

## Actualización del controlador para un Robot PUMA 761 e implementación de sistema de visión para crear una aplicación por el método de control “Point-and Click” CSM.

Ing. Samantha Berenice Luna Gutiérrez, UASLP  
Ing. Felipe Antonio de la Peña Contreras, UASLP

Dr. Juan Antonio Cárdenas Galindo, UASLP  
Dr. Emilio J. González Galván, UASLP

Centro de Investigación y Estudios de Posgrado,  
Facultad de Ingeniería, UASLP  
Ave. Dr. Manuel Nava No. 8, San Luis Potosí, S.L.P.  
C.P.78290, San Luis Potosí, S.L.P., México.  
[sblgca@hotmail.com](mailto:sblgca@hotmail.com), [felipep@uaslp.mx](mailto:felipep@uaslp.mx)

[jjcardena@uaslp.mx](mailto:jjcardena@uaslp.mx), [egonzale@uaslp.mx](mailto:egonzale@uaslp.mx)

### Resumen

En este documento se detalla la actualización del controlador original de un robot industrial PUMA 761, la implementación del sistema de visión a través de una interfaz gráfica del usuario (GUI<sup>3</sup>) y la aplicación del paradigma de control basado en visión conocido como Manipulación en espacio de cámara (CSM<sup>2</sup>), lo anterior para lograr posicionar el efector final de un robot manipulador en un punto seleccionado (en el espacio tridimensional) utilizando el método “Point-and-Click”<sup>1</sup> señalado en espacio de cámara (imagen bidimensional) por el usuario. Este posicionamiento contará con orientación perpendicular a la superficie que lo contenga en espacio tridimensional. El primer propósito de este trabajo es crear un sistema de control de visión, utilizando para ello un robot con el que ya se contaba e implementando el *hardware* necesario para el desarrollo del controlador. El segundo propósito de este trabajo es implementar este método de control (CSM) generando todo el *Software* necesario para lograr la aplicación deseada de orientación y posición vía CSM “Point-and-Click” como una aplicación de código abierto.

Tópicos: Visión por computadora, Algoritmos de control, Robótica, CSM.

### I. INTRODUCCIÓN

Los Robots PUMA (Programmable Universal Machine for Assembly), son probablemente los robots más utilizados en los Laboratorios en las Universidades, y son de los ensambles de robots más comunes. La serie 700 de UNIMATE es la que tiene los Robots más grandes y más poderosos eléctricamente en la línea PUMA.

El potencial de expandir sistemas de visión simples y mejorar el desarrollo de manipuladores con múltiples grados de libertad ha sido limitado en la práctica debido a problemas que están asociados con la calibración y el análisis de imágenes. Los problemas de calibración, incluyen la calibración de la relación entre las coordenadas del vector de la junta del manipulador con la posición y orientación del efector final, y la relación entre la apariencia de los objetos

en la imagen plana con su posición física (en el mundo real). CSM elimina completamente la dependencia del control de cualquier marco de referencia particular, y usa un modelo de estimación estocástico no lineal de bajo orden para estimar en forma precisa y continua la relación directa entre las coordenadas del vector de la junta y la posición en cada imagen plana del objeto manipulado [14].

Pueden definirse muchas tareas que requieren destreza robótica en términos de las acciones de su efector final relativos a los puntos diseñados en una superficie. Estas tareas incluyen por ejemplo el proceso de soldado, lavado, detección de defectos de materiales por pruebas no destructivas, grabado, pintado, etc. El método de CSM es aplicable para realizar todas estas tareas a través del análisis de imágenes inherentes a la determinación de la superficie de trabajo, esto, con la combinación de una GUI y un punto láser montado en una unidad de movimiento de dos grados de libertad (Pan-Tilt), y marcas visuales distintivas colocadas en la herramienta que se localice en el efector final. La concurrencia del método CSM, la GUI, el láser, la unidad Pan-Tilt y las marcas visuales colocadas en la herramienta trabajando, todos ellos en forma conveniente y la interconexión entre dichos elementos para lograr este control es el método CSM “Point-and-Click” (Patente US 6,304,050; Octubre 2001).

