

Keywords: PID controller, robotic, artificial vision.

Resumo

O objetivo do estudo foi aplicar um controlador de PID (proporcional, Integral, derivativo) a 2 graus de condicionamento de braço robótico de liberdade com uma web cam, para acompanhamento de objetos usando a visão artificial e controlar o seu movimento através do objeto de cor. Ele foi projetado e construído um braço robótico com dois servos motores em cada junta do tipo SG 5010. Controle de hardware foi realizada com a placa eletrônica Arduino Uno, usando o software Matlab. Desenhamos um programa que decodifica as imagens capturadas pela câmera, que fornece as coordenadas do objeto a seguir; em seguida, os dados obtidos foram processados usando o controlador PID e enviado para o Arduino, que envia que um sinal elétrico para o braço pode mover-se para que seguir determinado objeto. A aplicação de um controlador PID para um braço de robô pode ser usada no controle de robôs humanoides; além disso, o estudo permitiu para desenvolver algoritmos de processamento de imagem flexível.

Palavras-chave: PID controlador, robótica, visão.

¹Universidad Nacional de Trujillo, lbeltranc@unitru.edu.pe

Recibido, 2 de mayo de 2016
Aceptado, 8 de junio de 2016

Introducción

Actualmente, el uso de los brazos robóticos ha tenido un importante apogeo dentro de los procesos industriales a nivel nacional. Una de las funciones principales es que realizan tareas repetitivas de manera más rápida, eficiente y segura, comparada con un humano promedio. Estas acciones que realiza el brazo robot, con una programación previa, es lo que se conoce como automatización. Los brazos robóticos industriales, por ejemplo, contienen su propio programa y sistema de control, por lo que el usuario solo está limitado a indicarle al brazo la trayectoria que tiene que recorrer y que acción ejecutar al llegar al final de la trayectoria.

A fin de lograr una mayor eficiencia en el uso de los brazos robóticos es de interés desarrollar investigaciones orientadas a incorporarle visión artificial para mejorar su control y conocer su posición en cada espacio de tiempo (Cuevas y col.,2010). El sentido de la visión en los robots es un elemento primordial, ya que permite dotar de características de adaptabilidad y flexibilidad. La visión artificial nos proporciona la descripción del estado deseado así como su evolución en el tiempo; esta información es usada en el sistema de control del robot para la generación y modificación de su plan de trabajo, lo cual le permite evolucionar en ambientes variables (Aybar y col., 2010).

La visión artificial trata de simular el proceso del sentido de la visión de los seres vivos, y nos permite describir de forma automática la estructura y propiedades de un mundo tridimensional, a partir de imágenes bidimensionales (monocromáticas o a colores) del mundo. Las propiedades que se desean rescatar del mundo tridimensional pueden ser geométricas (forma, tamaño y localización de objetos) o materiales (color, textura y composición).

Uno de los métodos de control más utilizados en la industria, a fin de que el brazo robótico pueda seguir el objeto mediante la visión artificial, es el método de control de tipo PID (Kuo, 1996; Ogata, 2003). Un controlador PID es una técnica común utilizada para controlar una amplia variedad de maquinaria incluyendo vehículos, robots e incluso cohetes. El controlador PID se representa mediante la ecuación:

$$u(t) = K_p \left(e(t) + \frac{1}{\tau_i} \int_{-\infty}^t e(t) dt + \tau_d \frac{de(t)}{dt} \right)$$

En dicha ecuación permite relacionar la variable de salida (ángulo de giro del servo motor del brazo) con la variable de entrada (error) (Kuo, 1996; Ogata, 2003). Aplicando la transformada de Laplace, a la ecuación anterior se obtiene la ecuación lineal:

$$G_c(s) = K_p \left(1 + \frac{1}{\tau_i s} + \tau_d s \right)$$

El problema formulado fue: ¿De qué manera un controlador PID (Proporcional, Integral, Derivativo) aplicado a un brazo robótico de 2 grados de libertad permite el seguimiento de objetos mediante visión artificial? Los objetivos de la investigación fueron: a) Aplicar un controlador PID a un brazo robótico de 2 grados de libertad, para el seguimiento de objetos mediante visión artificial; b) Acondicionar una cámara web en el efector del brazo y controlar su movimiento mediante el “object tracking”, haciendo que la cámara siga la trayectoria de un objeto dentro de un área determinada.

