

Control Visual De Un Sistema De Extracción Para Hornos Giratorios: Servo Control Visual.

M. Sandoval, A. Espinosa, E. Dean, L.A. Muñoz
LI²CoViR *

Facultad de Matemáticas, Universidad Autónoma de Yucatán
Mérida, Yucatán
M é x i c o



Figure 1: Brazo descargando un tocho del horno giratorio.

Resumen

En el presente trabajo se muestra los resultados obtenidos en el desarrollo de un prototipo robotizado para la tarea de descarga de tochos (barras sólidas de acero) en un proceso industrial realista usando servo control visual. Nuestro trabajo consistió en desarrollar e implementar los algoritmos de control visual para la automatización del sistema a partir solo de imágenes. Este trabajo se realizó en el Laboratorio de Investigación en Instrumentación, Control, Visión y Robótica de la Facultad de Matemáticas de la Universidad Autónoma de Yucatán. El servo control visual consiste en generar los movimientos del robot de forma automática acorde los a requerimientos de las tareas, en tiempo real.

1 Introducción

Este trabajo está basado en un proceso industrial real. Este básicamente consiste en extraer barras sólidas de acero, denominadas tochos, de un horno giratorio, por medio de un brazo de descarga. Las características generales del proceso son las siguientes:

- El horno giratorio tiene una forma redonda del

tipo de una dona y posee un piso sólido (suela) que gira en su interior, de allí su nombre.

- Las piezas que son extraídas del horno giratorio son barras cilíndricas sólidas de acero, denominadas tochos. Los tochos tienen varios diámetros y longitudes conocidos. La temperatura del tocho es de 1240 a 1280 C.
- La temperatura en la zona de descarga del horno giratorio es de aproximadamente 1290 C.
- El control del brazo de descarga está a cargo de un PLC.¹
- El proceso de descarga del horno actualmente se realiza manualmente; hay una persona dedicada exclusivamente para realizar esta actividad pieza por pieza.

Para efectos del presente trabajo hemos realizado ciertas modificaciones a las condiciones del proceso industrial. En nuestro caso el sistema de visión está a cargo del control del brazo robot. Se realizó la construcción de un prototipo robot del brazo de extracción, con el fin de comprobar experimentalmente el funcionamiento del sistema de visión.

La visión computacional es la rama de la Inteligencia Artificial que describe la deducción de la estructura y propiedades de un mundo tridimensional, posiblemente dinámico, a partir de una o múltiples imágenes bidimensionales del mismo mundo.

Dichas imágenes pueden ser monocromáticas (blanco y negro) o a color, y pueden ser capturadas por una o varias cámaras donde cada cámara puede ser móvil o fija. Las propiedades y estructura que intentamos deducir incluyen propiedades geométricas

*Laboratorio de Investigación en Instrumentación, Control, Visión y Robótica. E-mail:liicovir@yahoo.com

¹PLC (Programmable Logic Controllers):Microcontrolador orientado a señales industriales.

(formas, tamaños y localización de los denominados tochos), propiedades materiales (luminosidad y oscuridad de las superficies, sus colores, sus texturas y la composición de sus materiales) y la luminosidad frente a las propiedades del interior del horno.

Una de las principales aplicaciones de la Visión Computacional incluye la inicialización de pose de objetos y el seguimiento de su movimiento a partir de una estimación inicial cuyo modelo geométrico se conoce.

Existen varios algoritmos que logran solucionar este problema con cierto margen de error y con diferentes complejidades, tanto de espacio como de tiempo. Uno de los algoritmos más eficientes es el descrito en el artículo Registration of CAD-Models to Images by Iterative Inverse Perspective Matching de P. Wunsch y G. Hirzinger, que consta de dos pasos iterativos: en el primero se determinan las correspondencias entre las características de la imagen y los datos del modelo, mientras que en el segundo se calcula la transformación del cuerpo rígido tridimensional que minimiza el desplazamiento del mapeo de los puntos. En otras palabras, buscaremos el objeto en el mismo espacio tridimensional en el que se encuentra.

La Visión Computacional basada en modelos, asume que la información acerca del mundo real está disponible en forma de modelos de objetos tridimensionales. Por lo tanto, los sistemas de visión basados en modelos pueden saber como está formada la escena en términos de tales modelos, y pueden comunicar sus observaciones regresando la posición tridimensional y rotación relativa al sensor de un caso particular de un objeto modelado.

Con lo anterior en mente, el problema general fué dividido en dos partes: el servo control visual del prototipo robot del brazo de descarga y la inicialización de pose de los tochos.