### II. ANTECEDENTES

*Comparación entre las principales Técnicas de Visión por Computadora.*

Visión computacional, también llamado entendimiento de imágenes, describe la deducción automática de las estructuras y propiedades de la posible dinámica en el mundo tridimensional a partir de una o varias imágenes

bidimensionales. Las imágenes pueden ser monocromáticas o a color, y cada una puede ser fija o estacionaria [12].

Controlar robots usando sensores de visión es simplemente el factor más limitante en el campo de la robótica. Se han realizado muchos esfuerzos en este campo en años recientes, entre las estrategias más populares de visión para control de robots están; la Teleoperación, Métodos de Calibración, Servo Visual (VS) y Manipulación en Espacio de Cámara. Aquí se sintetizan algunas de las características de estas estrategias.

**Teleoperación:** Es la manera más simple de controlar robots usando el sentido de la vista. El control de las articulaciones del robot para realizar la tarea deseada es realizado por un ser humano, basado en la retroalimentación desde el sensor de visión. El control usado es el reconocimiento a través de visión realizado por un operador que usa su propia vista o a través de un sensor de visión y transmisión de imágenes hacia el controlador humano. La teleoperación toma ventaja de la capacidad de percepción del sentido de la vista humano [1]. Una de las dificultades de estos sistemas es el tiempo de retraso asociado con la señal de video. El proceso de completar una tarea se puede volver sofocante, si no es que enteramente imposible, si estos retrasos son largos (cuando se hace en forma remota).

**Método de Calibración:** Mucha de la investigación está dirigida hacia el uso de métodos de calibración. Tales métodos encabezan dos eventos de calibración separados, los cuales son igualmente críticos para el éxito de la maniobra: 1. Calibración de las cámaras relativas a un sistema de coordenadas físicas “mundiales”. 2. Calibración de la cinemática del robot relativa al sistema de coordenadas físico [3].

La toma de imagen y la manipulación son pasos separados. En el primer paso las cámaras localizan las coordenadas físicas de la pieza de trabajo. Entonces las coordenadas internas del robot se ajustan a una posición final calculada en base a la cinemática inversa. La desventaja de este método es la sensibilidad de los sensores de visión a condiciones ambientales de operación. Además los parámetros de cámara que se calibran pueden cambiar significativamente debido a variaciones en la temperatura o pequeñas perturbaciones físicas, además de que es difícil mantener la calibración del robot debido al cambio de cinemática, dinámica, etc.

**Servo Visual:** En este método los pasos de toma de imagen y manipulación no necesariamente se hacen por separado. Los organismos proveen un ejemplo de coordinación mano-ojo, que está basado en la realización de maniobras en términos de “espacio-sensor”, sin referencia a ningún sistema de coordenadas absoluto [1]. Las disparidades entre la localización de las características clave en el marco de sensado actual y el deseado, se usan en una ley de control para buscar un estado de error cero. El paradigma requiere estimación Jacobiana en tiempo real para las actualizaciones de datos en lazo cerrado. El retraso de la actualización de información interrumpe la ejecución del método.

**Manipulación en Espacio de Cámara (CSM):** Esencialmente esta técnica identifica las relaciones de entrada y salida de una planta mediante un sistema de control en lazo abierto para llegar a su estado objetivo [3]. CSM [2] comparte con VS la propiedad de trabajar sólo en referencia al marco de visión del sensor. Sin embargo, las siguientes características favorecen el uso de CSM sobre métodos VS: 1) CSM Trabaja en el marco de referencia de visión de los sensores, evitando la necesidad de calibración de las cámaras y el robot, 2) CSM usa modelos algebraicos para estimar la relación entre el espacio de cámara y el espacio de articulación del robot [6]. 3) A diferencia de VS, CSM no requiere una estimación en tiempo real del Jacobiano de la imagen, en lugar de esto se determina la cinemática en espacio de cámara por la estimación continua de los seis parámetros de visión [7]. 4) CSM trabaja en lazo abierto utilizando los nuevos datos adquiridos para la actualización de los parámetros estimados. En la Fig 1 se muestra un sistema típico para una aplicación que utiliza CSM.