Material y métodos

Diseño y construcción del brazo robótico

Se diseñó y construyó un brazo robótico de dos grados de libertad cuyo efector se mueve en el eje de las abscisas y ordenadas; esto se explica porque la imagen busca el objeto de color y el programa nos calcula las dos coordenadas X e Y de su centroide. El diseño contiene dos servos motores en cada articulación de tipo SG 5010 (figura 1A), voltaje de 5v y torque de 6Kg/cm. Los eslabones del brazo fueron de madera por ser un material liviano, de bajo costo y fácil de trabajar. La base del robot fue de plástico, la cual contiene la placa Arduino Uno (figura 1B), que controla los ángulos de giro de los ejes de los servo motores.

Al efector del brazo robótico se le acondicionó una cámara web Deluxe 866 (figura 1C), pues se requiere conocer los datos de las coordenadas en todo momento del efector para que el brazo pueda seguir el objeto de color. De esta manera habría la retro alimentación necesaria del sistema y con ello se pueda dar el control de tipo PID. La cámara Deluxe 866 ajusta su foco manualmente, posee una resolución máxima de 320x240, y su valor del umbral es constante ante la escasa o excesiva iluminación.

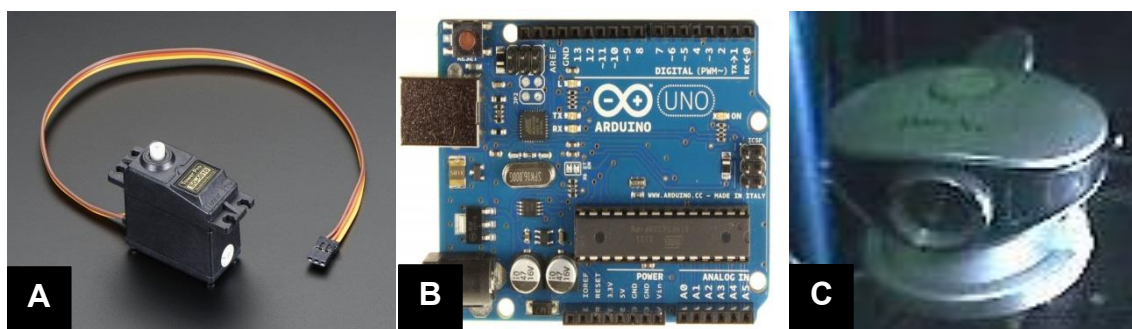


Figura 1. Materiales electrónicos usados en la construcción del brazo robótico: A) Servo motor SG 5010; B) Placa Arduino Uno; C) Cámara Deluxe 866.

El control del hardware se realizó con la placa electrónica Arduino Uno, la cual envía pulsos eléctricos para mover el brazo robot a través de los servos motores situados en las articulaciones (figura 2). Adicionalmente se usó un protoboard para el cableado entre los servos y la placa. Normalmente el Arduino tiene su lenguaje de programación donde desde allí se compila y se carga un código para que el usuario pueda realizar las acciones deseadas (Hanif y col., 2012). Sin embargo, para el estudio se realizó un previo procesamiento de imagen para obtener datos del entorno, grabándose un código de conexión entre Arduino y Matlab, a fin que este software controle el procesamiento de señal y el movimiento del brazo robótico.



Figura 2. El autor del trabajo y el sistema mecatrónico del brazo robótico a término.

Resultados

Obtención de las coordenadas de la visión artificial

En la presente investigación el color a seguir fue el rojo, debido a que es uno de los colores básicos RGB (Red, Green, Blue).

Las etapas del sistema de visión artificial para detectar un objeto de color rojo fueron:

a. Adquisición de la imagen

En esta primera etapa se obtiene una imagen por medio de la cámara, es decir transforma las señales luminosas que aparecen en una escena en señales analógicas capaces de ser transmitidas (Cazorla, 2010).

b. Pre procesamiento de la imagen

La imagen original se transformó en otra en que se han aplicado filtros para eliminar los problemas de ruido, con lo cual se mejoran las imágenes digitales obtenidas. El código del filtrado usado para eliminar fue: filtro = medfilt2(resta, [3 3]).

c. Segmentación

La imagen se descompuso en regiones y se evaluó si cada pixel de la imagen pertenece al objeto de interés; en este caso nuestro objeto es si existen pixeles de color rojo. Como resultado se obtiene una imagen binaria, donde los pixeles que pertenecen al objeto de interés se representan con '1', mientras los que no pertenecen se representan con '0'. El número 0.22 es el umbral calibrado, pues el Laboratorio de Robótica de Ingeniería Mecatrónica, donde se investigó, tiene características propias de iluminación, color de pared, textura del objeto y posición de la cámara en un lugar donde haya sombra. El código para la segmentación fue: binario= im2bw(filtro,0.22).

