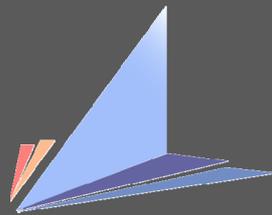


# SISTEMA DE CONTROL MULTISENSORIAL APLICADO A TAREAS DE MANIPULACIÓN AUTOMÁTICA

Gabriel Jesús García Gómez

Director:  
Jorge Pomares Baeza

Memoria de Suficiencia Investigadora  
Doctorado: Interuniversitario en Automática y Robótica



Escuela Politécnica Superior de Alicante



**Universitat d'Alacant**  
**Universidad de Alicante**

Julio 2007



# SISTEMA DE CONTROL MULTISENSORIAL APLICADO A TAREAS DE MANIPULACIÓN AUTOMÁTICA

Gabriel Jesús García Gómez

Director: Jorge Pomares Baeza

Memoria de Suficiencia Investigadora  
Doctorado: Interuniversitario en Automática y Robótica



Escuela Politécnica Superior de Alicante



Universitat d'Alacant  
Universidad de Alicante

Julio 2007



# Agradecimientos

---

Quizás me olvide, o quizás no. Si no te nombro no pienses que no me acuerdo de ti. Durante estos dos años de investigación dentro del grupo AUROVA me he apoyado en tres pilares básicos.

El primero de ellos ha sido Jorge. A Jorge le debo mucho de lo que soy en estos momentos. Sé que a veces le quita el sueño llevarme por el buen “camino”... sé que sin él no existirían estas líneas, estas hojas... sé que algún día me ganará al tenis... Sabes que te aprecio y admiro. Tus críticas siempre son acertadas, tus correcciones las mejores (no hay forma de pillarte), y tus ánimos siempre han llegado en el momento justo. Me siento orgulloso de ser tu amigo.

Fernando me ha apoyado desde el principio, ha buscado siempre lo mejor para mí, me ha dado la posibilidad de realizar estas investigaciones y sobre todo, ha confiado en mí ciegamente. Me ha hecho sentirme muy respaldado.

El laboratorio, mi AUROVA Boreal... el segundo pilar. María ha pasado a ser mi hermanita (tú sabes cuánto te admiro), Carolina es nuestra niña alocada (no cambies nunca). Son mis chicas... ellas lo saben. Hemos pasado momentos inolvidables y las quiero mucho. Juan Antonio es un amigo increíble. Es un crack. Debes saber que te admiro mucho. Jara es otro crack (estoy rodeado de ellos y ellas). Es incansable, muy serio en su trabajo y gran amigo fuera. Man espero que siga mucho tiempo con nosotros, es el COA personificado y siempre me ha gustado tenerlo cerca. Y Gonzalo siempre me hace reír. Sabe siempre como rebajar la tensión con alguna gracia (aunque algún día le suelto a mi gato...). Todos me han aportado mucho, a todos los aprecio y me gustaría seguir siendo su amigo siempre. Gracias por hacerme reír cada día.

A Laura tengo que agradecerle también mucho, qué bien me ha cuidado siempre con sus bizcochos... ¡madre! Gracias por prestarme a Jorge tanto tiempo.

Al resto de compañeros del grupo AUROVA, a Paco, Pablo y Santiago, también debo agradecerles su ayuda en un momento u otro de este periodo de investigación.

No quiero terminar sin antes agradecer el apoyo de mi familia. Ha habido meses que casi no nos hemos podido ver. Sé que siempre están apoyándome y que confían mucho en mí y en lo que hago. ¡Que la alegría os acompañe!

También tengo que agradecer a Simba su compañía por las noches, aunque tu cama sea más cómoda más de una vez te quedabas a mi lado.

El tercer pilar es el más importante para mí. Olga, gracias por aguantar tantos días y noches sin mí, gracias por venir de Carboneras, gracias por cuidarme como nadie haría, por entenderme en los malos momentos y pensar siempre en mí, por pensar en nuestro futuro, por hablarme con la mirada... Te quiero muchísimo.

Alicante, Julio de 2007  
Gabriel Jesús García Gómez



<b>Bloque I. Periodo de Docencia (2005 – 2006)</b>	<b>1</b>
<b>Curso 1. Visión por Computador: Color</b>	<b>3</b>
C1.1. Descripción.....	3
C1.2. Objetivos .....	3
C1.3. Métodos de evaluación.....	4
C1.4. Bibliografía.....	4
<b>Curso 2. Protocolos para transmisión on-line de información y calidad de servicio</b>	
C2.1. Descripción.....	5
C2.2. Objetivos .....	6
C2.3. Métodos de evaluación.....	6
C2.4. Bibliografía.....	7
<b>Curso 3. Sistemas de Tiempo Real</b>	
C3.1. Descripción.....	9
C3.2. Objetivos .....	10
C3.3. Métodos de evaluación.....	10
C3.4. Bibliografía.....	10
<b>Curso 4. Programación y Control de Robots</b>	
C4.1. Descripción.....	11
C4.2. Objetivos .....	12
C4.3. Métodos de evaluación.....	12
C4.4. Bibliografía.....	12
<b>Curso 5. Control Sensorial de Robots</b>	
C5.1. Descripción.....	13
C5.2. Objetivos .....	14
C5.3. Métodos de evaluación.....	14
C5.4. Bibliografía.....	14
<b>Curso 6. Manipulación Inteligente y Aplicaciones de la Robótica</b>	
C6.1. Descripción.....	17
C6.2. Objetivos .....	18
C6.3. Métodos de evaluación.....	18

C6.4. Bibliografía.....	18
<b>Curso 7. Metodologías en Investigación Científica y Tecnológica</b>	
C7.1. Descripción.....	19
C7.2. Objetivos.....	20
C7.3. Métodos de evaluación.....	20
C7.4. Bibliografía.....	20
<b>Bloque II. Periodo de Investigación (2006 – 2007)</b>	21
<b>Capítulo 1. Introducción</b>	23
1.1. Motivación.....	24
1.2. Marco de la Memoria.....	26
1.3. Aportaciones.....	28
1.4. Estructura de la Memoria.....	29
<b>Capítulo 2. Control Visual</b>	31
2.1. Revisión histórica.....	32
2.1.1. Introducción.....	32
2.1.2. Arquitectura del sistema de visión.....	34
2.1.2.1. Cámara externa al robot.....	35
2.1.2.2. Cámara en el extremo del robot.....	36
2.1.3. Clasificación de los sistemas de control visual.....	37
2.1.3.1. Ver y mover estático.....	37
2.1.3.2. Ver y mover dinámico.....	38
2.1.3.3. Control visual directo.....	39
2.1.3.4. Control basado en posición y basado en imagen.....	40
2.1.3.5. Sistemas finales de bucle abierto y bucle cerrado.....	43
2.1.4. Estado del arte en control visual indirecto basado en imagen.....	44
2.1.4.1. Características extraídas en control visual.....	46
2.2. Control basado en imagen.....	48
2.2.1. Jacobiano de la imagen.....	50
2.2.2. Controlador utilizando el Jacobiano de la imagen.....	57
2.3. Caso práctico.....	63
2.3.1. Planteamiento del problema.....	63
2.3.2. Arquitectura del sistema.....	64
2.3.3. Sistema de visión.....	65
2.3.4. Control visual. Formulación.....	66

2.3.5. Resultados .....	67
2.3.5. Conclusiones .....	75
2.4. Conclusiones .....	75
<b>Capítulo 3. Control Visual Virtual</b> .....	<b>77</b>
3.1. Introducción .....	78
3.2. Formulación .....	78
3.3. Resultados en la Calibración de una Cámara Usando Control Visual Virtual .....	80
3.3.1. Prueba 1 .....	82
3.3.2. Prueba 2 .....	84
3.4. Aplicación del Control Visual Virtual para Posicionamiento Preciso de la Cámara en una Tarea de Ensamblado .....	86
3.5. Conclusiones .....	89
<b>Capítulo 4. Seguimiento de trayectorias en el espacio imagen</b> .....	<b>91</b>
4.1. Introducción .....	92
4.2. Generación de la trayectoria en el espacio imagen .....	95
4.3. Control visual basado en flujo de movimiento .....	97
4.4. Seguimiento Intemporal de Trayectorias en el Espacio Imagen Basado en la Norma de la Matriz de Homografía 2D .....	102
4.5. Seguimiento Intemporal de Trayectorias Basado en Control Visual Virtual .....	111
4.6. Implementación .....	117
4.6.1. Sistema de simulación .....	117
4.6.2. Arquitectura del sistema .....	118
4.7. Resultados .....	119
4.7.1. Resultados de simulación .....	119
4.7.1.1. Trayectoria con las características iniciales en la trayectoria deseada .....	119
4.7.1.2. Trayectoria con las características iniciales fuera de la trayectoria deseada .....	123
4.7.2. Resultados experimentales .....	129
4.7.2.1. Trayectoria 1: trayectoria en espacio imagen muy oscilante .....	129
4.7.2.2. Comparativa frente a sistemas de seguimiento con control visual dependientes del tiempo .....	132
4.7.3. Comparativa con el control visual basado en flujo de movimiento .....	136
4.7.4. Estudio del comportamiento frente a variación de los parámetros del sistema .....	141
4.7.4.1. Estudio de la respuesta del sistema ante variaciones en la velocidad .....	141

4.7.4.2. Estudio de la respuesta del sistema ante variaciones en el muestreo de la trayectoria deseada .....	144
4.8. Conclusiones .....	148
<b>Capítulo 5. Seguimiento de trayectorias con control visual adaptativo</b> .....	<b>151</b>
5.1. Introducción .....	152
5.2. Combinando control visual y control visual virtual .....	153
5.3. Calibración multi-imagen .....	154
5.4. Seguimiento intemporal de trayectorias en la imagen basado en control visual adaptativo .....	155
5.5. Resultados .....	156
5.5.1. Prueba 1 .....	157
5.5.2. Prueba 2 .....	159
5.5.3. Prueba 3 .....	161
5.6. Conclusiones .....	162
<b>Capítulo 6. Seguimiento de trayectorias con control visual adaptativo</b> .....	<b>165</b>
6.1. Introducción .....	166
6.1.1. Control visual-fuerza híbrido .....	167
6.1.2. Control de impedancia .....	171
6.1.3. Otras aproximaciones .....	173
6.2. Fusión de control visual-fuerza basado en control de impedancia .....	174
6.2.1. Esquema de control de impedancia con realimentación visual .....	175
6.2.2. Modificación de la trayectoria en la imagen a partir de las fuerzas de interacción .....	177
6.2.3. Detección de un cambio en la superficie .....	181
6.2.3.1. Detección de discontinuidad de superficies a partir de las fuerzas de interacción .....	181
6.2.3.2. Luz estructurada para la detección de zonas de discontinuidades en el espacio imagen .....	184
6.2.3.3. Luz estructurada y fuerzas de interacción para la detección conjunta de cambios en las superficies .....	185
6.2.4. Resultados .....	187
6.2.4.1. Arquitectura del sistema .....	187
6.2.4.2. Detección de cambios en la superficie .....	188
6.2.4.3. Mejora de la detección de cambios en la superficie usando el sistema multisensorial .....	189

6.2.4.4. Tarea de inserción que requiere la fusión de visión y fuerza .....	191
6.3. Fusión de control visual-fuerza basado en la matriz de interacción Fuerza-Imagen .....	194
6.3.1. Control Visual-Fuerza mediante la matriz de interacción fuerza- imagen.....	195
6.3.2. Estimación de la matriz de interacción fuerza-imagen.....	196
6.3.3. Resultados .....	199
6.3.3.1. Resultados de simulación.....	199
6.3.3.2. Resultados experimentales.....	200
6.3.3.2.1. Experimento 1 .....	201
6.3.3.2.2. Experimento 2 .....	202
6.3. Conclusiones .....	202
<b>Capítulo 7. Conclusiones</b> .....	205
7.1. Conclusiones .....	206
7.2. Publicaciones .....	207
7.2.1. Revistas .....	207
7.2.2. Congresos internacionales.....	208
7.2.3. Congresos nacionales .....	209
7.3. Trabajos futuros .....	210
<b>Anexo 1. Cuaternios</b> .....	213
A1.1. Definición.....	214
A1.2. Transformaciones básicas .....	214
A1.3. Interpolación de la Orientación.....	215
<b>Anexo 2. Método de Optimización de Levenberg - Marquardt</b> .....	219
A2.1. Definición.....	220
A2.2. Transformaciones básicas .....	220
<b>Referencias</b> .....	223



Bloque I

Periodo de Docencia (2005 – 2006)



# CURSO 1

## Visión por Computador: Color

---

**Profesor:** Francisco Ortiz Zamora

**Código de la Asignatura:** 63059

**Tipo:** Fundamental

**Créditos:** 3

**Calificación Obtenida:** Sobresaliente (9)

---

### C1.1. Descripción

La asignatura versa sobre el estudio de los espacios de color y sus formas de representación más adecuadas para el procesamiento digital de imágenes. Se estudiarán el procesamiento vectorial de imágenes.

Los principales temas impartidos en este curso fueron:

- Espacios de color fundamentales.
- Espacios de color intuitivos.
- Procesamiento vectorial de imágenes digitales.
- Aplicaciones del procesamiento de color.

### C1.2. Objetivos

- Dar a conocer los fundamentos del procesamiento vectorial.
- Dar a conocer los espacios de color existentes en la actualidad.
- Dar a conocer las aplicaciones y métodos de procesamiento de imágenes digitales en color.

### **C1.3. Método de Evaluación**

El sistema de evaluación de esta asignatura constó de la presentación en clase de un artículo. El artículo propuesto “Improving Robot Vision by Color Information”, de Andreas Koschan, presentado en el 7th International Conference on Artificial Intelligence and Information-Control Systems of Robots en 1997. En él se han presentado varias aproximaciones al emparejamiento estéreo con color. Los resultados del emparejamiento por bloques se pueden mejorar considerablemente usando información de color y empleando una aproximación jerárquica. Además, la paralelización del algoritmo de emparejamiento por bloques es más intuitiva en el modelo jerárquico. El emparejamiento en estéreo basado en características se mejora considerablemente con información de color. Finalmente, se enumeran distintas técnicas para clasificar los bordes según su origen físico que mejoraría considerablemente el emparejamiento en estéreo basado en características.

### **C1.4. Bibliografía**

- Revistas científicas especializadas:
  - IEEE T. On image processing.
  - Journal of Image Processing.
  - Real Time Imaging.
  - Journal of Color Research.
- Aplicaciones del procesamiento de color.

## CURSO 2

# Protocolos para transmisión on-line de información y calidad de servicio

---

**Profesores:** Francisco Candela Herías  
**Código de la Asignatura:** 63060  
**Tipo:** Fundamental  
**Créditos:** 3  
**Calificación Obtenida:** Sobresaliente (10)

---

### C2.1. Descripción

Existen muchas aplicaciones de transmisión de datos, desde las que trabajan con tráfico multimedia hasta las dedicadas a controlar o monitorizar de forma remota y on-line sistemas industriales, que no funcionan correctamente si se emplean los mismos protocolos y técnicas que utilizan las aplicaciones clásicas de transmisión de archivos, páginas Web o e-mails. Esas aplicaciones, que intercambian información sensible a los retardos, requieren protocolos especiales, o consideraciones especiales cuando se usan los protocolos clásicos. Además, para conseguir el rendimiento adecuado de estas aplicaciones se debe tener en cuenta el control de la calidad de servicio que ofrece la red datos para los distintos tipos de tráfico que por ella circulan. Por ejemplo, se puede reservar parte del ancho de banda de un acceso a Internet para determinada aplicación on-line, limitando por otra parte el tráfico de acceso a Web o e-mails. En las clases de la asignatura se introducirá toda esta problemática al alumno, y se presentaran los principales protocolos y técnicas que la solucionan. También se propondrá al alumno realizar un trabajo práctico de programación de un sencillo protocolo con control de tiempo.

Los principales temas impartidos en este curso fueron:

- Conceptos básicos sobre protocolos y transmisión de datos. Información asíncrona e información sensible al retardo.
- Protocolos típicos TCP/IP para transmisión de información asíncrona.
- Los protocolos típicos y la transmisión de información sensible al retardo.
- Protocolos específicos para transmitir información sensible al retardo.
- Control de la calidad de servicio en redes de datos.
- Descripción de las herramientas software para el ejercicio práctico.
- Ejemplos de configuración de encaminadores.

## **C2.2. Objetivos**

- Presentar la necesidad de transmisión de datos de forma on-line de ciertas aplicaciones de automatización.
- Conocer las características de la transmisión de información sensible al retardo en contraste con los de la transmisión clásica de datos asíncronos.
- Conocer soluciones que aumenten el rendimiento de las aplicaciones que requieren una transmisión on-line de información.

## **C2.3. Método de Evaluación**

El sistema de evaluación de esta asignatura constó de la presentación de un informe sobre un artículo relacionado con la temática del curso. El informe elegido fue “Data Communications for Internet Robots”, de Liu, P.X., Meng, M.Q.-H. y Yang, S.X. publicado en la revista Autonomous Robots 15, en 2003. El artículo trata de los protocolos de envío de datos a través de Internet. Primero desarrolla los problemas que TCP y UDP tienen para la transmisión de datos de tiempo real a través de Internet. En el artículo se presenta un nuevo protocolo de transporte llamado trinomial protocol para robots operados a través de Internet. Se han considerado las siguientes métricas de rendimiento para la evaluación de los protocolos para operar robots bajo Internet: tasa de transmisión; justicia entre protocolos y dentro del mismo protocolo; y eficiencia. La principal idea del trinomial protocol es que eligiendo una apropiada combinación de  $\alpha$ ,  $\beta$  y  $\gamma$ , la respuesta transitoria del protocolo a variaciones del ancho de banda de la red es rápida manteniendo la tasa de transmisión poco oscilante en el régimen permanente. El protocolo se ha probado tanto en simulación (con la herramienta de simulación de redes ns-2), como con un robot Pioneer 2 PeopleBot situado en un laboratorio de la Universidad de Alberta en Canadá.

## C2.4. Bibliografía

- W. Stallings. Redes e Internet de Alta velocidad: Rendimiento y Calidad de Servicio. 2a edición. Ed. Pearson-Prentice Hall. 2003. (<http://williamstallings.com/HsNet2e.html>)
- W. Stallings. Comunicaciones y Redes de Computadores. 7a Edición. Ed. Pearson-Prentice Hall. 2004. (<http://williamstallings.com/DCC/DCC7e.html>)



## CURSO 3

### Sistemas de Tiempo Real

---

**Profesores:** Francisco Candela Herías  
**Código de la Asignatura:** 63061  
**Tipo:** Fundamental  
**Créditos:** 3  
**Calificación Obtenida:** Sobresaliente (10)

---

#### C3.1. Descripción

Por motivos de rendimiento puede ser necesario recurrir a sistemas multiprocesador o multi-computador para realizar determinados sistemas industriales. Por ejemplo, si se piensa en un sistema de inspección industrial, que requiere mantener una determinada cadencia de análisis de los productos que salen de un proceso de transformación. Pero, no basta con usar un hardware multiprocesador, también se requiere organizar las operaciones o tareas a ejecutar para alojarlas en los procesadores más adecuados, y para ordenarlas temporalmente, con el objetivo de aprovechar la capacidad de multiprocesamiento todo lo posible. Además, esta organización, denominada planificación de las tareas, debe tener en cuenta las restricciones del sistema y de las propias tareas a ejecutar. En las clases de esta asignatura se introdujo toda esta problemática, se estudiaron los distintos tipos de planificación, y se presentaron ejemplos de aplicación, especialmente con sistemas de visión artificial.

Los principales temas impartidos en este curso fueron:

- Introducción a los Sistemas en Tiempo Real.
- Clasificación de un STR.
- Estructura de un STR.

- Diseño de un STR.
- Fiabilidad y tolerancia a fallos.
- Particionamiento software.
- Ejemplos de distintos algoritmos de particionamiento.
- *Allocation*, asignación espacial de tareas a procesadores.
- *Scheduling*, planificación temporal de tareas.
- Tipos de algoritmos uni-procesador y multi-procesador.

### **C3.2. Objetivos**

- Comprender la necesidad de sistemas multiprocesador y de técnicas de planificación.
- Conocer los conceptos de asignación espacial y planificación temporal de tareas.
- Distinguir los principales tipos de planificación y algoritmos de planificación, y sus ámbitos de aplicación.
- Conocer el concepto de particionamiento software y algún algoritmo de particionamiento.
- Ver ejemplos de aplicación, como sistemas de inspección y de visión por computador.

### **C3.3. Método de Evaluación**

El sistema de evaluación de esta asignatura constó del desarrollo de una práctica realizada con el software llamado Grafos y Algoritmos desarrollado en el grupo AUROVA, el cual sirve para aplicar distintos métodos de scheduling y allocation dado un grafo de representación de las tareas de un sistema. Entonces, dado un grafo de representación de tareas, se debió analizar y clasificar el algoritmo de tiempo real propuesto, de forma que se pudieran comparar los distintos métodos de asignación temporal y espacial vistos en clase.

### **C3.4. Bibliografía**

- Nimal Nissanke. "Realtime Systems", Ed. Prentice Hall, 1997.  
C. M. Krishna, Kang G. Shin. "Real-Time Systems". Ed. MCGraw-Hill, 1997.

## CURSO 4

### Programación y Control de Robots

---

**Profesores:** Jorge Pomares Baeza y Santiago T. Puente Méndez

**Código de la Asignatura:** 63062

**Tipo:** Fundamental

**Créditos:** 3

**Calificación Obtenida:** Sobresaliente (10)

---

#### C4.1. Descripción

En esta asignatura se plantea el control de robots, siendo también un aspecto de gran relevancia a tener en consideración a la hora de definir una aplicación. Es por ello que en esta asignatura se plantea el estudio del control cinemático y dinámico de un robot. Juntamente con el control de los robots se plantean los métodos de programación. La programación de robots constituye un aspecto fundamental de la robótica, existiendo en el mercado multitud de lenguajes de programación de robots, tanto orientados a la tarea como orientados al robot. Además existen distintos tipos de programación según cuando se realice esta, pudiendo distinguir entre programación on-line y programación off-line. En esta asignatura se van a ver los distintos tipos de programación de robots así como la utilidad de cada una de ellas según la circunstancia en que se esté trabajando.

Los principales temas impartidos en este curso fueron:

- Introducción a la programación y control de robots.
- Control cinemático de robots.
- Control dinámico de robots.

- Tipos de programación de robots.
- Lenguajes de programación de robots.

#### **C4.2. Objetivos**

- Conocer los métodos de control cinemático de robots.
- Conocer los métodos más relevantes de control dinámico de robots.
- Conocer los distintos tipos de programación de robots existentes.
- Conocer algún lenguaje de programación de robots.
- Realizar un trabajo de programación y control de robots.

#### **C4.3. Método de Evaluación**

La asignatura se evaluaba con la realización de un programa para uno de los dos robots situados en el laboratorio del grupo AUROVA de la Universidad de Alicante. En mi caso, se eligió el Mitsubishi PA-10, que es un robot de 7 grados de libertad. El lenguaje de programación era C++ sobre la plataforma Visual C 6.0. Se programó una aplicación de control visual de forma que se le enviaban al robot las velocidades en el extremo para realizar el seguimiento de un objeto a partir de la información obtenida de la cámara que tenía situada en su extremo. Además, se realizó una demo para el otro robot, un Scorbot IX de 5 grados de libertad. Esta demo realizaba unos movimientos predefinidos que el robot repetía situado en una exposición en el MUA. El lenguaje de programación para este robot es el ACL.

#### **C4.4. Bibliografía**

- Robots y Sistemas Sensoriales. F. Torres et al. 2002. Pearson Educación.
- Control de Movimiento de Robots Manipuladores. R. Nelly & V. Santibáñez. 2003. Pearson Educación.
- Introduction to Robotics. Mechanics and Control. J. Craig. 1995. Addison Wesley.

## CURSO 5

### Control Sensorial de Robots

---

**Profesor:** Jorge Pomares Baeza  
**Código de la Asignatura:** 63063  
**Tipo:** Fundamental  
**Créditos:** 3  
**Calificación Obtenida:** Sobresaliente (10)

---

#### C5.1. Descripción

En general se puede afirmar que los sistemas de control sensorial aumentan la flexibilidad y autonomía de los sistemas robóticos. En aplicaciones de la robótica como las de ensamblado o soldadura, a menudo se requieren altas prestaciones y la aplicación de sistemas de guiado sensorial debe incorporarse para hacer a estos sistemas más robustos. En este curso se pretende dotar al alumno del conocimiento y las habilidades requeridas para el diseño de los sistemas sensoriales más adecuados para la tarea a realizar, profundizando en las estrategias de control. Se empezará estudiando los sistemas de control sensorial en general para pasar posteriormente a ver en detalle dos casos específicos: control visual y control de fuerza. En muchas ocasiones para obtener un sistema de control más robusto y fiable es necesario dotarlo de distintos sistemas sensoriales que trabajen de forma cooperativa. Por esta razón el curso finaliza con el estudio de sistemas de control multisensorial y sistemas de fusión sensorial para su aplicación a la robótica.

Los principales temas impartidos en este curso fueron:

- Introducción al control sensorial.

- Control de robots.
- Control visual.
- Control de fuerza.
- Control multisensorial.
- Fusión sensorial.

## C5.2. Objetivos

- Describir las principales arquitecturas de control sensorial existentes.
- Dar a conocer en detalle los sistemas de control visual para su aplicación a la robótica. Mostrar distintas técnicas de control de fuerza para su aplicación a tareas de interacción del robot con el entorno.
- Describir las principales técnicas de fusión sensorial.
- Mostrar las últimas líneas de investigación existentes dentro del campo del control sensorial, prestando una especial atención a los sistemas de control visual y de control de fuerza.

## C5.3. Método de Evaluación

El sistema de evaluación de esta asignatura constó de la lectura e investigación sobre un artículo dado en clase. El artículo elegido fue un capítulo del libro “Robot Force Control” de Bruno Siciliano y Luigi Villani, publicado por Kluwer Academic Publishers en 1999. Posteriormente, se realizaba una presentación en clase acerca del artículo propuesto. El Capítulo 4 del libro trata sobre el control de fuerza indirecto, el directo y las ventajas e inconvenientes que tienen uno respecto al otro. Posteriormente se entra a formular la regulación de fuerza usando control de fuerza indirecto y directo, para terminar explicando cómo realizar un control de la posición y una regulación de la fuerza.

## C5.4. Bibliografía

- Corke, P. I. “Visual control of robots” Research studies press ltd. 1996.
- Ghosh, K. B., Xi, N., Tarn, T. J. “Control in robotics and automation. Sensor-Based integration”. Academic Press. 1999.
- Hutchinson, S.; Hager, G. D.; Corke, P.I. “A tutorial on visual servo control”. IEEE Transactions on Robotics and Automation, Vol. 12, No. 5. Pp. 651-670, 1996.

- 
- Torres, F., Pomares, J., Gil, P., Puente, S., Aracil, R. “Robots y Sistemas Sensoriales” Prentice Hall, 2002.
  - Gorinevsky, D. M., Formalsky, A. M., Schneider, A. Y. “Force control of robotic systems”. CRC Press. 1997.



## CURSO 6

# Manipulación Inteligente y Aplicaciones de la Robótica

---

**Profesor:** Santiago T. Puente Méndez  
**Código de la Asignatura:** 63065  
**Tipo:** Fundamental  
**Créditos:** 3  
**Calificación Obtenida:** Sobresaliente (10)

---

### C6.1. Descripción

En la robótica actual los sistemas de manipulación cobran cada vez una mayor importancia, siendo de vital importancia para conseguir una que una aplicación alcance los objetivos deseados. Cada vez se requieren mayor información a la hora de realizar la manipulación de un elemento, es por ello que en esta asignatura se van a estudiar los sistemas que se pueden utilizar para conseguir una manipulación óptima para la aplicación deseada.

Los principales temas impartidos en este curso fueron:

- Introducción y presentación.
- Métodos de manipulación inteligente.
- Manipulación inteligente mediante sensores de contacto.
- Aplicaciones actuales de la robótica.

## C6.2. Objetivos

- Estudiar los métodos de manipulación inteligente.
- Estudiar los métodos para una manipulación inteligente mediante contacto.
- Describir aplicaciones de la robótica.

## C6.3. Método de Evaluación

La evaluación de la asignatura constó del estudio de un artículo relacionado con la temática del curso. El artículo elegido fue “Integration of Vision, Force and Tactile Sensing for Grasping”, de Peter K. Allen et al., publicado en Int. Journal of Intelligent Mechatronics, Vol. 4, No. 1 de Enero 1999. El agarre de objetos con manos robóticas sigue siendo una tarea muy difícil con muchos problemas de investigación abiertos. Para conseguir agarres correctos se necesita un gran número de variables de entrada y realimentación que nos debe dar la herramienta de agarre y la sensorización externa. Normalmente se necesita:

- Estimación del punto de contacto.
- Medición de la normal y la curvatura de superficies.
- Conocimiento de las fuerzas aplicadas e inducidas en los dedos de la mano.

Tanta información sensorial merma las capacidades mecánicas de la mano. Se describe el diseño y aplicación de una serie de sistemas sensoriales adicionales que aumentan la capacidad de una mano robótica. Estos sensores pueden usarse de manera complementaria para integrar información de visión, fuerza y táctil en tareas que requieren un control en bucle cerrado y tiempo real. En los experimentos se trabaja con una mano de Barrett, que es una mano robótica de 3 dedos con 4 grados de libertad. La mano de Barrett dispone de un sistema interno de sensorización de fuerza en cada dedo. Se añaden dos sistemas de sensorización diferentes al interno: un sistema de visión en tiempo real; y un sistema de sensores táctiles en los dedos y en la palma de la mano.

## C6.4. Bibliografía

- Robots y Sistemas Sensoriales, F. Torres et al. 2002. Pearson Educación.
- Revistas especializadas:
  - IEEE Sensors Journal.
  - Robótica.
  - Advanced Robotics.

## CURSO 7

### Metodologías en Investigación Científica y Tecnológica

---

**Profesor:** Fernando Torres Medina

**Código de la Asignatura:** 63074

**Tipo:** Metodológicas

**Créditos:** 3

**Calificación Obtenida:** Sobresaliente (10)

---

#### C7.1. Descripción

La asignatura Metodologías en Investigación Científica y Tecnológica se plantea como asignatura de tipo fundamental para todos aquellos alumnos de tercer ciclo del programa de doctorado que deseen adquirir los conocimientos y fundamentos que le permitan iniciarse rápidamente en las formas de acceder a bases de datos científicas, formas de seleccionar fuentes, como escribir documentos científicos, etc.

Los principales temas impartidos en este curso fueron:

- Introducción a las metodologías en investigación científica y tecnológica.
- Formas de acceder a bases de datos.
- Formas de seleccionar bases de datos.
- Formas de presentar documentos escritos de tipo científico.
- Formas de presentar oralmente trabajos científicos.

## **C7.2. Objetivos**

- Dar a conocer las metodologías de investigación científica.
- Dar a conocer las metodologías de investigación tecnológica.
- Dar a conocer los sistemas de información y búsqueda bibliográfica más extendidos.
- Dar a conocer las formas de escribir y presentar textos de tipo científico.

## **C7.3. Método de Evaluación**

La evaluación de esta asignatura constó de dos partes. La primera parte consistió en la realización de un estado del arte sobre la temática más afín a nuestras investigaciones. En mi caso realice el estado de arte sobre fusión de control visual y control de fuerza. Este estado del arte se ha adjuntado en la memoria, en el Capítulo 6, donde se describe una propuesta nueva de fusión de visión e imagen para controlar un robot que interacciona con el entorno. La segunda parte de la evaluación consistió en describir al resto de compañeros cómo se había realizado la búsqueda de artículos para elaborar este estado del arte. Las principales herramientas para la elaboración del estado del arte fueron las páginas de Scopus, ISI e Inspec.

## **C7.4. Bibliografía**

Revistas científicas y tecnológicas relativas al ámbito de la Automática y la Robótica.

Bloque II  
Periodo de Investigación (2006 – 2007)



# Capítulo 1

## Introducción

## 1.1. Motivación

---

Los sistemas robotizados para la manipulación automática se han ido implantando con mayor frecuencia en los últimos años en la industria en general. Estos sistemas van desde el más simple de empaquetado o agarre de objetos en cintas transportadoras hasta los más complejos en la industria del automóvil. En muchas ocasiones, no se conoce con exactitud las características del objeto a manipular o estos presentan una gran variedad, por lo que en estos casos es necesaria la aplicación de sistemas de control sensorial para el guiado y manipulación en un entorno robotizado. En general, se puede afirmar que los sistemas de control sensorial aumentan la flexibilidad y autonomía de los sistemas robóticos. En aplicaciones de la robótica como las de desensamblado o reciclado, a menudo se requieren altas prestaciones y la aplicación de sistemas de guiado sensorial debe incorporarse para hacer a estos sistemas más robustos.

Para el guiado de robots uno de los sistemas sensoriales con más auge en la actualidad son los basados en visión por computador. La visión artificial constituye uno de los temas de investigación que posee en la actualidad un espectro más amplio de posibles aplicaciones industriales, y que en un futuro adquirirá todavía una mayor relevancia. Muestra de ello son tanto los esfuerzos que dedican al tema los principales centros de investigación del mundo entero como el interés que demanda la industria en estas aplicaciones. Sus características especiales les hace capaces de capturar una gran cantidad de información de su entorno y facilitar el guiado del robot sin la necesidad de disponer de un conocimiento exhaustivo previo del espacio de trabajo. Así surgieron los sistemas de control visual [Hutchinson-1996] para el guiado de robots empleando información visual. Sin embargo, la aplicación práctica de estos sistemas de control en entornos no estructurados implican una serie de problemas que aún en la actualidad siguen sin estar resueltos. Uno de estos problemas es la necesidad de disponer de un sistema calibrado de la cámara empleada, de forma que si se modifica alguna propiedad de la cámara, es necesaria una nueva calibración para determinar de nuevo sus parámetros intrínsecos. A pesar de que en la actualidad es posible encontrar sistemas de control visual para el guiado de robots independientemente a cambios en los parámetros intrínsecos [Malis-2004a], en tareas de manipulación en ocasiones es necesario conocerlos a pesar de posibles cambios en los mismos durante la tarea. Estos aspectos requieren disponer en cierta medida de un entorno estructurado en el que tanto las propias características del objeto a manipular como los parámetros intrínsecos de la cámara utilizada deben ser conocidos a priori.

Uno de los objetivos de la investigación ha sido el de profundizar en el estudio de los sistemas de control visual para proporcionarles la suficiente flexibilidad y robustez en aplicaciones de manipulación en las que no se dispone de un conocimiento previo ni de la cámara utilizada ni del

objeto a ser manipulado. Cuando el sistema robotizado controlado mediante visión se enfrenta al desensamblado de una pieza cualquiera que se desea reciclar, se encuentra con el problema de saber qué material es el que se desea separar. La elección y reconocimiento de este material no es sencilla dado el amplio abanico de texturas y formas que puede tener. La variación de los parámetros de la cámara permiten obtener una mayor precisión o velocidad en estas tareas de detección del material a extraer y de sus características.

En tareas en las cuales el robot debe interactuar con su entorno (eliminación de soldaduras, desatornillado, etc.) los sistemas de visión artificial no pueden proporcionar información acerca de la fuerza de interacción entre el extremo del robot y su entorno. El tener acceso a información procedente de un sensor de fuerza constituye un elemento indispensable para aplicaciones en las cuales los robots deben interactuar directamente con objetos del entorno. La propia naturaleza y la precisión limitada de los sistemas de visión artificial utilizados para el control de posición implican que este tipo de sistemas no son adecuados para controlar la interacción.

Se pretende abordar el diseño de un sistema para el guiado de robots en tareas de manipulación, es por esto por lo que es necesario controlar no sólo la posición del robot con respecto al espacio de trabajo, sino también su fuerza de interacción. Esta investigación se desarrolla dentro de un área de trabajo de creciente interés y que requiere la utilización conjunta de los sistemas de visión artificial y el control de la interacción del robot con su entorno.

Las principales aproximaciones empleadas hasta la actualidad para combinar información visual y de fuerza son control híbrido y control de impedancia. En el caso de control híbrido, el espacio de control se divide en direcciones controladas por fuerza y direcciones controladas por visión siendo requisito para aplicar esta estrategia el disponer de un conocimiento previo detallado de la geometría de la tarea y el entorno en el que se va a desarrollar. En el caso de emplear control de impedancia se especifica la relación dinámica entre fuerza y movimiento. En cualquier caso, el espacio de trabajo se supone conocido a priori, lo que constituye una limitación de cara a su aplicación a entornos de reciclado.

A partir de estos trabajos se observa la necesidad de crear un nuevo sistema de control visual-fuerza que sea capaz, no solo de controlar una misma dirección de movimiento, sino hacerlo de manera que se eviten posibles contradicciones en las acciones de control obtenidas a partir de ambos sistemas sensoriales.

En aquellas tareas en las que se debe realizar la manipulación de un determinado objeto del espacio de trabajo se hace necesario el uso de forma combinada de los controladores visuales junto con controladores de fuerza. En este caso los sistemas de visión tienen la función de establecer la trayectoria del robot durante la tarea, actuando como controladores de posición, mientras el sistema

sensor de fuerza se encarga de controlar la interacción del robot con el entorno. Durante estos meses se ha investigado dentro de este ámbito, dada su importancia dentro de los objetivos prioritarios del proyecto. Así, las principales investigaciones se enmarcan dentro de la necesidad de dotar la mayor robustez y flexibilidad a estos sistemas, pudiendo desarrollar la tarea ante incertidumbres en los parámetros de la cámara y sin la necesidad de realizar una calibración cinemática previa.

A grandes rasgos otro aspecto fundamental que ha motivado la presente memoria de investigación es el comportamiento limitado de los sistemas de control visual empleados actualmente para el seguimiento de trayectorias. Para poner un ejemplo, en una tarea de ensamblado de una barandilla de aluminio, haciendo uso de una estrategia de control visual clásico para guiar el robot hasta el punto de inserción, éste sistema permitirá asegurar que el punto alcanzado será el deseado, pero no se podrá especificar la trayectoria que habrá de seguir el robot hasta alcanzar este punto. En esta memoria de investigación se plantea, por lo tanto, un nuevo sistema de seguimiento de trayectorias con la particularidad de no verse sometido a restricciones temporales así como la de presentar un comportamiento totalmente ajustable en precisión y velocidad.

## **1.2. Marco de la Memoria**

---

Esta memoria de investigación se ha desarrollado dentro del ámbito de la beca FPI de la Generalitat Valenciana (BFPI06/015) y de dos proyectos de investigación.

El primer proyecto de investigación es: “Diseño, Implementación y Experimentación de Escenarios de Manipulación Inteligentes para Aplicaciones de Ensamblado y Desensamblado Automático” financiado por el Ministerio de Ciencia y Tecnología (DPI2005-06222), cuyo principal objetivo es el de diseñar, implementar y experimentar un escenario inteligente, adecuadamente sensorizado, que permita abordar con garantías tareas complejas de manipulación tales como ensamblado y desensamblado, a realizar bien de forma totalmente automática por un sistema multirobot, o semi-automáticamente con intervención humana.

Tanto el ensamblado como el desensamblado automáticos requieren un alto grado de flexibilidad y adaptabilidad a los distintos cambios que puedan surgir durante el propio proceso debido a la incertidumbre existente con respecto al producto a desensamblar, al estado de conservación del mismo, al gran número de fabricantes existentes y su gama de productos, etc. Todo ello confiere un grado de dificultad superior, siendo necesarios sistemas de guiado de robots que empleen sistemas multisensoriales [Torres-2004].

Uno de los pilares de este proyecto, y donde se enmarca la presente memoria, es la investigación de sistemas multisensoriales. Los sistemas multisensoriales juegan un papel importante

en diferentes ámbitos y aplicaciones industriales, representando una posibilidad interesante para la satisfacción de crecientes requerimientos en el ámbito de los sistemas automatizados de producción. En concreto, los datos que suministran diferentes fuentes y sensores juegan un papel importante en la estimación de la posición de objetos en el espacio para su aplicación en tareas de desensamblado y reciclaje. En muchos casos, la información que se puede extraer de cámaras es insuficiente en muchos sentidos y surge la necesidad de incorporar otros elementos de medida, como sensores de fuerza, que permitan que la localización espacial de objetos para su desensamblado y separación de materiales se haga con la mayor precisión posible. Es por esta razón, por la que fundamentalmente no pueden emplearse las técnicas de control visual existentes hasta la actualidad que hacen uso únicamente de visión artificial, sino que es necesario extender estas estrategias a la utilización de sensores de fuerza que permitan controlar simultáneamente la trayectoria y la interacción del robot con el objeto que está siendo manipulado.

La fusión sensorial tiene como objetivo extraer información y datos procedentes de diferentes fuentes y sensores para obtener una información estructurada que permita modelar el entorno de la célula de desensamblado, y con ello, sea posible planificar los movimientos de los robots y realizar las tareas más adecuadas durante un proceso de desensamblado y reciclaje. Además, las restricciones y limitaciones que conlleva cada tipo de sensor en particular hacen pensar que no siempre es factible conseguir buenas estimaciones de posición, orientación, trayectoria, etc. empleando sensores de un único tipo. En gran cantidad de aplicaciones, en las que se puede incluir el desensamblado, el uso combinado de diferentes tipos de sensores permiten obtener más información y más precisa. De hecho, dentro del proyecto en el que se encuadra la memoria de investigación, los principales sensores empleados durante el proceso de desensamblado son los sensores visual y de fuerza, razón por la cual las estrategias aportadas en esta memoria de investigación son de fusión control visual-fuerza.

El otro proyecto en el que se enmarcan las investigaciones desarrolladas es el proyecto de investigación “Diseño y experimentación de estrategias de control visual-fuerza para sistemas flexibles de manipulación” financiado por la Generalitat Valenciana. En este proyecto se desarrolla la estrategia de control visual más adecuada para realizar el guiado del robot con las características visuales más adecuadas y determinando de forma automática y sin conocimiento previo los parámetros intrínsecos del sistema de visión. Por otro lado, se estudian los sistemas de control de fuerza a aplicar, así como su combinación con los anteriores para su actuación de forma conjunta. Uno de los objetivos del proyecto es el desarrollo de un nuevo algoritmo de fusión control visual-fuerza que permita al robot adaptarse a las características geométricas del objeto manipulado, no requiriendo una planificación previa de las direcciones a ser controladas por cada sensor.

## 1.3. Aportaciones

---

Una de las principales aportaciones de esta memoria de investigación ha sido el desarrollo de una nueva técnica de control visual, denominada seguimiento intemporal de trayectorias en la imagen basado en control visual virtual. Esta nueva estrategia ha sido ideada fundamentalmente para evitar las limitaciones de los sistemas clásicos basados en imagen en cuanto a la trayectoria descrita por el robot durante la tarea. En estos últimos sistemas únicamente se especifica la localización deseada de las características en la imagen, sin embargo, no se conoce la trayectoria 3-D que seguirá el robot hasta alcanzar estas características. Empleando un sistema de seguimiento intemporal de trayectorias en la imagen basado en control visual virtual no sólo se podrá especificar esas características finales, sino que se podrá definir qué trayectoria tanto en la imagen como en el espacio 3-D seguirá el robot hasta alcanzar dichas características.

Sin embargo, la formulación realizada del sistema de seguimiento de trayectorias permite solventar otros problemas derivados de los sistemas de control visual existentes hasta la actualidad, y que constituyen aportaciones de esta memoria de investigación. Así, cabe mencionar que el sistema presenta un comportamiento independiente del tiempo lo que permitirá realizar el seguimiento sin la necesidad de cumplir determinadas restricciones temporales.

Otro aspecto importante y que diferencia el método propuesto de los existentes hasta la actualidad, es la posibilidad de ajustar tanto la velocidad como la precisión del seguimiento. Así, dependiendo de las características de la aplicación a desarrollar podrán modificarse ciertos parámetros del sistema, obteniendo un comportamiento más rápido o preciso según se requiera. En resumen, es un método que consigue ajustar la velocidad de seguimiento con menos oscilaciones y mayor precisión que los métodos propuestos hasta la fecha para el seguimiento intemporal de trayectorias en la imagen.

Los sistemas de control visual virtual permiten obtener los parámetros extrínsecos e intrínsecos de una cámara real realizando variaciones de estos parámetros en una cámara virtual, de forma que al converger el sistema, se tengan en la cámara virtual los parámetros reales. Estos sistemas dependen mucho de la inicialización de los parámetros extrínsecos para la cámara virtual. Una aportación de esta memoria de suficiencia es la calibración multi-imagen, que permitirá obtener de forma precisa la calibración de la cámara independientemente de la inicialización de los parámetros extrínsecos al mismo tiempo que se realiza una tarea de control visual. El conocimiento de los parámetros intrínsecos reales a lo largo de la trayectoria seguida usando el método de seguimiento de trayectorias en la imagen basado en control visual virtual permite mejorar su comportamiento en imagen y en 3D, así como realizar el seguimiento en condiciones cambiantes de los parámetros intrínsecos de la cámara, como por ejemplo la realización de zoom.

La última fase de esta memoria de investigación se centra en los sistemas de control visual-fuerza. Para ello se ha extendido la estrategia de seguimiento intemporal de trayectorias en la imagen basado en control visual virtual para incluir la información de interacción con el entorno. El correcto comportamiento del sistema en el espacio 3-D se va a convertir en uno de los aspectos fundamentales del éxito de la estrategia, ya que no sólo va a permitir llegar a una posición dada sino que dicha posición podrá ser alcanzada cumpliendo las restricciones impuestas por los objetos manipulados.

Dentro de este ámbito también destacan ciertas aportaciones dentro de las cuales cabe citar el desarrollo de una nueva estrategia global control visual-fuerza aplicada a tareas de manipulación. El objetivo fundamental de esta estrategia es el posibilitar hacer uso de los sensores visual y de fuerza en entornos no estructurados y empleando control compartido. Se ha desarrollado un nuevo sistema de control visual-fuerza aplicado a control compartido, es decir, una misma dirección de movimiento puede ser controlada simultáneamente empleando control visual y fuerza. Se ha descrito que, debido a la diferencia entre las propiedades obtenidas a partir de cada uno de los sensores, puede ocurrir que las acciones de control obtenidas en un momento determinado sean contradictorias. Se ha prestado una especial atención a este aspecto, garantizando la coherencia entre la información obtenida a partir de cada uno de los sensores y constituyendo una aportación de los algoritmos planteados.

Así, se ha obtenido un sistema capaz de controlar la interacción del robot con el entorno mientras realiza el seguimiento de una trayectoria en la imagen. Cuando se detecta una colisión con el entorno de trabajo, se recalcula la trayectoria deseada en la imagen de forma que se cumplan las restricciones introducidas por las fuerzas de interacción. Por último, cuando el sistema interactúa con una nueva superficie, el sistema debe recalcular la trayectoria deseada en la imagen. Por ello, se propone un sistema que permite detectar de manera robusta los cambios en las superficies a partir de la información del sensor de fuerzas y de patrones de luz estructurada.

Por último, se propone un nuevo sistema de fusión de la información visual y de fuerza que no requiere conocimiento a priori del modelo cinemático del robot, la calibración de la cámara o de calibración sensorial entre la cámara y el sensor de fuerza. Este método se basa en estimar la matriz de interacción fuerza-imagen, que consigue relacionar variaciones en la imagen con variaciones en las fuerzas, permitiendo obtener una relación directa entre ambos sistemas sensoriales.

## 1.4. Estructura de la Memoria

A continuación se detalla la estructura de la memoria de investigación. Tras este capítulo de introducción, en el 2 se detallan conceptos fundamentales relativos a los sistemas de control visual, prestando mayor atención a aquellos en los que se centrarán las aportaciones realizadas a lo largo de la

memoria. Este primer capítulo no sólo es puramente descriptivo de los sistemas de control visual, sino que además se detallan las distintas tendencias en la investigación en esta materia, mostrando las carencias que se pretenden resolver con la realización de esta memoria de investigación.

Los capítulos 3, 4, 5 y 6, parte central de esta memoria de suficiencia, describen las aportaciones realizadas. Se ha procurado un nivel alto de detalle que permita la fácil comprensión de los algoritmos propuestos, manteniendo el formalismo necesario, así como mostrar distintos experimentos teóricos y prácticos que avalen las aportaciones realizadas.

El bloque central de la memoria de investigación empieza con la descripción del control visual virtual. Así, en el capítulo 3 se describe la calibración de una cámara a partir de una imagen tomada por ésta. La calibración de la posición de la cámara respecto a un objeto se consigue de forma sencilla a partir de la aplicación de control visual virtual. Sin embargo, la obtención precisa de los parámetros intrínsecos de la cámara no se consigue cuando se usa una sola imagen. En el capítulo 5 se describirá la técnica de la calibración multi-imagen, que permite obtener los parámetros extrínsecos e intrínsecos de una cámara mientras realiza una tarea de control visual.

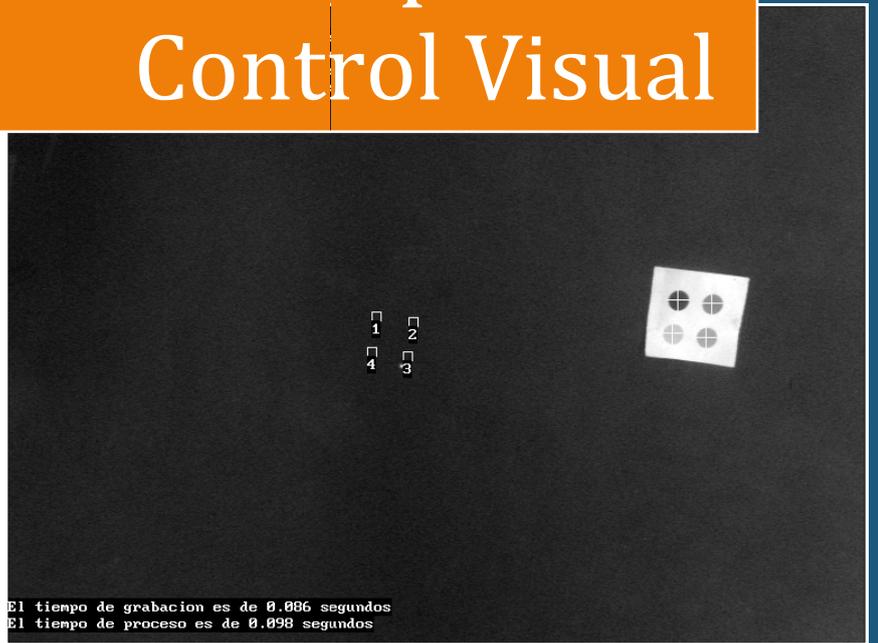
En el capítulo 4 se describe un nuevo método de guiado de robots por visión, denominado seguimiento intemporal de trayectorias en la imagen basado en control visual virtual, que resuelve las carencias de los sistemas de control visual desarrollados hasta la actualidad para el seguimiento de trayectorias. No solo consigue obtener un comportamiento intemporal, sino que además hace uso de la calibración multi-imagen que se describirá en el capítulo 5 para mejorar el comportamiento en la imagen y realizar el seguimiento a pesar de variaciones en los intrínsecos en tiempo real. En este capítulo, tras realizar una descripción detallada del método, se muestran distintos resultados experimentales con los que se comprueba la correcta respuesta del mismo y se muestra su comportamiento frente a otros métodos de seguimiento temporal e intemporal de trayectorias en la imagen y ante la variación de los distintos parámetros que rigen su funcionamiento.

El bloque central de la memoria de suficiencia finaliza con el capítulo 6, en el que se describen las aportaciones realizadas dentro del campo de la fusión de los sistemas de control visual con los de fuerza. En este capítulo se incluyen los elementos necesarios a la estrategia de seguimiento empleando visión descrita en el capítulo 4, para permitir la utilización de control de fuerza y su aplicación a tareas de manipulación.

En el capítulo 7 se resumen las principales conclusiones que se han derivado de la realización de la memoria, describiendo las publicaciones generadas durante estos dos años de investigación y los posibles trabajos futuros relacionados con las investigaciones desarrolladas. Finalmente, la memoria concluye con la lista detallada de las referencias bibliográficas consultadas.

# Capítulo 2

## Control Visual



## 2.1. Revisión histórica

---

### 2.1.1. Introducción

Inicialmente, los sistemas robóticos que integraban visión por computador trabajaban en bucle abierto, lo que se conoce como “ver y mover” (en la literatura anglosajona se referencia como “look and move”). Es decir, en primer lugar ver y reconocer el entorno con ayuda de un sistema de visión por computador, y posteriormente realizar el movimiento en función de la información capturada en la etapa previa. Una de estas primeras aplicaciones de guiado de robots por visión fue la desarrollada por Shirai e Inoue [Shirai-1973], quienes describen como puede utilizarse la realimentación visual para corregir la posición de un robot y aumentar la precisión. Esta primera aplicación consistía en utilizar realimentación visual para el guiado de un robot de forma que realice el agarre de un prisma y su posterior inserción en una caja. Durante esta década se desarrollaron las primeras aplicaciones de control visual para el agarre de objetos en movimiento por cintas transportadoras.

Una alternativa al planteamiento anterior es emplear control en bucle cerrado mediante realimentación de la información visual para controlar la localización del efector final del robot con relación a un objeto, lo que también se conoce en la literatura anglosajona como visual servoing (término definido por Hill y Park [Hill-1979]). Por tanto, el control visual está basado en el uso de información visual en la realimentación del bucle de control. Más concretamente, un sistema de control visual utiliza la información captada de una escena por una o más cámaras conectadas a un sistema de visión por computador, para controlar la localización del efector final del robot con relación a la pieza de trabajo.

En 1980 Sanderson y Weiss [Sanderson-1980] establecen una clasificación de los sistemas de control visual, describiendo asimismo el primer sistema de control visual basado en imagen. Desde la aparición de estos primeros sistemas de control visual a principios de los 80, su evolución ha sido lenta, sin embargo en los últimos años han aparecido crecientemente nuevos desarrollos debidos principalmente a las capacidades de procesamiento de los nuevos ordenadores personales que permiten procesar una escena a una frecuencia cada vez mayor.

Ya a finales de los 80 comienza a aplicarse el control visual a muy distintos ámbitos [Corke-1994] como puede ser el agrícola, en el que se encuentran aplicaciones para la recogida de frutas empleando realimentación visual [Harrell-1989], o la teleoperación de robots. Dentro de este último ámbito el control visual permite al operador humano especificar la tarea en términos de características en la imagen [Papanikolopoulos-1992, Vikramaditya-1997].

Otro ámbito de interés en control visual es el de la vigilancia. En muchas ocasiones el sistema de visión se localiza sobre una estructura móvil de forma que es capaz de mantener en el campo de visión un determinado objeto que se encuentra bajo vigilancia.

Durante los 90 y con el aumento de las capacidades de los sistemas de visión artificial aparecen un gran número de aplicaciones. Dentro de este nuevo conjunto de aplicaciones cabe mencionar algunas como el guiado de vehículos [Dickmanns-1988] en las que se requiere una gran capacidad de procesamiento y una alta frecuencia en la realimentación visual. Otro ámbito en el que el control visual presenta una importancia creciente en la actualidad es el del guiado de robots aéreos. Aplicaciones dentro de este ámbito no han sido explotadas en gran medida, sin embargo, ya empiezan a surgir vehículos aéreos, como helicópteros [Chriette-2001], dotados de un sistema de visión que realimenta al controlador del vehículo de forma que permite realizar su guiado a partir de características del entorno, evitando obstáculos e incluso permitiendo realizar un aterrizaje automático. El desarrollo de sistemas de control visual en estos vehículos permite realizar aplicaciones como vigilancia, seguimiento, inspección, trabajo en lugares peligrosos, etc.

Con la miniaturización y el aumento de precisión de los componentes de los robots, van aumentando las posibilidades de la micromanipulación empleando realimentación visual procedente, por ejemplo, de microscopios. Dentro de este ámbito cabe destacar que cuando se observan microorganismos, los elementos desaparecen del campo de visión muy rápidamente. Se vienen aplicando en la actualidad sistemas de control visual con el objetivo de mantener en la imagen a lo largo del tiempo ciertos microorganismos con el propósito de su posterior manipulación [Oku-2000].

Esta mejora en el desempeño de los sistemas robóticos y de visión artificial ha propiciado la aparición de aplicaciones a robots submarinos [Lots-2001]. En este caso se dispone de vehículos submarinos guiados por visión o sistemas de manipulación submarina cuya tarea es guiada mediante realimentación visual.

Un ámbito de aplicación con mayor impacto social son las aplicaciones médicas [Wei-1997, Ginhoux-2005]. La alta precisión alcanzada con estos sistemas ha propiciado la aparición de sistemas para la realización de telecirugía así como robots que realizan tareas quirúrgicas de forma autónoma. A pesar de que estos sistemas aumentan el coste por intervención, la alta precisión junto con el pequeño espacio de trabajo requerido en, por ejemplo, cirugía cerebral, hace que estos sistemas se apliquen con éxito. Dentro de este ámbito y volviendo a las aplicaciones de guiado de vehículos son de destacar las investigaciones para aplicación de control visual para el guiado de sillas de ruedas para discapacitados.