2 Servo Control Visual del Prototipo Robot

2.1 Modelo Dinámico del Robot

Un robot manipulador es un sistema bastante complejo que se constituye de elementos cuyo comportamiento es altamente no lineal.

Para fines de éste trabajo, consideraremos el término tarea servo visual como el uso de información visual para el control de la pose del efector final de un robot, relativo a un objetivo que puede ser un objeto o un conjunto de características [1].

La dinámica de un robot manipulador de n-eslabones en serie, rígidos, no redundantes y completamente actuados puede ser escrita como se muestra:

$$H(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = KP \cdot \epsilon + \frac{1}{KD} \cdot \epsilon \quad (1)$$

donde $q_0 \in R^n$ es el vector de desplazamientos articulares generalizados, con $\dot{q} \in R^n$ su velocidad, $H(q) \in R^{n \times n}$ es la matriz de inercia, simétrica y definida positiva, $C(q, \dot{q})\dot{q} \in R^n$ es el vector de aceleraciones centrípetas y de Coriolis, $G(q) \in R^n$ es el vector de fuerzas gravitacionales, KP es una constante de proporcionalidad del error, KD es una constante de proporcionalidad de la derivada del error, ϵ es el error entre la posición actual y la posición deseada, $\epsilon = qd - qa$.

Debido a que el manipulador está siendo operado en el plano, perpendicular a la gravedad, quitamos el vector de fuerzas gravitacionales, quedando nuestro modelo de la siguiente manera:

$$H(q)\ddot{q} + C(q, \dot{q})\dot{q} = KP \cdot \epsilon + \frac{1}{KD} \cdot \epsilon \quad (2)$$

En nuestro caso se utilizará un robot planar de dos grados de libertad.

2.2 Descripción del Prototipo Robot.

El prototipo robot utilizado fue desarrollado por Juan de Dios Alonzo Centeno, Gilda Eugenia Gutierrez Valencia, Rodrigo Esparza Sanchez, Geliztle Parra Escamilla y Alejandro Cob Tun, todos alumnos de cuarto semestre de la Licenciatura de Ciencias de la Computación de la Facultad de Matemáticas de la Universidad Autónoma de Yucatán.

Las características generales del prototipo son las siguientes:

- Las piezas del robot fueron hechas en aluminio con el fin de disminuir el peso.
- La longitud de ambos eslabones es de 22 cm.
- El brazo robot cuenta con motores de corriente directa modelo HN-GH12-1634T.
- Se empleó un microcontrolador pic 16f877 para el control de los motores de corriente directa y para manejar la comunicación entre la computadora y los motores.

2.3 Sistema de Servo Control Visual

Para implementar el sistema de servo control visual, se empleó una cámara CCD marca Sony modelo DFW-VL500. La cámara fue colocada en forma perpendicular al plano de trabajo del brazo robot.

Se colocaron marcas de referencia en las articulaciones y en el efector final del brazo robot así como en el tocho. Estas marcas de referencia consisten de círculos de color blanco de diferentes

dímetros. El algoritmo recibe la imagen capturada por la cámara y localiza las marcas de referencia, e identifica por medio de los diámetros, de qué parte del brazo robot se trata; el de mayor diámetro corresponde a la articulación 1, el que le sigue corresponde a la articulación 2, y así respectivamente para el efector final y el tocho.

La localización de los círculos dentro de las imágenes, se realizó de la siguiente manera: a la imagen obtenida de la cámara, se le aplica una binarización con cierto umbral, un valor cercano a blanco, con el fin de tratar de dejar solo los círculos blancos. Luego a los píxeles negros los dejamos blancos y viceversa, de manera que los objetos de color cercano al blanco se convierten en negro, mientras que el resto se vuelve blanco. Posteriormente se recorre la imagen, en busca de un píxel gro, cuando se encuentra uno se llama a una función que marca a todos los píxeles negros que tengan 4 conexión con éste píxel. Esto se hace recursivamente. Al final tenemos todas las manchas negras en la imagen identificadas. Pero desafortunadamente, no todas las manchas corresponden a marcas circulares, sino también pueden haber reflejos o ruido. Por lo anterior hay que hacer una discriminación entre las marcas circulares y el ruido. Esto se realiza a través de tres criterios:

1.- Dentro de la función que realiza la identificación de las manchas negras se calcula también las coordenadas más grandes y más pequeñas tanto en X como en Y de la mancha. Con esto se puede calcular el diámetro. Por lo que se pueden ignorar las manchas que tengan un diámetro menor que un cierto valor que permita no confundirlo con una marca circular.