d. Descripción y extracción de características

Después se evaluó la característica del objeto frente a otros con los mismos segmentos. En este caso se ha tomado el área, es decir si la cámara detecta pequeños objetos de color rojo, tendrán un área menor a 300 pixeles, que es el área promedio que tiene el objeto rojo que se ha escogido para que el robot pueda realizar su seguimiento. El código usado fue: bn = bwareaopen(binario,300);

e. Reconocimiento de patrones

En este proceso se realizó la categorización de los datos de entrada en clases identificadas, en este caso se escogió calcular las coordenadas del centroide del área del objeto rojo y se guardaron en 2 variables: X e Y. Se tomó como plano de referencia al plano de la imagen obtenida por la cámara donde el punto (0,0) se ubica en la esquina superior izquierda.

f. Toma de decisiones y comunicación con el robot

Una vez reconocido el objeto presente en el campo de visión de la cámara, el sistema de visión artificial deberá brindar la información al controlador PID, y con ello a

través del algoritmo de movimiento le envía las órdenes al robot para realizar la tarea de seguimiento de objeto.

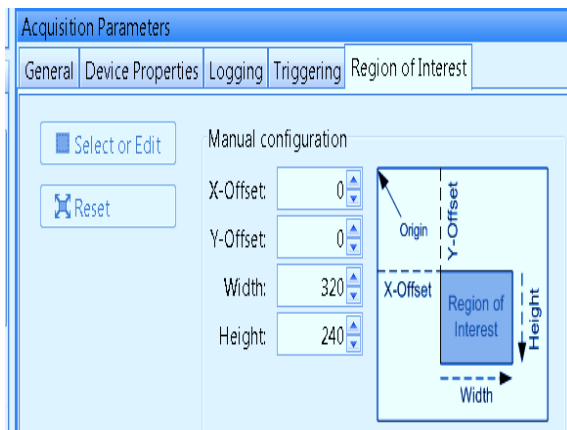


Figura 3. Recorte del imaqttool para determinar las coordenadas del centroide en la imagen

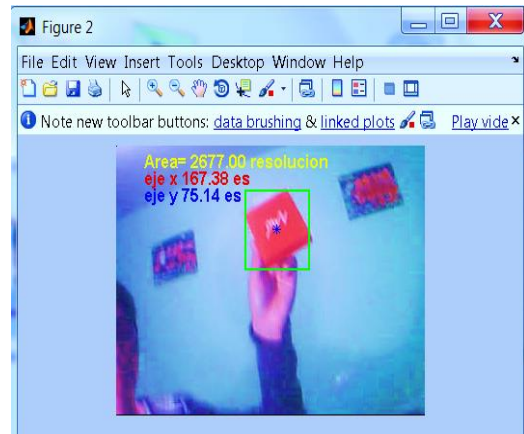


Figura 4. Obtención de las coordenadas del centroide del objeto rojo

Así, las coordenadas del centroide del área de la visión artificial fueron eje “x” 167.38 es y eje “y” 75.14 es (figuras 3 y 4).

Implementación del sistema de control

Luego, al dato obtenido de la visión artificial se le aplicó el controlador PID para hacer el seguimiento de objetos. La parte de control se desarrolló con el software de Matlab para cada articulación. Al presentar una resolución de 320x240, se obtuvieron valores para dos sistemas. Uno para el servo que desplaza la cámara de manera horizontal y el otro que lo desplaza en forma vertical.

En primer lugar se estableció el set-point, el cual es el punto que está ubicado del centroide del objeto rojo en el centro de la cámara y para ello el robot debe de moverse en caso no este allí. Por lo tanto el error es la diferencia entre el punto del centroide y el punto central de la cámara. El error es el valor deseado menos el valor real. El valor deseado es el centro de la cámara es decir la mitad del total de los pixeles de la imagen P(160, 120). El valor real es lo que va variando en el tiempo, es decir al mover el objeto para que el robot lo siga.

A partir de los datos de los datos de la cámara se calculó el error para cada sistema:

$$\text{Error en X} = 160 - X$$

$$\text{Error en Y} = 120 - Y$$

Inicialmente se calculó la constante proporcional del controlador para el primer servo motor, dividiendo el rango del ángulo total del servomotor entre la resolución total, que sería 30 grados aprox. y 320 pix. La variable para el eje “x” fue KPX, el cual tiene un valor de: $30/320 = 0.09375$.

Luego se calculó la variación proporcional del ángulo del robot respecto a la posición del objeto rojo para el segundo servo motor al dividir el rango del ángulo total

del servomotor entre la resolución total, que sería 20 grados aprox. y 240 pix. La variable para el eje “y” fue KPX, el cual tiene un valor de: $20/240 = 0.083$.