En la actualidad, las mejoras de los tiempos de procesamiento de los ordenadores han permitido a las aplicaciones de robótica basadas en visión desarrollarse en tiempo real. Esto ha hecho

posible incluir las características extraídas de las imágenes en un bucle de control. De hecho, en la actualidad encontramos aplicaciones muy diversas en el ámbito del control visual como puede ser posicionamiento preciso de un robot, control remoto o realimentación para teleoperación, micromanipulación, agarre de objetos en movimiento, sistemas de posicionamiento de un avión sobre la pista de aterrizaje [Bourquardez-2007], cámaras de seguimiento de misiles, recolección de frutas, robots que juegan al ping-pong [Ángel-2004], etc.

### 2.1.2. Arquitectura del sistema de visión

Uno de los factores a los que se debe el auge actual en los sistemas de control visual es el desarrollo de nuevos sistemas de visión. Estos nuevos sistemas han propiciado que la información visual sea cada vez más utilizada, debido a la disminución de retardos de captura y procesamiento, lo que a su vez ha permitido el desarrollo de sistemas basados en distintas arquitecturas. En este apartado se describen las configuraciones más utilizadas para la ubicación de cámaras (una o varias) respecto al robot y al espacio de trabajo donde se desarrolla la tarea.

Supóngase una aplicación en la que un robot deba realizar el agarre de un objeto en movimiento. Los componentes más importantes en un sistema de control visual para una aplicación de seguimiento de este tipo son los mostrados en la Figura 2-1.

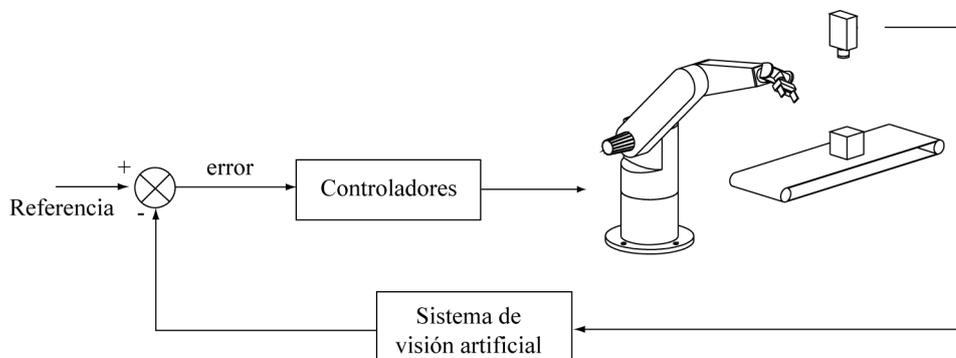


Figura 2-1. Componentes principales de un sistema de control visual.

En la Figura 2-1 se representa un esquema genérico en el que se observa el bloque denominado “sistema de visión artificial” que es el encargado de procesar las imágenes y realimentar constantemente al sistema con esta información. La entrada al sistema de control es una referencia, de manera que el error que anula el controlador es la diferencia entre el valor actual y el deseado. Este controlador se encarga de dirigir el manipulador hacia la pieza que se está siguiendo.

Es posible ubicar la cámara en diferentes posiciones. En muchas ocasiones se monta en el extremo del robot, en este caso observa el entorno en el que se realiza la tarea. Sin embargo, para la

realización de otras tareas puede ser necesario que la cámara observe tanto al robot como a su entorno. En este último caso se suele utilizar un sistema de cámara fija o montada en un segundo robot.

Independientemente del tipo de configuración empleada, es necesario realizar la calibración de la cámara para determinar sus parámetros intrínsecos como distancia focal, precisión de píxel y punto principal. Por lo tanto, se puede afirmar que los sistemas de servovisión usan típicamente dos configuraciones de cámara: montada en el efector final del robot (Figura 2-2.a), o externa al robot (Figura 2-2.b).

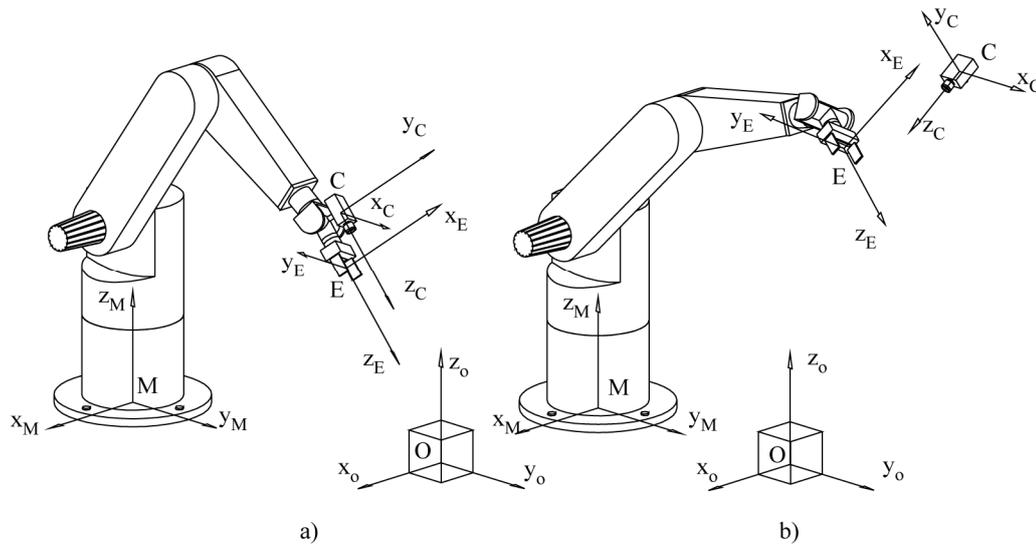


Figura 2-2. a) Configuración de cámara montada en el extremo del robot. b) Configuración de cámara externa.

### 2.1.2.1. Cámara externa al robot

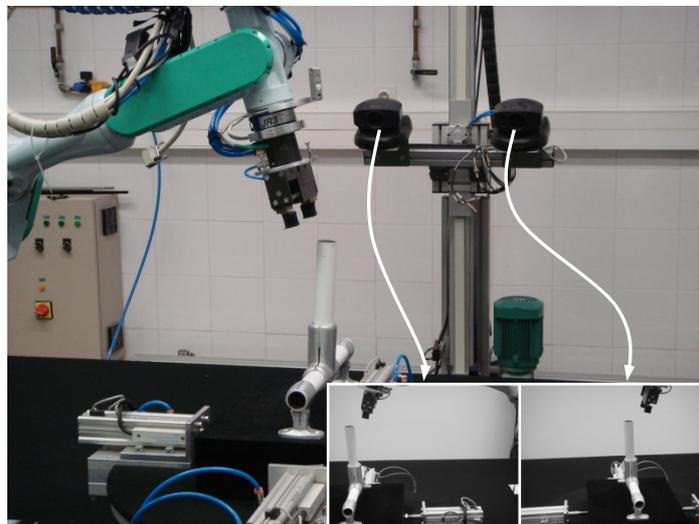
En este caso la cámara no tiene una relación mecánica con el robot que está siendo controlado visualmente, pero es conocida la relación entre la cámara y el sistema de coordenadas asociado a la base del robot. En principio, la vista obtenida es fija, salvo que la cámara esté ubicada sobre una estructura móvil. Una variante consiste en ubicar la cámara en el extremo de otro robot o bien emplear un cabezal con movimientos pan-tilt, con el propósito de conseguir visualizar el robot de la mejor forma. La cámara se posiciona de manera que el elemento terminal y los objetos situados en el espacio de trabajo del robot se encuentren en su campo de visión. Con esta configuración la información visual va a permitir obtener la localización tanto de un objeto situado en el espacio de trabajo como la del extremo del robot.

En la Figura 2-2.b se muestran los sistemas de coordenadas principales utilizados en un sistema de control visual en el que el sistema de visión está compuesto por una única cámara fija observando el espacio de trabajo y el extremo del robot. La tarea a realizar se va a indicar respecto al sistema de coordenadas del mundo o de referencia, M, situado en la base del robot. Se conoce la relación entre los sistemas de coordenadas de la Figura 2-2.b de manera que se pueda comparar la

localización del extremo del robot y la del objetivo a alcanzar utilizando un mismo sistema de coordenadas de referencia.

En el plano imagen de la cámara fija se obtiene una representación 2-D del espacio de trabajo a partir de la cual se extraen las características necesarias para el desarrollo de la tarea (posición de un objeto, su tamaño, centroide...). Estas características están expresadas respecto al sistema de la cámara,  $C$ , para referirlas respecto al sistema de coordenadas del mundo,  $M$ ; se necesitará disponer de la información acerca de la localización relativa entre la cámara y la base del robot,  ${}^M T_C$ , que permita realizar la transformación de coordenadas.

Por otro lado, en una aplicación de control visual en la que el robot deba interactuar con los objetos del espacio de trabajo, será necesario disponer de información acerca de la localización del extremo del robot referida al sistema de coordenadas de referencia,  $M$ . Esta información podrá extraerse igualmente a partir de la información visual procedente del sistema de visión, realizando la suposición de que el sistema de visión capta información tanto del espacio de trabajo como del extremo del robot. Con esto, ya se dispondrá de información relativa al robot y al espacio de trabajo referida a un único sistema de coordenadas, el del mundo,  $M$ . En ocasiones el número de cámaras puede ser mayor que uno para realizar una reconstrucción geométrica del entorno más fiable. En la Figura 2-3 se muestra un ejemplo de sistema de control visual con cámara externa al robot que utiliza un par estéreo.



*Figura 2-3. Arquitectura de cámara-robot donde la cámara externa es un par estereoscópico montado en un robot cartesiano.*

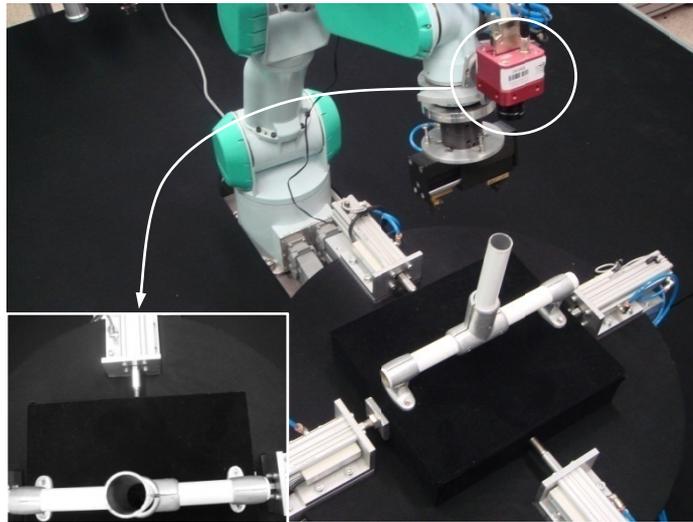
### 2.1.2.2. Cámara en el extremo del robot

En esta solución, denominada en la literatura anglosajona como “eye-in-hand”, la cámara se encuentra solidaria al extremo del robot de manera que los objetos del espacio de trabajo se encuentran

dentro de su campo visual. En este caso el sistema de visión se emplea para determinar la localización de un objeto con respecto al extremo del robot.

En la Figura 2-2.a se ha representado la ubicación de los distintos sistemas de coordenadas utilizados en un sistema de control visual con una cámara colocada en el extremo del robot. En este caso la relación entre el sistema de coordenadas del extremo del robot, E, y el de la cámara, C,  ${}^E T_C$  es conocida y fija durante el desarrollo de la tarea.

En la Figura 2-4 se muestra un ejemplo de sistema de control visual con configuración de cámara en el extremo del robot.



*Figura 2-4. Arquitectura de cámara-robot donde la cámara está situada en el extremo del robot.*

### **2.1.3. Clasificación de los sistemas de control visual**

En este apartado se va a mostrar una clasificación de los sistemas de control visual. En primer lugar, atendiendo al tipo de arquitectura y enfoque del problema de control visual (Apartados 2.1.3.1, 2.1.3.2 y 2.1.3.3), posteriormente teniendo en cuenta la utilización de las características extraídas de la imagen (Apartado 2.1.3.4). Por último, en el Apartado 2.1.3.5 se clasifican estos sistemas considerando el área observada por la cámara.

#### **2.1.3.1. Ver y mover estático**

Inicialmente, los sistemas robóticos que integraban visión por computador trabajaban en bucle abierto, en una configuración secuencial, primero ver y reconocer, y a continuación mover. Obviamente, la precisión de la operación a realizar depende directamente de la precisión del sensor de visión y del efector final del robot. Como se observa en la Figura 2-5, el robot ejecuta las tareas suponiendo que ningún objeto en el entorno se modifica desde que se realiza la captura de la imagen

hasta que se lleva a cabo la tarea. En esta arquitectura la única realimentación presente es la de los bucles internos de control de los servomotores.

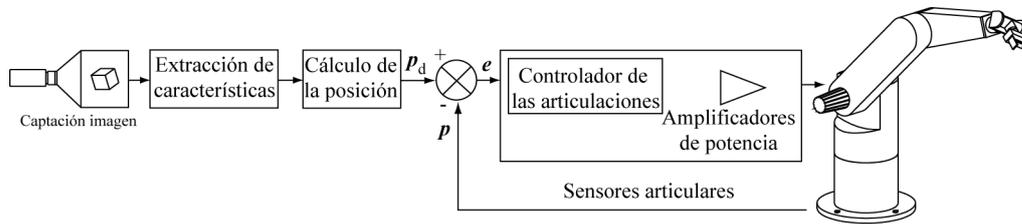


Figura 2-5. “Ver y mover” estático.

Supóngase una aplicación en la que un robot deba alcanzar la posición en la que se encuentra un objeto del espacio de trabajo. Para determinar la posición del objeto se utiliza un sistema de visión que indica en qué lugar se encuentra el objeto a alcanzar. Utilizando una estrategia de “ver y mover” estático, el robot se dirigiría a continuación hacia la posición indicada por el sistema de visión haciendo la suposición de que una vez alcanzada esa posición ya se habrá asimismo alcanzado el objeto. Por lo tanto, no se comprueba, ni durante la trayectoria del robot ni una vez alcanzada la posición final, que efectivamente se haya alcanzado al objeto, con lo que se supone que, desde que se obtuvo la posición con el sistema de visión hasta que se alcanza la posición final, no se ha alterado la posición del objeto.

### 2.1.3.2. Ver y mover dinámico

Sobre el sistema “ver y mover” anterior, que se podría decir que es estático, se puede añadir una realimentación visual obteniendo otro tipo de sistema de control visual denominado “ver y mover” dinámico. Como se observa en la Figura 2-6, en este tipo de arquitectura el sistema de visión es usado en un primer bucle de realimentación; existe un segundo lazo interno donde se realimenta la información articular (con una velocidad de muestreo mayor). Se realiza lo que se denomina un control visual indirecto, y frente al “ver y mover” estático, permite realimentarse visualmente mientras el robot se encuentra realizando la tarea.

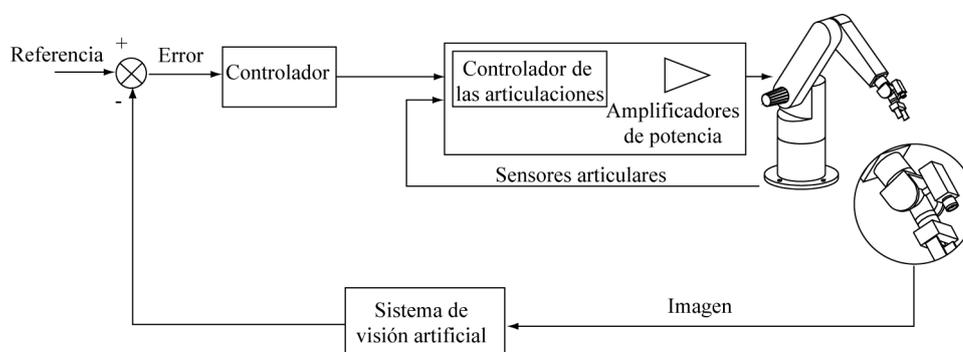


Figura 2-6. “Ver y mover” dinámico.

Utilizando el ejemplo presentado en el Apartado 2.1.3.1, si se empleara una estrategia de tipo “ver y mover” dinámico, el sistema se realimentaría constantemente con la información visual. Esta continua realimentación permitiría al sistema corregir posibles errores en la posición del objetivo determinada en instantes anteriores o modificar la trayectoria del robot ante posibles movimientos o alteraciones en los objetos del espacio de trabajo.

### 2.1.3.3. Control visual directo

Una alternativa al planteamiento anterior es emplear control en bucle cerrado mediante realimentación de la información visual para controlar la localización del efector final del robot con relación a un objeto, lo que también se conoce como control visual directo. En este caso no se requiere el bucle de control a nivel articular (Figura 2-7), presentando sólo el bucle externo de control basado en la información visual, en el cual se calculan las consignas para cada una de las articulaciones del robot.

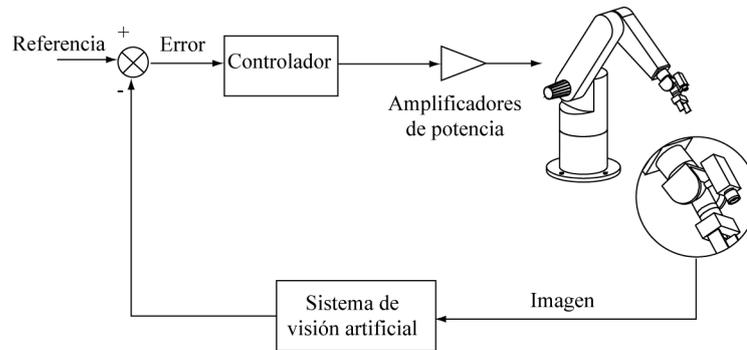


Figura 2-7. Control visual directo.

Retomando el ejemplo mostrado en los Apartados 2.1.3.1 y 2.1.3.2, si se utilizara un control visual directo, al igual que en el caso de “ver y mover” dinámico el sistema se realimentará constantemente con información visual del entorno; sin embargo, en este caso no estará presente el controlador del robot. Este último regulador será sustituido por el controlador visual que utiliza información procedente del sistema de visión directamente para regular y estabilizar al sistema.

Los primeros trabajos encaminados hacia el diseño de reguladores PI y PID en sistemas de control visual directo se deben a Weiss [Weiss-1987] o más recientemente los trabajos de Kelly [Kelly-2000]. Dentro de este tipo de control visual cabe mencionar los estudios de Corke [Corke-1996a] en los que se realiza el diseño de reguladores mediante la introducción de polos, o los de Hashimoto [Hashimoto-1996] dirigidos hacia el desarrollo de técnicas de control óptimo considerando el caso en el que se hace uso de características visuales redundantes, o los trabajos de Gangloff [Gangloff-1998] centrados en el desarrollo de un controlador predictivo que tenga en consideración la dinámica total del sistema robot y la cámara. Dentro de esta línea de investigación cabe mencionar los trabajos de Nelson, Papanikolopoulos y Khosla [Nelson-1993] que aplicaron sus desarrollos a tareas

de ensamblado. Cabe citar asimismo los trabajos descritos en [GarcíaN-2002] o los de Nasisi y Carelli [Nasisi-2003] que hacen uso de reguladores adaptativos para posicionamiento y seguimiento.

#### 2.1.3.4. Control basado en posición y basado en imagen

Se suele introducir una segunda clasificación [Sanderson-1980] de los sistemas de control visual que se realiza dependiendo si el control se ejecuta basado en posición o basado en características extraídas de las imágenes. En los sistemas de control basados en posición se dispone de un modelo del espacio de trabajo de forma que, a partir de las características extraídas del sistema de visión, se estima la localización del objeto a alcanzar respecto al sistema de coordenadas de la cámara. Esta localización estimada se compara a su vez con la deseada o de referencia; la diferencia entre ambas localizaciones será la entrada al regulador. A su vez, teniendo en cuenta la clasificación dada en los apartados anteriores, se pueden establecer las siguientes configuraciones para los sistemas de control visual basados en posición:

- “Ver y mover” estático basado en posición. Este esquema se corresponde con el mostrado en la Figura 2-5, en el que se observa que las características extraídas son traducidas a una localización,  $p_d$ , en la que se desea que se posicione el robot.
- “Ver y mover” dinámico basado en posición. Esta arquitectura se muestra en la Figura 2-8. En la realimentación del bucle de control se extraen las características observadas en la imagen a partir de las cuales se estima la localización del objetivo a alcanzar,  $p^C$  (posición),  $\varphi^C$  (orientación). La entrada de referencia al bucle de control es la localización deseada del robot,  $p_d^C$ ,  $\varphi_d^C$ . La diferencia entre ambas posiciones, referidas a un sistema común de referencia, es la entrada al regulador que se encargará de ir disminuyendo progresivamente esta distancia.

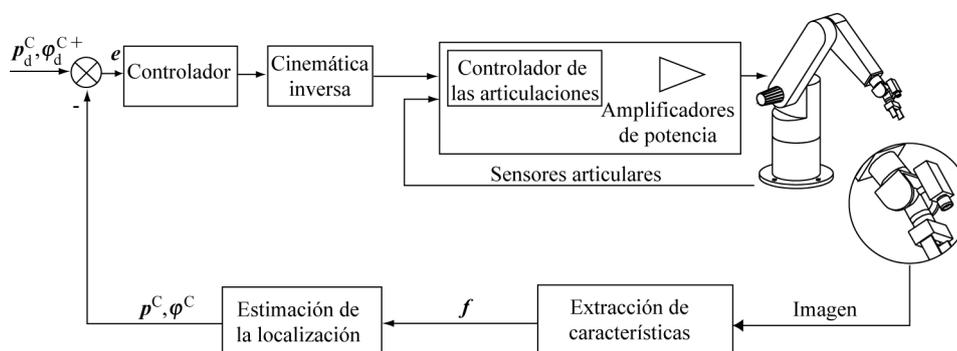


Figura 2-8. “Ver y mover” dinámico basado en posición.

Se observa, por lo tanto, la necesidad en este tipo de sistemas de estimar la localización. Dentro de los trabajos de reconstrucción a partir de primitivas de tipo punto cabe resaltar la contribución de DeMenthon [DeMenthon-1995] consistente en una técnica iterativa que

permite la estimación rápida de la posición de un objeto a partir de 4 puntos o más no coplanares, o a partir de 4 puntos coplanares [Oberkampff-1996]. Más recientemente se ha desarrollado una modificación de este algoritmo para permitir simultáneamente la correspondencia y determinación de la posición a partir de los puntos [David-2002]. Asimismo, es posible encontrar trabajos encaminados a estimar la localización utilizando primitivas como segmentos. Así, cabe mencionar los estudios de Horaud [Horaud-1997] consistente en una adaptación del algoritmo de DeMenthon para el caso de extracción de segmentos. Otros trabajos reseñables dentro de control visual basado en posición son los de Allen [Allen-1992] en los que se estima la posición de un objeto móvil a partir de la medida del flujo óptico con un par estereoscópico y haciendo uso de filtros  $\alpha, \beta, \gamma$  para predecir la posición a alcanzar. Wilson [Wilson-1996] hace uso de un filtro de Kalman para estimar la posición relativa entre la cámara y el objeto. En este último trabajo se desarrolla un regulador PD en el espacio articular. Sandini y Grosso [Grosso-1996] estiman el estado del efector final de un robot a partir de la medida del flujo óptico con dos cámaras fijas. Recientemente cabe resaltar los trabajos de Martinet [Martinet-1999, Martinet-2001, Cervera-2003] en sistemas estereoscópicos para su utilización en sistemas de control visual basados en posición, así como la inclusión de características 3-D en los sistemas de control visual basados en imagen. Por último, cabe asimismo mencionar los trabajos de Lippiello, Siciliano y Villani [Lippiello-2003] centrados en el desarrollo de sistemas de control visual basados en posición haciendo uso de sistemas multi-cámaras para el seguimiento de objetos en movimiento.

- *Control visual directo basado en posición.* Este esquema, mostrado en la Figura 2-9, utiliza al igual que en el caso anterior, un modelo del objetivo a alcanzar para determinar su localización a partir de las características observadas por el sistema de visión. La entrada de referencia también será la localización deseada del robot. Sin embargo, en este caso el regulador se encargará de realizar la estabilización del robot.

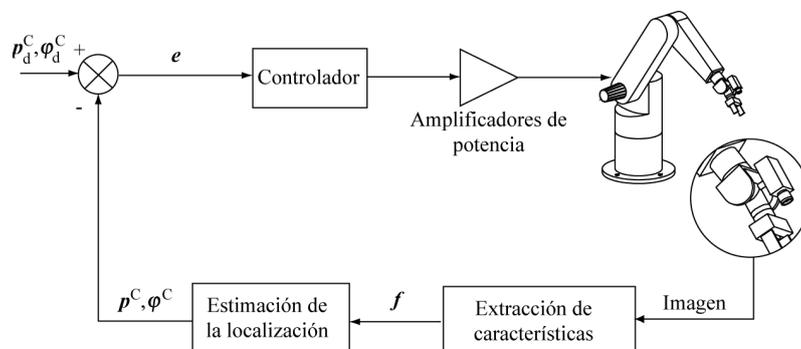


Figura 2-9. Control visual directo basado en posición.

Por el contrario, en los sistemas de control basados en imagen, el control se realiza directamente a partir de las características extraídas por el sistema de visión, que representan la proyección del objetivo a alcanzar en el plano imagen. De esta manera, la entrada al regulador será una comparación entre las características observadas y las deseadas o de referencia. De nuevo, atendiendo a la clasificación dada en los Apartados 2.1.3.2 y 2.1.3.3 se pueden distinguir las siguientes configuraciones de sistemas de control visual basados en imagen:

- *Ver y mover dinámico basado en imagen.* La arquitectura para un sistema de este tipo se muestra en la Figura 2-10.

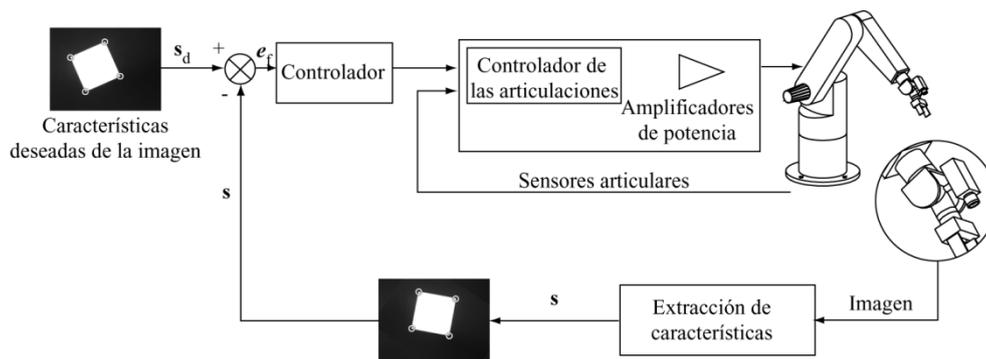


Figura 2-10. “Ver y mover” dinámico basado en imagen.

La entrada de referencia,  $s_d$ , se expresa en forma de características observadas por el sistema de visión (puntos, segmentos, círculos, esquinas...). En la realimentación del sistema se encuentra el sistema de visión que se encarga, durante el desarrollo de la tarea, de realizar la extracción de estas características,  $s$ . El regulador, comparando las características reales y deseadas, realiza las acciones de control necesarias para que se consiga alcanzar la posición en la que  $s = s_d$ . Una revisión histórica de este tipo de sistemas se muestra en el Apartado 2.1.4.

- *Control visual directo basado en imagen (Figura 2-11).* De nuevo se observa cómo la función de error de entrada al controlador,  $e_f$ , es seleccionada para conseguir que las características observadas sean igual a las deseadas. Al utilizar un control visual directo el regulador se encargará asimismo de realizar la estabilización del robot.

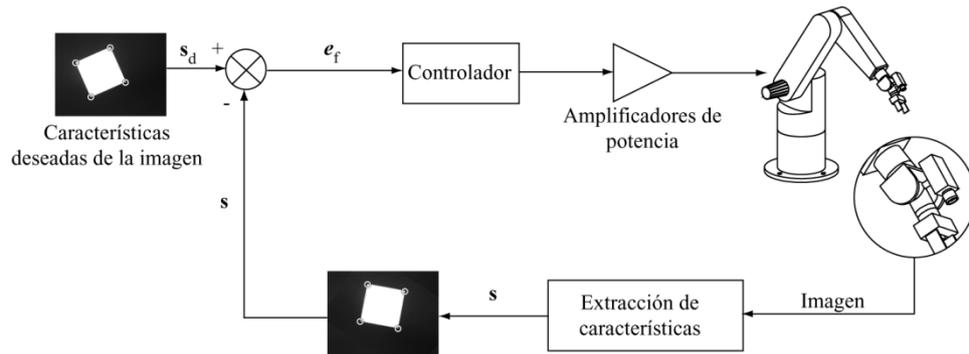


Figura 2-11. Control visual directo basado en imagen.

En el Apartado 2.2 se presentan más en detalle los sistemas de control basados en imagen. Se describirá la implementación utilizando una arquitectura de tipo “ver y mover” dinámico, indicando sus características y consideraciones necesarias para su desarrollo.

### 2.1.3.5. Sistemas finales de bucle abierto y bucle cerrado

Por otra parte, según el área final observada por la cámara se pueden distinguir los sistemas de control visual en:

- *Sistemas finales de bucle abierto o EOL (denominados en la literatura anglosajona como “end point open loop”).* En ellos la cámara observa únicamente el objeto destino y no se obtiene información visual de la posición actual del robot. De esta manera, se determina y controla la posición del extremo del robot exclusivamente con una combinación de conocimiento interno de la posición del extremo del robot a partir de su cinemática y de la calibración cámara-objeto. Bajo estas consideraciones, la precisión en el posicionamiento del robot depende de la precisión de la calibración y de la exactitud en la reconstrucción de la escena.
- *Sistemas finales de bucle cerrado o ECL (denominados en la literatura anglosajona como “end point closed loop”).* En ellos la cámara observa el objeto destino y el manipulador final del robot.

Un sistema EOL se puede convertir en un sistema ECL incluyendo la observación del efector final del manipulador. Aunque teóricamente parece preferible un sistema ECL a un EOL, la necesidad de seguir al efector final además del objeto destino, conlleva a que los problemas de visión tengan más peso en el diseño del controlador de un ECL, y a que los requerimientos de campo visual no puedan ser satisfechos siempre.

### 2.1.4. Estado del arte en control visual indirecto basado en imagen

En este apartado se muestran las principales líneas de investigación actual dentro del control visual basado en imagen. En lo siguiente, salvo que se especifique lo contrario, se supondrá control visual indirecto. Se muestra con mayor detalle el estado del arte dentro de este campo del control visual debido a que será aquí donde se enmarcan las principales aportaciones de la investigación realizada.

Una de las primeras líneas de investigación abiertas dentro de este campo fue la desarrollada por Chaumette en su tesis doctoral [Chaumette-1990] consistente en la aplicación de la función de la tarea [Samson-1991] al caso de la información captada por una cámara. A partir de entonces han surgido un gran número de autores interesados en distintos aspectos del control visual basado en imagen. A continuación se citarán las principales líneas de investigación surgidas desde entonces.

Muchos autores se han interesado en las distintas maneras de calcular el Jacobiano de la imagen [Chaumette-1990, Sebastián-2005], así como determinar el comportamiento del sistema ante simplificaciones o mejoras en su cálculo. Así, cabe destacar el cálculo del Jacobiano de la imagen de forma analítica como propone Chaumette en su tesis doctoral. Un problema que subyace ante la necesidad del cálculo de esta matriz es la necesidad de determinar en cada momento la profundidad o distancia cámara-objeto. Para determinar este valor se han desarrollado trabajos en los que se estima la profundidad a partir de conocimiento previo [Dornaika-1996], sin conocimiento previo [Jagersand-1997] o empleando redes neuronales [Suh-1993]. Dentro de este último campo se puede reseñar los estudios de Hosoda y Jagersand [Hosoda-1995, Jagersand-1997]. En la actualidad, la mayoría de las implementaciones emplean una simplificación de esta matriz consistente en determinar su valor en el equilibrio para evitar el cálculo de la profundidad.

Relacionados con los trabajos anteriores se encuentran los estudios realizados por distintos autores relativos a los problemas de singularidad, robustez y estabilidad en los sistemas de control visual basados en imagen. Chaumette en [Chaumette-1998] desarrolla un estudio acerca de la estabilidad y convergencia de estos sistemas indicando distintas situaciones que pueden provocar mínimos locales. En [Chaumette-2006], se hace una revisión de estos problemas de estabilidad. Para evitar estos problemas de singularidad y mínimos locales se han desarrollado trabajos como [Marchand-1996] en el que se propone la inclusión de tareas secundarias dentro de la propia función de tarea para garantizar la convergencia.

Algunos de los problemas observados en las investigaciones citadas anteriormente se pueden solucionar con un nuevo tipo de control visual, denominado control visual 2-D1/2, desarrollado por Malis [Malis-1999] que se encuentra a mitad de camino entre el control basado en posición y el basado en imagen. Este nuevo método permite evitar distintas desventajas aparecidas tanto en los

sistemas de control visual basados en imagen como la existente en los basados en posición. Básicamente consiste en el cálculo en cada iteración de la homografía entre las características actuales y deseadas. A partir de la homografía, se obtiene la rotación de la cámara, las características en la imagen y su profundidad relativa. Esta información es utilizada para generar una ley de control que permite asegurar la convergencia aunque la distancia al objeto sea grande (garantiza la convergencia aunque las características deseadas se encuentren lejanas a las actuales). Por otro lado, no requiere un conocimiento geométrico 3-D exhaustivo del entorno como en el caso del control visual basado en posición. Una importante línea de investigación abierta por Malis recientemente [Malis-2003] son los sistemas de control visual no calibrados e invariantes a errores en los parámetros intrínsecos de la cámara, presentando, por ejemplo, un correcto comportamiento ante errores en la determinación del valor de la focal de la cámara utilizada. El problema del uso de la homografía es el cálculo de la distancia entre la cámara y el plano de referencia. Para calcular esta constante, Chen propone el uso de técnicas adaptativas [Chen-2005].

Con el objetivo de evitar los problemas de convergencia citados anteriormente cabe reseñar asimismo los trabajos de Corke y Hutchinson [Corke-2001] relativos a control visual particionado. En estos trabajos se propone una estrategia de control que trata de forma separada distintos grados de libertad de la tarea para garantizar la convergencia ante distintas situaciones.

También es posible combinar diferentes características 2D y 3D. En [Cervera-2003] se utiliza esta técnica para obtener la matriz de interacción, multiplicando a las coordenadas homogéneas del conjunto de características en la imagen el valor de la distancia entre la cámara y cada característica. En [Schramm-2004] se demuestra que esta selección de características no converge hacia mínimos locales.

Otra línea de investigación dentro de los sistemas de control visual basados en imagen es su aplicación a la visión activa para imponer movimientos que permitan conservar ciertas propiedades con el entorno. Dentro de este campo se enmarcan los trabajos de Marchand [Marchand-1996] que se emplean para la reconstrucción de escenas estáticas. Cabe citar también los estudios encaminados a solucionar los problemas relacionados con la navegación por control visual, dentro de los cuales se encuentran trabajos como [Pavlidis-1996, Janabi-Sharifi-1997, Blanc-2005]. En algunas ocasiones se ha resuelto el problema de navegación mediante el encadenamiento de tareas de control visual [Marchand-1996, Martinet-1999].

Otro campo de trabajo actual es el del control visual robusto [Kragic-2002, Kragic-2003]. Son reseñables los trabajos desarrollados por la misma autora en cuanto a seguimiento de objetos, sistemas de manipulación basados en control visual o utilización de distintos tipos de características visuales y su posterior fusión.

En diversos trabajos como [Mezouar-2002b, Schramm-2006, Corke-2001, Chesi-2005] se muestra la necesidad de mantener en el campo de visión de la cámara las características del objeto implicado en el control visual. Para mantener durante la tarea estas características en la imagen se suelen emplear técnicas consistentes en la generación de un potencial en los límites de la imagen que genere una fuerza de repulsión sobre las características.

Un campo de estudio actual en los sistemas de control visual es el de planificación de trayectorias en el espacio imagen para su posterior seguimiento con control visual [Mezouar-2002a, Mezouar-2003, Schramm-2006]. Estos estudios introducen una ley de control dependiente del tiempo para el seguimiento de trayectorias generadas en el espacio imagen que permitan cumplir determinadas restricciones como la evitación de configuraciones singulares o mantener las características en la imagen durante la trayectoria. Schramm muestra una aproximación en la que se mantienen las características en el campo de visión de la cámara mientras se sigue una trayectoria sin necesidad de calibrar previamente la cámara.

Otro aspecto que caracteriza a los sistemas de control visual basados en imagen es que se puede especificar la posición final del robot respecto a un objeto del entorno, sin embargo, no es posible definir la trayectoria que describirá el robot desde la localización inicial a la final en el espacio 3-D. Sin embargo, en la mayoría de las aplicaciones es necesario, no solo alcanzar una determinada posición, sino también hacerlo de una manera dada que cumpla determinadas restricciones en el espacio 3-D. Se observa, por lo tanto, una importante carencia de los sistemas de control visual clásicos basados en imagen y que constituirá una aportación de esta investigación. Hasta ahora, los sistemas para el seguimiento de trayectorias que hacen uso de control visual se basan en una sucesión en el tiempo de características deseadas en la imagen u objetivos intermedios a alcanzar que, como se desarrollará en el Capítulo 4, no garantiza el correcto seguimiento de una trayectoria (ni en el espacio imagen ni en el espacio 3-D), presentando importantes problemas de estabilidad. Con el objetivo de solucionar estos problemas se ha desarrollado un nuevo método de seguimiento intemporal de trayectorias en la imagen basado en control visual virtual, que permite el seguimiento de trayectorias ante oclusiones o movimiento del objeto implicado en el control visual.

#### **2.1.4.1. Características extraídas en control visual**

De forma general se puede afirmar que una característica en la imagen es una estructura, compuesta de píxeles, que se encuentran en un contexto lógico y que puede ser medida y extraída de la imagen. A menudo una característica en la imagen se corresponde con una característica física del objeto, por ejemplo, una línea en la imagen se corresponde con una arista del objeto físico. La selección de buenas características es crucial para el desempeño de un sistema de control visual. Una característica utilizable para realizar control visual será aquella que puede ser localizada de manera no ambigua en diferentes vistas de la escena. Otros criterios que se deben considerar en la selección de

características son robustez, unicidad, controlabilidad y sensibilidad. Un parámetro de característica en la imagen se trata de un valor cuantificable que puede ser determinado a partir de una o más características en la imagen. La naturaleza de estos parámetros puede ser geométrica (p. ej. posición, superficie de un área, longitud, distancia, etc.) o fotométrica (p. ej. color, intensidad de píxel, etc.). Algunos ejemplos de parámetros de características utilizados en control visual son:

- *Principales características geométricas.* En [Chaumette-1990] se realiza un estudio detallado acerca de las distintas consideraciones para obtener una ley de control adecuada cuando las características extraídas en la imagen son puntos, líneas, segmentos, círculos, elipses, esferas, etc. Obviamente el sistema de visión artificial debe ser capaz de extraer los parámetros adecuados de la primitiva geométrica seleccionada para posteriormente aplicar el Jacobiano de la imagen adecuado. A lo largo de esta memoria se supondrá el conocimiento de las coordenadas de ciertos puntos en la imagen, por lo tanto, la primitiva seleccionada será el punto. Estas coordenadas pueden ser obtenidas previa detección de esquinas, o lo que es más común, determinar el centroide de los objetos presentes en la imagen. El centroide o centro de gravedad de un objeto es una característica muy utilizada y se define como el punto más representativo del objeto en la imagen, permitiendo tener una aproximación de la localización del objeto en la imagen. Normalmente se utiliza el espacio cartesiano para representar la posición de una característica en la imagen. Sin embargo, Iwatsuki [Iwatsuki-2005] propone el sistema de coordenadas cilíndrico para mejorar el comportamiento del control visual.
- *Patrones de intensidad.* Empleando una técnica semejante a la desarrollada para los métodos basados en ventana se puede detectar la localización de un determinado patrón de intensidad a lo largo de una secuencia de imágenes. Como ejemplo de este tipo de información empleada en control visual cabe mencionar trabajos como [Crétual-1998] que se utilizan principalmente para el seguimiento de objetos complejos o de los que no es posible extraer una característica determinada durante toda la tarea.
- *Snakes* [Abolmaesumi-2002, Hollinghurst-1993, Perrin-2003]. Se tratan de contornos activos que se mueven y deforman de acuerdo a varias “fuerzas”, y que tienden a adaptarse al objeto seguido. Dichas “fuerzas” están compuestas típicamente por fuerzas externas que son propiedades locales de la imagen, y fuerzas internas que son funciones de la propia forma del snake. Usualmente, un snake es atraído al máximo cuanto mayor es el gradiente en la imagen y así es utilizado para seguir contornos de un objeto en movimiento.
- *Contornos.* Cabe reseñar algún trabajo como [Collewet-2000] que hacen uso de la información relativa al contorno de un determinado objeto presente en la imagen para realizar el guiado de robots.

- *Momentos en la imagen.* Existen investigaciones actuales [GarcíaGJ-2005, Tahri-2005, Chaumette-2004, Chaumette-2002] relativas a la utilización de distintos momentos en la imagen.
- *Combinación de distintas características.* Cuando de una determinada escena puede extraerse más de una característica, existen trabajos como [Kragic-2001] encaminados a obtener una acción de control más robusta obteniendo una información visual más fiable que en el caso de la utilización de una única característica.
- *Información del modelo 3D del objeto.* La utilización de información procedente de un modelo del objeto a seguir permite al sistema de control visual ser más robusto. El sistema permitiría, por ejemplo, seguir con la tarea ante oclusiones del objeto [Marchand-2005]. El trabajo realizado por Collewet [Collewet-2004] permite reconstruir este modelo 3D sin información previa.
- *Esferas.* En [Tatsambon-2007] se propone el uso de esferas para realizar el control visual usando un modelo de proyección esférica.

A la hora de seleccionar o hacer uso de características es necesario considerar problemas como la posible oclusión de dichas características [Pomares-2004b]. La oclusión es un problema típico que surge en tiempo de ejecución y que puede ser parcialmente evitado utilizando características en la imagen redundantes. Este tipo de problemas también se pueden evitar utilizando sistemas inteligentes con la habilidad de almacenar la desaparición de una característica y la estimación de la localización actual [Pomares-2006c]. Otro problema que puede surgir en sistemas de control visual es la aparición de singularidades visuales. Una configuración de un robot encuentra una singularidad visual cuando un movimiento dado en el robot no produce cambios en las características de la imagen.

## 2.2. Control basado en imagen

---

Frente al control basado en posición, el basado en imagen es más apropiado cuando no se dispone de un modelo geométrico de la tarea a desarrollar. Para determinar la ley de control en esta aproximación se utilizan las características del objeto en el sensor visual sin calcular su localización real, con lo que será más robusta frente a errores de calibración.

Control visual basado en imagen es una manera de guiado de robots donde la ley de control es expresada en términos de características en las imágenes, es decir, se minimiza el error entre las características de la imagen medidas y las deseadas. De esta manera, un sistema de control visual basado en imagen implica la determinación de una función de error,  $e$ , que valdrá cero cuando la tarea se haya desarrollado correctamente y haya llegado a su fin. En este momento las características

observadas en la imagen,  $s$ , se corresponderán con las características deseadas  $s_d$ . En este caso la acción de control es calculada en el espacio 2-D de la imagen; por esta razón, a esta aproximación también se le denomina control visual 2-D (o su equivalente en inglés 2-D visual servoing).

En una aplicación típica de control visual en la que el robot debe posicionarse a una distancia respecto a un objeto del espacio de trabajo, el robot describirá una determinada trayectoria a lo largo de la cual las imágenes captadas por el sistema de visión se irán modificando progresivamente. Es el controlador el que se encargará de ejecutar las acciones oportunas sobre el robot de forma que las características observadas en las imágenes se vayan aproximando progresivamente a las deseadas, es decir, a la entrada de referencia del bucle de control.

En la Figura 2-10 y la Figura 2-11 se presentaron, respectivamente, los bucles “ver y mover” dinámico y control visual directo para los sistemas de control basados en imagen. Sin embargo, será la estrategia “ver y mover” dinámica la que se desarrolle en este apartado.

Las características de imagen usadas para el control visual suelen tratarse de formas geométricas elementales (como pueden ser puntos característicos o esquinas) y van a permitir reconocer en una imagen la proyección de un objeto. Imágenes típicas captadas por un sistema de control basado en imagen serían las que se muestran en la Figura 2-12.a y en la Figura 2-12.b.

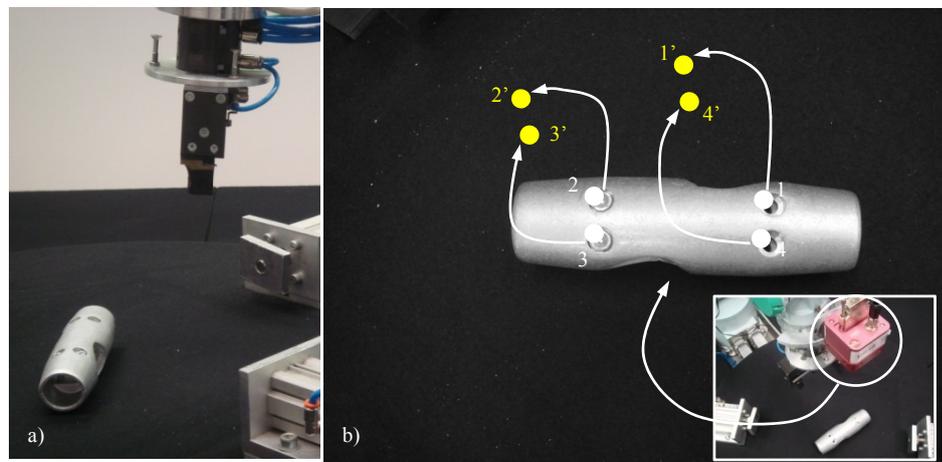


Figura 2-12. a) Imagen captada con cámara externa al robot. b) Imagen captada con cámara en el extremo del robot.

La imagen de la Figura 2-12.a es captada por un sistema de cámara externa fija. En esta figura se observa la presencia de la pinza del robot que se acerca al agarre de una pieza. La extracción de características sobre la Figura 2-12.a permitirá realimentar al sistema y la diferencia entre estas características y las deseadas será la función de error que el controlador deberá anular progresivamente.

Por otro lado, en la Figura 2-12.b se muestra una imagen obtenida con una cámara colocada en el extremo del robot. En esta imagen se han mostrado las posiciones de las características actuales

(1,2,3,4) y las deseadas (1',2',3',4'). La diferencia entre la posición de ambas características será el error,  $e_f$ , que deberá ir reduciendo progresivamente el sistema de control. Así, las acciones de control del regulador harán que el robot se mueva de forma que las características observadas en la imagen se vayan acercando progresivamente a las deseadas, de forma que una vez que la función de error se anule (se igualan las características deseadas y observadas) se supondrá que la tarea ha finalizado.

Los objetos son a menudo simplificados al máximo para realizar el control basado en imagen. Para la extracción de características se suelen utilizar técnicas como el marcado de puntos característicos en el objeto sobre un fondo uniforme. El tratamiento de la imagen se limita entonces a realizar una detección de estas características y el cálculo del centro de gravedad de éstos con el fin de determinar el centro de cada una de las características.

Como ya se ha comentado anteriormente, en los sistemas de control basados en imagen el bucle de control es realimentado con las características extraídas de las imágenes, sin necesidad de determinar la localización del objetivo a alcanzar. Esto va a suponer una ventaja frente a los basados en posición ya que, además de hacer al sistema menos sensible a errores de calibración, se van a reducir tiempos de procesamiento.

Para realizar este tipo de control, el sistema de cámaras utilizado puede ser de cámara externa al robot o un sistema de cámara situada en el extremo del mismo. A continuación se va a describir cómo desarrollar la regulación proporcional necesaria para la reducción del error,  $e$ , en un sistema de cámara fija externa al robot. Para ello se va a utilizar el concepto de Jacobiano de la imagen descrito por primera vez por Sanderson et al. [Sanderson-1987].

### 2.2.1. Jacobiano de la imagen

Para comprender la utilidad del Jacobiano de la imagen supóngase que se dispone de un sistema de cámara fija que observa el movimiento del extremo de un robot. A partir de las imágenes capturadas, el sistema de visión es capaz de extraer un punto característico,  $p_E^C$ , situado en el extremo del robot.

En un instante de tiempo, el punto  $p_E^C$  situado en el extremo del robot se moverá con una velocidad de rotación  $\omega_E^C(\alpha_E^C, \beta_E^C, \gamma_E^C)$ , así como también, una velocidad de traslación  $v_{TE}^C(\dot{x}_{TE}^C, \dot{y}_{TE}^C, \dot{z}_{TE}^C)$ , ambas con respecto al sistema de coordenadas de la cámara.

A su vez, el sistema de visión captará la evolución temporal de esta característica,  $f$ , en el plano de la imagen  $(\dot{f}_x, \dot{f}_y)$ .

El Jacobiano de la imagen,  $\mathbf{J}_f$ , es una expresión en forma de matriz que relaciona las velocidades de las características en el plano de la imagen con las velocidades (de traslación y rotación) relativas de los puntos correspondientes a las características con respecto al sistema de coordenadas de la cámara. De esta forma se puede conocer cómo cambian las características en el plano de la imagen cuando se produce un cambio en el extremo del robot en movimiento:

$$\begin{bmatrix} \dot{f}_x \\ \dot{f}_y \end{bmatrix} = \mathbf{J}_f(\mathbf{p}_E^C) \cdot \begin{bmatrix} \mathbf{v}_{tE}^C \\ \boldsymbol{\omega}_E^C \end{bmatrix} \quad (2.1)$$

En un caso general, considerando  $\mathbf{p}_E$  la posición del extremo del robot en un espacio de dimensión  $j$ ,  $[\mathbf{v}_{tE}, \boldsymbol{\omega}_E]^T$  será su velocidad (compuesta por velocidad de traslación y rotación). Se va a representar con  $\mathbf{s}$  un vector de  $i$  características observadas en la imagen, mientras que  $\dot{\mathbf{s}}$  será el flujo óptico o variación de estas características en la imagen. El Jacobiano de la imagen (también denominada como matriz de interacción o matriz de sensibilidad) se representará como  $\mathbf{J}_f$  y realizará la siguiente transformación:

$$\begin{bmatrix} \dot{f}_{1x} \\ \dot{f}_{1y} \\ \vdots \\ \dot{f}_{ix} \\ \dot{f}_{iy} \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1(\mathbf{p}_E)}{\partial p_1} & \dots & \frac{\partial f_1(\mathbf{p}_E)}{\partial p_j} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_i(\mathbf{p}_E)}{\partial p_1} & \dots & \frac{\partial f_i(\mathbf{p}_E)}{\partial p_j} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{v}_{tE} \\ \boldsymbol{\omega}_E \end{bmatrix} \quad (2.2)$$

Como se observa en la expresión (2.2), el número de columnas en el Jacobiano de la imagen variará dependiendo de la tarea (se ha considerado un espacio de dimensión  $j$ ). En general la matriz tendrá  $i$  filas y  $j$  columnas y no se puede asegurar que sea cuadrada.

A continuación se va a determinar de forma teórica el valor del Jacobiano. Para ello se parte del modelo de cámara pinhole. Sea un punto  $\mathbf{p}^M$  de la escena 3-D en coordenadas del mundo  $\mathbf{p}^M(x_p^M, y_p^M, z_p^M)$  y sea  $\mathbf{p}^S(x_p^S, y_p^S)$  su proyección en el plano imagen; la proyección  $\mathbf{p}^S$  en el plano imagen del punto  $\mathbf{p}^M$  se obtiene como intersección de la línea que une  $\mathbf{p}^M$  y el centro óptico de la cámara  $C$  con el plano imagen. La distancia entre el centro óptico  $C$  y el plano imagen se denomina distancia focal o focal  $f$ . Al punto que se obtiene como intersección del eje principal (línea perpendicular al plano imagen que pasa por el centro óptico  $C$ )  $z_C$ , y al plano imagen se le llama punto principal (Figura 2-13).

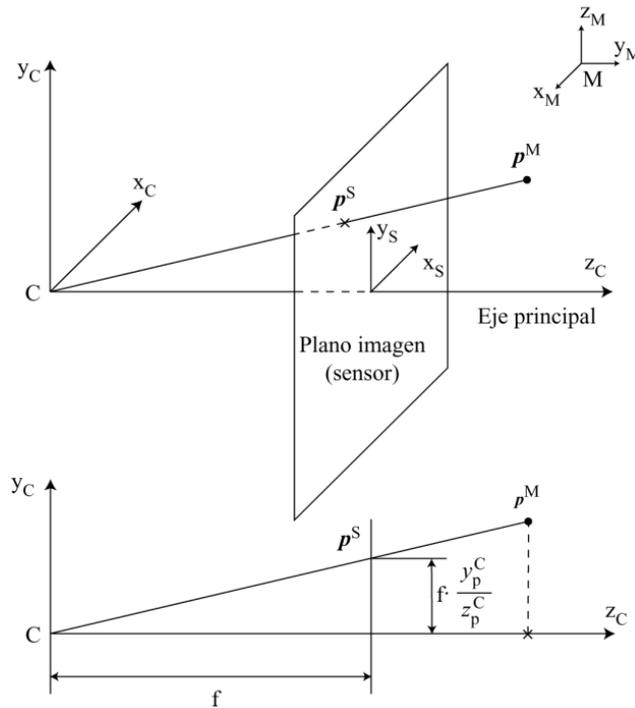


Figura 2-13. Principales sistemas de coordenadas en el modelo pinhole.

En el modelo de cámara pinhole, el sistema de lentes de la cámara es aproximado por un único punto. Con esta aproximación la imagen siempre se encuentra enfocada sin importar la distancia entre el objeto y las lentes o la distancia entre las lentes y el plano imagen. De esta manera, cuando se utiliza el modelo pinhole un punto en el sistema de coordenadas 3-D de la cámara, \$C\$, se proyecta en un punto en el sistema de coordenadas 2-D del sensor, \$S\$, de la siguiente manera:

$$p^S = \frac{f}{z^C} \cdot p^C \tag{2.3}$$

donde \$f\$ es la focal de la cámara.

