

CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS AVANZADOS  
DEL  
INSTITUTO POLITÉCNICO NACIONAL  
DEPARTAMENTO DE CONTROL AUTOMÁTICO

## **Visión artificial estero con aplicación al control de un brazo de robot**

TESIS QUE PRESENTA EL:  
**M. en C. Marco Antonio Moreno Armendariz**

PARA OBTENER EL GRADO DE  
**Doctor en Ciencias en la Especialidad de Control Automático**

DIRECTOR DE TESIS:  
**Dr. Wen Yu Liu**  
**Dr. Alexander S.Pozniak**

**México, D.F., septiembre del 2003.**



# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Objetivo de la tesis . . . . .	2
1.2. Contenido de la tesis . . . . .	3
1.3. Publicaciones realizadas . . . . .	5
1.3.1. Artículos en revista . . . . .	5
1.3.2. Publicaciones en congreso . . . . .	5
<b>2. Diseño de la plataforma de estéreo visión</b>	<b>7</b>
2.1. Introducción . . . . .	7
2.2. Componentes . . . . .	8
2.2.1. Robot CRS Robotics A465 . . . . .	8
2.2.2. Cámaras analógicas . . . . .	13
2.2.3. Lentes Cosmimar C815b . . . . .	13
2.2.4. Tarjeta digitalizadora de video . . . . .	14
2.2.5. Software especializado . . . . .	15
<b>3. Visión por computadora</b>	<b>21</b>
3.1. Introducción . . . . .	21
3.2. Análisis de las imágenes y resultados experimentales . . . . .	23
3.2.1. Captura y binarización de las imágenes . . . . .	25
3.2.2. Detección de las marcas visuales . . . . .	26

3.2.3.	Separación de las marcas visuales . . . . .	28
3.2.4.	Obtención del centroide de las marcas visuales . . . . .	29
3.2.5.	Clasificación de las marcas visuales . . . . .	30
3.3.	Eliminación de la distorsión radial . . . . .	31
<b>4.</b>	<b>Modelo óptico de estéreo visión</b>	<b>35</b>
4.1.	Antecedentes . . . . .	35
4.2.	Introducción . . . . .	35
4.3.	Cálculo de los valores requeridos para el modelo . . . . .	36
4.3.1.	Distancia real entre el codo y el órgano terminal del robot A465 . . . . .	37
4.3.2.	Ángulo de rotación de las cámaras . . . . .	37
4.3.3.	Distancia entre las lentes ocular y objetivo . . . . .	38
4.3.4.	Factor de conversión de pixel a centímetro . . . . .	39
4.4.	Cálculo de la posición en el espacio de un punto . . . . .	41
4.4.1.	Etapa 1: Cálculo de la distancia entre las cámaras y las marcas visuales del robot . . . . .	43
4.4.2.	Etapa 2: cálculo de las coordenadas $(x, y)$ del órgano terminal del robot . . . . .	48
4.5.	Cinemática del robot A465 . . . . .	52
4.6.	Cinemática visual del robot A465 . . . . .	54
4.6.1.	Desarrollo y resultados experimentales . . . . .	54
4.7.	Ventajas y Desventajas del modelo . . . . .	56
4.7.1.	Ventajas . . . . .	57
4.7.2.	Desventajas . . . . .	58
<b>5.</b>	<b>Aplicación: control visual difuso</b>	<b>61</b>
5.1.	Introducción . . . . .	61
5.2.	Diseño de sistemas difusos mediante el esquema de matriz difusa . . . . .	63
5.2.1.	Definir los conjuntos difusos para cubrir los espacios de entrada y salida . . . . .	63
5.2.2.	Generar una regla para cada par entrada-salida . . . . .	64
5.2.3.	Asignar un grado a cada regla difusa . . . . .	66

## ÍNDICE GENERAL

III

5.2.4. Crear la base de reglas difusas . . . . .	67
5.2.5. Construcción del sistema difuso . . . . .	68
5.3. Diseño de un controlador difuso . . . . .	68
5.3.1. Diseño de un controlador difuso mediante el enfoque de prueba y error	70
5.4. Diseño del controlador visual difuso . . . . .	71
5.4.1. Introducción . . . . .	71
5.4.2. Desarrollo y resultados experimentales . . . . .	73
<b>6. Análisis de Estabilidad: control visual PD</b>	<b>81</b>
6.1. Introducción . . . . .	81
6.2. Robot de dos grados de libertad . . . . .	82
6.2.1. Preliminares . . . . .	82
6.2.2. Controlador visual estándar . . . . .	86
6.2.3. Controlador visual Proporcional-Derivativo con observador . . . . .	90
6.2.4. Controlador visual Proporcional-Derivativo con Redes Neuronales Mul- ticapa . . . . .	97
6.2.5. Controlador visual Proporcional-Derivativo con observador y redes neuronales multicapa . . . . .	104
6.3. Robot de tres grados de libertad . . . . .	110
6.3.1. Preliminares . . . . .	110
6.3.2. Control visual estándar . . . . .	114
6.3.3. Controlador visual Proporcional-Derivativo con observador . . . . .	116
6.3.4. Controlador visual Proporcional-Derivativo con redes neuronales mul- ticapa . . . . .	119
6.3.5. Controlador visual Proporcional-Derivativo con observador y redes neuronales multicapa . . . . .	123
<b>7. Conclusiones</b>	<b>129</b>
7.1. Trabajo futuro . . . . .	131

<b>A. Modelo de Grosso</b>	<b>141</b>
A.1. Modelo de Grosso . . . . .	141
<b>B. Glosario</b>	<b>145</b>
B.1. Términos de control visual [43] . . . . .	145
B.1.1. Tipos de cámaras de video . . . . .	145
B.1.2. Configuraciones de la cámara . . . . .	145
B.1.3. Tipos de control visual dependiendo del uso de la información visual .	146
B.1.4. Tipos de arquitectura dependiendo del lazo de retroalimentación . . .	146
B.1.5. Tipos de control dependiendo la información que reciba el controlador	147

# Índice de figuras

2.1. Arquitectura de la plataforma de estéreo visión. . . . .	9
2.2. Robot CRS Robotics A465. . . . .	10
2.3. Robot A465 preparado con el sistema de visión por computadora. . . . .	10
2.4. Controlador CRS Robotics C 550C. . . . .	11
2.5. Control manual usado para manejar al robot A465. . . . .	12
2.6. Cámara Pulnix TM-72EX. . . . .	13
2.7. Tarjeta Digitalizadora PCI 1408. . . . .	15
2.8. Módulos del software especializado. . . . .	16
2.9. Intefaz del usuario del software especializado. . . . .	18
2.10. Diagrama de bloques de las funciones del software. . . . .	19
3.1. Robot CRS Robotics A465 con 6 grados de libertad. . . . .	23
3.2. Información a obtener por los algoritmos de visión por computadora: (a) coordenadas $(x, y)$ de las marcas visuales; (b) ángulos de las cámaras con respecto al eje $x$ del robot. . . . .	24
3.3. Algoritmos adoptados para el análisis de las imágenes. . . . .	24
3.4. Imagen binaria de 640 x 480 pixeles. . . . .	25
3.5. Resultado del proceso (a) Imagen redimensionada a 91 x 68 pixeles. (b) lista de valores que representan a las marcas visuales. . . . .	27
3.6. Resultado de realizar la separación en objetos. . . . .	28
3.7. Obtención del centroide de los puntos visuales, usando los bordes del círculo. . . . .	29
3.8. Lista final con el centroide obtenido para las marcas visuales. . . . .	30

3.9.	Diferentes posiciones del robot A465 donde la condición se cumple y permite libre movimiento.	30
3.10.	Resultado de realizar la clasificación de las marcas visuales.	31
3.11.	Espacio de trabajo para el robot A465.	32
3.12.	Malla para detectar la distorsión radial. con distorsión (izquierda) y sin distorsión (derecha).	32
4.1.	El robot A465 con sus 6 grados de libertad (a). Los ángulos $\theta_{1a}$ y $\theta_{1b}$ de las cámaras con respecto al eje x del robot (b).	37
4.2.	Procedimiento para calcular el ángulo de rotación de las cámaras.	38
4.3.	Lente Cosmimar C815b (a). Diagrama esquemático de los lentes (b).	39
4.4.	Vistas del riel fabricado para obtener el factor de conversión.	40
4.5.	Experimento diseñado para obtener el factor de conversión.	41
4.6.	Se desea localizar la marca visual $c_{des}$ cuando el órgano terminal del robot está en otra posición diferente a la de inicio.	42
4.7.	El modelo geométrico se construye con la distancias entre las cámaras y dos marcas visuales del robot A465.	43
4.8.	Puntos visuales $c_1$ , $c_2$ y la distancia entre ellos.	44
4.9.	La distancia $c_{proy}$ se obtiene de proyectar la distancia real entre los marcas $c_1$ y $c_2$ en el eje y.	44
4.10.	Primer parte del modelo geométrico para obtener las distancias de la cámara izquierda $e_1$ , $e_2$ .	46
4.11.	Primer parte del modelo geométrico para obtener las distancias de la cámara derecha $f_1$ , $f_2$ .	47
4.12.	El robot se mueve a una nueva posición (diferente de la posición de Ready), por lo que ahora la distancia $c'$ es desconocida.	48
4.13.	Segunda parte del modelo geométrico para el cálculo de las coordenadas $(x, y)$ del punto desconocido.	51
4.14.	Diagrama de los ejes coordenados para las articulaciones y marcas visuales del robot A465.	52
4.15.	Ubicación de las marcas visuales en el robot A465.	54
4.16.	Ángulos $\theta_2$ y $\theta_3$ a ser calculados visualmente.	55
4.17.	Movimiento realizado por la segunda articulación del robot durante el experimento.	56

4.18. Resultados obtenidos de calcular el ángulo $\theta_3$ mediante información visual. . . . .	57
5.1. Configuración básica de los sistemas difusos. . . . .	62
5.2. Ejemplo de funciones de pertenencia para pares entrada-salida en el caso de dos entradas. .	65
5.3. Matriz difusa generada por los conjuntos difusos de las variables de entrada que será llenada con la base de reglas difusas. . . . .	67
5.4. Control difuso versus Control convencional. . . . .	69
5.5. Sistema original formado por el robot A465 y su controlador PID articular. . . . .	72
5.6. Esquema general de sistema de visión por computadora con el controlador visual difuso. .	72
5.7. Las variables de control son los ángulos de las primeras tres articulaciones $(\theta_1, \theta_2, \theta_3)$ . .	73
5.8. Esquema desglosado del controlador visual difuso. . . . .	74
5.9. Conjuntos difusos para las variables $e$ , $\Delta e$ y $\gamma$ . . . . .	77
5.10. Matriz difusa llenada con las reglas difusas SI-ENTONCES. . . . .	78
5.11. Desplazamientos del robot A465 para llegar a la posición deseada. . . . .	79
6.1. Diagrama de flujo para diseñar el observador. . . . .	94
6.2. Estructura del controlador visual tipo PD con compensación y observador. . . . .	104
A.1. Geometría del modelo propuesto por Grosso y sus colegas. . . . .	142



# Capítulo 1

## Introducción

En tiempos recientes se ha llevado a cabo la automatización de diferentes tipos de procesos como los industriales o los administrativos. Esto trae como beneficio la reducción del tiempo, gracias al avance en los sistemas de cómputo y al surgimiento de nuevas tecnologías de computación inteligente como los sistemas de visión por computadora y la lógica difusa entre otras. Estos nuevos desarrollos han facilitado el trabajo del ser humano haciéndolo más eficiente.

En el presente trabajo se hará uso de un sistema de visión por computadora y de la lógica difusa. Existen diversos tipos de sistemas de visión por computadora, catalogados por el número y tipo de sensores. Principalmente se componen de cámaras de video especializadas encargadas de realizar la adquisición de las imágenes y de un conjunto de algoritmos cuyo objetivo es extraer de éstas la información necesaria para el proceso a realizar.

Actualmente la mayoría de los sistemas de visión por computadora que se emplean se encuentran instalados en las plantas manufactureras para efectuar tareas de control de la calidad de los productos, ésto se puede llevar a cabo mediante algoritmos de conteo y detección de patrones, los cuales son seleccionados por un experto de acuerdo a las necesidades de cada proceso. Para este tipo de actividades es suficiente utilizar una sola cámara o sensor ya que no es necesario conocer la ubicación de un objeto en el espacio.

Por otra parte existen tareas como en una línea de producción automatizada donde un

conjunto de robots elaboran un producto en etapas. Para ello, estos deben ser previamente programados con rutinas específicas que ejecutarán repetidamente. Sin embargo, si se deseara construir una celda de manufactura flexible donde los robots tuvieran la capacidad de hacer diversos tipos de ensamblajes sin la necesidad de tener que reprogramarlos cada vez, se podría realizar incorporando a éstos algunas de las tecnologías de computación inteligente como las mencionadas anteriormente, con el fin de brindar mayor flexibilidad a este tipos de mecanismos.

Por ejemplo, para los sistemas de visión por computadora que usen la configuración de cámara fija para robótica (ver glosario, pp. 143) una de las tareas mas comunes a efectuar es obtener la posición del órgano terminal<sup>1</sup> del robot en el espacio de trabajo mediante la información visual. El valor obtenido de la posición se incorpora en el lazo de control del sistema para que éste lleve a cabo la tarea asignada.

En las plantas manufactureras se han automatizado muchos procesos principalmente mediante el uso de autómatas, ya sea de manipulación de objetos, de ensamble o transporte de material; esto gracias a la reducción de los tiempos en los procesos de operación. Los avances realizados en la ingeniería han permitido la creación de manipuladores confiables y muy precisos.

## 1.1. Objetivo de la tesis

El objetivo de este trabajo se divide en dos partes:

1. Demostrar la estabilidad en el sentido de Lyapunov de un robot manipulador en lazo cerrado con retroalimentación visual considerando cuatro casos que comprenden diferentes incertidumbres en el sistema.
2. Desarrollar un nuevo modelo óptico de estereó visión que sea el encargado de obtener la posición en el espacio del órgano terminal del robot manipulador. Como prueba de

---

<sup>1</sup>*gripper*, por su término en inglés.

desempeño se diseñó un controlador visual estático difuso que será mostrado a nivel experimental para el caso de control de posición.

Para realizar los experimentos se diseñará y implementará una plataforma de visión por computadora en el laboratorio de Robótica de la Escuela de Ingeniería de la Universidad La Salle<sup>2</sup>. Esta plataforma utilizará la arquitectura *ver y mover*<sup>3</sup> basada en la imagen con la configuración de cámara fija (ver glosario, pp. 143).

## 1.2. Contenido de la tesis

En el **Capítulo 2**, se presentará la primera **contribución** que consiste en el diseño e implementación de una nueva plataforma de visión estéreo por computadora, un robot manipulador de seis grados de libertad y un nuevo software proyectado en forma modular que incorpora: la comunicación serial con el robot, la captura de las imágenes y todos los algoritmos presentados en esta tesis.

En el **Capítulo 3**, se presentará una segunda **aportación** con, un nuevo conjunto de algoritmos de visión por computadora. Para hacer mas fácil la comprensión de su funcionamiento se incorporará un ejemplo obtenido experimentalmente usando la plataforma de estéreo visión. El diseño de los algoritmos busca tener un adecuado balance entre la precisión del valor de los datos que se extraeran de las imágenes y su velocidad de procesamiento. Este se llevará a cabo en dos etapas:

1. Diseño y simulación de los algoritmos en matlab<sup>®</sup>. Una vez que se cumpla con el objetivo planteado se pasará la segunda etapa.
2. Se programarán los algoritmos obtenidos en lenguaje Borland C++<sup>®</sup>. Se realizarán pruebas en la plataforma de estéreo visión para garantizar el buen funcionamiento de éstos. Se crearán rutinas de despliegue de la información extraída de las imágenes y del controlador articular del robot para monitoriar su funcionamiento.

---

<sup>2</sup>se encuentra ubicada en Benjamin Franklin 47, Col. Condesa, C.P. 06140, México D.F., Tel. 5278-9500

<sup>3</sup>*look and move*, por su término en inglés.

En el **Capítulo 4**, se presentará la **contribución principal**: el modelo óptico de estéreo visión. Éste se diseñará con la filosofía de ser sencillo, eficiente y con el propósito de poder ser utilizado en diversas aplicaciones en el campo de visión por computadora. Aquí se presentarán varias aplicaciones en el campo de la robótica.

Se efectuará el estudio telescópico de la lente compuesta y se implementará el algoritmo de eliminación de la distorsión presentando en [10]. Esta distorsión es generada por una lente con un ángulo de visión muy grande, la cual fue seleccionado, pues, se requiere que aparezca en las imágenes el robot completo y la mayor parte de su espacio de trabajo.

Además, como se explicará mas adelante el modelo óptico de estéreo visión requiere conocer el valor del ángulo de rotación de las cámaras con respecto al *eje x* del robot, por lo que se incorporará un algoritmo para calcularlos.

En el **Capítulo 5**, se presentará la metodología de diseño para controladores difusos y se definirá el concepto de controlador visual difuso. Otra **aportación** de esta tesis será la utilización de esta metodología para diseñar e implementar un nuevo controlador de este tipo en la plataforma de estéreo visión. Finalmente se mostrarán los resultados de los experimentos realizados.

En el **Capítulo 6**, se presentará última **contribución** que comprende, el diseño de tres nuevos controladores visuales para un robot planar (dos dimensiones) y su análisis de estabilidad. Estos se diseñarán para los siguientes casos:

- **Caso 1:** Se desconocerá el valor de la velocidad de las articulaciones del robot, por lo que se usará un observador de modos deslizantes para estimarlo. Una vez obtenido este valor se incluirá en la ley de control visual.
- **Caso 2:** Se usará una red neuronal multicapa para identificar el valor de la gravedad y la fricción. Además, se considerará la existencia de incertidumbres en la matriz Jacobiana.
- **Caso 3:** Este controlador tratará las problemáticas planteadas en los dos casos anteriores.

En todos los casos el análisis de estabilidad que se realizó fué de tipo Lyapunov; con el cual, se asegura la convergencia del sistema a una región acotada de manera local.

Este trabajo está dirigido a aquellas personas interesadas en la investigación en visión artificial y robótica con conocimientos de robótica básica, programación estructurada y programación lineal.

## 1.3. Publicaciones realizadas

En general, se puede decir que en esta tesis se obtuvo una contribución en el área de control visual. Las publicaciones realizadas son:

### 1.3.1. Artículos en revista

1. **PD-like visual servoing with velocity estimation and uncertainties compensation**, *Journal of Dynamic Systems, Measurement, and Control*. Se está en espera de su aceptación.

### 1.3.2. Publicaciones en congreso

1. **A Fuzzy Controller Based on Computer Vision**, Marco A. Moreno-Armendariz, Eduardo Gómez-Ramírez, Wen Yu and Alexander S. Poznyak, “International Conference on Computer, Communication and Control Technologies: CCCT '03”, Orlando, Florida, 31 de julio al 2 de agosto del 2003.
2. **An Optical Stereo Vision Model**, Marco A. Moreno-Armendariz, Julian Aguilar-Luna, Eduardo Gómez-Ramírez and X. Vilasis-Cardona, “3rd IASTED International Conference on Circuits, Signals, and Systems”, Cancun, México, 19 al 21 de Mayo del 2003.
3. **A New Visual Algorithms for Stereo Vision in Robotics**, Marco A. Moreno, E. Gómez-Ramírez, Wen Yu & A.S. Poznyak, “Congreso Latinoamericano de Control Automatico”, Guadalajara, Jalisco, México, 4 al 6 de diciembre del 2002.

4. **An Auto-calibration Stereo Vision for Robotics**, Marco A. Moreno, E. Gómez-Ramírez, Wen Yu & A.S. Poznyak, “International Symposium on Robotics and Automation”, Toluca, Edo. de México, 1 al 4 de septiembre del 2002.
5. **Stable 3-D Visual Servoing: An Experimental Comparison**, M.A. Moreno Armendariz, Wen Yu & A.S. Poznyak, “2001 IEEE Conference on Control Applications(CCA) and the International Symposium on Intelligent Control(ISIC) ”, México D.F., México, del 5 al 7 de Sep. de 2001.
6. **A Simple Algorithm for Stereo Vision in Robotics**, E. Gómez-Ramírez, M. A. Moreno & A.S. Poznyak, “VI Iber-American Symposium on Pattern Recognition”, Florianopolis, Brasil, 12 al 14 de octubre del 2001.
7. **Visual Robot Kinematics Based on Stereo Vision**, Marco A. Moreno-Armendariz, E. Gómez-Ramírez, Wen Yu & A.S. Poznyak, “IASTED Robotics and Applications”, Clearwater (Tampa), Florida, 19 al 22 de noviembre del 2001.
8. **Stability Analysis of 3-D Visual Servoing**, Wen Yu & M. A. Moreno Armendariz, “International Symposium on Robotics and Automation”, Monterrey, N.L, México, del 10 al 12 de Noviembre del 2000.

# Capítulo 2

## Diseño de la plataforma de estéreo visión

### 2.1. Introducción

Una plataforma de visión por computadora consiste en incorporar a un sistema dinámico un sensor externo, como lo es una cámara de video encargada de obtener información visual que sea usada para controlar al sistema. Esta técnica es conocida así ya que incorpora al sistema dinámico mayor flexibilidad y la capacidad de obtener información de su entorno.

La visión por computadora utiliza las imágenes capturadas por las cámaras y la procesa mediante un conjunto de algoritmos diseñados específicamente para extraer la *información relevante* [64]. Ésta se usa para manipular al sistema dinámico y realizar la tarea asignada.

El incremento en la velocidad de procesamiento de las computadoras y su bajo costo han permitido el desarrollo de nuevas plataformas de visión cuyo diseño tiene como objetivo tener una respuesta de control rápida y un tiempo de retardo pequeño [7].

Para tener tiempos de procesamiento rápidos, las primeras plataformas de visión por computadora se basaban en computadoras muy costosas y con equipo muy especializado [54] o con los sistemas operativos desarrollados específicamente para realizar una tarea [4], [9], [31], [37], [54]. Estas utilizan un solo sensor externo (cámara de video) con la cual, es

posible trabajar en dos dimensiones.

Una plataforma de estéreo visión incorpora un par de sensores externos al sistema (cámaras de video) por lo que es posible trabajar en tres dimensiones. Esto permite realizar otro tipo de tareas como calcular la ubicación de un punto en el espacio o efectuar la reconstrucción de objetos.

En este Capítulo se presenta el diseño de la plataforma de estéreo visión diseñada para implementar los algoritmos de visión por computadora, modelo de estéreo visión y control visual difuso. Esta plataforma está formada por:

1. Un robot CRS Robotics A465.
2. Dos cámaras analógicas PULNIX.
3. Una tarjeta digitalizadora National Instruments.
4. Una computadora.
5. Un software especializado.

## 2.2. Componentes

En la Figura 2.1 se muestra la plataforma de estéreo visión diseñada. A continuación se describe cada uno de sus componentes:

### 2.2.1. Robot CRS Robotics A465

El Robot CRS Robotics A465 [63], [17], [18] es un brazo mecánico con seis grados de libertad como se muestra en la Figura 2.2. Las dimensiones del robot se asemejan a las del brazo humano en lo referente a proporción. El robot A465 se puede manejar utilizando el lenguaje *RAPL-II*<sup>1</sup>.

---

<sup>1</sup>*Robot Automation Programming Language - II*, por sus siglas en inglés.

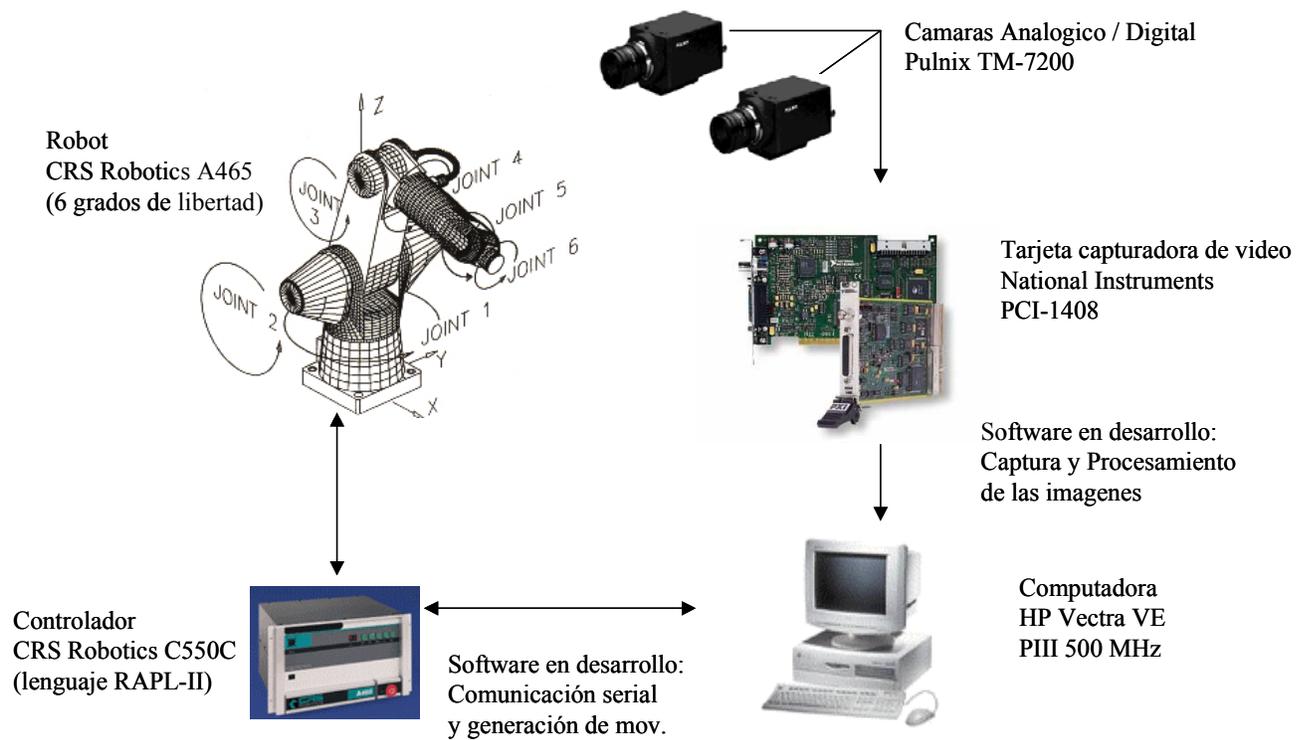


Figura 2.1: Arquitectura de la plataforma de estereo visión.

Cada grado de libertad se encuentra basado en Servomotores con sensores ópticos que pueden llevar al órgano terminal a las coordenadas u orientaciones definidas por el usuario. El controlador permite la manipulación del robot ya sea en coordenadas absolutas con respecto a su espacio de trabajo o en coordenadas relativas con respecto a su posición, así como la manipulación de cada grado de libertad. Para que el robot pueda cumplir su tarea se le debe instalar un órgano terminal adecuado a labor que vaya a realizar (ver Figuras 2.2, 2.3).



Figura 2.2: Robot CRS Robotics A465.

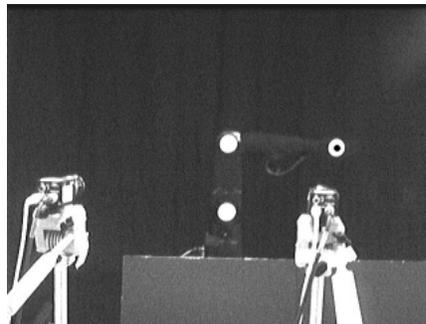


Figura 2.3: Robot A465 preparado con el sistema de visión por computadora.

El robot A465 puede ser operado por una consola de operación<sup>2</sup> vía comunicación serial

---

<sup>2</sup>*Teach Pendant*, por su término en inglés.

o por medio de un sistema de cómputo (PC) a través de alguna de las dos configuraciones de transmisión serial que el controlador tiene ya definidas [17].

### Controlador C 550C

Es el sistema que se encarga de la coordinación de los movimientos de articulaciones del robot. Manipula y calcula su cinemática y su dinámica por medio de un controlador C 550C con algoritmos de control tipo *PID*, en conjunto con un procesador INTEL 80286 con coprocesador matemático 80287 que soporta el lenguaje de programación RAPL-II [63], [18].

Contiene 256 KB de memoria RAM que le permiten almacenar variables y programas; tiene la capacidad de ejecutar programas y realizar el cálculo de trayectorias a través de diversos algoritmos. Este controlador es capaz de comunicarse con un sistema de cómputo personal por medio de una interfaz de comunicación serial Dual RS232 a una velocidad configurable estándar de 38400 BPS para ampliar la manipulación del robot.

Este controlador (ver Figura. 2.4) posee dieciséis entradas y doce salidas de datos opto-aisladas, las cuales pueden expandirse por medio del uso de un PLC y es capaz de manejar hasta ocho ejes simultáneamente.



Figura 2.4: Controlador CRS Robotics C 550C.

### Consola de operación

Es un dispositivo [63] que se conecta al controlador por medio de un cable serial de tres metros de longitud, posee una pantalla tipo LCD de despliegue de cuatro filas por

veinte caracteres diseñada para mostrar los mensajes del controlador al usuario del robot. Cuenta con un interruptor de activación y un botón de paro de emergencia. Desde aquí es posible la manipulación del robot, la fijación de puntos para generación de trayectorias y la configuración del controlador (ver Figura. 2.5) .



Figura 2.5: Control manual usado para manejar al robot A465.

### **Cables umbilicales**

Son los que permiten la alimentación de los motores al robot y la retroalimentación de los sensores hacia el controlador.

### **Órgano terminal Neumático**

Es un actuador neumático que acoplado a una bomba de aire, produce la apertura o cierre del dispositivo manipulador.

### **Lenguaje RAPL-II**

El robot A465 utiliza el lenguaje de programación RAPL-II. Este lenguaje permite por medio de instrucciones simples, la manipulación del robot, ya sea de cada grado de libertad

o movimientos más elaborados para traslación del órgano terminal en coordenadas absolutas o relativas a su posición. También es usado en los programas de control y manipulación del robot desarrollados para esta investigación [63].

La compañía CRS Robotics es propietaria del lenguaje, usa el sistema llamado *English-Like* para generar comandos de movimiento del robot, así como la capacidad para incluir expresiones matemáticas en la sintaxis de los comandos, su manipulación y programación es muy similar a la del lenguaje C [17], [19].

### 2.2.2. Cámaras analógicas

Se utiliza las cámaras analógicas Pulnix TM-72EX [55] (ver Figura. 2.6), su característica principal es que son Dispositivos de Acople por Carga<sup>3</sup> de alta eficiencia de 2/3 de pulgada de diámetro, que capturan imágenes de alta resolución a través de la utilización de micro lentes. Pueden alcanzar resoluciones de hasta 768(H) x 493(V) pixeles en el formato de captura (RS-170) y pueden capturar imágenes de 1/60 seg. Poseen la capacidad de realizar ajustes de corrección de ganancia, gama y selección de cuadro .



Figura 2.6: Cámara Pulnix TM-72EX.

### 2.2.3. Lentes Cosmimar C815b

Las características de estos lentes son:

---

<sup>3</sup>CCD, por su siglas en inglés.

Especificaciones			
Distancia Focal		8,5 mm.	
Proporción máxima de apertura		1 : 1,5	
Rango del iris		$F,15 \sim \text{Close}$	
Método de operación	Iris	Manual	
	Foco	Manual	
	Zoom	No tiene	
Distancia mínima al objeto		0,2 m.	
Rango de temperatura		$-20^{\circ}C \sim +50^{\circ}C$	
Dimensiones		$\varnothing 42 \times 40$ cm.	
Peso		120 g.	
Formato		$\frac{2}{3}$	

Estos lentes tienen la capacidad de cubrir áreas grandes (1.5 m x 2 m aprox.) en distancias razonables (2 m aprox.). Debido a que se desea que aparezca en las imágenes el robot completo y la mayor parte de su espacio de trabajo. Se eligió este modelo pues permite cubrir adecuadamente toda el área (ver mas detalles en el capítulo 4).

#### 2.2.4. Tarjeta digitalizadora de video

La tarjeta digitalizadora de video<sup>4</sup> utilizada, es la tarjeta PCI - 1408 de la compañía National Instruments (Figura 2.7). Esta tarjeta es monocromática con cuatro entradas de video y una resolución espacial de 1 Mega pixel [50].

Las imágenes obtenidas por medio de la tarjeta son convertidas a mapas de bits, en matrices. Estas matrices son guardadas en buffers de información y manipuladas por el programa de control y comunicación al robot.

<sup>4</sup>*Frame Grabber*, por su término en inglés.



Figura 2.7: Tarjeta Digitalizadora PCI 1408.

### 2.2.5. Software especializado

En esta sección se describe los módulos que componen el software de visión por computadora. El esquema general se muestra en la Figura 2.8.

A continuación se describen cada uno de estos módulos:

1. **Tratamiento de imágenes:** Incluye todos los algoritmos descritos en el Capítulo 2 y el algoritmo encargado de obtener los ángulos de las cámaras del Capítulo 3, hace uso del bloque de comunicación con las cámaras. Recibe las imágenes del robot y regresa la posición de las marcas visuales en las imágenes.
2. **Modelo óptico de estéreo visión:** Contiene todas las ecuaciones del modelo óptico de estéreo visión del Capítulo 3, hace uso del bloque de tratamiento de imágenes para obtener los valores de entrada que requiere. Toma la posición de las marcas visuales en las imágenes y regresa las coordenadas del órgano terminal del robot en el espacio.
3. **Comunicación con el robot y las cámaras:** Se encarga de enviar las instrucciones generadas al robot A465 mediante el puerto serial. Contiene un módulo donde se codificó las instrucciones usadas del lenguaje para el robot llamado RAPL-II. Otro más se comunica con la tarjeta capturadora de *National Instruments* para recibir en memoria las imágenes capturadas por las cámaras.
4. **Cinemática visual:** Tiene cargada la cinemática del robot A465, presentada en el

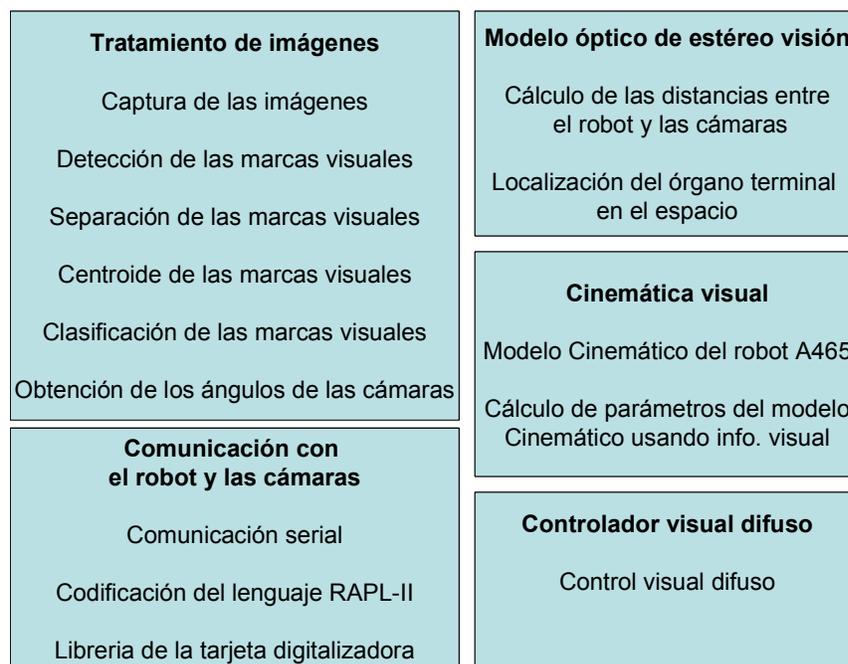


Figura 2.8: Módulos del software especializado.

Capítulo 2. Además, tiene un rutina que hace uso del modelo óptico de estereo visión para obtener la posición en el espacio de las tres marcas visuales que recibe información visual y regresa el valor de los ángulos  $\theta_1$  y  $\theta_2$ , que forman parte de modelo cinemático del robot.

5. **Controlador visual difuso:** Se encarga de mover al robot a la posición deseada indicada por el usuario. Contiene un módulo encargado de convertir la posición deseada a valores angulares para las articulaciones del robot. Con estos ángulos el módulo del controlador visual difuso reciben las imágenes del robot, calcula la posición actual del órgano terminal y realiza la misma conversión a ángulos. Usando estos datos se obtiene el valor del error entre el ángulo actual y el deseado para cada articulación y se genera la acción de control.

El software especializado fue escrito en el lenguaje Borland C++<sup>®</sup>. La interfase del usuario se muestra en la Figura 2.9.

En la Figura 2.10 se indican el nombre de los archivos que contienen un conjunto de funciones encargadas de realizar una tarea específica.