Sin embargo al aplicar esto dentro del programa se mantiene un error constante en el estado estacionario del sistema, y en algunos casos hasta estados oscilatorios, por lo que se disminuye un poco los valores, obteniendo finalmente: $KPX = 0.09$ y $KPY = 0.07$.

Como la constante proporcional origina un error en estado estacionario, el sistema debe poseer una constante integral para eliminar el error. Para ello se uso el método de sintonización de Ziegler Nichols, obteniendo una buena respuesta frente al movimiento del objeto rojo y se logra un buen desempeño con los siguientes valores: $KIX = 0.04$ y $KIY = 0.03$

Como el movimiento del objeto es lento, no necesita una rápida respuesta de establecimiento, por lo que la constante diferencial tiene el valor de cero y se ha omitido en la tabla 1 y en el programa

Se obtiene finalmente la ecuación para el primer motor (movimiento en el eje X):

$$Gx(s) = \frac{0.09s + 0.04}{s}$$

Y del segundo motor (movimiento en el eje Y)

$$Gy(s) = \frac{0.07s + 0.03}{s}$$

Integración de todo el sistema

Finalmente, se unió el código que conecta el Matlab con Arduino, el código que procesa la imagen del entorno y el código que realiza el control PID. Todo ellos se ejecuta en un solo programa y se establecen las conexiones necesarias hacia el hardware para que envíe una señal eléctrica hacia el brazo y este pueda moverse de manera que siga el objeto indicado.

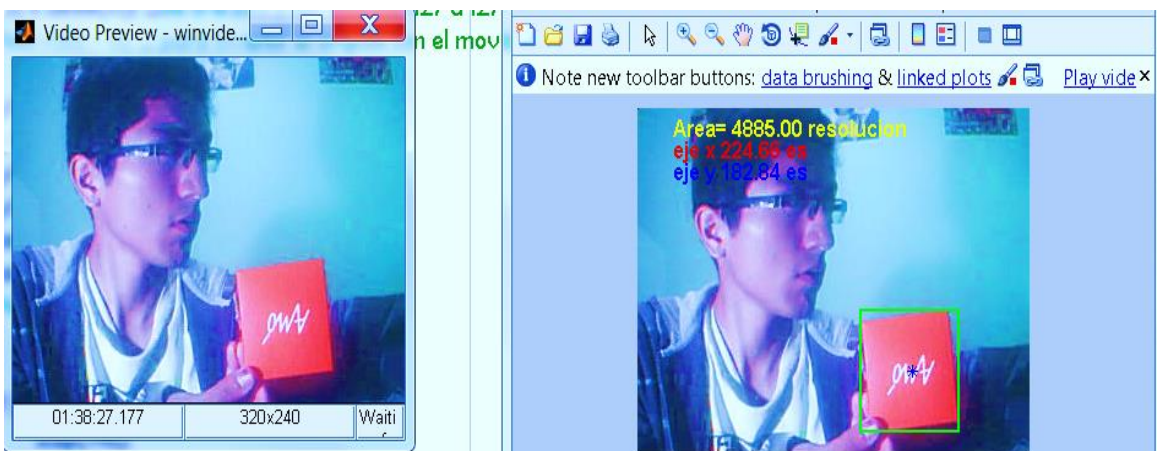


Figura 5. Imágenes observadas por la visión artificial del brazo robótico.

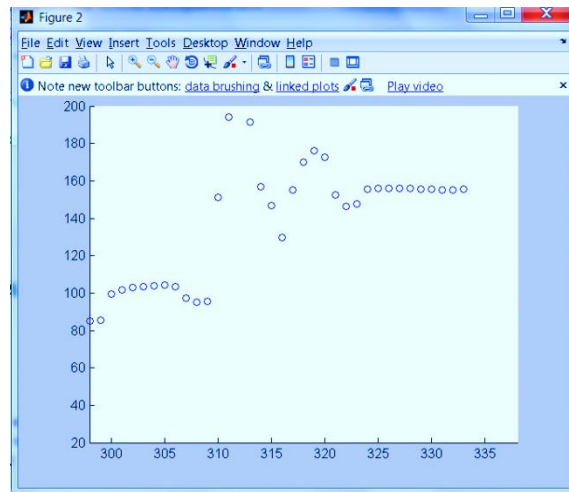


Figura 6. Respuesta a lazo cerrado con control PID del brazo robótico en el programa Matlab.