Derivando con respecto al tiempo la expresión anterior, se obtiene:

$$\dot{p}^S = \frac{-f \cdot \dot{z}^C}{(z^C)^2} \cdot p^C + \frac{f}{z^C} \dot{p}^C \tag{2.4}$$

Considerando una cámara en movimiento y un objeto fijo, la ecuación fundamental de la cinemática toma el siguiente valor:

$$v^C = \omega^C \times p^C + v_t^C \tag{2.5}$$

Desarrollando la expresión anterior:

$$\mathbf{v}^c = \begin{bmatrix} \dot{\alpha}^c \\ \dot{\beta}^c \\ \dot{\gamma}^c \end{bmatrix} \times \begin{bmatrix} x^c \\ y^c \\ z^c \end{bmatrix} + \begin{bmatrix} \dot{x}_t^c \\ \dot{y}_t^c \\ \dot{z}_t^c \end{bmatrix} = \begin{bmatrix} \dot{x}_t^c - y^c \dot{\gamma}^c + z^c \dot{\beta}^c \\ \dot{y}_t^c - z^c \dot{\alpha}^c + x^c \dot{\gamma}^c \\ \dot{z}_t^c - x^c \dot{\beta}^c + y^c \dot{\alpha}^c \end{bmatrix} \quad (2.6)$$

Sustituyendo los valores obtenidos en la ecuación (2.6) en la expresión (2.4), se obtiene:

$$\dot{\mathbf{p}}^s = -f \frac{\dot{z}_t^c - x^c \dot{\beta}^c + y^c \dot{\alpha}^c}{(z^c)^2} \begin{bmatrix} x^c \\ y^c \\ z^c \end{bmatrix} + \frac{f}{z^c} \begin{bmatrix} \dot{x}_t^c - y^c \dot{\gamma}^c + z^c \dot{\beta}^c \\ \dot{y}_t^c - z^c \dot{\alpha}^c + x^c \dot{\gamma}^c \\ \dot{z}_t^c - x^c \dot{\beta}^c + y^c \dot{\alpha}^c \end{bmatrix} \quad (2.7)$$

Desarrollando la expresión anterior:

$$\dot{\mathbf{p}}^s = \begin{bmatrix} \frac{f \cdot \dot{x}_t^c}{z^c} - \frac{x^c \dot{z}_t^c f}{(z^c)^2} - \frac{x^c y^c \dot{\alpha}^c f}{(z^c)^2} + \left( f + \frac{(x^c)^2 f}{(z^c)^2} \right) \dot{\beta}^c - \frac{y^c \dot{\gamma}^c f}{z^c} \\ \frac{y_t^c f}{z^c} - \frac{y^c \dot{z}_t^c f}{(z^c)^2} - \left( f + \frac{(y^c)^2 f}{(z^c)^2} \right) \dot{\alpha}^c + \frac{x^c y^c \dot{\beta}^c f}{(z^c)^2} + \frac{x^c \dot{\gamma}^c f}{z^c} \\ 0 \end{bmatrix} \quad (2.8)$$

Sustituyendo las ecuaciones correspondientes al modelo de cámara pinhole,  $x^s = \frac{f}{z^c} x^c$  y

$y^s = \frac{f}{z^c} y^c$ , en la expresión anterior:

$$\begin{bmatrix} \dot{x}^s \\ \dot{y}^s \end{bmatrix} = \begin{bmatrix} \frac{f \cdot \dot{x}_t^c}{z^c} - \frac{x^s \dot{z}_t^c}{z^c} - \frac{x^s y^s \dot{\alpha}^c}{f} + \left( f + \frac{(x^s)^2}{f} \right) \dot{\beta}^c - y^s \dot{\gamma}^c \\ \frac{y_t^c f}{z^c} - \frac{y^s \dot{z}_t^c}{z^c} - \left( f + \frac{(y^s)^2}{f} \right) \dot{\alpha}^c + \frac{x^s y^s \dot{\beta}^c}{f} + x^s \dot{\gamma}^c \end{bmatrix} \quad (2.9)$$

Con lo que se obtiene la siguiente relación en la que se observa el valor del Jacobiano de la imagen:

$$\begin{bmatrix} \dot{x}^s \\ \dot{y}^s \end{bmatrix} = \begin{bmatrix} \frac{f}{z^c} & 0 & -\frac{x^s}{z^c} & -\frac{x^s y^s}{f} & \frac{f^2 + (x^s)^2}{f} & -y^s \\ 0 & \frac{f}{z^c} & -\frac{y^s}{z^c} & \frac{-f^2 - (y^s)^2}{f} & \frac{x^s y^s}{f} & x^s \end{bmatrix} \cdot \begin{bmatrix} \dot{x}_t^c \\ \dot{y}_t^c \\ \dot{z}_t^c \\ \dot{\alpha}^c \\ \dot{\beta}^c \\ \dot{\gamma}^c \end{bmatrix} \quad (2.10)$$

Suponiendo que el efector final del robot se está moviendo con una velocidad con respecto al sistema de coordenadas de la cámara en un sistema de cámara fija  $(\mathbf{v}_{tE}^C \ \boldsymbol{\omega}_E^C) = (\dot{x}_{tE}^C \ \dot{y}_{tE}^C \ \dot{z}_{tE}^C \ \dot{\alpha}_E^C \ \dot{\beta}_E^C \ \dot{\gamma}_E^C)$ , la velocidad de un punto,  $p_E^C(x_E^C, y_E^C, z_E^C)$ , en el extremo del robot se podrá expresar según la Ecuación (2.11) como se vio en el apartado de control basado en posición, pero en este caso referido al sistema de coordenadas de la cámara, de la siguiente manera:

$$\begin{bmatrix} \dot{x}_E^C \\ \dot{y}_E^C \\ \dot{z}_E^C \end{bmatrix} = - \begin{bmatrix} 0 & -z_E^C & y_E^C \\ z_E^C & 0 & -x_E^C \\ -y_E^C & x_E^C & 0 \end{bmatrix} \cdot \begin{bmatrix} \dot{\alpha}_E^C \\ \dot{\beta}_E^C \\ \dot{\gamma}_E^C \end{bmatrix} + \begin{bmatrix} \dot{x}_{tE}^C \\ \dot{y}_{tE}^C \\ \dot{z}_{tE}^C \end{bmatrix} \quad (2.11)$$

A partir de la Ecuación (2.10), el valor del Jacobiano de la imagen que muestra la relación entre la velocidad de una característica en el plano de la imagen (extraída del extremo del robot) y la velocidad (de rotación y traslación) del punto correspondiente con respecto al sistema de coordenadas de la cámara será:

$$\begin{bmatrix} \dot{f}_x \\ \dot{f}_y \end{bmatrix} = \begin{bmatrix} \frac{f}{z_E^C} & 0 & -\frac{f_x}{z_E^C} & -\frac{f_x \cdot f_y}{f} & \frac{f^2 + f_x^2}{f} & -f_y \\ 0 & \frac{f}{z_E^C} & -\frac{f_y}{z_E^C} & \frac{-f^2 - f_y^2}{f} & \frac{f_x \cdot f_y}{f} & f_x \end{bmatrix} \cdot \begin{bmatrix} \dot{x}_{tE}^C \\ \dot{y}_{tE}^C \\ \dot{z}_{tE}^C \\ \dot{\alpha}_E^C \\ \dot{\beta}_E^C \\ \dot{\gamma}_E^C \end{bmatrix} \quad (2.12)$$

Supóngase ahora el caso en el que el sistema de visión es capaz de extraer  $n$  puntos característicos de la imagen. En este caso, el Jacobiano de la imagen obtenido se puede generalizar para varios puntos de la siguiente manera:

$$\mathbf{J}_f = \begin{bmatrix} \mathbf{J}_{f1}(p_1) \\ \mathbf{J}_{f2}(p_2) \\ \vdots \\ \mathbf{J}_{fn}(p_n) \end{bmatrix} \quad (2.13)$$

Las dimensiones del Jacobiano serán  $i \times j$  siendo  $i/2$  el número de características observadas en la imagen y  $j$  los grados de libertad de la tarea.

Como se ha procedido para puntos, puede obtenerse el Jacobiano de la imagen para el caso que se desee realizar el seguimiento de elipses. Considerando que se extraen el centro  $(X_c, Y_c, z_c)$  y los momentos de orden dos,  $\mu_{20}, \mu_{02}, \mu_{11}$ , podrá obtenerse el Jacobiano de la imagen de la siguiente manera:

$$\mathbf{J}_m = \left[ \mathbf{J}_{X_c} \ \mathbf{J}_{Y_c} \ \mathbf{J}_{\mu_{20}} \ \mathbf{J}_{\mu_{02}} \ \mathbf{J}_{\mu_{11}} \right]^T \quad (2.14)$$

$$\mathbf{J}_{X_c} = \begin{bmatrix} -\frac{1}{z_c} & 0 & \frac{X_c}{z_c} + a\mu_{20} + b\mu_{11} & X_c Y_c + \mu_{11} & -1 - X_c^2 - \mu_{20} & Y_c \end{bmatrix} \quad (2.15)$$

$$\mathbf{J}_{Y_c} = \begin{bmatrix} 0 & -\frac{1}{z_c} & \frac{Y_c}{z_c} + a\mu_{11} + b\mu_{02} & 1 + Y_c^2 + \mu_{02} & -X_c Y_c - \mu_{11} & -X_c \end{bmatrix} \quad (2.16)$$

$$\mathbf{J}_{\mu_{20}} = [-2(a\mu_{20} + b\mu_{11}) \quad 0 \quad 2[(1/z_c + aX_c)\mu_{20} + bX_c\mu_{11}] \quad 2(Y_c\mu_{20} + X_c\mu_{02}) \quad -4\mu_{20}X_c \quad 2\mu_{11}] \quad (2.17)$$

$$\mathbf{J}_{\mu_{02}} = [0 \quad -2(a\mu_{11} - b\mu_{02}) \quad 2[(1/z_c + bY_c)\mu_{02} + aY_c\mu_{11}] \quad 4Y_c\mu_{02} \quad -2(Y_c\mu_{11} + X_c\mu_{02}) \quad -2\mu_{11}] \quad (2.18)$$

$$\mathbf{J}_{\mu_{11}}^T = \begin{bmatrix} -a\mu_{11} - b\mu_{02} \\ -a\mu_{20} - b\mu_{11} \\ aY_c\mu_{20} + \left(\frac{3}{z_c} - c\right)\mu_{11} + bX_c\mu_{02} \\ 3Y_c\mu_{11} + X_c\mu_{02} \\ -Y_c\mu_{20} - 3X_c\mu_{11} \\ \mu_{02} - \mu_{20} \end{bmatrix} \quad (2.19)$$

donde:

$$a = \frac{n_x}{n_x c_x + n_y c_y + n_z c_z} \quad b = \frac{n_y}{n_x c_x + n_y c_y + n_z c_z} \quad c = \frac{n_z}{n_x c_x + n_y c_y + n_z c_z} \quad (2.20)$$

y  $\mathbf{n}=(n_x, n_y, n_z)$  es la normal al plano formado por la elipse.

En general, el Jacobiano de la imagen es función de:

- Constantes calculadas a partir del proceso de calibración de la cámara.
- El valor actual de las características extraídas en la imagen.
- De información 3-D relativa a los puntos correspondientes a las características en la imagen (p. ej. la distancia de la cámara a la característica del objeto correspondiente).

Una cuestión no trivial es la de determinar la forma de actualizar el Jacobiano en el bucle de control debido a las dependencias mostradas anteriormente. Es posible distinguir dos casos dependiendo si la actualización se realiza en línea o a priori:

- *Actualización a priori del Jacobiano.* En este caso, el Jacobiano de la imagen es calculado una única vez antes de comenzar la tarea de control visual. El valor de las características en la imagen de las que depende el Jacobiano son consideradas iguales a las características deseadas, por lo tanto la distancia de la cámara al objeto será la deseada y consecuentemente la que se obtendría una vez finalizada la tarea. Esta aproximación disminuye la cantidad de cálculos a realizar durante el desarrollo de la tarea de control visual preservando la convergencia [Marchand-1996].

- *Actualización en línea del Jacobiano.* En este caso el Jacobiano de la imagen es recalculado en cada iteración en función de las características extraídas de la imagen. Como se observa en la Ecuación (2.12), el Jacobiano depende del valor  $z_E^C$ , que es la distancia de la cámara a la característica del objeto correspondiente, o en este caso al extremo del robot. Atendiendo a la forma de actualizar este valor se pueden encontrar las siguientes implementaciones:

- Extracción del valor  $z_E^C$  en cada iteración. Este caso se corresponde al cálculo más completo posible e implica la estimación, en cada iteración del bucle, de la profundidad del objeto (o del extremo del robot) respecto a la cámara [Dornaika-1996]. Si se dispone de un sistema de cámara fija, una manera obvia de determinar la profundidad es utilizando un par estereoscópico. En determinadas aplicaciones se parte de disponer previamente de una calibración que permita determinar en cada iteración el valor de esta distancia. Se puede utilizar otro tipo de sensor (aparte del visual), como pueden ser ultrasonidos, láser, etc.
- Aproximación del valor  $z_E^C$  al que se desea que se encuentre el robot una vez finalizada la tarea. En este caso se realiza la aproximación de que el objeto se encuentra situado a una distancia constante a la cámara. Esta aproximación es más eficiente computacionalmente; sin embargo, no se asegura la convergencia, en particular si la posición inicial de las características en la imagen se encuentran alejadas de su localización deseada [Chaumette-1997], siendo únicamente válido en una pequeña región del espacio.
- Cálculo mediante métodos de optimización que utilizan la regla de actualización de Broyden dada por [Hosoda-1994, Jagersand-1997]. Este método se ha generalizado al caso de objetos en movimiento en [Piepmeier-2004].

Recientemente, Malis [Malis-2004b] fusiona ambos métodos para conseguir mejorar la respuesta del control visual, mejorando la oscilación de las velocidades y obteniendo una trayectoria más suave en la imagen y en 3D. Sequeira et al. [Sequeira-2004] utiliza un modelo fuzzy de la matriz de interacción para controlar directamente el robot sin necesidad de calcular en cada iteración el Jacobiano de la imagen.

### 2.2.2. Controlador utilizando el Jacobiano de la imagen

Se va a suponer que el objetivo de una tarea particular va a ser alcanzar un conjunto de características en la imagen que se van a representar con un vector de características deseadas  $\mathbf{s}_d$ . En un instante de tiempo determinado, las características observadas en la imagen son  $\mathbf{s}$ . El regulador seleccionado debe ser tal que aplique sobre el robot las acciones oportunas de manera que los valores de  $\mathbf{s}$  se vayan acercando progresivamente a los de  $\mathbf{s}_d$ .

Haciendo uso de la aproximación de la función de tarea propuesta por Samson [Samson-1991], una tarea de control visual se puede expresar como una regulación a cero de la siguiente función de tarea:

$$\mathbf{e}(\mathbf{r}, t) = \hat{\mathbf{J}}_f^+ (\mathbf{s}(\mathbf{r}, t) - \mathbf{s}_d) \quad (2.21)$$

donde se observa que tanto el error,  $\mathbf{e}$ , como las características observadas en la cámara  $\mathbf{s}$  dependen de la localización de la cámara respecto al sistema de coordenadas del mundo,  $\mathbf{r}$ , y del tiempo,  $t$ . Además, de lo visto a lo largo de este capítulo se puede deducir que  $\mathbf{e}_f(\mathbf{r}, t) = \mathbf{s}(\mathbf{r}, t) - \mathbf{s}_d$ .

Con  $\hat{\mathbf{J}}_f$  se representa una estimación del Jacobiano de la imagen ya que, como se comentó anteriormente, depende de la distancia de la cámara al objeto seguido, y en muchas ocasiones se considera esta matriz constante igual a la correspondiente a la posición final deseada.

Como se observa, en la Ecuación (2.21) se ha empleado la pseudoinversa del Jacobiano que posteriormente también será utilizada para expresar la acción de control en forma de velocidad de traslación y rotación del extremo del robot en función del error expresado en términos de características en la imagen.

En la Expresión (2.21) se podría haber utilizado la inversa bajo la suposición de que la matriz Jacobiana sea cuadrada ( $i=j$ ), no singular y por tanto exista la inversa. Para que ocurra esto el número de características extraídas en la imagen debe ser 3, con lo que el Jacobiano de la imagen será de  $6 \times 6$ . Si se observan  $i/2$  características en la imagen de manera que  $i > j$ , se utiliza la siguiente pseudoinversa:

$$\mathbf{J}_f^+ = (\mathbf{J}_f^T \cdot \mathbf{J}_f)^{-1} \cdot \mathbf{J}_f^T \quad (2.22)$$

En el caso de que  $i < j$ , el sistema no observa suficientes características para determinar de manera unívoca el movimiento del objeto.

En este caso la matriz pseudoinversa utilizada tiene el siguiente valor:

$$\mathbf{J}_f^+ = \mathbf{J}_f^T \cdot (\mathbf{J}_f \cdot \mathbf{J}_f^T)^{-1} \quad (2.23)$$

Se desea que la función de tarea decrezca de manera exponencial, de forma que se cumpla:

$$\dot{e} = -k \cdot e \quad (2.24)$$

siendo  $k > 0$ . Por otro lado, como  $e$  es función de la localización de la cámara,  $r$ , y del tiempo,  $t$ , su derivada se puede expresar de la siguiente manera:

$$\dot{e} = \left( \frac{\partial e}{\partial r} \right) v^c + \frac{\partial e}{\partial t} \quad (2.25)$$

donde  $v^c$  es la velocidad de la cámara, y  $\frac{\partial e}{\partial t}$  representa las variaciones en el error debidas al movimiento del objeto. Por otro lado, a partir de la definición de la función de tarea se puede afirmar que  $\frac{\partial e}{\partial r}$  viene dado por:

$$\frac{\partial e}{\partial r} = \hat{J}_f^+ J_f \quad (2.26)$$

A partir de las Ecuaciones (2.24) y (2.25) es posible obtener la siguiente expresión para la velocidad de la cámara:

$$v^c = \left( \frac{\partial \hat{e}}{\partial r} \right)^+ \left( -ke - \frac{\partial \hat{e}}{\partial t} \right) \quad (2.27)$$

donde:

- $\left( \frac{\partial \hat{e}}{\partial r} \right)$  puede ser igualado a la matriz identidad ya que:

$$\frac{\partial \hat{e}}{\partial r} = \hat{J}_f^+ \hat{J}_f = I_6 \quad (2.28)$$

- $\frac{\partial \hat{e}}{\partial t}$  es una estimación del movimiento del objeto en la imagen. Si este término no es correctamente estimado se obtendrán errores en las tareas de seguimiento de un objeto en movimiento [Pomares-2006c].

Utilizando la ley de control mostrada en la Ecuación (2.27), el comportamiento de la función de la tarea vendrá dado por:

$$\dot{e} = \left( \frac{\partial e}{\partial r} \right) \left( \frac{\partial \hat{e}}{\partial r} \right)^+ \left( -ke - \frac{\partial \hat{e}}{\partial t} \right) + \frac{\partial e}{\partial t} \quad (2.29)$$

A partir de la ecuación anterior se puede obtener una condición suficiente de estabilidad asegurando el decrecimiento exponencial de la función de error:

$$\left(\frac{\partial e}{\partial r}\right)\left(\frac{\partial \hat{e}}{\partial r}\right)^+ = \left(\frac{\partial e}{\partial r}\right) > 0 \quad (2.30)$$

Se puede conseguir el comportamiento deseado  $\dot{e} = -ke$  bajo las condiciones siguientes:

$$\left(\frac{\partial e}{\partial r}\right)\left(\frac{\partial \hat{e}}{\partial r}\right)^+ = \left(\frac{\partial e}{\partial r}\right) = I_6 \quad (2.31)$$

$$\frac{\partial \hat{e}}{\partial t} = \frac{\partial e}{\partial t} \quad (2.32)$$

La condición (2.31) se puede expresar como  $\left(\frac{\partial e}{\partial r}\right) = \hat{\mathbf{J}}_f^+ \mathbf{J}_f = I_6$  que se verifica si el modelo  $\hat{\mathbf{J}}_f$

elegido para  $\mathbf{J}_f$  es exacto. En caso contrario, los errores introducidos en este modelo provocarán variaciones en el comportamiento exponencial de la función de tarea. La segunda condición muestra la importancia de estimar correctamente el movimiento del objeto seguido con la finalidad de reducir errores en el seguimiento.

Para ilustrar lo visto en este apartado supóngase que se dispone de un sistema compuesto por una cámara localizada en el extremo de un robot y un objeto del cual se extraen cuatro características. En la Figura 2-14 se observa la localización inicial de la cámara junto con las características extraídas, mientras que en la Figura 2-15 se representa la localización deseada.

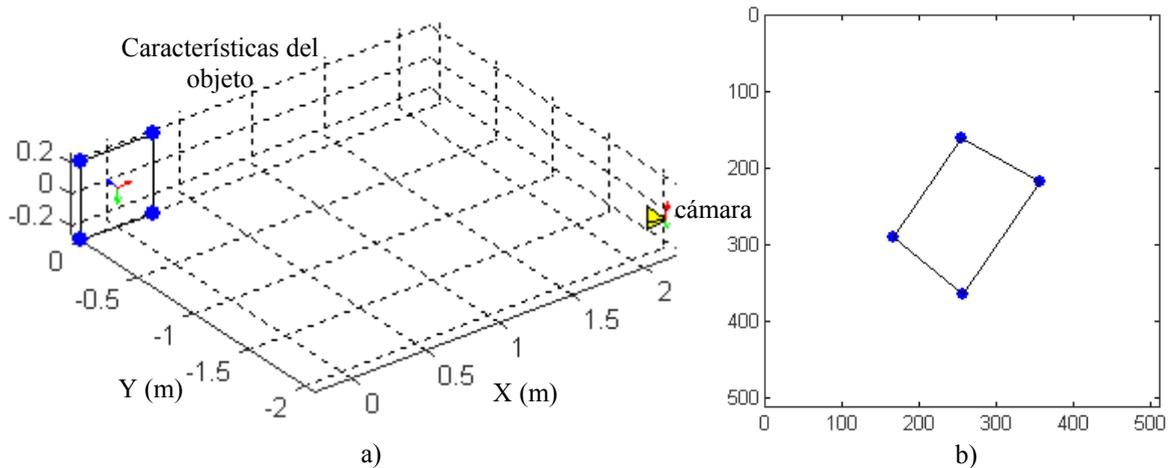


Figura 2-14. a) Posición inicial de la cámara. b) Imagen obtenida por la cámara en la que se observan las características iniciales del objeto en la imagen.

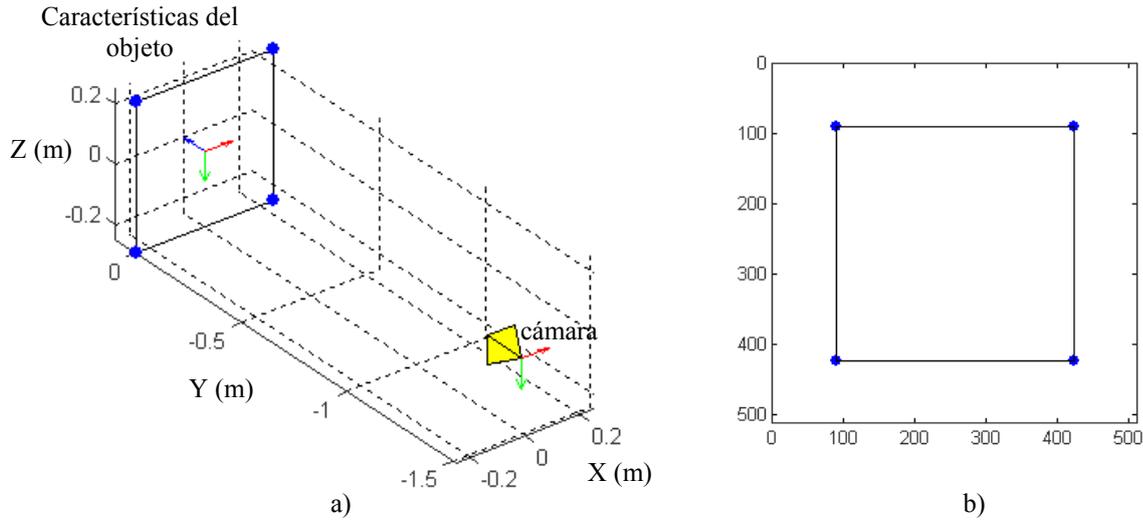


Figura 2-15. a) Posición deseada de la cámara. b) Imagen obtenida por la cámara en la que se observan las características deseadas del objeto en la imagen.

Bajo estas consideraciones el Jacobiano de la imagen tomará el siguiente valor:

$$\begin{bmatrix} \dot{f}_{1x} \\ \dot{f}_{1y} \\ \dot{f}_{2x} \\ \dot{f}_{2y} \\ \dot{f}_{3x} \\ \dot{f}_{3y} \\ \dot{f}_{4x} \\ \dot{f}_{4y} \end{bmatrix} = \begin{bmatrix} \frac{f}{z_1} & 0 & -\frac{f_{1x}}{z_1} & -\frac{f_{1x} \cdot f_{1y}}{f} & \frac{f^2 + f_{1x}^2}{f} & -f_{1y} \\ 0 & \frac{f}{z_1} & -\frac{f_{1y}}{z_1} & \frac{-f^2 - f_{1y}^2}{f} & \frac{f_{1x} \cdot f_{1y}}{f} & f_{1x} \\ \frac{f}{z_2} & 0 & -\frac{f_{2x}}{z_2} & -\frac{f_{2x} \cdot f_{2y}}{f} & \frac{f^2 + f_{2x}^2}{f} & -f_{2y} \\ 0 & \frac{f}{z_2} & -\frac{f_{2y}}{z_2} & \frac{-f^2 - f_{2y}^2}{f} & \frac{f_{2x} \cdot f_{2y}}{f} & f_{2x} \\ \frac{f}{z_3} & 0 & -\frac{f_{3x}}{z_3} & -\frac{f_{3x} \cdot f_{3y}}{f} & \frac{f^2 + f_{3x}^2}{f} & -f_{3y} \\ 0 & \frac{f}{z_3} & -\frac{f_{3y}}{z_3} & \frac{-f^2 - f_{3y}^2}{f} & \frac{f_{3x} \cdot f_{3y}}{f} & f_{3x} \\ \frac{f}{z_4} & 0 & -\frac{f_{4x}}{z_4} & -\frac{f_{4x} \cdot f_{4y}}{f} & \frac{f^2 + f_{4x}^2}{f} & -f_{4y} \\ 0 & \frac{f}{z_4} & -\frac{f_{4y}}{z_4} & \frac{-f^2 - f_{4y}^2}{f} & \frac{f_{4x} \cdot f_{4y}}{f} & f_{4x} \end{bmatrix} \begin{bmatrix} \dot{x}_{iO}^C \\ \dot{y}_{iO}^C \\ \dot{z}_{iO}^C \\ \dot{\alpha}_O^C \\ \dot{\beta}_O^C \\ \dot{\gamma}_O^C \end{bmatrix} \quad (2.33)$$

A partir de la Ecuación (2.27) y considerando que el objeto del que se extraen las características no se encuentra en movimiento, se obtiene la siguiente ley de control:

$$\mathbf{v}^C = -\mathbf{k} \cdot \mathbf{J}_f^+ \cdot (\mathbf{s} - \mathbf{s}_d) \quad (2.34)$$

Por lo tanto, el sistema de control basado en imagen realizará las acciones necesarias sobre el robot para que se produzcan los cambios en las características observadas en la imagen que hagan que

estas características,  $s$ , alcancen el valor de las características deseadas,  $s_d$ . Aplicando la Ecuación (2.34) se obtiene la siguiente evolución con el número de iteraciones de las velocidades aplicadas al robot utilizando una ganancia de 0,125:

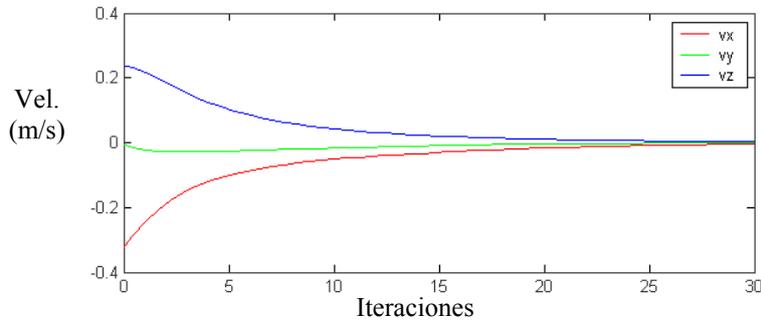


Figura 2-16. Evolución de la velocidad lineal aplicada al robot.

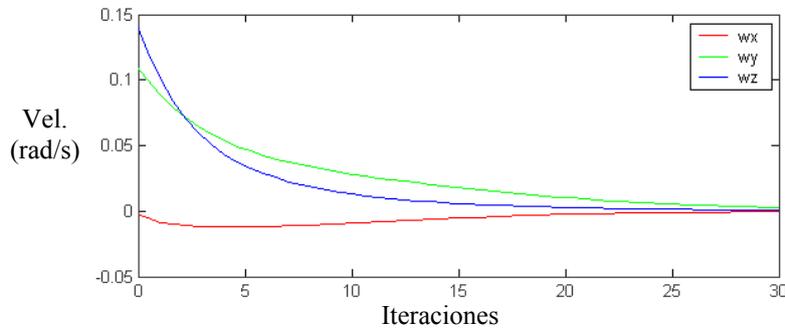


Figura 2-17. Evolución de la velocidad angular aplicada al robot.

Con esta acción de control la evolución del error de cada una de las características será la siguiente:

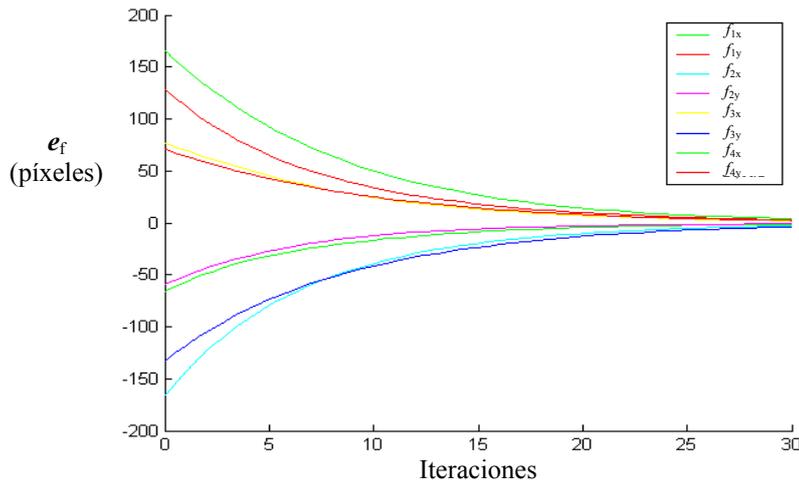


Figura 2-18. Evolución del error de cada característica expresada en píxeles ( $f_i - f_{id}$ )  $i = x, y$ .

En la Figura 2-19.a se muestra la trayectoria en el espacio cartesiano descrita por la cámara para alcanzar la posición deseada desde su posición inicial. En la Figura 2-19.b se muestra la evolución, durante la trayectoria de la cámara, de las características extraídas en la imagen:

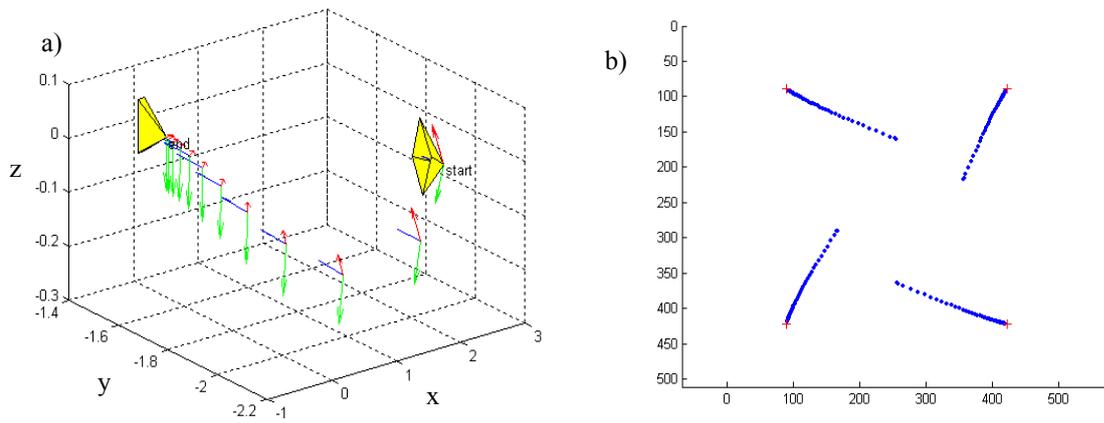


Figura 2-19. a) Trayectoria seguida por la cámara. b) Trayectoria de las características extraídas en la imagen.

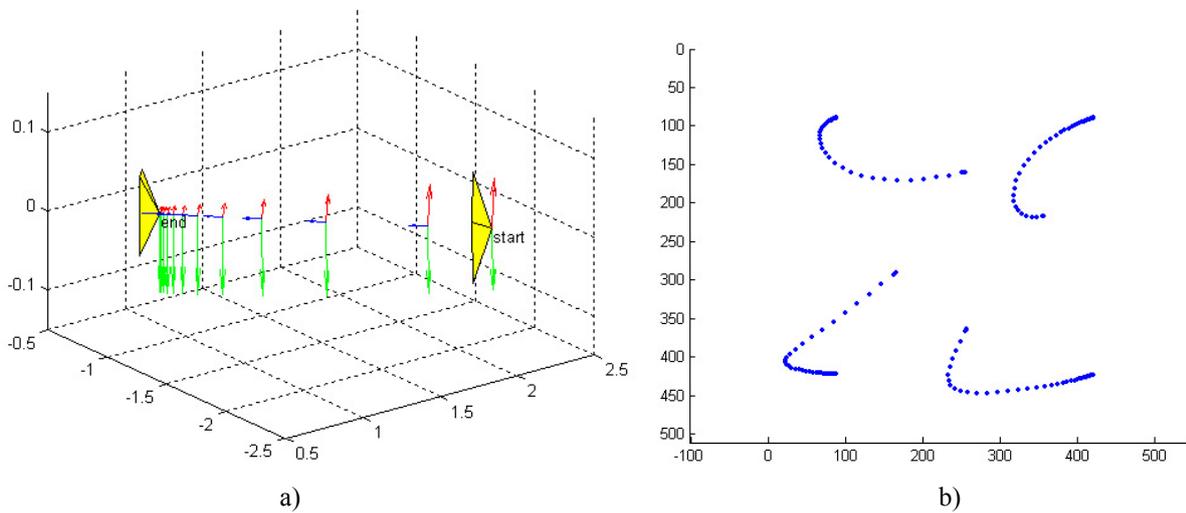


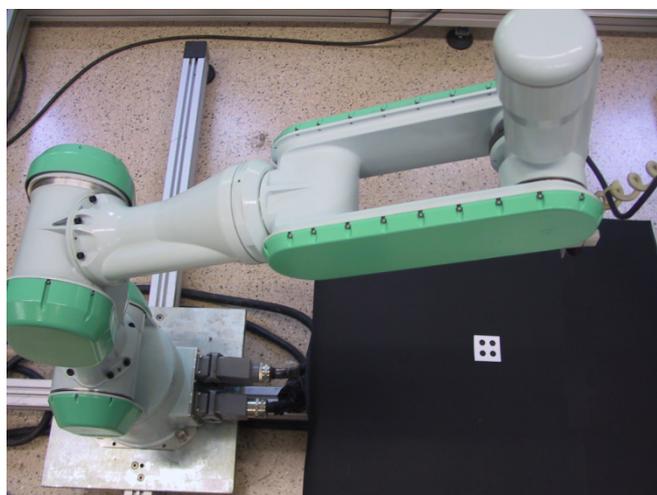
Figura 2-20. a) Trayectoria seguida por la cámara en el caso de control visual basado en posición. b) Trayectoria de las características extraídas en la imagen en el caso de control visual basado en imagen considerando la distancia de la cámara al objeto igual a la deseada.

En la Figura 2-20.a se muestra la trayectoria que se obtendría para la misma tarea, pero aplicando control visual basado en posición. Se observa como la trayectoria obtenida para la cámara es recta, frente a la trayectoria obtenida empleando control visual basado en imagen (Figura 2-19.a). En este último caso se obtiene una trayectoria recta en las características extraídas en la imagen (Figura 2-19.b). Como se comentó anteriormente, en muchas ocasiones cuando se emplea control visual basado en imagen se evita calcular la distancia de la cámara al objeto, considerándola como la deseada, es decir, la que se obtendrá una vez finalizada la tarea. Esta aproximación, que aligera notablemente los cálculos, provoca que se obtenga una evolución más oscilatoria de las características extraídas en la imagen (Figura 2-20.b).

## 2.3. Caso práctico

### 2.3.1. Planteamiento del problema

En este apartado se van a describir los distintos componentes de un sistema de control visual cuyos detalles se pueden encontrar en [GarcíaGJ-2004]. Este sistema está formado por un robot antropomórfico de 7 grados de libertad que será el encargado de realizar el seguimiento de un objeto a lo largo de su recorrido (ver Figura 2-21). Para realizar este seguimiento se va a hacer uso de una cámara situada en el extremo del robot.



*Figura 2-21. Sistema de control visual con cámara en el extremo para realizar el seguimiento de un patrón. Se observa cómo se simplifica los procesamientos en la imagen marcando cuatro puntos característicos.*

La estrategia de control visual utilizada ha sido de tipo “ver y mover” dinámico basado en imagen. De esta manera, las acciones de control se obtienen a partir de características extraídas de la imagen sin necesidad de determinar la localización 3-D del objeto seguido.

El sistema de control visual debe encargarse de ejercer las acciones de control adecuadas sobre el robot de manera que su extremo siga un determinado objeto a lo largo de su movimiento. El objeto elegido (Figura 2-22) es un cuadrado blanco de cinco centímetros de lado, en cuyo interior se han situado formando otro cuadrado de menor tamaño cuatro puntos negros de un centímetro de diámetro. Se ha seleccionado este patrón debido a que simplifica enormemente los procesamientos en la imagen, a la vez que permite reducir retardos debido a dichos procesamientos.

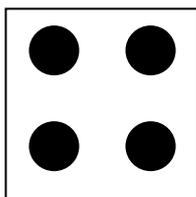


Figura 2-22. Patrón característico del objeto seguido.

### 2.3.2. Arquitectura del sistema

Los distintos componentes, mostrados en la Figura 2-23, que intervienen en el sistema de seguimiento son:

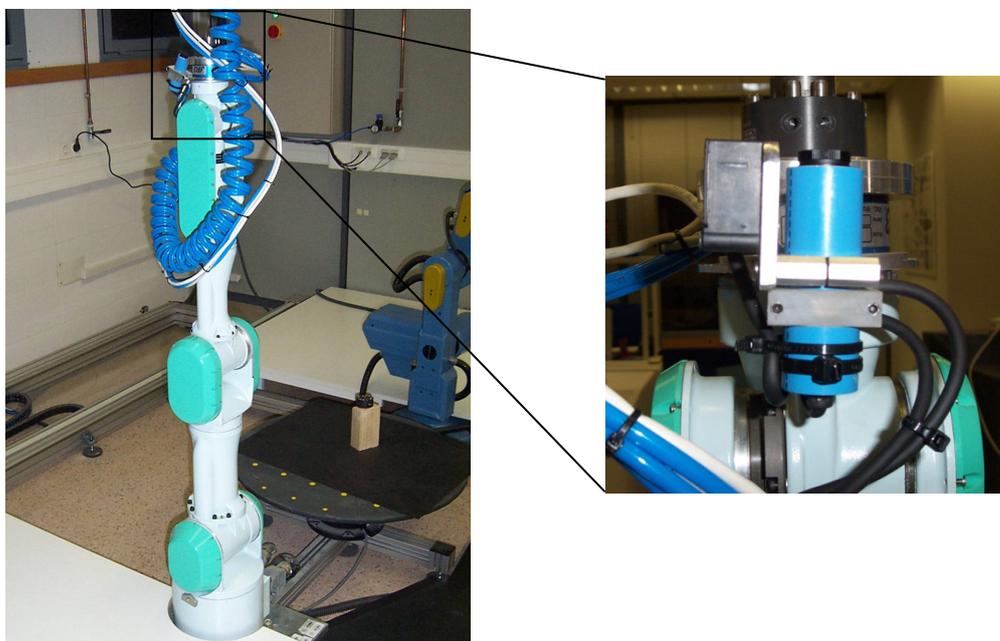


Figura 2-23. Arquitectura del sistema. En la figura de la derecha se muestra un detalle de la cámara situada en el extremo del robot.

- *Robot.* El robot utilizado es el modelo PA-10 de Mitsubishi. Se trata de un robot antropomórfico de 7 grados de libertad que será el encargado de realizar el seguimiento manteniendo una distancia fija entre su extremo y el objeto.
- *Sistema de adquisición de imágenes.* La adquisición de imágenes se realiza haciendo uso de una minicámara JAI CV-M536 ubicada en el extremo del robot. Se ha utilizado una tarjeta MATROX GENESIS para la adquisición y procesamiento de las imágenes.
- *Objeto seguido.* El objeto seguido se trata de un patrón cuadrado con cuatro marcas que serán las que se reconocerán durante el seguimiento. En la Figura 2-21 se mostró una vista

superior del sistema propuesto en la que se observa el patrón que se empleará para realizar el seguimiento.

### 2.3.3. Sistema de visión

A continuación se van a exponer las principales cuestiones a considerar relativas a los componentes de visión artificial correspondientes al sistema de control visual propuesto.

Debido a que se pretende realizar control visual basado en imagen, es necesario extraer las suficientes características en la imagen que permitan realizar el guiado del robot, sin necesidad de determinar la localización del objeto seguido. Asimismo, estas características habrán de extraerse con el menor retardo posible para permitir una mayor cadencia en el sistema.

En la Figura 2-21 se ha mostrado el patrón correspondiente al objeto seguido en el que se observa como claramente dispone de 4 puntos característicos que van a facilitar el proceso de extracción de las características utilizadas en el control visual. De esta manera se obtendrá una matriz de  $8 \times 6$  como Jacobiano de la imagen. Esta matriz será función de  $z_i$ , es decir, de la distancia de la cámara a la característica del objeto observada, por lo tanto, será necesario extraer información relativa a la profundidad de cada una de las tres características.

Los procesamientos en la imagen para la determinación de estas características se van a limitar a una binarización (haciendo uso de un umbral determinado empíricamente) y una posterior detección de centros de gravedad que permitirá obtener las coordenadas  $(x, y)$  de las tres características.

En sistemas como el que aquí se presenta, en los que se utiliza una única cámara, la estimación de esta profundidad es compleja y, a menudo, se evita considerando los valores de  $z_i$  del Jacobiano de la imagen como la distancia deseada de la cámara al objeto,  $z_d$ . Sin embargo, estas aproximaciones sólo aseguran la disminución exponencial de la función de error en un entorno cercano a la posición deseada, con lo que la opción adoptada ha sido la de determinar la profundidad en cada iteración del bucle de control.

Más concretamente, el método seleccionado consiste en, conocido el modelo geométrico del objeto seguido, realizar una calibración previa, determinando correspondencias entre distancias cámara-objeto y área encerrada por las 4 marcas presentes en el objeto seguido medida en píxeles. Seguidamente se obtiene una interpolación lineal (Figura 2-24) que pasa por los puntos anteriores.

La función representada en la Figura 2-24 será la utilizada para determinar en cada iteración del bucle de control la distancia de la cámara al objeto en función del área encerrada entre las marcas del patrón. De esta manera, los valores  $z_i$  del Jacobiano de la imagen serán los correspondientes a esta distancia.

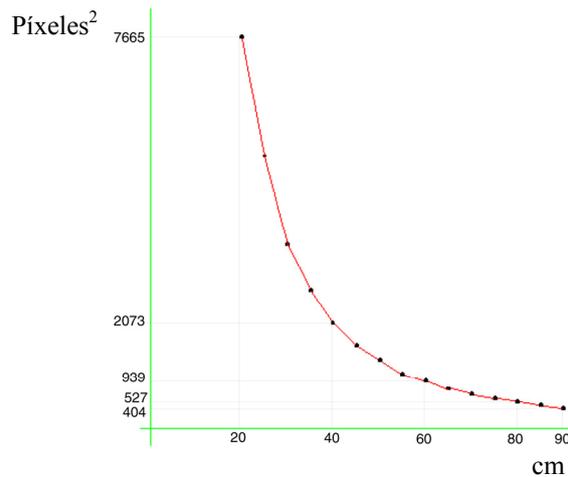


Figura 2-24. Interpolación lineal para la obtención de la distancia cámara-objeto.

### 2.3.4. Control visual. Formulación

Realizando la suposición de que el objeto seguido se encuentra fijo, y que el sistema de visión artificial extrae unas características  $\mathbf{s}$  de la imagen, se puede afirmar que este conjunto dependerá únicamente de la posición de la cámara respecto al objeto seguido. Bajo estas consideraciones puede utilizarse la ley de control mostrada en la Ecuación (2.34) para la disminución del error de forma progresiva.

En una tarea de seguimiento como la implementada, el objeto se encuentra en movimiento, con lo que la acción de control tomará el siguiente valor:

$$\mathbf{v}^C = -\mathbf{k} \cdot \mathbf{J}_f^+ \cdot (\mathbf{s} - \mathbf{s}_d) - \frac{\partial \hat{\mathbf{e}}}{\partial t} \quad (2.35)$$

donde el ultimo término de la expresión anterior expresa una estimación de la variación del error debido al movimiento del objeto seguido.

Solo queda considerar que las velocidades se encuentran expresadas respecto al sistema de referencia de la cámara, pero es necesario transformarlas respecto al sistema de coordenadas del extremo del robot para poder enviarlas al controlador (véase Figura 2-25).

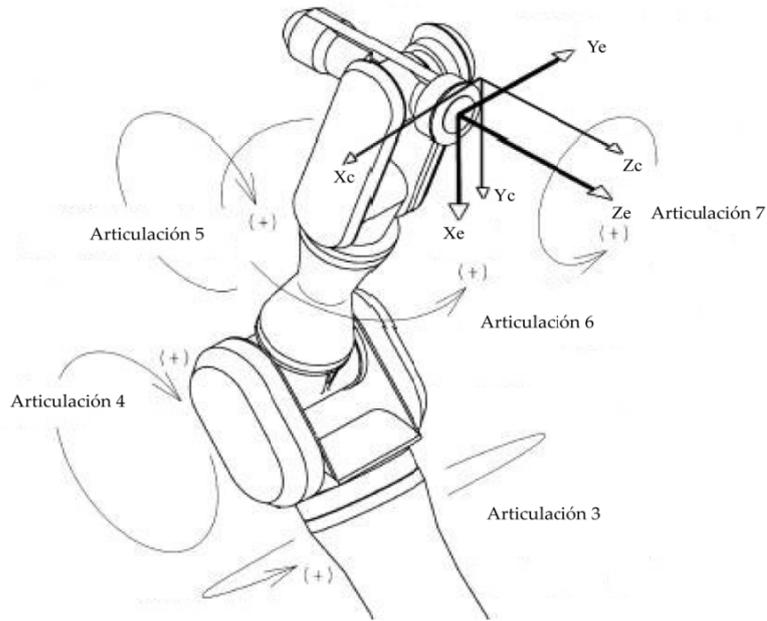


Figura 2-25. Sistemas de coordenadas asociados al extremo del robot y a la cámara.

### 2.3.5. Resultados

A continuación se muestran distintas pruebas en las que se observa el funcionamiento del sistema. En el conjunto de pruebas se considera que el objetivo a alcanzar se encuentra rotado y localizado en la parte central de la imagen, de forma que el movimiento que habría de realizar el robot consistiría en una rotación y un desplazamiento en horizontal hacia la derecha, de forma que quedarán las características centradas a la derecha. Para realizar estas pruebas se ha utilizado la herramienta denominada ViSeC descrita en [GarcíaGJ-2004] y que se muestra en la Figura 2-26.

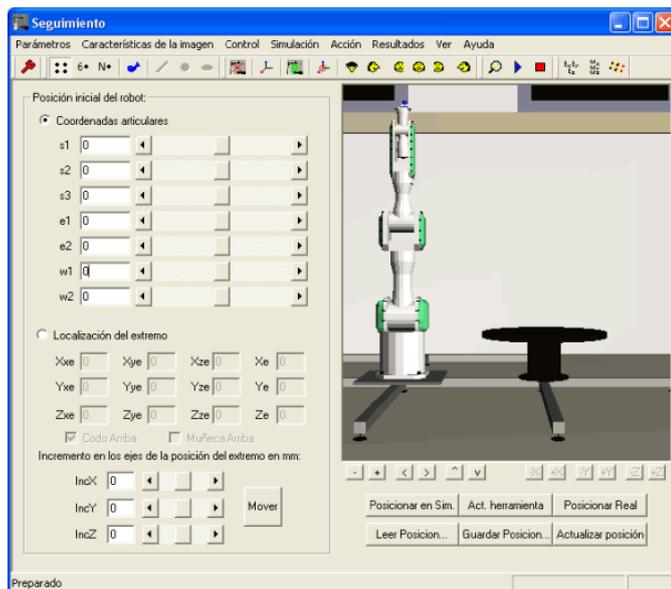


Figura 2-26. ViSeC: Aplicación utilizada para realizar las pruebas de control visual.

En la Figura 2-27 se observan los puntos detectados marcados con una cruz en su centro de gravedad. A la derecha aparecen los cuadrados de error numerados, correspondientes a la situación de las características deseadas. En la detección aparece también el número de blobs detectados y las coordenadas del centro de gravedad y el área de cada uno de ellos. Por último se observa en la parte inferior el tiempo de grabación (que es el tiempo invertido en la captura de una imagen desde la cámara), y el tiempo de proceso (el tiempo invertido en la detección de los blobs que engloba tareas como la binarización, la eliminación de blobs por los umbrales de área o la ordenación de las características). La calibración previa de la cámara permite determinar que la focal empleada en la matriz de interacción será de 7,44 mm.



Figura 2-27. Detección de los puntos característicos. Estado previo al control visual.

Al ejecutar se observa en el plano imagen como los cuatro puntos se dirigen a la posición deseada. Las velocidades altas iniciales se distinguen en la separación inicial de los puntos. Lógicamente, cuando disminuye el error las velocidades disminuyen y los puntos se comprimen más. Todo esto se observa en la Figura 2-28, que muestra el estado de la monitorización del control visual con la traza visual activada.

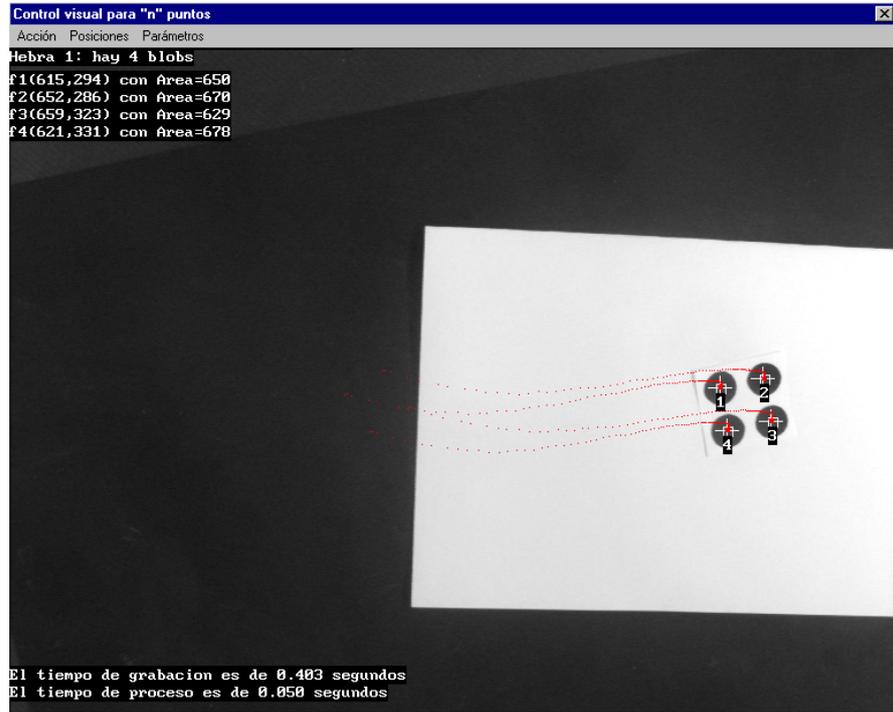


Figura 2-28. Evolución de las características de la imagen desde la posición original a la deseada.

Tras realizar el control visual se puede ver el comportamiento del sistema en las gráficas. Las velocidades lineales obtenidas en el caso anterior se muestran en la Figura 2-29.

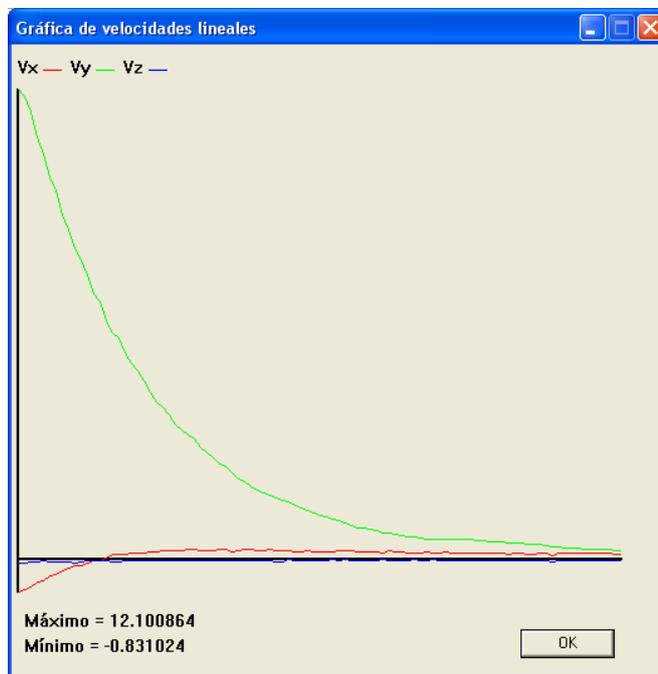


Figura 2-29. Gráfica de las velocidades lineales para los cuatro puntos.

Las velocidades tienden a cero rápidamente, se observa que la máxima velocidad la tenemos en el eje X ya que es aquí donde más error se tiene como se observa en la Figura 2-30.

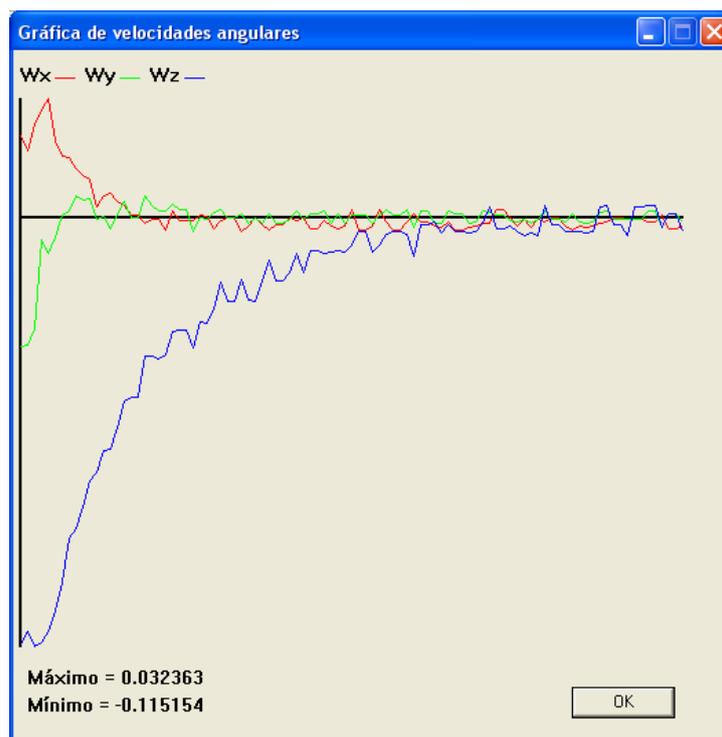


Figura 2-30. Gráfica de las velocidades angulares para los cuatro puntos.

En las velocidades angulares se obtiene también convergencia. Se observa que aparecen vibraciones que se reducirán con ganancias menores. En este caso se ha usado una ganancia de 0,3.

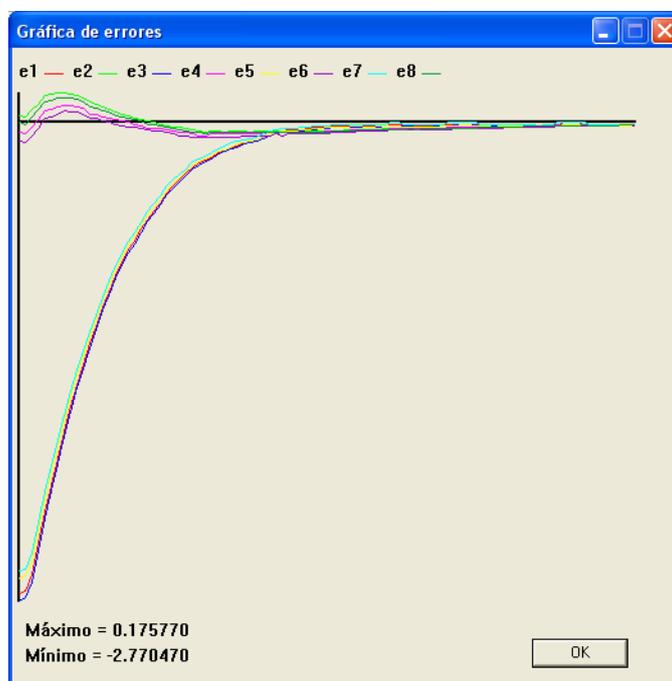
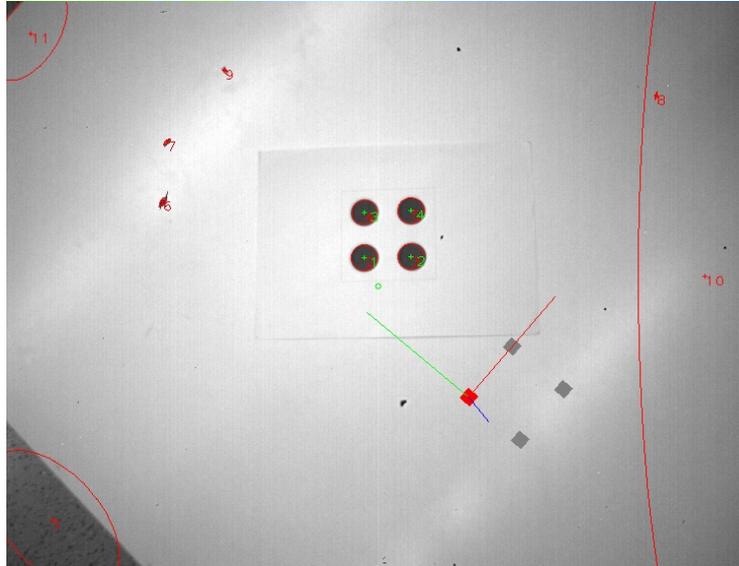


Figura 2-31. Gráfica de la evolución del error para los cuatro puntos.

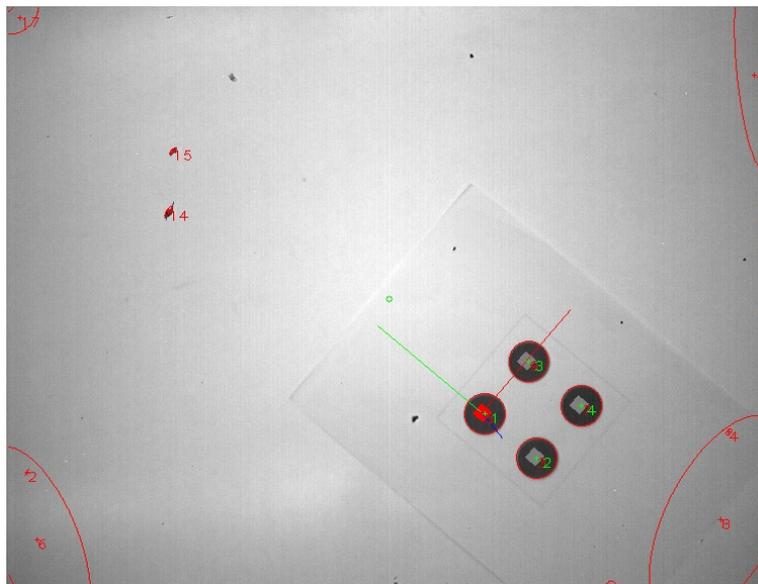
Las curvas de error convergen a cero sin fluctuaciones, produciéndose una trayectoria suave desde la posición inicial hasta la deseada (ver Figura 2-31).