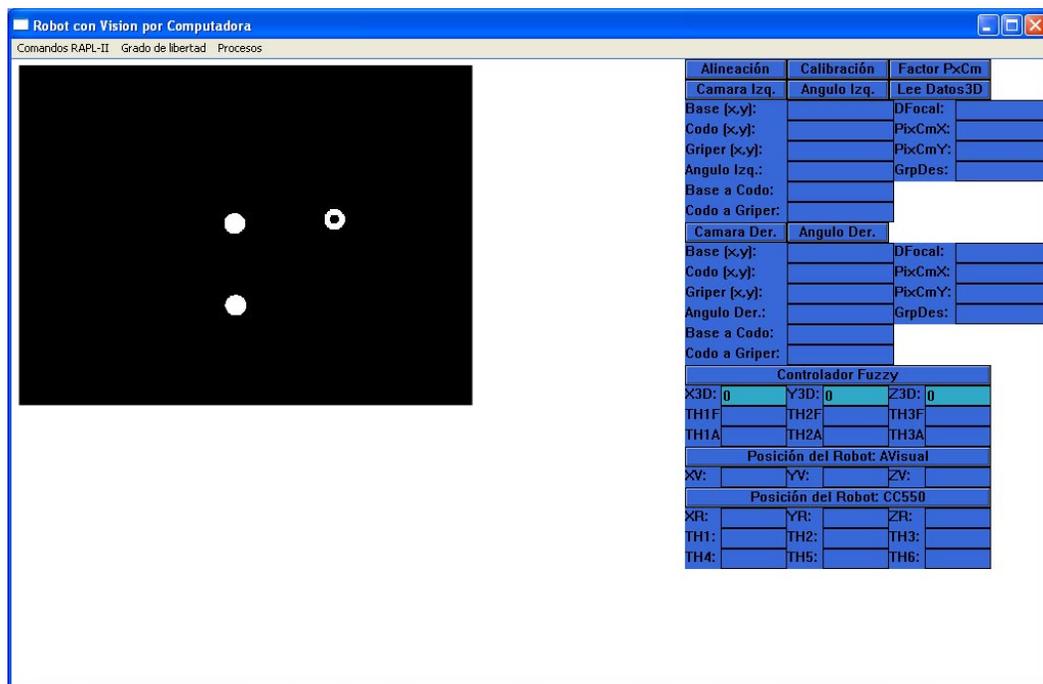


Figura 2.9: Intefaz del usuario del software especializado.

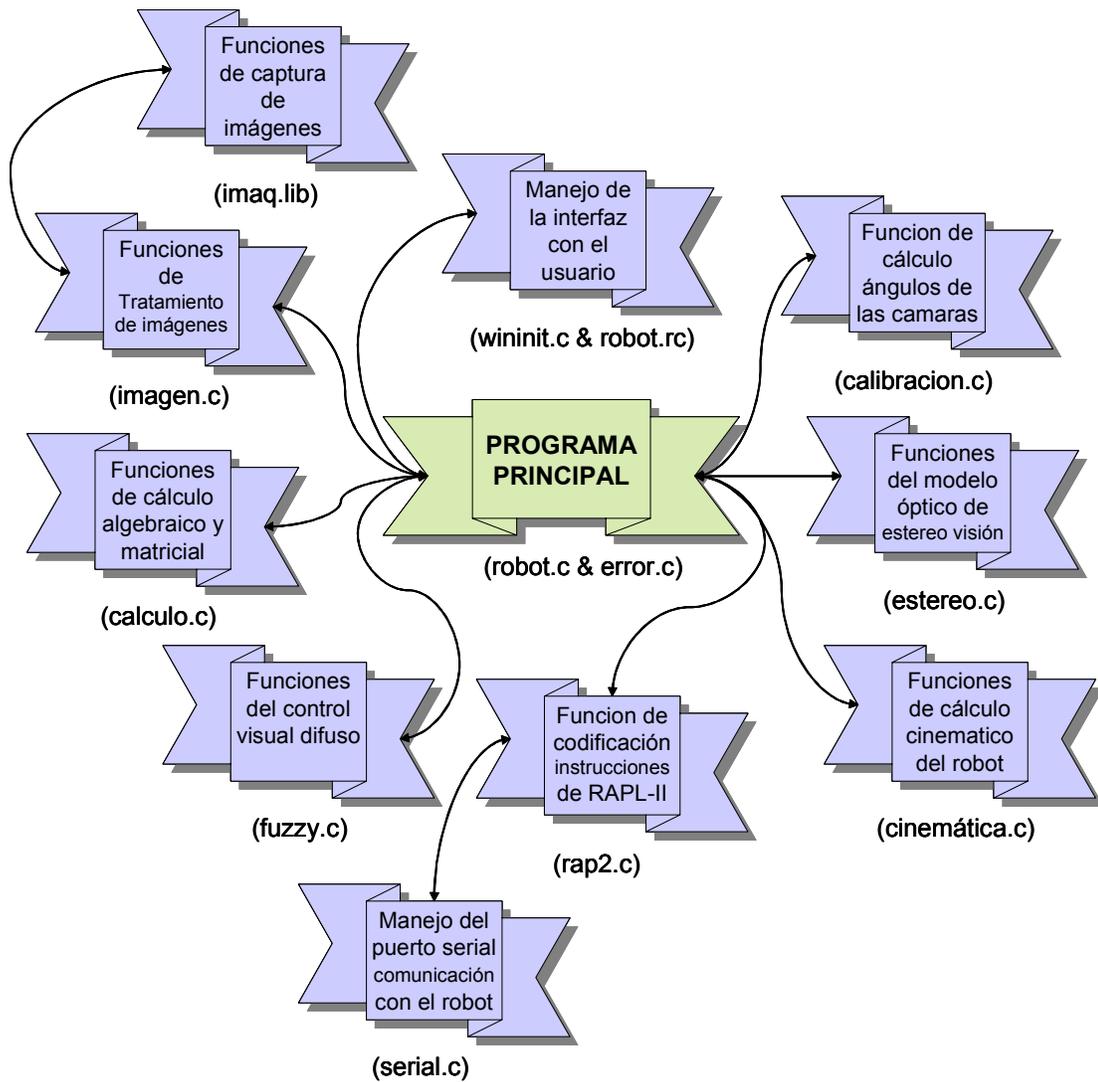


Figura 2.10: Diagrama de bloques de las funciones del software.



# Capítulo 3

## Visión por computadora

### 3.1. Introducción

El término “visión por computadora”, dentro del campo de la Computación Inteligente, puede considerarse como el conjunto de todas aquellas técnicas y modelos que nos permitan el procesamiento, análisis y explicación de cualquier tipo de información espacial obtenida a través de imágenes digitales [68]. Desde sus inicios, ha inspirado sus desarrollos en el estudio del sistema visual humano, el cual sugiere la existencia de diferentes tipos de tratamiento de la información visual dependiendo de metas u objetivos específicos, es decir, la información visual percibida es procesada en distintas formas con base en las características particulares de la tarea a realizar, por lo que propone varias técnicas que permiten obtener una representación del mundo a partir del análisis de imágenes obtenidas desde cámaras de video.

Debido a que la información visual es una de las principales fuentes de datos del mundo real, resulta útil el proveer a una computadora digital del sentido de la vista (a partir de imágenes tomadas con cámaras digitales o analógicas), que junto con otros mecanismos como el aprendizaje hagan de ésta una herramienta capaz de detectar y ubicar objetos en el mundo real, objetivo principal de la visión por computadora.

Antes de presentar los algoritmos de visión por computadora, se explicarán algunos conceptos importantes que se mencionarán durante todo el capítulo.

Una *imagen digital monocromática de dimensión 2* es una función tal que  $f : \mathbb{Z}^2 \rightarrow \mathbb{Z}$ . En la práctica se usan los siguientes tipos de imágenes:

1. **Imagen binaria:** Contiene sólo información de brillo, no de color y se conocen como imágenes de un bit por pixel. Así sólo se tiene el valor uno para representar al(los) objeto(s) y un valor cero para el fondo de la imagen. Este tipo de imágenes se usan en aquellas aplicaciones donde la información del contorno es importante (ejemplo OCR<sup>1</sup>), en el caso de que haya objetos con intensidad o brillo contrastantes (ejemplos: Robótica, conteo de objetos).
2. **Imagen en tonos de gris o monocromática:** Al igual que las binarias sólo contienen información de brillo, sin embargo el número de bits usados para cada pixel varía dependiendo de la cantidad de tonos de gris que se desea representar (ejemplo: para usar 256 tonos se requiere de 8 bits). Este tipo de imágenes se usan para aplicaciones en astronomía, biología, control de calidad, etc.
3. **Imágenes a color:** Se describen por 3 imágenes monocromáticas, cada una sintonizada en una banda (rojo, verde y azul). Este tipo de imágenes se utilizan en aquellas aplicaciones donde la información de color es indispensable (ejemplo: control de calidad en la producción de fármacos).
4. **Imagen multi-espectral:** Es una generalización de las imágenes a color, este tipo se puede modelar por una familia de imágenes monocromáticas (una para cada valor del rango de interés del espectro, como lo son por ejemplo el infrarrojo, el ultravioleta, los rayos X, el radar y el sonar). Se emplean en aquellas aplicaciones donde se requiere obtener información de objetos no visibles a simple vista, por ejemplo: el conjunto de imágenes tomadas por un satélite.

---

<sup>1</sup>*Optical Character Recognition*, por sus siglas en inglés.

## 3.2. Análisis de las imágenes y resultados experimentales

Comúnmente en este tipo de aplicaciones, el análisis consiste en seleccionar o diseñar un conjunto de algoritmos de tratamiento de imágenes *ad hoc* con el fin de extraer la información que se desea.

Para esta tesis el objetivo del análisis de las imágenes es obtener la información necesaria para el algoritmo de estereo visión que se presenta en el siguiente capítulo. Para mejor comprensión de estos algoritmos en cada paso se da una explicación de como funciona cada algoritmo, junto con un ejemplo de su funcionamiento. Este ejemplo, se calcula a partir de los resultados experimentales.



Figura 3.1: Robot CRS Robotics A465 con 6 grados de libertad.

Debido a que el algoritmo de estereo visión se implementará en el robot A465 (ver Figura 3.1), la información que se necesita es: las coordenadas  $(x, y)$  de las *marcas visuales* colocadas en la base, codo y el órgano terminal del robot A465 y los ángulos de rotación de las cámaras con respecto al *eje x* (ver Figura 3.2).

Para realizar esta tarea de forma más sencilla, las marcas visuales serán círculos blancos y el resto de la escena estará en color negro.

Para cumplir el objetivo planteado se muestra en la Figura 3.3 el diagrama general del sistema de análisis de las imágenes diseñado en este trabajo.

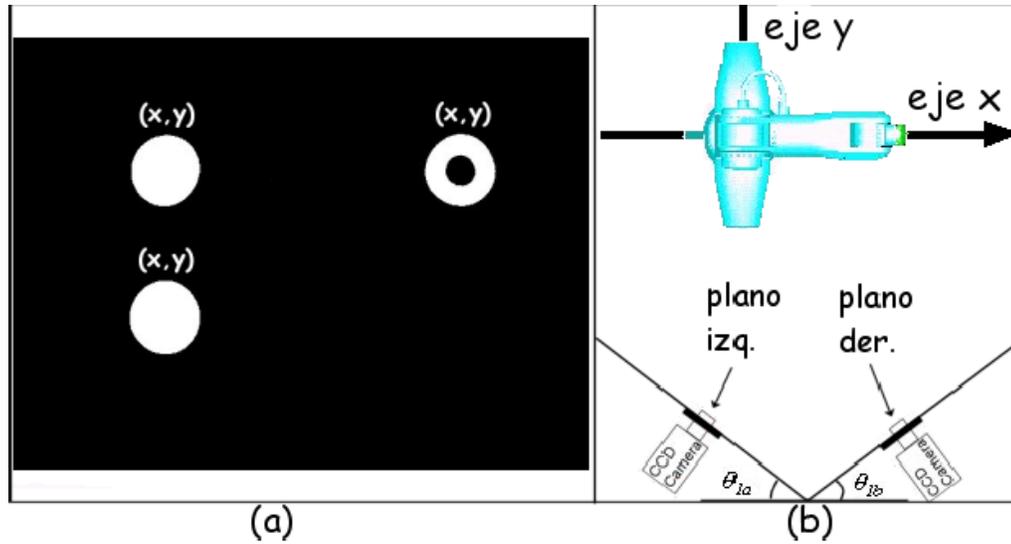


Figura 3.2: Información a obtener por los algoritmos de visión por computadora: (a) coordenadas  $(x, y)$  de las marcas visuales; (b) ángulos de las cámaras con respecto al eje  $x$  del robot.

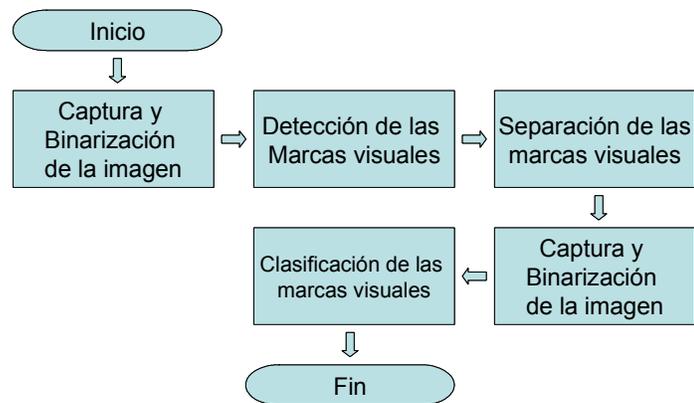


Figura 3.3: Algoritmos adoptados para el análisis de las imágenes.

### 3.2.1. Captura y binarización de las imágenes

El proceso de análisis de las imágenes comienza con la captura de éstas, para esto se utiliza el sistema de visión por computadora formado por: un par de cámaras analógicas Pulnix, un par de lentes Cosmicar C815B y un Tarjeta digitalizadora de National Instruments (para más detalles, ver el capítulo anterior).

Las cámaras se encargan de capturar las imágenes. A éstas, la tarjeta digitalizadora le aplica un proceso conocido como *binarización* en donde se fija un valor de *umbral de binarización* (ejemplo: umbral = 100), aquellos pixeles que sobrepasen este valor o tono de gris se representan con el valor uno y los restantes con el valor cero y de esta forma se obtiene una imagen binaria.

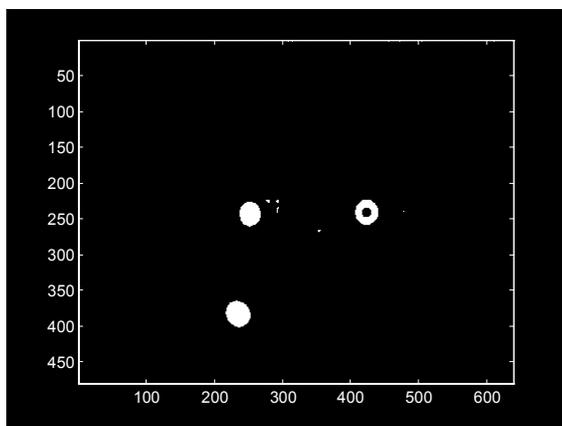


Figura 3.4: Imagen binaria de 640 x 480 píxeles.

Para esta aplicación se fijó el valor del umbral = 160. Se llegó a este valor realizando una serie de experimentos con el sistema de visión por computadora a diferentes horas del día para considerar las variaciones en las condiciones de iluminación en las que se encuentra el robot en el laboratorio. Dado que las variaciones se encuentran en un rango pequeño, el proceso de binarización funcionó adecuadamente durante todos los experimentos.

Es importante aclarar que si las condiciones de iluminación variaran demasiado se requeriría contar con un valor de umbral variable que daría mayor flexibilidad al sistema.

Este tipo de análisis no es considerado en esta tesis, sin embargo, existen trabajos que han resuelto exitosamente este problema como el presentado en [59].

El proceso de binarización lo lleva a cabo automáticamente La tarjeta digitalizadora de video, por lo que el software de visión por computadora recibe directamente la imagen binaria (ver Figura 3.4).

### 3.2.2. Detección de las marcas visuales

Para este proceso se utiliza como entrada la imagen binaria. El objetivo en esta parte es obtener la ubicación donde se encuentran las marcas visuales (círculos) dentro de esta imagen.

Para cumplir con este objetivo se lleva a cabo un proceso de transformación de *cambio de escala* [26] . En éste se establece un factor de escala ( $f$ ) y un valor de umbral. Se genera una nueva imagen que se nombra como *imagen escalada* que es  $f$ -veces menor que el tamaño de la imagen binaria y se rellena con valor cero.

Luego se toma una mascarilla de tamaño ( $f$ ), se coloca sobre la imagen binaria y se efectúa un conteo de cuantos pixeles con valor de uno se encuentran en la región cubierta por la mascarilla, es decir, en una superficie de  $f * f$  pixeles. Si el valor del conteo realizado es igual o mayor al valor de umbral previamente establecido, se pone un pixel con valor uno en la imagen escalada. Este proceso se repite hasta que se lleve a cabo el barrido de toda la imagen.

Es importante mencionar que cada región de  $f * f$  pixeles de la imagen binaria que sobrepasó el valor del umbral es representada en la imagen escalada con un sólo pixel con valor uno, dado que ésta es  $f$ -veces mas pequeña. De esta forma se realiza la detección de las marcas visuales.

Debido a que la precisión en la ubicación de las marcas visuales es muy importante, se usó para la imagen binaria una resolución de 640 x 480 pixeles, dado que el algoritmo de estéreo visión se basa en los valores calculados en este proceso.

Se obtiene experimentalmente el diámetro ( $d$ ) del área cubierta por las marcas visuales en

la imagen binaria que es aproximadamente de 26 pixeles ( $d = 26$ ). Durante los experimentos el área varía conforme se mueve el robot, por lo que para garantizar que siempre sean detectadas todas las marcas visuales (que al menos la imagen escalada tenga un bit con valor uno para cada una de ellas), el valor de escala  $f$  se define como:

$$f = \frac{d * \sin(45^\circ)}{2,5} \quad (3.1)$$

Por lo tanto, el valor de escala es de  $f = 7$ , la imagen escalada es de 91 x 68 pixels. Con este valor de escala para la mascarilla, se consiguió que cada una de las marcas visuales sean representadas por al menos tres pixeles en esta nueva imagen en todos los experimentos.

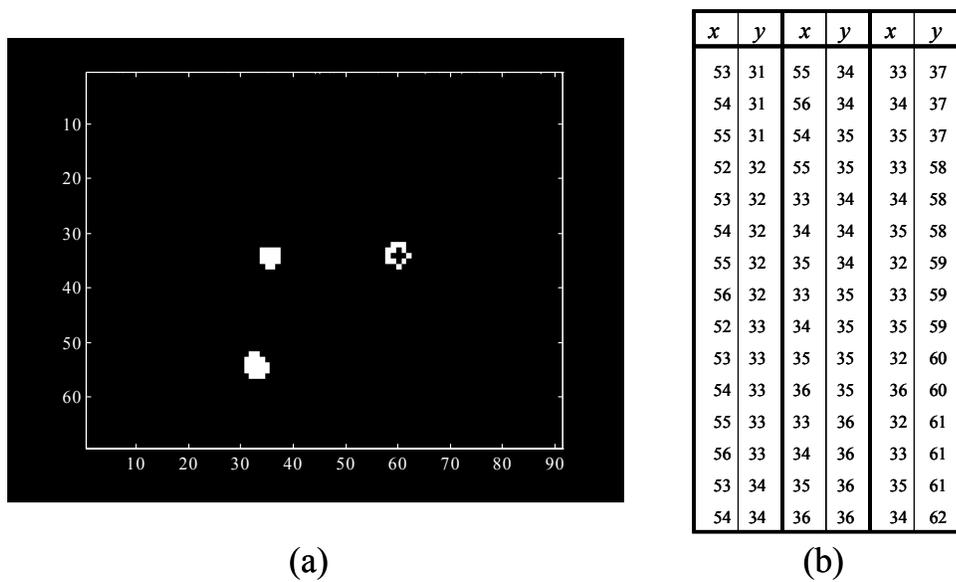


Figura 3.5: Resultado del proceso (a) Imagen redimensionada a 91 x 68 pixeles. (b) lista de valores que representan a las marcas visuales.

En la Figura 3.5.a se muestra el resultado de aplicarle el proceso de escalado a la imagen binaria de la Figura 3.4. En la Figura 3.5.b se enlistan las coordenadas  $(x, y)$  de dichos puntos.

### 3.2.3. Separación de las marcas visuales

Para este proceso se usa como entrada la lista con las coordenadas  $(x, y)$  extraída de la imagen por la detección de los puntos visuales (ver 3.5.b). El objetivo de esta parte del proceso es tomar esta lista de coordenadas  $(x, y)$  y separar sus elementos en tres grupos u objetos.

Para separarlos, se utiliza un criterio conocido como *segmentación por regiones* [27], que para cumplir con el objetivo propuesto consiste en realizar la búsqueda de objetos *8-conexos maximales*. Es decir, se busca el mayor grupo posible de vecinos que tiene cada elemento (formado por sus coordenadas  $(x, y)$ ) contenido en la lista.

Dado el correcto tamaño obtenido, para la mascarilla empleada para reducir la imagen binaria y obtener la imagen escalada se supone que el conjunto de píxeles que representa a cada una de las marcas visuales es siempre *8-conexo*.

En la Figura 3.6 se muestra el resultado de realizar la separación de la lista obtenida en el paso anterior (ver Figura 3.5.b) en tres objetos que representan a las marcas visuales.

Objeto 1				Objeto 2				Objeto 3			
$x$	$y$	$x$	$y$	$x$	$y$	$x$	$y$	$x$	$y$	$x$	$y$
53	31	54	33	33	34	36	36	33	58	35	61
54	31	55	33	34	34	33	37	34	58	34	62
55	31	56	33	35	34	34	37	35	58		
52	32	53	34	33	35	35	37	32	59		
53	32	54	34	34	35			33	59		
54	32	55	34	35	35			35	59		
55	32	56	34	36	35			32	60		
56	32	54	35	33	36			36	60		
52	33	55	35	34	36			32	61		
53	33			35	36			33	61		

Figura 3.6: Resultado de realizar la separación en objetos.

### 3.2.4. Obtención del centroide de las marcas visuales

Para este proceso se utilizan como entradas tanto la imagen binaria como la escalada y consiste en tomar uno a uno los objetos obtenidos en el paso anterior (ver Figura 3.6).

Se comienza tomando las coordenadas  $(x, y)$  de un elemento del objeto y se multiplican por el factor de escala (en nuestro caso  $f = 7$ ).

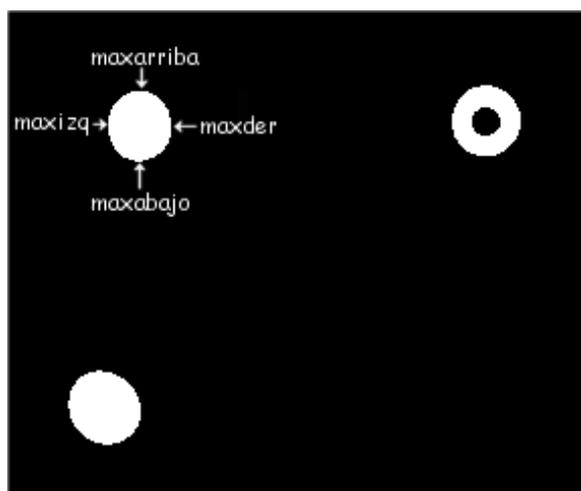


Figura 3.7: Obtención del centroide de los puntos visuales, usando los bordes del círculo.

Con este nuevo par  $(x, y)$  ya sin el efecto de la escala, se utilizan estos valores para ubicarse en la imagen binaria (640 x 480 píxeles). Esta coordenada se usa como punto de partida para buscar los bordes de la marca visual en las cuatro direcciones: arriba, abajo, derecha e izquierda (ver Figura 3.7). Esta búsqueda se repite con todos los elementos del objeto y empleando los valores máximos el centroide  $(c_x, c_y)$  se obtiene:

$$\begin{aligned} c_x &= \frac{\text{máx abajo} + \text{máx arriba}}{2} \\ c_y &= \frac{\text{máx der} + \text{máx izq}}{2} \end{aligned} \quad (3.2)$$

Finalmente, este proceso se repite para los otros dos objetos, obteniendo el valor de los centroides para los tres objetos como se muestra en la Figura 3.8. Es importante aclarar que

aunque no se puede garantizar que los máximos sean únicos, esto no afecta el cálculo dado que es suficiente con encontrar uno de los elementos con máximo valor. Además, el valor de las coordenadas  $(x, y)$  se realiza de forma independiente como se observa en la ecuación 3.2.

Objeto 1		Objeto 2		Objeto 3	
$x$	$y$	$x$	$y$	$x$	$y$
381.5	233.0	243.0	251.0	240.0	422.0

Figura 3.8: Lista final con el centroide obtenido para las marcas visuales.

### 3.2.5. Clasificación de las marcas visuales

Para este proceso se utiliza como entradas tanto la lista de centroides de los objetos (ver Figura 3.8) como la imagen binaria. Para garantizar siempre la correcta clasificación de los puntos, se considera que el codo del robot nunca se encuentre a la misma altura o por debajo de la base del robot. Esta condición permite al robot moverse libremente para tomar piezas de su mesa de trabajo y levantarlas para moverlas sin ningún problema, como se muestra en la Figura 3.9.

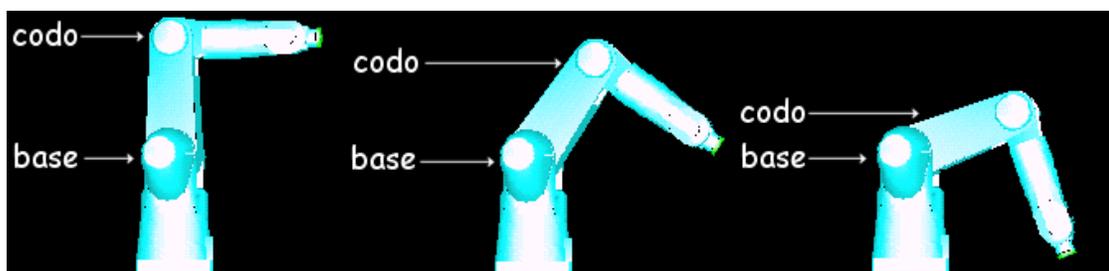


Figura 3.9: Diferentes posiciones del robot A465 donde la condición se cumple y permite libre movimiento.

Como se muestra en la Figura 3.4, uno de las marcas visuales tiene en su interior un círculo negro, el cual no afecta el desempeño de los pasos anteriores y se usará para realizar

la clasificación. Primero, se usa uno a uno el valor de los centroides, para posicionarse en el centro de las marcas visuales y se verifica en su  $\delta$ -vecindad si hay pixeles con valor cero. Si existen al menos seis pixeles con este valor, esto indica que la marca visual en el que se encontraron estos pixeles se se puede etiquetar con el nombre de órgano terminal.

Ahora, para clasificar a los dos marcas visuales restantes, se usa la condición establecida anteriormente (ver Figura 3.9). Sabemos que uno debe estar ubicado en el codo y el otro en la base del robot, así que dado que el codo siempre debe estar arriba de la base, es suficiente con comparar la coordenada  $x$  de las dos marcas y la que tenga el valor más grande se etiqueta como *base* y a la última como *codo*.

Base		Codo		Órgano terminal	
$x$	$y$	$x$	$y$	$x$	$y$
381.5	233.0	243.0	251.0	240.0	422.0

Figura 3.10: Resultado de realizar la clasificación de las marcas visuales.

En la Figura 3.10 se muestra el resultado de aplicar este proceso a la lista de centroides de la Figura 3.8. Los criterios usados para etiquetar los puntos han funcionado exitosamente en todos los experimentos, por lo cual, el problema queda resuelto completamente.

### 3.3. Eliminación de la distorsión radial

Para que el sistema de visión por computadora cubra el espacio de trabajo del robot A465 (ver Figura 3.11) a una distancia de 1.70 metros se eligió colocar a las cámaras los lentes Cosmimar C815B(C30811)[45].

Sin embargo, para cubrir todo el espacio de trabajo, el lente Cosmimar C815B utiliza un ángulo de visión muy grande. Esto genera que las imágenes capturadas se vean distorsionadas (ver Figura 3.12.a). En particular la distorsión para este lente, se conoce como distorsión radial [10].

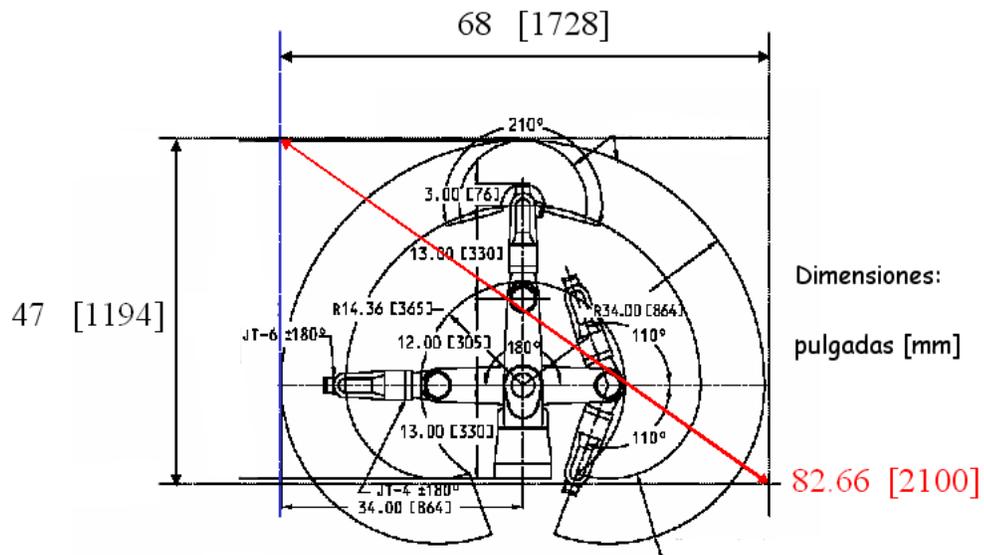


Figura 3.11: Espacio de trabajo para el robot A465.

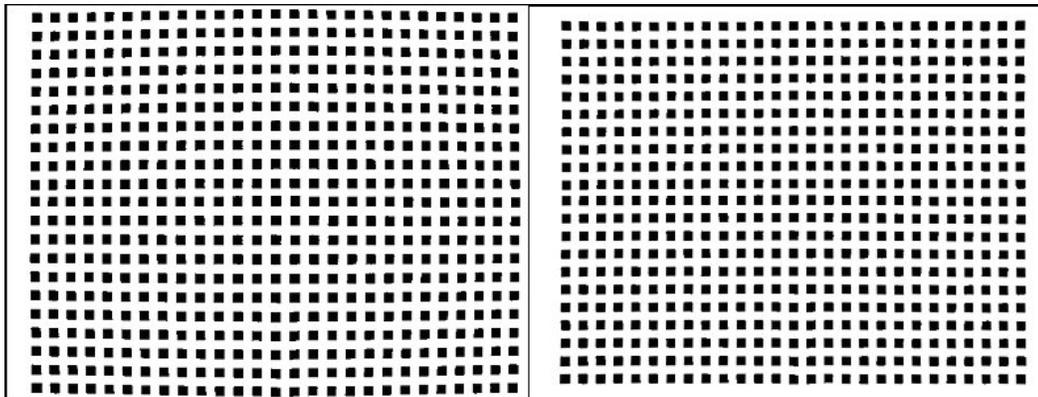


Figura 3.12: Malla para detectar la distorsión radial. con distorsión (izquierda) y sin distorsión (derecha).

Dado que el modelo óptico de estereo visión que se presenta en el capítulo siguiente utiliza los centroides de las marcas visuales, es muy importante eliminar la distorsión radial que aparece en las imágenes originales y obtener la posición real de las marcas en la escena con respecto a ellas.

Para eliminar la distorsión radial se usa el algoritmo presentado en [10], el cual, se basa en el siguiente modelo:

$$r = \frac{r'}{1 + d_2 r'^2 + d_4 r'^4} \quad (3.3)$$

donde  $r$  es el radio corregido para la imagen y  $r'$  es el radio que genera la distorsión. Usando la imagen distorsionada se obtiene de forma experimental los siguientes valores:

$$d_2 = -6,166e^{-7} \quad d_4 = 2,083e^{-12}$$

Con estos valores para cada punto en la imagen sus nuevas coordenadas sin distorsión radial son:

$$x = X + \frac{x' - X}{1 + d_2 r'^2 + d_4 r'^4} \quad y = Y + \frac{y' - Y}{1 + d_2 r'^2 + d_4 r'^4} \quad (3.4)$$

donde  $(x', y')$  son las coordenadas del punto en la imagen distorsionada,  $(x, y)$  son las nuevas coordenadas sin distorsión del mismo punto y  $(X, Y)$  son las coordenadas del centro de la distorsión cuyos valores obtenidos de forma experimental son de (320, 245). La Figura 3.12.b muestra la misma malla sin distorsión radial.



# Capítulo 4

## Modelo óptico de estéreo visión

### 4.1. Antecedentes

El primer paso que se efectuó fue un estudio bibliográfico de trabajos realizados en la solución del problema de estéreo correspondencia, con el fin de conocer esfuerzos realizados anteriormente y con ello poder distinguir adecuadamente la aportación del modelo de estéreo visión propuesto en este trabajo. Uno de los modelos más similares es el presentando en [28] ( ver apéndice).

### 4.2. Introducción

En este capítulo se muestra un nuevo modelo óptico de estéreo visión. Este usa la información visual obtenida por los algoritmos de visión artificial explicados en el capítulo anterior y la información de algunos parámetros internos o externos de la cámara y del lente. Con estos datos, el modelo es capaz de ubicar un punto en el espacio.

De esta forma que los valores de entrada al modelo son:

1. La distancia real (cm) que hay entre el codo y el órgano terminal del robot A465.
2. El valor del ángulo de rotación de ambas cámaras con respecto al *eje x* del robot.

3. Dado que las cámaras usan un lente compuesto se requiere el valor de la distancia entre los dos lentes que lo componen, es decir, la distancia entre el ocular y el objeto.
4. El factor de conversión de pixel a centímetro para el eje horizontal.

La salida del modelo es la posición en espacio de un punto (sus coordenadas  $(x, y, z)$ ).

El problema de estereo correspondencia se divide en dos partes:

**Problema 4.1** *El problema de correspondencia que consiste en saber que partes de la imagen derecha e izquierda son proyecciones del mismo elemento que se encuentra en la escena real.*

**Problema 4.2** *El problema de reconstrucción que consiste en, dado un conjunto de elementos de las partes ubicadas en ambas imágenes, y posiblemente con alguna información de la geometría del sistema de visión por computadora, se pretende obtener la ubicación en el espacio del órgano terminal del robot.*

Este modelo tiene la finalidad de calcular la posición de un punto usando dos puntos conocidos cualesquiera. A continuación se presenta su explicación y aplicación considerando que éstos se encuentran colocados en las articulaciones del robot manipulador.

### 4.3. Cálculo de los valores requeridos para el modelo

En esta sección se expone cómo realizar el cálculo de los valores que usará el modelo óptico de estereo visión. Los procedimientos son simples y fáciles de implementar. Es importante mencionar que estos valores sólo requieren calcularse una sola vez ya que se montan las cámaras en una posición fija. La única excepción es el centroide de las marcas visuales, cuyos valores si se necesitan recalcular cada vez que el robot se mueva.

### 4.3.1. Distancia real entre el codo y el órgano terminal del robot A465

El valor de la distancia se obtiene directamente de las especificaciones dadas por CRS Robotics [18]. Es importante mencionar que este valor no incluye la distancia de la herramienta que se le coloque al órgano terminal, por lo tanto, la distancia en centímetros entre el codo y el órgano terminal es de 42 cm. Este valor se nombra como  $c$  para este modelo.

### 4.3.2. Ángulo de rotación de las cámaras

El valor del ángulo de rotación de las cámaras con respecto al *eje x* del robot A465 se muestra en la Figura 4.1.b.

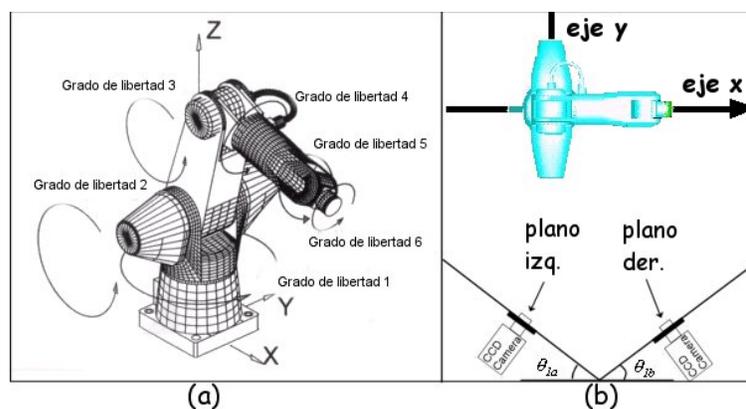


Figura 4.1: El robot A465 con sus 6 grados de libertad (a). Los ángulos  $\theta_{1a}$  y  $\theta_{1b}$  de las cámaras con respecto al eje x del robot (b).

Los valores de los ángulos de rotación se nombran como  $\theta_{1a}$  y  $\theta_{1b}$  para este modelo y se obtienen mediante el siguiente experimento:

1. Se ubica al robot A465 en su posición de inicio<sup>1</sup> (ver Figura 4.1.a). Entonces se gira su 1<sup>er</sup> grado de libertad y se captura una imagen del robot con la cámara derecha.

<sup>1</sup>ready, por su término en inglés.

2. Se analiza la imagen mediante los algoritmos de visión por computadora explicados en el capítulo anterior. Se obtiene la posición de las marcas visuales.
3. Se verifica si las marcas visuales del codo y la base están alineados (ver Figura 4.2). Si no están alineados se gira de nuevo el 1<sup>er</sup> grado de libertad del robot A465 y se regresa al paso 2.
4. Si los puntos visuales están alineados se lee del controlador del robot el ángulo del 1<sup>er</sup> grado de libertad. Este ángulo se le nombra  $\theta_{1b}$ .
5. Se repite del paso 1 al paso 4 con la cámara izquierda. El ángulo que se obtiene se nombra  $\theta_{1a}$ .