En una aplicación CSM típica, el “reconocimiento a través de visión” es realizado por operadores humanos en la estación de trabajo usando una Interfaz Gráfica de Usuario la cual despliega el espacio de trabajo del robot. La GUI permite al operador seleccionar la pieza de trabajo y controlar un conjunto de parámetros necesarios para la operación. Por ejemplo, la velocidad del robot, la tarea a desarrollar, la separación entre la pieza de trabajo y el efector, etc. En la Fig 2 se muestra un ejemplo de la interfaz gráfica.

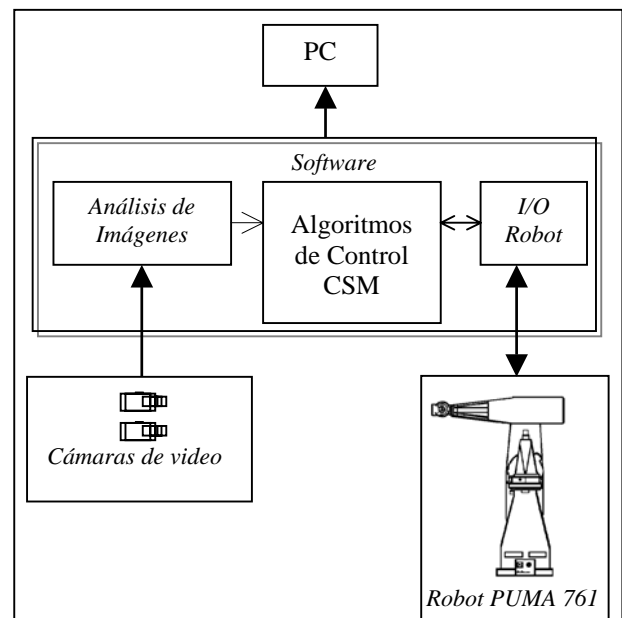


Fig. 1 Diagrama del sistema CSM

### III. ACTUALIZACIÓN DEL CONTROLADOR E IMPLEMENTACIÓN DE SISTEMA DE VISIÓN PARA APLICACIÓN CSM.

#### Arquitectura del Sistema.

El sistema consta de 2 cámaras CCD<sup>4</sup>, una unidad Pan-Tilt, de dos grados de libertad, un robot industrial PUMA 761, y una PC de control (a la cual están conectados todos los dispositivos anteriores). En la PC de control los algoritmos de control CSM y las rutinas controladoras del brazo robot son ejecutadas.

#### Robot.

El lenguaje de programación del controlador original del Robot PUMA 761 es el VAL II., cuya capacidad es de 64 KW CMOS en memoria de usuario (RAM), una unidad de disco de 5 ¼, un módulo I/O de 32 entradas, 32 salidas. Cuenta con 16 tarjetas de control y 5 tarjetas de alto poder (tres tarjetas amplificadoras para los ejes 1,2 y 3, una tarjeta de funciones de alto poder y una tarjeta de interfase C).

Se trata de un robot de 6 grados de libertad, cuyas juntas son tipo revoluta, y cada una de ellas es movida por un servomotor de imán permanente. Cada servomotor cuenta con frenos de disco electromecánico los cuales se activan cada vez que se desenergiza el robot o cuando hay algún paro de emergencia. Así también, cada motor cuenta con un encoder incremental de tres canales y un potenciómetro. Las juntas más grandes cuentan con switch límite (J1,J2 y J3). Cada motor cuenta con un tren de engranes que reduce la velocidad resultante de cada junta, las cuales se muestran en la Fig 3. Existe dependencia mecánica del movimiento entre los ejes 4,5 y 6, donde el movimiento del eje 4 mueve los ejes 5 y 6, y el movimiento del eje 5 mueve el eje 6. El sistema original cuenta con frenado dinámico para paro de emergencia.

Para el desarrollo del presente proyecto se reemplazó el controlador original por una PC (PC de control), una tarjeta controladora de movimiento Galil PCI DMC-1806 para 6

ejes, un módulo de interconexión con tarjetas amplificadoras Galil AMP-1902 para 2 ejes y AMP-1904 para 4 ejes, ambos PWM, una fuente de 24 VCD para la liberación de los frenos electromecánicos y una fuente de 60VCD para la energización de las tarjetas amplificadoras.