En los siguientes experimentos se va a probar el sistema sin el módulo que permite calcular en cada iteración la distancia entre la cámara y el objeto. En la Figura 2-32 y en la Figura 2-33 se observan las situaciones inicial y final, respectivamente.



*Figura 2-32. Posición inicial de la cámara.*

La Figura 2-34 muestra el error total cometido durante la tarea de control visual. La Figura 2-35 representa el error cometido en píxeles durante la tarea para cada característica por separado. En la Figura 2-36 y la Figura 2-37 se muestran, respectivamente, las velocidades traslacional y rotacional aplicadas al robot durante la tarea de control visual. En cuanto a estas gráficas se puede comentar que aquí el hecho de que la matriz de interacción sea solamente una estimación (ya que se usa la matriz de interacción de la posición deseada) influye de forma significativa pero no determinante a suavizar la convergencia y producirla de forma más lenta.



*Figura 2-33. Posición final de la cámara.*

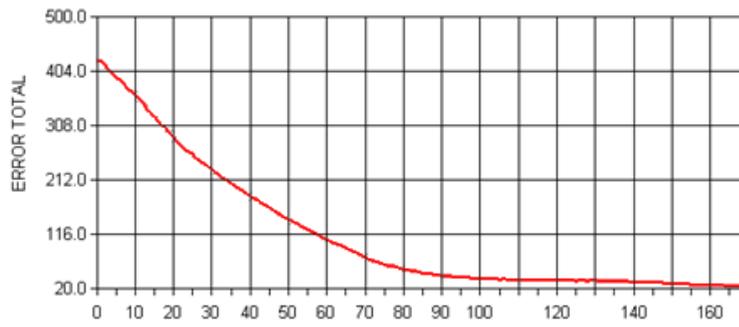


Figura 2-34. Gráfica de error total.

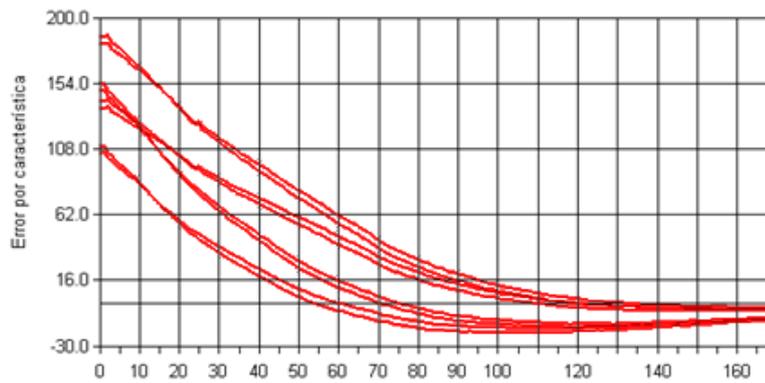


Figura 2-35. Gráfica de error por característica.

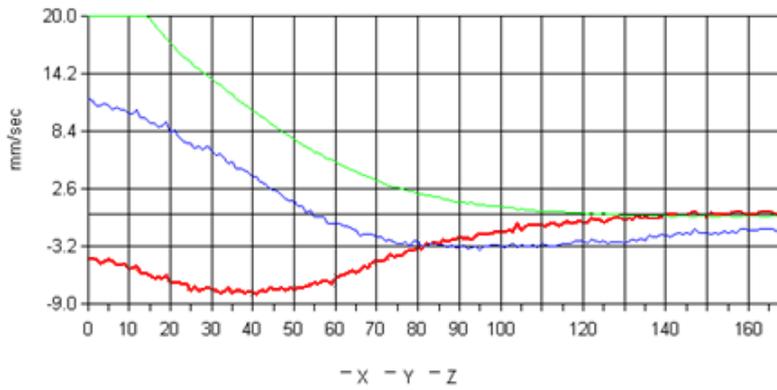


Figura 2-36. Gráfica de velocidades traslacionales.

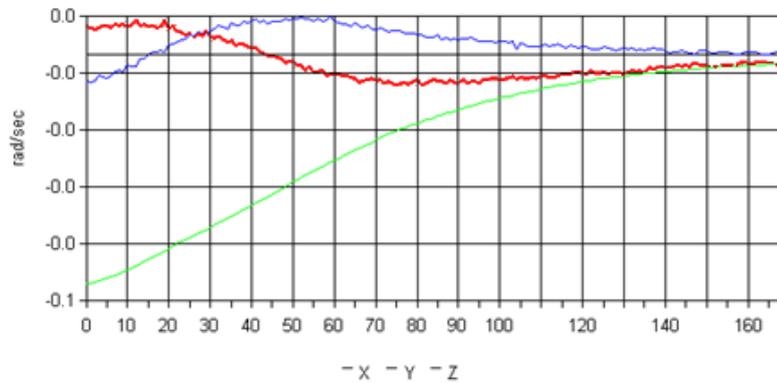
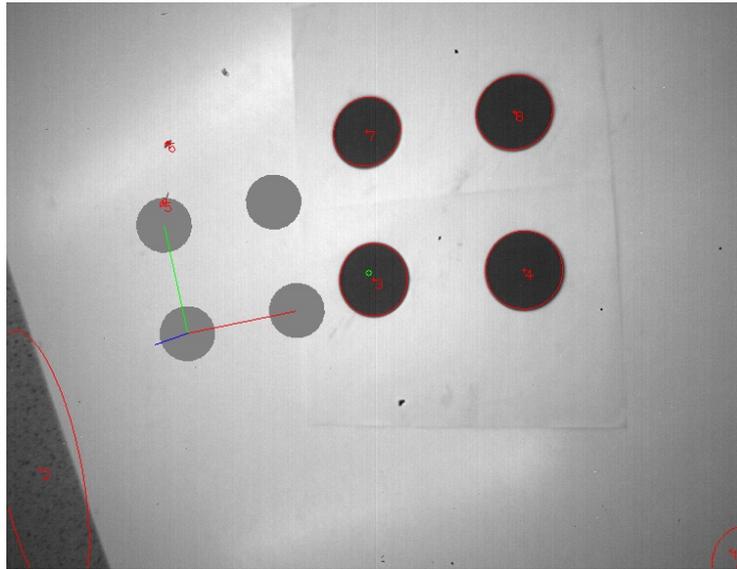


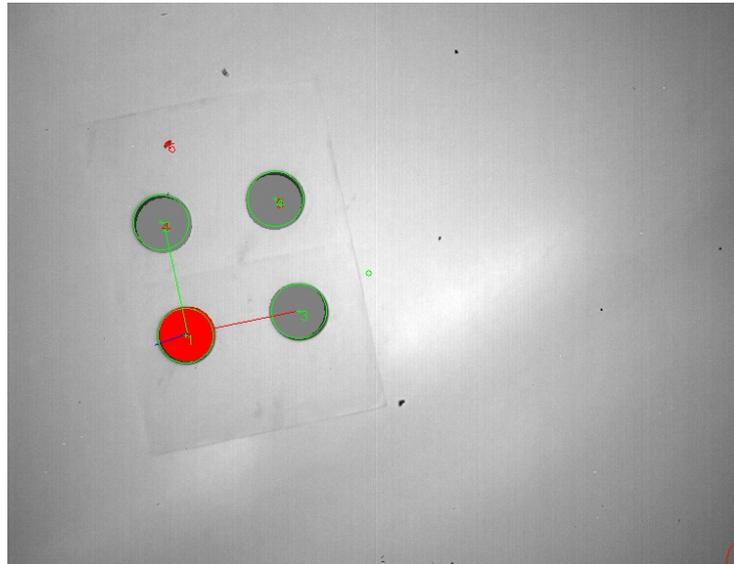
Figura 2-37. Gráfica de velocidades rotacionales.

A continuación se describe un experimento en el que se toman como características para realizar el control visual las elipses. Además, se pretende conseguir un cambio de profundidad mayor que en los otros experimentos. Debido a la razón antes comentada, la estimación de la matriz de interacción mejora conforme se mueve la cámara a la misma profundidad y pierde validez cuando se acerca o aleja de esa distancia.

En la Figura 2-38 se muestra la posición inicial y en la Figura 2-39 la final de la cámara del robot.



*Figura 2-38. Posición inicial de la cámara (elipses).*



*Figura 2-39. Posición final de la cámara (elipses).*

La Figura 2-40 muestra la gráfica del error total cometido en píxeles. Se observa como el cálculo de la matriz de interacción afecta a este error total debido a la estimación sobre todo de esa componente de profundidad que falta. La Figura 2-41 muestra la evolución del módulo de error expresado en píxeles de la imagen con el número de iteraciones de cada una de las características. Por

último, en la Figura 2-42 se ha representado la velocidad lineal enviada al robot durante la tarea y en la Figura 2-43 se muestra la velocidad angular aplicada al robot frente al número de iteraciones. En la gráfica de la Figura 2-42 se observa que el error inicial es tan alto que las velocidades lineales se limitan a 20 mm/s, que es el valor máximo a enviar al robot por seguridad.



Figura 2-40. Gráfica de error total (elipses).

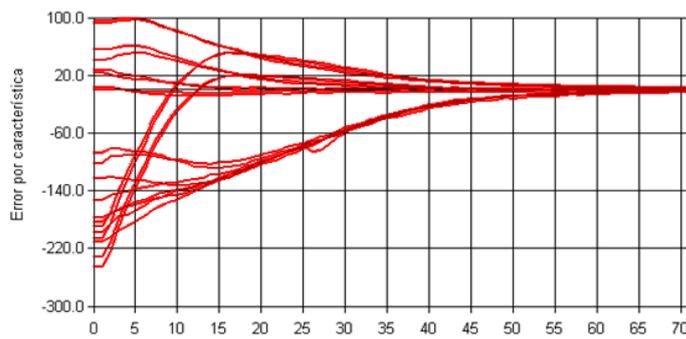


Figura 2-41. Gráfica de error por característica (elipses).

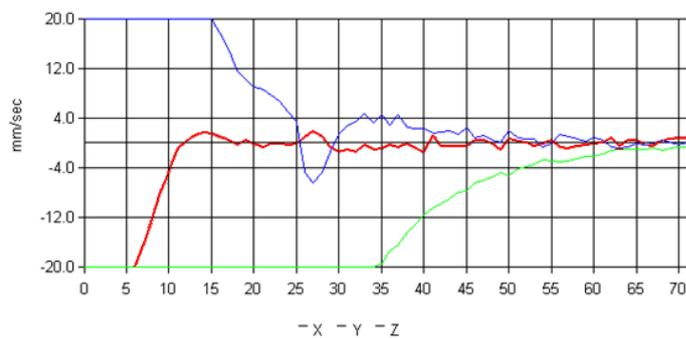


Figura 2-42. Gráfica de velocidades traslacionales (elipses).

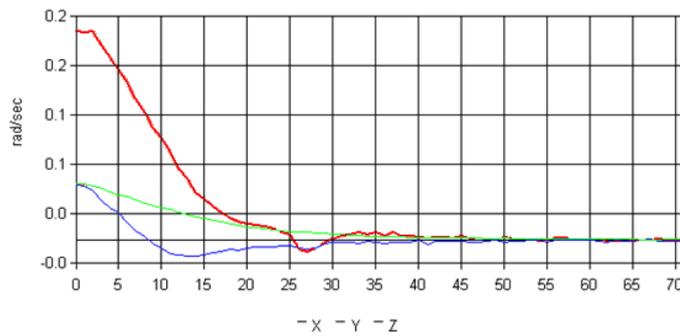


Figura 2-43. Gráfica de velocidades rotacionales (elipses).

### 2.3.6. Conclusiones

Las experiencias previas descritas en lo largo de este apartado han permitido corroborar ciertos aspectos de los sistemas de control visual que anteriormente habían sido expuestos teóricamente. Se ha comprobado que, ante unas determinadas características deseadas en la imagen que se pretenden alcanzar, no es posible determinar la trayectoria 3-D seguida por el robot.

Se han comprobado otros aspectos relacionados con los sistemas clásicos de control basado en imagen tales como la necesidad de estimar el error para compensar el movimiento del objeto seguido, complejidad de la determinación de la distancia cámara-objeto así como la necesidad de reducir retardos para mejorar el desempeño de este tipo de sistemas.

De esta manera se puede concluir que, si bien estos sistemas presentan un correcto comportamiento para movimientos punto a punto en los que los puntos origen y destino se encuentran cercanos, no resultan adecuados para el seguimiento de una determinada trayectoria deseada entre dichos puntos.

## 2.4. Conclusiones

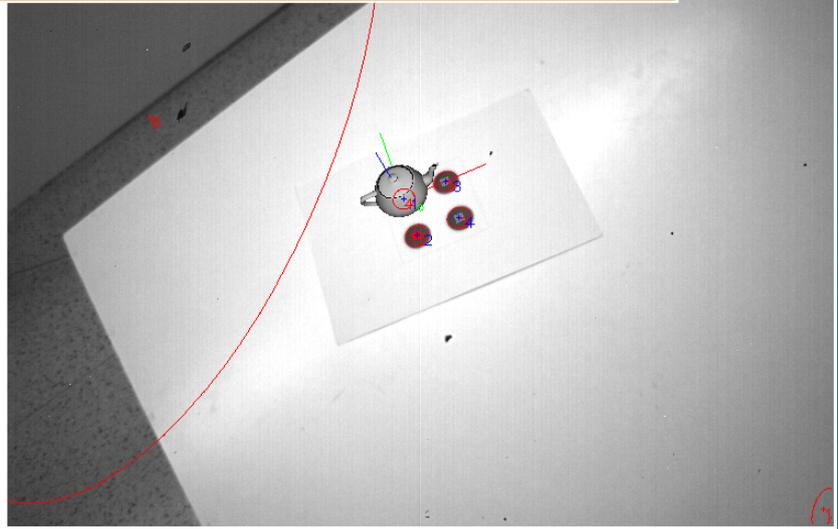
A lo largo de este capítulo se ha mostrado una visión general de los sistemas de control visual, citando los principales modos de control, así como una evolución de las distintas líneas de investigación dentro de este campo hasta llegar a la actualidad. Una primera conclusión que se puede obtener de este estudio es que se trata de un campo relativamente reciente con muchas líneas de investigación aún abiertas.

Durante este capítulo se ha prestado una mayor atención a los sistemas de control visual indirectos basados en imagen ya que será ésta la estrategia de control básica empleada en los capítulos siguientes, por lo que durante la presente memoria de investigación se hará referencia a los distintos conceptos, resultados y desarrollos matemáticos que se han expuesto a lo largo de este capítulo.

Se han detallado las consideraciones necesarias para el desarrollo de un sistema de control visual basado en imagen mostrando incluso una aplicación real. Se podría concluir que lo atractivo de estos sistemas es la posibilidad de realizar el guiado de un robot sin la necesidad de disponer de un conocimiento exhaustivo del entorno, y haciendo uso únicamente de información acerca de determinadas características del objeto implicado en la tarea. Sin embargo, se han detectado ciertos inconvenientes que es necesario solventar, en mayor medida cuando se aplican a entornos reales. Uno de estos inconvenientes es el hecho de que el sistema tiende a converger hacia unas determinadas características deseadas en la imagen, pero no se conoce cuál será su evolución en el espacio cartesiano 3-D. Esto, unido a los problemas de convergencia existentes cuando las características iniciales se encuentran lejanas a las deseadas, ha propiciado el diseño de un nuevo método para realizar el seguimiento de trayectorias definidas en el espacio imagen usando control visual.

# Capítulo 3

## Control Visual Virtual



## 3.1. Introducción

---

El presente capítulo describe la técnica de control visual virtual. El control visual virtual es una técnica que permite obtener los parámetros extrínsecos de una cámara a partir de una imagen tomada por ésta. En el Capítulo 4 se describe un método de seguimiento intemporal de trayectorias en la imagen que hace uso de las propiedades del control visual virtual para obtener la posición de la cámara respecto del objeto que se pretende seguir.

A continuación se detallan los fundamentos teóricos del control visual virtual. El problema de la calibración de la cámara usando control visual virtual ya ha sido descrito en [Marchand-2001, Kinoshita-1994]. Básicamente, el control visual virtual trata de minimizar el error entre las características observadas en la imagen y las características calculadas por retroproyección a partir de los parámetros intrínsecos y extrínsecos actuales. Estos parámetros se van modificando en cada iteración hasta que las características observadas y proyectadas coinciden. En este momento se habrán obtenido los parámetros extrínsecos e intrínsecos de la cámara real.

En las siguientes secciones se muestran los resultados obtenidos para la calibración de una cámara a partir de distintas posiciones iniciales de la cámara virtual. El resultado es válido para los parámetros extrínsecos, pero no lo es tanto para el cálculo de los intrínsecos. Por ello, en el Capítulo 5 se propone un método para obtener buenos resultados en la calibración de los parámetros intrínsecos, lo que permitirá entre otras cosas realizar un control visual con cámaras en las que se varíen los intrínsecos durante la tarea.

## 3.2. Formulación

---

Esta sección describe la notación y principales consideraciones acerca de la aproximación de control visual virtual desarrollada. En adelante,  $\mathbf{p}_d$  son las posiciones deseadas de las características en la imagen para el control visual virtual (es decir, las características observadas) y  $\mathbf{p}$  son las posiciones actuales de las características en la imagen (estas características son generadas a partir de los parámetros intrínsecos y extrínsecos estimados, como se describe en (3.1)).  $\xi$  representa los parámetros intrínsecos actuales (punto principal, focal) y  ${}^c\mathbf{M}_o$  son los parámetros extrínsecos (posición y orientación del sistema de coordenadas del objeto con respecto al sistema de coordenadas de la cámara).

La proyección de un punto del objeto en la imagen es:

$$\mathbf{p} = p r_{\xi} \left( {}^c \mathbf{M}_o \circ \mathbf{P} \right) \quad (3.1)$$

Para obtener la calibración del sistema, se debe minimizar progresivamente el error entre los datos observados,  $\mathbf{p}_d$ , y la posición de las mismas características calculadas mediante retroproyección utilizando los parámetros extrínsecos e intrínsecos actuales,  $\mathbf{p}$  (3.1). De esta forma, el error se define como:

$$\mathbf{e} = \mathbf{p} - \mathbf{p}_d \quad (3.2)$$

La derivada en el tiempo del error será:

$$\dot{\mathbf{e}} = \dot{\mathbf{p}} - \dot{\mathbf{p}}_d = \frac{\partial \mathbf{p}}{\partial \mathbf{r}} \frac{\partial \mathbf{r}}{\partial \mathbf{t}} + \frac{\partial \mathbf{p}}{\partial \xi} \frac{\partial \xi}{\partial \mathbf{t}} \quad (3.3)$$

Que se puede reescribir como:

$$\dot{\mathbf{e}} = \mathbf{H}_p \mathbf{V} \quad (3.4)$$

donde:

$$\mathbf{V} = \begin{pmatrix} \mathbf{T}_c \\ \dot{\xi} \end{pmatrix} \quad (3.5)$$

$$\mathbf{H}_p = \begin{pmatrix} \frac{\partial \mathbf{p}}{\partial \mathbf{r}} & \frac{\partial \mathbf{p}}{\partial \xi} \end{pmatrix} \quad (3.6)$$

Para hacer  $\mathbf{e}$  decrecer exponencialmente a 0,  $\dot{\mathbf{e}} = -\lambda_1 \mathbf{e}$ , se obtiene la siguiente acción de control:

$$\mathbf{V} = -\lambda_1 \mathbf{H}_p^+ \mathbf{e} \quad (3.7)$$

Por lo tanto, se van variando progresivamente los parámetros intrínsecos y extrínsecos, de forma que una vez que el error se anula, se obtienen los parámetros intrínsecos y extrínsecos reales de la cámara.

En la Tabla 3-1 se observa de un vistazo las diferencias existentes entre el control visual clásico y el control visual virtual.

	Control Visual Clásico	Control Visual Virtual
<b>Características iniciales</b>	Obtenidas por la cámara del robot inicialmente	Aproximación de los parámetros de la cámara virtual y retroproyección inicial.
<b>Características actuales</b>	Obtenidas por la cámara del robot en cada iteración (en cada iteración se necesita obtener una nueva imagen para aportar información)	Retroproyectadas por la cámara virtual con los parámetros de esta iteración (no es necesario obtener nueva imagen)
<b>Características deseadas</b>	Fijadas como objetivo por el usuario	Las obtenidas en la imagen real

Tabla 3-1. Diferencias entre el control visual clásico y el control visual virtual.

### 3.3. Resultados en la Calibración de una Cámara Usando Control Visual Virtual

Se ha empleado una cámara eye-in-hand que ha sido sometida a un proceso previo de calibración para estimar los parámetros intrínsecos usando un método clásico de calibración. Usando el método de Zhang [ZhangZ-1999] se han obtenido los siguientes parámetros intrínsecos: las coordenadas en píxeles del punto principal son  $(u_0, v_0) = (168.75, 121.5)$ , y la distancia focal en las direcciones  $u$  y  $v$  son  $(412.92, 423.69)$  respectivamente. Se han usado 10 imágenes para hacer la calibración.

Para la calibración usando la aproximación de control visual virtual se ha empleado el objeto 3D representado en la Figura 1 del que se pueden extraer 5 elipses.

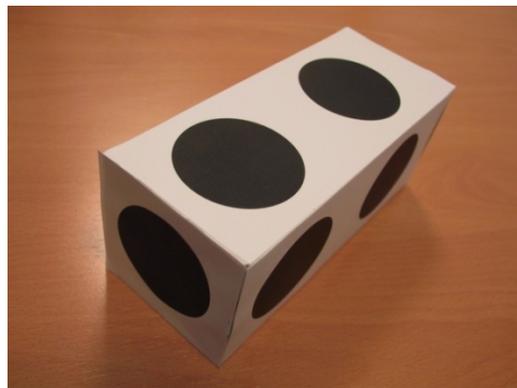


Figura 3-1. Objeto 3D para la calibración de la cámara

La calibración se lleva a cabo utilizando únicamente una imagen. Los parámetros intrínsecos iniciales considerados en la calibración son los provistos por el fabricante  $(f_u, f_v) = (313.17, 301)$  y  $(u_0, v_0) = (154, 99)$ . Con el objetivo de evaluar el comportamiento de la calibración se ha obtenido la Tabla 3-2. Esta tabla indica los parámetros intrínsecos obtenidos desde diferentes posiciones iniciales de la cámara virtual. La cámara real está situada en la posición  $(3.68, 39.766, 303.374)$  mm, y en la orientación según los ángulos de Euler  $(\alpha, \beta, \gamma) = (2.97, 0.86, 1.57)$ . Las trayectorias obtenidas durante los diferentes experimentos de calibración se representan en la Figura 3-2. En estos experimentos, se observa que los parámetros intrínsecos obtenidos varían según la posición inicial. Además, en la Tabla 3-2 se puede observar un experimento en el que los parámetros intrínsecos obtenidos están muy lejos de los reales (ver el último marcado en gris).

Posición inicial $(p_x, p_y, p_z, \alpha, \beta, \gamma)$	Parámetros intrínsecos obtenidos $(f_u, f_v), (u_0, v_0)$
116.47, 61.51, 252.8, 3.0522, 0.8165, 1.5206	(399.52, 404.06), (167.13, 120.68)
-61.93, 57.51, 252.8, 3.0522, 0.8165, 1.5206	(398.73, 403.05), (164.82, 120.65)
180.5, -10.49, 464.8, -2.9187, 0.0252, 1.7817	(406.59, 415.34), (167.27, 119.98)
-124.3, -5.69, 464.8, -2.8861, 0.9891, 1.8289	(411.46, 422.99), (168.16, 120.22)
78.07, 50.31, 171.2, 3.0522, 0.8165, 1.5206	(412.72, 424.73), (170.37, 120.79)
-29.13, 63.91, 202.4, -2.1540, 0.7652, 0.7585	(406.06, 413.13), (170.3, 120.91)
159.7, 119.9, 418.4, 2.5514, 0.4488, 1.9160	(411.67, 422.84), (169.68, 121.65)
34.07, 27.11, 160, 2.3255, 0.6177, 1.9713	(412.88, 423.28), (169.41, 121.03)
-130.7, 117.5, 418.4, 1.8827, 0.3372, 2.1547	(905.66, 1053.23), (-203.84, -149.26)

Tabla 3-2. Parámetros intrínsecos obtenidos en diferentes experimentos de calibración.

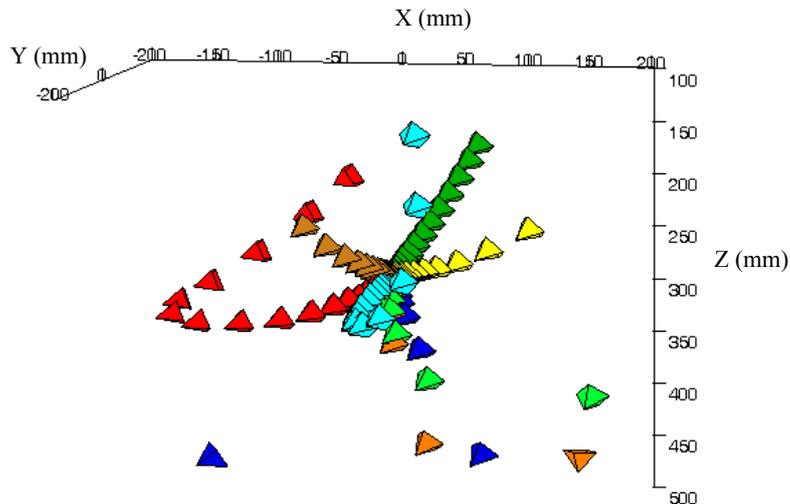


Figura 3-2. Trayectorias 3D en diferentes experimentos de calibración.

En las secciones siguientes se detallan dos de estas pruebas (ver Tabla 3-3):

- Prueba 1. Esta prueba funciona perfectamente, obteniendo parámetros de calibración cercanos a los reales. Sin embargo, estos resultados se deben mejorar.

- Prueba 2. Experimento en el que los parámetros extrínsecos e intrínsecos obtenidos no son correctos.

Prueba	Parámetros intrínsecos ( $f_u, f_v$ ), ( $u_0, v_0$ )
1	(406.59, 415.34), (167.27, 119.98)
2	(905.661, 1053.23), (-203.843, -149.264)

Tabla 3-3. Parámetros intrínsecos obtenidos en las pruebas.

### 3.3.1. Prueba 1

Para observar el comportamiento de la calibración mediante control visual virtual, se pasa a presentar una descripción detallada del tercer experimento representado en la Tabla 3-2. La Figura 3-3 muestra la evolución del error medio durante la tarea de control visual virtual. En este experimento se puede ver que el error decrece exponencialmente. La evolución de los parámetros intrínsecos se muestra en la Figura 3-4. Se representan también en las Figura 3-5 y en la Figura 3-6 las acciones de control y la trayectoria de la cámara durante la tarea, respectivamente. Los resultados de la calibración de la cámara son:  $(u_0, v_0) = (406.59, 415.34)$ , y la distancia focal en las direcciones  $u$  y  $v$  son  $(167.27, 119.98)$ . Se puede observar que los parámetros intrínsecos obtenidos están muy cercanos a los obtenidos usando los métodos de calibración clásicos off-line, sin embargo, aparecen errores importantes. Los parámetros extrínsecos obtenidos son: posición  $(9.95, 42.35, 302.3)$ mm, y orientación según los ángulos de Euler  $(\alpha, \beta, \gamma) = (2.98, 0.85, 1.57)$ .

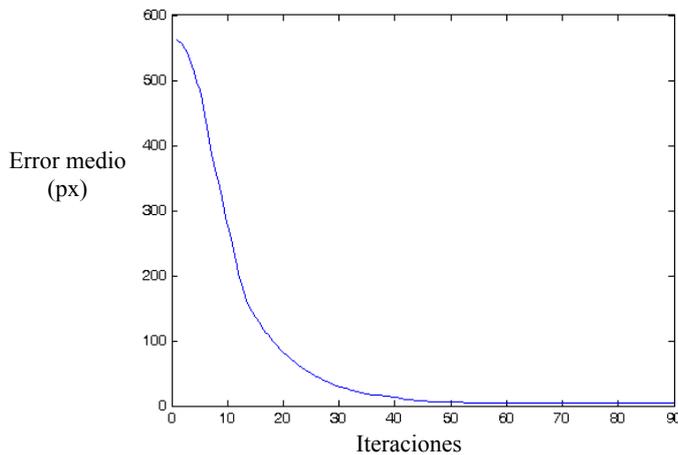


Figura 3-3. Error medio en píxeles. Prueba 1.

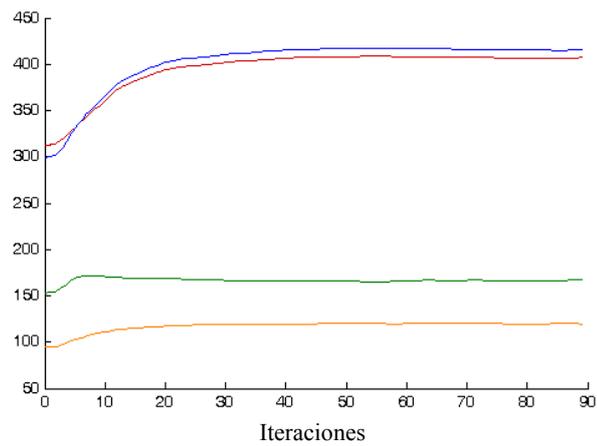


Figura 3-4. Evolución de los parámetros intrínsecos. Prueba 1.

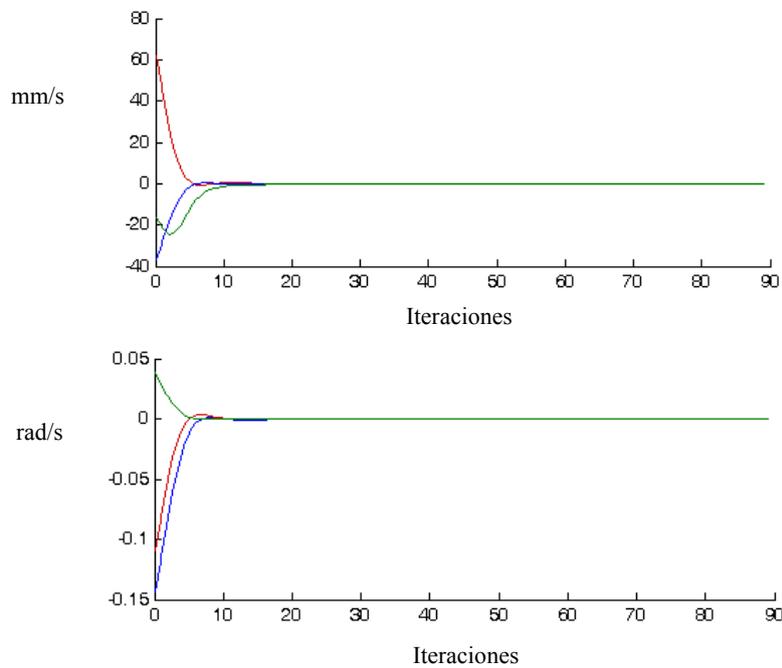


Figura 3-5. Velocidades traslacionales y rotacionales. Prueba 1.

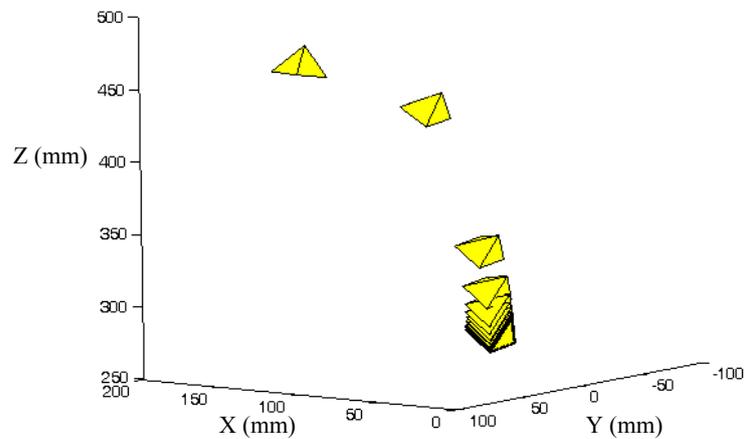


Figura 3-6. Trayectoria de la cámara. Prueba 1.

En este primer experimento se puede observar que el método es válido para el cálculo de los parámetros extrínsecos de la cámara. Sin embargo, el método tiene problemas en la correcta calibración de los parámetros intrínsecos. Para solucionar estos problemas de calibración en el Capítulo 5 se propone un método que permite obtener los parámetros intrínsecos a partir de más de una imagen, y que es una aportación de esta memoria de investigación.

### 3.3.2. Prueba 2

Con respecto a la convergencia de la calibración, se puede afirmar que el sistema es capaz de converger cuando aparecen grandes errores en la inicialización. Sin embargo, se ha observado que los parámetros intrínsecos obtenidos pueden variar dependiendo de los parámetros extrínsecos iniciales (ver Tabla 3-2). Por ejemplo, considerando los parámetros intrínsecos previamente mencionados, si la posición inicial de la cámara es (-130.7, 117.5, 418.4) y la orientación en ángulos de Euler (1.8827, 0.3372, 2.1547), una vez que se desarrolla la tarea de control visual virtual, los parámetros intrínsecos que se obtienen son:  $(u_0, v_0) = (-203.843, -149.264)$  y la distancia focal es igual a (905.661, 1053.23). Los parámetros extrínsecos obtenidos son: posición (-307.25 -250.71 -800.56)mm, y la orientación en términos de ángulos de Euler  $(\alpha, \beta, \gamma) = (2.9, 0.41, -1.82)$ . En la Figura 3-7 y la Figura 3-8 se representa la evolución del error medio y de los parámetros intrínsecos. También se representan en la Figura 3-9 y la Figura 3-10 las acciones de control y la trayectoria de la cámara, respectivamente.

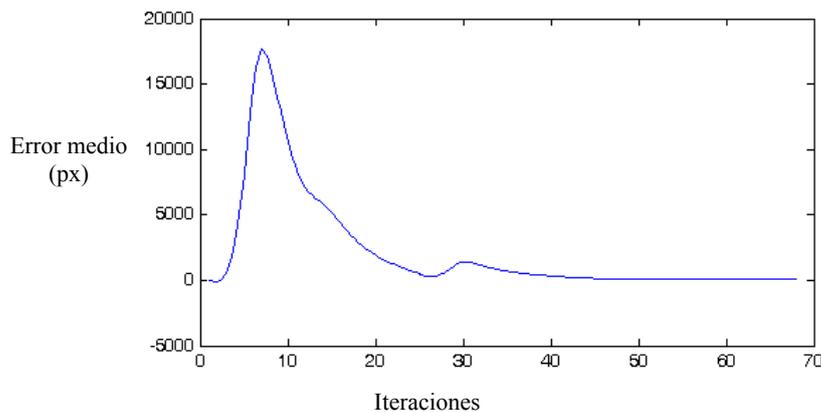


Figura 3-7. Error medio en píxeles. Prueba 2.

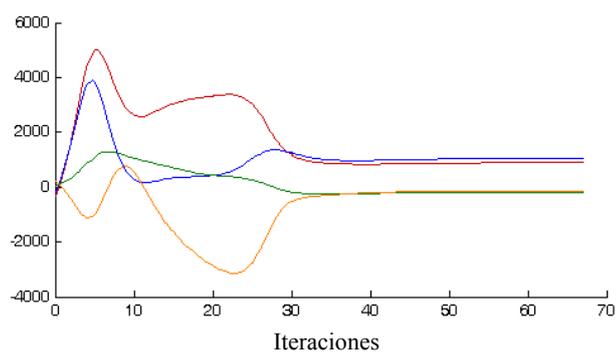


Figura 3-8. Evolución de los parámetros intrínsecos. Prueba 2.

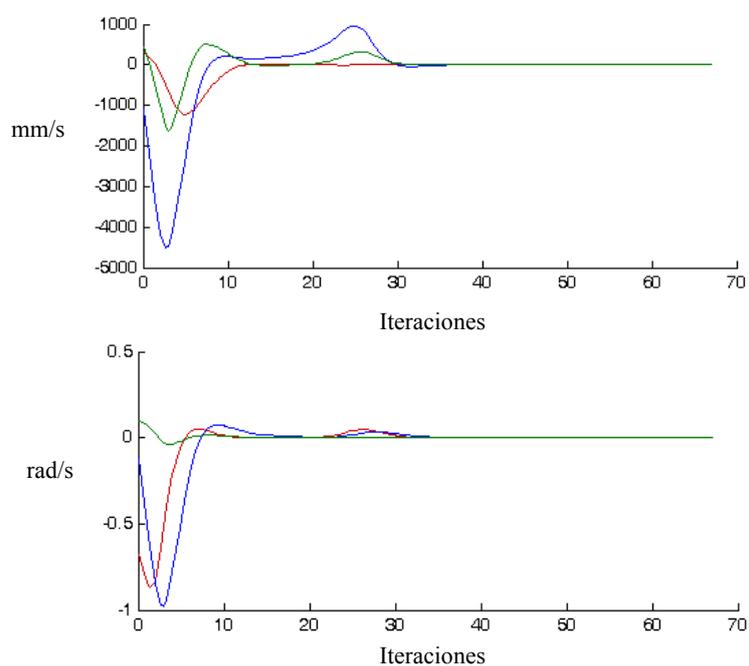


Figura 3-9. Velocidades traslacionales y rotacionales. Prueba 2.

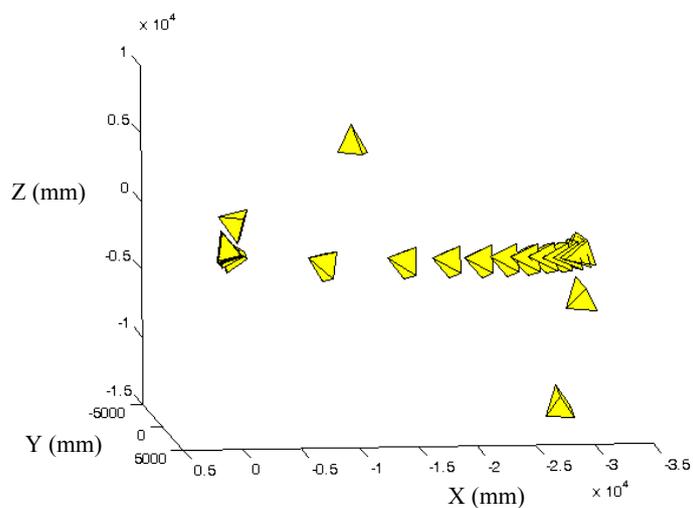


Figura 3-10. Trayectoria de la cámara. Prueba 2.

Como se observa en la Figura 3-8, durante la calibración pueden aparecer valores imposibles de los parámetros intrínsecos. En este momento la calibración debería acabar dando un error. Esto también ocurre en control visual clásico debido a mínimos locales. Estos problemas también se resuelven con el método que se propondrá en el Capítulo 5.

### 3.4. Aplicación del Control Visual Virtual para Posicionamiento Preciso de la Cámara en una Tarea de Ensamblado

El proceso para ensamblar de forma automática distintos elementos en una estructura requiere del uso de distintas técnicas. A continuación se describe una aplicación de ensamblado de una estructura formada por piezas de aluminio (Figura 3-11). Este experimento se enmarca dentro de mi participación en el proyecto del MCYT: “Diseño, Implementación y Experimentación de Escenarios de Manipulación Inteligentes para Aplicaciones de Ensamblado y Desensamblado Automático” (DPI2005-06222).

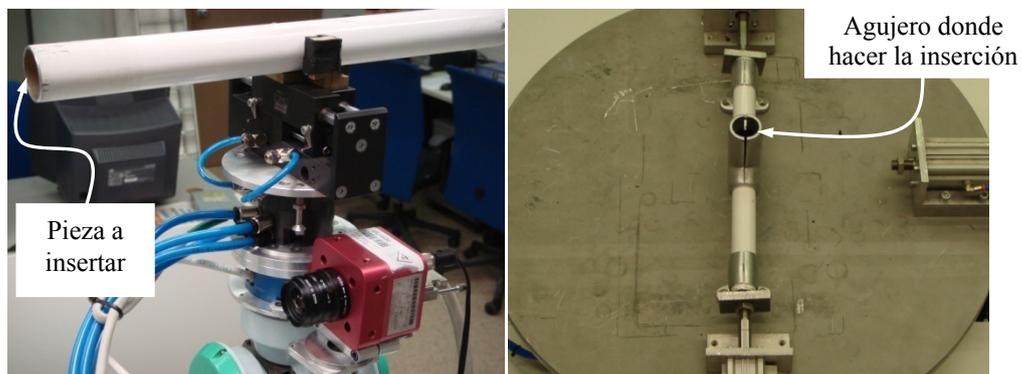


Figura 3-11: Estructura de aluminio base para la tarea de inserción.

Para realizar de forma flexible y satisfactoria la tarea, se hace necesario el empleo de técnicas como control visual con elipses, control visual virtual y control de fuerza. Se ha pretendido dotar a la aplicación de la suficiente flexibilidad para acometer la inserción en los diferentes tipos de piezas y en distintas posiciones. Por eso se deben realizar una serie de pasos previos a la inserción de la barra. Estos pasos previos están encaminados a posicionar el extremo de la barra justo sobre el eje de inserción de la pieza, de forma que un simple desplazamiento en la dirección de ese eje, unido a la acción del controlador de fuerza, permitan cumplir con éxito la tarea. Las distintas fases para la

correcta ejecución de la tarea se pueden ver en la Figura 3-12, donde se representa en un esquema un resumen del proceso.

El control visual con elipses permitirá posicionar de forma precisa el extremo del robot sobre el agujero de inserción. La flexibilidad que aporta sobre el control visual clásico por puntos lo hace en este caso indispensable, al no disponer de puntos característicos que extraer de la estructura. En este caso sólo se dispone del agujero de inserción como característica válida para realizar el control visual. Se trata del primer paso del proceso. Gracias a su uso, el robot se posicionará siempre a la misma distancia respecto del agujero de inserción, independientemente de la orientación de éste o su posicionamiento dentro de la mesa de trabajo.

Una vez que se tiene situada la cámara en la posición deseada sobre el agujero de inserción se debe calcular la posición del agujero respecto a la cámara. Esta posición servirá para, mediante traslaciones de los distintos sistemas de referencia, obtener la posición del extremo del robot que permita realizar la inserción. Para obtener esta posición de inserción el robot se debe situar orientando la barra según el eje  $z$  del agujero de inserción. Mediante control visual virtual [GarcíaGJ-2006] se obtiene la matriz de parámetros extrínsecos de la cámara. Esta matriz proporciona la posición del objeto de referencia en el control visual virtual (en este caso el círculo del agujero de inserción) con respecto a la cámara. Gracias a esta matriz se puede reconstruir la posición que tendrá el extremo del robot para proceder a la inserción en la dirección correcta. Esta posición se le pasa al robot separándose en el eje de inserción una distancia fija de 5cm. Esta distancia de seguridad servirá para activar el controlador de fuerza en una fase posterior.

La última fase es la propia de inserción. Para ello se activa el control de fuerza, dirigiendo la barra al mismo tiempo hacia el agujero de la pieza de ensamblado en la dirección del eje de inserción.

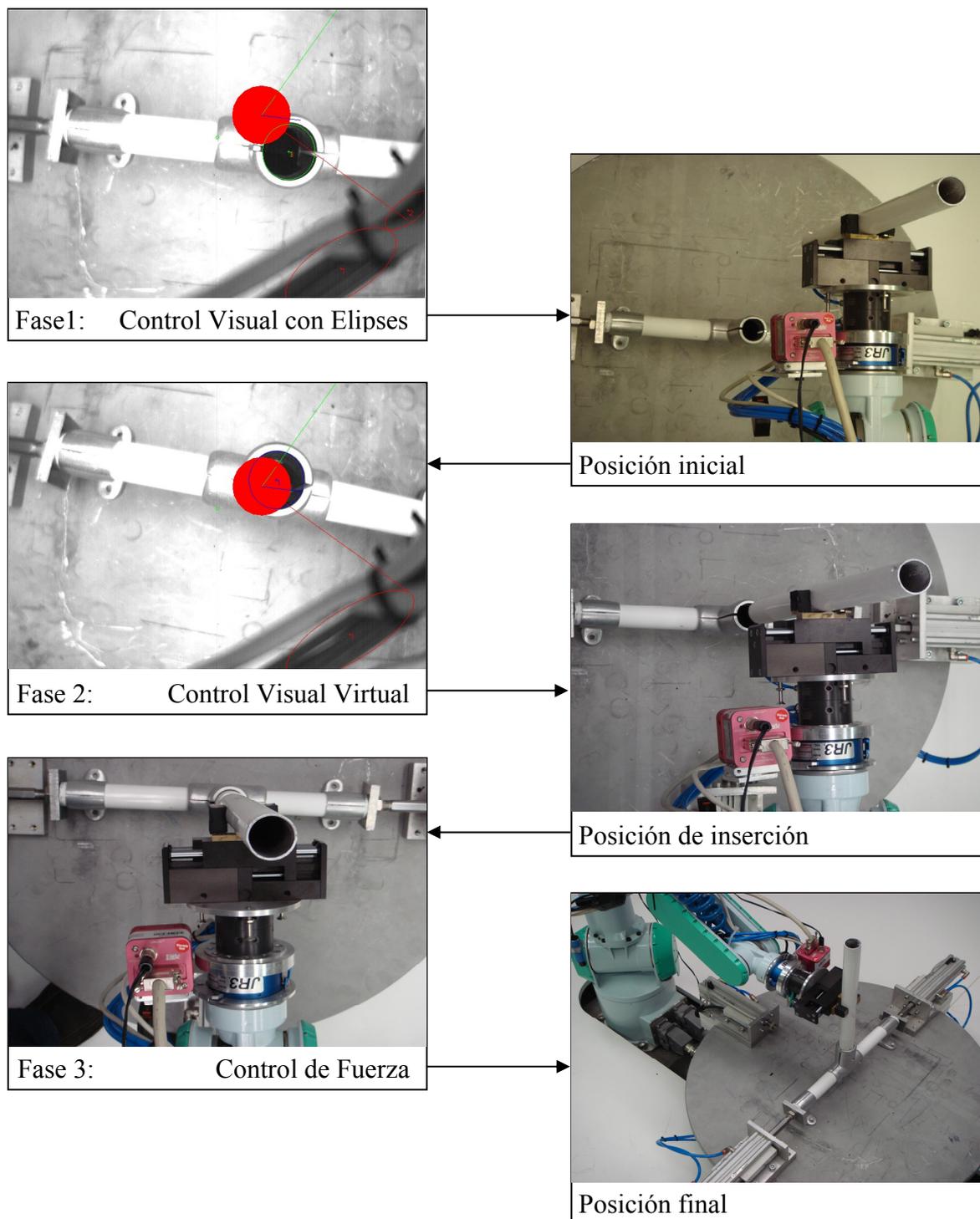


Figura 3-12: Esquema del proceso de inserción.

## 3.5. Conclusiones

---

En el presente capítulo se ha descrito el control visual virtual. Esta técnica permite calcular los parámetros extrínsecos e intrínsecos de una cámara a partir de una imagen capturada por ésta. Las pruebas realizadas permiten concluir que el método es válido para el cálculo de la posición de la cámara con respecto al objeto visto por ésta. Sin embargo, la calibración de los parámetros intrínsecos a partir de una imagen no es válida, ya que dependiendo de la posición inicial de la cámara virtual, se obtienen distintos valores de los intrínsecos. Además, durante la calibración aparecen valores de los intrínsecos que no son válidos.

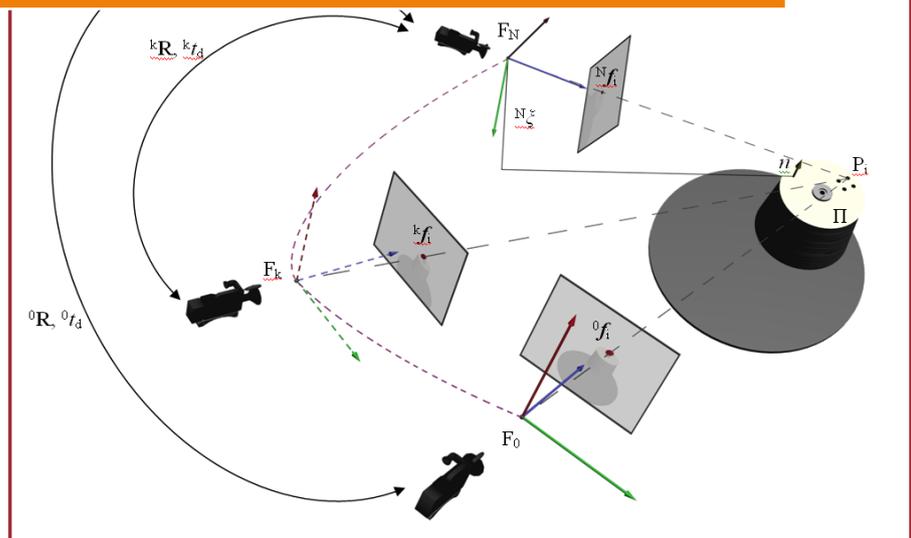
Para resolver los problemas detectados durante las pruebas de calibración, en el Capítulo 5 de la memoria se propone un método que se basa en el uso de más de una imagen para realizar la calibración. La integración de este método junto con el sistema de seguimiento de trayectorias propuesto en el siguiente capítulo permitirá realizar el seguimiento de objetos con cámaras que varíen sus parámetros intrínsecos durante la tarea de control visual.

Sin embargo, mediante control visual virtual tal y como se describe en este capítulo es posible encontrar los parámetros extrínsecos de la cámara a partir de una sola imagen. En el siguiente capítulo se describe un método de seguimiento intemporal de trayectorias en el espacio imagen que hace uso del control visual virtual precisamente para calcular la posición de la cámara con respecto al objeto que se pretende seguir.



# Capítulo 4

## Seguimiento de trayectorias en el espacio imagen



## 4.1. Introducción

En tareas en las que se desea posicionar un robot respecto a un elemento del entorno, la utilización de sistemas de control visual permite hacer uso de visión artificial para guiar el robot respecto a dicho elemento. El uso de un sistema de control visual basado en imagen frente a uno basado en posición ofrece distintas ventajas para este tipo de aplicaciones dentro de las cuales cabe mencionar su robustez frente a errores de modelado y perturbaciones externas [Hutchinson-1996]. Sin embargo, en este tipo de sistemas de control visual la convergencia generalmente sólo está garantizada localmente [Chaumette-1998].

Otro aspecto a tener en cuenta y que caracteriza a los sistemas de control visual clásicos basados en imagen es que se puede especificar la posición final del robot respecto a un objeto del entorno, sin embargo, no es posible definir la trayectoria que describirá el robot desde la localización inicial a la final en el espacio 3-D. Este efecto se muestra en la Figura 4-1. Mientras que en el espacio de la imagen la trayectoria de las características tiende a seguir una línea recta, en el espacio Cartesiano 3-D esta trayectoria es impredecible y a menudo describe una parábola entre las configuraciones inicial y final.

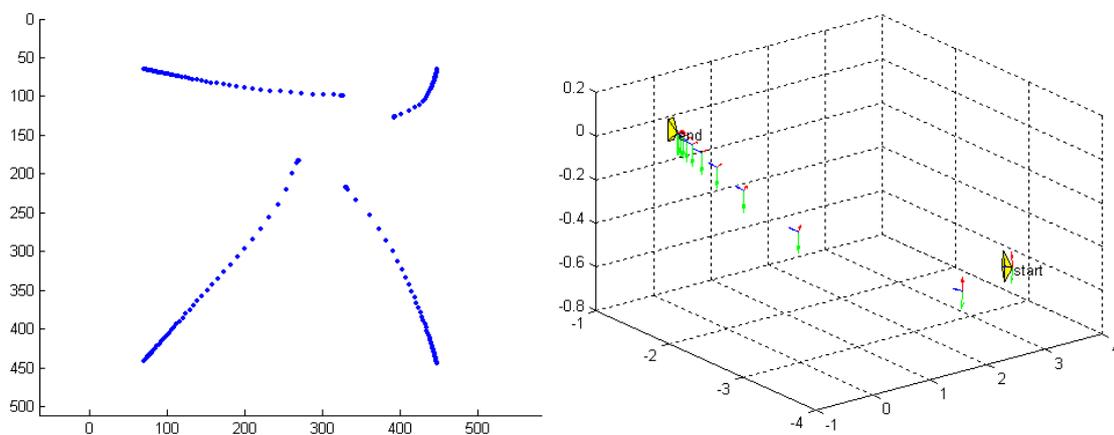


Figura 4-1. Ejemplo de tarea de control visual para el posicionamiento respecto a un objeto. a) Trayectoria de las características en la imagen. b) Trayectoria 3-D de la cámara.

Los sistemas de control visual han sido hasta la actualidad utilizados en un amplio rango de aplicaciones [Corke-1994]. La mayoría de éstas son aplicaciones punto a punto en las que se desea alcanzar una determinada configuración de las características deseadas en la imagen a partir de una configuración inicial. Así, en tareas en las que se desea posicionar un robot respecto a un elemento del entorno, el empleo de sistemas de control visual permiten hacer uso de visión artificial para guiar el robot respecto a dicho elemento.

Como se ha mostrado en distintos trabajos [Hashimoto-1994, Chaumette-1998] si las configuraciones inicial y final se encuentran muy alejadas es difícil asegurar la convergencia, más aún

si existen errores de modelado. Asimismo, haciendo uso de un sistema de control visual clásico basado en imagen no se puede especificar una trayectoria entre las configuraciones inicial y deseada. Únicamente se indica qué configuración se desea alcanzar y no qué trayectoria se desea seguir para llegar a la configuración deseada.

Este último aspecto constituye un elemento fundamental en cualquier aplicación en la que se desea hacer uso de visión artificial para guiar un robot en un determinado entorno de trabajo. En estas aplicaciones se desea guiar al robot de forma que se posicione respecto a un determinado objeto del entorno, pero a su vez se requiere que durante su posicionamiento se siga una trayectoria concreta que cumpla determinadas restricciones espaciales como la evitación de posibles colisiones con elementos del espacio de trabajo. Este ha sido uno de los principales motivos que ha propiciado la definición de un nuevo método de seguimiento de trayectorias, denominado seguimiento intemporal de trayectorias en la imagen basado en control visual virtual, que constituye una aportación descrita en este capítulo.

Como se ha descrito anteriormente, los sistemas de control visual se clasifican en basados en posición y basados en imagen [Hashimoto-1994, Hutchinson-1996]. El primer caso está basado en el cálculo de un error en el espacio Cartesiano 3-D, para lo que se requiere un modelo CAD perfecto del entorno implicado en la tarea de control visual y un modelo de cámara calibrado. Este tipo de sistemas son muy sensibles a errores de modelado y a perturbaciones externas. En los sistemas de control visual basados en imagen el error es determinado directamente en la imagen. Esta aproximación asegura una mayor robustez respecto a errores de modelado, pero en general se obtiene un movimiento no adecuado de la cámara en el espacio cartesiano. Por otro lado, la convergencia de los sistemas de control visual basados en imagen generalmente sólo está garantizada localmente [Soato-1996, Wunsch-1997], lo que permite asegurar una correcta convergencia si la configuración deseada de las características en la imagen se encuentran suficientemente cercanas a las actuales. En este capítulo se va a hacer uso de control basado en imagen como estrategia básica para realizar el seguimiento de trayectorias previamente generadas en el espacio de la imagen y, para confirmar la validez de los conceptos desarrollados, se ha aplicado a un sistema de cámara en el extremo de un robot.

Relativo a planificación de trayectorias en el espacio de la imagen existen algunos trabajos como [Cowan-1999, Cowan-2002, Hosada-1995, Zhang-2002, Remaizelles-2006]. En los dos primeros se emplean funciones de navegación como base para el seguimiento de trayectorias, mostrando una formulación que permite asegurar la convergencia globalmente. El tercer trabajo se trata de un generador de trayectorias aplicado a la evitación de obstáculos empleando sistemas de visión artificial estereoscópicos. En el cuarto artículo, Zhang y Ostrowski describen una aplicación de control visual a robots móviles en una situación donde la localización del robot no es conocida o lo es con muy poca precisión. Finalmente, Remaizelles propone una base de datos de imágenes del entorno para controlar un robot mediante la planificación on line de la trayectoria sin usar reconstrucción 3D.

Estas aproximaciones son empleadas para alcanzar unas características deseadas, evitando obstáculos y garantizando las restricciones de visibilidad, pero no para realizar el seguimiento de una trayectoria completa. Sólo recientemente es posible encontrar algún trabajo relativo a la utilización de sistemas de control visual para el seguimiento de trayectorias en la imagen [Chesi-2005, Malis-2004a, Mezouar-2002a]. Chesi propone un método que genera la trayectoria en imagen manteniendo una línea recta entre la posición inicial y final en el espacio Cartesiano 3D, manteniendo las características dentro de la imagen. En esta misma línea, Malis presenta un esquema de control visual capaz de seguir una línea recta en 3D con la ventaja añadida de no depender de la correcta calibración de los parámetros intrínsecos de la cámara. En el último artículo, en primer lugar se describe un método para generar trayectorias cumpliendo ciertas restricciones espaciales que posteriormente serán empleadas para realizar el control de un robot utilizando una estrategia de control basado en imagen. En [Park-2003] se describe un método para la generación de trayectorias en el espacio imagen consistente en simular la trayectoria que habría de describir el extremo del robot (la pinza en este caso) desde una localización inicial a la final. Una vez determinada la trayectoria se emplea un controlador como el descrito en [Mezouar-2002a] para realizar el seguimiento. El objetivo de este capítulo no es el de describir la planificación en el espacio de la imagen, y se hará la suposición de que se ha realizado una generación previa de la trayectoria que se desea seguir en la imagen. Tras disponer de esta información, se desarrollará la estrategia de seguimiento intemporal de trayectorias en la imagen basado en control visual virtual para realizar el seguimiento de la trayectoria deseada obteniendo un correcto desempeño en el espacio Cartesiano 3-D.