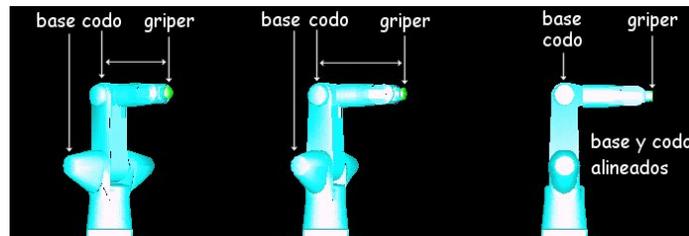


Figura 4.2: Procedimiento para calcular el ángulo de rotación de las cámaras.

### 4.3.3. Distancia entre las lentes ocular y objetivo

Una aportación es el estudio realizado al lente compuesto modelo Cosmimar C815b (ver Figura 4.3.a). El valor resultante obtenido experimentalmente, de la distancia entre los lentes internos (ocular y objetivo) es de 5 cm y se nombra como  $d_{int}$  para este modelo (ver Figura 4.3.b).

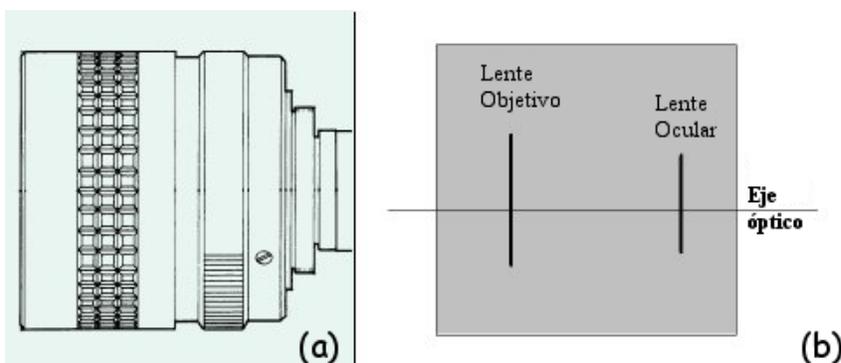


Figura 4.3: Lente Cosmicar C815b (a). Diagrama esquemático de los lentes (b).

#### 4.3.4. Factor de conversión de pixel a centímetro

Dado que éste es un modelo se establece al proyectar el valor de la distancia entre las dos marcas visuales (cm) en el CCD de la cámara y recibir la imagen en pantalla (pixels). Se requiere de un *factor de conversión* para que todos los valores del modelo estén en centímetros.

Para obtener el valor de este factor se diseñó y construyó un riel para montar una de las cámaras del sistema de visión por computadora (ver Figura 4.4). El riel se empleó para tomar una serie de fotos a diferentes distancias del robot A465.

El valor del factor de conversión  $k_f$  se calcula mediante el siguiente experimento:

1. Se monta la cámara sobre el riel a una distancia que se define como  $r$ . El valor inicial de la distancia es de 236 cm y se considera que la cámara se encuentra en paralelo con el *eje x* del robot (ver Figura 4.5). Esta condición elimina la posibilidad de obtener imágenes que sean proyecciones de la escena.
2. Se toma una foto de la escena. Se procesa con los algoritmos de visión por computadora y se obtienen los centroides de las marcas visuales.
3. Se calcula la distancia  $a'$  entre las marcas visuales ubicadas en el codo y órgano terminal del robot y se almacena. Esta distancia está en pixeles.

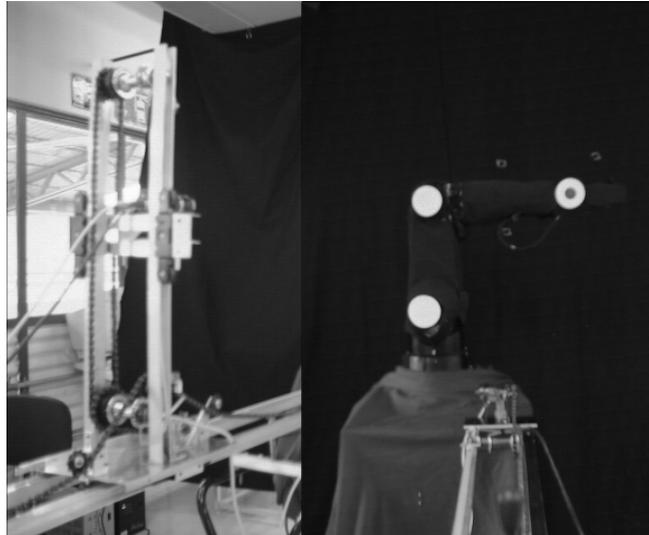


Figura 4.4: Vistas del riel fabricado para obtener el factor de conversión.

4. Recorriendo sobre el riel la cámara hacia adelante 2 cm y se repite el segundo paso hasta que la cámara se encuentre una distancia de 64 cm de la escena.

La distancia  $a$  está en centímetros (ver Figura 4.5 ). La forma de calcular esta distancia sería:

$$a = a' k_f \quad (4.1)$$

De la Figura 4.5 se obtiene:

$$\tan \alpha = \frac{\frac{a}{2}}{f} = \frac{\frac{d}{2}}{f + r} \quad (4.2)$$

Sustituyendo la ec. 4.1 en la ec. 4.2 se obtiene:

$$a' k_f = \frac{d f}{f + r}$$

$$k_f = \frac{d f}{a' (f + r)} \quad (4.3)$$

Del experimento anterior, se tiene un conjunto de datos  $(r, a')$  que se sustituyen en la ec. 4.3. Finalmente, el valor promedio para el factor de conversión es:

$$k'_f \text{ promedio} = 1,38e^{-6} \quad (4.4)$$

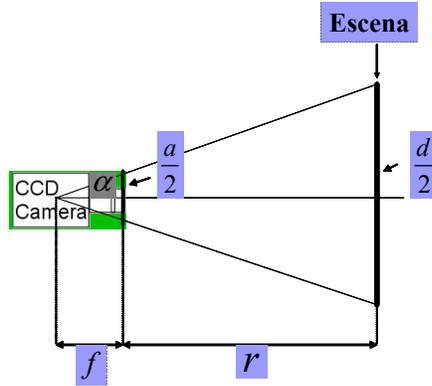


Figura 4.5: Experimento diseñado para obtener el factor de conversión.

## 4.4. Cálculo de la posición en el espacio de un punto

En esta sección se explica a detalle el modelo óptico de estereo visión propuesto en este trabajo. El objetivo del modelo es encontrar la ubicación en el espacio de un punto desconocido usando la información visual obtenida por el sistema con visión artificial. Para el caso particular del robot A465, el punto desconocido es la marca visual ubicada en el órgano terminal del robot cuando se encuentra en cualquier otra lugar distinta a la posición de inicio (ver Figura 4.4). Para el modelo, el punto desconocido se nombra como  $c_{des}$  (ver Figura 4.6).

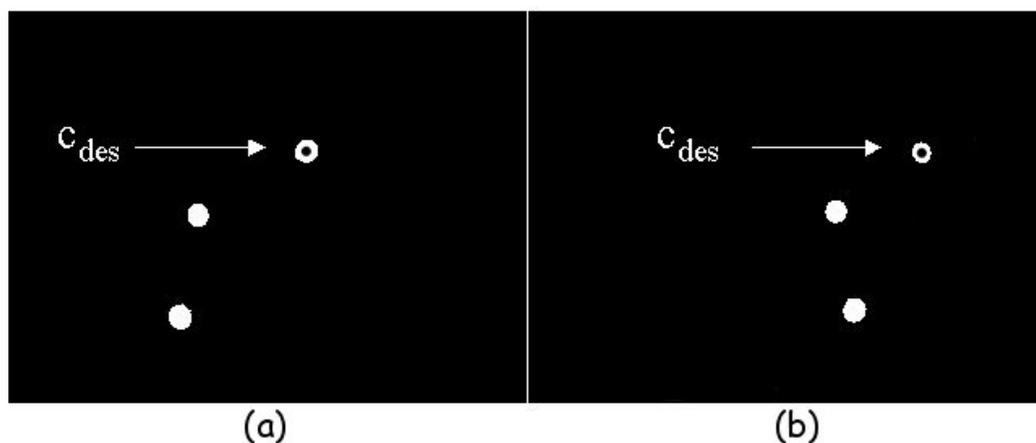


Figura 4.6: Se desea localizar la marca visual  $c_{des}$  cuando el órgano terminal del robot está en otra posición diferente a la de inicio.

La coordenada  $z$  (ver Figura 4.1.a) se obtiene mediante la ec. 4.5. Esta ecuación se cumple si las dos cámaras se encuentran a la misma altura. Para garantizar que se cumpla esta condición se diseñó y se construyó un soporte ajustable para las cámaras. Este soporte permite mover las cámaras hasta que se satisfaga esta condición.

$$z = k_f u_L = k_f u_R \quad (4.5)$$

donde  $k_f$  es el factor de conversión pixel a centímetro,  $u_L$  y  $u_R$  representan las coordenadas  $c_x$  (ver ec. 3.2) para la cámara izquierda y derecha, respectivamente.

Ahora, para las coordenadas  $(x, y)$  se establece un modelo geométrico (ver Figura 4.7) donde se usa una vista superior del robot A465 y el sistema de visión por computadora.

Para mayor claridad el desarrollo del modelo se divide en dos etapas:

1. El cálculo de la distancia entre las cámaras y las marcas visuales colocadas en el codo y órgano terminal del robot.
2. El cálculo de las coordenadas  $(x, y)$ .

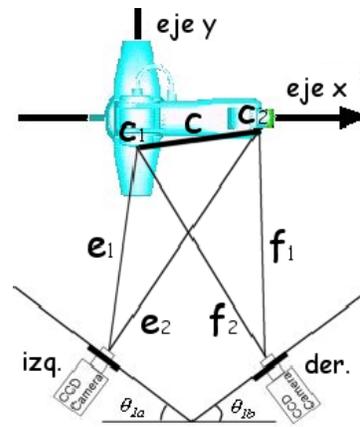


Figura 4.7: El modelo geométrico se construye con la distancias entre las cámaras y dos marcas visuales del robot A465.

#### 4.4.1. Etapa 1: Cálculo de la distancia entre las cámaras y las marcas visuales del robot

Esta etapa consiste en escribir las ecuaciones necesarias para calcular las distancias de las cámaras izquierda ( $e_1, e_2$ ) y derecha ( $f_1, f_2$ ) a las marcas visuales ubicadas en el codo  $c_1$  y el órgano terminal  $c_2$  del robot A465 (ver Figura 4.7). Adicionalmente, se obtienen los valores de los ángulos  $\vartheta$  y  $\theta_4$ .

Estos cálculos se realizan cada vez que se desea obtener el valor de las coordenadas  $(x, y)$ . De esta forma el modelo es flexible, en el sentido de que si es necesario mover las cámaras de su ubicación original, el modelo no se verá afectado, pues se recalculan las distancias de las cámaras del robot y continua funcionando.

Es importante aclarar que las cámaras se deben mantener fijas durante todo el desarrollo de un experimento. Terminado el experimento, las cámaras pueden moverse hacia adelante o atrás, con la única condición de que el robot y su espacio de trabajo aparezcan completos en las imágenes.

Para obtener la distancias de las cámaras al robot A465, se realiza el siguiente experimento:

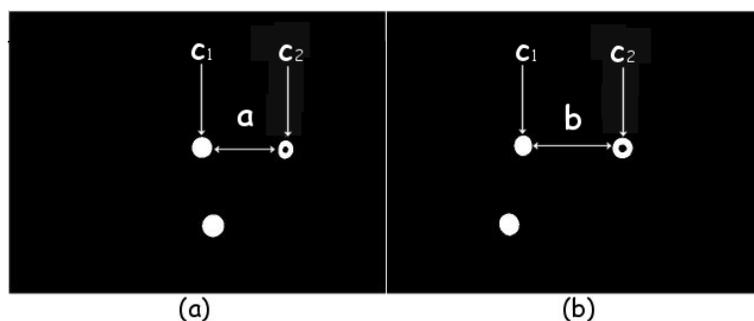


Figura 4.8: Puntos visuales  $c_1$ ,  $c_2$  y la distancia entre ellos.

1. Se lleva a cabo el experimento explicado en la sección anterior y se obtiene el valor de los ángulos  $\theta_{1a}$  y  $\theta_{1b}$ .
2. Se emplea los algoritmos de visión por computadora para encontrar el centroide de las marcas visuales  $c_1$  y  $c_2$  que aparecen en ambas imágenes. Se calcula el valor de la distancia proyectada  $c_{proy}$  entre ambos puntos (ver Figura 4.9), el valor de esta distancia en la imagen izquierda se nombra  $a$  y en la derecha se llama  $b$  (ver Figura 4.8). Estas distancias están en píxeles.

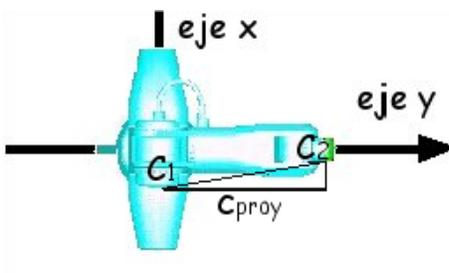


Figura 4.9: La distancia  $c_{proy}$  se obtiene de proyectar la distancia real entre los marcas  $c_1$  y  $c_2$  en el *eje y*.

3. De la Figuras 4.10 y 4.11 se obtiene el conjunto de ecuaciones que se muestran a continuación:

Cámara izquierda	Cámara derecha
$\beta_1 = \begin{cases} \tan^{-1}\left(\frac{d_{int}}{L_2-L_1}\right), & L_1 < L_2 \\ 90^\circ + \tan^{-1}\left(\frac{L_1-L_2}{d_{int}}\right), & L_1 \geq L_2 \end{cases}$	$\gamma_3 = \begin{cases} \tan^{-1}\left(\frac{d_{int}}{L_2-L_3}\right), & L_3 < L_2 \\ 90^\circ + \tan^{-1}\left(\frac{L_3-L_2}{d_{int}}\right), & L_3 \geq L_2 \end{cases}$
$a_1 = \sqrt{(d_{int})^2 + (L_2 - L_1)^2}$	$b_2 = \sqrt{(d_{int})^2 + (L_2 - L_3)^2}$
$a_2 = \sqrt{(a_1)^2 + (a)^2 - (2 a a_1 \cos(\beta_1))}$	$b_1 = \sqrt{(b_2)^2 + (b)^2 - (2 b b_2 \cos(\gamma_3))}$
$\beta_2 = \text{sen}^{-1}\left(\frac{a}{a_2} \text{sen}(\beta_1)\right)$	$\gamma_2 = \text{sen}^{-1}\left(\frac{b}{b_1} \text{sen}(\gamma_3)\right)$
$\beta_3 = 180^\circ - \beta_1 - \beta_2$	$\gamma_1 = 180^\circ - \gamma_2 - \gamma_3$
$\beta_6 = 180^\circ - \beta_1 - \theta_{1a}$	$\gamma_6 = \gamma_2 + \gamma_1 - \theta_{1b}$
$\beta_5 + \alpha_1 = 180^\circ - \beta_6 - \beta_2$	$\alpha_2 + \gamma_5 = 180^\circ - \gamma_6 - \alpha_{int}$
$\alpha_2 = 180^\circ - (\beta_5 + \alpha_1) - \beta_2$	$\alpha_1 = \gamma_6 + \alpha_{int} - \gamma_2$

Se calculan los valores de la distancia de las cámaras al robot:

Cámara izquierda	Cámara derecha
$e_1 = c \frac{\text{sen}(\alpha_2)}{\text{sen}(\beta_2)}$	$f_2 = c \frac{\text{sen}(\alpha_1)}{\text{sen}(\gamma_2)}$
$e_2 = c \frac{\text{sen}(\beta_5 + \alpha_1)}{\text{sen}(\beta_2)}$	$f_1 = c \frac{\text{sen}(\alpha_2 + \gamma_5)}{\text{sen}(\gamma_2)}$

(4.6)

donde:

$L_1$ : distancia en la imagen del borde izquierdo a la marca visual  $c_1$  en la cámara izquierda.

$L_3$ : distancia en la imagen del borde izquierdo a la marca visual  $c_1$  en la cámara derecha.

$L_2$ : la mitad del ancho total de la imagen binaria (320 pixeles).

$c$ : distancia entre el codo y el órgano terminal (42,7126 cm.).

$\alpha_{int}$ : ángulo entre la línea  $c$  y el eje  $x$  del robot (4,1620°).

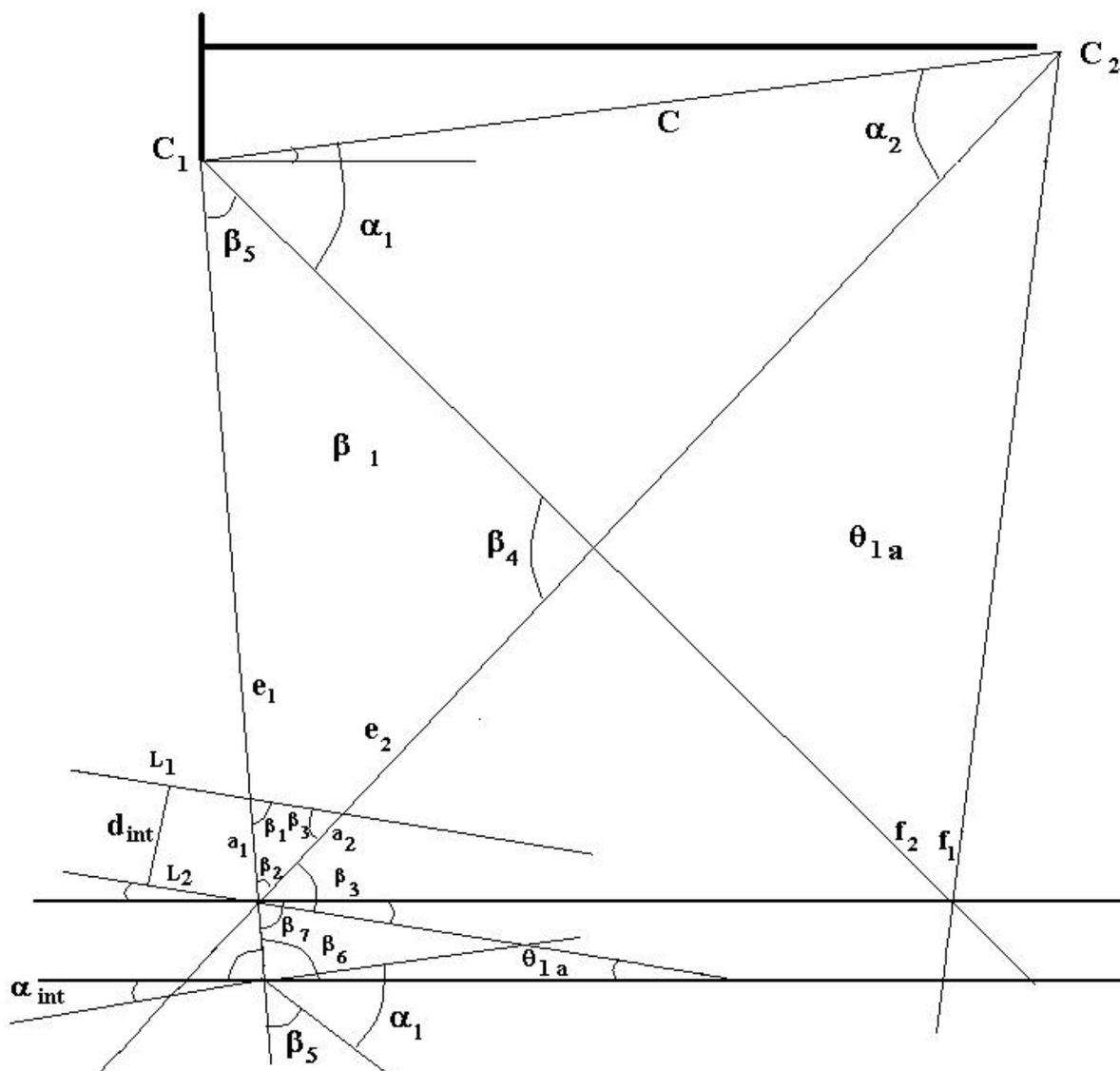


Figura 4.10: Primer parte del modelo geométrico para obtener las distancias de la cámara izquierda  $e_1$ ,  $e_2$ .

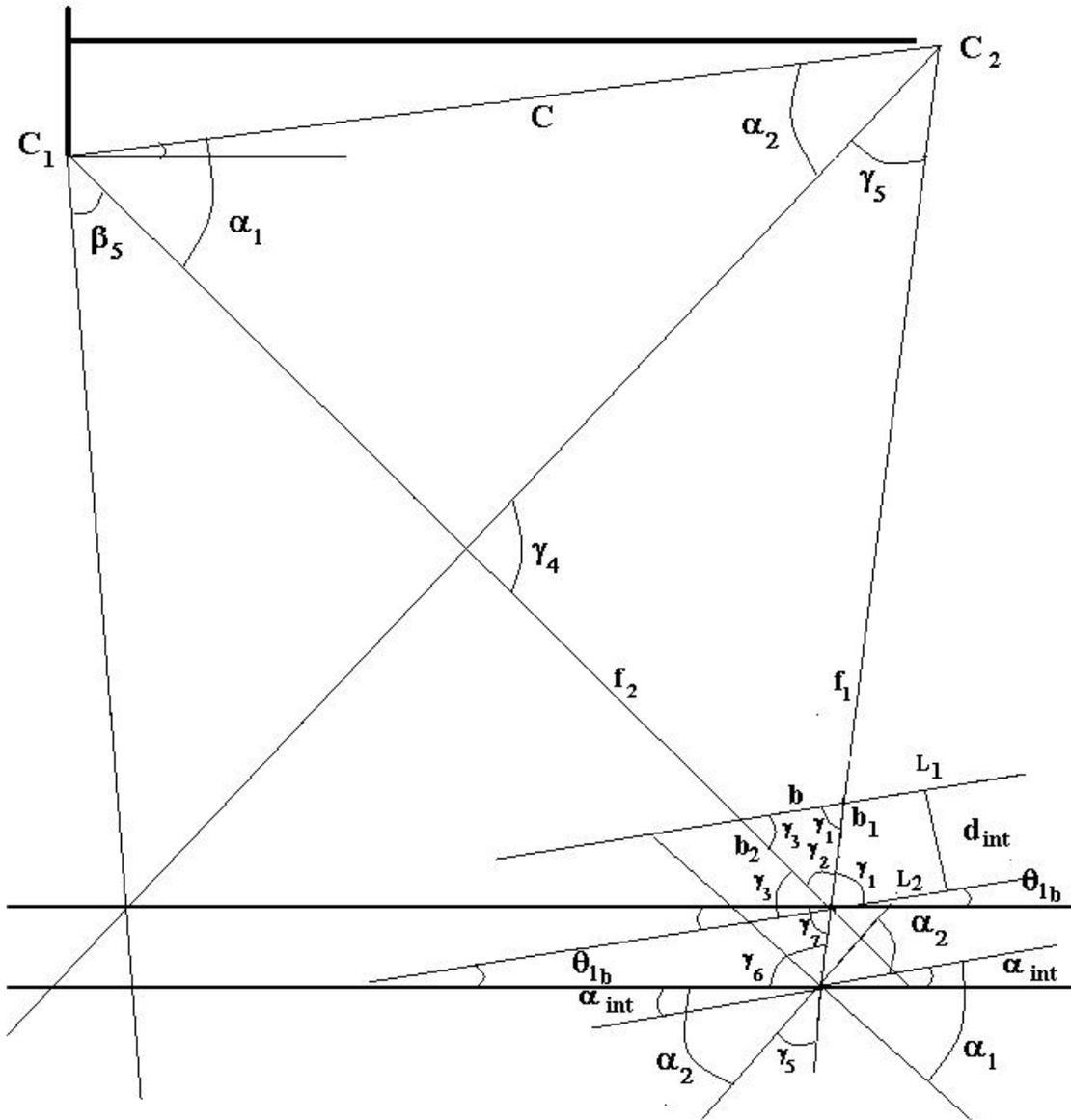


Figura 4.11: Primer parte del modelo geométrico para obtener las distancias de la cámara derecha  $f_1, f_2$ .

#### 4.4.2. Etapa 2: cálculo de las coordenadas $(x, y)$ del órgano terminal del robot

En esta etapa se encontrarán las coordenadas  $(x, y)$  del punto desconocido  $c_{des}$  (ver Figura 4.6). Para ello, se requiere obtener la distancia del punto desconocido a una de las marcas visuales, se selecciona la marca visual que llamaremos el punto  $c_1$ , esta distancia se nombra como  $c'$  (ver Figura 4.12) y el ángulo formado por la línea descrita por el punto  $c_1$  y el punto desconocido  $c_{des}$  con el *eje x* del robot se nombra como  $\alpha'$ .

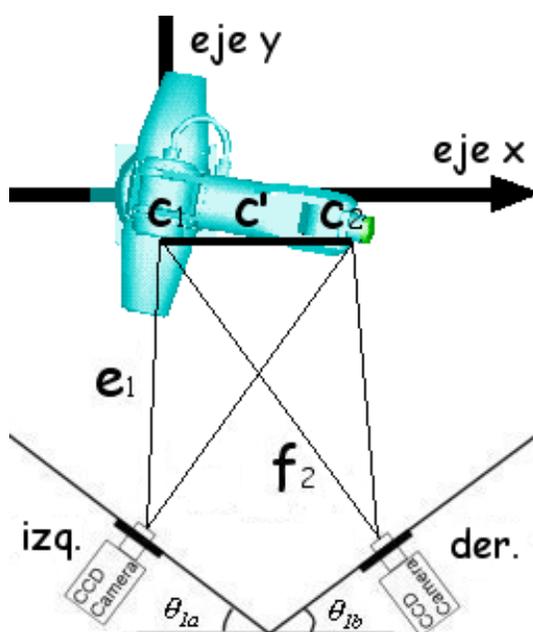


Figura 4.12: El robot se mueve a una nueva posición (diferente de la posición de Ready), por lo que ahora la distancia  $c'$  es desconocida.

Los valores de la distancia  $c'$  y el ángulo  $\alpha'$  se obtienen de la siguiente forma:

1. A la marca visual  $c_1$  se renombra como  $c_{con}$  y se toman las distancias de esta marca a las cámaras que son  $e_1$  y  $f_2$  (obtenidas en la etapa anterior, ver Figura 4.10).

2. Con estos datos y la Figura 4.13 se obtienen las siguientes ecuaciones:

Cámara izquierda	Cámara derecha
$\beta'_1 = \begin{cases} \tan^{-1}\left(\frac{d_{int}}{L'_2-L'_1}\right), & L'_1 < L'_2 \\ 90^\circ + \tan^{-1}\left(\frac{L'_1-L'_2}{d_{int}}\right), & L'_1 \geq L'_2 \end{cases}$	$\gamma'_3 = \begin{cases} \tan^{-1}\left(\frac{d_{int}}{L'_2-L'_3}\right), & L'_3 < L'_2 \\ 90^\circ + \tan^{-1}\left(\frac{L'_3-L'_2}{d_{int}}\right), & L'_3 \geq L'_2 \end{cases}$
$a'_1 = \sqrt{(d_{int})^2 + (L'_2 - L'_1)^2}$	$b'_2 = \sqrt{(d_{int})^2 + (L'_2 - L'_3)^2}$
$a'_2 = \sqrt{(a'_1)^2 + (a')^2 - 2 a' a'_1 \cos(\beta'_1)}$	$b'_1 = \sqrt{(b'_2)^2 + (b')^2 - 2 b' b'_2 \cos(\gamma'_3)}$
$\beta'_2 = \text{sen}^{-1}\left(\frac{a'}{a'_2} \text{sen}(\beta'_1)\right)$	$\gamma'_2 = \text{sen}^{-1}\left(\frac{b'}{b'_2} \text{sen}(\gamma'_3)\right)$
$\beta'_3 = 180 - \beta'_1 - \beta'_2$	$\gamma'_1 = 180 - \gamma'_2 - \gamma'_3$

(4.7)

$\delta_3 = 180^\circ + \theta_{1a} + \theta_{1b} - \beta'_3 - \gamma'_1$
$\theta'_4 = \delta_3$
$\beta'_4 = 180^\circ - \theta'_4$
$\gamma'_4 = \beta'_4$
$\gamma'_5 = 180^\circ - \gamma'_2 - \gamma'_4$

(4.8)

$e = e_1 \left( \frac{\text{sen}(\beta'_2)}{\text{sen}(\beta'_4)} \right)$	$f = (f_2 - e) \left( \frac{\text{sen}(\gamma'_2)}{\text{sen}(\gamma'_5)} \right)$
--	--

(4.9)

$c' = \sqrt{(e)^2 + (f)^2 - 2 e f \cos(\theta'_4)}$
$\alpha_1 = \text{sen}^{-1}\left(\frac{f}{c'} \text{sen}(\theta'_4)\right)$
$\alpha_T = \gamma'_1 - \theta_{1b}$
$\alpha' = \alpha_1 - \alpha_T$

(4.10)

3. Mediante las ecuaciones 4.8, 4.9 y ??, se calcula  $e$  y  $f$ . Por lo que las coordenadas  $(x, y)$ :

$x = c' \cos(\alpha')$
$y = c' \text{sen}(\alpha')$

(4.11)

donde:

$a'$ : distancia entre el punto  $c_1$  y el punto desconocido  $c_2$  en la cámara izquierda (píxeles)

$b'$ : distancia entre el punto  $c_1$  y el punto desconocido  $c_2$  en la cámara derecha (píxeles)

$L'_1$ : distancia en la imagen del borde izquierdo a la marca visual  $c_1$  en la cámara izquierda.

$L'_3$ : distancia en la imagen del borde izquierdo a la marca visual  $c_1$  en la cámara derecha.

$L'_2$ : la mitad del ancho total de la imagen binaria (320 píxeles).

De este desarrollo se puede observar que las relaciones entre las ecuaciones proporcionan un modelo muy práctico para la obtención de las coordenadas  $(x, y)$  y con la ecuación 4.5 se encuentra la coordenada  $z$  de cualquier punto desconocido. En resumen:

$x = c' \cos(\alpha')$	(4.12)
$y = c' \operatorname{sen}(\alpha')$	
$z = k_f u_L$	

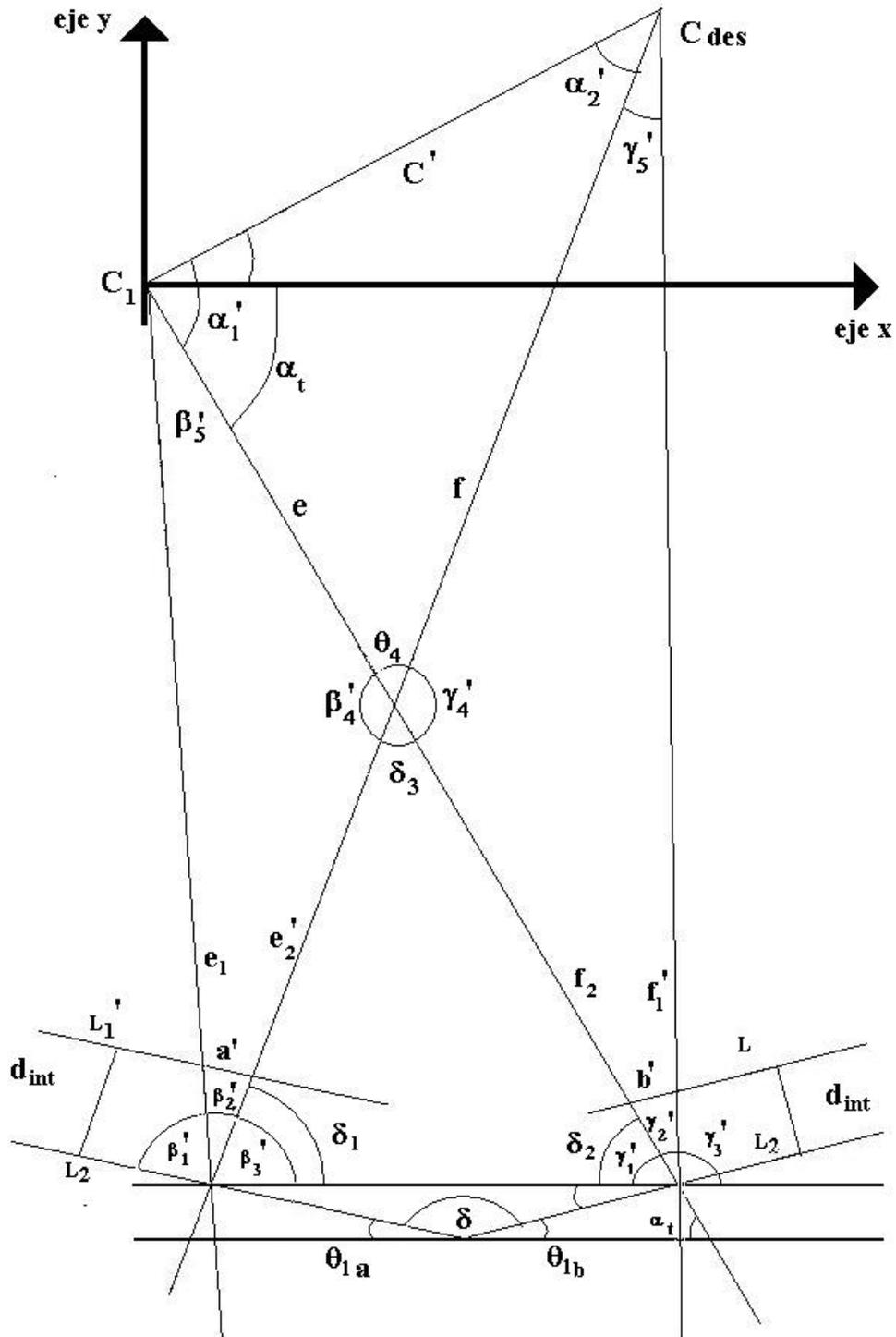


Figura 4.13: Segunda parte del modelo geométrico para el cálculo de las coordenadas  $(x, y)$  del punto desconocido.

## 4.5. Cinemática del robot A465

Una forma de describir la cinemática de un robot es mediante cuatro parámetros para cada articulación [15]. Dos de ellos para describir la articulación y los otros dos para representar la conexión con sus articulaciones vecinas.

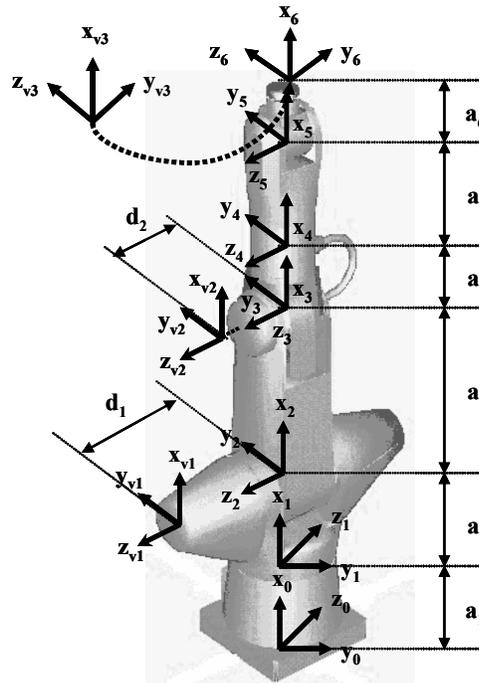


Figura 4.14: Diagrama de los ejes coordenados para las articulaciones y marcas visuales del robot A465.

La descripción de un robot mediante estos cuatro parámetros se conoce como la notación de Denavit-Hartenberg [21] y para el robot A465 se obtiene de la siguiente forma:

1. Se asigna un eje coordenado para las seis articulaciones del robot y para las tres marcas visuales, como se muestra en la Figura 4.14.

