

Clasificador de Objetos Utilizando Robot Delta 3GDL y Visión Artificial.

José Gálvez
Facultad de Ingeniería
Universidad Don Bosco
El Salvador, Centro América
josegalttoy@gmail.com

Noé Monterrosa
Facultad de Ingeniería
Universidad Don Bosco
El Salvador, Centro América
noe.m92@hotmail.com

José Barrera
Facultad de Ingeniería
Universidad Don Bosco
El Salvador, C.A.
ren_barrera@hotmail.com

Resumen— Dentro de las aplicaciones principales de los robots delta en la industria, se encuentra la manipulación de objetos pequeños a grandes velocidades, es por ellos que es necesario una gran cantidad de información sensorial para lograr identificar donde se encuentran, qué dimensiones poseen o incluso de qué color son, a veces los sensores convencionales no son suficientes para la captación de dicha información, es por ello que con la ayuda de visión artificial es posible proporcionar toda la información necesaria para lograr manipular los distintos objetos.

En el presente artículo se muestra la realización de un identificador y clasificador de objetos (cilíndricos con tapaderas de distintos colores) trasladados mediante una banda transportadora ubicados de forma aleatoria. Se realiza la identificación con el uso de un sistema de visión artificial y la clasificación de los objetos empleando un robot delta 3GDL. La operación completa del proceso se auxilia de una interfaz Matlab-Arduino.

Palabras Clave— Cinemática inversa, GUI, pixel, robot delta, visión artificial, umbral, Arduino.

I. INTRODUCCION

Debido a los grandes avances de las distintas áreas industriales dentro de la ingeniería que se están produciendo en la actualidad se necesita de un alto desempeño en la producción y lograr una mayor eficiencia. La introducción de la robótica en la industria ha sido la clave para lograr dichos objetivos, así como para reducir el riesgo de accidentes para los operarios.

Dependiendo de la aplicación específica a desarrollarse es la configuración morfológica y parámetros característicos de los robots industriales a emplear, éstos pueden ser clasificar según la geometría de sus estructuras mecánicas (seriales o paralelos) o de acuerdo a ciertos parámetros como: número de grados de libertad, espacio de trabajo, capacidad de posicionamiento del punto terminal (precisión), capacidad de carga y velocidad.

Los robots delta 3GDL según su estructura mecánica son de los más destacados dentro de las configuraciones de los robots paralelos, estos consisten de tres brazos conectados a una unión universal en su base, por el modo en que está diseñada su estructura, con una configuración de paralelogramos en la

geometría de sus eslabones. Este robot mantiene la orientación de su efector final siempre de modo paralela a la base.

Industrias como la farmacéutica, alimenticia, automotriz, electrónica entre otras son un claro ejemplo del empleo de robots paralelos dentro de sus líneas de producción, la manipulación de objetos pequeños de pesos no tan considerables a grandes velocidades es una de las finalidades de utilizar un robot paralelo como el delta de 3GDL. Para lograr la ardua tarea de mover grandes cantidades de objetos es necesario un procedimiento adecuado al momento de sensar los parámetros indispensables para identificar por completo todos los objetos a mover y sus características, es por eso que hoy en día es común encontrar que este tipo de procesos son auxiliados por sistemas de visión artificial.

II. ROBOT DELTA

De acuerdo a la geometría de su estructura mecánica los robots se pueden clasificar en dos categorías, los robots seriales y paralelos. Los robots paralelos a comparación de los robots seriales tienen la ventaja de ser más rápidos y más precisos [1], por estas características son usados en muchas áreas de la automatización en la industria.

Una parte muy importante de nuestro proyecto es la implementación de una función con la cual se logre hacer la cinemática inversa del robot delta. La cinemática del robot estudia la relación entre la posición de los actuadores y la posición de la plataforma móvil. Para los robots paralelos, como es el caso del robot delta, el análisis de la cinemática inversa es mucho menos complicado que la de los robots seriales. Sabiendo las coordenadas de la plataforma móvil se busca, con la cinemática inversa, conocer la posición de los actuadores, en este caso los servomotores 1, 2 y 3. En el sistema, la cámara web por medio de Matlab proveerá las coordenadas X y Y de los centroides de los objetos a clasificar, una vez ya almacenadas las coordenadas de todos los objetos y se ha detectado que están en la posición de la banda donde el robot los puede alcanzar se procede a realizar la función de cinemática inversa para mover los actuadores del robot y posicionar la plataforma móvil en los centroides de cada uno de

los objetos para luego tomarlos por medio de una ventosa neumática y clasificarlos de acuerdo a su color. Las coordenadas Z de los objetos a tomar son fijas debido a que dependen de la altura de la banda y del objeto a clasificar, las cuales se mantienen constantes.