La tarjeta controladora de movimiento cuenta con su driver de instalación, un software desde donde se fueron realizando las pruebas para cada eje, y las librerías correspondientes para la activación de los comandos para la programación de la tarjeta. Basados en estas librerías se implementó en lenguaje C++ los algoritmos para controlar el Robot desde la PC. Consideraciones como son: la dependencia mecánica entre ejes, la cinemática interna del manipulador para la obtención de la posición tridimensional con respecto a la base del robot, las relaciones de transformación, los límites por software, etc. fueron implementados en los algoritmos de dicho programa, de tal forma que todo esto es transparente para el usuario. La operación del robot se efectúa desde la PC de forma similar a como se haría con su controlador original para efectuar movimientos del tipo punto a punto con el efector final.

#### Interfaz Gráfica.

El hardware para la digitalización de imágenes es una tarjeta PCI Frame Grabber Data Translation y dos cámaras Sony de alta resolución B&W CCD de 1/3", modelo SSC-M383, con lentes varifocales Rainbow 6.5-82.5 mm modelo 1639VCS.



Fig. 2 GUI

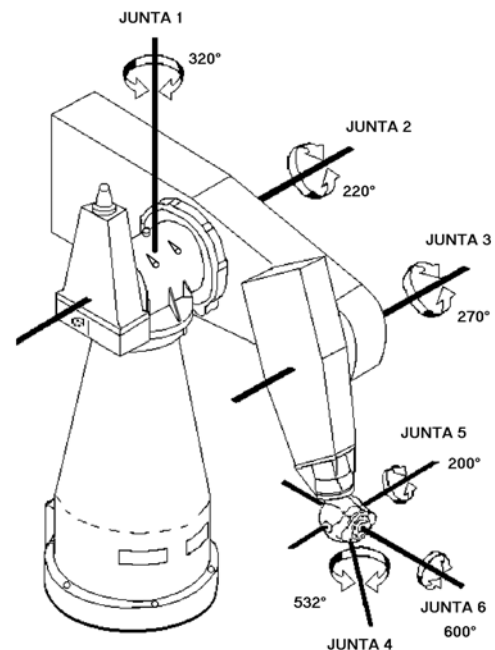


Fig. 3 Juntas del Robot PUMA 761

Para el control supervisor de alto nivel ha sido desarrollada una GUI para habilitar el acceso visual al espacio de trabajo. La GUI despliega la imagen capturada transmitida a través de la tarjeta Frame Grabber. Gracias a ésto el usuario con el control puede definir el punto de interés en el espacio de trabajo que se presenta en el monitor de la computadora. Fue generado en lenguaje C++ un programa para la adquisición de imágenes.

#### *Ubicación de Objetivos Seleccionados en espacio de Cámara en el Espacio de Trabajo*

La unidad Pan-Tilt es una unidad de dos grados de libertad como se muestra en la Fig 4. Se compone de dos motores a pasos controlados desde la PC de control por medio de una unidad MD-2 (Dual Stepper Motor System, Arrick Robotics) y una caja reductora de engranes en la cual van montados los motores. En el motor encargado del movimiento vertical (Tilt) se monta la unidad láser. Para lograr la orientación correcta del punto láser se utiliza el jacobiano de sensibilidad de primer orden de acuerdo a ecuación (1):

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{bmatrix} \begin{bmatrix} \Delta \alpha_1 \\ \Delta \alpha_2 \end{bmatrix}, \quad (1)$$

donde  $\Delta \alpha_1$  y  $\Delta \alpha_2$  son los incrementos en los ángulos de las juntas de la unidad Pan-Tilt a las que se debe el movimiento horizontal y vertical (o vertical y horizontal) respectivamente,  $J$  es la matriz jacobiana,  $\Delta x$  y  $\Delta y$  son los desplazamientos correspondientes en espacio de cámara en el eje  $x$  &  $y$  respectivamente (en píxeles).