En [Mezouar-2002a] se muestra una aproximación consistente en un sistema de control visual basado en imagen para el seguimiento de una trayectoria en la imagen dependiente del tiempo. En contraste con la aproximación descrita por Mezouar y Chaumette, en [Pomares-2005a, Pomares-2004a] se emplea control visual basado en imagen para el seguimiento de trayectorias de forma independiente del tiempo empleando los denominados flujos de movimiento. Si se formula el seguimiento como una trayectoria en la imagen dependiente del tiempo, la configuración actual y la deseada se encontrarán siempre separadas por un intervalo de tiempo,  $\Delta t$ . Sin embargo, el hecho de alcanzar una determinada configuración de las características en un instante de tiempo determinado no es crítico comparado con la necesidad de mantener la trayectoria en la imagen. De esta manera, si un sistema de control visual basado en imagen es empleado para el seguimiento de trayectorias dependientes del tiempo, puede correr el riesgo de no seguir la trayectoria adecuada a costa de intentar mantener las restricciones temporales. Por otro lado, el hecho de que se alcancen unas características deseadas en un instante de tiempo determinado no garantiza el correcto seguimiento de la trayectoria. Pomares y Torres, en el denominado control visual basado en flujo de movimiento codifican la tarea a desarrollar por el robot en el espacio de configuraciones de la imagen, es decir, especificando una configuración deseada  $s_d$  para cada posible configuración  $^k s$ .

En el presente capítulo se detalla un método de seguimiento de trayectorias en el espacio imagen también independiente del tiempo. Las aportaciones de este método son las siguientes:

- Permite un seguimiento intemporal de la trayectoria en la imagen.
- Es capaz de mantener una velocidad constante durante el seguimiento.
- Tiene un comportamiento menos oscilatorio ante aumentos de la velocidad de seguimiento que los métodos previos.
- Asegura en todo momento que la posición 3D de la cámara representa una configuración válida.

Para conseguir el comportamiento intemporal, al igual que Pomares y Torres con el control visual basado en flujo de movimiento, se codifica la tarea en el espacio de configuraciones de la imagen [GarcíaGJ-2007b]. A partir del muestreo de la trayectoria en la imagen el sistema determina cuál debe ser la siguiente configuración deseada  $s_d$  para ser empleada por un sistema de control visual basado en imagen. El método propuesto en este capítulo trata de mantener una velocidad constante en el seguimiento de la trayectoria. Para ello, se busca la configuración  ${}^k s$  dentro de la trayectoria muestreada que supera esa velocidad deseada. El ajuste preciso de la velocidad se consigue generando una trayectoria 3D para la cámara entre las configuraciones  ${}^k s$  y  ${}^{k-1} s$ . Para ello se hace uso del control visual virtual, que permite obtener la posición y orientación de la cámara desde las que se observan esas dos configuraciones. El resultado es un método que permite realizar de forma intemporal el seguimiento de una trayectoria previamente definida en el espacio imagen.

A lo largo del capítulo se detallarán las características del sistema de seguimiento intemporal de trayectorias en la imagen basado en control visual virtual, mostrando la precisión alcanzada para el seguimiento de trayectorias tanto en el espacio imagen como en el espacio Cartesiano 3-D.

## 4.2. Generación de la trayectoria en el espacio imagen

A partir de una trayectoria del robot,  $\gamma(t)$ , generada con el algoritmo descrito en [Pomares-2004c, Pomares-2006d], se obtiene la trayectoria discreta en 3D,  $\tau$ , que debe seguir el robot para realizar satisfactoriamente la tarea. Esta trayectoria discreta es una secuencia de  $N$  valores discretos de  $\gamma(t)$ , cada uno representando una posición intermedia de la cámara  $\tau = \{ {}^k \gamma / k \in 1 \dots N \}$ . La trayectoria de las características del objeto implicado en la tarea de control visual en la imagen se obtiene a partir de esta trayectoria 3D,  $\tau$ . Esta trayectoria discreta en la imagen estará formada por  $M$  puntos, de forma

que en un instante  $k$ , se tendrá una determinada configuración  ${}^k\mathbf{s} = \{ {}^k\mathbf{f}_i / i \in 1 \dots M \}$ , donde  ${}^k\mathbf{f}_i$  representa la posición en el espacio imagen del punto característico  $i$  en el instante  $k$ . La trayectoria 3D queda de esta forma codificada en un vector de configuraciones  $T = \{ {}^k\mathbf{s} / k \in 1 \dots N \}$ .

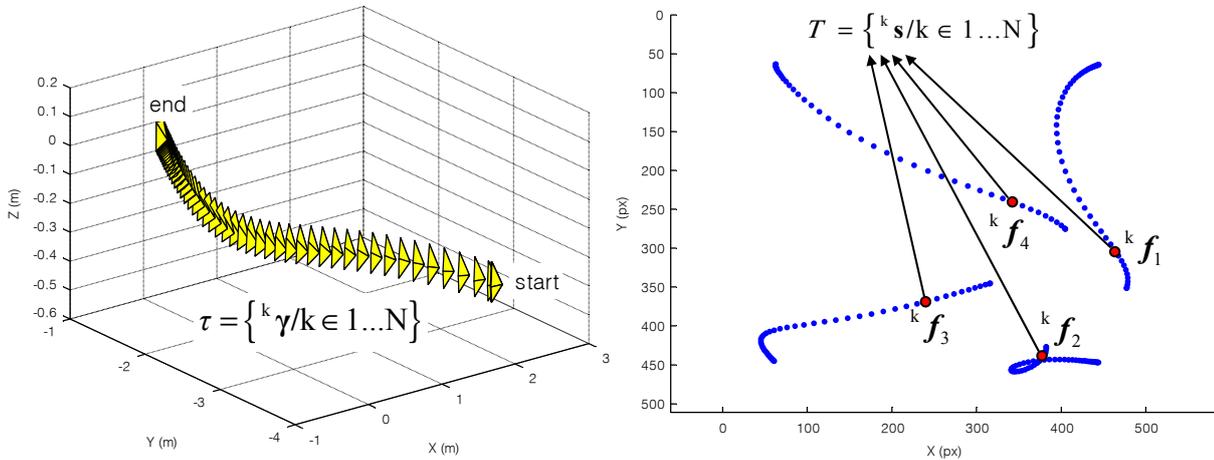


Figura 4-2. Trayectoria discreta  ${}^k\mathbf{f}_i$  de una característica en la imagen.

Dada una determinada característica, la evolución de  ${}^k\mathbf{f}_i$  (véase un ejemplo en la Figura 4-3 en la que se han considerado muestras de una imagen de 50 por 50) se corresponde con la trayectoria discreta que se desea que siga la característica en la imagen a lo largo de la tarea.

Este será el muestreo de la trayectoria deseada en la imagen que deberá ser seguida utilizando el sistema de seguimiento intemporal de trayectorias en la imagen basado en control visual virtual.

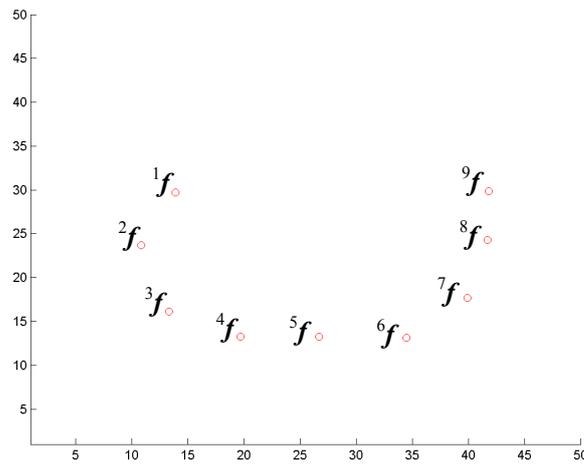


Figura 4-3. Trayectoria discreta  ${}^k\mathbf{f}_i$  de una característica en la imagen.

## 4.3. Control Visual basado en Flujo de Movimiento

En este apartado se va a describir la técnica de control visual basado en flujo de movimiento [Pomares-2005a]. Esta técnica permite realizar un seguimiento intemporal de trayectorias en el espacio imagen. En el presente capítulo, se describe un nuevo sistema de seguimiento intemporal de trayectorias en la imagen basado en control visual virtual. Este nuevo método trata de resolver los problemas encontrados en los sistemas de control visual basado en flujo de movimiento, al tiempo que aproveche las principales ventajas que aporta el método, como el seguimiento intemporal de las trayectorias.

El flujo de movimiento es un campo de vectores que converge hacia la trayectoria deseada en la imagen. Se obtiene utilizando trayectorias no dependientes del tiempo especificadas en el espacio de la imagen. El flujo de movimiento permite determinar la configuración deseada en la imagen a partir de una dada  $y$ , por lo tanto, realizar el seguimiento de la trayectoria independientemente del tiempo.

El flujo de movimiento,  $\Phi$ , se define como:

$$\Phi: \mathcal{S} \rightarrow T\mathcal{S} \quad (4.1)$$

donde  $\mathcal{S}$  es el espacio de la imagen y  $T\mathcal{S}$  es el espacio tangente a la imagen. Se observa que el flujo de movimiento define un vector tangente en cada punto de la imagen de manera que proporciona un vector de velocidad en la imagen deseada en cada configuración posible. El flujo de movimiento queda definido realizando el trazado de un conjunto de vectores que convergen a la trayectoria deseada en la imagen. En general, el flujo de movimiento tiene las dos siguientes propiedades:

- Sus valores en cada punto de la trayectoria deseada son tangentes a dicha trayectoria.
- Fuera de la trayectoria tienden a reducir errores en el seguimiento.

Por lo tanto, el flujo de movimiento es un campo de vectores que indica la dirección en la cual deben encontrarse las características deseadas empleadas como referencias por un sistema de control basado en imagen, permitiendo de esta manera el seguimiento de la trayectoria. Así, considerando un sistema de control basado en imagen, la velocidad a aplicar al robot respecto al sistema de coordenadas de la cámara será:

$$\mathbf{v}^C = -k \cdot \hat{\mathbf{J}}_f^+ \cdot \mathbf{e}_f \quad (4.2)$$

donde  $k$  es la ganancia del controlador proporcional;  $\hat{\mathbf{J}}_f^+$  es la pseudoinversa de la matriz de interacción,  $\mathbf{e}_f = \mathbf{s} - \mathbf{s}_d$ ,  $\mathbf{s} = [f_1, f_2, \dots, f_M]^T$  son las características extraídas en la imagen,  $\mathbf{s}_d = [f_1 + m_1 \Phi_1(f_1), f_2 + m_2 \Phi_2(f_2), \dots, f_M + m_M \Phi_M(f_M)]^T$ ,  $\Phi_i$  es el flujo de movimiento para la característica  $i$ ,

y  $\mathbf{m}=\{m_1, m_2, \dots, m_M\}$  determina la velocidad de progresión, de forma que modificando el valor de  $\mathbf{m}$  se obtendrá un seguimiento más o menos rápido.

Para definir el flujo de movimiento se va a considerar que, para una característica  $i$ , se desea seguir una trayectoria parametrizada en la imagen  $\mathbf{f}_{id}:\Gamma \rightarrow \mathfrak{S}$  donde  $\Gamma \subset \mathfrak{R}$ , de forma que si  $\Gamma$  fuera el tiempo se tendría la trayectoria clásica en la imagen dependiente del tiempo. Considerando que  $\mathbf{f}_i$  son las coordenadas de la característica  $i$  en la imagen y que las coordenadas del punto más cercano en la trayectoria deseada son  $\mathbf{f}_{ind}=(f_{xind}, f_{yind})$ , se define el vector de error  $\mathbf{E}_i(\mathbf{f}_i)=(E_{xi}, E_{yi})$  donde  $E_{xi}=(f_{xi}-f_{xind})$  y  $E_{yi}=(f_{yi}-f_{yind})$ . A partir de este error se define la función de potencial  $U_i:\mathfrak{R}^n \rightarrow \mathfrak{R}$  calculada según se describe posteriormente.  $U_i$  aumenta según  $\mathbf{f}_i$  se desvía de su localización deseada  $\mathbf{f}_{ind}$  y alcanza su mínimo cuando el error es cero.

A partir las funciones anteriores se define el flujo de movimiento para una característica  $i$ ,  $\Phi_i$ , como una combinación lineal de dos términos:

$$\Phi_i(\mathbf{f}_i) = \begin{pmatrix} \Phi_{xi}(\mathbf{f}_i) \\ \Phi_{yi}(\mathbf{f}_i) \end{pmatrix} = G_1(\mathbf{f}_i) \cdot \begin{pmatrix} \frac{\partial f_{xid}(\tau)}{\partial \tau} \\ \frac{\partial f_{yid}(\tau)}{\partial \tau} \end{pmatrix} - G_2(\mathbf{f}_i) \cdot \begin{pmatrix} \frac{\partial U_i}{\partial E_{xi}} \\ \frac{\partial U_i}{\partial E_{yi}} \end{pmatrix} \quad (4.3)$$

donde  $G_1, G_2: \mathfrak{S} \rightarrow \mathfrak{R}^+$  son funciones de peso de forma que  $G_1 + G_2 = 1$ . El primer término de la expresión (4.3),  $\dot{\mathbf{f}}_{id}(\tau)$ , se obtiene calculando los coeficientes de Taylor de la trayectoria deseada.

Como se muestra en [Li-2001], fruto de la aplicación del campo de velocidad codificado en  $\Phi_i$ , la evolución esperada para una característica  $i$ , y considerando un valor de  $\beta > 0$ , será:

$$\dot{\mathbf{f}}_i = \beta \cdot \Phi_i \quad (4.4)$$

Con lo que  $\mathbf{E}_i$  variará según:

$$\dot{\mathbf{E}}_i = -\beta \cdot G_2(\mathbf{f}_i) \cdot \frac{\partial U_i}{\partial \mathbf{E}_i} \quad (4.5)$$

Por lo tanto, el error evoluciona en la dirección del gradiente negativo del potencial y convergerá en un punto de  $U_i$  en el que  $E_{xi}=E_{yi}=0$ , con lo que  $\mathbf{f}_i \rightarrow \mathbf{f}_{id}$ .

A continuación se describe la función de potencial empleada por el flujo de movimiento para una única característica, por lo tanto, por claridad, se omite el subíndice que indica que característica se está considerando.

La función de potencial  $U$  puede ser utilizada para medir la precisión en el seguimiento, por lo tanto, debe alcanzar su valor más bajo cuando el error  $\mathbf{E}=\mathbf{f}-\mathbf{f}_{nd}$  sea cero y aumentar según  $\mathbf{f}$  se desvíe de su localización deseada  $\mathbf{f}_{nd}$ , siendo  $\mathbf{f}$  la característica extraída en la imagen y  $\mathbf{f}_{nd}$  la característica en

$f_d(\tau)$  más cercana a  $f$ . Este aspecto habrá de considerarse a la hora de definir la función de potencial y, para ello, se va a considerar como  $\mathbf{I}$  la imagen deseada o imagen que se obtendría tras representar la trayectoria deseada.

El primer paso para determinar la función de potencial es calcular el gradiente de  $\mathbf{I}$ . Para ello se utiliza la combinación de operaciones de morfología matemática consistentes en la sustracción de la erosión,  $\varepsilon$ , de la imagen  $\mathbf{I}$  de la dilatación,  $\delta$ , de  $\mathbf{I}$  [Serra-1981]. Tanto en la erosión como en la dilatación se ha empleado un elemento estructurante cuadrado de tamaño 3x3. Como resultado de estas operaciones se obtiene la imagen gradiente  $\mathbf{I}_g$ :

$$\mathbf{I}_g = \psi(\mathbf{I}) = \delta(\mathbf{I}) - \varepsilon(\mathbf{I}) \quad (4.6)$$

En la Figura 4-4.a se muestra un ejemplo de imagen  $\mathbf{I}$  que representa la trayectoria deseada de una característica, que se corresponde con la representada en la Figura 4-3. En la Figura 4-4.b se muestra la imagen resultado de calcular su gradiente,  $\mathbf{I}_g$ .

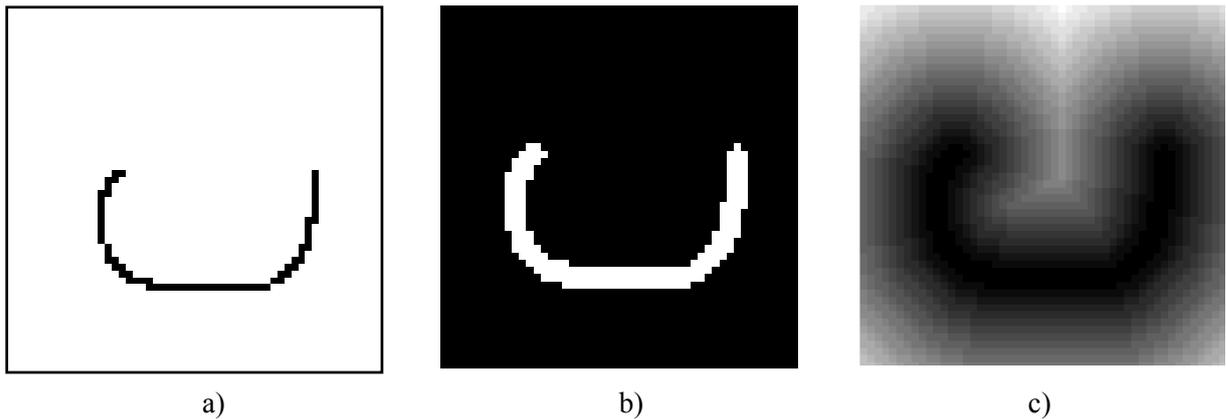


Figura 4-4. a) Trayectoria deseada en la imagen  $f_d(\tau)$  para una característica. b) Imagen gradiente. c) Mapa de distancias.

Una vez obtenida la imagen  $\mathbf{I}_g$ , el siguiente paso para determinar la función de potencial consiste en generar un mapa de distancias, que crea una imagen distancia  $\mathbf{I}_d$  a partir de la imagen binaria  $\mathbf{I}_g$ . El valor de  $\mathbf{I}_d$  en el píxel  $\mathbf{x}$  es la distancia Euclídea de  $\mathbf{x}$  al complemento de  $\mathbf{I}_g$ , es decir, la distancia de  $\mathbf{x}$  al punto más cercano del complemento de  $\mathbf{I}_g$  [Lotufo-2001]:

$$\mathbf{I}_d = \Psi_{B_c}(\mathbf{I}_g) = \sum_i \varepsilon_{i, B_c}(\mathbf{I}_g) \quad (4.7)$$

donde  $\varepsilon$  lleva a cabo la erosión de la imagen  $\mathbf{I}_g$  haciendo uso de un elemento estructurante cuadrado de tamaño 3x3,  $B_c$ . El mapa de distancias  $\mathbf{I}_d$  es una imagen que almacena en cada píxel información relativa a su menor distancia hasta  $f_d(\tau)$  y, por lo tanto, puede ser empleada como función de error. En la Figura 4-4.c se muestra el mapa de distancias para la trayectoria de ejemplo de la Figura 4-3. A partir de esta última figura es posible generar una representación tridimensional del mapa de distancias (Figura 4-5) donde el valor de la coordenada  $z$  representa la distancia de cada píxel hasta el píxel más

cercano dentro de la trayectoria deseada. En esta figura se puede observar que, para cada píxel de la imagen, es posible determinar el píxel más cercano en la trayectoria deseada realizando un descenso por gradiente. Como se observa en esta figura, durante el descenso por gradiente nunca se producirán mínimos locales.

A partir de la representación del mapa de distancias de la Figura 4-5 se obtiene una representación de líneas equipotenciales (Figura 4-6.b) que muestra la distribución de la función de potencial  $U$  obtenida.

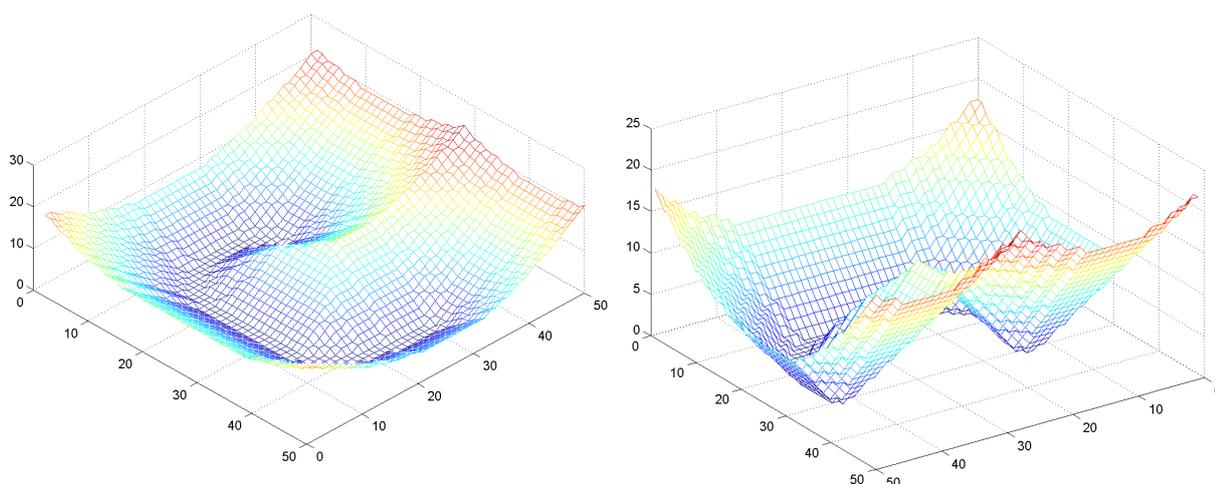


Figura 4-5. Representación tridimensional del mapa de distancias.

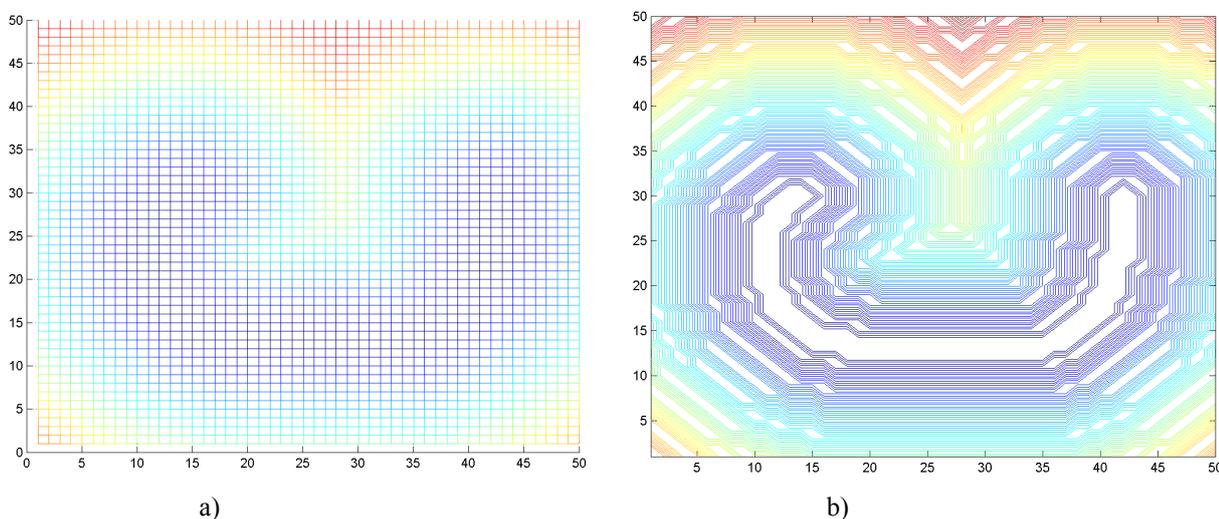


Figura 4-6. a) Vista superior del mapa de distancias de la Figura 4-5. b) Líneas equipotenciales.

La Figura 4-6.a es la vista superior del mapa de distancias tridimensional representado en la Figura 4-5. Se observa que esta vista se corresponde con las líneas de potencial mostradas en la Figura 4-6.b.

A partir de  $U$  se obtienen las componentes  $x$  e  $y$  del gradiente numérico bidimensional. Este gradiente crea una matriz de vectores que apuntan en la dirección en la que los valores de  $U$  aumentan.

Por lo tanto, si se representa  $-\nabla U$  para el caso de la trayectoria de ejemplo mostrada en la Figura 4-3, se obtendrá el campo de vectores representado en la Figura 4-7 que dirige cada punto de la imagen hacia el píxel más cercano en  $f_d(\tau)$ .

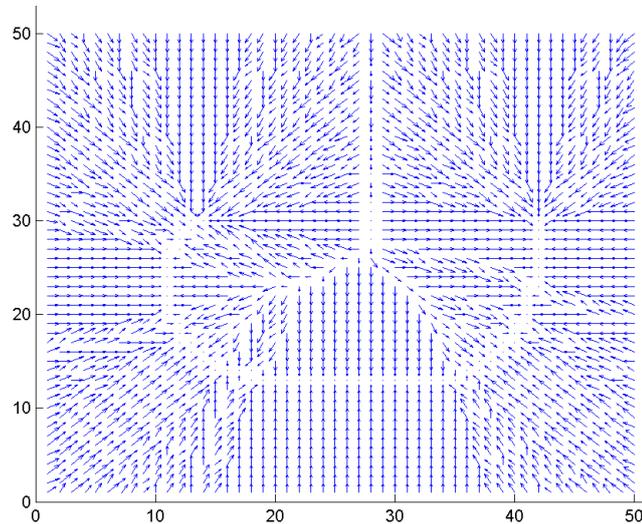


Figura 4-7. Campo de vectores correspondiente al gradiente negativo de la función de potencial.

Una consideración aparte debe realizarse con la velocidad de progresión. Hasta ahora se ha considerado un conjunto de  $M$  características de forma que cada una de las cuales debe seguir una trayectoria deseada en la imagen. Sin embargo, cada una de las  $M$  trayectorias debe progresar coordinadamente, de manera que las trayectorias más cortas reduzcan su velocidad para adaptarse a las más largas. Este es el principal problema del flujo de movimiento. No se tratan las características visuales del objeto en su conjunto. Para resolver este problema, se incluyen en el método una serie de mecanismos para coordinar las velocidades de cada una de las  $M$  trayectorias. Estos mecanismos impiden realizar el seguimiento con una velocidad constante. Además, al intentar aumentar la velocidad de seguimiento, se introduce mucha oscilación debido a que el sistema debe rectificar el error cometido al pasarse de la trayectoria deseada y tener que esperar la rectificación en el resto de trayectorias. Por ello, en las siguientes secciones se presentan distintas propuestas para mejorar el método de control visual basado en flujo de movimiento. Se trata de obtener un método de seguimiento intemporal de trayectorias en el plano imagen que unifique las características visuales tratándolas como un todo. De esta forma, se pretende a su vez conseguir mantener una velocidad constante durante el seguimiento de la trayectoria, disminuyendo la oscilación en torno a la trayectoria de referencia.

## 4.4. Seguimiento Intemporal de Trayectorias en el Espacio Imagen Basado en la Norma de la Matriz de Homografía 2D

Como se ha explicado en el apartado anterior, el control visual basado en flujo de movimiento trata cada característica por separado. Este es, sin duda, el principal problema del método, ya que no permite garantizar teóricamente que en un momento dado la configuración de las características en la imagen se corresponda con una posición 3D válida. Para solucionarlo, en una primera aproximación, se va a intentar agrupar las características mediante la matriz de homografía 2D. Esta matriz relaciona puntos 3D situados sobre un plano en dos imágenes obtenidas por distintas cámaras o por la misma cámara desde distintas posiciones de ésta (ver Figura 4-8).

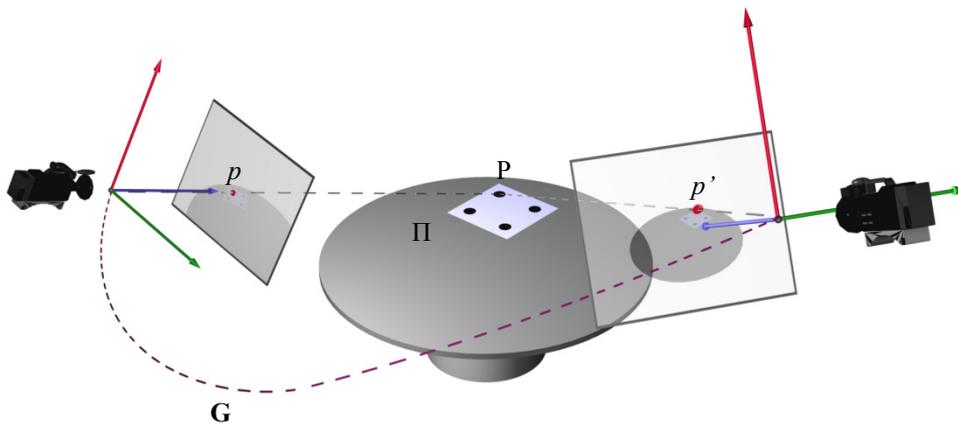


Figura 4-8. Homografía 2D.

Si la cámara capta dos imágenes de un mismo plano  $\Pi$ , existe una matriz que relaciona los puntos del plano vistos en una de las imágenes con los mismos puntos vistos en la otra imagen. Sea  $P$  un punto situado en el plano  $\Pi$ , dicho punto se proyecta en la primera imagen en la posición  $p$  y en la segunda imagen en  $p'$ . La homografía proyectiva se define como:

$$\mu p = G p' \quad (4.8)$$

La matriz de homografía 2D se obtiene a partir de la expresión (4.8) si se tienen al menos cuatro puntos coplanares [Hartley-2000].

Mediante la matriz de homografía se consigue agrupar las características observadas del objeto. Este era el primer paso a resolver, ya que con control visual basado en flujo de movimiento se tenía un excelente controlador para seguir la trayectoria predefinida de una característica en concreto.

Sin embargo, al tener que seguir todas las trayectorias al mismo tiempo, se debe introducir un mecanismo que frene aquella trayectoria que va más rápida.

A continuación se describe el método desarrollado a partir de esta matriz de homografía 2D para seguir trayectorias previamente definidas en el plano imagen (tal y como se ha explicado en el apartado 4.2).

El primer paso es construir el vector de matrices de homografía 2D correspondiente a la trayectoria deseada,  $\mathbf{M} = \{^k \mathbf{G} / k \in 1 \dots N - 1\}$ . A partir de la trayectoria deseada,  $\mathbf{T}$ , se tiene la posición en imagen de los puntos característicos para cada posición  $k$  de la cámara,  $^k \mathbf{s}$  (véase apartado 4.2). Seleccionando como base siempre la primera posición de la trayectoria,  $^1 \mathbf{s}$ , se calculan las matrices de homografía 2D para el resto de muestras de la trayectoria, obteniendo así el vector de matrices de homografía que codifica la trayectoria deseada.

Se ha conseguido de esta forma agrupar las características en una matriz. Sin embargo, la matriz de homografía 2D es una matriz de  $3 \times 3$ , donde se tienen 8 gdl, y por lo tanto se podría entender que no se ha conseguido el objetivo que se deseaba. Hace falta un valor que represente de forma lo más precisa posible la información almacenada en la matriz de homografía.

$$\mathbf{G} = \begin{pmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & 1 \end{pmatrix} \quad (4.9)$$

Para resolver el problema, se hace uso de una norma definida para este tipo de matriz de homografía 2D. Esta norma, se calcula de la siguiente forma:

$$\|\mathbf{G}\| = k_1 \cdot \sqrt{g_{13}^2 + g_{23}^2} + k_2 \cdot \sqrt{g_{31}^2 + g_{32}^2} + k_3 \cdot (g_{11} \cdot g_{22} - g_{21} \cdot g_{11}) \quad (4.10)$$

Para su construcción se ha tenido en cuenta lo que representa cada componente  $g_{ij}$  de la matriz de homografía  $\mathbf{G}$ . El primer término del sumando engloba una traslación, así como el segundo. El tercer término engloba una transformación afín, así como el escalado y la rotación. Las constantes  $k_1$ ,  $k_2$ ,  $k_3$  representan el peso que se le quiera dar a cada componente dependiendo de la trayectoria que se quiera codificar. Para los ejemplos que se muestran posteriormente se han utilizado los siguientes valores:

$$\begin{aligned} k_1 &= 1 \\ k_2 &= 1000000 \\ k_3 &= 50 \end{aligned} \quad (4.11)$$

Dado que se tiene codificada la trayectoria deseada como un vector de matrices de homografía 2D, se calcula también el vector que almacena la norma de cada una de estas matrices.

El siguiente paso del método es transformar el espacio de la imagen a lo que se ha denominado espacio de las normas. El espacio de normas (ver Figura 4-9) es una figura donde el eje Y representa la norma de la matriz de homografía, y el eje X representa la posición en el vector de matrices de homografía que codifica la trayectoria deseada,  $M$ .

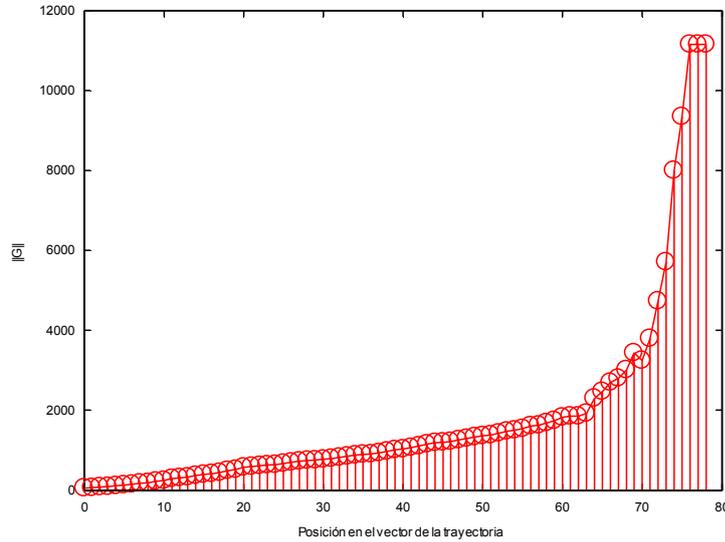


Figura 4-9. Espacio de normas de las matrices de homografía 2D.

El objetivo del sistema es mantener una velocidad constante durante el seguimiento. Para ello, la velocidad que se aplicará al robot viene dada por la ley de control visual basada en imagen:

$$v = -\lambda \cdot \hat{J}^+ \cdot (s - Gs_1) \quad (4.12)$$

donde  $\lambda$  es la ganancia del controlador proporcional y  $\hat{J}^+$  es la pseudoinversa de la matriz de interacción.

Se quiere relacionar la variación en la velocidad con variaciones en el producto  $Gs_1$ , que permitiría reducir el error en la velocidad modificando la posición de las características en la imagen. Para ello, se calcula la matriz de interacción que relaciona variaciones en la velocidad con variaciones en  $Gs_1$ :

$$\frac{\partial v}{\partial Gs_1} = \lambda \hat{J}^+ \quad (4.13)$$

La inversa de esta matriz permitirá obtener valores de variaciones en las características en la imagen con variaciones en la velocidad de seguimiento:

$$\frac{\partial Gs_1}{\partial v} = \frac{1}{\lambda} \hat{J} \quad (4.14)$$

Dado un error en velocidad, aplicando (4.14) se obtiene la variación de las características,  $\Delta s$ , en la imagen necesaria para anular dicho error.

Estas variaciones de las características para mantener la velocidad no permitirían seguir la trayectoria deseada, ya que son variaciones en una dirección arbitraria de la cámara, pero que mantienen la velocidad constante que se desea. Para llevar esas variaciones en imagen hacia la trayectoria deseada se calcula la nueva matriz de homografía  $G'$  que resulta de relacionar la primera posición de la trayectoria,  ${}^1s$  con la posición resultante de sumar las variaciones obtenidas a la posición actual  $s+\Delta s$ .

A partir de la imagen obtenida en la iteración actual del seguimiento, se calcula la matriz de homografía que relaciona los puntos en la primera posición de la trayectoria,  ${}^1s$ , con los puntos en la imagen adquirida,  $s$ .

A continuación, se calcula la norma de estas dos matrices de homografía  $G$  y  $G'$ . El sistema utilizará la diferencia de estas dos normas,  $\|G'\|-\|G\|$ , para obtener una medida de hacia dónde se debe mover el sistema en el espacio de normas para seguir la trayectoria deseada.

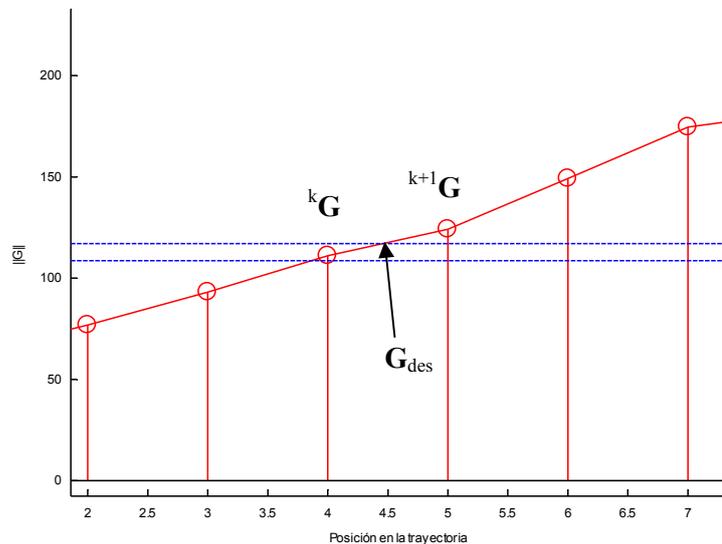


Figura 4-10. Espacio de normas de las matrices de homografía 2D.

Tal como muestra la Figura 4-10, con el error en norma, se generan las dos rectas de error y se avanza hacia la siguiente posición en la trayectoria. La posición de las características en la imagen se obtienen a partir de la matriz de homografía  $G_{des}$  que se obtiene de interpolar entre la última matriz visitada de la trayectoria deseada,  ${}^kG$ , y la siguiente,  ${}^{k+1}G$ . De esta forma, se asegura que las características en la imagen están dentro de la trayectoria deseada, aun manteniendo la velocidad deseada.

Para interpolar entre dos matrices de homografía se interpolan sus elementos linealmente. Esto permite obtener un comportamiento lineal de las características en la imagen tal como muestra la Figura 4-11. Este resultado se obtuvo tras probar la interpolación de la descomposición de las transformaciones proyectivas que generan la matriz de homografía.

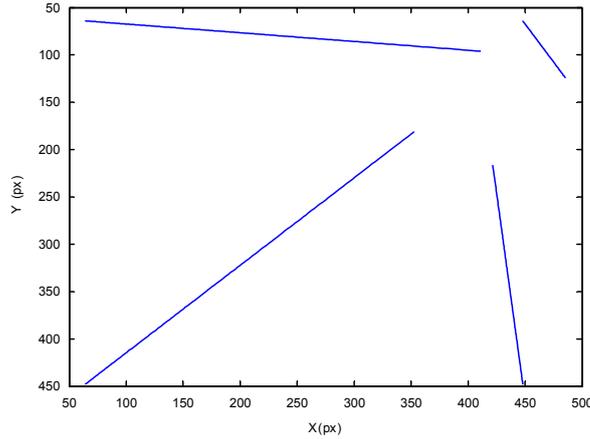


Figura 4-11. Evolución lineal de las características en la interpolación lineal de los elementos de las matrices de homografía 2D.

La descomposición de transformaciones proyectivas permite obtener la matriz de homografía **H** como resultado de distintas transformaciones proyectivas:

$$\mathbf{G} = \mathbf{G}_S \mathbf{G}_A \mathbf{G}_P = \begin{bmatrix} s\mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{K} & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{v}^T & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{t} \\ \mathbf{v}^T & 1 \end{bmatrix} \quad (4.15)$$

Esta descomposición es única si se elige un escalado,  $s$ , positivo;  $\mathbf{K}$  es una matriz triangular superior que cumple que su determinante es 1;  $\mathbf{R}$  es la matriz de rotación generada a partir del ángulo  $\theta$  sobre el eje  $\mathbf{Z}$  de la cámara; y  $\mathbf{t}$  es un vector que almacena las traslaciones de la cámara en el plano  $\mathbf{XY}$ . A partir de esta descomposición se intentó interpolar linealmente cada uno de los parámetros que aparecen en esta descomposición:  $s$ ,  $\theta$ ,  $t_x$ ,  $t_y$ ,  $v_x$ ,  $v_y$ ,  $k_{11}$ ,  $k_{12}$ ,  $k_{22}$ . El resultado, tal y como muestra la Figura 4-12 es una evolución de las características en la imagen no deseada, ya que se trata de parábolas que se pueden salir del plano imagen. Por eso se ha elegido finalmente la interpolación lineal de cada elemento de la matriz de homografía por separado, lo que permite obtener una evolución lineal de las características entre las dos homografías.

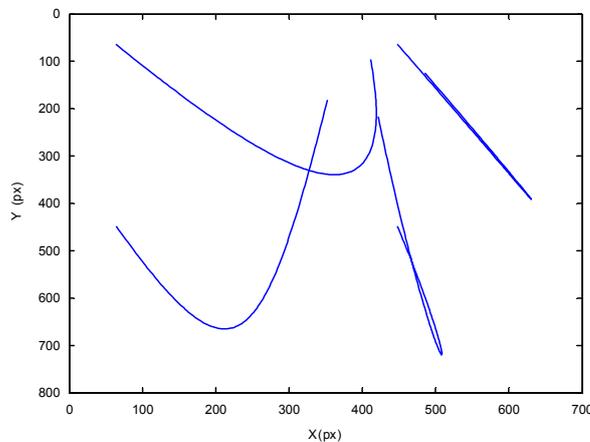


Figura 4-12. Evolución no lineal de las características en la interpolación lineal de la descomposición de transformaciones proyectivas de las matrices de homografía 2D.

Para comprobar el correcto funcionamiento del sistema se va a probar el sistema propuesto de seguimiento intemporal de trayectorias en la imagen basado en la norma de la matriz de homografía 2D para seguir la trayectoria mostrada en la Figura 4-13. La evolución en el espacio de las normas se representa en la Figura 4-14.

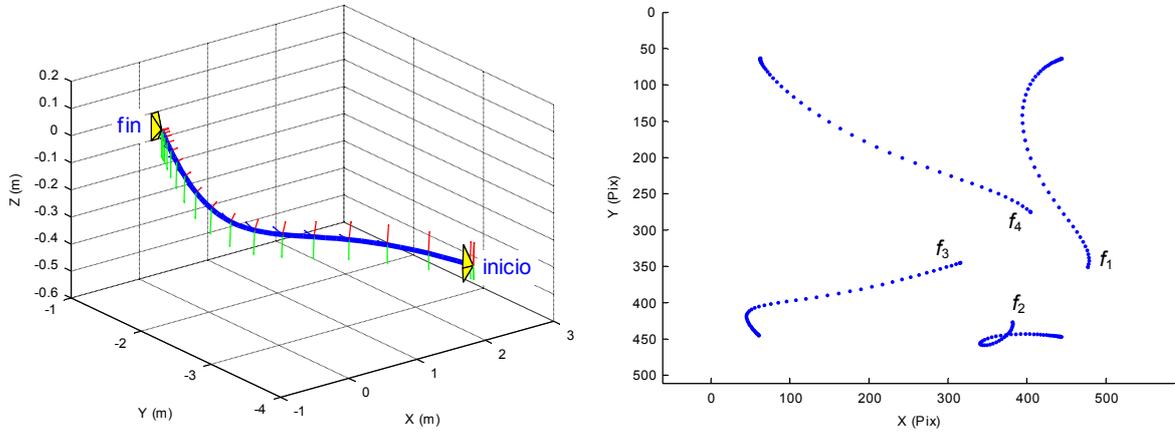


Figura 4-13. Muestreo de la trayectoria deseada en el espacio cartesiano 3-D y en la imagen.

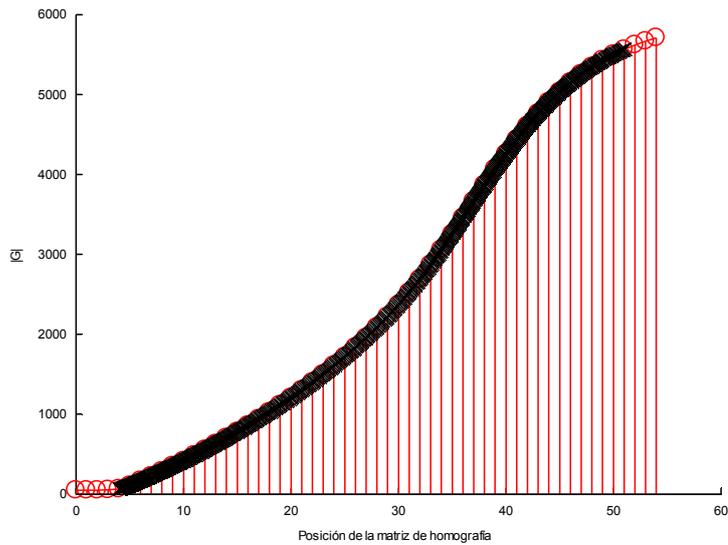


Figura 4-14. Evolución del seguimiento en el espacio de las normas.

Lo primero que se observa es que el seguimiento en imagen es el correcto, como se puede ver en la Figura 4-15.

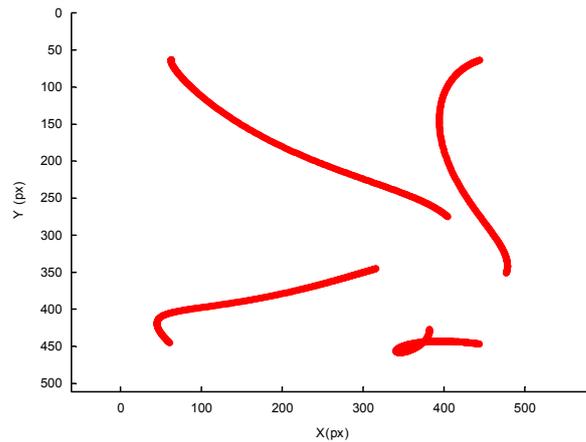


Figura 4-15. Evolución de las características en la imagen tras el seguimiento con el método basado en la norma de la matriz de homografía 2D.

El comportamiento es válido si se observa la trayectoria de la cámara en el espacio 3D (ver Figura 4-16). Además, se mejora el comportamiento del seguimiento con flujo de movimiento, como se podrá comprobar en los resultados mostrados en el apartado siguiente.

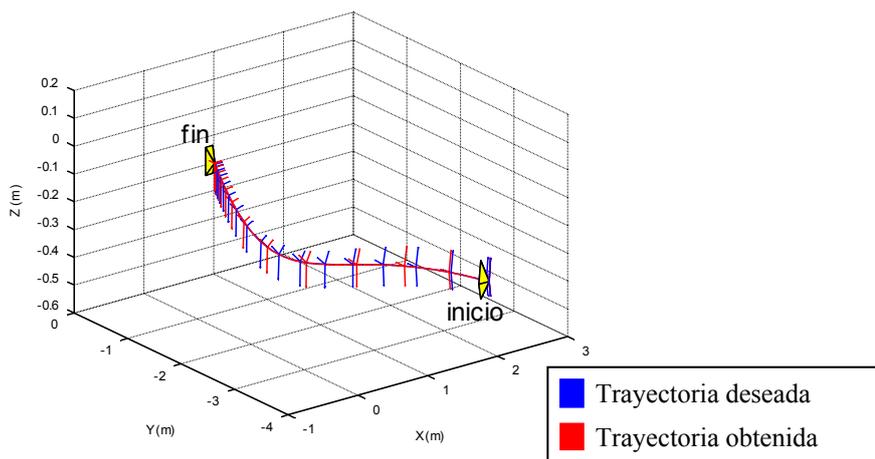


Figura 4-16. Evolución de la cámara en el espacio cartesiano 3D tras el seguimiento con el método basado en la norma de la matriz de homografía 2D.

Sin embargo, como podremos comprobar en la Figura 4-17, la velocidad no se ha conseguido controlar al valor deseado como se esperaba. Para el seguimiento se ha introducido una velocidad deseada de 2,75 mm/s. Sin embargo, en la Figura 4-17.b se puede ver que no se consigue controlar el módulo de la velocidad a los 2,75 mm/s. Si bien en un principio parece que el sistema tiende a alcanzar esa velocidad, poco a poco ésta disminuye hasta que el sistema termina el seguimiento.

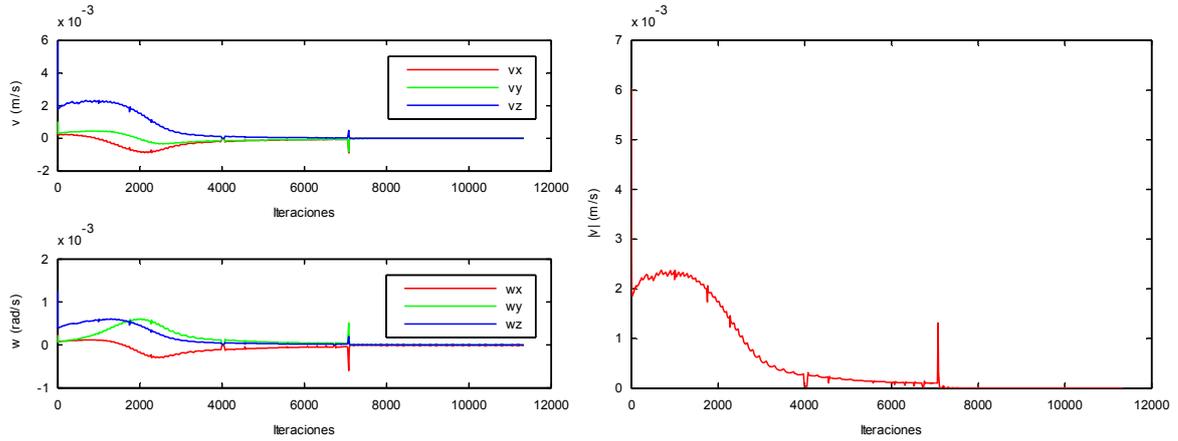


Figura 4-17. Evolución de la velocidad enviada a la cámara tras el seguimiento con el método basado en la norma de la matriz de homografía 2D: a) velocidad lineal y rotacional. b) módulo de la velocidad.

El comportamiento ha sido el correcto en el seguimiento en imagen y en el espacio 3D, sin embargo no se ha conseguido mantener una velocidad constante durante el seguimiento. Dado que con el primer experimento no se ha conseguido el objetivo que se buscaba de controlar la velocidad durante el seguimiento, se prueba el sistema en el seguimiento de otra trayectoria en la imagen. La trayectoria probada ahora es la mostrada en la Figura 4-18.

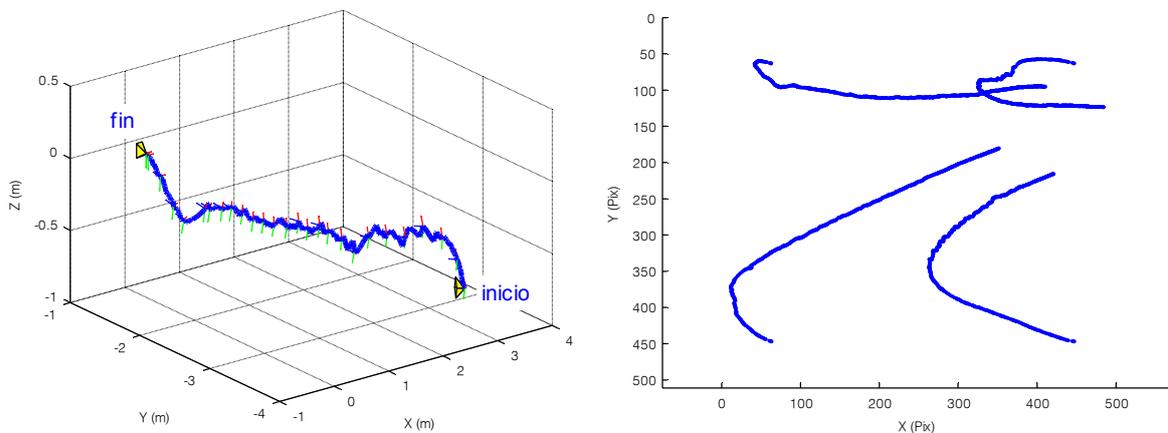


Figura 4-18. Muestreo de la trayectoria deseada en el espacio cartesiano 3-D y en la imagen.

El comportamiento del sistema al intentar seguir esta trayectoria en la imagen empeora. El sistema se para en mitad del seguimiento, obteniendo únicamente un seguimiento parcial de la trayectoria como se puede ver en la Figura 4-19. Llegado a un punto del espacio de normas, el sistema es incapaz de variar las características en la imagen y se queda parado. El sistema es incapaz de variar la velocidad para obtener variaciones en el espacio imagen de las características, con lo que no se puede continuar con el seguimiento. En la prueba anterior el resultado en imagen era satisfactorio, sin embargo, ahora se muestra que el método no es válido tal como está estructurado para el seguimiento de trayectorias en la imagen.

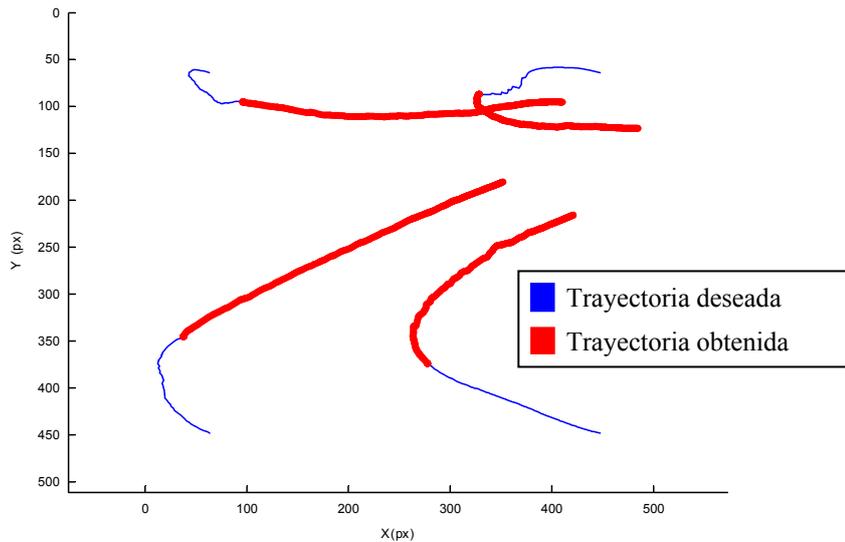


Figura 4-19. Evolución de las características en la imagen tras el seguimiento con el método basado en la norma de la matriz de homografía 2D.

Sin embargo, hasta el momento de pararse, el sistema consigue realizar de forma precisa el seguimiento de la trayectoria en la imagen y en el espacio 3D (ver Figura 4-20).

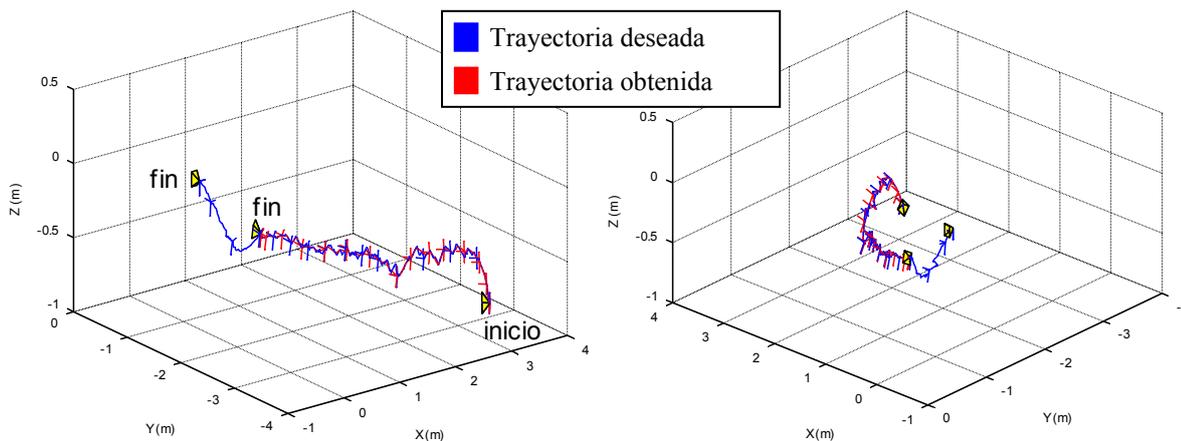


Figura 4-20. Evolución de la cámara en el espacio cartesiano 3D tras el seguimiento con el método basado en la norma de la matriz de homografía 2D.