2. Se encuentra la matriz de transformación para cada una de estas articulaciones:

$$\begin{array}{|c|c|}
 \hline
 {}^6_5T = \begin{bmatrix} c \theta_6 & -s \theta_6 & 0 & a_6 \\ 0 & 0 & 1 & 0 \\ -s \theta_6 & -c \theta_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^5_4T = \begin{bmatrix} c \theta_5 & -s \theta_5 & 0 & a_5 \\ s \theta_5 & c \theta_5 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 \hline
 {}^4_3T = \begin{bmatrix} 1 & 0 & 0 & a_4 \\ 0 & c \alpha_4 & -s \alpha_4 & 0 \\ 0 & s \alpha_4 & c \alpha_4 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^3_2T = \begin{bmatrix} c \theta_3 & -s \theta_3 & 0 & a_3 \\ s \theta_3 & c \theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 \hline
 {}^2_1T = \begin{bmatrix} c \theta_2 & -s \theta_2 & 0 & a_2 \\ s \theta_2 & c \theta_2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^1_0T = \begin{bmatrix} 1 & 0 & 0 & a_1 \\ 0 & c \alpha_1 & -s \alpha_1 & 0 \\ 0 & s \alpha_1 & c \alpha_1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 \hline
 \end{array} \tag{4.13}$$

donde:

$\theta_i, \alpha_i$  son variables de rotación

$a_i, d_i$  son variables de translación

$c = \cos$  y  $s = \sin$

Y para las marcas visuales que se muestran en la Figura 4.15, las transformaciones son:

$$\begin{array}{|c|c|}
 \hline
 {}^{v_1}_2T = \begin{bmatrix} c \theta_2 & -s \theta_2 & 0 & 0 \\ s \theta_2 & c \theta_2 & 0 & d_1 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & \\
 \hline
 {}^{v_2}_3T = \begin{bmatrix} c \theta_3 & -s \theta_3 & 0 & 0 \\ s \theta_3 & c \theta_3 & 0 & d_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^{v_3}_5T = \begin{bmatrix} c \theta_5 & -s \theta_5 & 0 & 0 \\ s \theta_5 & c \theta_5 & 0 & d_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 \hline
 \end{array} \tag{4.14}$$

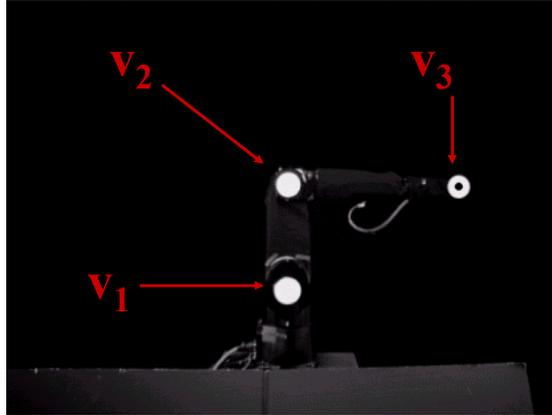


Figura 4.15: Ubicación de las marcas visuales en el robot A465.

## 4.6. Cinemática visual del robot A465

### 4.6.1. Desarrollo y resultados experimentales

En esta sección se presenta una aplicación para el modelo óptico de estereo visión. La aplicación consiste en calcular los valores de dos parámetros  $\theta_2$  y  $\theta_3$  del modelo cinemático del robot A465 (ver Figura 4.16).

El procedimiento a seguir fue:

1. Usando el modelo óptico de estereo visión propuesto se obtiene la posición en el espacio de las tres marcas visuales, que nombramos:

$P_1 = (X_1, Y_1, Z_1)$
$P_2 = (X_2, Y_2, Z_2)$
$P_3 = (X_3, Y_3, Z_3)$

2. Con estos tres puntos se tiene los siguientes vectores:

$V_1 = (X_2 - X_1)i + (Y_2 - Y_1)j + (Z_2 - Z_1)k$
$V_2 = (X_3 - X_2)i + (Y_3 - Y_2)j + (Z_3 - Z_2)k$
$V_3 = (X_2 - X_1)i$

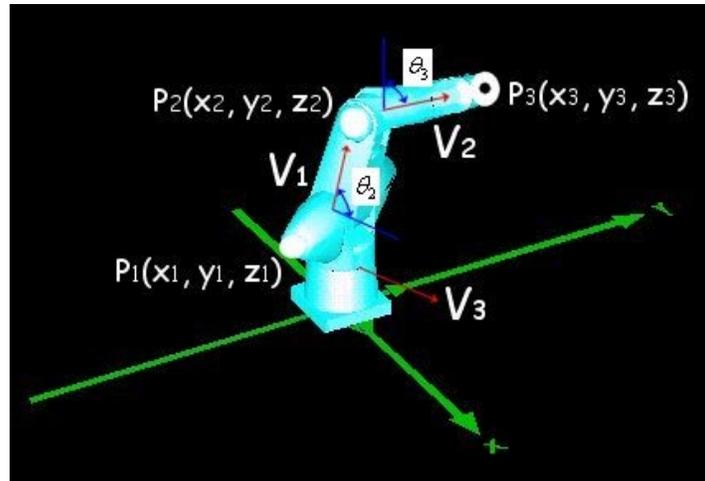


Figura 4.16: Ángulos  $\theta_2$  y  $\theta_3$  a ser calculados visualmente.

3. Donde  $V_1, V_2$  Y  $V_3$  son los vectores formados por las coordenadas de los puntos localizados visualmente (ver Figura 4.16), una vez obtenidos estos vectores, se les multiplica utilizando el producto punto para calcular los ángulos buscados:

$V_2 \cdot V_3 = (X_2 - X_1) \cdot (X_3 - X_2) + (Y_2 - Y_1) \cdot (Y_3 - Y_2) + (Z_2 - Z_1) \cdot (Z_3 - Z_2)$
$V_1 \cdot V_3 = (X_2 - X_1) \cdot (X_3 - X_1)$
$ V_1  = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2 + (Z_2 - Z_1)^2}$
$ V_2  = \sqrt{(X_3 - X_2)^2 + (Y_3 - Y_2)^2 + (Z_3 - Z_2)^2}$
$ V_3  = X_3 - X_1$

Los ángulos son obtenidos con la siguiente expresión:

$\theta_2 = \cos^{-1} \frac{V_1 \cdot V_3}{ V_1  \cdot  V_3 }$
$\theta_3 = \cos^{-1} \frac{V_2 \cdot V_3}{ V_2  \cdot  V_3 }$

4. Enseguida se diseñó un experimento donde se movió la segunda articulación del robot A465 comenzado en  $90^\circ$  y terminando en  $-90^\circ$  como se muestra en la Figura 4.17.

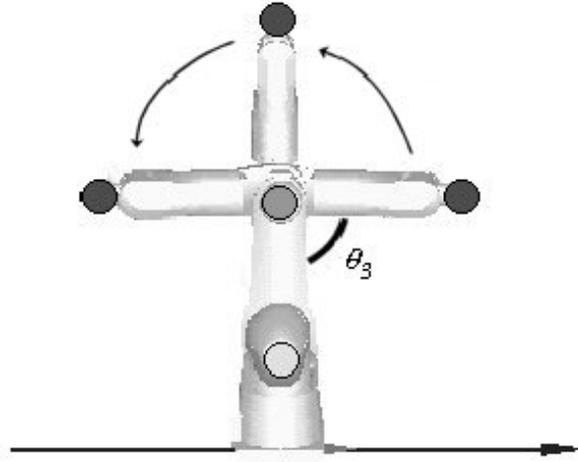


Figura 4.17: Movimiento realizado por la segunda articulación del robot durante el experimento.

El valor para los demás parámetros del modelo cinemático son:

$a_1 = 7$	$a_2 = 6$	$a_3 = 12$	$a_4 = 4$	$a_5 = 9$	$a_6 = 3$
$d_1 = 7$	$d_2 = 3$	$d_3 = 1$			
$\alpha_1 = 0$	$\theta_2 = 0$		$\alpha_4 = 0$	$\theta_5 = 0$	$\theta_6 = 0$

y el ángulo de las cámaras  $\theta_{1a} = 12,676$  y  $\theta_{1b} = 15,181$ . Los valores de  $a_i$ ,  $i = 1, \dots, 6$  y  $d_j$ ,  $j = 1, 2, 3$  están en pulgadas y los de los ángulos en grados.

El resultado obtenido se muestra en la Figura 4.18 y se observa que el error fue de  $\pm 2^\circ$ .

## 4.7. Ventajas y Desventajas del modelo

Dado que para este modelo se realizó el desarrollo teórico y su implementación en laboratorio, a continuación se enumeran sus principales ventajas y desventajas.

$\theta_3$ real	X	Y	Z	$\theta_2$	$\theta_3$
-90	0	0	0	0	-92.784
-75	1.7895	-0.9240	10.288	0	-76.71
-60	5.1599	1.0904	19.904	0	-60.778
-45	11.652	0.36275	28.148	0	-45.662
-30	20.708	-0.5310	34.18	0	-30.681
-15	30.692	-2.4409	38.679	0	-15.778
0	41.964	-5.4864	39.827	0	0.9057
15	53.024	-10.158	38.25	0	13.889
30	63.62	-15.524	34.664	0	28.646
45	72.446	-20.866	28.155	0	43.489
60	79.238	-24.966	19.767	0	58.249
75	83.456	-26.794	10.332	0	72.746
90	84.684	-28.192	0.6477	0	88.112

Figura 4.18: Resultados obtenidos de calcular el ángulo  $\theta_3$  mediante información visual.

#### 4.7.1. Ventajas

1. **Uso general:** Este modelo puede ser usado en aquellas aplicaciones donde se requiera conocer la ubicación en el espacio de un punto mediante visión por computadora. Esto es posible, pues solo el modelo requiere de colocar dos marcas visuales en la escena para llevar a cabo el cálculo. Para esta tesis se eligió como punto desconocido, el órgano terminal del robot manipulador A465.
2. **Ubicación de las cámaras:** Bajo el esquema de cámara fija, al realizar diferentes experimentos es posible mover las cámaras de posición o rotarlas sobre el *eje z* con la única restricción de que siempre aparezca el robot completo en las imágenes. Esto es permitido, pues el modelo no se ve afectado si cambia la posición, dado que se elimina la distorsión producida por los lentes y en rotación dado que si el valor de los ángulos de la cámaras llegara a cambiar sólo es necesario ejecutar la rutina encargada de calcularlos automáticamente .
3. **Flexibilidad:** El modelo no depende de valores como la distancia entre las cámaras

o entre las cámaras y la escena. Lo cual da mucha flexibilidad en el espacio necesario para la instalación del sistema de visión por computadora. Esto es una ventaja pues el modelo sería adecuado para trabajar en espacios reducidos.

4. **Sencillo:** El número y tipo de ecuaciones necesarias para obtener la posición en el espacio de un punto desconocido, es muy adecuado para desarrollar aplicaciones pues, una vez que se cuenta con sus valores de entrada, los cálculos son simples y rápidos de ejecutar.
5. **Usa solo información visual:** Éste no usa información de otros elementos del sistema como la cinemática del robot A465, el modelo de la cámara o el modelo geométrico del objeto en la escena.

#### 4.7.2. Desventajas

1. **Aplicación solo a tareas de baja precisión:** De los resultados mostrados en la aplicación de la cinemática visual se tiene un error de  $\pm 2^\circ$ . Este rango no es aceptable para realizar aplicaciones que requieran un alto grado de precisión.
2. **Limitación del espacio de trabajo:** El uso de dos marcas visuales provoca que cuando el robot gira sobre su base la marca ubicada sobre el órgano terminal, reduzca su área hasta que deja de ser detectada por el sistema de visión por computadora. Para evitar esto, fue necesario limitar este giro del robot a  $\pm 30^\circ$ .

Hasta este capítulo se cuenta con una herramienta capaz de ubicar un punto en el espacio mediante un sistema de visión por computadora. Esta es una **aportación** importante que puede ser usada para desarrollar aplicaciones en campos como:

- **El arte y diseño:** Reconstrucción de objetos. Se podría usar un rayo láser para apuntar la objeto de interes. Este punto (generado por el láser al incidir sobre el objeto) será el punto desconocido, del cual con el modelo se calcula su posición en el espacio. Este

proceso se repite barriendo toda la pieza con el láser. Finalmente, con esta información se obtiene un modelo tridimensional del objeto en la computadora.

- **Ingeniería:** Control y Automatización. Se considera un punto del objeto a manipular por el órgano terminal del robot, el punto desconocido, del cual con el modelo se calcula su posición en el espacio. Este valor se incorpora al lazo de retroalimentación del sistema para generar la acción de control.
- **Procesos industriales:** control de calidad.

Dado que es de interés de este trabajo el área de control de sistemas físicos, en el capítulo siguiente se desarrolló como aplicación para esta herramienta un controlador visual difuso para un robot manipulador. Dentro de esta misma área también es importante realizar un estudio de estabilidad, el cual se presentará en el capítulo 6. Los controladores presentados serán tipo PD con las siguientes variantes:

- **Controlador visual con observador de modos deslizantes:** Se considera que la velocidad de las articulaciones  $\dot{q}$  no es medible. Por ello, se usa un observador libre del modelo del sistema para obtener la estimación  $\hat{q}$ , la cual se incluye en la ley de control.
- **Controlador visual con redes neuronales multicapa:** Se supone que los términos de gravedad y fricción no son medibles. Por lo tanto, se efectúa una compensación de estas fuerzas mediante una red neuronal multicapa, ésta se incorpora en la ley de control.
- **Controlador visual con observador de modos deslizantes y redes neuronales multicapa:** Se contemplan los dos casos anteriores, por lo que la ley de control incluye a la estimación  $\hat{q}$  y otra red neuronal multicapa.



# Capítulo 5

## Aplicación: control visual difuso

### 5.1. Introducción

La teoría de la lógica difusa fue por primera vez presentada por Lofti A. Zadeh en 1965 en su artículo “*Fuzzy Sets*” [71]. Él desarrolló el concepto de “*estado*”, que es la base de la teoría de control moderna. A principios de los 60’s, él pensó que la teoría clásica de control ponía demasiado énfasis en la precisión y, por lo tanto no podía manejar sistemas complejos. En 1962, él escribió que para manejar sistemas biológicos “se necesita un tipo radicalmente diferente de matemáticas, las matemáticas de las cantidades difusas o borrosas que no pueden ser descritas en términos de las distribuciones probabilísticas” [72]. Después, él formalizó estas ideas en su artículo “*Fuzzy Sets*”.

Desde su nacimiento, la teoría difusa ha generado controversia. Algunos, como Richard Bellman, estuvo a favor y comenzó a trabajar en este campo. Otros estuvieron en contra y consideraron que esta nueva teoría estaba en contra de los principios básicos de la ciencia. El mayor reto, sin embargo, venía de los matemáticos en estadística y probabilidad que decían que la probabilidad era suficiente para caracterizar incertidumbres y que cualquier tipo de problemas que podía resolver la lógica difusa, podían ser igual o mejor resueltos con la teoría de probabilidad. Dado que no había aplicaciones reales en el comienzo de la lógica difusa, fue difícil defenderla solamente desde un punto de vista filosófico. Sin embargo, la teoría difusa

no sucumbió en medio de la tormenta. A finales de los 60's muchos métodos difusos como los algoritmos difusos, toma de decisiones difusa, etc., fueron presentados.

Es justo decir que el establecimiento de la lógica difusa como un campo independiente fue en mucho gracias a la dedicación y el extenso trabajo de Zadeh. La mayoría de los conceptos fundamentales fueron propuestos por Zadeh a finales de los 60's y principios de los 70's. Después de la introducción de los conjuntos difusos en 1965, el propuso los conceptos de los algoritmos difusos [73], toma de decisiones difusa en 1970 [5], y ordenamiento difuso en 1971 [74]. En 1973, publicó otro artículo, "Outline of a new approach to the analysis of complex systems and decision processes" [75], donde estableció los fundamentos del control difuso. En este artículo, introdujo el concepto de las variables lingüísticas y propuso usar las reglas difusas **SI-ENTONCES** para formular el conocimiento humano.

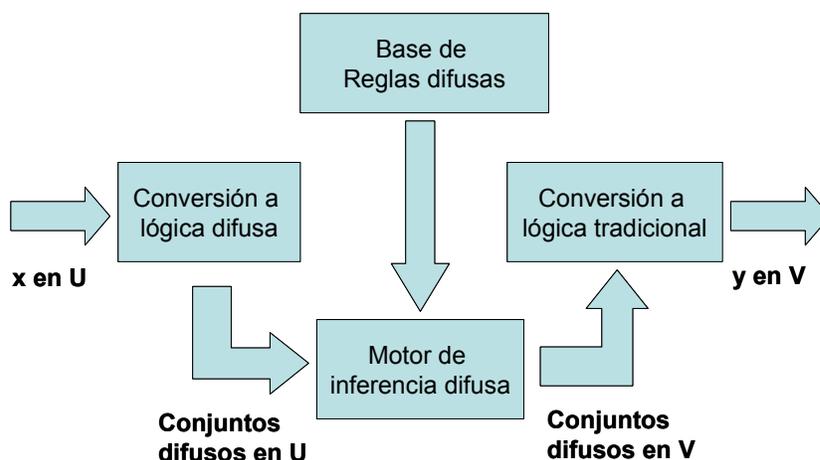


Figura 5.1: Configuración básica de los sistemas difusos.

Una importante contribución de la teoría de los sistemas difusos es que provee un procedimiento sistemático para transformar una base de conocimientos en un mapeo no-lineal. Mediante esta transformación es posible usar sistemas basados en conocimiento (sistemas difusos) en aplicaciones de ingeniería

Un gran evento en los 70's fue el nacimiento de los controladores difusos para sistemas en

tiempo real. En 1975, Mamdani y Assilian establecieron la estructura básica del controlador difuso (ver Figura 5.1). Sus resultados fueron publicados en el artículo “An experiment in linguistic synthesis with a fuzzy logic controller” [48]. Ellos encontraron que un controlador difuso era muy fácil de construir y trabajaba notoriamente bien.

En 1980, Sugeno creó en Japón la primera aplicación difusa: el control de una planta de purificación de agua de Fuji Electric. En 1983, fue pionero en trabajos usando la lógica difusa para robots, un automóvil con sistema de auto-estacionado fue controlado con esta lógica [61]. Para comienzos de los 90’s, una enorme cantidad de productos comerciales con lógica difusa aparecieron en el mercado.

## 5.2. Diseño de sistemas difusos mediante el esquema de matriz difusa

Suponga que se tienen los siguientes pares de entrada-salida [69]:

$$(x_0^p, y_0^p), p = 1, 2, \dots, N \quad (5.1)$$

donde  $x_0^p \in U = [\alpha_1, \beta_1] \times \dots \times [\alpha_n, \beta_n] \subset \mathbb{R}^n$  y  $y_0^p \in V = [\alpha_y, \beta_y] \in \mathbb{R}$ . El objetivo es diseñar un sistema difuso  $f(x)$  basado en estos  $N$  pares entrada-salida.

Ahora se presenta en cinco pasos el procedimiento de diseño de sistemas difusos mediante el esquema de matriz difusa.

### 5.2.1. Definir los conjuntos difusos para cubrir los espacios de entrada y salida

Específicamente para cada  $[\alpha_i, \beta_i]$ ,  $i = 1, 2, \dots, n$ , se definen  $N_i$  conjuntos difusos  $A_i^j$  ( $j = 1, 2, \dots, N_i$ ), que se requieran para estar completo en  $[\alpha_i, \beta_i]$ ; esto es, para cualquier  $x_i \in [\alpha_i, \beta_i]$ , existe  $A_i^j$  tal que  $\mu_{A_i^j}(x_i) \neq 0$ . donde  $\mu_{A_i^j}(\cdot)$  es la función de pertenencia.

Por ejemplo, podemos escoger que  $\mu_{A_i^j}(x_i)$  sean funciones de pertenencia pseudo-trapezoidales,

tal que:  $\mu_{A_i^j}(x_i) = \mu_{A_i^j}(x_i, a_i^j, b_i^j, c_i^j, d_i^j)$  donde  $a_i^1 = b_i^1 = \alpha_i$ ,  $c_i^j = a_i^{j+1} < b_i^{j+1} = d_i^j$  ( $j = 1, 2, \dots, N_i - 1$ ) y  $c_i^{N_i} = d_i^{N_i} = \beta_i$ .

De forma similar, se define  $N_y$  conjuntos difusos  $B_j$ , ( $j = 1, 2, \dots, N_y$ ) tal que sea completo en  $[\alpha_y, \beta_y]$ . También se elige  $\mu_{B^j}(y)$  sean funciones de pertenencia pseudo-trapezoidales, tal que:  $\mu_{B^j}(y) = \mu_{B^j}(y, a^j, b^j, c^j, d^j)$ , donde  $a^1 = b^1 = \alpha_y$ ,  $c^j = a^{j+1} < b^{j+1} = d^j$  ( $j = 1, 2, \dots, N_y - 1$ ), y  $c^{N_y} = d^{N_y} = \beta_y$ . En la Figura 5.2 se muestra un ejemplo para  $n = 2$ , donde  $N_1 = 5, N_2 = 7, N_y = 5$ , las etiquetas difusas son para  $x_1 : S2, S1, CE, B1, B2$ ;  $x_2 : S3, S2, S1, CE, B1, B2, B3$ ;  $y : S2, S1, CE, B1, B2$  y las funciones de pertenencia son triangulares.

### 5.2.2. Generar una regla para cada par entrada-salida

Primero, para cada par de entrada-salida  $(x_{01}^p, \dots, x_{0n}^p; y_0^p)$ , se determinan los valores de pertenencia de  $x_{0i}^p$  ( $i = 1, 2, \dots, n$ ) en los conjuntos difusos  $A_i^j$  ( $j = 1, 2, \dots, N_i$ ) y los valores de pertenencia para  $y_0^p$  en los conjuntos difusos  $B^l$  ( $l = 1, 2, \dots, N_y$ ). Esto es calcular lo siguiente:  $\mu_{A_i^j}(x_{0i}^p)$  para  $j = 1, 2, \dots, N_i$ ,  $i = 1, 2, \dots, n$  y  $\mu_{B^l}(y_0^p)$  para  $l = 1, 2, \dots, N_y$ . Para el ejemplo de la Figura 5.2, tenemos que aproximadamente que:  $x_{01}^1$  tiene un valor de pertenencia de 0,8 en  $B1$ , 0,2 en  $B2$ , y cero en los demás conjuntos difusos;  $x_{02}^1$  tiene un valor de pertenencia de 0,6 en  $S1$ , 0,4 en  $S2$ , y cero en los demás conjuntos difusos;  $y_0^1$  tiene un valor de pertenencia de 0,8 en  $CE$ , 0,2 en  $B1$ , y cero en los demás conjuntos difusos.

Ahora, para cada variable  $x_i$  ( $i = 1, 2, \dots, n$ ), se determina el conjunto difuso en el cual  $x_{0i}^p$  tiene el valor de pertenencia mas grande, esto es, determinar  $A_i^{j*}$  tal que  $\mu_{A_i^{j*}}(x_{0i}^p) \geq \mu_{A_i^j}(x_{0i}^p)$  para  $j = 1, 2, \dots, N_i$ . Similarmente, determinar  $B^{l*}$  tal que  $\mu_{B^{l*}}(y_0^p) \geq \mu_{B^l}(y_0^p)$  para  $l = 1, 2, \dots, N_y$ . Por ejemplo, en la Figura 5.2 para el par de entrada-salida  $(x_{01}^1, x_{02}^1, y_0^1)$  el conjunto difuso sería  $A_1^{j*} = B1$ ,  $A_2^{j*} = S1$  y  $B^{l*} = CE$ , y para el par  $(x_{01}^2, x_{02}^2, y_0^2)$  el conjunto difuso sería  $A_1^{j*} = B1$ ,  $A_2^{j*} = CE$  y  $B^{l*} = B1$ .

Finalmente, se obtiene las reglas difusas SI-ENTONCES como:

$$\text{SI } x_1 \text{ es } A_1^{j*} \text{ y } \dots \text{ y } x_n \text{ es } A_n^{j*}, \text{ entonces } y \text{ es } B^{l*} \quad (5.2)$$

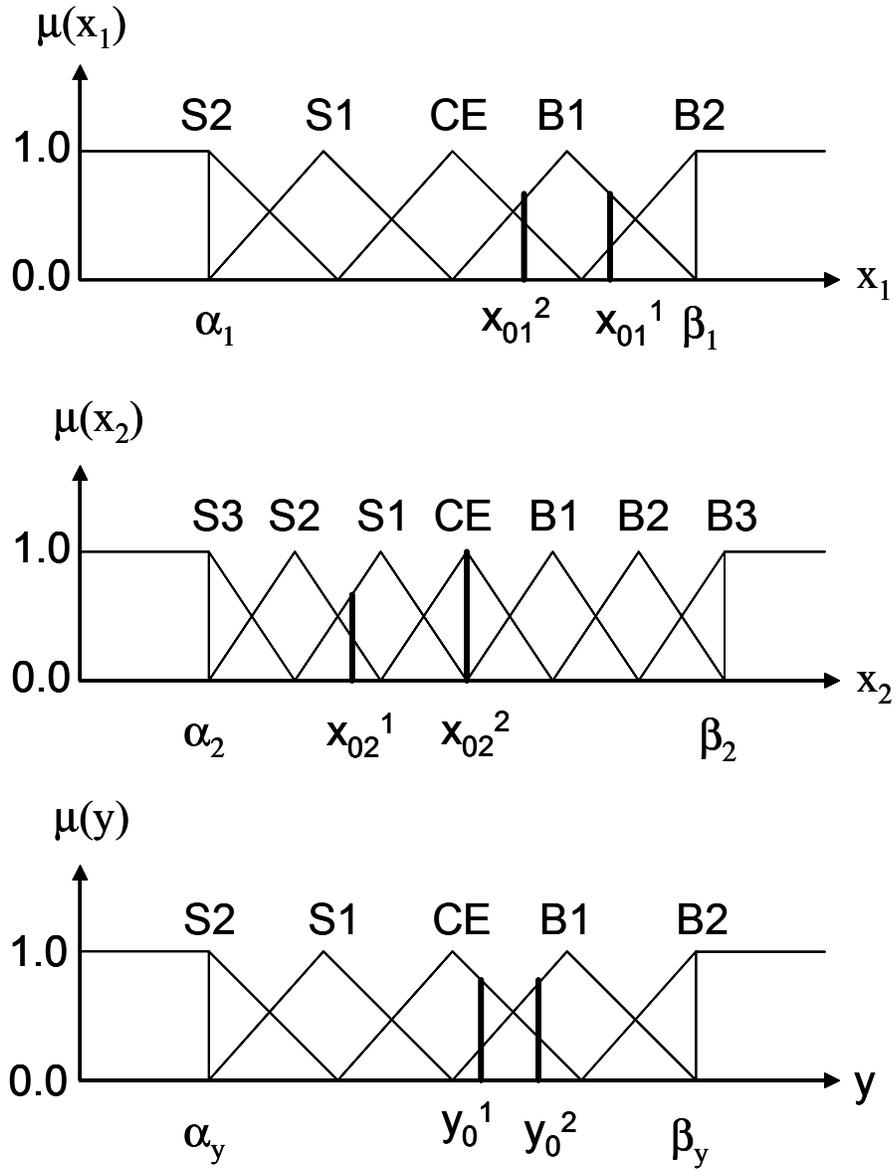


Figura 5.2: Ejemplo de funciones de pertenencia para pares entrada-salida en el caso de dos entradas.

Por ejemplo de la Figura 5.2 el par  $(x_{01}^1, x_{02}^1, y_0^1)$  tendrá la regla difusa:

$$\text{SI } x_1 \text{ es } B1 \text{ y } x_2 \text{ es } S1, \text{ entonces } y \text{ es } CE \quad (5.3)$$

y para el par  $(x_{01}^2, x_{02}^2, y_0^2)$ :

$$\text{SI } x_1 \text{ es } B1 \text{ y } x_2 \text{ es } CE, \text{ entonces } y \text{ es } B1 \quad (5.4)$$

### 5.2.3. Asignar un grado a cada regla difusa

Dado que el número de pares entradas-salidas es usualmente grande y con cada par se genera una regla, es altamente probable que haya reglas conflictivas, esto es, reglas que son iguales en la parte SI con diferente parte ENTONCES. Para resolver este conflicto, se asigna un grado a cada regla generada y se guardará solamente una regla por cada grupo de reglas con conflicto que será la regla con mayor grado. En esta forma no solamente se resuelve el problema de conflicto entre reglas sino además, el número de reglas es reducido de forma significativa.

El grado de una regla difusa es definido como sigue: Suponga que la regla (5.2) es generada del par entrada-salida  $(x_0^p; y_0^p)$ , entonces su grado es definido como:

$$D(rule) = \prod_{i=1}^n \mu_{A_i^j} (x_{0i}^p) \mu_{B_i^k} (y_0^p) \quad (5.5)$$

Para el ejemplo de la Figura 5.2, la regla generada por  $(x_{01}^1, x_{02}^1, y_0^1)$  tiene el grado:

$$\begin{aligned} D(rule1) &= \mu_{B1} (x_{01}^1) \mu_{S1} (x_{02}^1) \mu_{CE} (y_0^1) \\ &= 0,8 * 0,6 * 0,8 = 0,384 \end{aligned} \quad (5.6)$$

y para la regla generada por  $(x_{01}^2, x_{02}^2, y_0^2)$  sería:

$$\begin{aligned} D(rule1) &= \mu_{B1} (x_{01}^2) \mu_{CE} (x_{02}^2) \mu_{B1} (y_0^2) \\ &= 0,6 * 1 * 0,7 = 0,42 \end{aligned} \quad (5.7)$$

### 5.2.4. Crear la base de reglas difusas

La base de reglas difusas consiste en los siguientes tres conjuntos de reglas:

- Las reglas generadas en el paso 2 que no tengan conflicto con ninguna otra regla.
- La regla de cada grupo de reglas con conflicto que sea la regla con mayor grado.
- Reglas lingüísticas creadas por el experto humano en el proceso para el cual se está diseñando el sistema difuso (conocimiento consciente).

Dado que los primeros dos conjuntos son obtenidos de conocimiento inconsciente, la base de reglas combina ambos tipos de conocimiento consciente e inconsciente.

S3					
S2					
S1					
CE					
B1					
B2					
B3					
	S2	S1	CE	B1	B2

Figura 5.3: Matriz difusa generada por los conjuntos difusos de las variables de entrada que será llenada con la base de reglas difusas.

Intuitivamente, se puede ilustrar la base de reglas difusas como una matriz difusa para el caso de dos dimensiones. Por ejemplo, en la Figura 5.3 se muestra la matriz difusa de la base de reglas correspondiente a los conjuntos difusos de la Figura 5.2. Cada caja representa

una combinación de conjuntos difusos en  $[\alpha_1, \beta_1]$  y los conjuntos difusos en  $[\alpha_2, \beta_2]$  y sus posibles reglas. Un grupo de reglas con conflicto consiste de las reglas en una misma caja. Este método puede ser visto como llenar las cajas con las reglas apropiadas.

### 5.2.5. Construcción del sistema difuso

Se puede usar cualquiera de los esquemas mencionados en [69] para construir un sistema difuso como el de la Figura 5.1 basado en la base de reglas difusas que se acaba de obtener.

## 5.3. Diseño de un controlador difuso

El control difuso y control convencional tienen ciertas similitudes y diferencias [69]. Éstos son similares en los siguientes aspectos:

- Resuelven el mismo tipo de problemas, esto es, problemas de control. Por lo tanto, ellos deben tratar con todos los temas comunes en cualquier problema de control, por ejemplo, estabilidad y desempeño.
- Las herramientas matemáticas usadas para el análisis y el diseño de sistemas de control es similar, dado que ellos trabajan los mismos temas (estabilidad, convergencia, etc.) para el mismo tipo de sistemas.

Sin embargo, existe una diferencia fundamental entre el control difuso y el control convencional:

- El control convencional comienza con el modelo matemático del proceso, y los controladores son diseñados para este modelo; el control difuso, por otra parte, comienza con la heurística y la experiencia humana (en términos de reglas SI-ENTONCES), y los controladores son diseñados sintetizando estas reglas. Esto es, la información usada para construir los dos tipos de controladores son distintas; vea la Figura 5.4. Controles difusos avanzados pueden hacer uso de ambos modelos; el heurístico y matemático, ver [69].

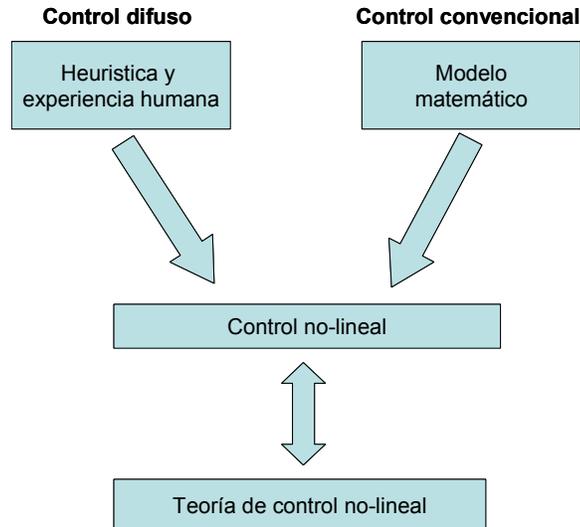


Figura 5.4: Control difuso versus Control convencional.

Para muchos problemas prácticos de control (por ejemplo, control de procesos industriales), es difícil obtener un modelo matemático sencillo y preciso, sin embargo, existen expertos humanos que pueden proveer información heurística que puede ser muy útil para controlar el proceso. El control difuso es muy útil para este tipo de problemas. Así, si el modelo matemático del proceso es desconocido o parcialmente desconocido, se pueden diseñar controladores difusos de una forma sistemática que garantice cierto criterio de desempeño.

Se puede clasificar las metodologías de diseño de controladores difusos en dos categorías:

1. El enfoque de prueba y error.
2. El enfoque teórico.

En el enfoque de prueba y error, un conjunto de reglas difusas SI-ENTONCES son desarrolladas mediante una verbalización introspectiva de una base de conocimiento obtenida a partir de la experiencia (por ejemplo, el manual operativo del sistema) y solicitando a los expertos que contesten un cuestionario diseñado cuidadosamente; entonces, los controladores

difusos son construidos con estas reglas difusas; finalmente, son probados en el sistema real y si el desempeño no es satisfactorio, las reglas son sintonizadas o rediseñadas en un ciclo de prueba y error hasta que el desempeño sea satisfactorio.

En el enfoque teórico, la estructura y los parámetros del controlador difuso son diseñados de tal forma que el criterio de desempeño (por ejemplo, estabilidad) esté garantizado. Por supuesto, en el diseño de controladores difusos para sistemas prácticos es recomendable que combinen ambos enfoques siempre que esto sea posible y así obtener el mejor controlador difuso.

### 5.3.1. Diseño de un controlador difuso mediante el enfoque de prueba y error

En esta sección se presentan los pasos principales para el diseño de controladores difusos mediante el enfoque de prueba y error. El diseño mediante esta orientación [69] puede ser resumido en los siguientes tres pasos:

- **Paso 1. Análisis del sistema real y seleccionar las variables de estado y las variables de control:** Las variables de estado deben representar completamente las características principales del sistema y las variables de control deben ser capaces de influenciar los estados del sistema. Las primeras son las entradas al controlador difuso y las segundas son las salidas del controlador difuso. Esencialmente, en este paso se define el dominio en el cual el controlador difuso va a operar.
- **Paso 2. Obtener las reglas difusas SI-ENTONCES que relacionen las variables de estado con las variables de control:** La formulación de estas reglas pueden obtenerse mediante dos enfoques heurísticos. El enfoque más común involucra la verbalización introspectiva del experto humano. El otro camino incluye una entrevista de los operadores expertos mediante un cuestionario diseñado cuidadosamente. Con estas técnicas, se puede obtener un prototipo de las reglas de control difusas.
- **Paso 3. Combinar las reglas difusas obtenidas en el sistema difuso y probar**

**el sistema en lazo cerrado con este sistema difuso como controlador:** Esto es, correr el sistema en lazo cerrado con este controlador difuso y si el desempeño no es satisfactorio, sintonizarlo o rediseñar el controlador mediante prueba y error y repetir este procedimiento hasta obtener un desempeño satisfactorio.

## 5.4. Diseño del controlador visual difuso

### 5.4.1. Introducción

Se seleccionó este tipo de controlador pues tiene la ventaja de ser fácil de implementar y no requiere del modelo matemático del sistema, además dado que no fue posible quitar el controlador articular que tiene el robot A465 (ver Figura 5.5). Esta fuerte restricción dejó como única opción, el poder controlar al robot solamente mediante instrucciones de su lenguaje nativo llamado RAPL-II. Por ello, el objetivo principal de este controlador es probar el desempeño del modelo óptico de estereo visión. Otro tipo de plataformas como la presentada en [23], es adecuada para realizar otro tipo de estudios como el análisis del comportamiento de diferentes tipos de controladores.

En esta sección se presenta el procedimiento de diseño de un controlador visual difuso al sistema original mostrado en la Figura 5.5, para lo que se agrega un segundo lazo cerrado, en el cual, se incluye un sistema de visión por computadora y el controlador visual difuso, como se muestra en la Figura 5.6.

En este trabajo se le nombra controlador visual difuso a:

- Un controlador difuso que será diseñado mediante el procedimiento descrito en el capítulo anterior y es,
- Visual pues requiere el valor de la posición deseada, la cual es calculada mediante el modelo óptico de estereo visión descrito en el capítulo 4 y por ello se usará información visual para generar la acción de control.

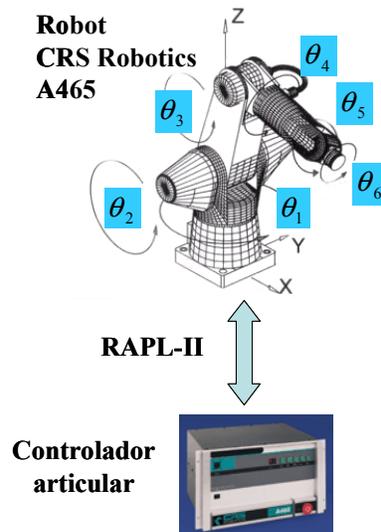


Figura 5.5: Sistema original formado por el robot A465 y su controlador PID articular.