Una vez que el usuario selecciona desde la pantalla, el punto sobre el cual debe posicionarse el efector final del robot, la unidad *Pan-Tilt* deberá posicionar el punto láser sobre la superficie de trabajo en el punto especificado [13]. Para dicho control se implementaron en Lenguaje C++ los algoritmos para el control de la unidad Pan-Tilt. El algoritmo utilizado es:

1. Colocar la unidad Pan-Tilt de tal forma que el punto láser apunte a la superficie de trabajo.
2. Fijar el error permitido en píxeles entre el punto seleccionado por el usuario y el punto láser.
3. Generar 2 puntos cercanos a dicho punto, para

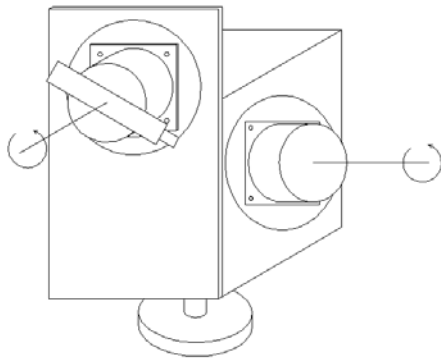


Fig. 4 Unidad Pan-Tilt

4. obtener aproximación del Jacobiano Ec. (1).
4. Con esta aproximación del jacobiano calcular los incrementos de los ángulos ( $\Delta \alpha_1$  y  $\Delta \alpha_2$ ) para posicionar el punto láser en el punto seleccionado por el usuario en espacio de cámara.
5. Desplazar  $\Delta \alpha_1$  y  $\Delta \alpha_2$ .
6. Si el error es menor que el permitido ir a 7, si es mayor ir a 3.
7. Terminar

Para ejecutar el algoritmo anterior cada que se posiciona el láser en un punto se toma una foto con y sin punto láser, se hace la diferencia de imágenes y luego se obtiene el centroide del punto láser, cuyas coordenadas se utilizan en el punto 3 del algoritmo anterior.

#### *Control de posición perpendicular de la herramienta del robot sobre la superficie seleccionada.*

Para lograr que el robot llegue a un punto determinado en espacio tridimensional se requiere obtener las coordenadas

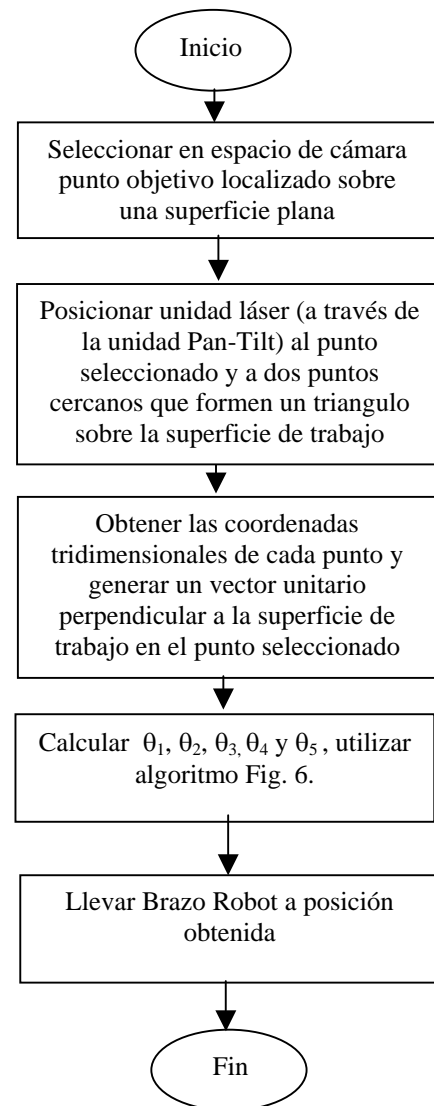


Fig.5 Algoritmo por CSM para posicionamiento de brazo robot a punto seleccionado con orientación perpendicular a la superficie de trabajo

de los  $n$ -grados de libertad del robot. Para ello se implementó en lenguaje C++ un algoritmo que controla todos los dispositivos (robot, cámaras, Unidad Pan-Tilt) y calcula la configuración interna del manipulador para llevar el robot a la posición dada. En el diagrama de flujo de la Fig. 5 se describe la estructura de dicho algoritmo. En la Fig. 6 se muestra el diagrama de flujo para la obtención de los ángulos del Robot ( $\theta_1, \theta_2, \theta_3, \theta_4$  y  $\theta_5$ ), para que la herramienta que se encuentra en el extremo del manipulador se posicione en el punto objetivo y se oriente en forma perpendicular a la superficie de trabajo que lo contiene.