Para lograr la función de la cinemática inversa se creó un grupo de diferentes funciones en Matlab, teniendo como resultado final una función que engloba a todas las demás para obtener la posición de cada actuador. Cada función responde a una fórmula determinada del análisis de la cinemática inversa de [2]. La fórmula para obtener θ_1 es la siguiente:

$$\theta_1 = 2 \tan^{-1} \frac{-e_2 - \sqrt{e_2^2 - 4e_1e_3}}{2e_1} \quad (1)$$

De la fórmula para obtener θ_1 se desprenden 3 términos: e_1 , e_2 y e_3 ; estos términos se obtienen por medio de 3 funciones diferentes en Matlab las cuales se llaman de la misma manera. Los valores de estas funciones dependen de la geometría del robot como el radio de su plataforma fija, radio de la plataforma móvil, longitud eslabón 1 y la longitud del eslabón 2; además de las coordenadas X, Y y Z de la plataforma móvil. Luego se creó otra función que engloba a las 3 primeras para poder obtener el resultado de la fórmula 1 y así obtener el ángulo θ_1 . Para obtener los ángulos de los 2 actuadores restantes se realizaron otras 2 funciones diferentes en las que para cada función se utilizó una matriz de rotación para transformar las coordenadas X y Y, la rotación fue de 120° para el actuador 2 y para el actuador 3 fue de 240° como nos lo indica [3].

Las 3 funciones donde se obtuvieron los datos de los ángulos se llaman $tetha1$, $tetha2$ y $tetha3$. Cabe recalcar que debido a la forma en que están desarrolladas las formulas se les tuvo que agregar un offset de -90° a la respuesta final de los ángulos, esto debido a que la posición horizontal de los actuadores representa 0° geométricos, sin embargo, para nuestro robot por la posición como se encuentran colocados los servomotores la posición horizontal representa 90° mecánicos, los servomotores son puestos de esta manera ya que es en esa posición donde se encuentran a la mitad de todo su recorrido y así se logra aprovechar los 180° mecánicos que tienen de libertad de movimiento. También en estas funciones se ocupó una sentencia if para detectar si el resultado era un ángulo negativo y al mismo tiempo obtener el ángulo complementario, con esto se logra obtener ángulos positivos que serán transformados en señales equivalentes para los servomotores por otra parte del programa.

Por último se creó una función global llamada $tethas$ la cual da por respuesta una matriz de 3×1 donde cada término es uno de los 3 ángulos calculados con las funciones $tetha1$, $tetha2$ y $tetha3$ que se encuentran dentro de esta función global.

Luego de haber obtenido los 3 ángulos estos se envían por medio de una interface con un Arduino UNO hacia los servomotores. El Arduino interpreta los ángulos de cada servo

y manda por sus pines 9,10 y 11 señales PWM equivalentes que manipulan la posición de los actuadores.

III. PROCESAMIENTO DIGITAL DE LA IMAGEN

El procesamiento digital de imágenes es un campo en el que poca gente se adentra, debido a que piensan que es poco rentable o poco aplicable para distintos ámbitos, lo cual no es cierto. El procesamiento digital de imágenes es una herramienta de alto nivel que permite obtener una imagen real de un proceso y a partir de esta extraer la información necesaria para el análisis de dicho proceso o de un sistema. El primer paso de la visión artificial del robot fue la adquisición de la imagen en tiempo real, para lo cual se adoptó la modalidad de video de Matlab y la imagen a analizar se extrajo mediante un "snapshot", es decir una imagen temporal, a la cual se le hace un tratamiento preliminar mediante un paso de la imagen RGB a binaria con un umbral adecuado para se limpie la imagen de las señales de ruido. Para el caso del robot delta se ha aplicado la visión artificial con el fin de conocer las coordenadas de los centroides de los botes que se transportaran en la banda. ¿Para qué sirve conocer los centroides?