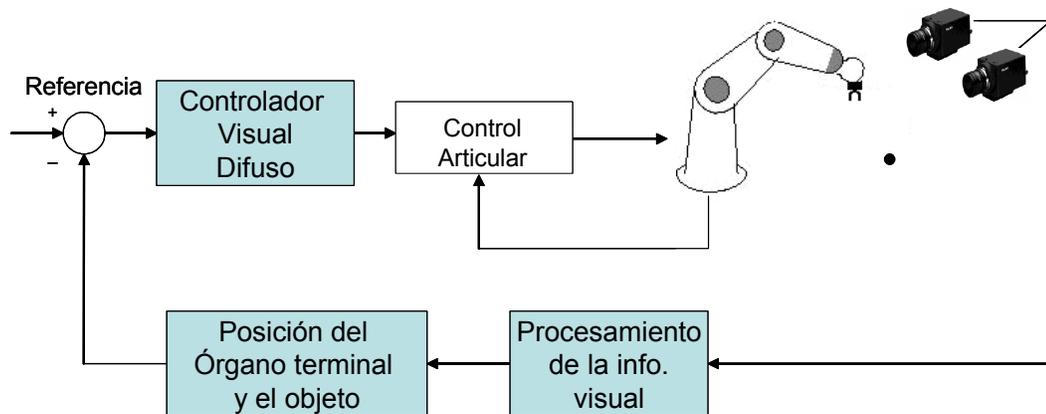


Figura 5.6: Esquema general de sistema de visión por computadora con el controlador visual difuso.

### 5.4.2. Desarrollo y resultados experimentales

En esta sección se presenta el diseño del controlador difuso mediante el esquema de prueba y error, dado que es un método iterativo, se usó matlab<sup>®</sup> para realizar el diseño y sintonización del sistema difuso. Una vez terminado, se programó en el lenguaje Borland C++<sup>®</sup> para realizar las pruebas con el sistema real.

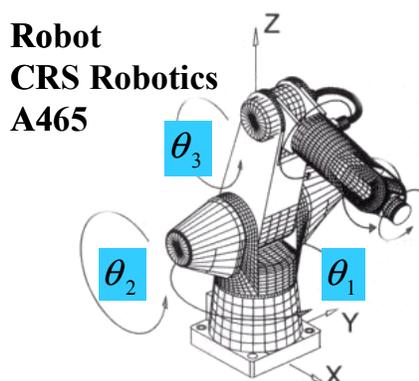


Figura 5.7: Las variables de control son los ángulos de las primeras tres articulaciones  $(\theta_1, \theta_2, \theta_3)$ .

Dado que el modelo óptico de estéreo visión nos da la posición deseada en el espacio, es decir, las coordenadas  $(x, y, z)$ . Si no consideramos la orientación del órgano terminal, sólo es necesario controlar los primeros tres grados de libertad del robot A465 como se muestra en la Figura 5.7. Además, se obtuvo un mejor desempeño del sistema usando tres controladores difusos uno para cada ángulo a controlar  $(\theta_1, \theta_2, \theta_3)$ . Para generar la acción del control, cada controlador difuso indica el valor angular (en radianes) que se requiere para mover cada ángulo. Con esta información se genera una sola instrucción del lenguaje RAPL-II que mueve al robot (ver Figura 5.8).

#### Paso 1: Elección de las variables de estado y de control

Se definen como variables de estados el error  $e$  y el cambio en el error  $\Delta e$  de la siguiente forma:

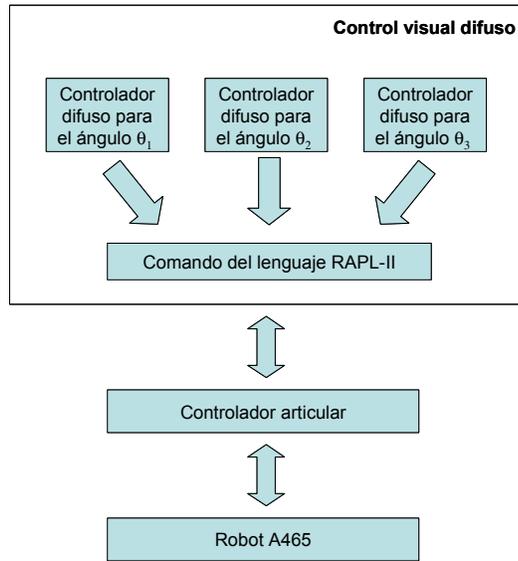


Figura 5.8: Esquema desglosado del controlador visual difuso.

$$\begin{aligned}
 e &= \theta_i^{actual} - \theta_i^* \\
 \Delta e &= e_i^{actual} - e_i^{anterior}, \quad i = 1, 2, 3
 \end{aligned}
 \tag{5.8}$$

donde

$\theta_i^{actual}$  es el valor actual del  $i$ -ésimo ángulo.

$\theta_i^*$  es el valor deseado del  $i$ -ésimo ángulo.

$e_i^{actual}$  es el valor actual del  $i$ -ésimo error.

$e_i^{anterior}$  es el valor anterior del  $i$ -ésimo error.

Éstas son las variables de entrada y serán las mismas para los tres controladores difusos. La variable de control será un valor angular para cada grado de libertad.

### Paso 2: Obtención de la base reglas difusas

Para este paso se comienza definiendo los conjuntos difusos para las variables de entrada-salida definidas en el paso anterior mediante el esquema de prueba y error. Se usaron todas las fuentes de información disponible: los manuales de operación del robot y la entrevista con el operador experto en el robot A465, después de realizar varias simulaciones en matlab<sup>®</sup> y luego las pruebas en el sistema real se obtuvieron los conjuntos difusos que se muestran en la Figura 5.9, donde las etiquetas difusas son:

1. Variables de entrada:

■ Error ( $e$ )

Etiqueta difusa	Descripción
GN1	Valor Grande Negativo
N1	Valor Negativo
C1	Valor Cero
P1	Valor Positivo
GP1	Valor Grande Positivo

■ Cambio en el error ( $\Delta e$ )

Etiqueta difusa	Descripción
GN2	Valor Grande Negativo
N2	Valor Negativo
C2	Valor Cero
P2	Valor Positivo
GP2	Valor Grande Positivo

2. Variable de salida

■ Valor angular ( $\gamma$ )

Etiqueta difusa	Descripción
GN3	Valor Grande Negativo
N3	Valor Negativo
C3	Valor Cero
P3	Valor Positivo
GP3	Valor Grande Positivo

Es importante mencionar que fue particularmente difícil ajustar el rango de la variable de control pues se comenzó con valores angulares muy grandes que provocaron comportamientos oscilatorios e incluso erráticos del robot, sin embargo, después de realizar la sintonización con ayuda del operador experto<sup>1</sup>, se obtuvo el rango adecuado.

Ahora se obtienen las reglas difusas y se llena la matriz difusa de la Figura 5.10.

### **Paso 3: Generación del controlador visual difuso y realización de las pruebas**

El controlador funciona de la siguiente forma:

1. El usuario introduce las coordenadas deseadas donde requiere ubicar al órgano terminal del robot.
2. Mediante la cinemática inversa se calcula la configuración angular requerida para que el órgano terminal del robot llegue a dicha posición.
3. Dado que el robot tiene un controlador a nivel articular se construyen tres bloques difusos, uno para cada grado de libertad.
4. Cada bloque genera el valor angular que su grado de libertad correspondiente se moverá.
5. Se unen estos tres valores angulares en una sola instrucción de RAPL-II, que se envía como señal de control.

---

<sup>1</sup>Ing. Dulce M. Reynoso, Coordinadora del Laboratorio de Robótica, Universidad La Salle, México D.F.

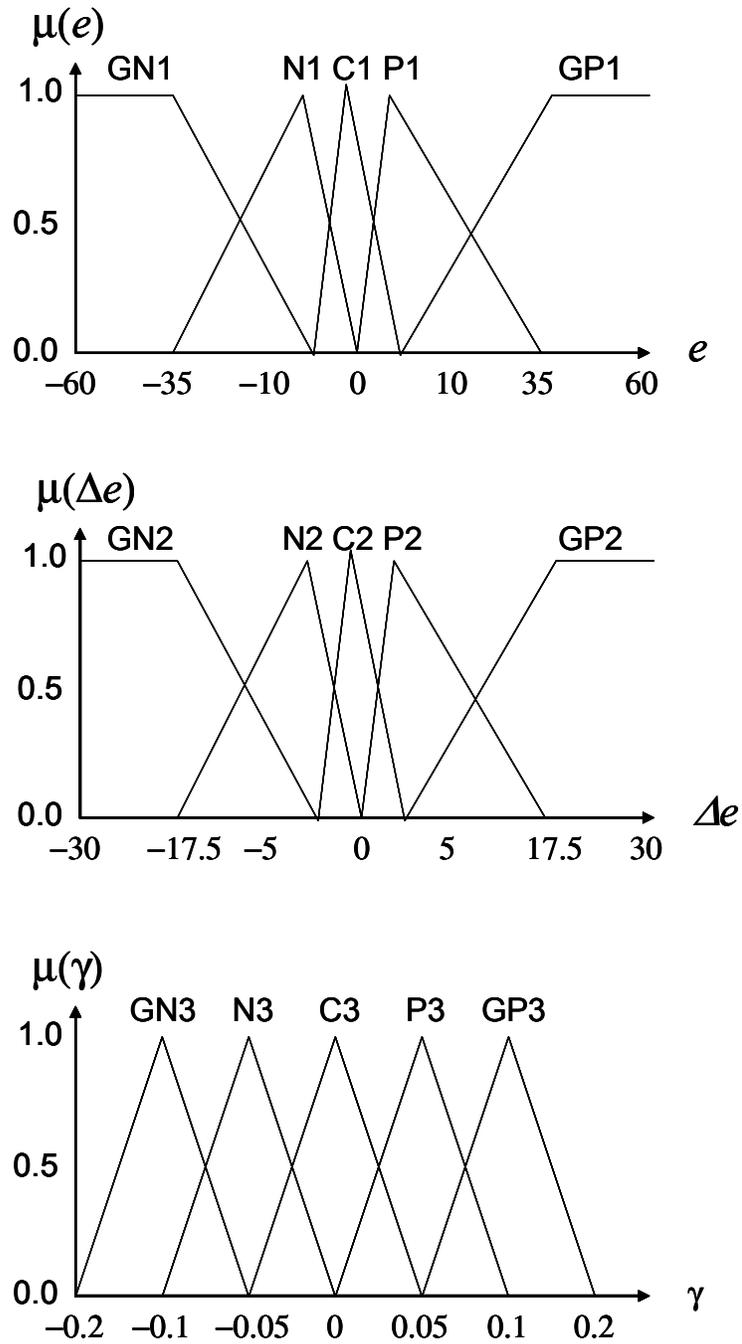


Figura 5.9: Conjuntos difusos para las variables  $e$ ,  $\Delta e$  y  $\gamma$ .

GN1	GP3	GP3	GP3	C3	C3
N1	GP3	P3	P3	P3	C3
C1	GP3	P3	C3	N3	GN3
P1	P3	C3	N3	N3	GN3
GP1	C3	C3	GN3	GN3	GN3
	GN2	N2	C2	P2	GP2

Figura 5.10: Matriz difusa llenada con las reglas difusas SI-ENTONCES.

- Se regresa al paso 4, hasta que el órgano terminal llegue a la posición deseada.

Todas estas rutinas se programaron en Borland C++<sup>®</sup> para realizar las pruebas de desempeño. En la Figuras 5.11 se muestra el resultado de usar este controlador visual difuso en el robot A465, donde se puede notar que existe un error de aproximadamente 3 cm. Con este error es posible trabajar en tareas que no requieran mucha precisión. Sin embargo, si se desea emplear para tareas de alta precisión se requiere continuar trabajando en la sintonización del controlador.

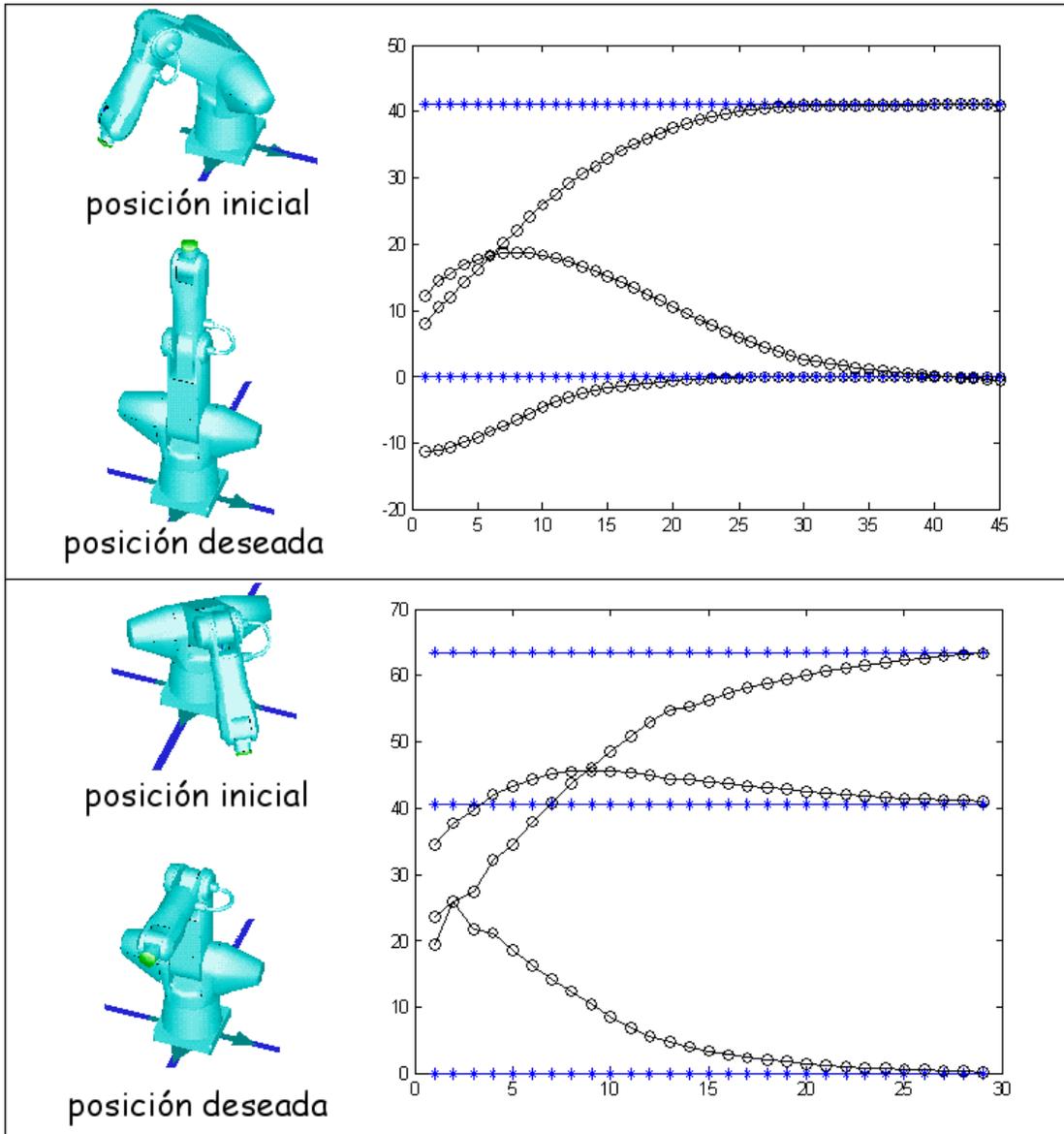


Figura 5.11: Desplazamientos del robot A465 para llegar a la posición deseada.



# Capítulo 6

## Análisis de Estabilidad: control visual PD

### 6.1. Introducción

En los capítulos anteriores se presentó el desarrollo teórico y experimental de la plataforma de estéreo visión con todos los algoritmos y el modelo óptico de estéreo visión. Adicionalmente, como parte final de este trabajo se presenta en este capítulo un análisis de estabilidad de diferentes controladores visuales para robots con dos y tres grados de libertad.

Se comienza con el caso de un robot con dos grados de libertad explicando el análisis de estabilidad para un controlador visual estándar, donde no existe ningún tipo de incertidumbre. Después se desarrolla la primera **aportación** de este capítulo que se refiere al análisis de estabilidad de un nuevo controlador visual que usa el valor estimado de la velocidad de las articulaciones del robot. Esta estimación es realizada mediante un observador de modos deslizantes.

Por otra parte, también se muestra el análisis de estabilidad de un controlador visual cuya ley de control incorpora una red neuronal multicapa para compensar los valores que se consideran desconocidos de las fuerzas de gravedad y fricción. Finalmente, se analiza el caso cuando los valores de la velocidad de las articulaciones del robot y las fuerzas de gravedad y

fricción son desconocidas. Este mismo caso se desarrolla para un robot de tres dimensiones que es la **contribución** principal del capítulo.

## 6.2. Robot de dos grados de libertad

### 6.2.1. Preliminares

#### Cinemática

La cinemática de un robot manipulador considera su posicionamiento relativo y no los efectos causados por el movimiento. Existe la cinemática inversa y directa. En la inversa dada la posición deseada del órgano terminal se calculan los valores angulares de las articulaciones del robot. En la directa dados los valores angulares de las articulaciones del robot se obtiene las coordenadas del órgano terminal del robot en el espacio de trabajo.

A partir de la dinámica directa de un robot planar de dos grados de libertad se obtiene la posición  $[x, y]^T \in \mathfrak{R}^2$  de su órgano terminal. Estas coordenadas son obtenidas con respecto al eje coordenado del robot en términos de la posición de sus articulaciones  $q(t) \in \mathfrak{R}^n$ , por lo que:

$$[x, y]^T = f(q)$$

donde  $f : \mathfrak{R}^n \longrightarrow \mathfrak{R}^2$ . La matriz Jacobiana  $J(\cdot) \in \mathfrak{R}^{2 \times n}$  del robot está definida como:

$$J(q) = \frac{\partial f(q)}{\partial q} \quad (6.1)$$

De la misma forma, para la posición deseada  $[x^*, y^*]^T \in \mathfrak{R}^2$ . Las coordenadas son obtenidas con respecto al eje coordenado del robot en términos de la posición de sus articulaciones  $q^*(t) \in \mathfrak{R}^n$ , por lo que:

$$[x^*, y^*]^T = f(q^*)$$

donde  $f : \mathfrak{R}^n \longrightarrow \mathfrak{R}^2$ . La matriz Jacobiana  $J(\cdot) \in \mathfrak{R}^{2 \times n}$  del robot está definida como:

$$J(q^*) = \frac{\partial f(q^*)}{\partial q} \quad (6.2)$$

Así  $\dot{X} = J\dot{Q}$ , donde  $J$  es el Jacobiano del robot. Para realizar el análisis de estabilidad se presentan las siguientes suposiciones sobre la matriz Jacobiana:

**Suposicin 6.1** *La matriz Jacobiana  $J(q)$  es cuadrada e invertible*

**Suposicin 6.2** *La matriz Jacobiana  $J(q)$  se encuentra acotada para toda  $q(t) \in \mathfrak{R}^n$ .*

### Modelo de la cámara

Dado que se trata de un robot de dos grados de libertad con una cámara de video es posible obtener la posición de su órgano terminal visualmente. Para ello se utiliza el modelo presentado en [43] donde la posición  $x_s$  del órgano terminal que se mueve en el plano en coordenadas de pantalla se calcula con:

$$\begin{bmatrix} x_s \\ y_s \end{bmatrix} = \alpha h R(\theta) \left\{ \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} O_1 \\ O_2 \end{bmatrix} \right\} + \begin{bmatrix} C_x \\ C_y \end{bmatrix}$$

donde

$\alpha$  es el factor de conversión de pixel a centímetro (pixels/cm).

$h$  es la proyección perspectiva que depende de  $\lambda$  que es la distancia focal y  $z$  es la distancia entre la cámara y el robot.

$R(\theta)$  es la matriz de rotación de la cámara sobre su eje óptico en el sentido del reloj con respecto al eje coordenado del espacio de trabajo del robot.

$\begin{bmatrix} O_1 & O_2 \end{bmatrix}^T$  son las coordenadas del punto de intersección de eje  $C_3$  en el plano  $X_1 - X_2$ .

$\begin{bmatrix} C_x & C_y \end{bmatrix}^T$  son las coordenadas del centro de la imagen.

De la misma manera, la posición deseada  $x_s^*$  del órgano terminal en el sistema de coordenadas del robot, se escribe como:

$$\begin{bmatrix} x_s^* \\ y_s^* \end{bmatrix} = \alpha h R(\theta) \left\{ \begin{bmatrix} x^* \\ y^* \end{bmatrix} - \begin{bmatrix} O_1 \\ O_2 \end{bmatrix} \right\} + \begin{bmatrix} C_x \\ C_y \end{bmatrix}$$

El error de posición medido por la cámara se nombra como  $\tilde{x}_s$  y se define como:

$$\tilde{x}_s = \begin{bmatrix} x_s^* \\ y_s^* \end{bmatrix} - \begin{bmatrix} x_s \\ y_s \end{bmatrix} = \alpha h R(\theta) \left( \begin{bmatrix} x^* \\ y^* \end{bmatrix} - \begin{bmatrix} x \\ y \end{bmatrix} \right)$$

El error de posición se puede reescribir como:

$$\tilde{x}_s = \alpha h R(\theta) [f(q^*) - f(q)] \quad (6.3)$$

y el error en velocidad es:

$$\dot{\tilde{x}}_s = \alpha h R(\theta) [J(q^*) \dot{q}^* - J(q) \dot{q}] \quad (6.4)$$

para el caso del problema de regulación el término  $J(q^*) = 0$ , así:

$$\dot{\tilde{x}}_s = -\alpha h R(\theta) J(q) \dot{q} \quad (6.5)$$

## Dinámica

La dinámica de un robot corresponde a las ecuaciones de movimiento, el medio por el cual, el robot se mueve en respuesta a los pares aplicados por el actuador o fuerzas externas. Para un robot manipulador con n-articulaciones rígidas puede ser descrita como [46]:

$$M(q) \ddot{q} + C(q, \dot{q}) \dot{q} + G(q) + F_d(\dot{q}) = \tau \quad (6.6)$$

donde:

$q$  es la posición de la articulación.

$\dot{q}$  es la velocidad de la articulación.

$\tau$  es el vector de control.

$M(q)$  es la matriz de inercia.

$C(q, \dot{q})$  es la matriz de coriolis y centrípeta.

$G(q)$  es el vector de gravedad.

$F_d(\dot{q})$  es el vector de de fricción.

La dinámica de un robot puede ser obtenida mediante varios métodos como el Euler-Lagrange y Newton-Euler [47], los cuales pueden ser descritos con respecto a los parámetros de Denavit Hartenberg [21]. Está dinámica satisface las siguientes propiedades físicas.

**Propiedad 6.1** *La matriz de inercia es simétrica y definida positiva, i. e.*

$$m_1 \|q\|^2 \leq q^T M(q)q \leq m_2 \|q\|^2; \forall q \in R^n$$

donde  $m_1, m_2$  son dos constantes escalares positivas conocidas, y  $\|\cdot\|$  representa la norma Euclidiana para el vector  $q$ .

**Propiedad 6.2** *La matriz de coriolis y centrípeta puede ser seleccionada de tal forma que:*

$$q^T [M(\dot{q}) - 2C(q, \dot{q})]q = 0; \forall q, \dot{q} \in R^n$$

**Propiedad 6.3** *La matriz de coriolis y centrípeta satisface la siguiente relación:*

$$C(q, x)y = C(q, y)x; \forall x, y \in R^n$$

**Propiedad 6.4** *La matriz de coriolis y centrípeta puede ser escrita de la siguiente forma:*

$$C(q, \dot{q}) = \sum_{k=1}^n C_{k,ij}(q)\dot{q}_k$$

donde:  $C_{k,ij}(q) = \left( \frac{\partial M_{ij}}{\partial q_k} + \frac{\partial M_{ik}}{\partial q_j} - \frac{\partial M_{jk}}{\partial q_i} \right)$

**Propiedad 6.5** *La matriz de coriolis y centrípeta satisface la siguiente relación:*

$$\|C(q, \dot{q})\|_{i\infty} \leq K_c \|\dot{q}\|$$

donde  $K_c = \frac{1}{2} \max_{q \in \mathbb{R}^n} \sum_{k=1}^n \|C_k(q)\|$  y  $\|\cdot\|_{i\infty}$  representa la matriz inducida por la norma infinita [67].

### Planteamiento del problema de control

**Suposicin 6.3** *La posición deseada o objetivo  $[x^* \ y^*]^T \in \mathbb{R}^2$  es conocida y estática (caso de regulación), por lo que:*

$$J(q^*) = \frac{\partial f(q^*)}{\partial q} = 0$$

**Suposicin 6.4** *La posición deseada se encuentra dentro del espacio de trabajo del robot y de ambas imagenes del sistema de visión por computadora, del tal forma que existe al menos un vector de posición  $\mathbf{x} = \begin{bmatrix} x & y \end{bmatrix}^T$ , con una cierta configuración de articulaciones  $q^* \in \mathbb{R}^2$  tal que el órgano terminal puede alcanzarla.*

**Comentario 6.1** *Si la anterior suposición no se cumple no es posible llevar al órgano terminal del robot a la posición deseada.*

**Problema 6.1** *El problema de regulación en el espacio visual consiste en obtener una ley de control visual tal que para cualquier condición inicial  $x_s(t_o)$  en el tiempo  $T(\eta_C, x_s(t_o))$  se tenga:*

$$\|x_s(t) - x_s^*\| \leq \eta_C \quad \forall t_o + T(\eta_C, x_s(t_o))$$

donde  $\eta_C$  es una constante positiva y  $x_s^*$  es la posición deseada.

### 6.2.2. Controlador visual estándar

En esta sección se diseñará una ley de control para un robot de dos grados de libertad cuya estructura se describe en la ec. (6.6) considerando que se puede medir exactamente todos los parámetros que intervienen en el sistema.

## Introducción

El control visual tiene la capacidad de proveer una solución de automatización de bajo costo y mantenimiento para ambientes industriales que requieren mayor flexibilidad en sus procesos. El incorporar un sistema de visión por computadora en el lazo de control de un robot manipulador se conoce como *visual servoing* o *control visual* [33]. La mayor parte de las investigaciones en este campo se encuentran en las siguientes categorías:

- **Control basado en la posición** [42]: En esta la información visual es interpretada con respecto a las coordenadas en el espacio del sistema. Requiere conocer completamente el modelo cinemático del robot, la localización exacta del objetivo o posición deseada y un modelo preciso de calibración para el sistema de visión por computadora. Todos estos requerimientos hace que esta técnica sea muy difícil de implementar en ambientes no estructurados.
- **Control basado en la imagen** [22]: En esta se usa información obtenida directamente del sistema de visión por computadora (plano de la imagen) como entrada al controlador. Está técnica es la usada en este trabajo.

Es bien conocido que la mayoría de los robots industriales están equipados con un controlador PD sencillo. Varios esquemas modificados de este tipo de controlador han sido publicados en [60], [53]. La técnica de control visual usa información visual y tiene una estructura similar al control PD [33]. La estabilidad de está técnica ha sido considerado por diferentes investigadores. En [39] y [76] se presenta la estabilidad robusta de un controlador visual de dos dimensiones con un controlador tipo PD. En [8] y [28] se describe un controlador visual robusto de tres dimensiones, para lo que se destinan controladores no-lineales.

## Análisis de estabilidad

La ley de control visual quedaría como:

$$\tau = JK_p \tilde{x}_s - K_d \dot{q} + (G + F_d) \quad (6.7)$$

donde  $K_p$  y  $K_d$  son matrices positivas definidas y simétricas.

**Teorema 6.1** *Si se aplica la ley de control visual (6.7) a un robot de dos grados de libertad (6.6), el error de posición  $\tilde{x}_s$  es acotado.*

**Demostracin.** Se propone la siguiente función candidata de Lyapunov:

$$V = \frac{1}{2} \dot{q}^T M \dot{q} + \frac{1}{2\alpha h} R^{-1} \tilde{x}_s^T K_p \tilde{x}_s$$

donde  $\dot{q}^T M \dot{q}$  es la energía cinética del robot. Su derivada en tiempo es:

$$\dot{V} = \dot{q}^T M \ddot{q} + \frac{1}{2} \dot{q}^T \dot{M} \dot{q} + \frac{1}{\alpha h} R^{-1} \dot{\tilde{x}}_s^T K_p \tilde{x}_s$$

y despejando  $\ddot{q}$  de (6.6):

$$\ddot{q} = M^{-1}(q) \left( \tau - C(q, \dot{q}) \dot{q} - G(q) - F_d(\dot{q}) \right) \quad (6.8)$$

entonces

$$\dot{V} = \frac{1}{2} \dot{q}^T \left\{ 2\tau - 2C(q, \dot{q}) \dot{q} - 2[G(q) + F_d(\dot{q})] + \dot{M} \dot{q} \right\} + \frac{1}{\alpha h} R^{-1} \dot{\tilde{x}}_s^T K_p \tilde{x}_s$$

usando (6.5), la *Propiedad 6.2* y para el caso de regulación, donde se cumple que  $J(q^*) = \frac{\partial K(q^*)}{\partial q} = 0$ , la derivada  $\dot{V}$  se reduce a:

$$\dot{V} = \dot{q}^T [\tau - (G + F_d) - JK_p \tilde{x}_s]$$

Seleccionando la ley de control visual como:

$$\begin{aligned} \tau - (G + F_d) - JK_p \tilde{x}_s &= -K_d \dot{q}, \quad K_d^T = K_d > 0 \\ \tau &= JK_p \tilde{x}_s - K_d \dot{q} + (G + F_d) \end{aligned}$$

sustituyéndola en  $\dot{V}$  queda finalmente:

$$\dot{V} = -\dot{q}^T K_d \dot{q} \leq 0$$

Así  $\dot{V}$  es semidefinida negativa. Ahora usando el teorema 4.5 mencionado en [49] (Teorema de estabilidad exponencial), el sistema es estable exponencialmente. ■

**Comentario 6.2** *Para este caso se usó la ley de control visual ideal (6.7), dado que requiere conocer información sobre la gravedad  $G$  y la fricción  $F$ , además no existe ninguna incertidumbre en la estructura del robot, i.e., el Jacobiano  $J$  se conoce exactamente. Así, esta ley de control puede garantizar que el griper del robot alcanzará la posición deseada. Si los valores de  $G$ ,  $F$  y  $J$  no son conocidas exactamente, se puede generar un error que lleve al robot a otra posición distinta a la deseada o hacer al sistema completamente inestable.*

### 6.2.3. Controlador visual Proporcional-Derivativo con observador

En esta sección se diseña una ley de control para un robot de dos grados de libertad (6.6) considerando que no se puede medir  $\dot{q}$  por lo cual, se usará un observador de modos deslizantes.

#### Introducción

La mayor parte de los robots manipuladores industriales están equipados con un controlador PD que tiene la forma

$$\tau = -K_p [q - q^*] - K_d [\dot{q} - \dot{q}^*] \quad (6.9)$$

Varias modificaciones a los esquemas modificados para este tipo de controlador con resultados experimentales satisfactorios han sido presentados en [60], [53]. En el caso de controladores visuales en esquema de control tipo PD es también muy popular [39], [22], [31] y se puede expresar como:

$$\tau = -K_p [x_s - x_s^*] - K_d [\dot{q} - \dot{q}^*] \quad (6.10)$$

Este tipo de controladores visuales se requiere de mediciones de la posición y velocidad de las articulaciones del robot. La velocidad de las articulaciones es medida usualmente mediante un tacómetro, el cual, es muy caro y sus mediciones muchas veces se ven contaminadas por ruido [41]. Por ello, es muy importante desarrollar un controlador visual con esta estructura empleando solamente la información de la posición de las articulaciones de robot por lo que se requiere utilizar un observador para estimar los valores de  $\dot{q}$  y  $\dot{q}^*$ .

Existen dos tipos de observadores que pueden ser usados:

1. El observador basado en el modelo del sistema que asume que el modelo dinámico del robot es parcial o completamente conocido. Por ejemplo, si el modelo del objeto en movimiento está disponible, un observador de Luenberger [25] puede ser usado. Un predictor de posición de ventana basado en un observador no-lineal es propuesto

en [56]. Un controlador visual basado en un observador no-lineal fue presentado por primera vez en [31]. Si la fuerza centrífuga y la fuerza de coliorisis son despreciadas, un observador linealizado es más sencillo de aplicar al controlador visual [32].

2. El observador libre de modelo no requieren conocimiento preciso de las características del robot. Los observadores mas populares de este tipo son los de alta ganancia, éstos pueden estimar la derivada de la salida del sistema [52]. Recientemente, un observador de este tipo basado en redes neuronales fue presentado en [41], donde las no-linealidades fueron estimadas mediante redes neronales estáticas.

En esta sección se presenta un nuevo algoritmo de control visual cuando sólo se cuenta con la medición de la posición de las articulaciones, por lo que se emplea un observador libre de modelo conocido como observador de modos deslizantes que se encarga de estimar la velocidad de las articulaciones del robot. La velocidad estimada se incorpora en la nueva ley de control y se obtiene la estabilidad en lazo cerrado del sistema mediante el método de Lyapunov.

### Observador de modos deslizantes

La dinámica del robot (6.6) puede ser expresada como:

$$\begin{aligned}\dot{x}_t &= f(x_t, \tau) \\ y_t &= Cx_t\end{aligned}\tag{6.11}$$

donde  $x_t = [q^T, \dot{q}^T]^T \in \mathfrak{R}^{2n}$ , y

$$f(x_t, \tau) = \begin{bmatrix} \dot{q} \\ M^{-1}(q) \left[ \tau - C(q, \dot{q}) \dot{q} - G(q) - F_d(\dot{q}) \right] \end{bmatrix}\tag{6.12}$$

Se considera que solamente la posición  $q$  es medible, por lo tanto se tiene que  $C = [I, 0]$ . Transformando el sistema (6.11) en la forma normal:

$$\begin{aligned}\dot{x}_t &= Ax_t + F(x_t, u_t) \\ y_t &= Cx_t\end{aligned}\tag{6.13}$$

donde  $F(x_t, u_t) := f(x_t, u_t) - Ax_t$ , con  $A$  tal que el par  $(A, C)$  es observable.

Se construye al observador de modos deslizantes como:

$$\begin{aligned}\dot{\bar{x}}_t &= A\bar{x}_t + S(\bar{x}_t, e) - Ke_t \\ \bar{y}_t &= C\bar{x}_t\end{aligned}\tag{6.14}$$

donde  $\bar{x}_t = \left[ \hat{q}^T, \dot{\hat{q}}^T \right]^T$  y  $e_t$  es el error de salida definido como:

$$e_t := \bar{y}_t - y_t = C\bar{\Delta}_t = C(\bar{x}_t - x_t),$$

donde  $\bar{\Delta}_t$  es el error de observación y  $S(\bar{x}_t, e_t)$  es seleccionada como:

$$S(\bar{x}_t, e_t) = -\rho \frac{P^{-1}C^T C \bar{\Delta}_t}{\|C \bar{\Delta}_t\|} = -\rho P^{-1}C^T \text{sign}(e_t)\tag{6.15}$$

donde  $P$  es una matriz definida positiva y  $\rho$  es una constante positiva. Claramente se puede ver que el observador de modos deslizantes (6.14) no depende del sistema no-lineal (6.11), solamente la salida  $y$  es necesaria.

La derivada del error de observación es:

$$\begin{aligned}\dot{\bar{\Delta}}_t &= A\bar{\Delta}_t + S(\bar{x}_t, e_t) - KC\bar{\Delta}_t - F(x_t, u_t) \\ &= (A - KC)\bar{\Delta}_t + S(\bar{x}_t, e_t) - F(x_t, u_t) \\ &= A_0\bar{\Delta}_t + S(\bar{x}_t, e_t) - F(x_t, u_t)\end{aligned}\tag{6.16}$$

donde  $A_0 := A - KC$ .

Dado que  $(A, C)$  es observable, existe  $K$  tal que  $A_0$  es estable. Por esto, la siguiente ecuación de Lyapunov tiene una solución positiva  $P$  para una matriz definida positiva  $Q$ :

$$A_0^T P + P A_0 = -Q, \quad Q = Q^T > 0$$

Dado que el sistema (6.12) es mecánico,  $f(x_t, u_t)$  es acotada

**Suposicin 6.5** *Sea:*

$$f(x_t, u_t) - Ax_t = -P^{-1}C^T h(x_t, u_t)$$

donde  $h(x_t, u_t)$  es acotada.

**Teorema 6.2** *Considerando la suposición 1, si la ganancia del observador satisface,*

$$\rho > \|h(x_t, u_t)\|$$

*el error entre el observador de modos deslizantes (6.14) y el robot descrito por (6.11) es asintóticamente estable, por lo que:*

$$\lim_{t \rightarrow \infty} \bar{\Delta}_t = 0 \quad (6.17)$$

**Demostracin.** Se propone la siguiente función candidata de Lyapunov:

$$V_t = \bar{\Delta}_t^T P \bar{\Delta}_t$$

Calculando su derivada:

$$\dot{V}_t = \bar{\Delta}_t^T (A_0^T P + P A_0) \bar{\Delta}_t + 2 \bar{\Delta}_t^T P [S(\bar{x}_t, e_t) - F(x_t, u_t)].$$

Usando la *Suposición 6.5*, se tiene:

$$F(x_t, u_t) = -P^{-1} C^T h(x_t, u_t), \quad \|h(x_t, u_t)\| < \rho.$$

Seleccionando a  $S(\bar{x}_t, e_t)$  como en (6.15), se obtiene:

$$\begin{aligned} \dot{V}_t &= -\bar{\Delta}_t^T Q \bar{\Delta}_t + 2 \bar{\Delta}_t^T C^T h(x_t, u_t) - 2 \frac{\bar{\Delta}_t^T C^T C \bar{\Delta}_t}{\|C \bar{\Delta}_t\|} \rho \\ &= -\bar{\Delta}_t^T Q \bar{\Delta}_t + 2 \bar{\Delta}_t^T C^T h(x_t, u_t) - 2 \|C \bar{\Delta}_t\| \rho \\ &\leq -\bar{\Delta}_t^T Q \bar{\Delta}_t + 2 \|C \bar{\Delta}_t\| (\|h(x_t, u_t)\| - \rho) \leq 0 \end{aligned}$$

Dado que  $\dot{V}_t \leq 0$ ,  $\bar{\Delta}_t \in L_\infty$ . De (6.16) del error, también se concluye que  $\dot{\bar{\Delta}}_t \in L_\infty$ . Como  $\dot{V}_t \leq -\bar{\Delta}_t^T Q \bar{\Delta}_t$  y  $V_t$  es un proceso acotado,  $\bar{\Delta}_t$  es cuadráticamente integrable y acotado  $\bar{\Delta}_t \in L_2$ . Usando el lema de Barbalat se obtiene que el error de observación  $\bar{\Delta}_t$  es asintóticamente estable, por lo tanto,  $\lim_{t \rightarrow \infty} \bar{\Delta}_t = 0$ . ■

El diagrama de diseño para el observador de modos deslizantes se muestra en la Figura 6.1.