Por último, un análisis de la evolución del módulo de la velocidad durante el seguimiento permite concluir que el sistema no consigue controlar de forma precisa la velocidad del sistema. La Figura 4-21 muestra esta evolución del módulo cuando se ejecuta el seguimiento con 1,75 y con 2,75 mm/s como módulo de la velocidad deseada. Sí se observa que el controlador consigue mantenerse en un entorno de la velocidad deseada, y que responde subiendo la velocidad cuando se aumenta la señal de control. Sin embargo, la oscilación excesiva hace que el método no termine de completar los objetivos marcados en un principio.

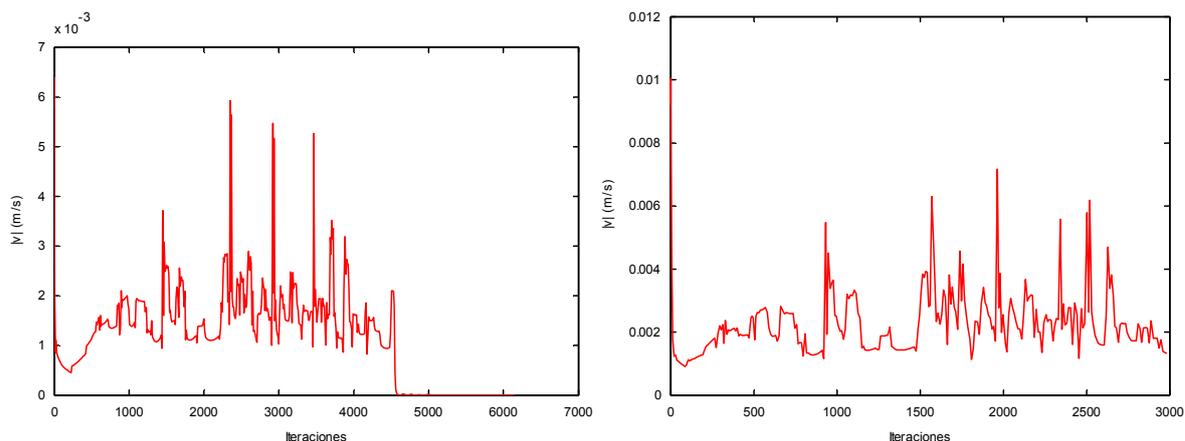


Figura 4-21. Evolución del módulo de la velocidad enviada a la cámara tras el seguimiento con el método basado en la norma de la matriz de homografía 2D: a) velocidad deseada de 1,75 mm/s. b) velocidad deseada de 2,75 mm/s.

Las pruebas realizadas muestran que la aproximación mediante la norma de la matriz de homografía 2D para el seguimiento intemporal de trayectorias en el espacio imagen, no resuelve los problemas encontrados en el control visual basado en flujo de movimiento. Si bien se consigue un seguimiento más preciso que en el flujo de movimiento, el sistema no asegura el seguimiento completo de la trayectoria. Además, no se ha conseguido mantener la velocidad constante que se desea durante el seguimiento. Por ello, en el siguiente apartado se presenta un método que si resuelve finalmente los problemas descritos, realizando de forma precisa el seguimiento de trayectorias previamente definidas en el espacio imagen, y manteniendo una velocidad constante durante este seguimiento.

## 4.5. Seguimiento Intemporal de Trayectorias Basado en Control Visual Virtual

En este apartado se describirá el método propuesto para el seguimiento intemporal de trayectorias en la imagen [GarcíaGJ-2007b]. Este método propuesto basado en control visual virtual permite el seguimiento de trayectorias previamente muestreadas en el espacio imagen tal y como se describe en el Apartado 4.3.

El control visual basado en imagen clásico no permite realizar el seguimiento de una trayectoria previamente definida en el espacio imagen. En una tarea de control visual basado en imagen, la evolución en el espacio imagen de una característica determinada tiende a seguir una línea recta. Por ello, es necesario definir un método que permita realizar el seguimiento en imagen de una

trayectoria predefinida. Si se formula el seguimiento como una trayectoria en la imagen dependiente del tiempo [Mezouar-2002a], la configuración actual y la deseada se encontrarán siempre separadas por un intervalo de tiempo,  $\Delta t$ . Sin embargo, si un sistema de control visual basado en imagen es empleado para el seguimiento de trayectorias dependientes del tiempo, puede correr el riesgo de no seguir la trayectoria adecuada a costa de intentar mantener las restricciones temporales. Por otro lado, el hecho de que se alcancen unas características deseadas en un instante de tiempo determinado no garantiza el correcto seguimiento de la trayectoria. El método propuesto en este apartado tiene un comportamiento intemporal. El control visual basado en flujo de movimiento descrito en el apartado 4.3 también tiene este comportamiento intemporal. Sin embargo, el método propuesto permite realizar el seguimiento de la trayectoria con una velocidad constante. Además, y tal y como se mostrará en el apartado 4.6 de resultados, la trayectoria seguida por este método propuesto es mucho más precisa, aún cuando se requiere una velocidad mayor, lo que en el control visual basado en flujo de movimiento provocaría un resultado muy oscilante. El método propuesto trata de resolver los problemas encontrados en el control visual basado en flujo de movimiento para obtener un método de seguimiento intemporal de trayectorias en el espacio imagen mejorado.

A continuación se detallan los pasos necesarios para realizar el seguimiento intemporal de una trayectoria previamente definida en el espacio imagen con el método propuesto basado en control visual virtual.

Tal como se describe en el apartado 4.2, en primer lugar es necesario muestrear en el espacio imagen la trayectoria seguida por el robot en el espacio Cartesiano 3-D. Esta trayectoria muestreada será la trayectoria que ha de seguir el robot para completar la tarea. Los valores en la imagen de cada una de las  $M$  trayectorias (correspondientes a cada una de las  $M$  características) en los instantes  $k$  serán  $C_i = \{^k f_i / k \in 1 \dots N\}$ . En cada instante  $k$  de la trayectoria deseada se tendrá de esta forma un vector de  $M$  características  $^k s = \{^k f_i / i \in 1 \dots M\}$ . Así, la trayectoria deseada quedará codificada en un vector de  $N$  posiciones para las  $M$  características  $T = \{^k s / k \in 1 \dots N\}$ .

Una vez obtenida la trayectoria deseada en la imagen  $T$ , el siguiente paso para el seguimiento de esta trayectoria será decidir qué tipo de seguimiento se quiere realizar. Atendiendo a la posición inicial de la cámara y dependiendo de la tarea a realizar, se puede desear un seguimiento completo de la trayectoria deseada (este sería el caso de corte de piezas) o simplemente acercarse desde esa posición inicial a la posición más cercana dentro de la trayectoria deseada para, a partir de ese momento, seguir la trayectoria hasta el final.

El algoritmo de seguimiento intemporal de trayectorias descrito en el presente apartado, recibirá el dato de cuál es la posición inicial dentro de la trayectoria deseada  $T$ . Para obtener este dato, se requiere conocer el tipo de seguimiento a desarrollar. En el caso que se quiera realizar el

seguimiento completo de la trayectoria deseada independientemente de la posición inicial de la cámara, se pasará al algoritmo de seguimiento la primera posición. Si lo que se desea es obtener la posición más cercana a la trayectoria deseada, se tendrá que realizar una búsqueda de esta posición. Para conseguir esta posición, se busca entre todas las posibles posiciones de  $T$  aquella configuración  ${}^k\mathbf{s}$  que permita llegar desde la posición inicial de la cámara a ella con la mínima velocidad.

Considerando un sistema de control visual basado en imagen, la velocidad a aplicar al robot respecto al sistema de referencia de la cámara será:

$${}^k\mathbf{v} = -\lambda \cdot \hat{\mathbf{J}}_f^+ \cdot \mathbf{e}_f \quad (4.16)$$

donde  $\lambda$  es la ganancia del controlador proporcional;  $\hat{\mathbf{J}}_f^+$  es la pseudoinversa de la matriz de interacción,  $\mathbf{e}_f = \mathbf{s} - {}^k\mathbf{s}$  y  $\mathbf{s} = [f_1, f_2, \dots, f_M]^T$  son las características extraídas en la imagen.

El algoritmo de búsqueda de la posición inicial en la trayectoria deseada a partir de la posición inicial de la cámara se puede ver esquematizado en la Figura 4-22.

A continuación se describe el algoritmo que permite realizar el seguimiento intemporal de trayectorias en el espacio imagen. En la Figura 4-23 se muestra un esquema de este algoritmo.

Tras obtener la posición inicial en la trayectoria deseada  $T$ , se comienza la búsqueda dentro de esa trayectoria de la configuración  ${}^k\mathbf{s}$  que está lo suficientemente alejada de la posición actual de la cámara como para enviar una velocidad al robot superior a la deseada. Se fija como  $|\mathbf{v}_d|$  el módulo de la velocidad deseada durante el seguimiento de la trayectoria. A partir de la posición inicial calculada según el algoritmo descrito en la Figura 4-22 se itera en busca de la configuración  ${}^k\mathbf{s}$  que, a partir de (4.16) (y tomando  $\mathbf{s}$  de la imagen captada por la cámara en la posición actual), permita superar en módulo la velocidad deseada.

Si se indicara este  ${}^k\mathbf{s}$  como referencia para la ley de control visual definida en (4.16), se tendría un controlador que permitiría realizar un seguimiento intemporal de la trayectoria deseada. Esto es así porque la posición de las características deseadas no dependen del tiempo, sino de la ubicación de las actuales. La información de la trayectoria a seguir está codificada en la imagen, y por tanto, no depende del tiempo. El resultado de realizar el seguimiento con este controlador no sería, sin embargo, satisfactorio. El problema está en que no se mantendría la velocidad durante el seguimiento de la trayectoria, ya que las distintas posiciones muestreadas de la trayectoria  $T$  no están separadas la misma distancia ni en imagen ni en 3-D.

Para conseguir mantener el módulo de la velocidad durante el seguimiento se necesita encontrar la posición 3-D entre dos muestras de la trayectoria deseada,  $T$ , que, al aplicar la ley de control visual basado en imagen clásica (considerando  $\mathbf{s}_d$  las características proyectadas en la imagen

desde esta posición), permita obtener exactamente esa velocidad deseada. Para ello, se introduce un bloque más al algoritmo, que se encargará de obtener las características deseadas en la imagen para mantener constante el módulo de la velocidad.

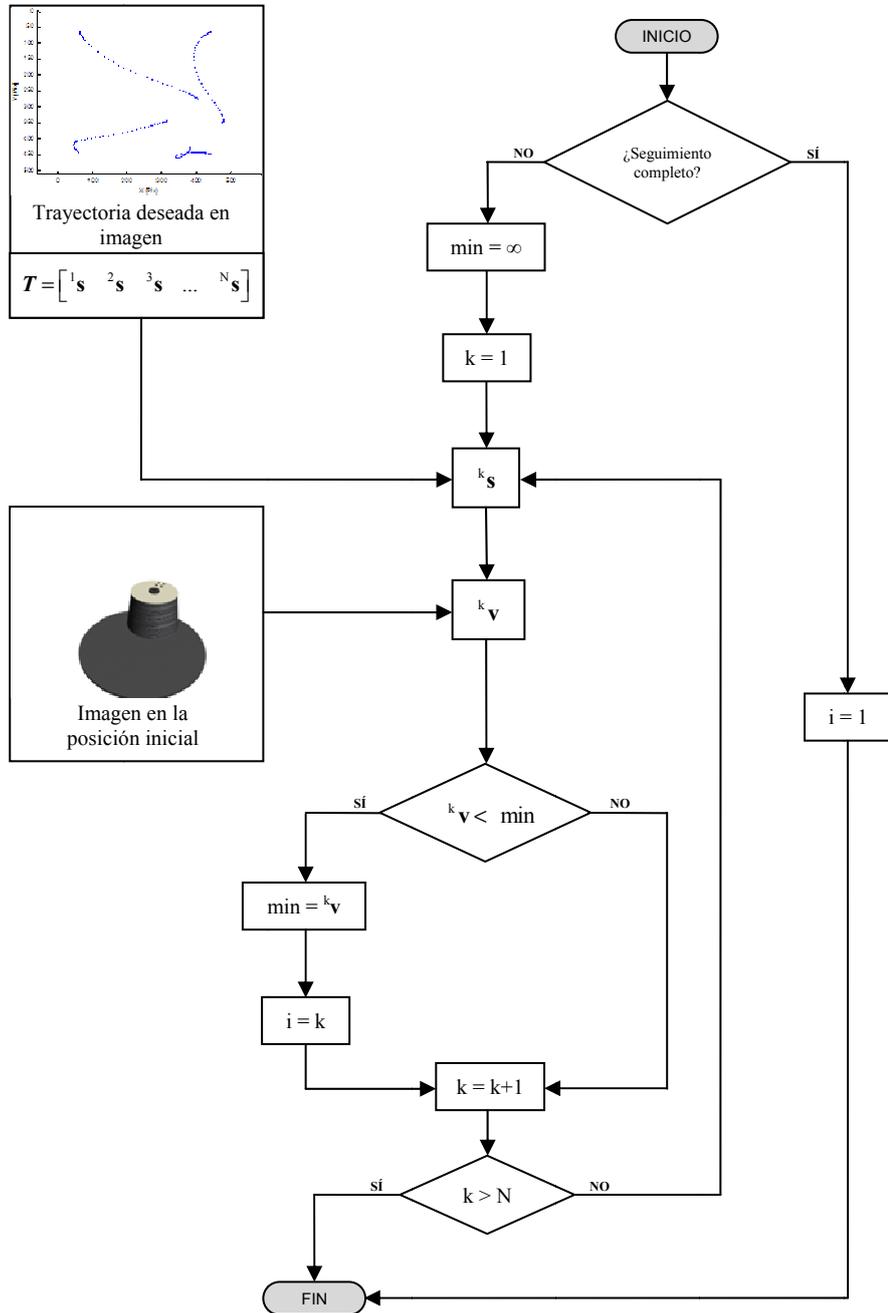


Figura 4-22. Esquema del algoritmo de búsqueda de la posición inicial dentro de la trayectoria deseada.

El problema que debe resolver este nuevo bloque es el de obtener una interpolación válida en el espacio Cartesiano 3-D entre dos posiciones muestreadas de la trayectoria deseada. Dado que se tiene únicamente un muestreo en imagen de la trayectoria deseada, se necesita reconstruir de alguna manera la trayectoria entre dos muestras en el espacio 3-D. Si se interpolara en la imagen, no se estaría asegurando un movimiento correcto del robot en 3-D. El método propuesto parte de realizar un control

visual virtual en cada una de las muestras de la trayectoria deseada entre las que se quiere interpolar. De esta forma se consigue obtener los parámetros extrínsecos de la cámara en cada una de las posiciones. Con los parámetros extrínsecos se tiene información 3-D de la posición de la cámara, con lo que se puede generar una trayectoria recta en el espacio Cartesiano entre las dos posiciones de la trayectoria.

Tal como se describe en el capítulo 3, empleando control visual virtual se pueden obtener los parámetros extrínsecos de la cámara a partir de una imagen captada por la cámara. En el vector  ${}^k\mathbf{s}$  se tiene codificada la posición de las características en la posición  $k$  de la trayectoria deseada  $\mathbf{T}$ . A partir de este vector  ${}^k\mathbf{s}$  se reconstruye la imagen que el control visual virtual utiliza en su ley de control para mediante iteraciones llegar a obtener los parámetros extrínsecos e intrínsecos de la cámara que ha captado esta imagen.

Los parámetros extrínsecos de la cámara  ${}^c\mathbf{M}_0$  almacenan la rotación  $\mathbf{R}$  y la traslación  $\mathbf{t}$  entre el sistema de referencia de la cámara y el sistema de referencia del objeto. Para interpolar entre dos posiciones contiguas ( $k-1$  y  $k$ ) de la trayectoria deseada  $\mathbf{T}$ , se pueden separar las dos componentes, por un lado la traslación y por otro la rotación. Aplicando control visual virtual en  $k-1$  y en  $k$ , se obtendrán las matrices de rotación  ${}^{k-1}\mathbf{R}$  y  ${}^k\mathbf{R}$ . También se obtendrán los vectores de traslación  ${}^{k-1}\mathbf{t}$  y  ${}^k\mathbf{t}$ .

Para interpolar según una línea recta en el espacio Cartesiano 3-D entre las dos posiciones, basta con seguir la siguiente ecuación:

$${}^\alpha\mathbf{t} = \alpha \cdot ({}^k\mathbf{t} - {}^{k-1}\mathbf{t}), 0 \leq \alpha \leq 1 \quad (4.17)$$

La interpolación de la orientación no es tan trivial. Tal como se describe en el Anexo 1, la interpolación mediante cuaternios sí que permite resolver los problemas que se presentan con los ángulos de Euler (presencia de singularidades), o la matriz de rotación (que necesita de 9 parámetros para su definición en vez de los 3 que se requieren para obtener los ángulos de Euler). Los cuaternios ofrecen una mayor eficiencia computacional que los demás métodos de representación de la orientación. En el Anexo 1 se describen las principales herramientas matemáticas para conseguir generar una trayectoria interpolada a partir de las matrices de rotación  ${}^{k-1}\mathbf{R}$  y  ${}^k\mathbf{R}$  obtenidas en el paso anterior con el control visual virtual. Para ello, a partir de (A1.5) se generan los cuaternios  ${}^{k-1}\mathbf{Q}$  y  ${}^k\mathbf{Q}$  correspondientes a  ${}^{k-1}\mathbf{R}$  y  ${}^k\mathbf{R}$ .

En (A1.13) se define la forma de interpolar la orientación entre dos orientaciones definidas con cuaternios [Shoemake-1985]. Aplicando esta ecuación se puede calcular cualquier orientación intermedia entre las dos orientaciones dadas expresada según el cuaternio  ${}^\alpha\mathbf{Q}$ . El último paso será reconstruir la matriz de rotación  ${}^\alpha\mathbf{R}$  a partir de  ${}^\alpha\mathbf{Q}$  aplicando la ecuación (A1.3).

Con los datos obtenidos hasta ahora, dependiendo del valor de  $\alpha$ , se podrá obtener una configuración de la cámara válida en el espacio Cartesiano 3-D. El siguiente objetivo es generar posiciones en imagen de las características  ${}^a\mathbf{s}$ , tal y como se observarían por la cámara desde cualquier posición  $\alpha$  de la trayectoria interpolada. Para ello, se reconstruyen los parámetros extrínsecos de la cámara a lo largo de la trayectoria interpolada en 3-D a partir de  ${}^a\mathbf{R}$  y  ${}^a\mathbf{t}$ .

$${}^c\mathbf{M}_o(\alpha) = \begin{pmatrix} {}^a\mathbf{R}_{3 \times 3} & {}^a\mathbf{t}_{3 \times 1} \\ \mathbf{0}_{1 \times 3} & 1_{1 \times 1} \end{pmatrix} \quad (4.18)$$

El método para retroproyectar las características en la imagen es el mismo que se usa en control visual virtual, es decir, se parte de la posición de la característica respecto al sistema de referencia del objeto  ${}^o\mathbf{x}$  y de los parámetros extrínsecos de la cámara en la trayectoria interpolada  ${}^c\mathbf{M}_o(\alpha)$ .

$${}^c\mathbf{x}(\alpha) = {}^c\mathbf{M}_o(\alpha) {}^o\mathbf{x} \quad (4.19)$$

Siendo  ${}^c\mathbf{x}(\alpha) = (X(\alpha), Y(\alpha), Z(\alpha))$ , la característica 2D asociada se obtiene de la siguiente manera:

$$x(\alpha) = \frac{X(\alpha)}{Z(\alpha)}, \quad y(\alpha) = \frac{Y(\alpha)}{Z(\alpha)} \quad (4.20)$$

Finalmente se obtiene el punto en el plano imagen a partir de los parámetros intrínsecos y de este punto 2D métrico.

$$u(\alpha) = u_0 + p_x x(\alpha), \quad v(\alpha) = v_0 + p_y y(\alpha) \quad (4.21)$$

Esta última ecuación permite obtener las características tal como se verían desde la cámara situada a lo largo de la trayectoria que se ha interpolado entre las dos posiciones consecutivas de la trayectoria deseada en imagen. Siguiendo (4.21) se construye el vector de características  $\mathbf{s}(\alpha)$  a partir del cual se va a buscar la configuración,  ${}^j\mathbf{s}$ , que permite mantener constante el módulo de la velocidad en el seguimiento de la trayectoria deseada.

Para encontrar el punto exacto  $j$  en el que las características en la imagen (obtenidas a partir de esta trayectoria interpolada) aseguren la velocidad deseada, se utiliza un método de optimización no lineal basado en mínimos cuadrados (ver Anexo 3). Para ello, es necesario minimizar el error  $\mathbf{e}_o = |\mu_c(\alpha) - |v_d||$ , donde:

$$\mu_c(\alpha) = -\lambda_1 \hat{\mathbf{J}}_s^+ (\mathbf{s} - \mathbf{s}^*(\alpha)) \quad (4.22)$$

y

$$\mathbf{s}^*(\alpha) = pr_{\xi}({}^c\mathbf{M}_o(\alpha) \circ \mathbf{P}) \quad (4.23)$$

donde  $\mathbf{P}$  es el conjunto de posiciones del objeto consideradas y  $pr_{\xi}$  representa el modelo de proyección de perspectiva de acuerdo a los parámetros intrínsecos,  $\xi$ .

Una vez que se obtiene el valor de  $\alpha$  que minimiza  $e_o$ , el conjunto de características  $\mathbf{j}_s$  será igual a  $\mathbf{s}^*(\alpha)$ .

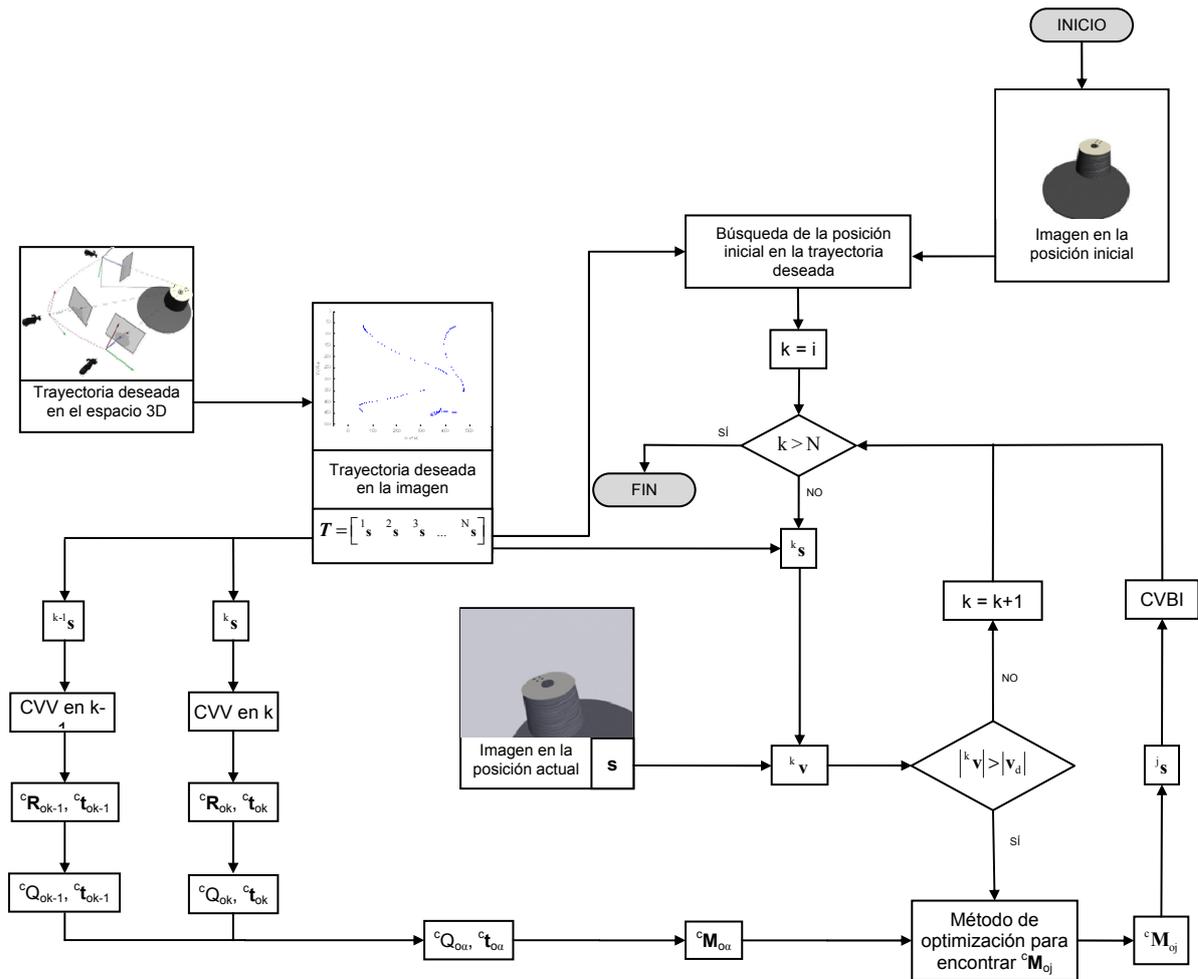


Figura 4-23. Esquema del algoritmo de seguimiento intemporal de trayectorias en el espacio imagen basado en Control Visual Virtual.

## 4.6. Implementación

Antes de exponer los distintos resultados obtenidos, en este apartado se describen brevemente algunos aspectos relativos a la implementación del sistema de seguimiento intemporal de trayectorias en la imagen basado en control visual virtual desarrollado. Para ello, en primer lugar se describe el sistema de simulación software desarrollado para posteriormente mostrar los distintos componentes de

la arquitectura del sistema donde se ha evaluado el sistema de seguimiento intemporal de trayectorias en la imagen basado en control visual virtual.

### 4.6.1. Sistema de simulación

Como base para las simulaciones desarrolladas y la posterior implementación se ha empleado la toolbox de control visual para MATLAB (cabe reseñar que, aunque las simulaciones se han desarrollado en MATLAB, la posterior implementación en un sistema real se ha llevado a cabo tras la conversión del código a C, lo que garantiza unos retardos menores y una mayor optimización del código).

### 4.6.2. Arquitectura del sistema

En la Figura 4-24 se muestra la arquitectura del sistema empleada para la realización de las pruebas de laboratorio.

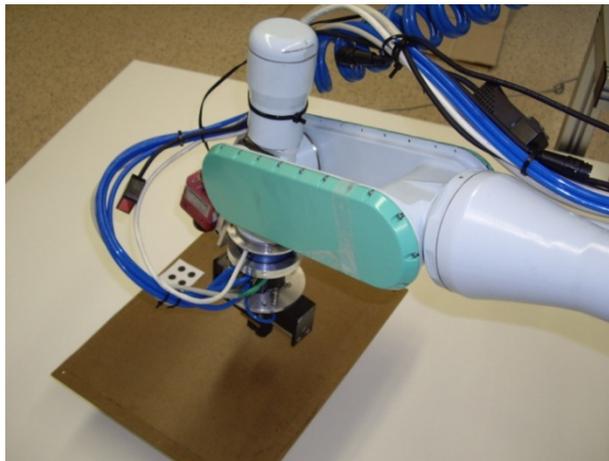


Figura 4-24. Principales componentes de la arquitectura del sistema.

Las características de cada uno de los dispositivos empleados son:

- Sistema de visión artificial: La captura de imágenes se realiza desde el extremo del robot empleando una cámara *PHOTONFOCUS MV-D752-160-CL-8*. Esta cámara está conectada a una tarjeta *Coreco X64CL-Pro* que permite realizar tareas de captura y procesamiento de imágenes. La cámara es capaz de adquirir y procesar imágenes de 320x240 a una velocidad de 100 frames/seg.
- Sistema robot: El robot empleado ha sido un *Mitsubishi PA-10* con 7 grados de libertad.
- Objetivo: El objeto seguido se trata de un patrón cuadrado con cuatro marcas que serán las que se extraerán durante el seguimiento.

## 4.7. Resultados

En este apartado se van a describir las distintas pruebas realizadas que muestran el correcto funcionamiento del sistema. Para ello, en primer lugar se muestran los resultados obtenidos en la simulación para posteriormente exponer los resultados correspondientes al utilizar la arquitectura descrita en el apartado 4.6.2.

### 4.7.1. Resultados de simulación

Para evaluar el sistema propuesto y compararlo con otros sistemas de seguimiento de trayectorias en el espacio imagen descritos en el presente capítulo, se presentan a continuación los resultados obtenidos en el seguimiento de distintas trayectorias. Estas trayectorias permiten comprobar la validez del sistema propuesto, destacando las mejoras obtenidas en cualquier tipo de trayectoria respecto a métodos previos.

#### 4.7.1.1. Trayectoria con las características iniciales en la trayectoria deseada.

Este primer experimento presenta un bucle en el espacio imagen para la trayectoria de una de las características. La presencia de un bucle en alguna de las trayectorias de las características provocaba ciertos problemas para el control visual basado en flujo de movimiento descrito en la sección 4.3. La Figura 4-25 muestra la trayectoria 3D que se desea seguir por la cámara situada en el extremo del robot, así como la trayectoria muestreada en el espacio imagen,  $T$ .

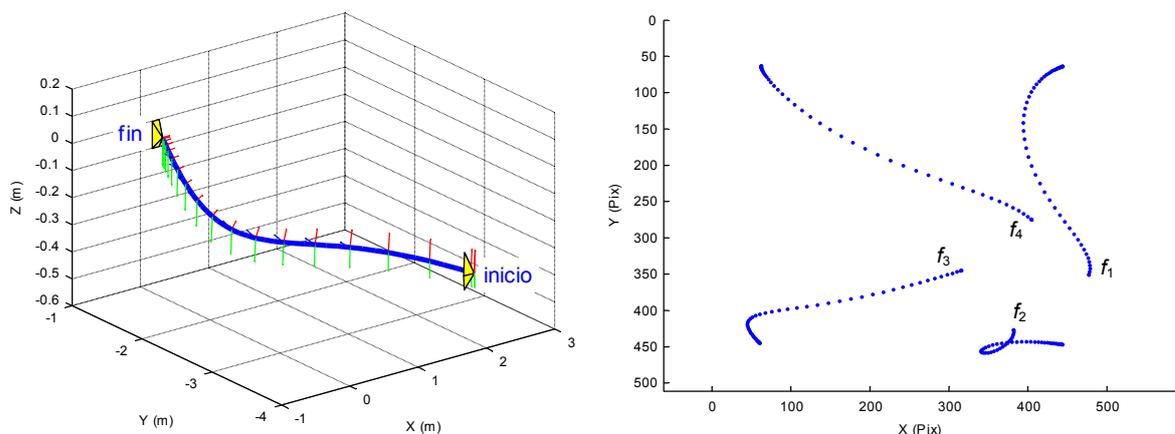


Figura 4-25. Muestreo de la trayectoria deseada en el espacio cartesiano 3-D y en la imagen.

Si se aplicara la técnica del control visual clásico, nunca podría seguirse la trayectoria deseada. El control visual clásico permite obtener en el espacio imagen trayectorias en línea recta entre la

configuración inicial y la final. Esta particularidad impide reconstruir la trayectoria deseada también en el espacio euclídeo. En la Figura 4-26 se muestra en azul la trayectoria deseada, tanto en el espacio 3-D como en el espacio imagen, mientras que en rojo se representa la trayectoria en ambos planos tras ejecutar control visual basado en imagen clásico.

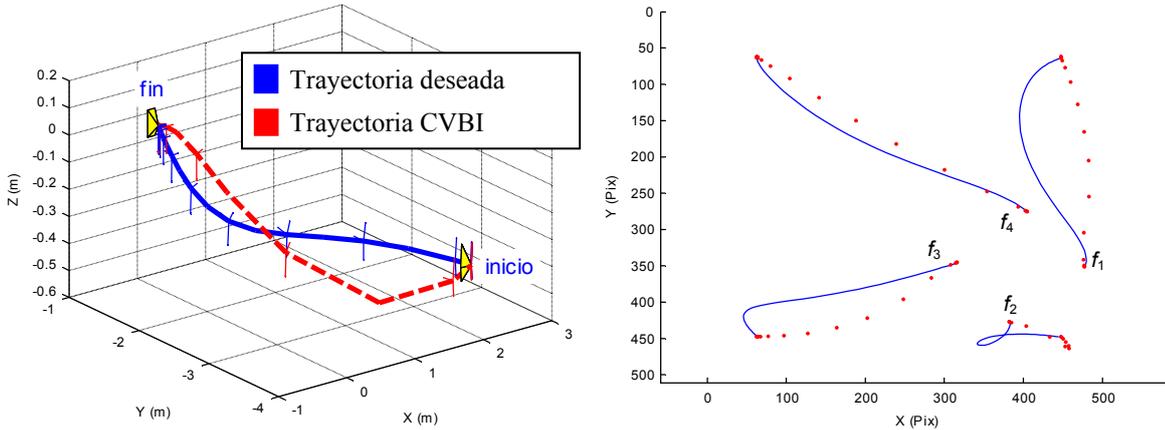


Figura 4-26. Comparativa entre la trayectoria deseada y la obtenida con el control visual basado en imagen clásico en el espacio cartesiano 3-D y en la imagen.

Se puede observar en la Figura 4-26 que la trayectoria del extremo del robot en el espacio 3-D no se aproxima a la deseada. Lo mismo ocurre en el espacio imagen. Para conseguir el correcto seguimiento de la trayectoria deseada se emplea la aproximación descrita en la sección 6. En la Figura 4-27 se muestra que el seguimiento con el método propuesto es completamente satisfactorio en el plano imagen.

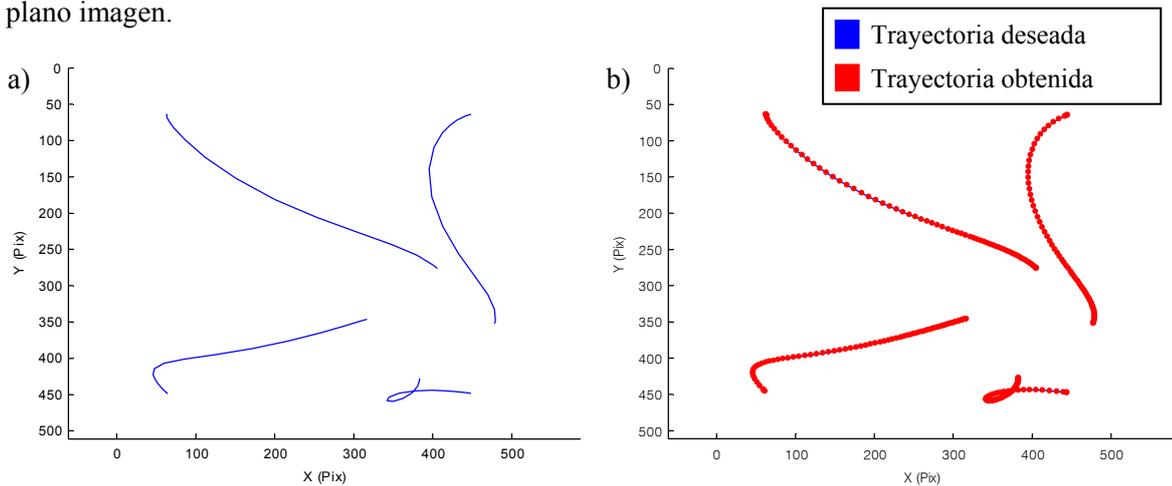


Figura 4-27. a) Trayectoria deseada en la imagen. b) Comparativa entre la trayectoria deseada y la obtenida con el método de seguimiento propuesto.

De nuevo en la Figura 4-27 se ha representado en azul la trayectoria deseada, y en rojo la trayectoria seguida con el método propuesto de seguimiento. La gráfica representada en la Figura 4-27 se ha obtenido controlando el módulo de la velocidad a 10 mm/s. En posteriores secciones se

mostrarán distintas gráficas indicando la influencia de este parámetro en el seguimiento de la trayectoria. Se observa en la Figura 4-27.b que el seguimiento con esta velocidad hace que se superpongan totalmente la trayectoria deseada y la real obtenida con el método propuesto de control visual basado en control visual virtual.

A pesar de que en la imagen el seguimiento es correcto, se debe demostrar también que se consigue el seguimiento de la trayectoria en el espacio cartesiano 3-D con la estrategia propuesta. Para comprobar esto, en la Figura 4-28 se muestran distintas vistas de la trayectoria 3D descrita en este experimento. En estas figuras se muestra la evolución del sistema de coordenadas de la cámara ubicada en el extremo del robot durante todo el seguimiento (se representa la trayectoria deseada y la real obtenida al aplicar el método propuesto). Puede observarse que se realiza un correcto seguimiento de la trayectoria en el espacio 3-D.

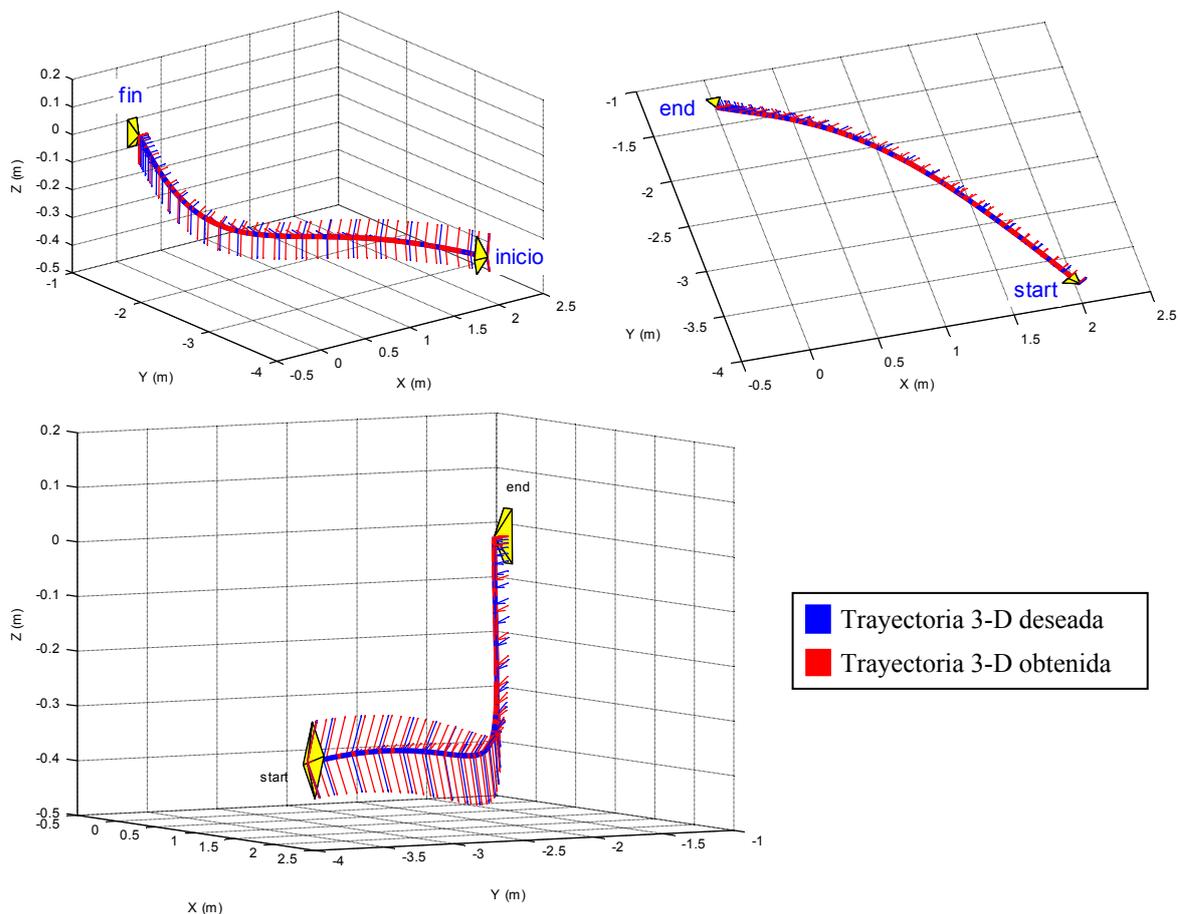


Figura 4-28. Distintas vistas de la trayectoria descrita por la cámara en el espacio Cartesiano 3-D.

Si se estudia la evolución del error en la imagen a lo largo del seguimiento de la trayectoria (Figura 4-29) se observa que éste no supera los 15 píxeles en el método propuesto. La evolución del error en el control visual basado en imagen clásico es exponencial. Es evidente que en la estrategia

propuesta para el seguimiento de la trayectoria no se obtiene una evolución exponencial del error, ya que la referencia es variable. El método trata de mantener constante el módulo de la velocidad, para lo cual se envía al controlador una referencia distinta en cada iteración.

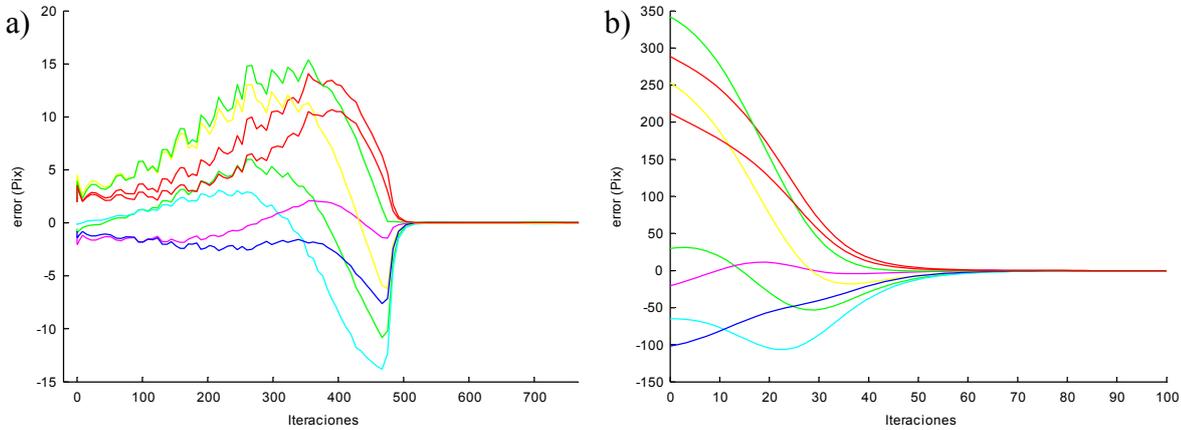


Figura 4-29. Error en la imagen. a) Empleando el método propuesto. b) Empleando control visual basado en imagen.

Para terminar de analizar los datos obtenidos durante el seguimiento de esta trayectoria, en la Figura 4-30 se representan las velocidades traslacionales y rotacionales enviadas al extremo del robot para los casos del método propuesto y para el control visual basado en imagen. En la Figura 4-30.a se observa que el módulo de la velocidad traslacional se mantiene tal y como se ha comentado en torno a 10 mm/seg. Se aprecia en la Figura 4-30.b que las velocidades con el control visual basado en imagen son mayores, y que en ningún momento se mantienen controladas, pudiendo sobrepasar la velocidad máxima del robot. En posteriores secciones se presentarán gráficas que muestran la flexibilidad del método propuesto para adaptarse a la velocidad requerida, obteniendo aún así muy buenas prestaciones tanto en el seguimiento en el plano imagen como en el plano cartesiano 3-D.

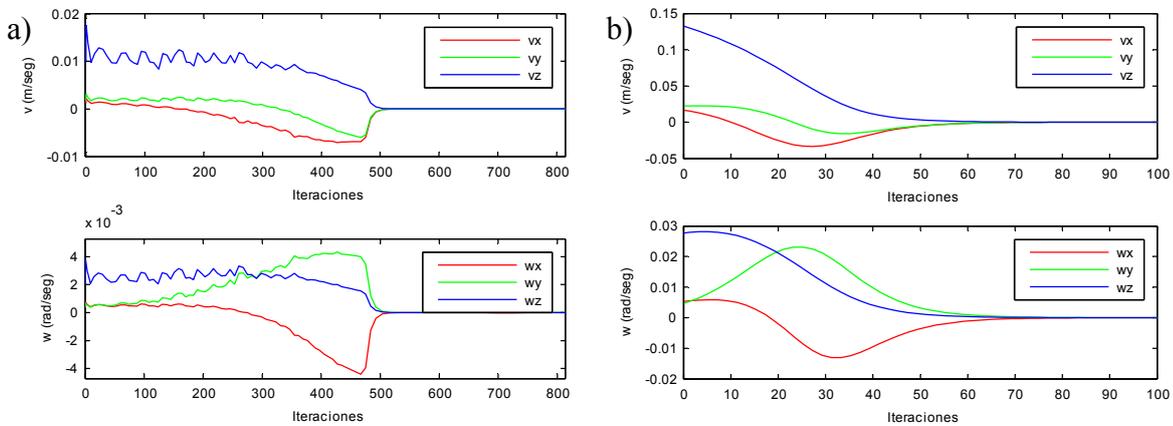


Figura 4-30. Velocidades aplicadas al robot durante el seguimiento de la trayectoria.

El método propuesto ha funcionado de forma correcta para esta primera trayectoria. En la siguiente prueba se va a introducir una modificación en la posición inicial de la cámara. De esta forma, la posición inicial de las características observadas no será igual que la posición inicial de las características en la trayectoria deseada.

#### 4.7.1.2. Trayectoria con las características iniciales fuera de la trayectoria deseada

En la imagen captada por el sistema de visión al inicio de este experimento no se observan las características correspondientes a la posición inicial de la trayectoria deseada. De hecho, la localización inicial del robot en el espacio Cartesiano 3-D no se corresponderá con ninguna de las representadas en el conjunto  $\tau = \{k\gamma/k \in 1 \dots N\}$  (trayectoria deseada del robot), como se puede apreciar en la Figura 4-31.

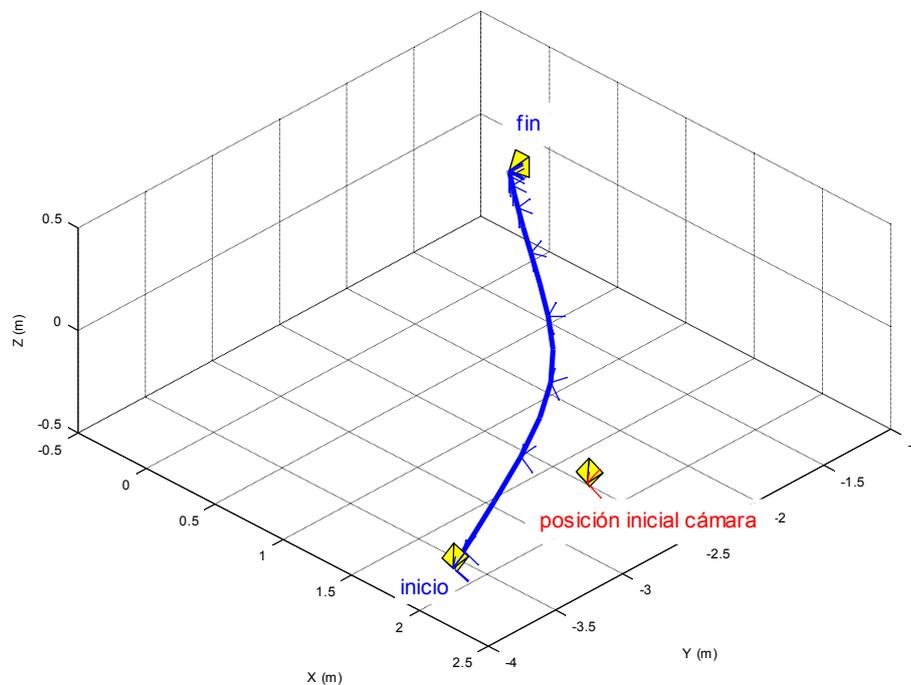


Figura 4-31. Posición inicial de la cámara fuera de la trayectoria en el espacio cartesiano 3-D.

Esta situación podría resolverse de dos formas distintas. En una primera aproximación se guiaría al robot hacia la posición inicial de la trayectoria deseada. Esta opción es la idónea para resolver el problema presentado en aquellas ocasiones en las que es absolutamente indispensable el seguimiento total de la trayectoria. Otra alternativa sería llevar al robot a una configuración  $k\gamma$  adecuada a partir de la cual pueda extraerse unas características en la imagen  $k\mathbf{s} = \{k\mathbf{f}_i/i \in 1 \dots M\}$  que se encuentren en la trayectoria deseada y, a su vez, conformen una configuración deseada de las características en la imagen.

El primer caso se resuelve de forma natural por el método propuesto, dado que el sistema obtendrá una velocidad excesiva en la primera iteración, lo que interpolará una configuración  ${}^k\mathbf{s}$  muy cercana a la primera de la trayectoria. El resultado final será la ejecución de control visual basado en imagen con aproximadamente la primera configuración  ${}^1\mathbf{s} = \{f_i / i \in 1 \dots M\}$  como posición final, para luego seguir la trayectoria deseada de forma completa. En la Figura 4-32 se muestra esta situación descrita para el plano imagen.

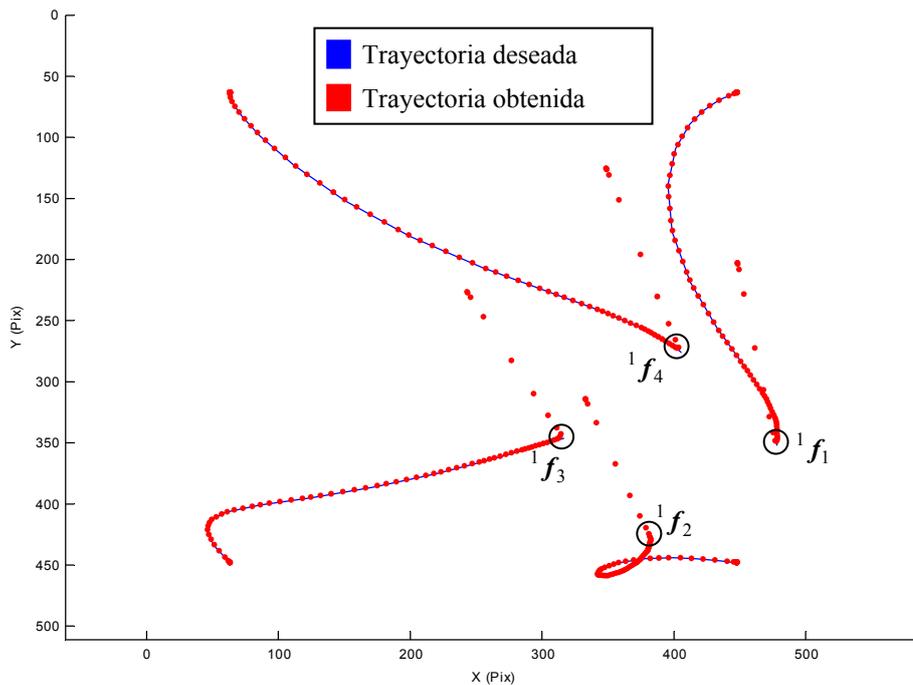


Figura 4-32. Comparativa entre la trayectoria real y la trayectoria deseada para la aproximación de seguimiento completo de la trayectoria. (La posición inicial no se encuentra en la trayectoria deseada).

En imagen se sigue correctamente el total de la trayectoria con esta aproximación. En el espacio cartesiano 3-D el resultado es también el esperado. En la Figura 4-33 se observa que la cámara evoluciona desde la posición inicial hasta la posición inicial de la trayectoria deseada, momento desde el que ya se efectúa de forma correcta el seguimiento de la trayectoria.

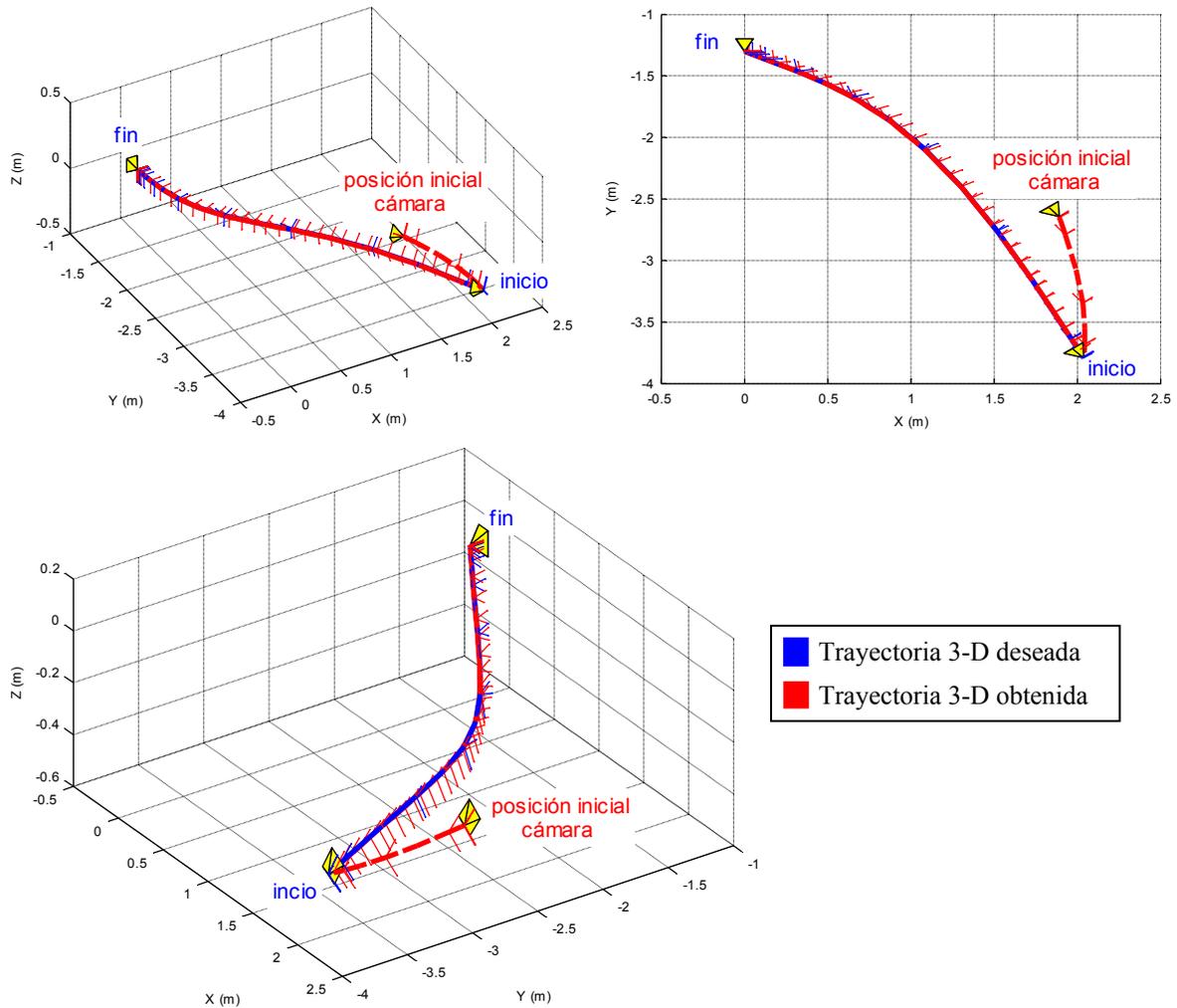


Figura 4-33. Trayectoria descrita por la cámara en el espacio 3-D. Posición inicial fuera de la deseada. Seguimiento completo de la trayectoria deseada.

Para abordar con éxito la segunda aproximación se debe recorrer en un principio la trayectoria deseada. Si las características iniciales,  ${}^{inic}s$ , no se encuentran situadas a lo largo de la trayectoria deseada, el sistema tendrá que converger hacia la configuración, que asegure la velocidad mínima desde la posición inicial de la cámara. Cuando se han evaluado todas las configuraciones de la trayectoria deseada, se tendrá la configuración  ${}^k s$ , más cercana a la configuración actual  ${}^{inic}s$ . Una vez determinada esta configuración  ${}^k s$ , el sistema convergerá hacia ella (empleando un sistema de control basado en imagen), consecuentemente, la trayectoria descrita desde  ${}^{inic}s$  hasta  ${}^k s$  no se encontrará en la trayectoria deseada (ni en la imagen ni en el espacio 3-D), sin embargo, una vez que se haya alcanzado  ${}^k s$ , se puede asegurar que se obtendrá una localización  ${}^k \gamma$  deseada para el robot en el espacio 3-D.

Tras la determinación de la configuración de control más cercana y la convergencia hacia ella se aplica lo desarrollado a lo largo del capítulo ya que, una vez alcanzada una localización deseada del

robot  ${}^k\gamma$ , se tiene una situación como la descrita para el caso 1 (características iniciales en la trayectoria deseada).

En la Figura 4-34 se muestra un experimento en el que la posición inicial de las características no se encuentra en la trayectoria deseada. Se observa que el controlador determina la configuración  ${}^k s$  más cercana a la configuración inicial. Una vez alcanzada  ${}^k s$  el sistema continúa la evolución de las características según la trayectoria deseada codificada en  $T$ .