Los centroides sirven para conocerlos debido a que proporcionan una coordenada en la que se puede ubicar el efector móvil del robot delta, es decir conocer las coordenadas en X y Y de la posición real de los botes en la banda, por ende permite la ubicación correcta de la ventosa neumática que movilizara los objetos. Además conocer la coordenada de los centroides permite que dentro del software se haga un reconocimiento del color de la tapa de cada bote, esto se produce así a raíz que el comando "impixel" es utilizado para conocer la cantidad de Rojo, Verde y Azul que hay en un pixel seleccionado, por lo que se selecciona la coordenada del centroide para que el programa se ubique de forma automática en un pixel de cada uno de los objetos de la imagen y no de manera manual como lo exige el comando. El comando "impixel" sirve para reconocer el color que se va a reconocer dentro del software, ¿Cómo se logra el reconocimiento de colores? El color se ajusta con unas pruebas preliminares, en las cuales se estableció rangos de porcentaje de Rojo, Verde y Azul aceptables para el color Rojo, rangos que son proporcionados en una matriz de $[3 \times 1]$ por el comando "impixel". La web Mathworks nos ayudó para comprender todos los comandos utilizados [4].

IV. CONSTRUCCIÓN DEL PROTOTIPO

Para la construcción de la estructura, la cual soportará al robot delta 3GDL y de igual forma a la cámara a utilizar, se empleó tubo cuadrado de hierro chapa 16 de una pulgada de ancho. En la figura 1 se puede observar el diseño CAD de una parte del prototipo.

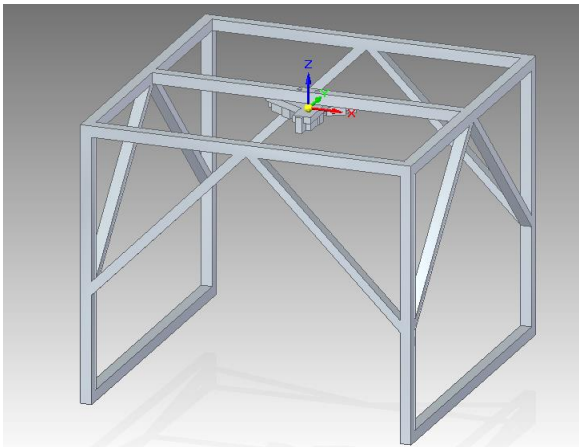


Fig.1 Armazon prototipo.

La base del robot delta 3GDL es realizada con madera en la cual se encuentran acoplados los servomotores quedando de la siguiente manera:

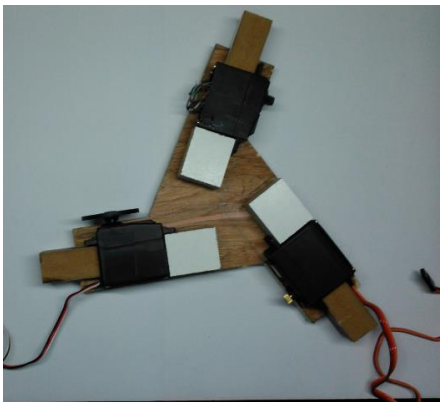


Fig.2 Base móvil robot delta.

Se muestra la implementación de la base de madera con los eslabones fabricados en acrilico.



Fig.3 Robot delta con banda transportadora

La banda donde se trasladarán los objetos para ser transportados fue construida con el mismo tubo utilizado en la estructura, funciona con un motor que trabaja a 120 VAC y una faja de cuero negro que evita el reflejo para una obtener una

mejor captura al momento de realizar el procesamiento de la imagen.



Fig.4 Banda transportadora.

Una vista completa del prototipo con el robot delta 3GDL, la banda y la cámara ya incorporada se puede observar en la figura 5.



Fig.5 Prototipo con camara ya incorporada.