2.- Ya que podemos calcular un diámetro en X y un diámetro en Y y como estamos buscando marcas circulares, podemos esperar que ambos radios difieran



Figure 2: Imagen del Prototipo Robot del brazo.

muy poco. Por lo tanto podemos desechar las marcas cuya diferencia entre los diámetros sea mayor que un cierto valor.

3.- El tercer criterio utilizado consiste en el factor de regularidad. El recíproco del factor de compacidad es llamado factor de regularidad o delgadez. Esta característica mide que tan simétrica es la forma de un objeto. Este factor toma el valor máximo de 1 cuando el objeto tiene forma circular, esto nos da una idea de la redondez del objeto. Un valor pequeño de este factor nos habla de la irregularidad del objeto.

Factor de compacidad o de irregularidad. Este rasgo es, tal vez, el rasgo más importante propuesto en la literatura. El factor de compacidad FC de una forma binaria se define como el perímetro al cuadrado del objeto dividido por 4 veces su área, esto es

$$FC = \frac{P^2}{4A} \quad (3)$$

donde P es el perímetro del objeto, A es su área. El valor de FC será mínimo para un objeto circular. Aunque el valor del factor de compacidad es invariante a traslaciones, rotaciones y cambios de escala, es, sin embargo, muy sensible al ruido en el contorno del objeto.

Esto tenderá a afectar muchas veces el valor del perímetro de la forma, afectando, en consecuencia el valor de FC. Bribiesca [2] propuso una manera alterna de calcular el factor de compacidad para una forma binaria. La ventaja principal del método es que el valor para FC obtenido es mucho más robusto al ruido en el contorno. Para esto, el autor propone usar el llamado perímetro de contacto del objeto y no sólo el perímetro del objeto.

El perímetro de contacto, P_c , de una forma compuesta de píxeles, cuadrados en nuestro caso, es igual a la suma de las longitudes de los segmentos comunes a dos celdas.

El perímetro de contacto de la forma mostrada en la figura se compone de 11 segmentos, marcados con líneas negritas.

Y una vez identificadas las marcas circulares, por medio de funciones trigonométricas se calculan los ángulos aplicados en las articulaciones del brazo, es decir se encuentra la posición del brazo robot.

De forma que:

$$q_1 = \arccos \frac{X_1 - X_2}{\sqrt{(Y_1 - Y_2)^2 + (X_1 - X_2)^2}} \quad (4)$$

$$T = \arccos \frac{X_2 - X_3}{\sqrt{(Y_2 - Y_3)^2 + (X_2 - X_3)^2}} \quad (5)$$

$$q_2 = T - q_1. \quad (6)$$

2.4 Interfaz Grafica del Sistema

Para realizar la interfaz gráfica del sistema, se utilizaron las librerías desarrolladas por el Ing. Emmanuel Carlos Dean León para el manejo de cámaras. La interfaz grafica fue desarrollada en el lenguaje GTK ver 2.0 bajo la plataforma Linux Red Hat 9.

En la figura se muestra la imagen de la interfaz gráfica del sistema de control. Se ve la imagen umbralizada y las marcas circulares detectadas. Un pequeño punto blanco seala el centro del círculo detectado. En la parte inferior izquierda se muestran los ángulos en grados de la primera articulación (q_1) y de la segunda (q_2). Las imágenes son tomadas en tiempo real.

Podemos mencionar que éste sistema de servo control visual así como el brazo robot, fueron expuestos en el noveno Concurso Nacional de Minirobótica organizado por la IEEE (El Instituto de Ingenieros en Electricidad y Electrónica) y la Universidad del Valle de México, Campus Querétaro durante el 6 y 7 de mayo de 2004 en Santiago de Querétaro, Querétaro. En éste concurso, se expusieron a competencia 15 proyectos en la rama de brazos manipuladores, ocupando nuestro trabajo el séptimo lugar de la competencia.

3 Inicialización de Pose

Visión es un proceso que produce, a partir de imágenes del mundo externo, una descripción que es útil para el observador y que esté libre de información irrelevante (Marr and Nishihara, 1978). Una imagen es un registro de valores organizados en forma bidimensional, generalmente representando intensidad de radiación electromagnética. Por eso la visión consiste en transformar una descripción icónica (registro de la señal) en una interpretación (descripción simbólica).