#### Estado actual del proyecto y resultados obtenidos.

En la actualidad se cuenta con los algoritmos y la implementación para el control de la unidad Pan Tilt y la Interfaz Gráfica en forma independiente.

Fue necesario determinar las conexiones y especificaciones para cada encoder y las conexiones para cada motor, así como las relaciones de transformación debido a las reducciones para cada junta, lo cual se hizo con ayuda de la tarjeta controladora de movimiento. Una vez determinadas las características eléctricas, mecánicas y de retroalimentación necesarias del robot, se realizó el control de cada eje y finalmente de los 6 ejes al mismo tiempo, todo ello desde la

terminal de la PC. Actualmente se tiene el control del robot desde la PC a partir de la tarjeta de control de movimiento, los amplificadores, y fuentes de poder, y se tiene el programa para controlar el robot desde la PC. Se cuenta además con el algoritmo para el posicionamiento perpendicular de la herramienta del robot sobre la superficie seleccionada, el cual converge favorablemente.

Al programa general ya se le incluyeron todos los algoritmos anteriores y en este momento se está en la etapa de depuración del programa de posicionamiento punto a punto. Se han realizado pruebas desde el algoritmo principal para las 3 últimas juntas y se está comenzando con las últimas tres. Una vez que se culminen estas acciones, se procederá a probar el algoritmo completo.

Después de ello se incluirán el frenado dinámico, el control de fuentes, la liberación y energización de los frenos electromecánicos, el paro de emergencia debido a la pérdida de alguna señal de retroalimentación (encoders), y la liberación manual desde el robot de los tres primeros ejes a la tarjeta controladora, actualmente algunas de estas operaciones básicas se están realizando en forma manual.

#### IV.- CONCLUSIONES Y TRABAJO FUTURO

Se está integrando un sistema de control por visión CSM a partir de un robot obsoleto, en donde fue escogida la aplicación específica de posicionamiento del brazo manipulador perpendicular a la superficie de trabajo como un primer paso en la exploración de este método. Debido a la naturaleza escalable del este proyecto es que se implementó una arquitectura abierta y modular. Esto debido al desarrollo de nuevos módulos, la prevención de la depreciación de la tecnología, y el soporte para trabajos futuros con antecedentes compatibles.

La finalidad de armar nuestro sistema completo de CSM es la realización de investigación futura en esta misma área. Próximamente se trabajará con la implementación de comunicación vía Internet2 para su operación desde distancias remotas en tiempo real.

#### V. RECONOCIMIENTOS

Para la realización de este proyecto se cuenta con el apoyo del CONACYT, quien proporciona las becas de estudio de maestría a Samantha Luna y Felipe de la Peña. Así también al CUDI y al CONACYT quienes financiaron en parte la compra del equipo.

Se agradece también al Fondo de Apoyo a la Investigación de las UASLP (FAI) por el apoyo recibido para la compra de equipo a través de convenio CO3-FAI-11-10.45.