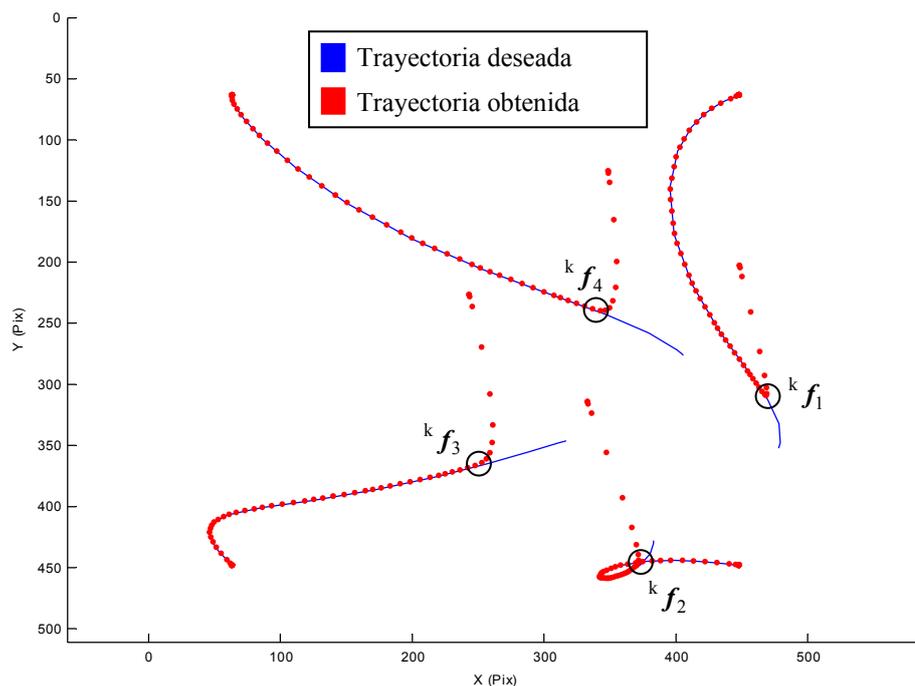


Figura 4-34. Comparativa entre la trayectoria real y la trayectoria deseada para la aproximación de búsqueda del lugar más cercano de la trayectoria. (La posición inicial no se encuentra en la trayectoria deseada).

Se puede observar que este experimento es semejante al que se mostró en el Apartado 4.7.1.1 por lo que la trayectoria 3-D a partir de la configuración de control debería ser la misma que la obtenida en dicho apartado. Esto se puede observar en la Figura 4-35 en la que se representa la evolución del sistema de coordenadas de la cámara ubicada en el extremo del robot. En esta figura se observa como la cámara tiende a aproximarse hacia la trayectoria requerida y, una vez alcanzada la configuración de control, continua su evolución según la trayectoria deseada.

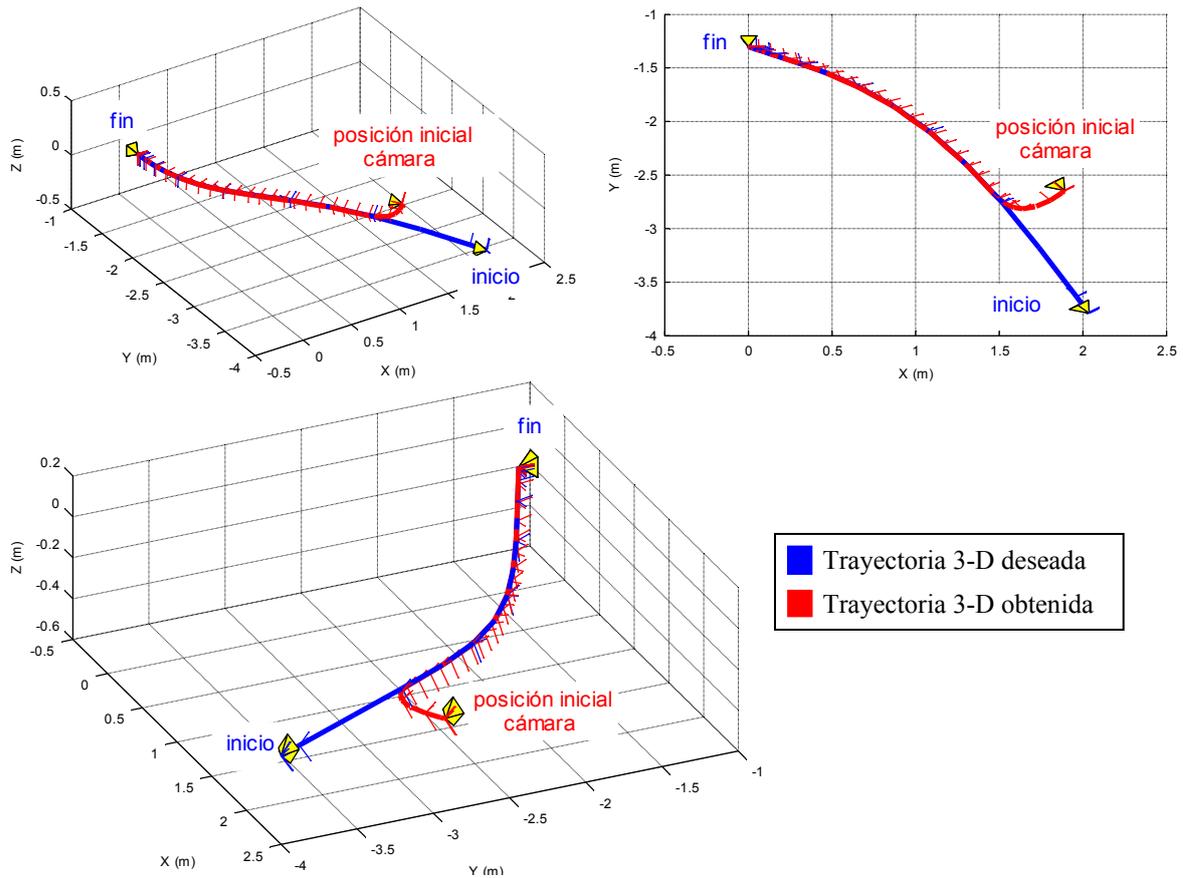


Figura 4-35. Trayectoria descrita por la cámara en el espacio 3-D. Posición inicial fuera de la deseada. Búsqueda de la configuración más cercana dentro de la trayectoria deseada.

El hecho de tener que realizar un desplazamiento previo entre la posición inicial de la cámara y la posición inicial de la trayectoria a seguir (en el primer caso), o la posición más cercana (en el segundo), implica obtener velocidades mayores hasta llegar al comienzo de la trayectoria. En la Figura 4-36 se representa el error cometido en píxeles en cada iteración. Se observa como en la primera aproximación (Figura 4-36.a) el error en píxeles es mayor al comienzo, que el obtenido en la segunda aproximación (Figura 4-36.b). Esto es lógico, ya que en la segunda aproximación se busca la configuración más cercana para moverse hacia ella, no importando realizar el seguimiento completo de la trayectoria deseada. En la primera aproximación, el hecho de tener que seguir completamente la trayectoria hace que se obtenga un mayor error inicial.

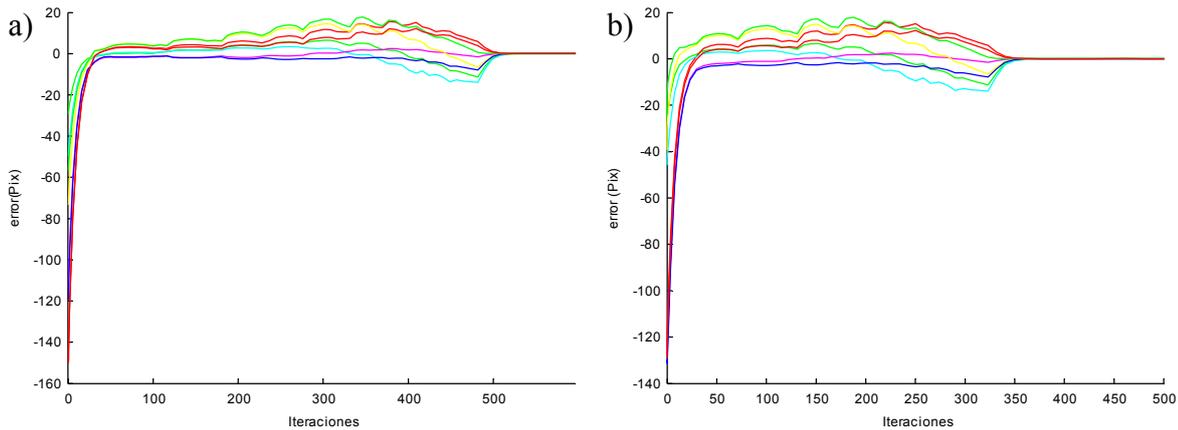


Figura 4-36. Error en la imagen. a) Primera aproximación: seguimiento de la trayectoria completa. b) Segunda aproximación: búsqueda de la posición inicial más cercana.

Este error en las primeras iteraciones entre las características actuales y las deseadas conlleva un aumento en las velocidades iniciales (ver Figura 4-37). Hay que recalcar, sin embargo, que en el momento en el que se alcanza alguna configuración dentro de la trayectoria deseada, se sigue realizando el control de velocidad descrito en los apartados anteriores. Esto ocurre en las dos aproximaciones.

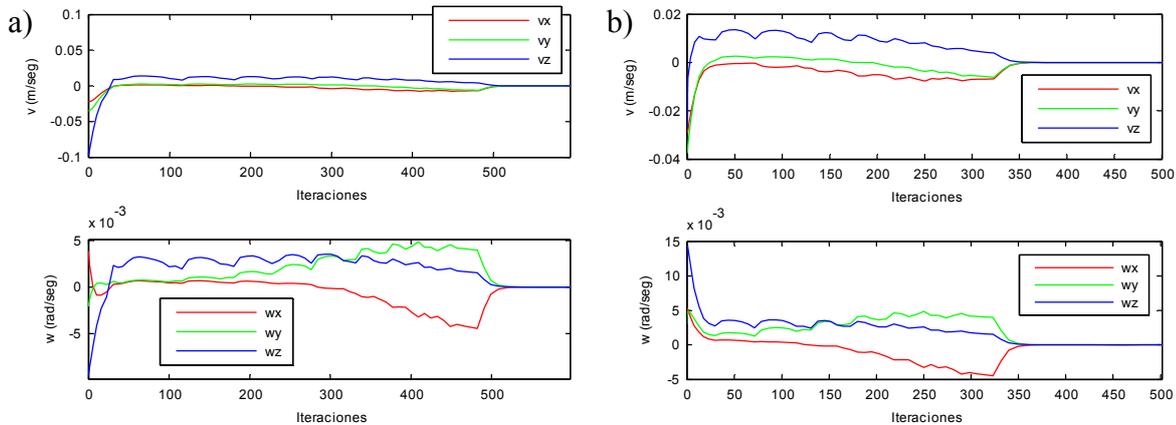


Figura 4-37. Evolución con el tiempo de la velocidad aplicada al robot. a) Primera aproximación: seguimiento de la trayectoria completa. b) Segunda aproximación: búsqueda de la posición inicial más cercana.

La Figura 4-38 permite comprobar que el control del módulo de la velocidad de traslación se sigue llevando a cabo en el momento en el que se alcanza una configuración válida. Se puede apreciar que la velocidad inicial es mayor en el primer caso (Figura 4-38.a) que en el segundo caso en el que se ha buscado la posición más cercana (Figura 4-38.b).

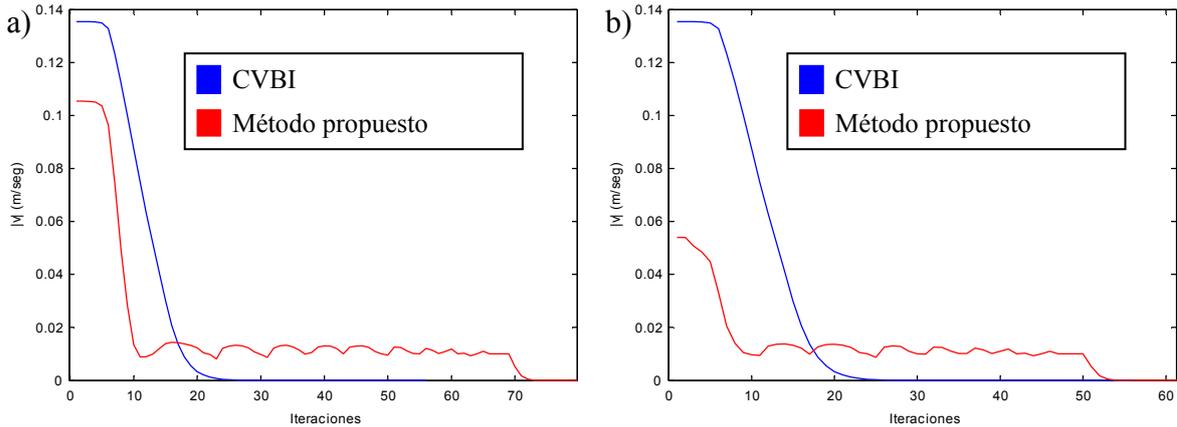


Figura 4-38. Evolución con el tiempo del módulo de la velocidad aplicada al robot. a) Primera aproximación: seguimiento de la trayectoria completa. b) Segunda aproximación: búsqueda de la posición inicial más cercana.

## 4.7.2. Resultados experimentales

Para demostrar el correcto funcionamiento en entornos reales de la estrategia propuesta de seguimiento de trayectorias en la imagen basado en control visual virtual, se van a mostrar los resultados obtenidos empleando un sistema robótico, cuya arquitectura se ha descrito en el apartado 4.6.2, para el seguimiento de distintas trayectorias representativas.

### 4.7.2.1. Trayectoria 1: trayectoria en espacio imagen muy oscilante.

La primera trayectoria con la que se ha evaluado el método propuesto presentaba un bucle en la evolución de una de las características. Este tipo de trayectorias planteaba ciertos problemas al control visual basado en flujo de movimiento descrito en el apartado 4.3. Con las pruebas realizadas se ha comprobado que el sistema propuesto en la sección 4.6 funciona correctamente ante situaciones como la descrita.

A continuación se va a evaluar el correcto seguimiento del método propuesto ante trayectorias deseadas muy oscilantes. Para ello, en la Figura 4-39 se puede observar el muestreo de la nueva trayectoria deseada, tanto en el espacio cartesiano 3-D como en el plano imagen. En este tipo de trayectorias se hace crítico el parámetro del muestreo de la trayectoria deseada. Con pocas muestras la trayectoria seguida no se ajustará a la deseada, mientras que al aumentar el número de muestras se observará como se consigue calcar la trayectoria tanto en imagen como en 3-D.

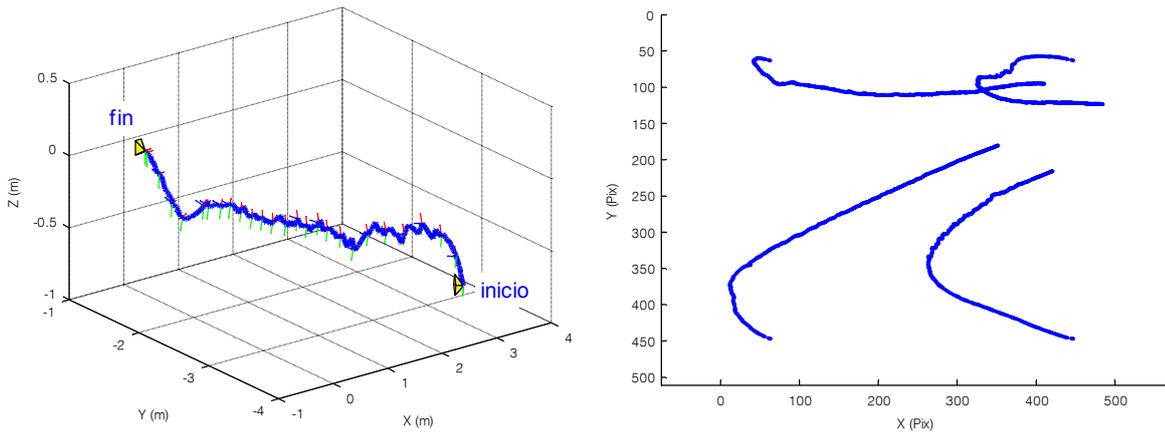


Figura 4-39. Muestreo de la trayectoria deseada en el espacio cartesiano 3-D y en la imagen.

Ajustando la velocidad a 10 mm/seg y con el muestreo máximo se consigue realizar un seguimiento perfecto de la trayectoria deseada. En la Figura 4-40.b se puede ver que tanto la trayectoria real obtenida con el método propuesto en el Apartado 4.6, como la trayectoria deseada se superponen.

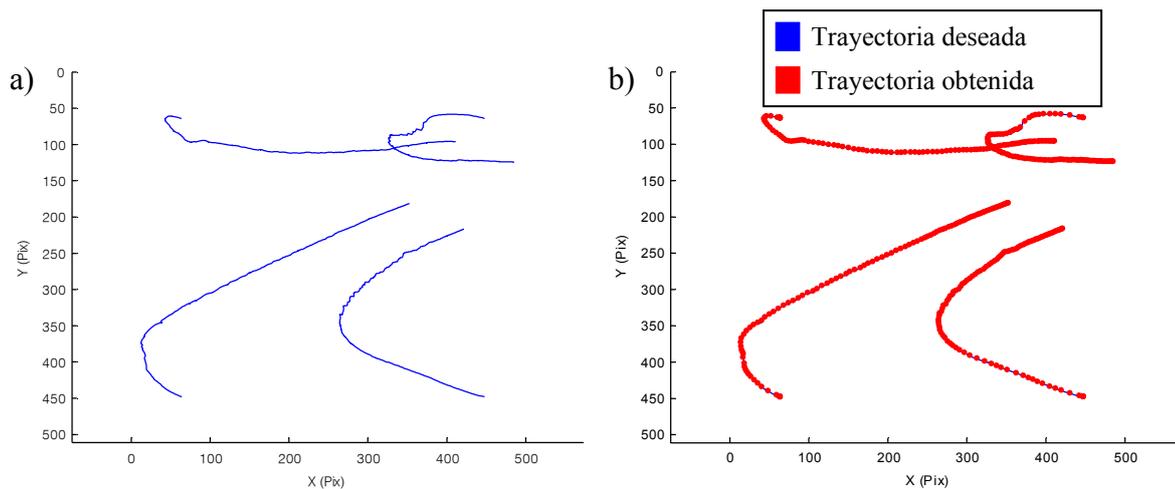


Figura 4-40. a) Trayectoria deseada en la imagen. b) Comparativa entre la trayectoria deseada y la obtenida con el método propuesto.

La Figura 4-41 muestra que el resultado es también satisfactorio en el espacio 3-D. El seguimiento empeora cuando se aumenta la velocidad del sistema o cuando se disminuye el número de muestras de la trayectoria deseada. Estos resultados se verán en las siguientes secciones.

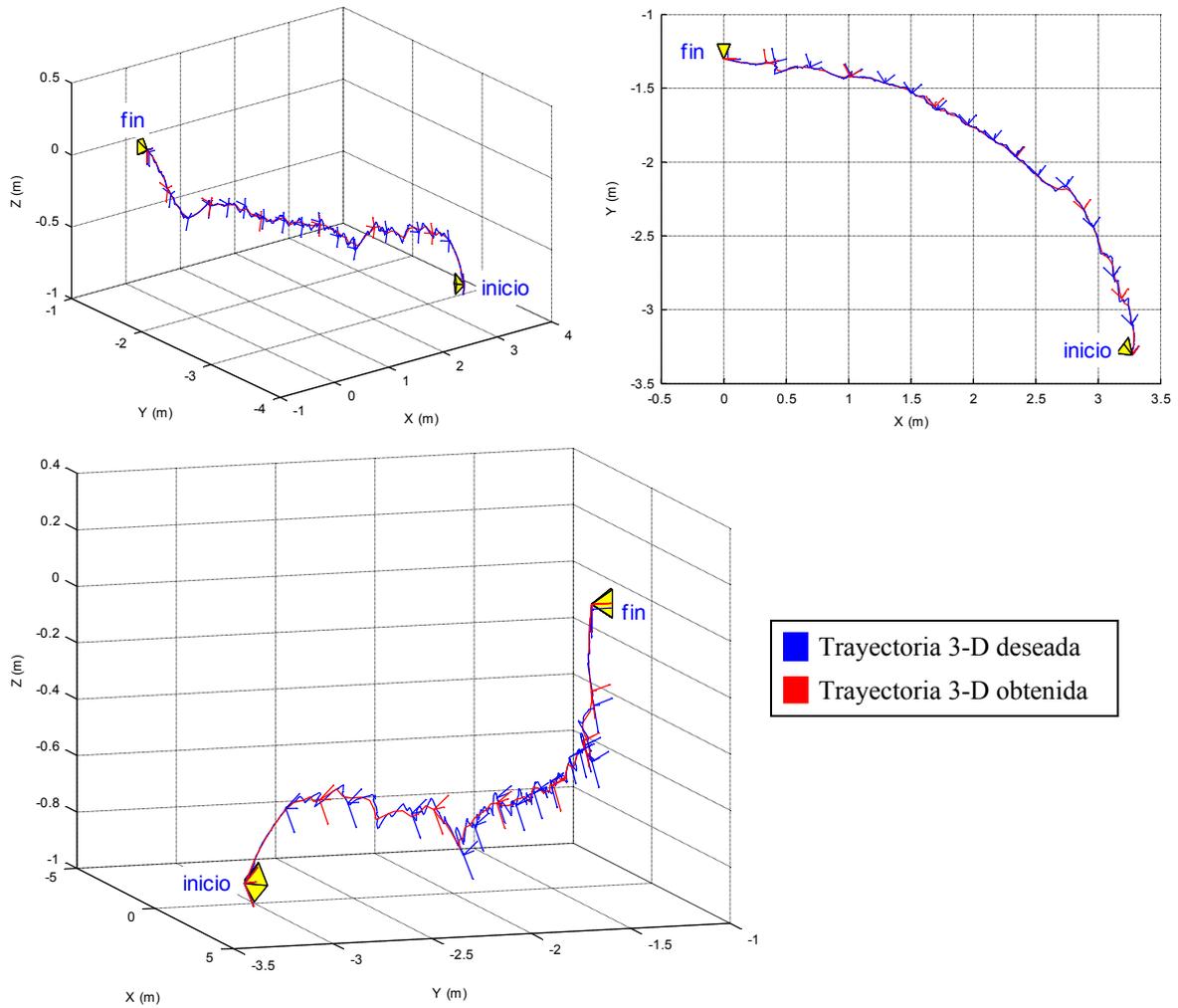


Figura 4-41. Distintas vistas de la trayectoria descrita por la cámara en el espacio Cartesiano 3-D.

Con el método propuesto se consigue mantener el error en la imagen por debajo de 3 píxeles durante el seguimiento de la trayectoria. En la Figura 4-42.a se observa además, que existe un error residual menor de un píxel que no afecta a la terminación de forma correcta de la trayectoria.

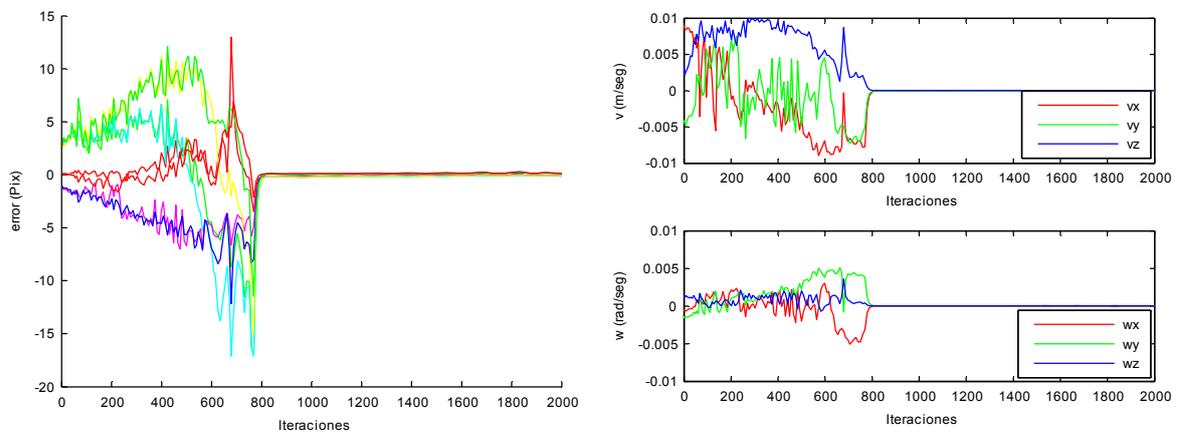


Figura 4-42. a) Error en la imagen. b) Evolución con el tiempo de la velocidad aplicada al robot.

La Figura 4-42.b muestra las velocidades enviadas al extremo del robot durante el seguimiento de la trayectoria. El módulo de la velocidad de traslación se mantiene constante en torno a 10 mm/seg tal y como se esperaba.

#### **4.7.2.2. Comparativa frente a sistemas de seguimiento con control visual dependientes del tiempo**

Como se ha descrito a lo largo de este capítulo, los sistemas que hacen uso de control visual para el seguimiento de trayectorias utilizados hasta la actualidad [Mezouar-2002a, Chesi-2005, Malis-2004a] emplean una referencia variable con el tiempo. Se ha expuesto que estos sistemas no garantizan un correcto seguimiento de la trayectoria, pudiendo ser las referencias excesivamente restrictivas, con lo que el sistema intentará en todo momento mantener estas restricciones temporales a pesar de no realizar un correcto seguimiento de la trayectoria. El hecho de emplear el seguimiento intemporal de trayectorias en la imagen basado en control visual virtual garantiza el correcto seguimiento, no viéndose el sistema afectado ante restricciones temporales.

Supóngase que el robot se encuentra trabajando en un entorno real en el que puede interactuar con cualquier elemento del entorno. Para posibilitar esto, se introduce en el robot un sensor de fuerza de forma que cuando se detecte una fuerza de interacción elevada se pueda detener el seguimiento de la trayectoria o modificar la misma. Con estas condiciones se realiza el seguimiento de la trayectoria deseada empleando seguimiento intemporal de trayectorias en la imagen basado en control visual virtual. Si no se realiza ninguna interacción del robot con el entorno, y el sistema de seguimiento intemporal de trayectorias en la imagen basado en control visual virtual actúa normalmente, la duración de la trayectoria es de 12 segundos. Sin embargo, para exponer la diferencia en el comportamiento del sistema con otro dependiente del tiempo, en la Figura 4-43 se muestra un ejemplo de seguimiento en el que durante el transcurso de la trayectoria ésta se obstruye durante una duración aproximada de 3 segundos. Durante la obstrucción el robot se sale de la trayectoria 3D deseada. En la Figura 4-43, además de la trayectoria 3-D descrita por el extremo del robot, también se muestra la deseada en azul. En esta figura se observa que, una vez finalizada la obstrucción de la trayectoria, el sistema continúa con el seguimiento de la misma a partir de la siguiente referencia.

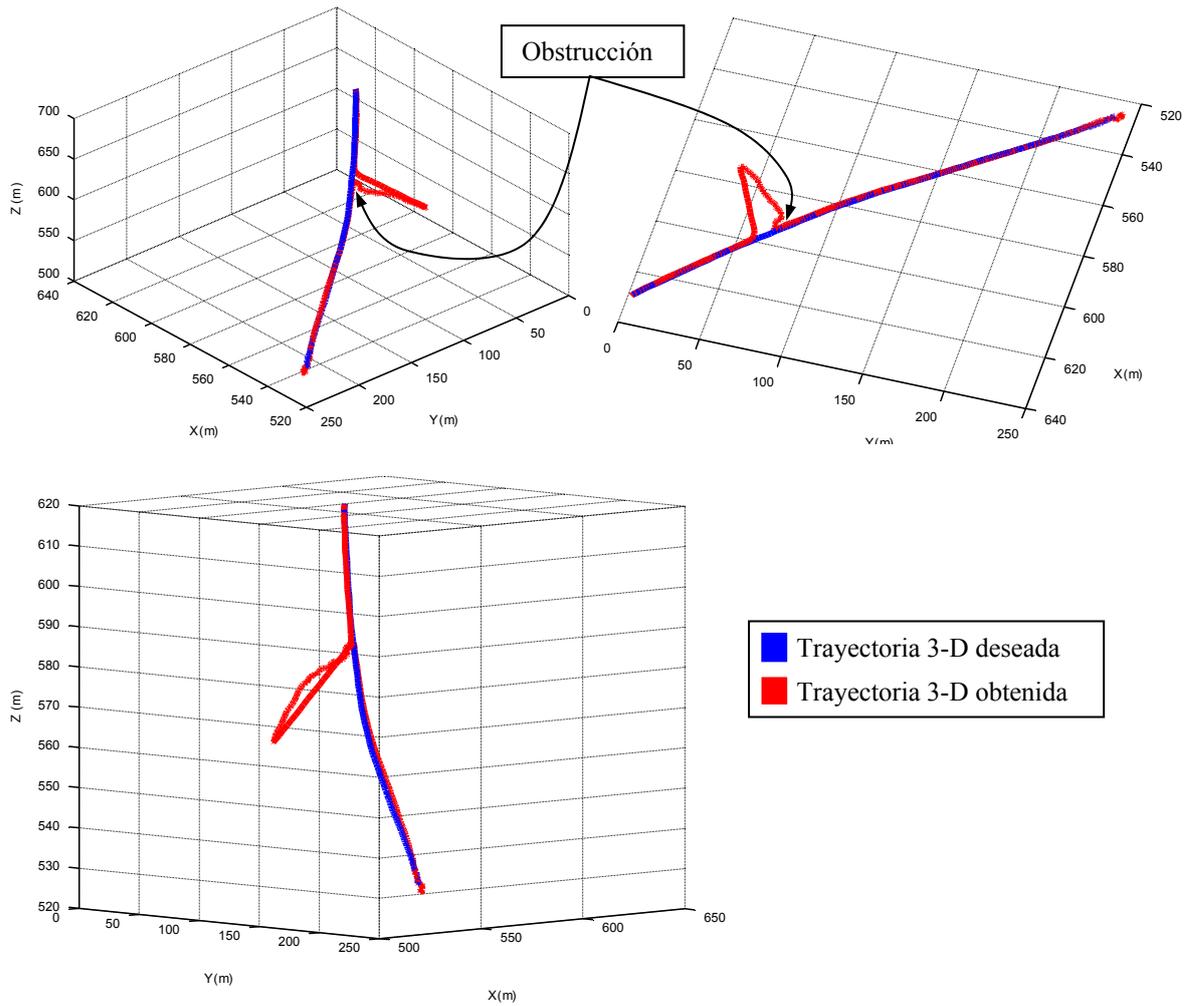


Figura 4-43. Trayectoria 3-D obtenida con seguimiento intemporal de trayectorias en la imagen basado en control visual virtual.

La Figura 4-44 muestra la evolución de las características en la imagen usando el método propuesto de seguimiento intemporal.

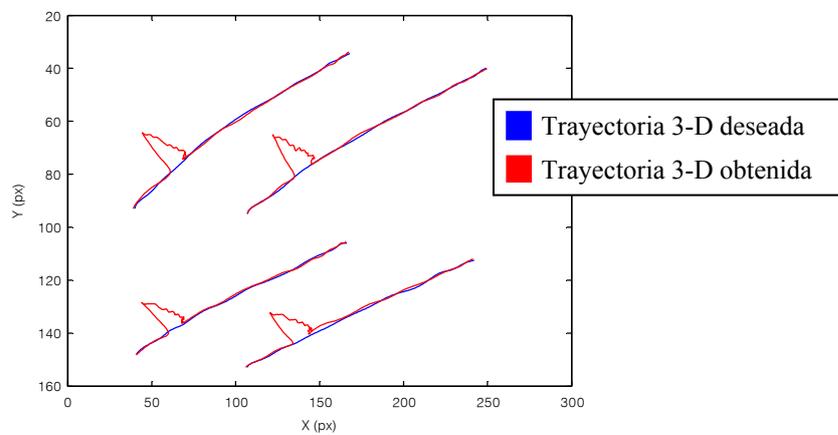


Figura 4-44. Evolución de las características en la imagen obtenida con seguimiento intemporal de trayectorias en la imagen basado en control visual virtual.

En la gráfica se puede observar el correcto seguimiento de la trayectoria deseada hasta el momento en el que se obstruye el seguimiento de manera externa. A partir de que se libera el robot el sistema vuelve a la trayectoria para terminarla a partir de la siguiente referencia.

Supóngase ahora que se aplica un sistema de control visual con referencias variables con el tiempo para el seguimiento de una trayectoria en imagen. Como se hizo en el caso anterior, durante el transcurso de la trayectoria, ésta se obstruye durante 3 segundos. El resultado del seguimiento obtenido se representa en la Figura 4-45. En esta figura se ha representado en azul la trayectoria deseada, y se observa claramente como el sistema de seguimiento no es capaz de continuar con la trayectoria una vez finalizada la obstrucción. Esto es debido a que al aparecer cualquier suceso imprevisto durante el seguimiento, como es en este caso una interacción con el entorno, las referencias temporales se desfasan y se hacen excesivamente restrictivas, con lo que el sistema tiende a intentar mantener estas restricciones temporales a pesar de no conseguir realizar el seguimiento de la trayectoria deseada.

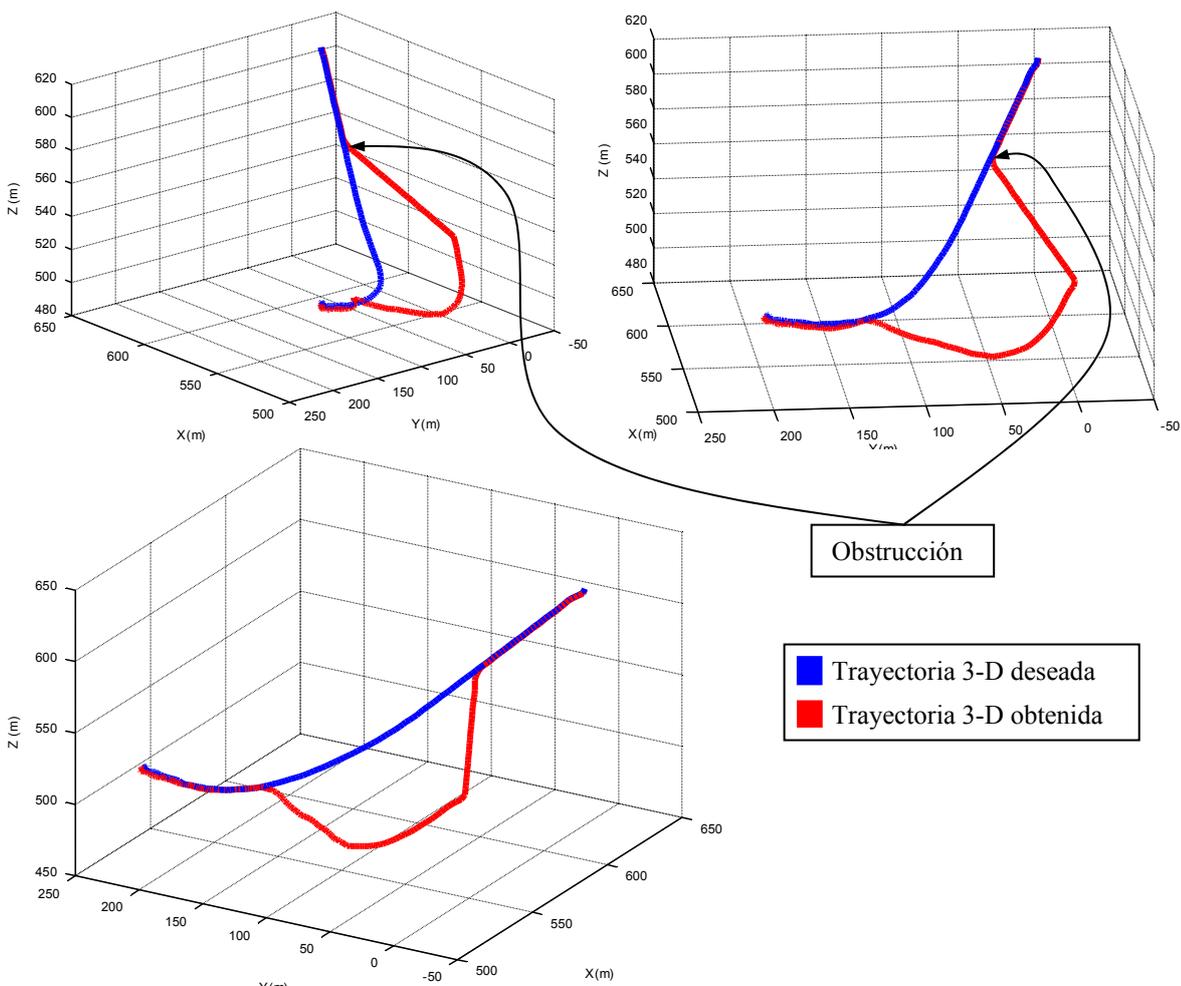


Figura 4-45. Trayectoria 3-D obtenida con sistemas dependientes del tiempo (Prueba 1).

El efecto expuesto en la Figura 4-45 se muestra con mayor claridad si la duración de la obstrucción es mayor. Para ello, en la Figura 4-46 se muestra un ejemplo de seguimiento en el que la duración de la obstrucción es de 6 segundos. Este aspecto resulta fundamental cuando el robot interactúa con el entorno, tal y como se verá en el Capítulo 6, ya que, por ejemplo, una operación de desensamblado puede retener el robot más tiempo del necesario y el comportamiento deseado sería el de continuar con la trayectoria independientemente de la duración de la obstrucción.

La Figura 4-47 muestra la evolución de las características en la imagen usando un método de seguimiento temporal. En la gráfica se puede observar el correcto seguimiento de la trayectoria deseada hasta el momento en el que se obstruye el seguimiento de manera externa. Sin embargo, a partir de que se libera el robot el sistema es incapaz de volver a la trayectoria deseada en el punto en el que había comenzado la obstrucción, ya que se han perdido las referencias temporales.

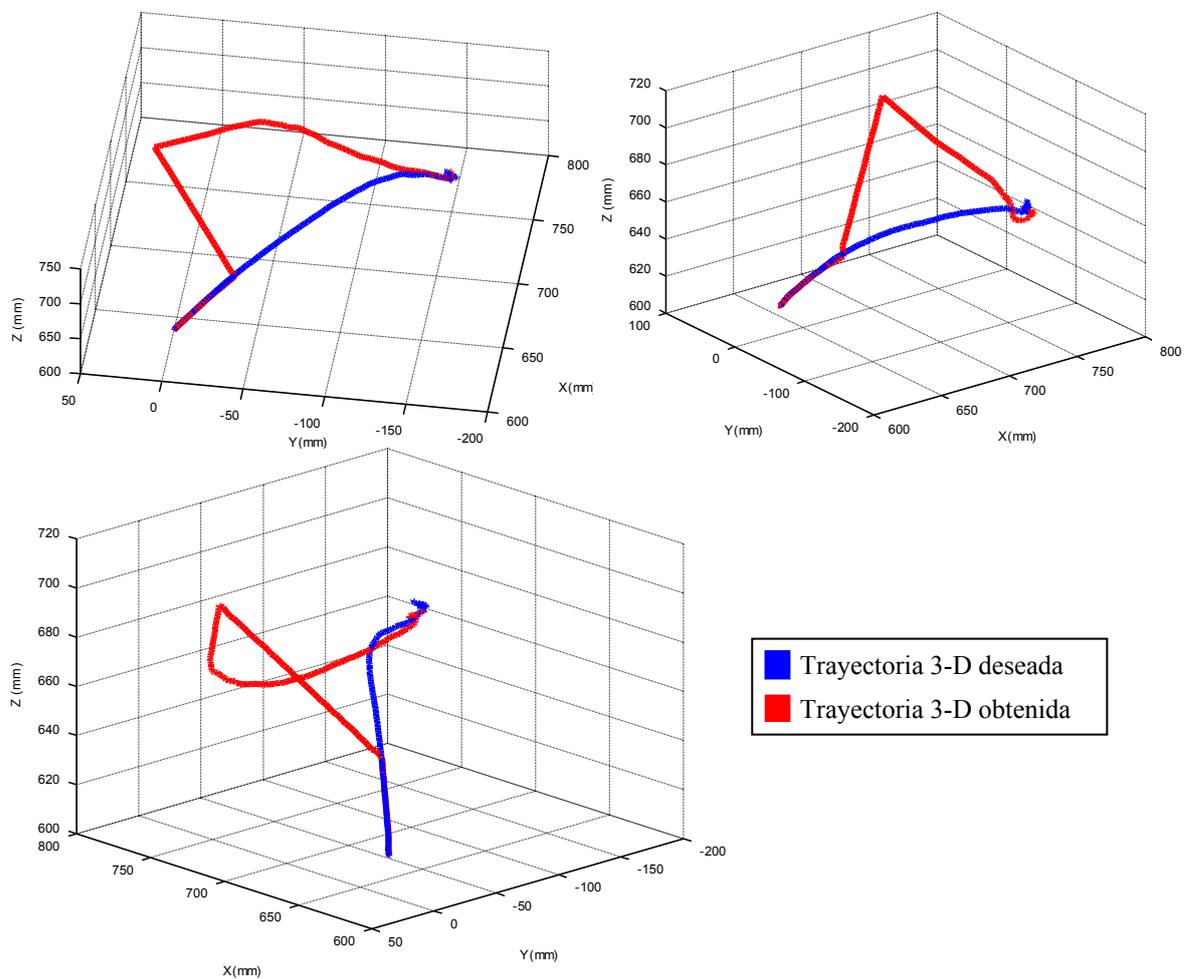


Figura 4-46. Trayectoria 3-D obtenida con sistemas dependientes del tiempo (Prueba 2).

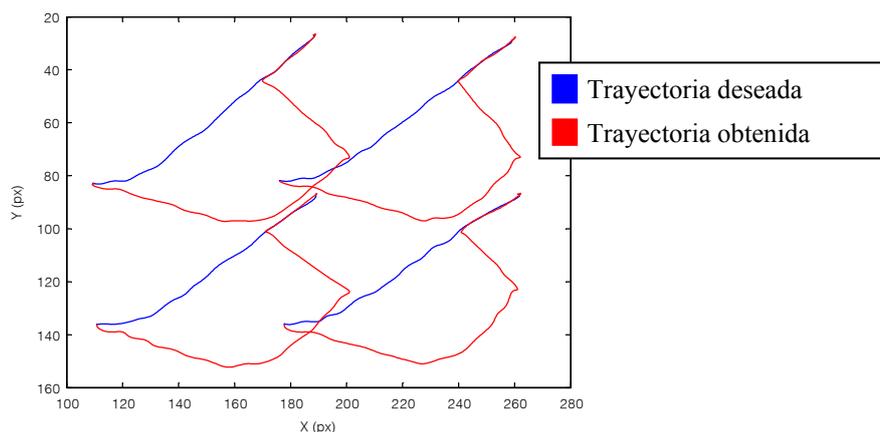


Figura 4-47. Evolución de las características en la imagen obtenida con seguimiento intemporal de trayectorias en la imagen basado en control visual virtual.

### 4.7.3. Comparativa con el control visual basado en flujo de movimiento

El método propuesto para el seguimiento de trayectorias en la imagen mejora ciertos aspectos del control visual basado en flujo de movimiento descrito en el apartado 4.3. La principal mejora radica en el hecho de tratar de forma conjunta la información de las características visuales observadas en una determinada posición de la cámara a lo largo de la trayectoria deseada. En el control visual basado en flujo de movimiento, las trayectorias se siguen de forma individual para cada característica. Esto obliga a retrasar el seguimiento de la trayectoria individual más rápida para que adecúe su velocidad a la del resto de trayectorias individuales. En el método propuesto, el seguimiento se realiza contemplando las características visuales como un único vector de características  $k_s$ . De forma que el sistema tratará de mover al robot hacia una posición válida de la cámara dentro de la trayectoria predefinida. Además, el nuevo método propuesto, permite especificar una velocidad de seguimiento. En el control visual basado en flujo de movimiento esto no es posible directamente. Para controlar la velocidad se tiene que modificar varios parámetros que tienen una influencia indirecta en el cambio de la velocidad, pero que a su vez introducen problemas de oscilaciones y error en el seguimiento en imagen y en 3D.

Para ilustrar las mejoras descritas, se ha realizado el seguimiento de una misma trayectoria con ambos métodos. La trayectoria es la mostrada en la Figura 4-48.

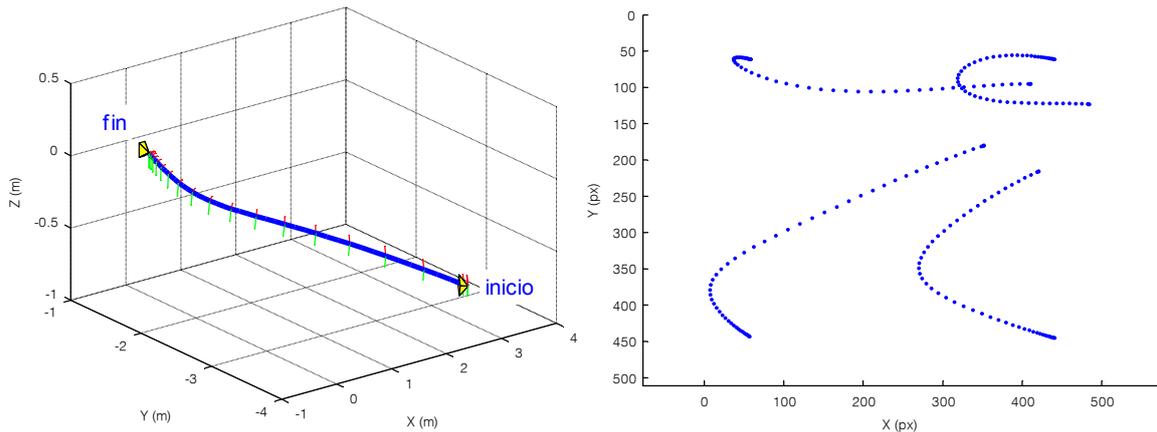


Figura 4-48. Muestreo de la trayectoria deseada en el espacio cartesiano 3-D y en la imagen.

En una primera prueba, se ejecuta primero el control visual basado en flujo de movimiento con una velocidad (controlada indirectamente a partir de otros parámetros) de aproximadamente 10 mm/s (ver Figura 4-49.a). El objetivo para compararlo con el método propuesto será realizar el seguimiento especificándole la misma velocidad de 10 mm/s (ver Figura 4-49.a).

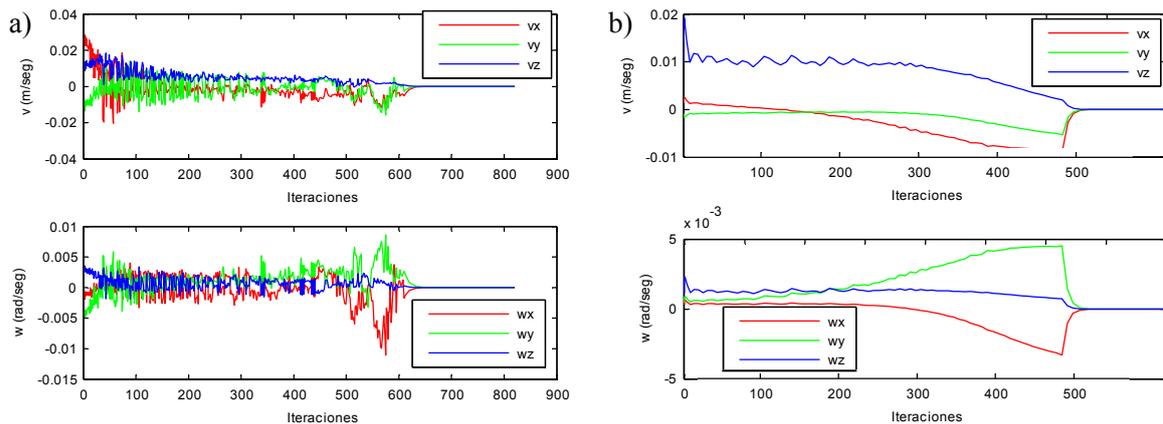


Figura 4-49. Evolución con el tiempo de la velocidad aplicada al robot. a) Control visual basado en flujo de movimiento. b) Método de seguimiento intemporal de trayectorias en la imagen basado en control visual virtual.

En la Figura 4-50 se muestra la evolución del módulo de la velocidad aplicada al robot al realizar la tarea de seguimiento de la trayectoria con ambos métodos. En rojo se representa la evolución del módulo de la velocidad usando control visual basado en flujo de movimiento. Se puede apreciar que el método no permite ajustar la velocidad de modo constante. El módulo de la velocidad es muy oscilante, aunque la media si se sitúa en esos 10 mm/s. En azul está representada la evolución del módulo durante el seguimiento con el método propuesto. Aquí sí se observa que se controla correctamente la velocidad del seguimiento, obteniendo los 10 mm/s deseados en régimen estacionario.

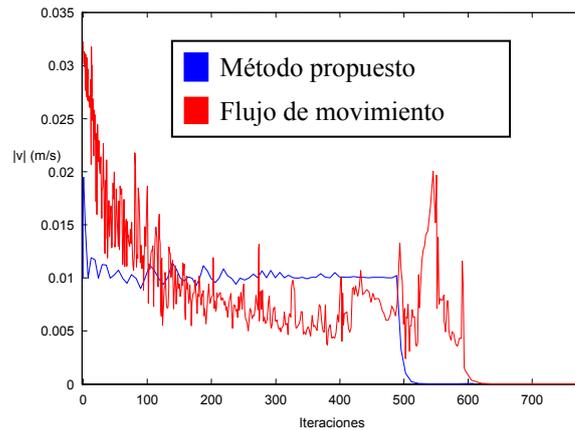


Figura 4-50. Evolución con el tiempo del módulo de la velocidad aplicada al robot.

Con esta velocidad, se obtiene un resultado aceptable tanto con el control visual basado en flujo de movimiento, como con el método propuesto de seguimiento de trayectorias en la imagen basado en control visual virtual. La Figura 4-51 muestra la trayectoria deseada en imagen, así como la evolución de las características en la imagen al utilizar ambos métodos para el seguimiento.

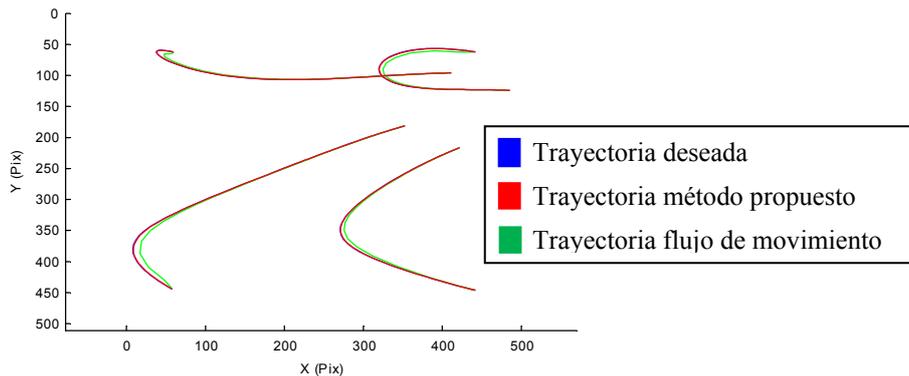


Figura 4-51. Comparativa entre la evolución de las características en la imagen con control visual basado en flujo de movimiento, con el método de seguimiento de trayectorias en la imagen basado en control visual virtual y la deseada.

El método propuesto consigue mayor precisión a la misma velocidad que el basado en flujo de movimiento. Este hecho se hace más patente al aumentar la velocidad del seguimiento. Con el objetivo de ilustrar este aspecto, se realiza la misma prueba modificando los parámetros del flujo de movimiento que permiten obtener una velocidad mayor durante el seguimiento. La velocidad, tal y como ocurría en la prueba anterior, es más oscilante en el caso de control visual basado en flujo de movimiento (ver Figura 4-52.a) que en el caso de usar el método propuesto (ver Figura 4-52.b). De nuevo, dado que con control visual basado en flujo de movimiento no se controla directamente la velocidad del seguimiento, se ejecuta primero este método para, una vez ha concluido la tarea, repetir la prueba con la velocidad con la que se ha realizado el seguimiento con flujo de movimiento.

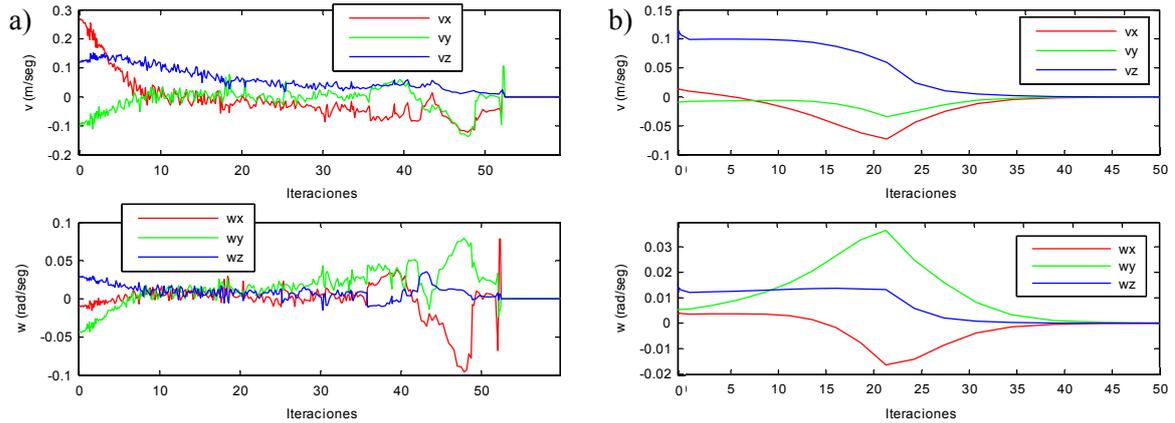


Figura 4-52. Evolución con el tiempo de la velocidad aplicada al robot. a) Control visual basado en flujo de movimiento. b) Método de seguimiento intemporal de trayectorias en la imagen basado en control visual virtual.

En la Figura 4-53 se muestra la evolución del módulo de la velocidad obtenido. Se observa que con el método propuesto se consigue controlar perfectamente la velocidad a 100 mm/s. Con control visual basado en flujo de movimiento se obtiene aproximadamente esa velocidad, pero como se observa, de una manera muy oscilante.

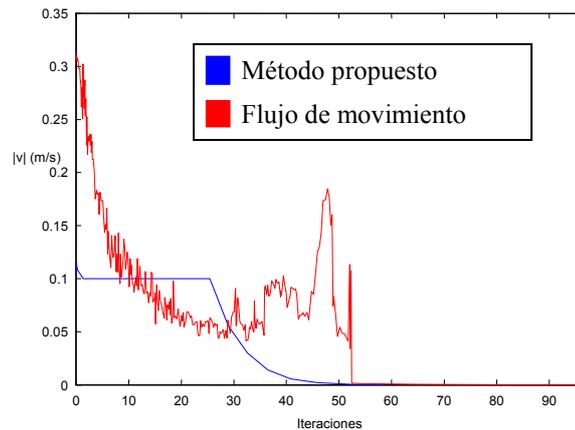


Figura 4-53. Evolución con el tiempo del módulo de la velocidad aplicada al robot.

La evolución en la imagen (ver Figura 4-54), sin embargo, parece mejor en el seguimiento realizado con control visual basado en flujo de movimiento. Sin embargo, tal como muestra la Figura 4-55, la trayectoria 3D de la cámara muestra claramente que el método propuesto se aproxima mucho más a la trayectoria 3D deseada a tan altas velocidades. El control visual basado en flujo de movimiento se vuelve demasiado oscilante, y al no controlar la velocidad directamente, se debe sacrificar precisión para conseguir elevar la velocidad. El resultado es un seguimiento mucho menos preciso de la trayectoria deseada, en más iteraciones y con ningún control sobre la velocidad del seguimiento.

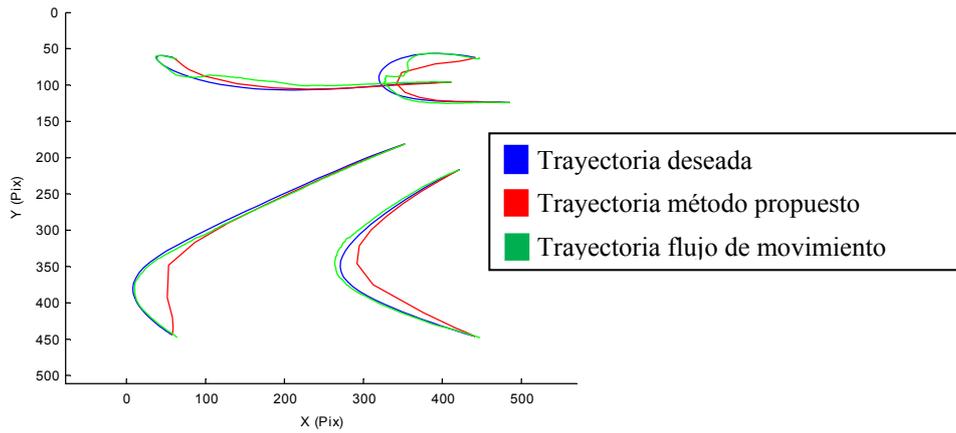


Figura 4-54. Comparativa entre la evolución de las características en la imagen con control visual basado en flujo de movimiento, con el método de seguimiento de trayectorias en la imagen basado en control visual virtual y la deseada.