V. SIMULACIÓN DEL PROTOTIPO

Lo primero que hacemos al iniciar la simulación es ajustar el área de interés de la cámara para evitar que otros objetos puedan interferir con el procesamiento de la imagen, como se aprecia en la figura 6.

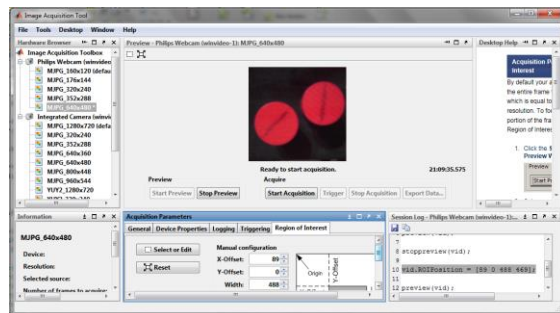


Fig.6 Ajuste de área de interés.

Se introduce el comando imaqttool y ahí se selecciona la cámara a utilizar, en este caso se utiliza una cámara web VGA Phillips con máxima resolución de 640x480 pixeles. Luego se procede a correr la GUI que se ha diseñado para el prototipo. Para motivos de la simulación y debido a que no se ha completado la parte del escalado se puso en la GUI las entradas de las coordenadas X, Y y Z, cabe recalcar que cuando se tenga el programa completo estas coordenadas serán provistas por el procesamiento de la imagen. Cuando se llena el espacio de las coordenadas y se oprime el botón CALCULAR de la GUI, se comienza una comunicación entre el software de Matlab y la tarjeta ARDUINO UNO, esta servirá como interface para comunicarse a los servomotores del robot, sensores y otros actuadores. Al establecerse la comunicación, la banda comienza a avanzar y el programa queda a la espera de que la señal del sensor se convierta en 0 lógico, el sensor que utilizamos para nuestro prototipo es el CNY-70 el cual es un sensor reflexivo infrarrojo, este sensor detecta cuando la luz emitida por su LED rebota contra una superficie reflejante y regresa a su fototransistor, en la banda se colocaron pequeños pedazos de cinta reflejante para poder hacer esta detección. Cuando el estado del sensor cambia, la banda se detiene y se procede a tomar la fotografía de los botes, esta fotografía es analizada y se detectan los centroides de los objetos como se puede observar en la figuras 7 y 8, los cuales se muestran en la GUI, cabe destacar que el programa también es capaz de reconocer los colores de las tapas, en esta simulación sin embargo, solo se utilizaron tapas rojas. Luego que se haga el escalado y se conviertan esas coordenadas de pixeles a coordenadas en centímetros se utilizaran para la función tethas la cual calculara los ángulos a los que se deben de poner los 3 servomotores para llegar a las coordenadas de los centroides, en esta simulación las coordenadas son puestas por el usuario. Por último se envían los valores de los ángulos calculados con la función de la cinemática inversa a los servomotores que proceden a tomar los objetos. En la figura 7 se puede visualizar la GUI utilizada en la simulación.

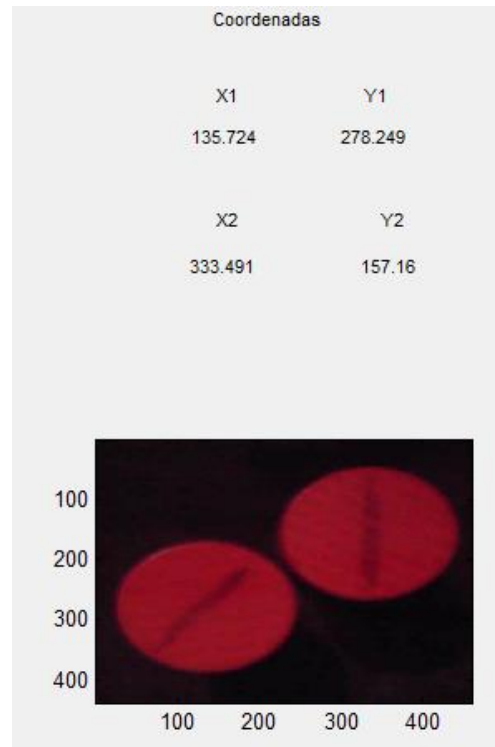


Fig.8 Acercamiento de la GUI donde se muestra la imagen tomada por la cámara y las coordenadas de los centroides.