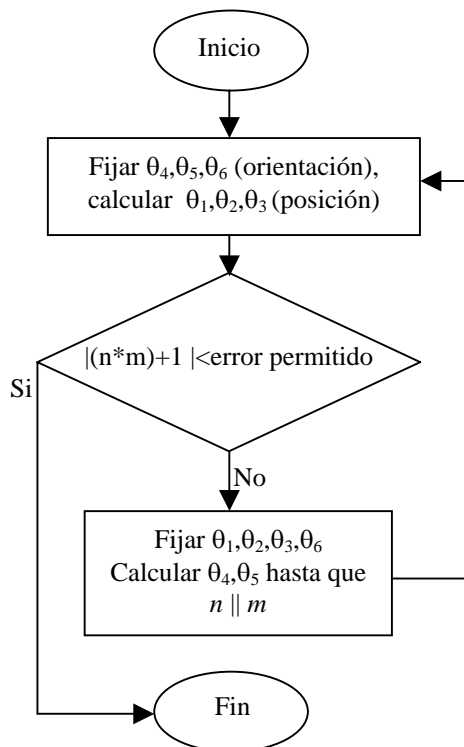


Fig.6 Algoritmo para obtener  $\theta_1, \theta_2, \theta_3, \theta_4$  y  $\theta_5$  con orientación perpendicular a la superficie que contiene el punto seleccionado donde  $n$  es el vector unitario normal a la superficie y  $m$  es el vector unitario que sale de la herramienta

## VI. REFERENCIAS

- [1] Seelinger MJ. "Point-and-Click" Camera-Space Manipulation and some Fundamental Issues regarding the Control of Robots using Vision". *Ph.D. Thesis. University of Notre Dame, Notre Dame IN*, 1999. p. 14-19.
- [2] Skaar SB, Brockman WH, Jang WS, "Three dimensional camera space manipulation". *International Journal of Robotics Research*, 1990, volume 9, no. 4, p. 22-39.
- [3] Goodwine B, Seelinger M, Skaar SB, Qun M. Nonholonomic "Camera space manipulation using Cameras mounted on mobile base". *University of Notre Dame, Notre Dame IN*.
- [4] Allen PK, Timcenko A, Yoshimi B, Michelman P. "Real-time visual servoing". *Proceedings of the IEEE International Conference on Robotics and Automation* 1992. p. 1850-6.
- [5] Seelinger M, Skaar SB, Robinson M. "An alternative approach for image-plane control of robots". A chapter. In: Kriegman DJ, Hager GD, Morse Astephen (editors): *Lecture Notes in Control and Information Sciences*. London: Springer-Verlag, Vol 237. p. 41-65. Limited, 1998
- [6] González-Galván EJ, Pazos-Flores F, Skaar SB, Cárdenas-Galindo A. "Camera pan/tilt to eliminate the workspace-size/pixel-resolution tradeoff with camera-space manipulation". *Robotics and Computer Integrated Manufacturing* 18 (2002) 95-104.
- [7] Horaud R, Dornaika F, Espiau B. "Visually guided object grasping". *IEEE Trans Robot Automat* 1998;14(4):525-32.
- [8] Baumgartner ET, Seelinger MJ, Fessler M, Aldekamp A, Gonzalez-Galvan EJ, Yoder J, Skaar B. "Accurate 3-d robotic point positioning using camera-space manipulation".
- [9] Cessati M, Bovet DP. "Driver for the Galil 1800 PCI Motion Controllers. Shell Technologies 2003. (<http://www.galil.com>)
- [10] Hager GD, Rizzi AA, Cavan NJ. "Driver for the frame grabber image acquisition card DT-3155".
- [11] Miguimi Lukito. "A point-and-click palletizing /depalletizing control system using camera space manipulation". *Master Thesis. Department of Aerospace and Mechanical Engineering. Notre Dame, In*. 2001.
- [12] Skaar, S.B., and Ruoff, C.F. (editors), "Teleoperation and Robotics in Space", Progress in Aeronautics and Astronautics Series, Vol. 161, AIAA: Washington, D.C., 1994.
- [13] Norma Yazmín Covarrubias Glez, "Algoritmo para la correspondencia y segmentación de imágenes con una interfaz en internet", *COMRob 2005, México*.
- [14] Emilio J. González Galván. "Introducción al Curso de Robótica y Visión por Computadora". Notas del curso de Robótica. *CIEP-FI, 2004. México*

---

<sup>1</sup> US Patent 6,304,050 B1; Oct. 2001.  
<sup>2</sup> « CSM » Camera Space Manipulation  
<sup>3</sup> « GUI » Graphic User Interface  
<sup>4</sup> « CCD » Charge-Coupled-Device « Point and Click Camera-Space Manipulation. »-