Discusión

La visión artificial nos proporciona la descripción del estado deseado que se encuentra el robot así como su evolución en el tiempo; esta información se obtiene gracias al programa de Matlab, el cual a través de la toolbox *imaq*, tomó las fotografías captadas por la cámara web y luego a través de las distintas fases del procesamiento de la imagen obtuvo las coordenadas del centroide del objeto a seguir. En la primera etapa del sistema de visión artificial para detectar un objeto de color rojo se tiene el digitalizador, el cual es el encargado de transformar la señal de video en una señal de digital capaz de ser capturada y almacenada en memoria para ser procesada por el software (Solomon & Breckon, 2011).

En las áreas de la Robótica e industria de servicios, el control de la trayectoria del brazo robot en el ejercicio constante y carga variable es importante; Tamanna y Zainul (2015) señalan que el control del brazo robótico es todo un reto debido a la participación de varias incertidumbres tales como tiempo de diferente carga, fricción y disturbios. Estos desafíos atraen a muchos investigadores para desarrollar estrategias de control avanzadas para el brazo robótico.

En el último par de décadas el avance en la tecnología de la robótica ha sido enorme en el empleo y motivación de los empleados en el sector de servicios, o en cuidado de la salud en particular (Owais & Sajjad, 2014). En usos médicos, los robots son usados en cirugía donde la precisión y la repetibilidad de un robot supera de la mano humana; también se utilizan robots para el diagnóstico (mediante endoscopios capsulares robóticos) y la activación física (mediante prótesis, sillas de ruedas autónomas) (Jobbágy y col., 2014; Stanton & Williams, 2008). La presente investigación permitirá desarrollar algoritmos de procesamiento de imágenes flexibles, el cual puede detectar objetos de todo color en distintos escenarios y desarrollar la idea que el brazo robótico no solo es adecuado para su uso en la industria o para probar nuevas tecnologías, sino también para ayudar a los seres humanos en realizar trabajos en las industrias, apoyar en las tareas normales del hogar y en la medicina. Los prototipos robóticos controlados por el método PID, podrían aplicarse en plantas industriales automatizadas del país.

Conclusiones

El controlador PID aplicado a un brazo robótico de dos grados de libertad permitió el seguimiento de objetos de color mediante visión artificial.

El efector del brazo robótico acondicionado con una cámara web y controlado su movimiento mediante el “object tracking”, permite que la cámara siga la trayectoria de un objeto dentro de un área determinada

La implementación del controlador PID a un brazo robótico permitió desarrollar algoritmos de procesamiento de imágenes flexibles

Referencias bibliográficas

- Aybar J., Tasayco P., Zevallos R. (2010). *Control de Robot por Visión*. Facultad de Ingeniería Electrónica, Universidad Peruana de Ciencia Aplicadas. Lima, Perú.
- Cazorla, M. (2010). *Robótica y Visión Artificial*. Universidad de Alicante. www.dccia.ua.es/miguel
- Cuevas E., Zaldivar D., Perez M. (2010). *Procesamiento digital de imágenes con MATLAB y Simulink*. México DF: Edit. Alfa omega.
- Hanif, W.H., Samin, R.E., & Kader, B.S. (2012). Internet Controlled Robotic Arm. *Procedia Engineering*, 41:1065–1071. doi: 10.1016/j.proeng.2012.07.284
- Jobbágy, B., Šimšik, D., Karchák, J., & Onofrejová, D. (2014). Robotic arm with artificial muscles in rehabilitation. *Procedia Engineering* 96: 195–202
doi: 10.1016/j.proeng.2014.12.143
- Kuo, B.C. (1996). *Sistemas de Control Automático* (7ma edic.). Pearson.
- Ogata, K. (2003). *Ingeniería de Control Moderna* (3ra edic.). Pearson.
- Owais Q., M. & Sajjad S.;R.(2014). The Impact of Robotics on Employment and Motivation of Employees in the Service Sector, with Special Reference to Health Care. *Safety and Health at Work*, 5:198-202.
<http://dx.doi.org/10.1016/j.shaw.2014.07.003>
- Solomon, C., & Breckon, T. (2011). *Fundamentals of digital image processing*. Edit. Wiley Brackwell.
- Stanton, C., & Williams, M.A. (2008). Robotics: State of the Art and Future Challenges, Imperial College Press. *Artificial Intelligence*, 172:1967–1972.
doi:10.1016/j.artint.2008.10.003
- Tamanna S., N. & Zainul A., N. (2015). Function Approximation Technique based Sliding Mode Controller Adaptive Control of Robotic Arm with Time-Varying Uncertainties. *Procedia Computer Science*, 76:87–94.
doi: 10.1016/j.procs.2015.12.283