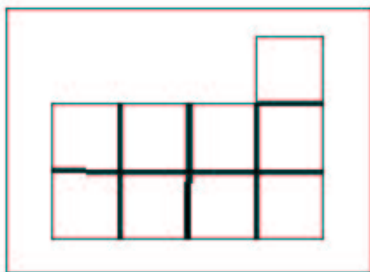


Figure 3: Imagen de Ejemplo del Perímetro de Contacto.

También se encarga de extraer propiedades invariantes en la imagen: tamaño, posición, color, forma, ... y por lo tanto, proveer información útil y que sirva de retroalimentación para las acciones que realiza el observador: manipulación, seguimiento, evitar obstáculos, reconocimiento, ...

La inicialización de pose (en Inglés registration) consiste en determinar por medio de cámaras de video la posición de un objeto fijo en un ambiente tridimensional. Este es un problema complejo ya que con una sola cámara de video solo contamos con imágenes bidimensionales del mundo, en las que el objeto a localizar se encuentra rodeado de todos los demás elementos que conforman el ambiente y que nos pueden crear varios problemas, al tratar de encontrar el objeto en la imagen, como son: oclusiones (una parte del objeto está oculta), ambigüedades (objetos que se parezcan al que buscamos), distorsiones de la imagen debido a la luz, etc. Existe también la visión computacional basada en modelos, que consiste en localizar un objeto del cual conocemos de antemano su modelo tridimensional (3D), es decir, conocemos su forma y sus dimensiones.

Un primer acercamiento al problema fue hecho con el algoritmo propuesto en [3]. Este método de Visión Computacional basada en modelos, considera técnicas relacionadas con el mapeo local y maneja la misma idea expuesta en [4]. Se estudió y aplicó el algoritmo a nuestro caso utilizando para ello una simulación, cuyo código se encuentra en [3]. Los resultados fueron los siguientes:

Utilizando el programa de simulación gráfica geomview corriendo bajo la plataforma Linux Red Hat 9, se creó un modelo del tocho. Las dimensiones del modelo fueron 5 cm de radio y 10 cm de alto. Se simuló el algoritmo de Visión Computacional basada en modelos en el programa de simulación gráfica geomview. Para ejecutar el algoritmo, se necesita una imagen del objeto que se desea localizar, y como se había real-

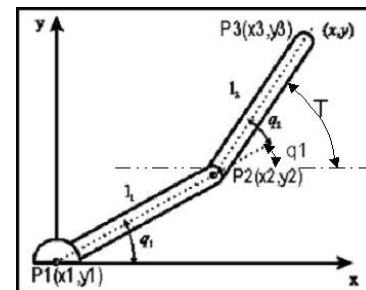


Figure 4: Calculo de los ángulos en las articulaciones del brazo.

izado el modelo en el programa geomview, lo que se hizo fue una captura de pantalla, de la ventana del programa. En la siguiente figura se ve la simulación del algoritmo, donde se muestra la imagen capturada del modelo que simula la imagen real que se tomaría del tocho, se ven los rayos de proyección que pasan por los puntos característicos del modelo en la imagen. En el fondo se ve el modelo CAD del tocho. Esta imagen corresponde al resultado final del algoritmo. Podemos ver que el modelo no toma la posición que se muestra en la imagen. Se observó que el algoritmo caía en muchos mínimos locales, por lo que el modelo CAD solo oscilaba, permaneciendo vertical. Una razón que se encontró fue la forma simétrica del modelo.

Por lo anterior se decidió que esta no sería una buena opción para solucionar nuestro problema. Por lo tanto se desarrolló un algoritmo para la inicialización de pose basada en modelos.

Ya que teníamos un proceso industrial real, se decidió usar las particularidades de él en el desarrollo del algoritmo, es decir, se desarrolló un algoritmo que resuelve este problema en particular. Los aspectos que se tomaron en consideración fueron: sólo se inicializaría la pose de tochos. Los tochos se mueven sobre un plano (la duela del horno giratorio).

Por lo tanto el algoritmo no funcionaría para otros tipos de modelos. Al considerar que los tochos se mueven sobre un plano, se limita el número de configuraciones que puede adquirir el tocho, lo que ayuda a la hora de calcular su pose.

3.1 Algoritmo de Inicialización de Pose Centrado en la Cara Circular del Tocho

Para la inicialización de pose se desea determinar la posición y la orientación del objeto en el espacio. Para ello, el algoritmo se centra principalmente en la cara circular del tocho. Se asume que en la imagen siempre aparecerá la cara circular del tocho.