VI. CONCLUSIONES

La visión artificial es una tecnología cada vez más usada a nivel mundial para la automatización de diferentes procesos de manufactura. Como se pudo comprobar con la construcción de este prototipo, puede trabajar en conjunto con la robótica para proveer a la industria de soluciones cada vez más automatizadas y eficientes, eliminando cada vez más el error y optimizando los procesos. En el prototipo, el programa fue capaz de detectar los centroides de las tapas de los objetos y sus colores, esto será útil para que el robot se pueda posicionar favorablemente para tomar el objeto y a su vez pueda clasificarlo de acuerdo a su color. También por medio del programa fue posible realizar una función para la cinemática inversa del robot, cuando se proveen los valores de las coordenadas X, Y y Z se obtienen los ángulos a los que se deben posicionar los servomotores para llegar a esa coordenada específica. Para futuros avances en los próximos meses se espera terminar el escalado para poder juntar completamente la parte de la visión artificial con la cinemática inversa, se espera mejorar la GUI para que muestre otras variables como el número total de objetos clasificados y el color de los mismos, y además, se le instalara una ventosa neumática a la plataforma móvil del robot delta.

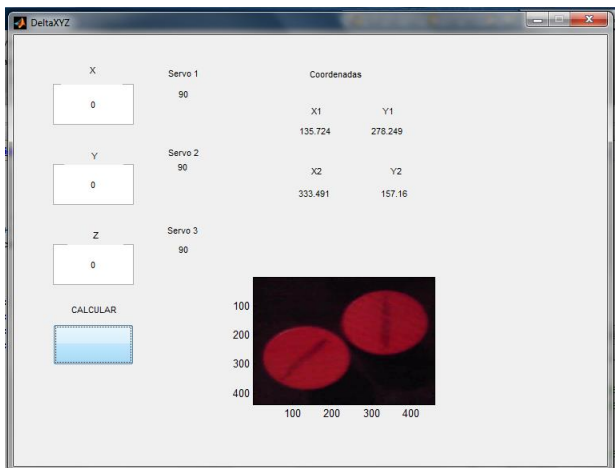


Fig.7 GUI utilizada para la simulación del prototipo.

REFERENCIAS

- [1] A. Barrientos, Fundamentos de Robotica, Mc Graw-Hill, 2007.
- [2] M. Cardona, "Una nueva metodologia para la solucion de la cinematica del robot paralelo delta," in *CONCAPAN XXXII*, Nicaragua, 2012.
- [3] M. Cardona, "Kinematics analysis of a delta parallel robot," in *CONCAPAN XXXI*, El Salvador, 2011.
- [4] Matlab , "Matlab - The language of technical computing," Julio 2015. [Online]. Available: <http://www.mathworks.com/products/matlab/>.



José Alfredo Gálvez Toyos: Nacido el 13 de Mayo de 1993, San Salvador, El Salvador. Graduado de Bachillerato General en el Colegio Champagnat (1998-2010), actual estudiante de quinto año de Ingeniería Mecatrónica en la Universidad Don Bosco. Capacitado en las ramas lingüísticas de inglés.



Noe José Monterrosa Vásquez, actualmente estudia quinto año de Ingeniería Mecatrónica en la Universidad Don Bosco, donde también es Presidente de la Asociación de Estudiantes de Ingeniería Mecatrónica (AEIMEC) y vicepresidente del Capítulo de Robótica y Automatización (RAS-IEEE). Es graduado en Técnico en Ingeniería Electrónica, con mención Cum Laude, de la Universidad Don Bosco. Se graduó con el título de Bachiller Técnico Vocacional en la rama de Electrónica del Instituto Técnico Ricaldone, consiguiendo el Diploma de Honor por las mejores notas de su promoción.



José Rene Alfaro Barrera. Nació en la ciudad de Zacatecoluca, La Paz el 28 de Junio de 1990. Se graduó de Bachillerato General, en el Centro Psicopedagógico Zacatecoluca, La Paz de 2007. Actualmente se encuentra estudiando la carrera de Ing. Mecatrónica en la Universidad Don Bosco.