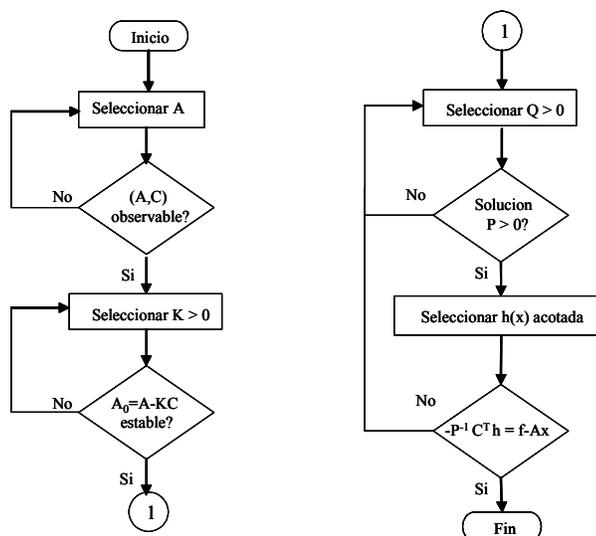


Figura 6.1: Diagrama de flujo para diseñar el observador.

### Análisis de estabilidad

Del *Teorema 6.2* se obtiene que  $\hat{q} = \dot{q}$  cuando  $t \rightarrow \infty$ , ahora para cualquier valor de tiempo el error de observación para  $\dot{q}$  está definido por:

$$\hat{q} = \dot{q} + \varepsilon$$

**Cota 6.1**  $\varepsilon$  está acotada de la siguiente manera:

$$\varepsilon^T K_d^T \Lambda_\varepsilon K_d \varepsilon \leq \bar{\eta}_\varepsilon \quad (6.18)$$

donde  $\Lambda_\varepsilon$  es una matriz definida positiva y  $\bar{\eta}_\varepsilon$  una constante definida positiva.

Así cuando  $\dot{q}$  de la ley de control visual (6.7) es estimada como  $\hat{q}$  por el observador de modos deslizantes (6.14). Se propone la siguiente ley de control visual:

$$\tau = JK_p \tilde{x}_s - K_d \hat{q} + G(q) + F_d(q) \quad (6.19)$$

donde  $K_p$ ,  $K_d$  son matrices definidas positivas. El siguiente teorema presenta la prueba de estabilidad de la ley de control visual (6.19).

**Teorema 6.3** *Considere la ley de control visual (6.19) y  $K_d$  es selecciona lo suficientemente grande, tal que:*

$$K_d > \Lambda_\varepsilon^{-1}$$

*y se usa el siguiente control de zona muerta, el error de posición  $\tilde{x}_s$  es acotado.*

- *si  $\|\dot{q}\|^2 > \lambda_{\min}^{-1}(Q)(\bar{\eta}_\varepsilon)$  entonces, se usa la ley de control visual (6.19)*
- *si  $\|\dot{q}\|^2 \leq \lambda_{\min}^{-1}(Q)(\bar{\eta}_\varepsilon)$  entonces, se deja de usar la ley de control visual (6.19)*

*donde  $Q = K_d - \Lambda_\varepsilon^{-1} > 0$ , entonces el error de seguimiento es acotado.*

**Demostracin.** Se propone la siguiente función candidata de Lyapunov:

$$V = \frac{1}{2} \dot{q}^T M \dot{q} + \frac{1}{2\alpha h} R^{-1} \tilde{x}_s^T K_p \tilde{x}_s$$

Su derivada en tiempo es:

$$\dot{V}_t = \dot{q}^T M \ddot{q} + \frac{1}{2} \dot{q}^T \dot{M} \dot{q} + \tilde{x}_s^T K_p \dot{\tilde{x}}_s$$

Sustituyendo  $\ddot{q}$  de (6.8),

$$\dot{V}_t = \frac{1}{2} \dot{q}^T \left( 2\tau - 2C(q, \dot{q}) \dot{q} - 2[G(q) + F_d(q)] + \dot{M} \dot{q} \right) + \tilde{x}_s^T K_p \dot{\tilde{x}}_s$$

Usando la *Propiedad 6.2* y con (6.5), se tiene:

$$\dot{V}_t = \dot{q}^T \{ \tau - G(q) - F_d(q) - J^T K_p \tilde{x}_s \}$$

Aplicando la ley de control visual (6.19), se obtiene:

$$\dot{V}_t = \dot{q}^T \left\{ -K_d \dot{q} - K_d \varepsilon + J^T K_p \tilde{x}_s - J^T K_p \tilde{x}_s \right\}$$

y

$$\dot{V} = -\dot{q}^T K_d \dot{q} - \dot{q}^T K_d \varepsilon \quad (6.20)$$

Usando la desigualdad matricial:

$$X^T Y + (X^T Y)^T \leq X^T \Lambda^{-1} X + Y^T \Lambda Y \quad (6.21)$$

donde  $X, Y$  son dos matrices cualesquiera,  $\Lambda$  es definida positiva. El término  $\dot{q}^T K_d \varepsilon$  puede ser estimado como:

$$\dot{q}^T K_d \varepsilon \leq \dot{q}^T \Lambda_\varepsilon^{-1} \dot{q} + \bar{\eta}_\varepsilon$$

Entonces (6.20) queda como:

$$\begin{aligned} \dot{V}_t &\leq -\dot{q}^T (K_d - \Lambda_\varepsilon^{-1}) \dot{q} + \bar{\eta}_\varepsilon \\ \dot{V}_t &\leq -\dot{q}^T Q \dot{q} + \bar{\eta}_\varepsilon \end{aligned}$$

donde  $Q = K_d - \Lambda_\varepsilon^{-1} > 0$ .

*Caso 1:* Si  $\left\| \dot{q} \right\|^2 > \lambda_{\min}^{-1}(Q) \bar{\eta}_\varepsilon$  entonces, se usa la ley de control visual y se concluye que:

$$\dot{V}_t \leq -\dot{q}^T Q \dot{q} + \bar{\eta}_\varepsilon \leq -\lambda_{\min}(Q) \left\| \dot{q} \right\|^2 + \bar{\eta}_\varepsilon < 0 \quad (6.22)$$

por lo tanto,  $V_t$  es acotada.

*Caso 2:* Si  $\left\| \dot{q} \right\|^2 \leq \lambda_{\min}^{-1}(Q) \bar{\eta}_\varepsilon$ , el término  $\frac{1}{2} \dot{q}^T M \dot{q}$  es acotado. Dado que la acción de control se deja de aplicar, el término  $\frac{1}{2\alpha h} R^{-1} \tilde{x}_s^T K_p \tilde{x}_s$  se mantiene constante y  $V_t$  también se mantiene constante.

Entonces  $V_t$  es acotado,  $\dot{q}$  y  $\tilde{x}_s$  son acotados. ■

### 6.2.4. Controlador visual Proporcional-Derivativo con Redes Neuronales Multicapa

En esta sección se diseña una ley de control para un robot de dos grados de libertad cuya estructura se describe en (6.6), considerando que no se puede medir  $G$  y  $F_d$  por lo cual, se usará una red neuronal multicapa para estimar sus valores.

#### Introducción

Dado que la fuerza de fricción y gravedad afecta las propiedades dinámicas del controlador PD, dos tipos de compensación pueden ser usados para un controlador PD estándar. La estabilidad global asintótica del controlador PD con compensación de gravedad se presenta en [62]. Si alguno de los parámetros en el vector del torque gravitacional es desconocido, una versión adaptable del controlador PD con compensador de gravedad es introducido en [66].

El controlador PID no requiere ningún componente de la dinámica del robot en su ley de control, sin embargo le falta una prueba de estabilidad global asintótica [40]. Agregando acciones integrales, la estabilidad global asintótica del controlador PD es propuesta en [37] y [57].

Usar redes neuronales para compensar el torque gravitacional y posibles incertidumbres en el robot es factible, dado que ninguna información de la estructura o modelo es necesaria [29], [41], [46]. En esta sección se presenta una extensión del controlador neuronal tipo PD para control visual que usa redes neuronales con funciones radiales básicas (RBF) presentado en [29], [44].

El control visual requiere usualmente de mover al griper del robot a la posición deseada, por lo que, el problema de control necesita una transformación del espacio de la tarea al espacio de las articulaciones. Para ello, se requiere el conocimiento exacto de la matriz Jacobiana. Diferentes esfuerzos se han realizado para hacer estable el sistema cuando se tienen incertidumbres en el Jacobiano [12], [13]. En estos trabajos los sistemas originales se convierten a sistemas auxiliares, tal que los controladores robustos puedan estabilizarlos.

Aquí, se presenta un nuevo controlador visual tipo PD con compensación directa (gravedad

y fricción) usando redes neuronales diferenciales. Se consideran dos tipos de incertidumbres. Primero, que no se conoce exactamente la matriz Jacobiana del robot y segundo, que las coordenadas del espacio y las imágenes no coinciden exactamente.

## Preliminares

**Incetidumbre en la matriz Jacobiana** La matriz Jacobiana  $J(q)$  de un robot manipulador puede ser caracterizada por un conjunto de parámetros tales como las longitudes de sus articulaciones, los de la cámara y las longitudes de los objetos [3]. Si alguno de éstos es desconocido o sólo se tiene su valor estimado se cuenta solamente con un estimado de esta matriz.

Este problema es importante de analizar, pues sería deseable que el robot manipulador tuviera la habilidad de trabajar en ambientes no estructurados donde un brazo humano es capaz de realizar tareas complejas muy hábilmente. El error de estimación es acotado debido a que cada elemento de la matriz Jacobiana es constante y con funciones trigonométricas de  $q$  (senoidal y cosenoidal) que son suaves y por lo tanto acotadas.

**Cota 6.2** *Se define a  $\hat{J}(q)$  como la estimación del real  $J(q)$ , además se asume que el error de estimación  $\eta_J$  es acotado como:*

$$\begin{aligned} J(q) - \hat{J}(q) &= \eta_J \\ \eta_J^T \Lambda_J \eta_J &\leq \bar{\eta}_J \end{aligned} \quad (6.23)$$

donde  $\bar{\eta}_J$  es la cota superior de la incertidumbre en la matriz Jacobiana y  $\Lambda_J = \Lambda_J^T > 0$ .

## Red neuronal multicapa

**Definición 6.1** *La red neuronal multicapa puede expresarse de la siguiente forma:*

$$y = W\sigma(Vx) \quad (6.24)$$

donde  $x = [x_1, x_2, \dots, x_n]^T$ ,  $y = [y_1, y_2, \dots, y_m]^T$ . Las matrices de pesos se definen como:  $W = [w_{ij}] \in R^{m \times k}$ ,  $V = [v_{jk}] \in R^{k \times n}$ ,  $\sigma(z) = [\sigma(z_1), \sigma(z_2), \dots, \sigma(z_k)]$ .  $k$  es el número de nodos de la capa oculta. La función de activación  $\sigma(z)$  es no-lineal acotada.

De acuerdo con el teorema de Stone-Weierstrass [20], los componentes de fricción y gravedad que aparecen (6.6) pueden ser descritos como (6.25) y el error de aproximación  $\eta_g$  está acotado. La magnitud de  $\eta_g$  depende de los valores de  $W^*$  y  $V^*$ .

$$G(q) + F_d(\dot{q}) = W^* \sigma(V^* x) + \eta_g \quad (6.25)$$

donde  $x = [q^T, \dot{q}^T]^T$ ,  $W^*$ ,  $V^*$  son cotas fijas para las matrices de pesos.

**Cota 6.3**  $\eta_g$  está acotado de la siguiente forma:

$$\eta_g^T \Lambda_g \eta_g \leq \bar{\eta}_g \quad (6.26)$$

donde  $\Lambda_g$  es una matriz definida positiva y  $\bar{\eta}_g$  es una constante positiva.

Los componentes de fricción y gravedad se pueden estimar como:

$$G(x) + F(x) \approx \widehat{W}_t \sigma(\widehat{V}_t x)$$

donde  $\widehat{W}_t$  y  $\widehat{V}_t$  son las matrices de pesos variantes en el tiempo de la red neuronal diferencial.

Se tiene la siguiente relación:

$$W^* \sigma(V^* x) - \widehat{W}_t \sigma(\widehat{V}_t x) = \widetilde{W}_t \sigma(V_t x) + W^* \tilde{\sigma} = \widetilde{W}_t \sigma(V_t x) + W^* D_\sigma \widetilde{V}_t x + \nu_\sigma \quad (6.27)$$

La función sigmoide, que es comúnmente usada como función de activación, satisface la condición de Lipschitz:

$$\tilde{\sigma} = \sigma(V^{*T} x) - \sigma(\widehat{V}_t^T x) = D_\sigma \widetilde{V}_t^T x + \nu_\sigma \quad (6.28)$$

donde  $\widetilde{W}_t = W^* - \widehat{W}_t$ ,  $\widetilde{V}_t = V^* - \widehat{V}_t$ ,  $D_\sigma = \frac{\partial \sigma^T(Z)}{\partial Z} \Big|_{Z=\widehat{V}_t^T x}$ ,

**Cota 6.4**  $\nu_\sigma$  está acotado de la siguiente forma:

$$\|\nu_\sigma\|_{\Lambda_\sigma}^2 = \nu_\sigma^T \Lambda_\sigma \nu_\sigma \leq \bar{\eta}_\sigma \quad (6.29)$$

donde  $\Lambda_\sigma$  es una matriz definida positiva y  $\bar{\eta}_\sigma$  es una constante positiva.

### Análisis de estabilidad

La ley de control visual con compensación mediante una red neuronal multicapa es:

$$\tau = \widehat{J}K_p\tilde{x}_s - K_d\dot{q} + \widehat{W}_t\sigma(\widehat{V}_tx) \quad (6.30)$$

donde  $K_p$  y  $K_d$  son matrices positivas definidas y simétricas.  $q^d$  y  $\dot{q}^d$  son la trayectoria deseada y su velocidad. El siguiente teorema presenta un algoritmo estable de aprendizaje para el contralador (6.30).

**Teorema 6.4** *Sea la trayectoria deseada acotada, y suponga que la siguiente condición de seguimiento se satisface:*

$$K_d > \Lambda_g^{-1} + \Lambda_\sigma^{-1} + \tilde{x}_s^T K_p^T \Lambda_J^{-1} K_p \tilde{x}_s$$

si se usan las siguientes leyes de aprendizaje para actualizar los valores de los pesos de la red neuronal multicapa:

$$\begin{aligned} \dot{\widehat{W}}_t &= -K_w\sigma(V_t x)\dot{q}^T \\ \dot{\widehat{V}}_t &= -K_v x W^* D_\sigma \dot{q}^T \end{aligned} \quad (6.31)$$

y si se emplea el siguiente control de zona muerta, entonces el error de seguimiento  $\tilde{x}_s$  es acotado.

- si  $\|\dot{q}\|^2 > \lambda_{\min}^{-1}(Q) (\bar{\eta}_J + \bar{\eta}_\sigma + \bar{\eta}_g)$  entonces, se usa la ley de control visual (6.30)
- si  $\|\dot{q}\|^2 \leq \lambda_{\min}^{-1}(Q) (\bar{\eta}_J + \bar{\eta}_\sigma + \bar{\eta}_g)$  entonces, se deja de usar la ley de control visual (6.30)

donde  $Q = K_d - [\Lambda_g^{-1} + \Lambda_\sigma^{-1} + \tilde{x}_s^T K_p^T \Lambda_J^{-1} K_p \tilde{x}_s]$ .

**Demostracin.** Se propone la siguiente función candidata de Lyapunov:

$$V = \frac{1}{2}\dot{q}^T M\dot{q} + \frac{1}{2\alpha h} R^{-1} \tilde{x}_s^T K_p \tilde{x}_s + \frac{1}{2} \text{tr} \left( \widehat{W}_t^T K_w^{-1} \widehat{W}_t \right) + \frac{1}{2} \text{tr} \left( \widehat{V}_t^T K_v^{-1} \widehat{V}_t \right) \quad (6.32)$$

donde  $K_w$  y  $K_v$  son matrices positivas definidas. Derivando (6.32) a lo largo de las trayectorias (6.6), obtenemos:

$$\dot{V} = \dot{q}^T M \ddot{q} + \frac{1}{2} \dot{q}^T \dot{M} \dot{q} + \frac{1}{\alpha h} R^{-1} \dot{\tilde{x}}_s^T K_p \tilde{x}_s + tr \left( \tilde{W}_t^T K_w^{-1} \dot{\tilde{W}}_t \right) + tr \left( \tilde{V}_t^T K_v^{-1} \dot{\tilde{V}}_t \right)$$

Sustituyendo  $\ddot{q}$  de (6.8),

$$\begin{aligned} \dot{V} = \frac{1}{2} \dot{q}^T & \left\{ 2\tau - 2C(q, \dot{q}) \dot{q} - 2 \left[ G(q) + F_d(\dot{q}) \right] + 2\dot{M} \dot{q} \right\} \\ & + \frac{1}{\alpha h} R^{-1} \dot{\tilde{x}}_s^T K_p \tilde{x}_s + tr \left( \tilde{W}_t^T K_w^{-1} \dot{\tilde{W}}_t \right) + tr \left( \tilde{V}_t^T K_v^{-1} \dot{\tilde{V}}_t \right) \end{aligned}$$

Utilizando la *Propiedad 6.2*, con (6.25) y (6.5)

$$\begin{aligned} \dot{V} = \dot{q}^T & \left[ \tau - (W^* \sigma(V^* x) + \eta_g) - JK_p \tilde{x}_s \right] \\ & + tr \left( \tilde{W}_t^T K_w^{-1} \dot{\tilde{W}}_t \right) + tr \left( \tilde{V}_t^T K_v^{-1} \dot{\tilde{V}}_t \right) \end{aligned} \quad (6.33)$$

Mediante (6.27), (6.30) y  $\hat{J} = -\eta_J + J$

$$\begin{aligned} \dot{V} = -\dot{q}^T & \left[ \eta_J K_p \tilde{x}_s + K_d \dot{q} + \tilde{W}_t \sigma(V_t x) + W^* D_\sigma \tilde{V}_t x + \nu_\sigma + \eta_g \right] \\ & + tr \left( \tilde{W}_t^T K_w^{-1} \dot{\tilde{W}}_t \right) + tr \left( \tilde{V}_t^T K_v^{-1} \dot{\tilde{V}}_t \right) \end{aligned} \quad (6.34)$$

Por (6.31)

$$\begin{aligned} \dot{V} = -\dot{q}^T & K_d \dot{q} - \dot{q}^T \eta_J K_p \tilde{x}_s - \dot{q}^T \left[ \nu_\sigma + \eta_g \right] \\ & + tr \left[ \left( K_w^{-1} \dot{\tilde{W}}_t - \sigma(V_t x) \dot{q}^T \right) \tilde{W} \right] + tr \left[ \left( K_v^{-1} \dot{\tilde{V}}_t - x W^* D_\sigma \dot{q}^T \right) \tilde{V} \right] \end{aligned} \quad (6.35)$$

Se usará la siguiente desigualdad matricial:

$$X^T Y + (X^T Y)^T \leq X^T \Lambda^{-1} X + Y^T \Lambda Y \quad (6.36)$$

para los términos:  $-\dot{q}^T [\nu_\sigma + \eta_g]$  y  $-\dot{q}^T \eta_J K_p \tilde{x}_s$  en (6.35) se obtienen las siguientes expresiones:

$$\begin{aligned} -\dot{q}^T [\nu_\sigma + \eta_g] &\leq \dot{q}^T (\Lambda_g^{-1} + \Lambda_\sigma^{-1}) \dot{q} + \bar{\eta}_g + \bar{\eta}_\sigma \\ -\dot{q}^T \eta_J K_p \tilde{x}_s &\leq \dot{q}^T \tilde{x}_s^T K_p^T \Lambda_J^{-1} K_p \tilde{x}_s \dot{q} + \bar{\eta}_J \end{aligned} \quad (6.37)$$

Así,

$$\dot{V} \leq -\dot{q}^T Q \dot{q} + \bar{\eta}_J + \bar{\eta}_\sigma + \bar{\eta}_g \quad (6.38)$$

donde  $Q = K_d - [\Lambda_g^{-1} + \Lambda_\sigma^{-1} + \tilde{x}_s^T K_p^T \Lambda_J^{-1} K_p \tilde{x}_s]$ .

*Caso 1:* Si  $\|\dot{q}\|^2 > \lambda_{\min}^{-1}(Q) (\bar{\eta}_J + \bar{\eta}_\sigma + \bar{\eta}_g)$  entonces, se usa la ley de control visual (6.30) y se concluye que:

$$\dot{V}_t \leq -\dot{q}^T Q \dot{q} + (\bar{\eta}_J + \bar{\eta}_\sigma + \bar{\eta}_g) \leq -\lambda_{\min}(Q) \|\dot{q}\|^2 + (\bar{\eta}_J + \bar{\eta}_\sigma + \bar{\eta}_g) < 0 \quad (6.39)$$

Por lo tanto,  $V_t$  es acotada.

*Caso 2:* Si  $\|\dot{q}\|^2 \leq \lambda_{\min}^{-1}(Q) (\bar{\eta}_J + \bar{\eta}_\sigma + \bar{\eta}_g)$ , el término  $\frac{1}{2} \dot{q}^T M \dot{q}$  es acotado. Dado que la acción de control se deja de aplicar, el término  $\frac{1}{2\alpha h} R^{-1} \tilde{x}_s^T K_p \tilde{x}_s$  se mantiene constante y  $V_t$  también se mantiene constante.

Entonces  $V_t$  es acotado,  $\dot{q}$ ,  $\tilde{x}_s$ ,  $\tilde{W}_t$  y  $\tilde{V}_t$  son acotados. ■

**Comentario 6.3** *El error de seguimiento converge a una bola de radio que se encuentra acotada por arriba por el valor  $(\bar{\eta}_g + \bar{\eta}_\sigma)$ , y se ve afectada por la matriz conocida a priori  $W^*$ . El teorema anterior muestra que  $W^*$  no tiene influencia en la estabilidad, por ello se puede seleccionar cualquier valor para  $W^*$  al inicio. El teorema anterior muestra que las reglas de aprendizaje (6.31) pueden hacer al error de identificación convergente. La matriz  $W^*$  puede ser elegida mediante los siguientes pasos:*

1. *Se comienza con cualquier valor inicial para la matriz  $W_0$ .*
2. *Se realiza una identificación en línea con la matriz  $W_t$ .*
3. *Sea la matriz  $W_t$  elegida ahora como la condición inicial, i.e.,  $W^* = W_t$ .*

4. *Si el error de identificación decrece, se repite el proceso de indentificación regresando al paso 2. De otra forma, se detiene la identificación y el valor de  $W_t$  será el valor final para  $W^*$ .*

**Comentario 6.4** *De la definición de la función de Lyapunov (6.32), esta estructura es diferente a las redes neuronales normalmente usadas para aproximar una función no-lineal. Los términos  $-K_w \sigma(V_t x) \dot{q}^T$  y  $-K_v x W^* D_\sigma \dot{q}^T$  son similares a los usados en el esquema de propagación hacia atrás, sólo que en el esquema de propagación hacia atrás se usa  $W_t$  en vez de  $W^*$ .*

### 6.2.5. Controlador visual Proporcional-Derivativo con observador y redes neuronales multicapa

En esta sección, se presenta un nuevo controlador visual tipo PD cuya estructura se muestra en la Figura 6.2 para un robot de dos grados de libertad (6.6) con compensación directa (gravedad y fricción) usando redes neuronales multicapa y además la velocidad  $\dot{q}$  no es medible, por lo que se aplica el observador de modos deslizantes (6.14) para obtener  $\hat{q}$ . Adicionalmente, se considera dos tipos de incertidumbres. Primero, que no se conoce exactamente la matriz Jacobiana del robot y segundo, que las coordenadas del espacio y las imagen no coinciden exactamente.

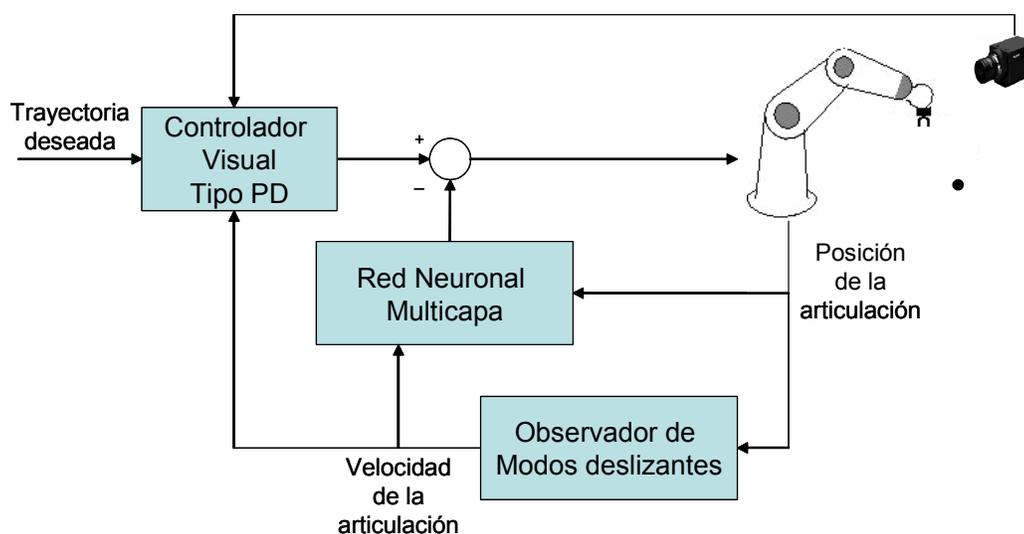


Figura 6.2: Estructura del controlador visual tipo PD con compensación y observador.

#### Preliminares

**Red neuronal multicapa** Dado que para este caso el valor de la velocidad  $\dot{q}$  de las articulaciones del robot es desconcida y se usa un observador de modos deslizantes para obtener el valor estimado  $\hat{q}$ . Se requiere definir una nueva red neural multicapa.

**Definición 6.2** *La red neuronal diferencial puede expresarse de la siguiente forma:*

$$y = W\sigma(Vs) \quad (6.40)$$

donde  $s = [s_1, s_2, \dots, s_n]^T$ ,  $y = [y_1, y_2, \dots, y_m]^T$ . Las matrices de pesos se definen como:  $W = [w_{ij}] \in R^{m \times k}$ ,  $V = [v_{jk}] \in R^{k \times n}$ ,  $\sigma(z) = [\sigma(z_1), \sigma(z_2), \dots, \sigma(z_k)]$ .  $k$  es el número de nodos de la capa oculta. La función de activación  $\sigma(z)$  es no-lineal acotada.

La única diferencia con la red neuronal multicapa de la sección anterior es que el nuevo estado  $s$  se define como:

$$s = \begin{bmatrix} q^T, \hat{q}^T \end{bmatrix}^T$$

de tal forma, que cumple con las mismas cotas para la aproximación de los términos de fricción y gravedad, así como para la función de activación presentadas en el caso anterior.

### Análisis de estabilidad

La ley de control visual con compensación mediante redes neuronales diferenciales y la velocidad  $\hat{q}$  obtenida mediante el observador de modos deslizantes es:

$$\tau = \hat{J}K_p \tilde{x}_s - K_d \dot{\hat{q}} + \hat{W}_t \sigma(\hat{V}_t s) \quad (6.41)$$

donde  $K_p$  y  $K_d$  son matrices positivas definidas y simétricas.  $q^d$  y  $\dot{q}^d$  son la trayectoria y velocidad deseada.  $\hat{W}_t$  y  $\hat{V}_t$  son las matrices de pesos variantes en el tiempo de la red neuronal multicapa.

Del *Teorema 6.2* se obtiene que  $\dot{\hat{q}} = \dot{q}$  cuando  $t \rightarrow \infty$ , ahora para cualquier valor de tiempo los errores de observación están definidos por:

$$\begin{aligned} \dot{\hat{q}} &= \dot{q} + \varepsilon \\ \ddot{\hat{q}} &= \ddot{q} + \varepsilon_1 \end{aligned} \quad (6.42)$$

donde  $\varepsilon$  está acotado por (6.18).

**Cota 6.5**  $\varepsilon_1$  está acotado de la siguiente forma:

$$\varepsilon_1^T \Lambda_{\varepsilon_1} \varepsilon_1 \leq \bar{\eta}_{\varepsilon_1} \quad (6.43)$$

donde  $\bar{\eta}_{\varepsilon_1}$  es una constante positiva.

Es importante aclarar que el valor  $\ddot{\hat{q}}$  no requiere estimarse pues no forma parte de las reglas de aprendizaje (6.45) de la red neuronal multicapa, por lo que sólo es necesario indicar que existe un error de estimación de  $\ddot{q}$  para propósitos del análisis de estabilidad planteado en el siguiente teorema.

**Teorema 6.5** Sea la trayectoria deseada acotada, y suponga que la siguiente condición de seguimiento se satisface:

$$K_d > \Lambda_g^{-1} + \Lambda_\sigma^{-1} + \tilde{x}_s^T K_p^T \Lambda_J^{-1} K_p \tilde{x}_s + M^T \Lambda_{\varepsilon_1}^{-1} M + \dot{M}^T \Lambda_\varepsilon^{-1} \dot{M} \quad (6.44)$$

donde  $0 < \Lambda_1 = \Lambda_1^T \in \mathfrak{R}^{n \times n}$  es una matriz conocida. Si se usan las siguientes leyes de aprendizaje para actualizar los valores de los pesos de la red neuronal multicapa:

$$\begin{aligned} \dot{\hat{W}}_t &= -K_w \sigma(V_t s) \hat{q}^T \\ \dot{\hat{V}}_t &= -K_v s W^* D_\sigma^T \hat{q} \end{aligned} \quad (6.45)$$

y si se utiliza el siguiente control de zona muerta, entonces el error de seguimiento  $\tilde{x}_s$  es acotado.

- si  $\left\| \dot{\hat{q}} \right\|^2 > \lambda_{\min}^{-1}(Q) (\bar{\eta}_J + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_\varepsilon + \bar{\eta}_{\varepsilon_1})$  entonces, se usa la ley de control visual (6.41)
- si  $\left\| \dot{\hat{q}} \right\|^2 \leq \lambda_{\min}^{-1}(Q) (\bar{\eta}_J + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_\varepsilon + \bar{\eta}_{\varepsilon_1})$  entonces, se deja de usar la ley de control visual (6.41)

$$\text{donde } Q = K_d - \left[ \Lambda_g^{-1} + \Lambda_\sigma^{-1} + \tilde{x}_s^T K_p^T \Lambda_J^{-1} K_p \tilde{x}_s + M^T \Lambda_{\varepsilon_1}^{-1} M + \dot{M}^T \Lambda_\varepsilon^{-1} \dot{M} \right].$$

**Demostacin.** Se propone la siguiente funci3n candidata de Lyapunov:

$$V = \frac{1}{2} \dot{\hat{q}}^T M \dot{\hat{q}} + \frac{1}{2\alpha h} R^{-1} \tilde{x}_s^T K_p \tilde{x}_s + \frac{1}{2} \text{tr} \left( \tilde{W}_t^T K_w^{-1} \tilde{W}_t \right) + \frac{1}{2} \text{tr} \left( \tilde{V}_t^T K_v^{-1} \tilde{V}_t \right) \quad (6.46)$$

donde  $K_w$  y  $K_v$  son matrices positivas definidas. Derivando (6.46) a lo largo de las trayectorias (6.6), obtenemos:

$$\dot{V} = \dot{\hat{q}}^T M \ddot{\hat{q}} + \frac{1}{2} \dot{\hat{q}}^T \dot{M} \dot{\hat{q}} + \frac{1}{\alpha h} R^{-1} \dot{\tilde{x}}_s^T K_p \tilde{x}_s + \text{tr} \left( \dot{\tilde{W}}_t^T K_w^{-1} \tilde{W}_t \right) + \text{tr} \left( \dot{\tilde{V}}_t^T K_v^{-1} \tilde{V}_t \right) \quad (6.47)$$

Ahora sustituyendo  $\ddot{\hat{q}}$  de (6.42) y despejando  $\ddot{q}$  de (6.8),

$$2M\ddot{\hat{q}} = M \left( \ddot{q} + \varepsilon_1 \right) = 2\tau - 2C \left( q, \dot{q} \right) \dot{q} - 2 \left[ G(q) + F_d \left( \dot{q} \right) \right] + 2M\varepsilon_1 \quad (6.48)$$

usando (6.48) y  $\dot{\hat{q}} = \dot{q} + \varepsilon$  en (6.47) se tiene:

$$\begin{aligned} \dot{V} = \frac{1}{2} \dot{\hat{q}}^T \left\{ 2\tau - 2C \left( q, \dot{q} \right) \dot{q} - 2 \left[ G(q) + F_d \left( \dot{q} \right) \right] + 2M\varepsilon_1 + 2\dot{M} \left( \dot{q} + \varepsilon \right) \right\} \\ + \frac{1}{\alpha h} R^{-1} \dot{\tilde{x}}_s^T K_p \tilde{x}_s + \text{tr} \left( \dot{\tilde{W}}_t^T K_w^{-1} \tilde{W}_t \right) + \text{tr} \left( \dot{\tilde{V}}_t^T K_v^{-1} \tilde{V}_t \right) \end{aligned}$$

Por la *Propiedad 6.2*,

$$\begin{aligned} \dot{V} = \dot{\hat{q}}^T \left\{ \tau - \left[ G(q) + F_d \left( \dot{q} \right) \right] + M\varepsilon_1 + \dot{M} \varepsilon \right\} \\ + \frac{1}{\alpha h} R^{-1} \dot{\tilde{x}}_s^T K_p \tilde{x}_s + \text{tr} \left( \dot{\tilde{W}}_t^T K_w^{-1} \tilde{W}_t \right) + \text{tr} \left( \dot{\tilde{V}}_t^T K_v^{-1} \tilde{V}_t \right) \end{aligned}$$

Aplicando la estimaci3n de la gravedad y fricci3n (6.27),

$$\begin{aligned} \dot{V} = \dot{\hat{q}}^T \left[ \tau - \left( W^* \sigma(V^* s) + \eta_g \right) - JK_p \tilde{x}_s + M\varepsilon_1 + \dot{M} \varepsilon \right] \\ + \text{tr} \left( \dot{\tilde{W}}_t^T K_w^{-1} \tilde{W}_t \right) + \text{tr} \left( \dot{\tilde{V}}_t^T K_v^{-1} \tilde{V}_t \right) \end{aligned}$$

Empleando (6.41) y (6.5),

$$\begin{aligned} \dot{V} = & -\hat{q}^T \left[ \eta_J K_p \tilde{x}_s + J K_p \tilde{x}_s - J K_p \tilde{x}_s + K_d \dot{\hat{q}} \right] \\ & + \tilde{W}_t \sigma(V_t s) + W^* D_\sigma \tilde{V}_t s + v_\sigma + \eta_g + M \varepsilon_1 + \dot{M} \varepsilon + \text{tr} \left( \tilde{W}_t^T K_w^{-1} \dot{\tilde{W}}_t \right) + \text{tr} \left( \tilde{V}_t^T K_v^{-1} \dot{\tilde{V}}_t \right) \end{aligned}$$

y

$$\begin{aligned} \dot{V} = & -\hat{q}^T \left[ \eta_J K_p \tilde{x}_s + K_d \dot{\hat{q}} + \tilde{W}_t \sigma(V_t s) + W^* D_\sigma \tilde{V}_t s \right] \\ & + \eta_\sigma + v_g + M \varepsilon_1 + \dot{M} \varepsilon + \text{tr} \left( \tilde{W}_t^T K_w^{-1} \dot{\tilde{W}}_t \right) + \text{tr} \left( \tilde{V}_t^T K_v^{-1} \dot{\tilde{V}}_t \right) \end{aligned}$$

con (6.45) tenemos:

$$\begin{aligned} \dot{V} = & -\hat{q}^T K_d \dot{\hat{q}} - \hat{q}^T \eta_J K_p \tilde{x}_s - \hat{q}^T [v_\sigma + \eta_g] - \hat{q}^T M \varepsilon_1 - \hat{q}^T \dot{M} \varepsilon \\ & + \text{tr} \left[ \left( K_w^{-1} \dot{\tilde{W}}_t - \sigma(V_t s) \hat{q}^T \right) \tilde{W} \right] + \text{tr} \left[ \left( K_v^{-1} \dot{\tilde{V}}_t - s W^* D_\sigma^T \hat{q}^T \right) \tilde{V} \right] \end{aligned} \quad (6.49)$$

Los términos  $\text{tr} \left[ \left( K_w^{-1} \dot{\tilde{W}}_t - \sigma(V_t s) \hat{q}^T \right) \tilde{W} \right]$ ,  $\text{tr} \left[ \left( K_v^{-1} \dot{\tilde{V}}_t - s W^* D_\sigma^T \hat{q}^T \right) \tilde{V} \right]$  se hacen cero, pues la red neuronal multicapa con su regla de aprendizaje es capaz de estimar adecuadamente la gravedad y la fricción.