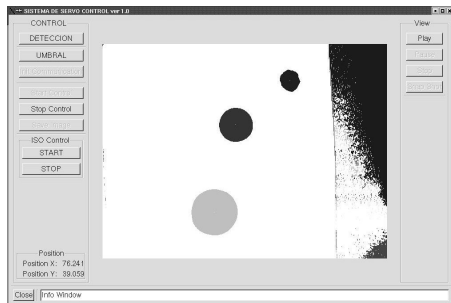


Figure 5: Interfaz gráfica del Sistema de Servo Control Visual.

Realicemos el siguiente experimento: tomemos una moneda y coloquemosla a una distancia dada y exactamente perpendicular a nuestros ojos. Lo que observamos, es que la moneda tiene la forma de un círculo. Ahora, manteniendo la moneda a la misma distancia y perpendicularmente a nuestros ojos, rotemos la moneda sobre un mismo eje. Notaremos como la moneda, de una forma circular que tenía, comienza a tomar la forma de una elipse. Si continuamos rotando la moneda, veremos como la elipse va desapareciendo hasta convertirse en una línea vertical, que mide lo mismo que el diámetro de la moneda. La elipse que se forma tiene la particularidad de que su eje mayor mide lo mismo que el diámetro del círculo original. Por lo tanto si conocemos el diámetro del círculo y conocemos el eje menor de la elipse que se forma al rotar el círculo, podemos encontrar mediante la función arco coseno, el ángulo que rotó el círculo para formar, en perspectiva, la elipse. Esta es la idea general que maneja el algoritmo para encontrar la orientación del tocho.

Considere la siguiente disposición: la cámara se coloca perpendicular al plano X-Y, sobre el eje Z. El tocho se coloca en perpendicular a la cámara, por lo que se observará un círculo. Se mide el diámetro en píxeles. Luego se rota el tocho con respecto al eje X, manteniendo la misma distancia entre la cámara y el objeto. Se mide el eje menor en píxeles. Luego se divide el eje menor de la elipse entre el diámetro del círculo y al resultado se calcula el arco coseno. En la figura podemos ver un esquema visto de manera perpendicular al plano Y-Z, donde D es el diámetro del círculo sin rotar, E es el eje de la elipse y α es el ángulo de rotación con respecto al eje X. Por lo tanto:

De manera análoga se sigue para la rotación con respecto al eje Y. Una rotación con respecto al eje Z no es considerada, ya que al tratarse de un cilindro, no representa un cambio que afecte el proceso. Hay que hacer notar que no solo la rotación produce

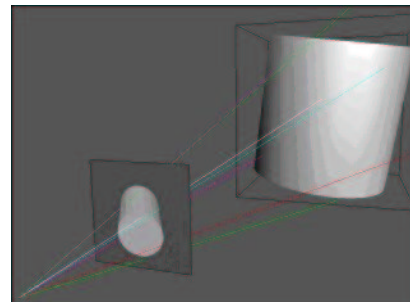


Figure 6: Imagen final de la simulación del algoritmo de inicialización de pose.

que un círculo se deforme en una elipse, también la translación produce esta deformación. Por lo que el objeto se coloca en el origen para evitar que la deformación se deba a la translación.

3.2 Construcción de la Maqueta

Para experimentar con el algoritmo desarrollado, se procedió a la construcción de una maqueta para simular el ambiente del horno.

4 Conclusión

Se presentan los resultados iniciales de un proceso de inicialización de pose y de control visual de una tarea de localización y prehensión de un objeto en un espacio tridimensional.

References

- [1] P. I. Corke S. Hutchinson, G. D. Hager, "A tutorial on visual servo control," . *Robotics and Automation.IEEE*, vol. 12, no. 5, october 1996.
- [2] E. Bribiesca, "Measuring 2d shape compactness using the contact perimeter," *Computers Math. Applications*, vol. 3, no. (11), pp. 1-9, 1997.
- [3] Minerva Rojas Sols, *Inicializacin de Pose para la Visin por Computadora basada en Modelos CAD*, Ph.D. thesis, Facultad de Matemáticas, Universidad Autónoma de Yucatán, 2003.
- [4] P. Wunsch y G. Hirzinger, "Registration of cad-models to images by iterative inverse perspective matching," *Proceedings of the 13th International Conference of Pattern Recognition*, pp. 77-83, 1996.

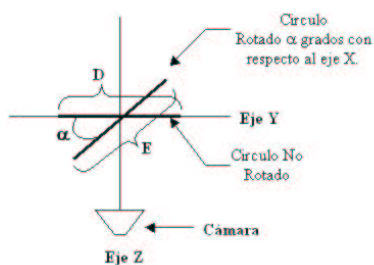


Figure 7: Esquema del método de inicialización de pose.