Se utilizará la siguiente desigualdad matricial:

$$X^T Y + (X^T Y)^T \leq X^T \Lambda^{-1} X + Y^T \Lambda Y \quad (6.50)$$

la cual es válida para cualquier  $X, Y \in \mathfrak{R}^{n \times k}$  y para cualquier matriz definida positiva a  $0 < \Lambda = \Lambda^T \in \mathfrak{R}^{n \times n}$ .

Ahora los términos  $\hat{q}^T M \varepsilon_1$ ,  $-\hat{q}^T \dot{M} \varepsilon$ ,  $-\hat{q}^T [v_\sigma + \eta_g]$  y  $-\hat{q}^T \eta_J K_p \tilde{x}_s$  de (6.49) se les aplica la desigualdad (6.50) y se reescriben como:

$$\begin{aligned} \hat{q}^T M \varepsilon_1 & \leq \hat{q}^T M^T \Lambda_{\varepsilon_1}^{-1} M \hat{q} + \bar{\eta}_{\varepsilon_1} \\ \hat{q}^T \dot{M} \varepsilon & \leq \hat{q}^T M^T \Lambda_{\varepsilon}^{-1} \dot{M} \hat{q} + \bar{\eta}_{\varepsilon} \\ -\hat{q}^T [v_\sigma + \eta_g] & \leq \hat{q}^T (\Lambda_g^{-1} + \Lambda_\sigma^{-1}) \dot{\hat{q}} + \bar{\eta}_g + \bar{\eta}_\sigma \\ -\hat{q}^T \eta_J K_p \tilde{x}_s & \leq \hat{q}^T \tilde{x}_s^T K_p^T \Lambda_J^{-1} K_p \tilde{x}_s \dot{\hat{q}} + \bar{\eta}_J \end{aligned} \quad (6.51)$$

Luego se sustituye (6.51) y se obtiene:

$$\begin{aligned} \dot{V} &\leq -\hat{q}^T \left[ K_d - \left( \Lambda_g^{-1} + \Lambda_\sigma^{-1} + \tilde{x}_s^T K_p^T \Lambda_J^{-1} K_p \tilde{x}_s + M^T \Lambda_{\varepsilon_1}^{-1} M + \dot{M}^T \Lambda_\varepsilon^{-1} \dot{M} \right) \right] \dot{\hat{q}} \\ &\quad + \bar{\eta}_J + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_\varepsilon + \bar{\eta}_{\varepsilon_1} \\ \dot{V}_t &\leq -\hat{q}^T Q \dot{\hat{q}} + \bar{\eta}_J + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_\varepsilon + \bar{\eta}_{\varepsilon_1} \end{aligned}$$

con  $Q = K_d - \left( \Lambda_g^{-1} + \Lambda_\sigma^{-1} + \tilde{x}_s^T K_p^T \Lambda_J^{-1} K_p \tilde{x}_s + M^T \Lambda_{\varepsilon_1}^{-1} M + \dot{M}^T \Lambda_\varepsilon^{-1} \dot{M} \right) > 0$ .

*Caso 1:* Si  $\left\| \dot{\hat{q}} \right\|^2 > \lambda_{\min}^{-1}(Q) (\bar{\eta}_J + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_\varepsilon + \bar{\eta}_{\varepsilon_1})$  entonces, se usa la ley de control visual (6.41), se concluye que:

$$\begin{aligned} \dot{V}_t &\leq -\hat{q}^T Q \dot{\hat{q}} + (\bar{\eta}_J + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_\varepsilon + \bar{\eta}_{\varepsilon_1}) \\ \dot{V}_t &\leq -\lambda_{\min}(Q) \left\| \dot{\hat{q}} \right\|^2 + (\bar{\eta}_J + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_\varepsilon + \bar{\eta}_{\varepsilon_1}) < 0 \end{aligned} \tag{6.52}$$

por lo tanto,  $V_t$  es acotada.

*Caso 2:* Si  $\left\| \dot{\hat{q}} \right\|^2 \leq \lambda_{\min}^{-1}(Q) (\bar{\eta}_J + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_\varepsilon + \bar{\eta}_{\varepsilon_1})$ , el término  $\frac{1}{2} \hat{q}^T M \dot{\hat{q}}$  es acotado. Dado que la acción de control se deja de aplicar, los términos  $\frac{1}{2\alpha h} R^{-1} \tilde{x}_s^T K_p \tilde{x}_s$  y  $V_t$  se mantienen constantes.

Entonces  $V_t$  es acotado,  $\hat{q}$ ,  $\tilde{x}_s$ ,  $\tilde{W}_t$  y  $\tilde{V}_t$  son acotados. ■

### 6.3. Robot de tres grados de libertad

En esta sección se presenta el caso para un robot de tres grados de libertad, por lo que es necesario obtener la posición en el espacio tridimensional de su órgano terminal. Se desea alcanzar dicha posición mediante un sistema de visión por computadora con el modelo óptico de estereo visión presentado en capítulo 4, cuya expresión es:

$$\begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix} = \begin{bmatrix} c' \cos(\alpha') \\ c' \operatorname{sen}(\alpha') \\ k_f u_L \end{bmatrix} \quad (6.53)$$

Con esta información visual a continuación se desarrollan los cuatro casos presentados en la sección anterior.

#### 6.3.1. Preliminares

##### Cinemática

A partir de la dinámica directa del robot de un robot planar de tres grados de libertad se obtiene la posición  $[x, y, z]^T \in \mathfrak{R}^3$  de su órgano terminal. Estas coordenadas son obtenidas con respecto el eje coordenado del robot en términos de la posición de sus articulaciones  $q(t) \in \mathfrak{R}^n$ , por lo que:

$$[x, y, z]^T = f(q)$$

donde  $f : \mathfrak{R}^n \longrightarrow \mathfrak{R}^3$ . La matriz Jacobiana  $J(\cdot) \in \mathfrak{R}^{3 \times n}$  del robot está definida como:

$$J(q) = \frac{\partial f(q)}{\partial q} \quad (6.54)$$

De la misma forma, para la posición deseada  $[x^*, y^*, z^*]^T \in \mathfrak{R}^3$ . Las coordenadas son obtenidas con respecto el eje coordenado del robot en términos de la posición de sus articulaciones  $q^*(t) \in \mathfrak{R}^n$ , por lo que:

$$[x^*, y^*, z^*]^T = f(q^*)$$

donde  $f : \mathfrak{R}^n \rightarrow \mathfrak{R}^3$ . La matriz Jacobiana  $J(\cdot) \in \mathfrak{R}^{3 \times n}$  del robot está definida como:

$$J(q^*) = \frac{\partial f(q^*)}{\partial q} \quad (6.55)$$

Así  $\dot{X} = J\dot{Q}$ , donde  $J$  es el Jacobiano del robot. Para realizar el análisis de estabilidad se usan las *suposiciones 6.1 y 6.2*.

### Modelo del sistema de estéreo visión

Dado que la relacion entre las coordenadas del espacio de trabajo y las coordenadas de pantalla de las imagenes capturadas por el sistema de estéreo visión (dos camaras) no es una expresion analítica directa. Se tiene la siguiente forma:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{pmatrix} f_1(x_L, y_L, x_R, y_R) \\ f_2(x_L, y_L, x_R, y_R) \\ f_3(x_L, y_L, x_R, y_R) \end{pmatrix}, \quad \begin{bmatrix} x^* \\ y^* \\ z^* \end{bmatrix} = \begin{pmatrix} f_1(x_L^*, y_L^*, x_R^*, y_R^*) \\ f_2(x_L^*, y_L^*, x_R^*, y_R^*) \\ f_3(x_L^*, y_L^*, x_R^*, y_R^*) \end{pmatrix}$$

donde  $f_1$ ,  $f_2$  y  $f_3$  son funciones suaves. En el caso de cuatro variables independientes, una función suave  $f$  se puede expresar con la fórmula de Taylor como:

$$f(x_1, x_2, x_3, x_4) = \sum_{k=0}^{l-1} \frac{1}{k!} \left[ (x_1 - x_1^0) \frac{\partial}{\partial x_1} + (x_2 - x_2^0) \frac{\partial}{\partial x_2} + (x_3 - x_3^0) \frac{\partial}{\partial x_3} + (x_4 - x_4^0) \frac{\partial}{\partial x_4} \right]^k f + R_L$$

Usando la aproximación de primer orden, se tiene:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} - \begin{bmatrix} x^* \\ y^* \\ z^* \end{bmatrix} = \begin{pmatrix} \frac{\partial}{\partial x_L} f_1 & \frac{\partial}{\partial y_L} f_1 & \frac{\partial}{\partial x_R} f_1 & \frac{\partial}{\partial y_R} f_1 \\ \frac{\partial}{\partial x_L} f_2 & \frac{\partial}{\partial y_L} f_2 & \frac{\partial}{\partial x_R} f_2 & \frac{\partial}{\partial y_R} f_2 \\ \frac{\partial}{\partial x_L} f_3 & \frac{\partial}{\partial y_L} f_3 & \frac{\partial}{\partial x_R} f_3 & \frac{\partial}{\partial y_R} f_3 \end{pmatrix} \begin{bmatrix} x_L - x_L^* \\ y_L - y_L^* \\ x_R - x_R^* \\ y_R - y_R^* \end{bmatrix} + R_L(x, y, z)$$

El error de posición medido visualmente  $\tilde{x}_s$  es:

$$\tilde{x}_s = \begin{bmatrix} x_L^* \\ y_L^* \\ x_R^* \\ y_R^* \end{bmatrix} - \begin{bmatrix} x_L \\ y_L \\ x_R \\ y_R \end{bmatrix} \quad (6.56)$$

y se puede reescribir como:

$$\tilde{x}_s = K(\psi) \left( \begin{bmatrix} x^* \\ y^* \\ z^* \end{bmatrix} - \begin{bmatrix} x \\ y \\ z \end{bmatrix} - R_L(x, y, z) \right)$$

donde  $\psi = [x_L, y_L, x_R, y_R]^T$  y,

$$K(\psi) = \left( \begin{array}{cccc} \frac{\partial}{\partial x_L} f_1 & \frac{\partial}{\partial y_L} f_1 & \frac{\partial}{\partial x_R} f_1 & \frac{\partial}{\partial y_R} f_1 \\ \frac{\partial}{\partial x_L} f_2 & \frac{\partial}{\partial y_L} f_2 & \frac{\partial}{\partial x_R} f_2 & \frac{\partial}{\partial y_R} f_2 \\ \frac{\partial}{\partial x_L} f_3 & \frac{\partial}{\partial y_L} f_3 & \frac{\partial}{\partial x_R} f_3 & \frac{\partial}{\partial y_R} f_3 \end{array} \right)^+$$

aquí el símbolo  $(\cdot)^+$  representa la matriz pseudoinversa en el sentido de Moor-Penrose [1] y se calcula como:

$$x^+ = \frac{x^T}{\|x\|^2}, \quad 0^+ = 0$$

De (6.54) y (6.55)

$$\tilde{x}_s = K(\psi) [f(q^*) - f(q) - R_L(f(q))] \quad (6.57)$$

y el error en velocidad es:

$$\dot{\tilde{x}}_s^T = K(\psi) \left[ J(q^*)\dot{q}^* - J(q)\dot{q} - \dot{R}_L J(q)\dot{q} \right] \quad (6.58)$$

para el caso del problema de regulación el término  $J(q^*) = 0$ , así:

$$\dot{\tilde{x}}_s^T = -K(\psi) J(q)\dot{q} - \dot{R}_L K(\psi) J(q)\dot{q} \quad (6.59)$$

**Cota 6.6** Sea el término  $\dot{R}_L K(\psi) J$  (debido a la aproximación de primer orden) acotado de la siguiente forma:

$$\left( \dot{R}_L K(\psi) J \right)^T \Lambda_d \left( \dot{R}_L K(\psi) J \right) \leq \bar{\eta}_d$$

donde  $\Lambda_d$  es una matriz definida y  $\bar{\eta}_d$  es una constante definida.

### Dinámica

Se usará la misma dinámica para el robot manipulador y las propiedades presentadas para el caso del robot de dos dimensiones.

### Planteamiento del problema de control

**Suposicin 6.6** La posición deseada o objetivo  $[x^*, y^*, z^*]^T \in \mathfrak{R}^3$  es conocida y estática (caso de regulación), por lo que:

$$J(q^*) = \frac{\partial f(q^*)}{\partial q} = 0$$

**Suposicin 6.7** La posición deseada se encuentra dentro del espacio de trabajo del robot y de ambas imágenes del sistema de visión por computadora, del tal forma que existe al menos un vector de posición  $\mathbf{x} = [x, y, z]^T$ , con una cierta configuración de articulaciones  $q^* \in \mathfrak{R}^3$  tal que el órgano terminal puede alcanzarla.

**Comentario 6.5** Si la anterior suposición no se cumple no es posible llevar al órgano terminal del robot a la posición deseada.

**Problema 6.2** El problema de regulación en el espacio visual consiste en obtener una ley de control visual tal que para cualquier condición inicial  $x_s(t_o)$  en el tiempo  $T(\eta_C, x_s(t_o))$  se tenga:

$$\|x_s(t) - x_s^*\| \leq \eta_C \quad \forall t_o + T(\eta_C, x_s(t_o))$$

donde  $\eta_C$  es una constante positiva y  $x_s^*$  es la posición deseada.

### 6.3.2. Control visual estándar

#### Análisis de estabilidad

La ley de control visual quedaría como:

$$\tau = K(\psi) JK_p \tilde{x}_s - K_d \dot{q} + (G + F_d) \quad (6.60)$$

donde  $K_p$  y  $K_d$  son matrices positivas definidas y simétricas.

**Teorema 6.6** *Sea la trayectoria deseada acotada, y suponga que la siguiente condición de seguimiento se satisface:*

$$K_d > \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s$$

donde  $0 < \Lambda_d = \Lambda_d^T \in \mathfrak{R}^{n \times n}$  es una matriz conocida y si se usa el siguiente control de zona muerta, entonces el error de seguimiento  $\tilde{x}_s$  es acotado.

- si  $\|\dot{q}\|^2 > \lambda_{\min}^{-1}(Q)(\bar{\eta}_d)$  entonces, se usa la ley de control visual (6.60)
- si  $\|\dot{q}\|^2 \leq \lambda_{\min}^{-1}(Q)(\bar{\eta}_d)$  entonces, se deja de usar la ley de control visual (6.30)

donde  $Q = K_d - \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s$

**Demostracin.** Se propone la siguiente función candidata de Lyapunov:

$$V = \frac{1}{2} \dot{q}^T M \dot{q} + \frac{1}{2} \tilde{x}_s^T K_p \tilde{x}_s$$

donde  $\frac{1}{2} \dot{q}^T M \dot{q}$  es la energía cinética del robot. La derivada en tiempo de esta función de Lyapunov es:

$$\dot{V} = \dot{q}^T M \ddot{q} + \frac{1}{2} \dot{q}^T \dot{M} \dot{q} + \tilde{x}_s^T K_p \dot{\tilde{x}}_s$$

Sustituyendo  $\ddot{q}$ ,

$$\dot{V} = \frac{1}{2} \dot{q}^T \left\{ 2\tau - 2C(q, \dot{q}) \dot{q} - 2 \left[ G(q) + F_d(\dot{q}) \right] + M \dot{q} \right\} + \tilde{x}_s^T K_p \dot{\tilde{x}}_s$$

Usando la *Propiedad 6.2*:

$$\dot{q}^T \left( \dot{M} - 2C \right) \dot{q} = 0, \quad \forall q, \dot{q}$$

Se obtiene,

$$\dot{V} = \dot{q}^T \{ \tau - [G + F_d] \} + \tilde{x}_s^T K_p \tilde{x}_s$$

Sustituyendo (6.59), y para el caso de regulación, donde  $J(q^*) = \frac{\partial K(q^*)}{\partial q} = 0$ , se tiene que:

$$\dot{V} = \dot{q}^T \left[ \tau - (G + F_d) - K(\psi) JK_p \tilde{x}_s - \dot{R}_L K(\psi) JK_p \tilde{x}_s \right]$$

Usando la desigualdad matricial,

$$X^T Y + (X^T Y)^T \leq X^T \Lambda^{-1} X + Y^T \Lambda Y$$

donde  $X, Y \in \mathfrak{R}^{n \times k}$  y  $0 < \Lambda = \Lambda^T \in \mathfrak{R}^{n \times n}$ , el término  $\dot{q}^T \dot{R}_L K(\psi) JK_p \tilde{x}$  puede ser reescrito como:

$$\begin{aligned} \dot{q}^T \dot{R}_L K(\psi) JK_p \tilde{x} &\leq \dot{q}^T \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s \dot{q} + \left( \dot{R}_L K(\psi) J \right)^T \Lambda_d \left( \dot{R}_L K(\psi) J \right) \\ &\leq \dot{q}^T \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s \dot{q} + \bar{\eta}_d \end{aligned}$$

Aplicando la ley de control (6.60):

$$\begin{aligned} \dot{V} &\leq -\dot{q}^T \left( K_d - \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \right) \dot{q} + \bar{\eta}_d \\ &\leq -\dot{q}^T Q \dot{q} + \bar{\eta}_d \end{aligned}$$

donde  $Q = K_d - \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p > 0$ .

*Caso 1:* Si  $\left\| \dot{q} \right\|^2 > \lambda_{\min}^{-1}(Q) (\bar{\eta}_d)$  entonces, se usa la ley de control visual (6.60), se concluye que:

$$\dot{V}_t \leq -\dot{q}^T Q \dot{q} + \bar{\eta}_d \leq -\lambda_{\min}(Q) \left\| \dot{q} \right\|^2 + \bar{\eta}_d < 0$$

por lo tanto,  $V_t$  es acotada.

*Caso 2:* Si  $\left\| \dot{q} \right\|^2 \leq \lambda_{\min}^{-1}(Q) (\bar{\eta}_d)$ , el término  $\frac{1}{2} \dot{q}^T M \dot{q}$  es acotado. Dado que la acción de control se deja de aplicar, los términos  $\frac{1}{2} \tilde{x}_s^T K_p \tilde{x}_s$  y  $V_t$  se mantienen constantes.

Entonces  $V_t$  es acotado,  $\dot{q}$  y  $\tilde{x}_s$  son acotados. ■

### 6.3.3. Controlador visual Proporcional-Derivativo con observador

Se propone la siguiente ley de control visual:

$$\tau = K(\psi) JK_p \tilde{x}_s - K_d \dot{\hat{q}} + G(q) + F_d(q) \quad (6.61)$$

donde  $K_p$ ,  $K_d$  son matrices definidas positivas. El siguiente teorema presenta la prueba de estabilidad de la ley de control visual (6.61).

**Teorema 6.7** *Considere la ley de control visual (6.61) y  $K_d$  es selecciona lo suficientemente grande, tal que:*

$$K_d > \Lambda_\varepsilon^{-1} + \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s$$

y se usa el siguiente control de zona muerta, el error de posición  $\tilde{x}_s$  es acotado.

- si  $\|\dot{q}\|^2 > \lambda_{\min}^{-1}(Q)(\bar{\eta}_\varepsilon)$  entonces, se usa la ley de control visual (6.61)
- si  $\|\dot{q}\|^2 \leq \lambda_{\min}^{-1}(Q)(\bar{\eta}_\varepsilon)$  entonces, se deja de usar la ley de control visual (6.61)

donde  $Q = K_d - \Lambda_\varepsilon^{-1} + \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s > 0$ , entonces el error de seguimiento es acotado.

**Demostracin.** Se propone la siguiente función candidata de Lyapunov:

$$V = \frac{1}{2} \dot{q}^T M \dot{q} + \frac{1}{2\alpha h} R^{-1} \tilde{x}_s^T K_p \tilde{x}_s$$

Su derivada en tiempo es:

$$\begin{aligned} \dot{V}_t &= \dot{q}^T M \ddot{q} + \frac{1}{2} \dot{q}^T \dot{M} \dot{q} + \tilde{x}_s^T K_p \dot{\tilde{x}}_s \\ \dot{V}_t &= \frac{1}{2} \dot{q}^T \left( 2\tau - 2C(q, \dot{q}) \dot{q} - 2[G(q) + F_d(q)] + \dot{M} \dot{q} \right) + \tilde{x}_s^T K_p \dot{\tilde{x}}_s \end{aligned}$$

Usando la *Propiedad 6.2*,

$$\dot{V}_t = \dot{q}^T (\tau - [G(q) + F_d(q)]) + \tilde{x}_s^T K_p \dot{\tilde{x}}_s$$

Sustituyendo (6.59), se tiene:

$$\dot{V}_t = \dot{q}^T \left\{ \tau - [G(q) + F_d(q)] + K(\Psi) JK_p \tilde{x}_s + \dot{R}_L K(\psi) JK_p \tilde{x}_s \right\}$$

Aplicando la ley de control visual (6.61), se obtiene:

$$\begin{aligned} \dot{V}_t &= \dot{q}^T \left\{ \begin{array}{l} K(\psi) JK_p \tilde{x}_s - K_d \dot{\hat{q}} + [G(q) + F_d(q)] - [G(q) + F_d(q)] \\ -K(\psi) JK_p \tilde{x}_s - \dot{R}_L K(\psi) JK_p \tilde{x}_s \end{array} \right\} \\ \dot{V}_t &= \dot{q}^T \left\{ -K_d \dot{\hat{q}} - \dot{R}_L K(\psi) JK_p \tilde{x}_s \right\} \end{aligned}$$

tomando  $\dot{\hat{q}} = \dot{q} + \varepsilon$

$$\dot{V} = -\dot{q}^T K_d \dot{q} - \dot{q}^T K_d \varepsilon - \dot{R}_L K(\psi) JK_p \tilde{x}_s \quad (6.62)$$

Usando la desigualdad matricial:

$$X^T Y + (X^T Y)^T \leq X^T \Lambda^{-1} X + Y^T \Lambda Y \quad (6.63)$$

donde  $X, Y$  son dos matrices cualesquiera,  $\Lambda$  es definida positiva. El término  $\dot{q}^T K_d \varepsilon$  puede ser estimado como:

$$\begin{aligned} \dot{q}^T K_d \varepsilon &\leq \dot{q}^T \Lambda_\varepsilon^{-1} \dot{q} + \bar{\eta}_\varepsilon \\ \dot{q}^T \dot{R}_L K(\psi) JK_p \tilde{x}_s &\leq \dot{q}^T \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s \dot{q} + \left( \dot{R}_L K(\psi) J \right)^T \Lambda_d \left( \dot{R}_L K(\psi) J \right) \\ &\leq \dot{q}^T \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s \dot{q} + \bar{\eta}_d \end{aligned}$$

Entonces (6.62) queda como:

$$\begin{aligned} \dot{V}_t &\leq -\dot{q}^T \left( K_d - \Lambda_\varepsilon^{-1} + \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s \right) \dot{q} + \bar{\eta}_\varepsilon + \bar{\eta}_d \\ \dot{V}_t &\leq -\dot{q}^T Q \dot{q} + \bar{\eta}_\varepsilon + \bar{\eta}_d \end{aligned}$$

donde  $Q = K_d - \Lambda_\varepsilon^{-1} + \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s > 0$ .

*Caso 1:* Si  $\left\|\dot{q}\right\|^2 > \lambda_{\min}^{-1}(Q)(\bar{\eta}_\varepsilon + \bar{\eta}_d)$  entonces, se usa la ley de control visual (6.61), se concluye que:

$$\dot{V}_t \leq -\dot{q}^T Q \dot{q} + (\bar{\eta}_\varepsilon + \bar{\eta}_d) \leq -\lambda_{\min}(Q) \left\|\dot{q}\right\|^2 + (\bar{\eta}_\varepsilon + \bar{\eta}_d) < 0 \quad (6.64)$$

por lo tanto,  $V_t$  es acotada.

*Caso 2:* Si  $\left\|\dot{q}\right\|^2 \leq \lambda_{\min}^{-1}(Q)(\bar{\eta}_\varepsilon + \bar{\eta}_d)$ , el término  $\frac{1}{2}\dot{q}^T M \dot{q}$  es acotado. Dado que la acción de control se deja de aplicar, los términos  $\frac{1}{2\alpha h} R^{-1} \tilde{x}_s^T K_p \tilde{x}_s$  y  $V_t$  se mantienen constantes.

Entonces  $V_t$  es acotado,  $\dot{q}$  y  $\tilde{x}_s$  son acotados. ■

### 6.3.4. Controlador visual Proporcional-Derivativo con redes neuronales multicapa

#### Análisis de estabilidad

**Cota 6.7** Sea el término  $K(\psi)\eta_{J3}$  (debido a la incertidumbre en la matriz Jacobiana) acotado como

$$(K(\psi)\eta_{J3})^T \Lambda_{J3} (K(\psi)\eta_{J3}) \leq \bar{\eta}_{J3}$$

donde  $\Lambda_{J3}$  es una matriz definida y  $\bar{\eta}_{J3}$  es una constante definida.

La ley de control visual con compensación mediante una red neuronal multicapa es:

$$\tau = K(\psi) \hat{J} K_p \tilde{x}_s - K_d \dot{q} + \hat{W}_t \sigma(\hat{V}_t x) \quad (6.65)$$

donde  $K_p$  y  $K_d$  son matrices positivas definidas y simétricas.  $q^d$  y  $\dot{q}^d$  son la trayectoria deseada y su velocidad. El siguiente teorema presenta un algoritmo estable de aprendizaje para el controlador (6.65).

**Teorema 6.8** Sea la trayectoria deseada acotada, y suponga que la siguiente condición de seguimiento se satisface:

$$K_d > \Lambda_g^{-1} + \Lambda_\sigma^{-1} + \tilde{x}_s^T K_p^T \Lambda_{J3}^{-1} K_p \tilde{x}_s + \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s$$

Se usan las siguientes leyes de aprendizaje para actualizar los valores de los pesos de la red neuronal multicapa:

$$\begin{aligned} \dot{\hat{W}}_t &= -K_w \sigma(V_t x) \dot{q}^T \\ \dot{\hat{V}}_t &= -K_v x W^* D_\sigma \dot{q}^T \end{aligned} \quad (6.66)$$

y se usa el siguiente control de zona muerta, entonces el error de seguimiento  $\tilde{x}_s$  es acotado.

- si  $\left\| \dot{q} \right\|^2 > \lambda_{\min}^{-1}(Q) (\bar{\eta}_{J3} + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_d)$  entonces, se usa la ley de control visual (6.65)
- si  $\left\| \dot{q} \right\|^2 \leq \lambda_{\min}^{-1}(Q) (\bar{\eta}_{J3} + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_d)$  entonces, se deja de usar la ley de control visual (6.65)

donde  $Q = K_d - [\Lambda_g^{-1} + \Lambda_\sigma^{-1} + \tilde{x}_s^T K_p^T \Lambda_{J_3}^{-1} K_p \tilde{x}_s + \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s]$ .

**Demostracin.** Se propone la siguiente función candidata de Lyapunov:

$$V = \frac{1}{2} \dot{q}^T M \dot{q} + \frac{1}{2} \tilde{x}_s^T K_p \tilde{x}_s + \frac{1}{2} tr \left( \tilde{W}_t^T K_w^{-1} \tilde{W}_t \right) + \frac{1}{2} tr \left( \tilde{V}_t^T K_v^{-1} \tilde{V}_t \right) \quad (6.67)$$

donde  $K_w$  y  $K_v$  son matrices positivas definidas. Derivando (6.67) a lo largo de las trayectorias (6.6), obtenemos:

$$\dot{V} = \dot{q}^T M \ddot{q} + \frac{1}{2} \dot{q}^T M \dot{q} + \dot{\tilde{x}}_s^T K_p \tilde{x}_s + tr \left( \tilde{W}_t^T K_w^{-1} \dot{\tilde{W}}_t \right) + tr \left( \tilde{V}_t^T K_v^{-1} \dot{\tilde{V}}_t \right)$$

Sustituyendo  $\ddot{q}$  de (6.8),

$$\begin{aligned} \dot{V} = & \frac{1}{2} \dot{q}^T \left\{ 2\tau - 2C(q, \dot{q}) \dot{q} - 2 \left[ G(q) + F_d(\dot{q}) \right] + 2M \dot{q} \right\} \\ & + \dot{\tilde{x}}_s^T K_p \tilde{x}_s + tr \left( \tilde{W}_t^T K_w^{-1} \dot{\tilde{W}}_t \right) + tr \left( \tilde{V}_t^T K_v^{-1} \dot{\tilde{V}}_t \right) \end{aligned}$$

Usando la *Propiedad 6.2*

$$\begin{aligned} \dot{V} = & \frac{1}{2} \dot{q}^T \left\{ 2\tau - 2 \left[ G(q) + F_d(\dot{q}) \right] \right\} \\ & + \dot{\tilde{x}}_s^T K_p \tilde{x}_s + tr \left( \tilde{W}_t^T K_w^{-1} \dot{\tilde{W}}_t \right) + tr \left( \tilde{V}_t^T K_v^{-1} \dot{\tilde{V}}_t \right) \end{aligned}$$

Se realiza la compensación de la gravedad y la fricción con (6.25)

$$\begin{aligned} \dot{V} = & \frac{1}{2} \dot{q}^T \left\{ 2\tau - 2 \left[ W^* \sigma(V^* x) + \eta_g \right] \right\} \\ & + \dot{\tilde{x}}_s^T K_p \tilde{x}_s + tr \left( \tilde{W}_t^T K_w^{-1} \dot{\tilde{W}}_t \right) + tr \left( \tilde{V}_t^T K_v^{-1} \dot{\tilde{V}}_t \right) \end{aligned}$$

y aplicando (6.59),

$$\begin{aligned} \dot{V} = & \dot{q}^T \left[ \tau - W^* \sigma(V^* x) + \eta_g - K(\psi) J K_p \tilde{x}_s - \dot{R}_L K(\psi) J K_p \tilde{x}_s \right] \\ & + tr \left( \tilde{W}_t^T K_w^{-1} \dot{\tilde{W}}_t \right) + tr \left( \tilde{V}_t^T K_v^{-1} \dot{\tilde{V}}_t \right) \end{aligned} \quad (6.68)$$

Usando (6.65) con  $\widehat{J} = -\eta_{J3} + J$ ,

$$\dot{V} = \dot{q}^T \left[ \begin{array}{l} -K(\psi) \eta_{J3} K_p \tilde{x}_s + K(\psi) J K_p \tilde{x}_s - K(\psi) J K_p \tilde{x}_s - \dot{R}_L K(\Psi) J K_p \tilde{x}_s \\ -K_d \dot{q} + \widehat{W}_t \sigma(\widehat{V}_t x) - W^* \sigma(V^* x) + \eta_g \\ +tr \left( \widehat{W}_t^T K_w^{-1} \dot{\widehat{W}}_t \right) + tr \left( \widehat{V}_t^T K_v^{-1} \dot{\widehat{V}}_t \right) \end{array} \right] \quad (6.69)$$

Mediante (6.27),

$$\dot{V} = -\dot{q}^T \left[ \begin{array}{l} K(\psi) \eta_{J3} K_p \tilde{x}_s + \dot{R}_L K(\psi) J K_p \tilde{x}_s + K_d \dot{q} + \widehat{W}_t \sigma(V_t x) + W^* D_\sigma \widehat{V}_t x + \nu_\sigma + \eta_g \\ +tr \left( \widehat{W}_t^T K_w^{-1} \dot{\widehat{W}}_t \right) + tr \left( \widehat{V}_t^T K_v^{-1} \dot{\widehat{V}}_t \right) \end{array} \right] \quad (6.70)$$

Por (6.66):

$$\dot{V} = -\dot{q}^T K(\psi) \eta_{J3} K_p \tilde{x}_s - \dot{q}^T \dot{R}_L K(\psi) J K_p \tilde{x}_s - \dot{q}^T K_d \dot{q} - \dot{q}^T [\nu_\sigma + \eta_g] \\ +tr \left[ \left( K_w^{-1} \dot{\widehat{W}}_t - \sigma(V_t x) \dot{q}^T \right) \widehat{W} \right] + tr \left[ \left( K_v^{-1} \dot{\widehat{V}}_t - x W^* D_\sigma \dot{q}^T \right) \widehat{V} \right] \quad (6.71)$$

Se usará la siguiente desigualdad matricial:

$$X^T Y + (X^T Y)^T \leq X^T \Lambda^{-1} X + Y^T \Lambda Y \quad (6.72)$$

para los términos:  $-\dot{q}^T [\nu_\sigma + \eta_g]$  y  $-\dot{q}^T \eta_J K_p \tilde{x}_s$  en (6.70) se obtienen las siguientes expresiones:

$$\begin{aligned} -\dot{q}^T [\nu_\sigma + \eta_g] &\leq \dot{q}^T (\Lambda_g^{-1} + \Lambda_\sigma^{-1}) \dot{q} + \bar{\eta}_g + \bar{\eta}_\sigma \\ -\dot{q}^T K(\psi) \eta_{J3} K_p \tilde{x}_s &\leq \dot{q}^T \tilde{x}_s^T K_p^T \Lambda_{J3}^{-1} K_p \tilde{x}_s \dot{q} + (K(\psi) \eta_{J3})^T \Lambda_{J3} (K(\psi) \eta_{J3}) \\ &\leq \dot{q}^T \tilde{x}_s^T K_p^T \Lambda_{J3}^{-1} K_p \tilde{x}_s \dot{q} + \bar{\eta}_{J3} \\ \dot{q}^T \dot{R}_L K(\psi) J K_p \tilde{x}_s &\leq \dot{q}^T \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s \dot{q} + \left( \dot{R}_L K(\psi) J \right)^T \Lambda_d \left( \dot{R}_L K(\psi) J \right) \\ &\leq \dot{q}^T \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s \dot{q} + \bar{\eta}_d \end{aligned} \quad (6.73)$$

Así,

$$\dot{V} \leq -\dot{q}^T Q \dot{q} + \bar{\eta}_{J3} + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_d \quad (6.74)$$

donde  $Q = K_d - [\Lambda_g^{-1} + \Lambda_\sigma^{-1} + \tilde{x}_s^T K_p^T \Lambda_{J_3}^{-1} K_p \tilde{x}_s + \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s]$ .

*Caso 1:* Si  $\|\dot{q}\|^2 > \lambda_{\min}^{-1}(Q) (\bar{\eta}_{J_3} + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_d)$  entonces, se usa la ley de control visual (6.65), se concluye que:

$$\begin{aligned} \dot{V}_t &\leq -\dot{q}^T Q \dot{q} + (\bar{\eta}_{J_3} + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_d) \\ &\leq -\lambda_{\min}(Q) \|\dot{q}\|^2 + (\bar{\eta}_{J_3} + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_d) < 0 \end{aligned}$$

por lo tanto,  $V_t$  es acotada.

*Caso 2:* Si  $\|\dot{q}\|^2 \leq \lambda_{\min}^{-1}(Q) (\bar{\eta}_{J_3} + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_d)$ , el término  $\frac{1}{2} \dot{q}^T M \dot{q}$  es acotado. Dado que la acción de control se deja de aplicar, los términos  $\frac{1}{2\alpha h} R^{-1} \tilde{x}_s^T K_p \tilde{x}_s$  y  $V_t$  se mantienen constantes.

Entonces  $V_t$  es acotado,  $\dot{q}$ ,  $\tilde{x}_s$ ,  $\tilde{W}_t$  y  $\tilde{V}_t$  son acotados. ■

### 6.3.5. Controlador visual Proporcional-Derivativo con observador y redes neuronales multicapa

#### Análisis de estabilidad

La ley de control visual con compensación mediante redes neuronales diferenciales y la velocidad  $\dot{\hat{q}}$  obtenida mediante el observador de modos deslizantes es:

$$\tau = K(\psi) \hat{J} K_p \tilde{x}_s - K_d \dot{\hat{q}} + \widehat{W}_t \sigma(\widehat{V}_t s) \quad (6.75)$$

donde  $K_p$  y  $K_d$  son matrices positivas definidas y simétricas.  $q^d$  y  $\dot{q}^d$  son la trayectoria y velocidad deseada.  $\widehat{W}_t$  y  $\widehat{V}_t$  son las matrices de pesos variantes en el tiempo de la red neuronal multicapa.

**Teorema 6.9** *Sea la trayectoria deseada acotada, y suponga que la siguiente condición de seguimiento se satisface:*

$$K_d > \Lambda_g^{-1} + \Lambda_\sigma^{-1} + \tilde{x}_s^T K_p^T \Lambda_{J_3}^{-1} K_p \tilde{x}_s + \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s + M^T \Lambda_{\varepsilon_1}^{-1} M + \dot{M}^T \Lambda_\varepsilon^{-1} \dot{M} \quad (6.76)$$

donde  $0 < \Lambda_1 = \Lambda_1^T \in \mathfrak{R}^{n \times n}$  es una matriz conocida. Se usan las siguientes leyes de aprendizaje para actualizar los valores de los pesos de la red neuronal multicapa:

$$\begin{aligned} \dot{\widehat{W}}_t &= -K_w \sigma(V_t s) \dot{\hat{q}}^T \\ \dot{\widehat{V}}_t &= -K_v s W^* D_\sigma^T \dot{\hat{q}} \end{aligned} \quad (6.77)$$

y se usa el siguiente control de zona muerta, entonces el error de seguimiento  $\tilde{x}_s$  es acotado.

- si  $\left\| \dot{\hat{q}} \right\|^2 > \lambda_{\min}^{-1}(Q) (\bar{\eta}_{J_3} + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_\varepsilon + \bar{\eta}_{\varepsilon_1} + \bar{\eta}_d)$  entonces, se usa la ley de control visual (6.75)
- si  $\left\| \dot{\hat{q}} \right\|^2 \leq \lambda_{\min}^{-1}(Q) (\bar{\eta}_{J_3} + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_\varepsilon + \bar{\eta}_{\varepsilon_1} + \bar{\eta}_d)$  entonces, se deja de usar la ley de control visual (6.75)

$$\text{donde } Q = K_d - \left[ \Lambda_g^{-1} + \Lambda_\sigma^{-1} + \tilde{x}_s^T K_p^T \Lambda_{J_3}^{-1} K_p \tilde{x}_s + \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s + M^T \Lambda_{\varepsilon_1}^{-1} M + \dot{M}^T \Lambda_\varepsilon^{-1} \dot{M} \right].$$

**Demostracin.** Se propone la siguiente función candidata de Lyapunov:

$$V = \frac{1}{2} \dot{\hat{q}}^T M \dot{\hat{q}} + \frac{1}{2} \tilde{x}_s^T K_p \tilde{x}_s + \frac{1}{2} \text{tr} \left( \tilde{W}_t^T K_w^{-1} \tilde{W}_t \right) + \frac{1}{2} \text{tr} \left( \tilde{V}_t^T K_v^{-1} \tilde{V}_t \right) \quad (6.78)$$

donde  $K_w$  y  $K_v$  son matrices positivas definidas. Derivando (6.78) a lo largo de las trayectorias (6.6), obtenemos:

$$\dot{V} = \dot{\hat{q}}^T M \ddot{\hat{q}} + \frac{1}{2} \dot{\hat{q}}^T \dot{M} \dot{\hat{q}} + \dot{\tilde{x}}_s^T K_p \tilde{x}_s + \text{tr} \left( \dot{\tilde{W}}_t^T K_w^{-1} \tilde{W}_t \right) + \text{tr} \left( \dot{\tilde{V}}_t^T K_v^{-1} \tilde{V}_t \right) \quad (6.79)$$

Ahora sustituyendo  $\ddot{\hat{q}}$  de (6.42) y despejando  $\ddot{q}$  de (6.8),

$$2M\ddot{\hat{q}} = M \left( \ddot{q} + \varepsilon_1 \right) = 2\tau - 2C \left( q, \dot{q} \right) \dot{q} - 2 \left[ G(q) + F_d \left( \dot{q} \right) \right] + 2M\varepsilon_1 \quad (6.80)$$

usando (6.80) y  $\dot{\hat{q}} = \dot{q} + \varepsilon$  en (6.79) se tiene:

$$\dot{V} = \frac{1}{2} \dot{\hat{q}}^T \left\{ 2\tau - 2C \left( q, \dot{q} \right) \dot{q} - 2 \left[ G(q) + F_d \left( \dot{q} \right) \right] + 2M\varepsilon_1 + 2\dot{M} \left( \dot{q} + \varepsilon \right) \right\} \\ + \dot{\tilde{x}}_s^T K_p \tilde{x}_s + \text{tr} \left( \dot{\tilde{W}}_t^T K_w^{-1} \tilde{W}_t \right) + \text{tr} \left( \dot{\tilde{V}}_t^T K_v^{-1} \tilde{V}_t \right)$$

Por la *Propiedad 6.2*,

$$\dot{V} = \dot{\hat{q}}^T \left\{ \tau - \left[ G(q) + F_d \left( \dot{q} \right) \right] + M\varepsilon_1 + \dot{M} \varepsilon \right\} \\ + \dot{\tilde{x}}_s^T K_p \tilde{x}_s + \text{tr} \left( \dot{\tilde{W}}_t^T K_w^{-1} \tilde{W}_t \right) + \text{tr} \left( \dot{\tilde{V}}_t^T K_v^{-1} \tilde{V}_t \right)$$

Aplicando la estimación de la gravedad y fricción (6.27),

$$\dot{V} = \dot{\hat{q}}^T \left[ \tau - (W^* \sigma(V^* s) + \eta_g) + M\varepsilon_1 + \dot{M} \varepsilon \right] \\ + \dot{\tilde{x}}_s^T K_p \tilde{x}_s + \text{tr} \left( \dot{\tilde{W}}_t^T K_w^{-1} \tilde{W}_t \right) + \text{tr} \left( \dot{\tilde{V}}_t^T K_v^{-1} \tilde{V}_t \right)$$

Sustituyendo (6.58),

$$\dot{V} = \dot{\hat{q}}^T \left[ \tau - (W^* \sigma(V^* s) + \eta_g) + M \varepsilon_1 + \dot{M} \varepsilon - K(\psi) J K_p \tilde{x}_s - \dot{R}_L K(\psi) J K_p \tilde{x}_s \right] \\ + tr \left( \tilde{W}_t^T K_w^{-1} \dot{\tilde{W}}_t \right) + tr \left( \tilde{V}_t^T K_v^{-1} \dot{\tilde{V}}_t \right)$$

Usando (6.75) con  $\hat{J} = -\eta_{J3} + J$ ,

$$\dot{V} = \dot{\hat{q}}^T \left[ \begin{aligned} & \left\{ -K(\psi) \eta_{J3} K_p \tilde{x}_s + K(\psi) J K_p \tilde{x}_s + K_d \dot{q} - K_d \varepsilon + \widehat{W}_t \sigma(\widehat{V}_t x) \right\} \\ & -K(\psi) J K_p \tilde{x}_s - \dot{R}_L K(\psi) J K_p \tilde{x}_s - (W^* \sigma(V^* s) + \eta_g) + M \varepsilon_1 + \dot{M} \varepsilon \end{aligned} \right] \\ + tr \left( \tilde{W}_t^T K_w^{-1} \dot{\tilde{W}}_t \right) + tr \left( \tilde{V}_t^T K_v^{-1} \dot{\tilde{V}}_t \right)$$

$$\dot{V} = -\dot{\hat{q}}^T \left[ \begin{aligned} & -K(\psi) \eta_{J3} K_p \tilde{x}_s - \dot{R}_L K(\psi) J K_p \tilde{x}_s + K_d \dot{q} - K_d \varepsilon + M \varepsilon_1 + \dot{M} \varepsilon \\ & + \widehat{W}_t \sigma(V_t s) + W^* D_\sigma \tilde{V}_t s + v_\sigma + \eta_g \end{aligned} \right] \\ + tr \left( \tilde{W}_t^T K_w^{-1} \dot{\tilde{W}}_t \right) + tr \left( \tilde{V}_t^T K_v^{-1} \dot{\tilde{V}}_t \right)$$

y

$$\dot{V} = -\dot{\hat{q}}^T \left[ \begin{aligned} & -K(\psi) \eta_J K_p \tilde{x}_s - \dot{R}_L K(\psi) J K_p \tilde{x}_s + K_d \dot{q} - K_d \varepsilon + M \varepsilon_1 + \dot{M} \varepsilon \\ & + \widehat{W}_t \sigma(V_t s) + W^* D_\sigma \tilde{V}_t s + \eta_\sigma + v_g \end{aligned} \right] \\ + tr \left( \tilde{W}_t^T K_w^{-1} \dot{\tilde{W}}_t \right) + tr \left( \tilde{V}_t^T K_v^{-1} \dot{\tilde{V}}_t \right)$$

con (6.77) tenemos:

$$\dot{V} = -\dot{\hat{q}}^T K(\psi) \eta_{J3} K_p \tilde{x}_s - \dot{\hat{q}}^T \dot{R}_L K(\psi) J K_p \tilde{x}_s - \dot{\hat{q}}^T K_d \dot{q} - \dot{\hat{q}}^T \eta_J K_p \tilde{x}_s \\ - \dot{\hat{q}}^T [v_\sigma + \eta_g] - \dot{\hat{q}}^T M \varepsilon_1 - \dot{\hat{q}}^T \dot{M} \varepsilon \quad (6.81) \\ + tr \left[ \left( K_w^{-1} \dot{\tilde{W}}_t - \sigma(V_t s) \dot{\hat{q}}^T \right) \tilde{W} \right] + tr \left[ \left( K_v^{-1} \dot{\tilde{V}}_t - s W^* D_\sigma^T \dot{\hat{q}}^T \right) \tilde{V} \right]$$

Los términos  $tr \left[ \left( K_w^{-1} \dot{\tilde{W}}_t - \sigma(V_t s) \dot{\hat{q}}^T \right) \tilde{W} \right]$ ,  $tr \left[ \left( K_v^{-1} \dot{\tilde{V}}_t - s W^* D_\sigma^T \dot{\hat{q}}^T \right) \tilde{V} \right]$  se hacen cero, pues la red neuronal multicapa con su regla de aprendizaje es capaz de estimar adecuadamente la gravedad y la fricción.

Se usará la siguiente desigualdad matricial:

$$X^T Y + (X^T Y)^T \leq X^T \Lambda^{-1} X + Y^T \Lambda Y \quad (6.82)$$

la cual es válida para cualquier  $X, Y \in \mathfrak{R}^{n \times k}$  y para cualquier matriz definida positiva a  $0 < \Lambda = \Lambda^T \in \mathfrak{R}^{n \times n}$ .

Ahora los términos  $\dot{\hat{q}}^T M \varepsilon_1$ ,  $-\dot{\hat{q}}^T M \varepsilon$ ,  $-\dot{\hat{q}}^T [\nu_\sigma + \eta_g]$ ,  $-K(\psi) \eta_J K_p \tilde{x}_s$ ,  $-R_L K(\psi) J K_p \tilde{x}_s$  de (6.81) se les aplica la desigualdad (6.82) y se reescriben como:

$$\begin{aligned} \dot{\hat{q}}^T M \varepsilon_1 &\leq \dot{\hat{q}}^T M^T \Lambda_{\varepsilon_1}^{-1} M \dot{\hat{q}} + \bar{\eta}_{\varepsilon_1} \\ \dot{\hat{q}}^T M \varepsilon &\leq \dot{\hat{q}}^T M^T \Lambda_\varepsilon^{-1} M \dot{\hat{q}} + \bar{\eta}_\varepsilon \\ -\dot{\hat{q}}^T [\nu_\sigma + \eta_g] &\leq \dot{\hat{q}}^T (\Lambda_g^{-1} + \Lambda_\sigma^{-1}) \dot{\hat{q}} + \bar{\eta}_g + \bar{\eta}_\sigma \\ -\dot{\hat{q}}^T K(\psi) \eta_{J3} K_p \tilde{x}_s &\leq \dot{\hat{q}}^T \tilde{x}_s^T K_p^T \Lambda_{J3}^{-1} K_p \tilde{x}_s \dot{\hat{q}} + (K(\psi) \eta_{J3})^T \Lambda_d (K(\psi) \eta_{J3}) \\ &\leq \dot{\hat{q}}^T \tilde{x}_s^T K_p^T \Lambda_{J3}^{-1} K_p \tilde{x}_s \dot{\hat{q}} + \bar{\eta}_{J3} \\ \dot{\hat{q}}^T R_L K(\psi) J K_p \tilde{x}_s &\leq \dot{\hat{q}}^T \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s \dot{\hat{q}} + \left( R_L K(\psi) J \right)^T \Lambda_d \left( R_L K(\psi) J \right) \\ &\leq \dot{\hat{q}}^T \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s \dot{\hat{q}} + \bar{\eta}_d \end{aligned} \quad (6.83)$$

Ahora se sustituye (6.83) y se obtiene:

$$\begin{aligned} \dot{V} &\leq -\dot{\hat{q}}^T \left[ K_d - \left( \Lambda_g^{-1} + \Lambda_\sigma^{-1} + \tilde{x}_s^T K_p^T \Lambda_{J3}^{-1} K_p \tilde{x}_s + \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s + M^T \Lambda_{\varepsilon_1}^{-1} M + \dot{M}^T \Lambda_\varepsilon^{-1} \dot{M} \right) \right] \dot{\hat{q}} \\ &\quad + \bar{\eta}_{J3} + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_\varepsilon + \bar{\eta}_{\varepsilon_1} + \bar{\eta}_d \\ \dot{V}_t &\leq -\dot{\hat{q}}^T Q \dot{\hat{q}} + \bar{\eta}_{J3} + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_\varepsilon + \bar{\eta}_{\varepsilon_1} + \bar{\eta}_d \end{aligned}$$

con  $Q = K_d - \left( \Lambda_g^{-1} + \Lambda_\sigma^{-1} + \tilde{x}_s^T K_p^T \Lambda_{J3}^{-1} K_p \tilde{x}_s + \tilde{x}_s^T K_p^T \Lambda_d^{-1} K_p \tilde{x}_s + M^T \Lambda_{\varepsilon_1}^{-1} M + \dot{M}^T \Lambda_\varepsilon^{-1} \dot{M} \right) > 0$ .

*Caso 1:* Si  $\left\| \dot{\hat{q}} \right\|^2 > \lambda_{\min}^{-1}(Q) (\bar{\eta}_{J3} + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_\varepsilon + \bar{\eta}_{\varepsilon_1} + \bar{\eta}_d)$  entonces, se usa la ley de control visual (6.75), se concluye que:

$$\begin{aligned} \dot{V}_t &\leq -\dot{\hat{q}}^T Q \dot{\hat{q}} + (\bar{\eta}_{J3} + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_\varepsilon + \bar{\eta}_{\varepsilon_1} + \bar{\eta}_d) \\ \dot{V}_t &\leq -\lambda_{\min}(Q) \left\| \dot{\hat{q}} \right\|^2 + (\bar{\eta}_{J3} + \bar{\eta}_\sigma + \bar{\eta}_g + \bar{\eta}_\varepsilon + \bar{\eta}_{\varepsilon_1} + \bar{\eta}_d) < 0 \end{aligned} \quad (6.84)$$

por lo tanto,  $V_t$  es acotada.

*Caso 2:* Si  $\left\| \dot{\hat{q}} \right\|^2 \leq \lambda_{\min}^{-1}(Q) (\bar{\eta}_{J3} + \bar{\eta}_{\sigma} + \bar{\eta}_g + \bar{\eta}_{\varepsilon} + \bar{\eta}_{\varepsilon_1} + \bar{\eta}_d)$ , el término  $\frac{1}{2} \dot{\hat{q}}^T M \dot{\hat{q}}$  es acotado. Dado que la acción de control se deja de aplicar, los términos  $\frac{1}{2\alpha h} R^{-1} \tilde{x}_s^T K_p \tilde{x}_s$  y  $V_t$  se mantienen constantes.

Entonces  $V_t$  es acotado,  $\dot{\hat{q}}$ ,  $\tilde{x}_s$ ,  $\tilde{W}_t$  y  $\tilde{V}_t$  son acotados. ■



# Capítulo 7

## Conclusiones

Este reporte de tesis, presentó el trabajo realizado durante tres años y medio (enero 2000 a mayo 2003). En este período se obtuvieron diversos resultados en el campo de la visión por computadora aplicada al control visual de robots manipuladores. Dichos resultados se enumeran a continuación:

1. **Diseño e implementación de una nueva plataforma de estéreo visión**, incluyendo la selección de los componentes necesarios que son: el robot manipulador A465, dos cámaras analógicas Pulnix, dos lentes Cosmimar C815b y una tarjeta digitalizadora de *National Instruments*. Además, se desarrolló un nuevo software especializado encargado de comunicarse con estos componentes y aplicar los algoritmos que fueron desarrollados en este trabajo.

El diseño del software fue modular, por lo que es portable a otros sistemas operativos como Unix o Linux. Solamente se tiene que reprogramar el módulo de comunicación con el robot y las cámaras.

2. **Diseño e implementación de un nuevo grupo de algoritmos de visión por computadora**, cuyo objetivo particular es extraer la información visual necesaria para el modelo óptico de estéreo visión. Esta información consiste en el valor de los centroides de las tres marcas visuales.

La adecuada selección de la ubicación de las marcas visuales, así como su tamaño, forma e incluir dentro de una de estas marcas un círculo negro, permitió establecer criterios simples para relacionar las marcas visuales con la parte correspondiente del robot A465 (base, codo y órgano terminal).

3. **Diseño e implementación de un modelo capaz de calcular la posición en el espacio tridimensional de un punto.** Se le nombra modelo óptico de estéreo visión, pues se realizó un estudio detallado de los lentes Cosmicar C815b.

Este modelo fue diseñado con el objetivo de ser sencillo y fácil de implementar. Se basa en información visual y el valor del ángulo de rotación de las cámaras con respecto al *eje x* del robot. Con estos valores calcula las distancias entre las cámaras y el robot. Dado que estas distancias se recalculan cada vez que se desea obtener la posición de un punto desconocido, es posible mover las cámaras hacia adelante o atrás (dentro de un rango aceptable). Para los ángulos de rotación, se implementó una rutina que calcula automáticamente estos valores usando solamente información visual. Con estas dos características adicionales el modelo logrado no sólo cumplió satisfactoriamente el objetivo planteado, sino además brinda la flexibilidad de colocar las cámaras en diversas posiciones sin que se vean afectados sus cálculos. Esto puede ser de gran utilidad si se desea implementar este modelo en ambientes no estructurados.

Por otra parte, como se puede ver en el capítulo 4, el cálculo de la posición en el espacio del órgano terminal se separó en dos partes. En la primera parte se obtiene el valor de la coordenada  $z$  y en la segunda el valor de las coordenadas  $(x, y)$ , esto permitió simplificar el número de ecuaciones a utilizar mejorando notablemente el desempeño del modelo.

4. **Diseño e implementación de un controlador visual difuso.** El proceso de diseño de este tipo de controladores es muy adecuado para realizar implementaciones en un periodo corto de tiempo en ambientes no estructurados y después en la marcha poder realizar pequeños ajustes hasta lograr el comportamiento deseado del sistema.

Se usó la metodología presentada en el capítulo 5 para diseñar e implementar un con-

trolador visual de este tipo. Los resultados experimentales de aplicar este controlador en la plataforma de estéreo visión fueron aceptables, con la restricción de que se use para tareas que no requieran mucha precisión.

5. **Se realizó el análisis de estabilidad de un nuevo controlador visual PD con observador y redes neuronales multicapa para un robot de dos y tres grados de libertad.** El observador se empleó para estimar la velocidad de las articulaciones del robot y con las redes neuronales se realizó la compensación de las fuerzas de gravedad y fricción. Aquí, la aportación mas importante fue el uso del observador de modos deslizantes para completar la información necesaria para las leyes de control propuestas.

## 7.1. Trabajo futuro

Algunas trabajos a realizar para continuar trabajando en esta línea de investigación son:

1. Diseñar e implementar una plataforma de estéreo visión usando cámaras digitales, pues éstas proporcionan mayor resolución en las imágenes y comparar los resultados con los obtenidos en esta tesis.
2. Implementación de los controladores visuales del capítulo 6.
3. Realizar el análisis de estabilidad para el caso de seguimiento de un objeto.
4. Desarrollar el análisis de estabilidad para el controlador visual difuso del capítulo 5.



# Bibliografía

- [1] A. Albert, Regression and the Moore-Penrose Pseudoinverse, Academic Press, 1972.
- [2] S. Arimoto, T. Naniwa, V. Parra-Vega, and L. Whitcomb, A quasi-natural potential and its role in design of hyper-stable PID servo-loop for robotic systems, *Proc. CAI Pacific Symp. Control and Industrial Automation Application*, Hong Kong, 1994, pp.110-117.
- [3] C.C. Cheah, S. Kawanura and S. Arimoto, Feedback Control for Robotic Manipulator with Uncertain Kinematics and Dynamics, Proc. IEEE International Conference on Robotics & Automation, Leuven, Belgium, 1998, pp. 3607-3612.
- [4] G.S. Bell, C. W. Hulls & W. J. Wilson, Relative End-Effector control using cartesian position based visual servoing, *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, 1996.
- [5] Bellman, R.E., and L.A. Zadeh, Decision-making in a fuzzy environment, *Management Science*, 17, no. 4, pp. 141-164.
- [6] L. Behera and N. Kirubanandan, A hybrid neural control scheme for visual-motor coordination, *IEEE Control System*, Vol.19, No.4, 34-41, 1999 *Automation*, Vol.12, No.5, 759-766, 1996.
- [7] P. Castillo-García, Plataforma de control visual para servomecanismos, *Tesis de maestría*, CINVESTAV-IPN, 2000.

- [8] F.Conticelli and B.Allotta, Robust Stabilization of Second-Order Image-Based Affine Systems, *Systems & Control Letters*, 245-253, Vol.39, 2000.
- [9] P. I. Corke, G. D. Hager & S. Hutchinson, A tutorial on visual servo control, *IEEE transactions on Robotics and Automation*, vol. 12, no. 5, 1996.
- [10] M. Chavarria, A. Soria and P. Wiederhold, Percepcion monocular para el control de robots, CLCA, Guadalajara, México D.F, 2002.
- [11] F.Chaumette, Potential problems of unstability and divergence in image-based and position-based visual servoing, *Proc. Europe Control Conference*, Karlsruhe, Germany, 1999
- [12] C.C.Cheah, S.Kawamura, S.Arimoto and K.Lee, PID Control of Robotic Manipulator with Uncertain Jacobian Matrix, *Proc. IEEE Conference on Robotic & Automation*, Detroit, USA, 1999.
- [13] C.C.Cheah, K.Lee, S.Kawamura and S.Arimoto, Asymptotic Stability of Robot Control with Approximate Jacobian Matrix and Its Application to Visual Servoing, submit to *40th Conference on Decision & Control*, Orlando, USA, 2001.
- [14] L.O.Chua and L.Yang, Cellular neural networks:theory, *IEEE Trans. on Circuits and Systems*, Vol.10, No.35, 1257-1272, 1988
- [15] J.J. Craig, Introduction to Robotics mechanics and control (Addison-Wesley Press, 1989).
- [16] CRS Automation Solutions, Inc. en <http://www.crsrobotics.com>, 2001.
- [17] *User Manual for CRS A465 Robot arm controller*. CRS Robotics Literature. CRS Robotics Corporation Products, 1997.
- [18] CRS Robotics Literature. *Technical Data for CRS A465*. CRS Robotics Corporation Products, 1997.

- [19] CRS Robotics Literature. *Programing RAPL & RAPL II for CRS robotics*. CRS Robotics Corporation Products, 1997.
- [20] G.Cybenko, Approximation by Superposition of Sigmoidal Activation Function, *Math.Control, Sig Syst*, Vol.2, 303-314, 1989.
- [21] J. Denavit and R.S. Hartenberg, A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices, *Journal of Applied Mechanics*, 1955, 215-221.
- [22] B.Espiau, F.Chaumette and O.Rives, A new approach to visual servoing in robotics, *IEEE Trans. on Robotics and Automation*, Vol.8, No.3, 313-326, 1992.
- [23] R. Garrido, A. Soria, P. Castillo, I. Vásquez, A Visual Servoing Architecture for Controlling Electromechanical Systems, *Proceedings of the 2001 IEEE International Conference on Control Applications*, pp. 35-40, September 5-7, 2001, México City, México.
- [24] J.A.Gangloff, M.F. Mathelin and G.Abba, High performance 6 DOF visual servoing using generalized predictive control, *Proc. Europe Control Conference*, Karlsruhe, Germany, 1999.
- [25] B.K.Ghosh and E.P.Loucks, A Realization Theory for Perspective Systems with Application to Parameter Estimation Problems in Machine Vision, *IEEE Trans. on Automatic Control*, Vol.41, No.12, 1706-1722, 1996.
- [26] R.C. González, R.E. Woods, *Tratamiento digital de imágenes*, Addison-Wesley, pag. 58, 1996.
- [27] R.C. González, R.E. Woods, *Tratamiento digital de imágenes*, Addison-Wesley, pag. 447, 1996.
- [28] E.Grosso, G.Metta, A.Oddera and G.Sandini, Robust visual servoing in 3-D reaching tasks, *IEEE Trans. on Robotics and Automation*, Vol.12, No.5, 732-741, 1996.

- [29] A. Heredia and W.Yu, A Modified PD Control of Robot Manipulator Using Neural Network Compensator, *Internal Joint Conference on Neural Networks, IJCNN '99*, Washington USA, Vol.4, 1999-2004, 1999.
- [30] K. Hashimoto, Visual Servoing Real-Time Control of Robot Manipulators Based on Visual Sensory Feedback, *World Scientific*, Singapore, 1993.
- [31] K.Hashimoto and H.Kimura, Visual Servoing with Nonlinear Observer, *Proc. IEEE Int. Conf. Robotics and Automation*, 484-498, 1995.
- [32] K.Hashimoto and T.Noritsugu, Visual Servoing with Linearized Observer, *Proc. IEEE Int. Conf. Robotics and Automation*, 263-268, 1999.
- [33] S.Hutchison, G.D.Hager and P.Corke, A Tutorial on Visual Servo Control, *IEEE Trans. on Robotics and Automation*, Vol.12, No.5, 651-670, 1996.
- [34] R.M.Haralick and L.G.Shapiro, *Computer and Robot Vision II*, Addison-Wesley Pub. Co., 1993.
- [35] A. Isidori, Nonlinear Control Systems, *Springer-Verlag*, London, 1995.
- [36] M.Jagersand, Visual servoing using trust region methods and estimation of the full coupled visual-motot Jacobian, *IASTED Application of Robotics and Control*, 1996.
- [37] R. Kelly & F. Reyes, Experimental evaluation of fixed-camera direct visual controllers on a direct-drive robot, *International Conference on Robotics and Automation*, Leuven, Belgium, 1998.
- [38] R. Kelly, Global Positioning on Robot Manipulators via PD Control Plus A Class of Nonlinear Integral Actions, *IEEE Trans. Automat. Contr.*, vol 43, No. 7, 934-938, 1998.
- [39] R. Kelly , Robust Asymptotically stable visual servoing of Planar Robots, *IEEE Trans. On Robotics and Autamation*, vol. 12, No. 5, 759-766, 1996.

- [40] R.Kelly, A Tuning Procedure for Stable PID Control of Robot Manipulators, *Robotica*, Vol.13, 141-148, 1995.
- [41] Y.H.Kim and F.L.Lewis, Neural Network Output Feedback Control of Robot Manipulator, *IEEE Trans. Neural Networks*, Vol.15, 301-309, 1999.
- [42] A.J.Koivo and N.Houshangi, Real-time version feedback for servoing robotic manipulator with self-tuning controller, *IEEE Trans. on Systems, Man and Cybernetics*, Vol.21, No.1, 134-141, 1992.
- [43] Gerardo Loreto, Visual Servoing con compensación utilizando redes neuronales, Tesis de Maestría, Departamento de Control Automático, CINVESTAV-IPN, Junio del 2002.
- [44] Gerardo Loreto, Wen Yu and Ruben Garrido, Stable Visual Servoing with Neural Network Compensation, *Proceedings of the 2001 IEEE International Conference on Control Applications*, pp. 183-188, September 5-7, 2001, México City, México.
- [45] Cosmocar CCTV Lenses en la web: <http://www.rmassa.com/manu/cosmicar.htm>, Mayo 2003.
- [46] F.L.Lewis, A.Yesildirek and K.Liu, Multilayer neural-net robot controller with guaranteed tracking performance, *IEEE Trans. on Neural Networks*, Vol.7, No.2, 388-399, 1996.
- [47] F.L Lewis, S. Jagannathan & A.Yesildirek, *Neural Network Control of Robot Manipulators and Nonlinear Systems*, Taylor & Francis, Philadelphia 1999.
- [48] Mamdani, E.H, and S. Assilian, An experiment in linguistic synthesis with a fuzzy logic controller, *Int. J. Man Mach. Studies*, 7, 1, pp. 1-13.
- [49] R. M. Murray, Z. Li, S. S. Sastry, *A mathematical introduction to robotics manipulation*, CRC Press, 1993.
- [50] National Instruments. *Getting started with your Imaq PCI-1408*. Noviembre 1996.

- [51] National Instruments, Inc. Cables Accessories in <http://sine.ni.com/apps/we/>, 2000.
- [52] S.Nicosia and A.Tornambe, High-Gain Observers in the State and Parameter Estimation of Robots Having Elastic Joints, *System & Control Letter*, Vol.13, 331-337, 1989
- [53] R.Ortega and M.W.Spong, Adaptive Motion Control of Rigid Robot: A Tutorial, *Automatica*, Vol.25, no.6, 877-888, 1989.
- [54] N. P. Papanikolopoulos & P. K. Khosla, Adaptive robotic visual tracking: Theory and Experiments, *IEEE Transactions on Automatic Control*, vol. 38, no. 2, 1993.
- [55] Pulnix American Inc., *Technical Features of TM-72EX B/W mini CCD camera*. 1999.
- [56] A.A.Rizzi and D.E.Koditchek, An Active Visual Estimator doe Dexterous Manipulator, *IEEE Trans. on Robotics and Automation*, Vol.12, No.5, 697-713, 1996.
- [57] V.Santibanez and R.Kelly, Global Asymptotic Stability of the PD Control with Computed Feedforward in Closed Loop with Robot Manipulators, *Proc.14th IFAC World Congress*, 197-203, Beijing, 1999.
- [58] A. C. Sanderson, L.E. Weiss, Image-based visual servo control using relational graph error signals, *Proc. IEEE*, pp.1074-1077, 1980.
- [59] A. Soria,P. Wiederhold and R. Garrido, Dynamic Threshold and Contour Detection: A more Robust Visual Centroid Recognition, CLCA, Guadalajara, México D.F, 2002.
- [60] M. Spong, M. Vidyasagar, *Robot Dynamics and Control*, New York, Wiley, 1989.
- [61] Sugeno, M., and M. Nishida, Fuzzy control of model car, *Fuzzy Sets and Systems*, pp. 103-113.
- [62] M.Takegaki and S.Arimoto, A New Feedback Method for Dynamic control of Manipulator, *ASME J. Dynamic Syst. Measurement, and Contr.*, Vol.103, 119-125, 1981.

- [63] Prácticas de Laboratorio de la Universidad La Salle, Talleres y Laboratorios, Laboratorio de Robótica. Universidad La Salle, 1999.
- [64] Trucco Emanuele, Verri Alesandro, Introductory Techniques for 3-D Computer Vision, Prentice Hall, pag. 68, 1998.
- [65] Trucco Emanuele, Verri Alesandro, Introductory Techniques for 3-D Computer Vision, Prentice Hall, pag. 140, 1998.
- [66] P.Tomei, Adaptive PD Controller for Robot Manipulator, *IEEE Tran. on Automatic Control*, Vol. 36, 556-570, 1992.
- [67] M. Vidyasagar, *Nonlinear Systems Analysis*, Englewood Cliffs, NJ: Prentice-Hall, 1978.
- [68] <http://verona.fi-p.unam.mx/fardi/pagina/VISONCOM.htm>, junio 2003.
- [69] Wang L., A course in fuzzy systems and control, *Prentice Hall*, capítulo 9, 1997.
- [70] Marco A.Moreno, Wen Yu and Alexander S. Poznyak, Stable 3-D Visual Servoing: An Experimental Comparison, *2001 IEEE Conference on Control Applications* , Mexico City, Mexico, 2001
- [71] Zadeh, L. A., Fuzzy Sets, *Informat. Control*, 8, pp. 338-353.
- [72] Zadeh, L. A., From circuit theory to systems theory, *Proc. Institution of Radio Engineers*, 50, pp. 856-865.
- [73] Zadeh, L. A., Fuzzy Algorithms, *Information and Control*, 12, no.2, pp. 94-102.
- [74] Zadeh, L. A., Similarity relations and fuzzy ordering, *Information Sciences*, 3, no.2, pp. 177-200.
- [75] Zadeh, L. A., Outline of a new approach to the analysis of complex systems and decision processes, *IEEE Trans. on Systems, Man and Cybern.*, 3, 1, pp.28-44.

- [76] E.Zergeroglu, D.M.Dawson, M.S. de Queiroz and S.Nagarkatti, Robust Visual-Servo Control of Robot Manipulators in the Presence of Uncertainty, *Proc. 38th Conference on Decision & Control*, 4137-4142, Phoenix, Arizona, USA, 1999.

# Apéndice A

## Modelo de Grosso

### A.1. Modelo de Grosso

A continuación se describe el modelo de estéreo visión reportado por Grosso y sus colegas [28], el cual es similar al propuesto en este trabajo. El objetivo es poder establecer, más adelante, las diferencias entre ese modelo y el propuesto en esta tesis. Debido a que el modelo de Grosso se incorporará con la parte de control, no fue posible probarlo en la plataforma de visión artificial.

Este modelo está basado en el espacio visual  $VS$  ( ver figura A.1). Usando el modelo conocido como *Pin-Hole* [28] se obtienen las siguientes ecuaciones para la cámara izquierda y derecha:

$$\begin{aligned}x_r &= z_r \frac{u_r}{\alpha} & x_l &= z_l \frac{u_l}{\alpha} \\y_r &= z_r \frac{v_r}{\alpha} & y_l &= z_l \frac{v_l}{\alpha}\end{aligned}\tag{A.1}$$

donde:

- $(u_l, v_l)$  y  $(u_r, v_r)$  son los valores de las coordenadas del punto en la imagen de la cámara derecha e izquierda.
- $\alpha$  incluye el valor de la distancia focal y el valor de la resolución de las imágenes.



En una vecindad del punto fijo, como se muestra en [28], las ec. A.2 se pueden reducir como:

$$\begin{aligned}x_{vs} &\simeq \frac{K(\Psi, \mathbf{b})}{\alpha} \cdot u_r \\y_{vs} &\simeq \frac{K(\Psi, \mathbf{b})}{\alpha} \cdot v_r \\z_{vs} &\simeq \frac{K(\Psi, \mathbf{b})}{\alpha \tan \Psi} \cdot (u_r - u_l)\end{aligned}\tag{A.3}$$

donde  $K(\Psi, \mathbf{b}) = \frac{b_1}{\sin \Psi}$ .

Si una estimación del ángulo  $\Psi$  no está disponible, el espacio visual no es cartesiano y las variables de estado del sistema se verán afectadas por errores.



# Apéndice B

## Glosario

### B.1. Términos de control visual [43]

#### B.1.1. Tipos de cámaras de video

1. **Cámaras analógicas:** Estos dispositivos son susceptibles a ruidos e interferencias debido a la conexión entre la cámara y la tarjeta de adquisición de datos, la cual realiza la conversión de la señal analógica a digital.
2. **Cámaras digitales:** En estos mecanismos la conversión de la señal analógica a digital es efectuada por la cámara, lo cual implica que existe poco ruido. La velocidad de captura de imágenes puede ser mayor que en las cámaras analógicas. Son mas costosas que las analógicas.

#### B.1.2. Configuraciones de la cámara

1. **Configuración de cámara en mano:** En esta estructura, la cámara se coloca sobre el órgano terminal del robot manipulador, lo que permite tener información visual de su espacio de trabajo.
2. **Configuración de cámara fija:** Aquí, la cámara se acomoda en un lugar fijo donde

se pueda obtener información visual del robot completo y de su espacio de trabajo (todo o una parte).

### B.1.3. Tipos de control visual dependiendo del uso de la información visual

1. **Control estático:** Se hace uso de la información visual una sola vez para determinar la posición deseada del órgano terminal del robot manipulador. Depende de los valores del estado del sistema y de señales de referencia externas.
2. **Control dinámico:** Éste hace uso de la información visual continuamente para controlar al robot [31] y depende de los valores del estado del sistema, de señales de referencia externas y de variables de estado adicionales que involucran la dinámica del sistema [35].

### B.1.4. Tipos de arquitectura dependiendo del lazo de retroalimentación

Esta clasificación fue presentada por primera vez en [58].

1. **Ver y Mover**<sup>1</sup>: Esta estructura tiene dos lazos de retroalimentación, por un lado el lazo del sistema de visión por computadora provee los datos que serán usados como referencia por el otro lazo que está a nivel articular. El segundo lazo lleva a cabo la retroalimentación articular y se encarga de la estabilización interna del robot manipulador.
2. **Control visual**<sup>2</sup>: Esta estructura tiene solo un lazo de retroalimentación, en el cual, usando la información visual se estabiliza el robot calculando directamente las acciones de control para las articulaciones.

---

<sup>1</sup>*look and move*, por su término en inglés.

<sup>2</sup>*Visual Servoing*, por su término en inglés.

### B.1.5. Tipos de control dependiendo la información que reciba el controlador

1. **Control basado en la posición:** En esta estrategia las características  $x_s$  son obtenidas de la imagen y usadas junto con modelos conocidos del sistema, como lo son: el modelo cinemático del robot manipulador, el de la cámara y el modelo geométrico del objeto a alcanzar, con el fin de determinar la posición de dicho objeto con respecto al órgano terminal del robot manipulador. La retroalimentación es utilizada para reducir el error de posicionamiento. Este enfoque no es adecuado para ambientes no estructurados, debido a que se requiere información *a priori*.
2. **Control basado en la imagen:** En este las mediciones para calcular la acción de control son obtenidas directamente de las características  $x_s$  obtenidas de la imagen. El objetivo de control es hacer al error (entre el órgano terminal y el objeto a alcanzar) en la imagen  $\tilde{x}_s$  igual a cero.