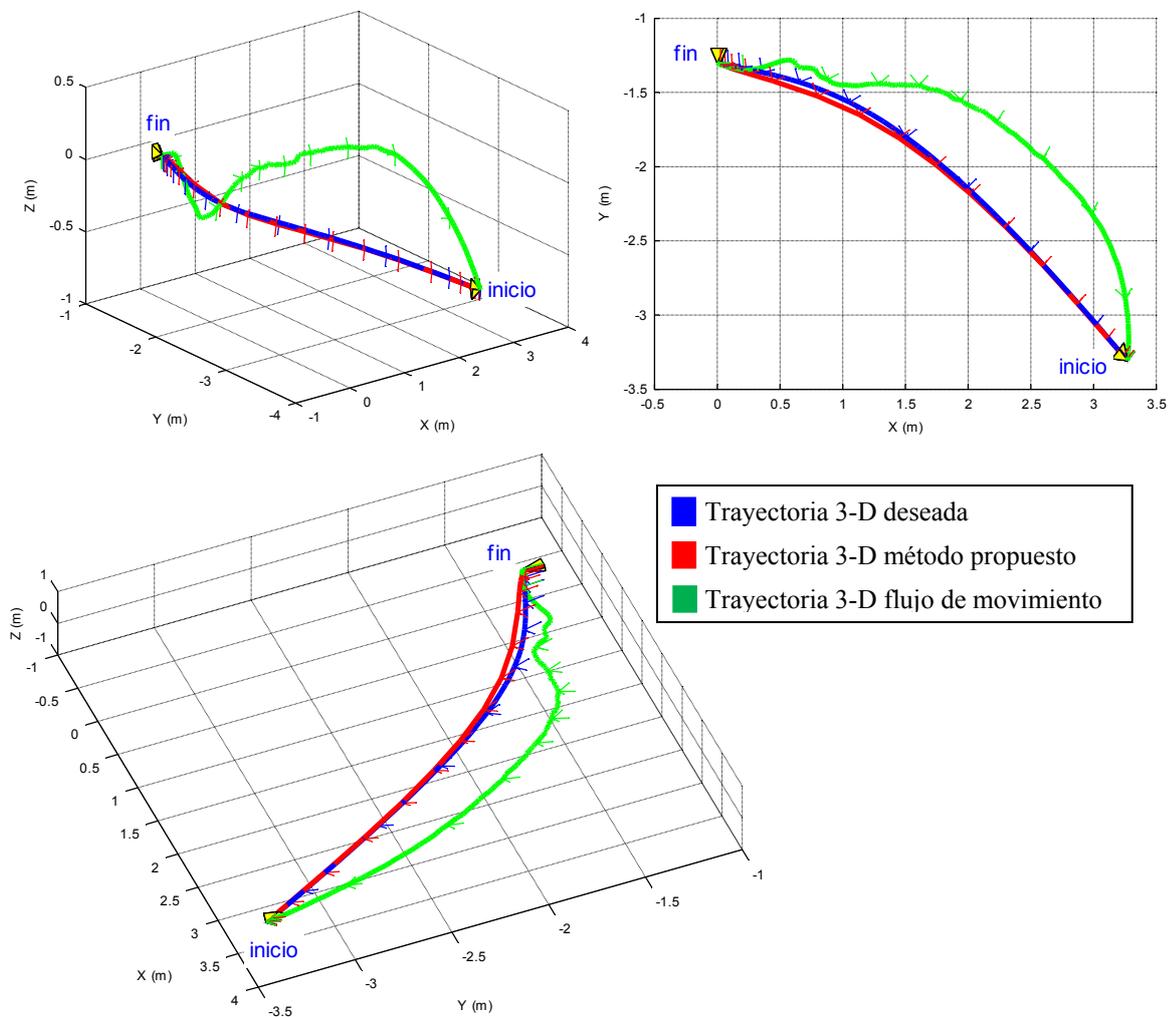


Figura 4-55. Distintas vistas de la trayectoria descrita por la cámara en el espacio Cartesiano 3-D.

#### 4.7.4. Estudio del comportamiento frente a variación de los parámetros del sistema

En este apartado se va a analizar la respuesta del sistema ante distintos valores de los parámetros que definen su comportamiento. Dos son los parámetros principales que definen el seguimiento de trayectorias en el espacio imagen utilizando el método propuesto: la velocidad deseada durante el seguimiento, y el muestreo de posiciones de la cámara en la trayectoria deseada.

##### 4.7.4.1. Estudio de la respuesta del sistema ante variaciones en la velocidad

Para analizar el comportamiento del sistema ante variaciones de este parámetro, en la Figura 4-56 se muestra la trayectoria obtenida en la imagen según se aumenta el valor de la velocidad. En esta figura se observa que según se aumenta la velocidad de seguimiento el error obtenido va siendo mayor. Esto es lógico, ya que al aumentar la velocidad se sacrifica precisión.

En la Figura 4-57 se ha representado la trayectoria 3D de la cámara para distintas velocidades. Para velocidades superiores a los 40 mm/s se empieza a observar pérdida en la precisión del seguimiento. Sin embargo, estas velocidades son muy altas, y el comportamiento es más que aceptable.

Finalmente, en la Figura 4-58 se muestra las velocidades aplicadas al robot en los distintos experimentos realizados. Se observa que la velocidad es controlada satisfactoriamente, a la vez que el número de iteraciones va disminuyendo conforme se sube la velocidad deseada.

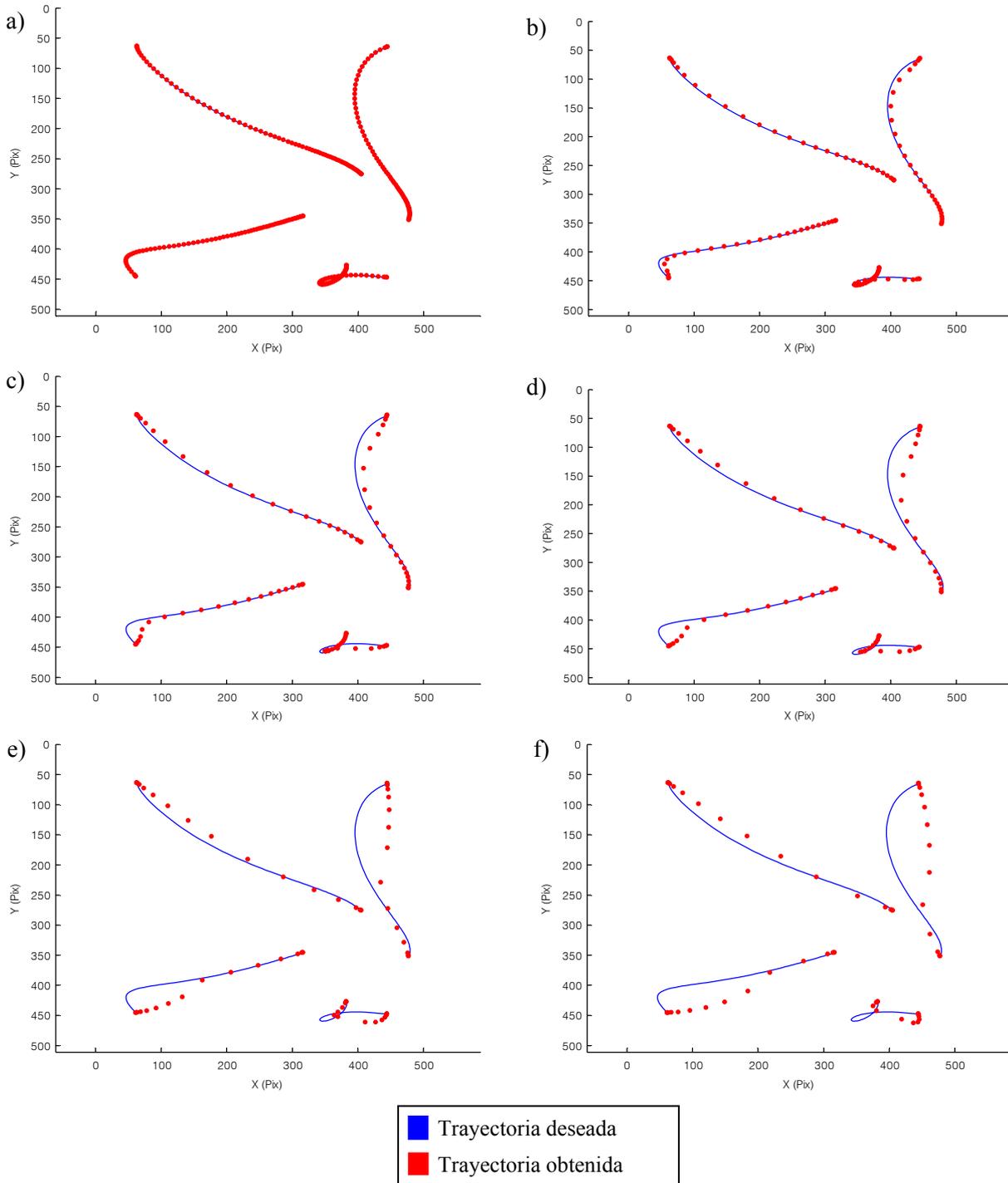


Figura 4-56. Resultados obtenidos para el seguimiento de trayectorias modificando la velocidad de seguimiento. a)  $|v|=10\text{mm/s}$ . b)  $|v|=40\text{mm/s}$ . c)  $|v|=70\text{mm/s}$ . d)  $|v|=100\text{mm/s}$ . e)  $|v|=200\text{mm/s}$ . f)  $|v|=400\text{mm/s}$ .

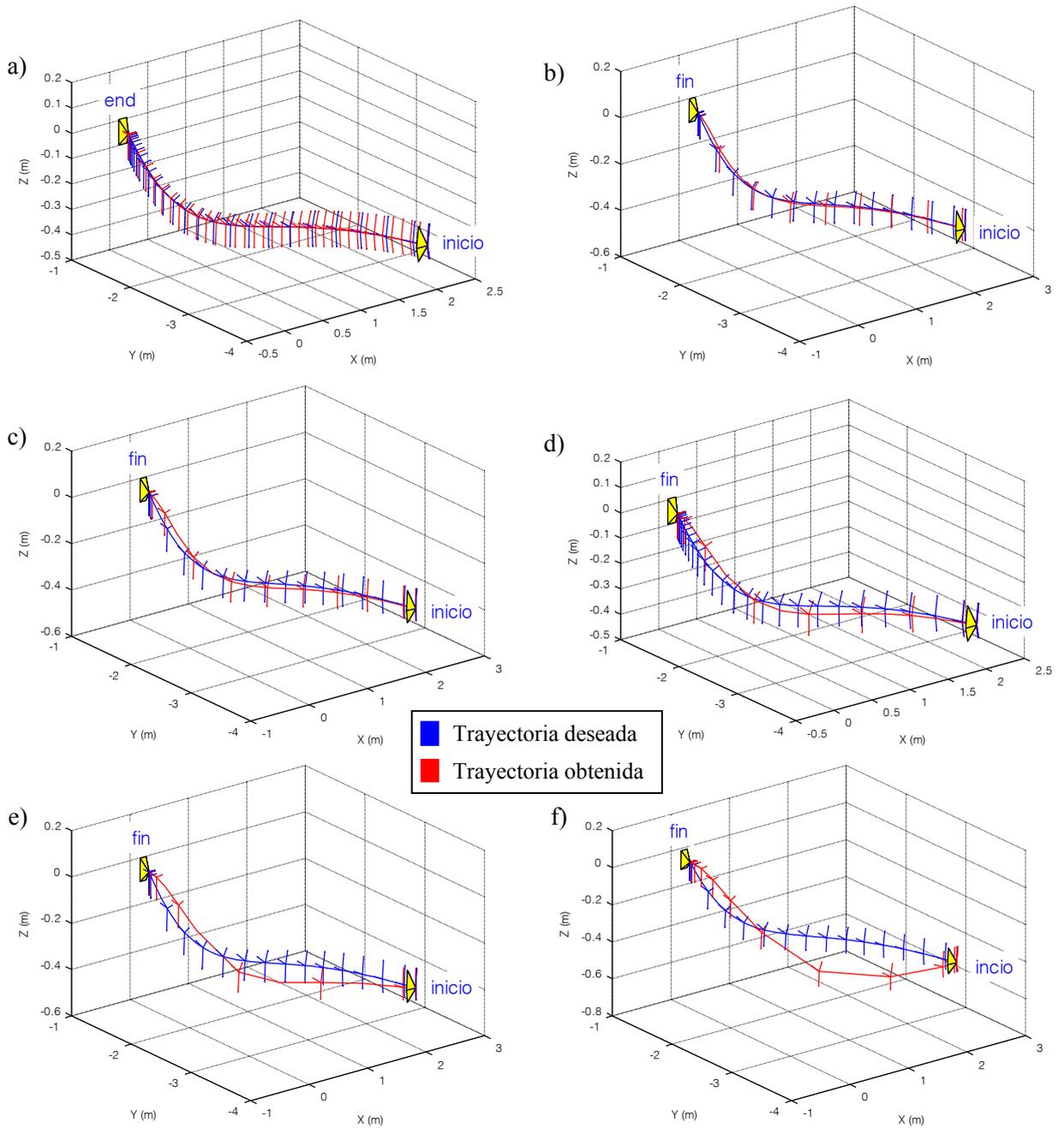


Figura 4-57. Seguimiento de trayectorias en el espacio 3-D para distintos valores de la velocidad de seguimiento. a)  $|v|=10\text{mm/s}$ . b)  $|v|=40\text{mm/s}$ . c)  $|v|=70\text{mm/s}$ . d)  $|v|=100\text{mm/s}$ . e)  $|v|=200\text{mm/s}$ . f)  $|v|=400\text{mm/s}$ .

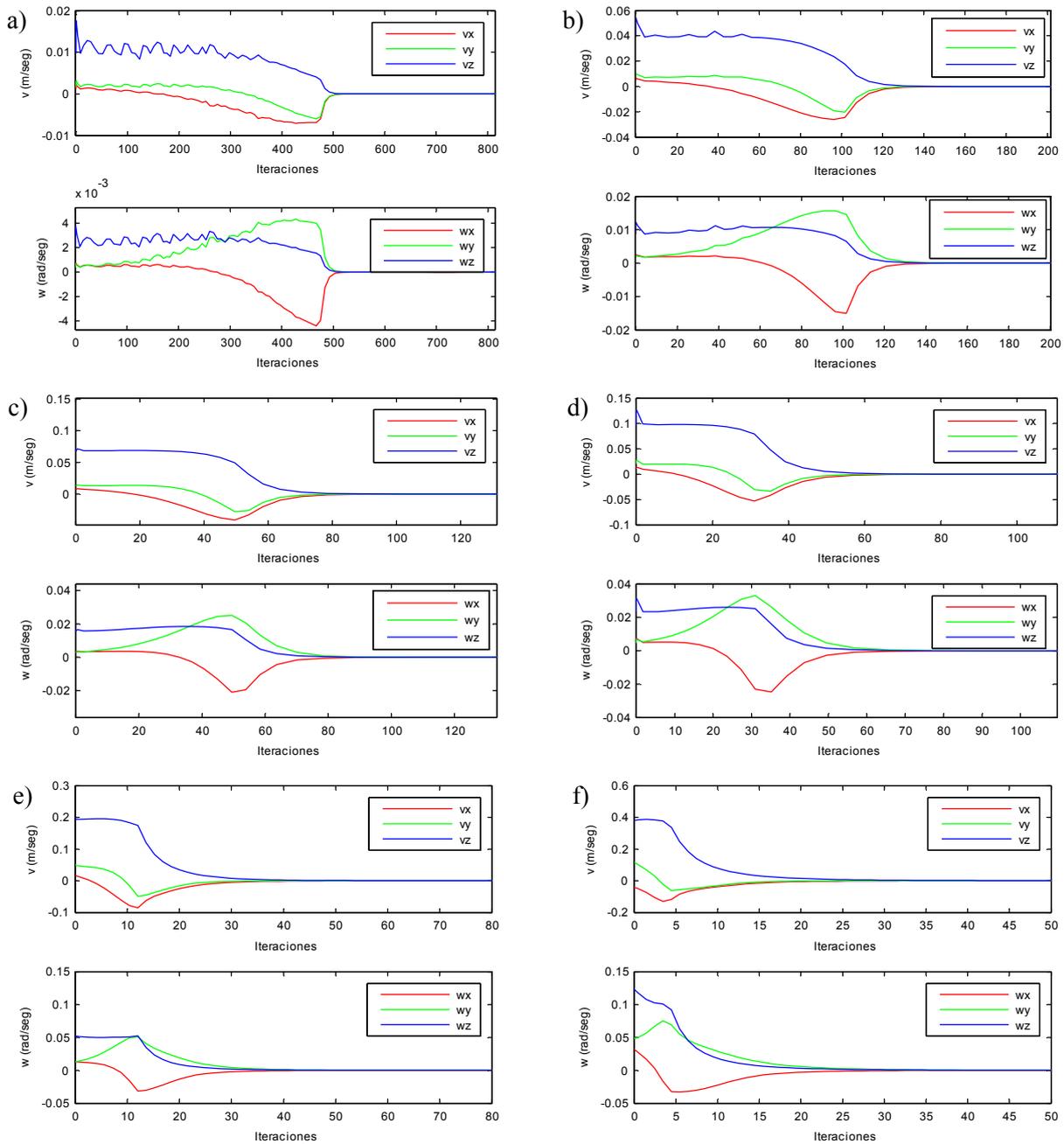


Figura 4-58. Evolución de la velocidad aplicada al robot para distintos valores de la velocidad de seguimiento. a)  $|v|=10\text{mm/s}$ . b)  $|v|=40\text{mm/s}$ . c)  $|v|=70\text{mm/s}$ . d)  $|v|=100\text{mm/s}$ . e)  $|v|=200\text{mm/s}$ . f)  $|v|=400\text{mm/s}$ .

#### 4.7.4.2. Estudio de la respuesta del sistema ante variaciones en el muestreo de la trayectoria deseada

En el apartado anterior se ha ilustrado el efecto de cambiar la velocidad deseada durante el seguimiento de la trayectoria dada. Otro parámetro que permite obtener distintas trayectorias, tanto en imagen como en 3D es el muestreo de la trayectoria deseada. El sistema es capaz de muestrear la trayectoria deseada para realizar el seguimiento. Hasta ahora, en todos los experimentos representados

usando el método propuesto se han mostrado trayectorias muestreadas tal y como se definieron en el apartado 4.2. En la Figura 4-59 se muestra la evolución de las características en la imagen para distinto muestreo de la trayectoria deseada. El muestreo original es de 1 muestra cada 3,5 cm en la trayectoria 3D. Se observa que al disminuir el número de muestras se obtiene menor precisión en el seguimiento de la trayectoria deseada.

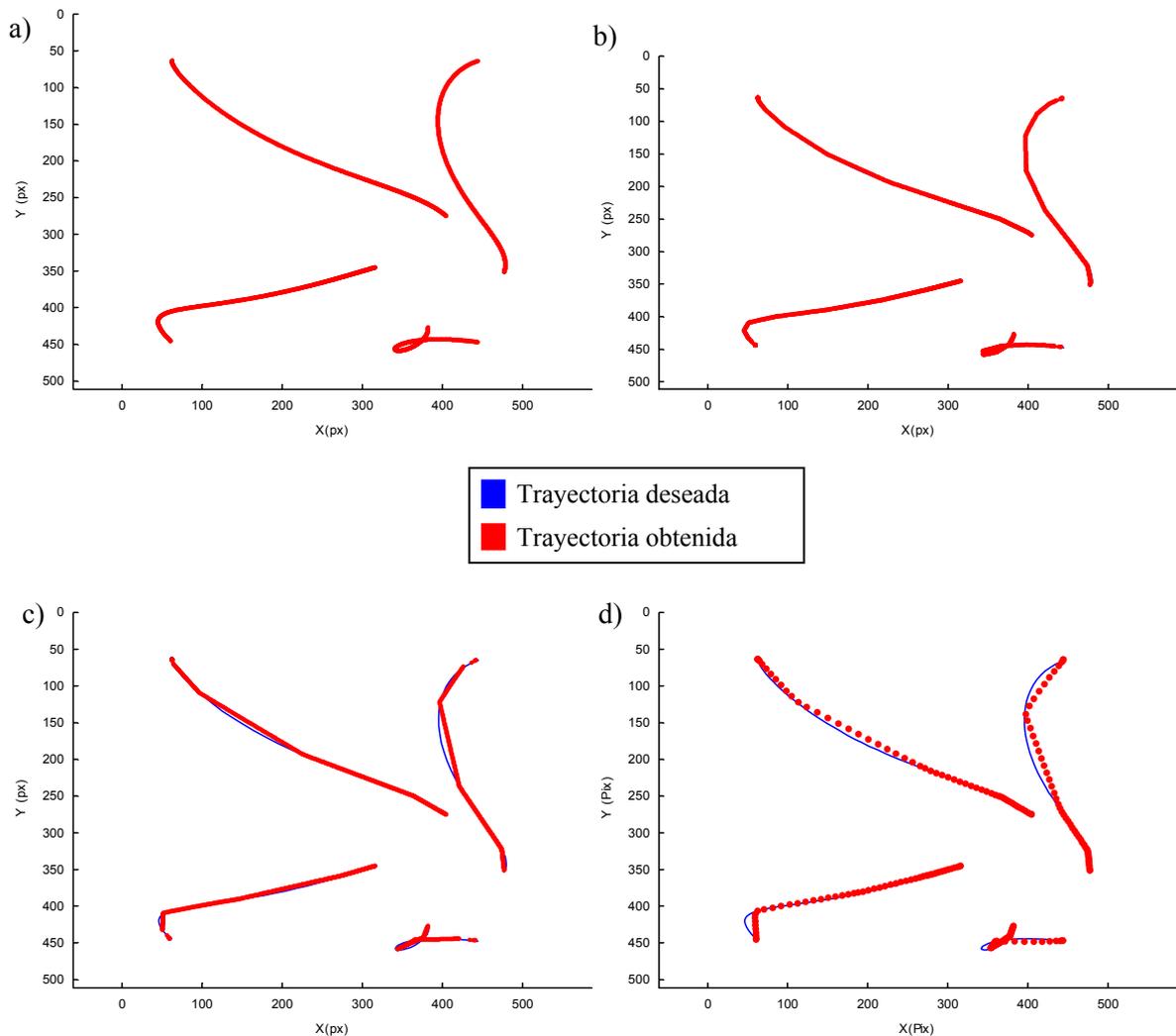


Figura 4-59. Evolución de las características en la imagen para: a) Muestreo original. b) 1 de cada 5 muestras. c) 1 de cada 10 muestras. d) 1 de cada 20 muestras.

En la Figura 4-60 se muestra la trayectoria 3D de la cámara cuando se selecciona 1 muestra de cada 10 en la trayectoria deseada en la imagen. La trayectoria 3D mantiene su paso por la trayectoria deseada en 1 de cada 10 muestras, realizando una interpolación en 3D (tanto en posición como en orientación) de la cámara entre muestra y muestra. El resultado es una trayectoria 3D válida muy próxima a la deseada, pero con menor número de iteraciones, como se puede ver en la Figura 4-61, donde se muestra la evolución del módulo de la velocidad para distintas pruebas donde se ha modificado el número de muestras de la trayectoria deseada.

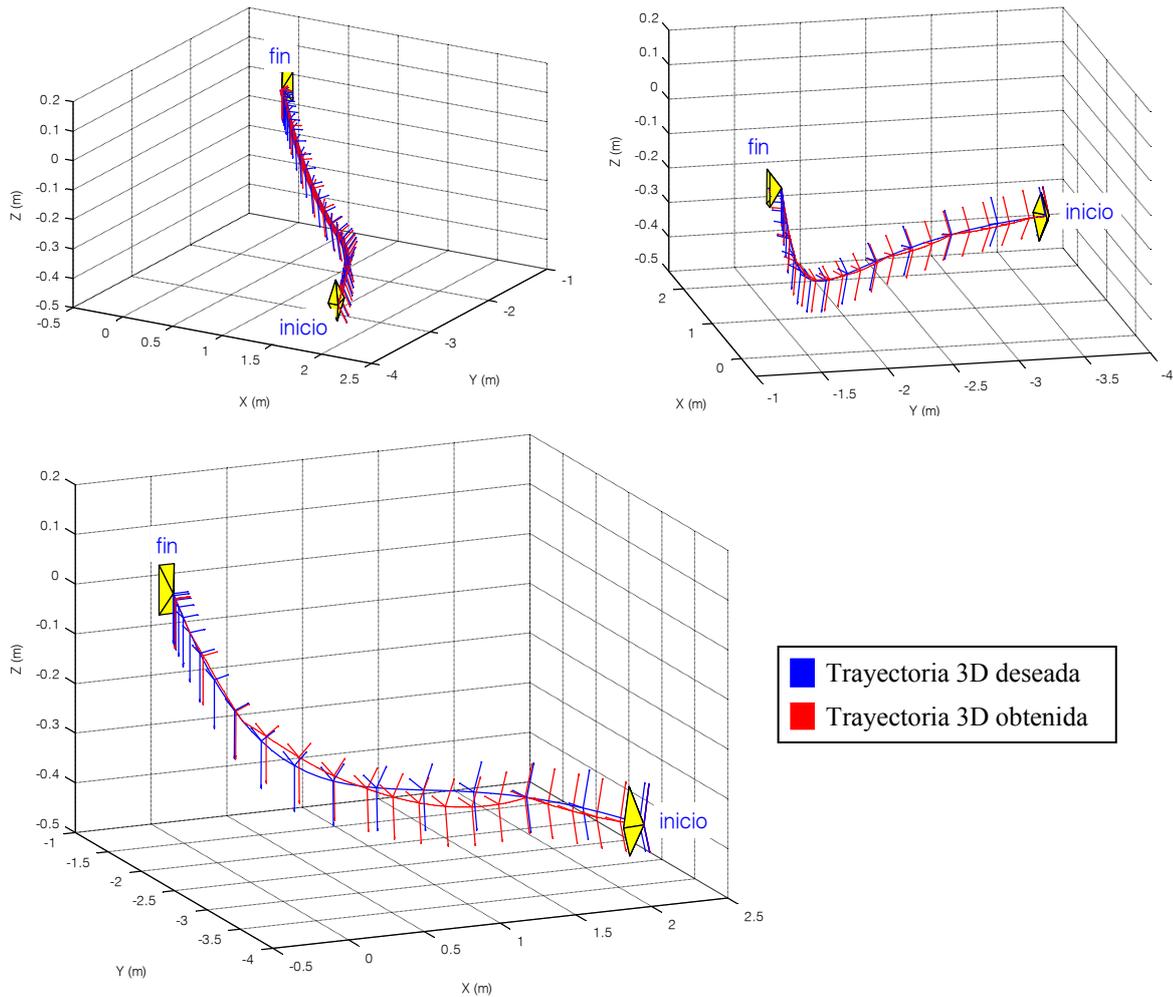


Figura 4-60. Distintas vistas de la trayectoria descrita por la cámara en el espacio Cartesiano 3-D para 1 de cada 10 muestras.

En la Figura 4-61 se ha representado en azul la evolución del módulo de la velocidad para el seguimiento de la trayectoria original, es decir, sin modificar el número de muestras. Las distintas gráficas representadas en la Figura 4-61 muestran la reducción en el número de iteraciones del seguimiento al disminuir el número de muestras de la trayectoria deseada. Al mismo tiempo permite comprobar que el módulo se mantiene constante en torno a los 10 mm/s.

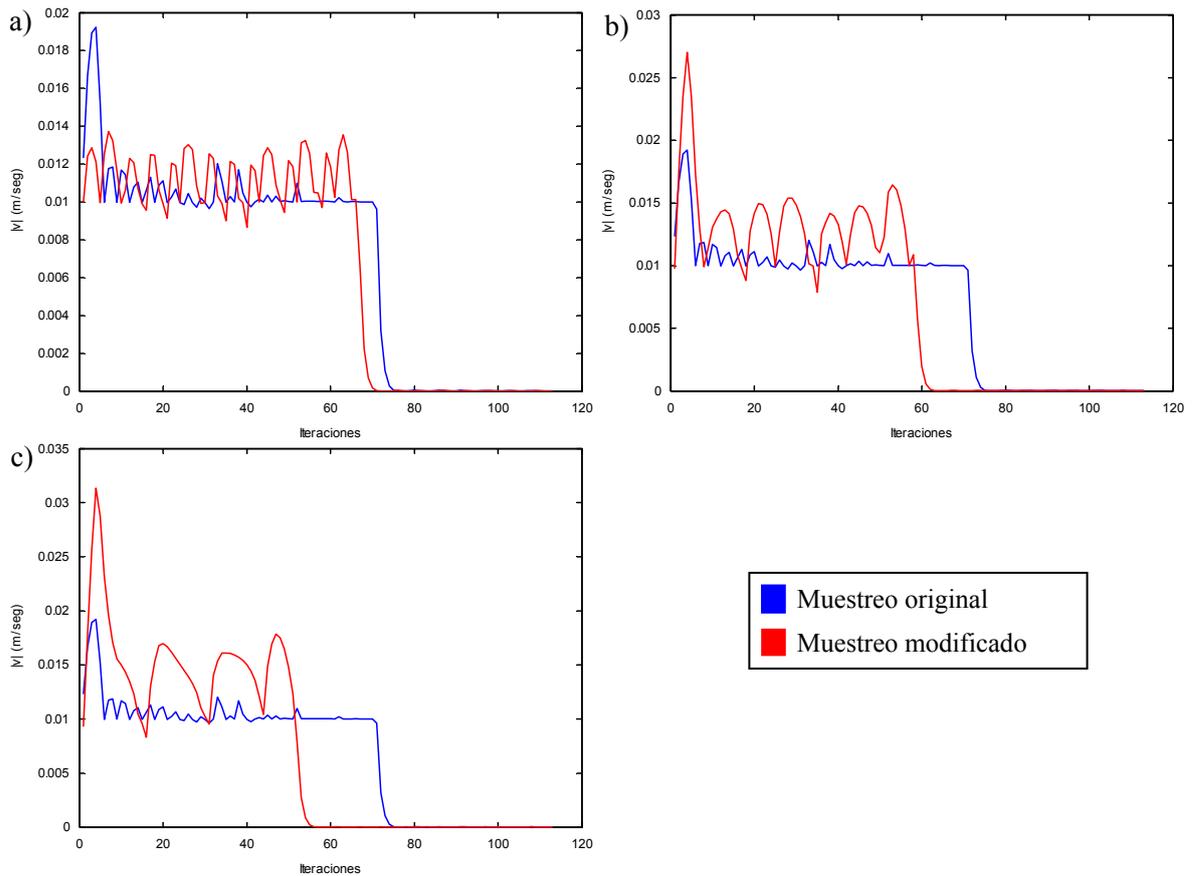


Figura 4-61. Evolución del módulo de la velocidad para: a) 1 de cada 5 muestras. b) 1 de cada 10 muestras. c) 1 de cada 20 muestras.

En la Figura 4-62 se ha representado para distinto muestreo de la trayectoria deseada la evolución del error en la imagen durante el seguimiento. Estas gráficas ilustran lo que se comentaba anteriormente de la disminución de la precisión al disminuir el número de muestras.

El seguimiento intemporal de trayectorias en la imagen basado en control visual virtual permite modificar el número de muestras de la trayectoria que se va a seguir. La modificación de este parámetro es, quizás, menos interesante que la modificación de la velocidad. El resultado es un seguimiento en menos iteraciones, con la consecuente pérdida de precisión, ya que se pierde información en la pérdida de muestras.

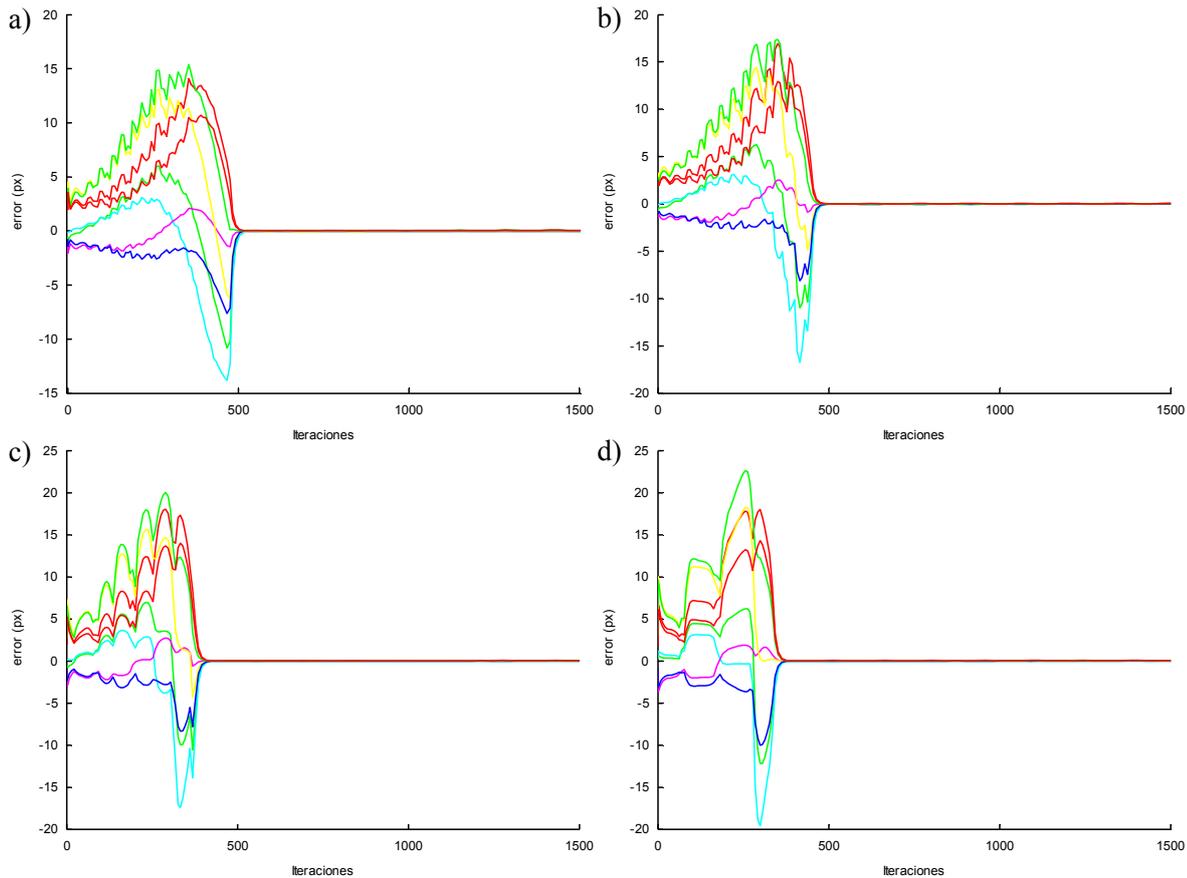


Figura 4-62. Evolución del error en la imagen durante el seguimiento para: a) Muestreo original. b) 1 de cada 5 muestras. c) 1 de cada 10 muestras. d) 1 de cada 20 muestras.

## 4.8. Conclusiones

En el presente capítulo, se ha presentado un nuevo método que permite realizar el seguimiento de una trayectoria previamente definida en el espacio imagen usando control visual. La mayoría de los métodos propuestos hasta la fecha para realizar esta tarea no tienen en cuenta posibles interacciones con el entorno, ya que presentan un comportamiento temporal. La obstrucción de la trayectoria no impide seguir enviando referencias al sistema y cuando consigue evitar el obstáculo o éste desaparece, el sistema no será capaz de seguir la trayectoria desde el punto en el que la dejó. Para resolver este problema se ha descrito el control visual basado en flujo de movimiento. Esta técnica sí que presenta un comportamiento intemporal y, por tanto, permite al sistema interactuar con el entorno completando con éxito la tarea de seguimiento de una trayectoria en el espacio imagen. Sin embargo, como se ha presentado en este capítulo, el flujo de movimiento presenta un problema fundamental, que es el de no tratar el seguimiento del objeto en el espacio imagen a partir de todas sus características en conjunto. El flujo de movimiento se calcula de forma individual para cada punto característico, de ahí que la

evolución en el espacio imagen de cada punto pueda tener distinta velocidad. Esto llevaría a casos en los que el sistema se encontrara en una configuración 3D de la cámara imposible.

Previamente a la propuesta realizada para el seguimiento de trayectorias en el espacio imagen, se ha propuesto un seguimiento intemporal basado en el cálculo de una norma de la matriz de homografía 2D. Este método presenta un comportamiento intemporal, y surge de la necesidad de agrupar las características en un solo parámetro que evite los problemas descritos en la sección 4.3 con el flujo de movimiento. Sin embargo, tras realizar distintos experimentos, se concluye que el método no es lo suficiente robusto y se plantea un nuevo método para realizar el seguimiento de las trayectorias en la imagen.

El método propuesto permite resolver los dos problemas. Por un lado presenta un comportamiento intemporal, lo que lo hace especialmente adecuado para tareas en las que haya interacción con el entorno. Por otro lado, este nuevo sistema de seguimiento de trayectorias en la imagen basado en control visual virtual resuelve el problema que presentaba el flujo de movimiento de tratar las características por separado. Agrupando las características durante el seguimiento se consigue obtener siempre una configuración válida en el espacio cartesiano 3D. Además, este nuevo método propuesto de seguimiento de trayectorias en la imagen permite mantener una velocidad constante durante la tarea.

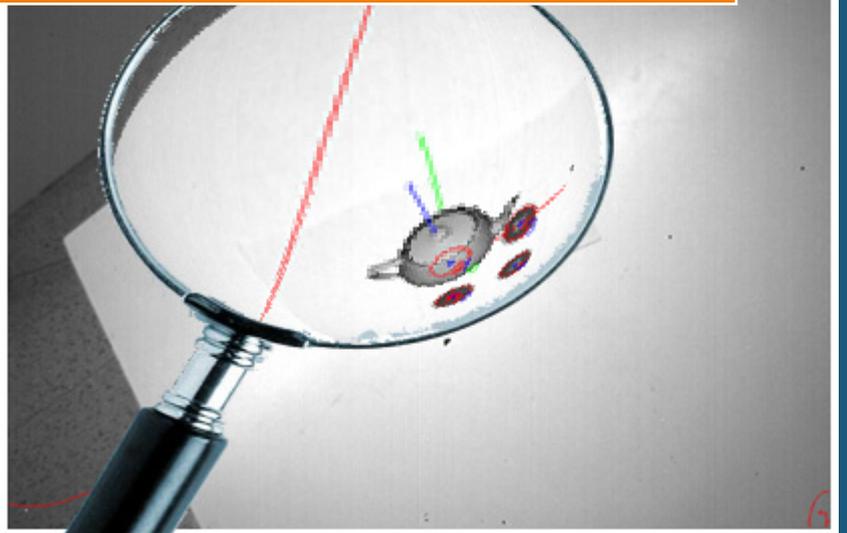
Como consecuencia de la utilización de esta información no dependiente del tiempo, se concluye un aspecto fundamental de la estrategia descrita en este capítulo que es el hecho de que permite el seguimiento de trayectorias especificadas sin tener en consideración el tiempo. Como han mostrado los resultados obtenidos, el uso del sistema propuesto permite realizar el seguimiento de trayectorias en el espacio de la imagen, evitando las limitaciones y problemas existentes en los sistemas de control visual tanto dependientes como independientes del tiempo que han sido utilizados hasta la actualidad para el seguimiento de trayectorias en el espacio de la imagen.

A partir de las pruebas realizadas, se puede concluir que con el sistema propuesto se obtiene no solo un seguimiento correcto en la imagen, sino también en el espacio Cartesiano 3-D, asegurando una velocidad de seguimiento constante.



# Capítulo 5

## Seguimiento de trayectorias con control visual adaptativo



## 5.1. Introducción

---

En el Capítulo 4 se ha presentado un método para el seguimiento de trayectorias en la imagen generadas a partir de una determinada trayectoria 3D. A lo largo de la trayectoria 3D seguida por el robot puede darse el caso de variaciones en los parámetros intrínsecos de la cámara situada en el extremo del robot. En este Capítulo se va a proponer un método para realizar el seguimiento de la trayectoria deseada incluso en situaciones como la descrita en las que se modifican los parámetros intrínsecos a lo largo de la tarea a realizar. Esto es posible gracias al uso del control visual virtual que permite obtener tanto los parámetros extrínsecos como los intrínsecos de la cámara a partir de la imagen captada por ésta. Este método también permitiría realizar el seguimiento de trayectorias cuando la cámara presenta un zoom variable, lo que provoca una modificación de los intrínsecos durante la tarea.

Como se pudo comprobar en los resultados del Capítulo 3, el control visual virtual permite calcular los parámetros extrínsecos de una cámara a partir de la información visual obtenida de una imagen captada por ésta. Sin embargo, las pruebas realizadas demostraban que, aunque se obtenían unos parámetros intrínsecos cercanos a los reales, su resultado dependía de la inicialización de la posición de la cámara virtual. Además, durante la calibración, se podían obtener valores de los intrínsecos imposibles o muy lejanos a los reales [Pomares-2007a].

En el presente capítulo se presenta un método que permite obtener los parámetros intrínsecos de forma precisa a partir de más de una imagen, incrementando la robustez de los sistemas de control visual virtual clásicos.

En [Malis-2004a] se propone un sistema de control visual invariante a posibles cambios en los parámetros intrínsecos de la cámara. Sin embargo, usando esta aproximación la tarea se tiene que especificar como una imagen deseada observada desde la cámara y realmente no se calculan los parámetros intrínsecos durante la tarea. A pesar de que los sistemas de control visual son robustos con respecto a errores en los parámetros intrínsecos, se obtiene un mejor comportamiento si estos parámetros son estimados correctamente (los parámetros intrínsecos son necesarios para estimar el Jacobiano que relaciona la velocidad de la cámara y las variaciones de las características en el espacio invariante). En este Capítulo se describe un sistema de control visual que aprovecha las imágenes capturadas durante la tarea para determinar los parámetros internos de la cámara. Este método permite completar la tarea, no sólo sin ningún conocimiento previo de la calibración de la cámara, sino también cuando los parámetros intrínsecos varían durante la tarea.

En las últimas décadas, los métodos de calibración de las cámaras han sido ampliamente investigados [ZhangZ-1999,Tsai-1987]. Entre estos métodos, se han propuesto sistemas de autocalibración [Hemayed-2003]. La mayoría de ellos referentes a parámetros intrínsecos desconocidos pero constantes. En este tipo de sistemas es a menudo complicado obtener buenos resultados y el comportamiento depende de la escena observada. En el presente Capítulo se propone el uso de la información de la imagen obtenida durante la ejecución de la tarea de control visual para calibrar la cámara. Consiguiendo esto, el comportamiento del control visual se mejora al mismo tiempo.

## 5.2. Combinando Control Visual y Control Visual Virtual

Con el objetivo de aumentar la precisión de los sistemas de control visual virtual clásicos y posibilitar su utilización para determinar los parámetros intrínsecos, en este apartado se propone la utilización conjunta de control visual virtual y control visual. Para ello, en cada iteración de un sistema de control visual se ejecutará una convergencia completa de control visual virtual. Además, para dotar de más robustez al sistema, se propone la utilización en la calibración de las últimas  $w$  imágenes, constituyendo lo que se ha denominado calibración multi-imagen.

Una tarea robótica puede describirse con una función que debe ser regulada a cero. La función de tarea  $\mathbf{e}_t$  se define como:

$$\mathbf{e}_t = \mathbf{s} - \mathbf{s}^* \quad (5.1)$$

donde  $\mathbf{s}$  es un vector que contiene  $k$  características visuales correspondientes con el estado actual, mientras que  $\mathbf{s}^*$  contiene los valores de las características visuales en la posición deseada.  $\mathbf{s}^*$  no sólo depende de la posición deseada de la cámara,  ${}^c\mathbf{M}_{od}$ , sino que también depende de los parámetros intrínsecos,  $\xi$ . Con el objetivo de cumplir con la tarea  $\mathbf{e}_t$  se puede definir una ley de control simple. Imponiendo un decrecimiento exponencial de la función de tarea  $\mathbf{e}_t$  ( $\dot{\mathbf{e}}_t = -\lambda_2 \mathbf{e}_t$ ) (siendo  $\lambda_2$  una ganancia positiva) se obtiene:

$$\mathbf{v}_c = -\lambda_2 \hat{\mathbf{L}}_s^+ (\mathbf{s} - \mathbf{s}^*) \quad (5.2)$$

donde  $\hat{\mathbf{L}}_s^+$  es la pseudoinversa de una estimación de la matriz de interacción. En la estimación de matriz de interacción,  $\hat{\mathbf{L}}_s$  se usan los resultados del método de calibración descrito en el Capítulo 3.

Por lo tanto, en cada iteración,  $i$ , de la tarea de control visual, se ejecuta una convergencia completa del sistema de control visual virtual descrito en el Capítulo 3. En el control visual virtual las características deseadas,  $\mathbf{p}_d$ , son las observadas en el instante  $i$  ( $\mathbf{p}_d = \mathbf{s}$ ). Las iniciales,  $\mathbf{p}$ , son calculadas por retroproyección usando los parámetros extrínsecos e intrínsecos determinados en el instante  $i-1$ . En cada iteración de la tarea de control visual, los resultados de la calibración se usan para actualizar  $\hat{\mathbf{L}}_s$  y  $\mathbf{s}^*$ .

### 5.3. Calibración Multi-Imagen

Como se muestra en [Marchand-2001] cuando se emplea control visual virtual para la calibración de una cámara, los parámetros intrínsecos obtenidos pueden variar dependiendo del punto de vista. Una posible solución es integrar en el proceso de minimización información procedente de diferentes imágenes adquiridas desde diferentes cámaras virtuales. Para conseguirlo, si la calibración de la cámara se realiza durante el proceso de control visual, se pueden usar en la calibración distintas imágenes obtenidas durante el movimiento real de la cámara. Considérese que se usan  $w$  imágenes. El conjunto de  $w$  imágenes también se llama ventana de calibración. Considérese también que los parámetros intrínsecos no varían o que en las  $w$  imágenes la variación de los parámetros intrínsecos puede considerarse nula.

La matriz de interacción considerada para la calibración multi-imagen será:

$$\mathbf{H}_p = \begin{pmatrix} \frac{\partial \mathbf{p}^1}{\partial \mathbf{r}} & 0 & \dots & 0 & \frac{\partial \mathbf{p}^1}{\partial \xi} \\ 0 & \frac{\partial \mathbf{p}^2}{\partial \mathbf{r}} & \dots & 0 & \frac{\partial \mathbf{p}^2}{\partial \xi} \\ \vdots & \vdots & & & \vdots \\ 0 & \dots & 0 & \frac{\partial \mathbf{p}^w}{\partial \mathbf{r}} & \frac{\partial \mathbf{p}^w}{\partial \xi} \end{pmatrix} \quad (5.3)$$

Por lo tanto, en este caso, la acción de control y error de la ecuación (3.7) será  $\mathbf{V} = (\mathbf{T}_c^1 \quad \mathbf{T}_c^2 \quad \dots \quad \mathbf{T}_c^w \quad \xi)^T$ , y  $\mathbf{e} = (\mathbf{p}^1 - \mathbf{p}_d^1 \quad \mathbf{p}^2 - \mathbf{p}_d^2 \quad \dots \quad \mathbf{p}^w - \mathbf{p}_d^w)^T$  respectivamente.

Un aspecto importante de la calibración multi-imagen es que permite la convergencia en situaciones en las que la calibración mono-imagen no podía proporcionar los parámetros intrínsecos correctos. Para demostrarlo, se considera el experimento presentado en la Sección 3.3.2 y se incrementa el tamaño de la ventana de calibración. La Tabla 5-1 muestra los parámetros intrínsecos obtenidos y el número de iteraciones necesarias para la convergencia dependiendo del tamaño de la

ventana de calibración. Esta tabla presenta también el porcentaje de error en los parámetros extrínsecos obtenidos. En esta tabla se puede ver que cuanto mayor sea la ventana de calibración más iteraciones se necesitan para la convergencia, sin embargo, la calibración es más precisa. De esta tabla se puede concluir que es necesario un tamaño de 6 de la ventana de calibración para obtener buenos resultados. Tal y como se describió en el Capítulo 3, se ha empleado una cámara eye-in-hand que ha sido sometida a un proceso previo de calibración para estimar los parámetros intrínsecos. Usando el método de Zhang [ZhangZ-1999] se han obtenido los siguientes parámetros intrínsecos: las coordenadas en píxeles del punto principal son  $(u_0, v_0) = (168.75, 121.5)$ , y la distancia focal en las direcciones  $u$  y  $v$  son  $(412.92, 423.69)$  respectivamente. Se han empleado 10 imágenes para hacer la calibración.

Tamaño de la ventana de calibración	Parámetros intrínsecos $(f_u, f_v), (u_0, v_0)$	% error extrínsecos	Iteraciones
1	(905.661, 1053.23), (-203.843, -149.264)	3181.2	40
2	(7.89, 757.101), (-523.47, 204.48)	561	55
3	(409.96, 420.96), (166.8, 120.8)	41.6623	50
4	(410.38, 421.43), (164.37, 121.15)	14.4884	70
5	(411.58, 422.88), (166.73, 120.3)	8.15	90
6	(412.74, 423.28), (168.43, 121.33)	3.1	100
7	(412.93, 423.65), (168.54, 120.93)	3.6	110
8	(412.98, 423.78), (169.03, 121.55)	4.2	120

Tabla 5-1. Efecto de incrementar el número de imágenes para realizar la calibración.

## 5.4. Seguimiento intemporal de trayectorias en la imagen basado en control visual adaptativo

La calibración multi-imagen permite obtener, mientras se realiza una tarea de control visual, los parámetros intrínsecos y extrínsecos de la cámara. En el Capítulo 4 se describió un método de seguimiento intemporal de trayectorias en la imagen basado en control visual virtual. Para que este método funcione mejor se realizaba una calibración previa de la cámara que permitiera conocer sus parámetros intrínsecos. Estos parámetros intrínsecos se utilizan en el cálculo del Jacobiano de la imagen que aparece en la ley de control visual basado en imagen. Aunque la determinación de forma precisa de los parámetros intrínsecos no es necesario para obtener la convergencia en los sistemas de control visual, el comportamiento del sistema mejora si se consigue estimar su valor de forma correcta. Con el método descrito de calibración multi-imagen se puede actualizar en cada iteración del control visual el valor de los parámetros intrínsecos, y de esta forma mejorar el comportamiento del sistema.

Además, la inclusión de la calibración multi-imagen en cada iteración del control visual aporta la propiedad de realizar un seguimiento de la trayectoria deseada en condiciones en las que los parámetros internos de la cámara se modifiquen. Esto permite, por ejemplo, realizar seguimiento de trayectorias en la imagen con variaciones en la distancia focal de la cámara. En la Figura 5-4 se muestra en rojo la aportación de la calibración multi-imagen al esquema del seguimiento intemporal de trayectorias en la imagen basado en control visual virtual descrito en el apartado 4.5.

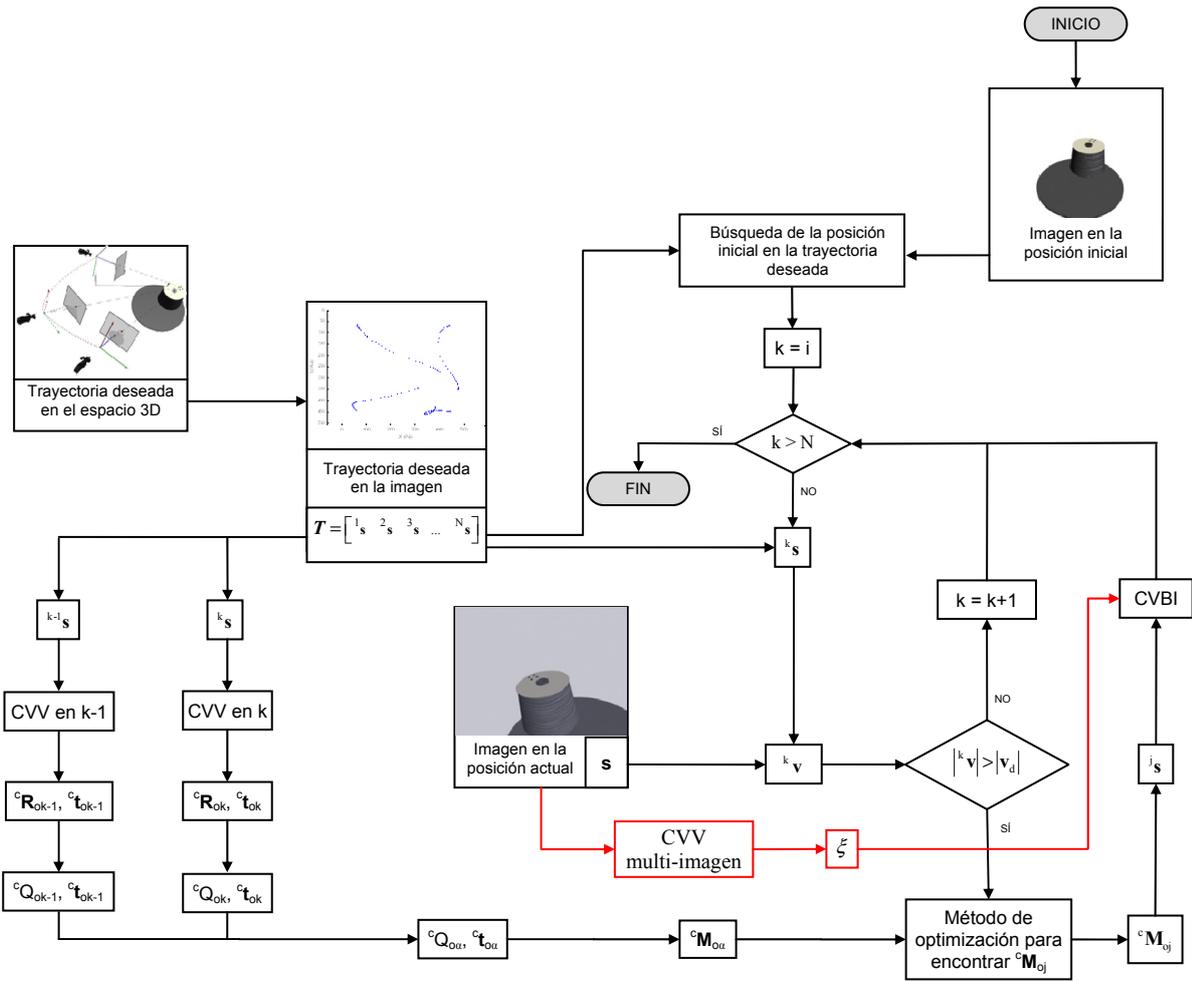


Figura 5-4. Esquema del método de seguimiento intemporal presentado en el apartado 4.5 con la mejora de la calibración multi-imagen.

## 5.5. Resultados

Para realizar la calibración usando la aproximación de control visual virtual se vuelve a emplear el objeto 3D representado en la Figura 3-1 del que se pueden extraer 5 elipses. Además, se ha empleado una cámara eye-in-hand que ha sido sometida a un proceso previo de calibración para estimar los parámetros intrínsecos. Las coordenadas en píxeles del punto principal son  $(u_0, v_0) =$

(168.75, 121.5), y la distancia focal en las direcciones  $u$  y  $v$  son (412.92, 423.69) respectivamente. Para probar la correcta calibración con el método propuesto se han realizado las siguientes pruebas:

- Prueba 1. Ventana de calibración de tamaño 6. Este experimento muestra que, usando calibración multi-imagen, el sistema es capaz de converger en una situación en la que la calibración mono-imagen no funciona correctamente. Se ha ejecutado el mismo experimento desarrollado en la Prueba 2 de la Sección 3.3.2, esta vez usando calibración multi-imagen y se ha obtenido una correcta convergencia.
- Prueba 2. Ventana de calibración de 4. Se ejecuta usando calibración multi-imagen el mismo experimento anterior. Sin embargo, en este caso, el robot está ejecutando una tarea de control visual. En esta prueba se realiza una convergencia completa del control visual virtual en cada iteración del sistema de control visual. El sistema funciona perfectamente. En este experimento es posible observar que usando una ventana de calibración más pequeña, el sistema converge correctamente.
- Prueba 3. Experimento similar al anterior pero en el que se modifica el zoom de la cámara durante el seguimiento.

Con el objetivo de resumir los resultados obtenidos en los tres experimentos, en la siguiente tabla se representan los parámetros intrínsecos obtenidos.

Prueba	Parámetros intrínsecos ( $f_u, f_v$ ), ( $u_0, v_0$ )
1	(412.74, 423.28), (168.43, 121.33)
2	(412.85, 423.45), (168.62, 121)
3	(944.9, 966.8), (155.5, 125)

Tabla 5-2. Parámetros intrínsecos obtenidos en las pruebas.

### 5.5.1. Prueba 1

En el Capítulo 3 se pudo concluir que los parámetros intrínsecos obtenidos desde diferentes puntos de vista podían diferir cuando se realiza la calibración con una sola imagen. En este apartado, se desarrolla una calibración multi-imagen para integrar en un único proceso de calibración información obtenida de imágenes distintas. Como se ha descrito previamente, para incrementar la robustez del sistema de calibración se ha propuesto integrar la información de las imágenes obtenidas durante el proceso de control visual.

Para ilustrar la correcta convergencia de la calibración multi-imagen en situaciones en las que la mono-imagen no funciona correctamente se presenta a continuación de forma detallada la convergencia con 6 imágenes. En este experimento, se usan los parámetros extrínsecos e intrínsecos iniciales descritos en la prueba 2 de la Sección 3.3.2. Se ha obtenido la evolución del error y de los parámetros intrínsecos mostrada en la Figura 5-4 y la Figura 5-5. Los parámetros intrínsecos obtenidos

son:  $(u_0, v_0) = (168.43, 121.33)$  y la distancia focal igual a  $(412.74, 423.28)$ . Los parámetros extrínsecos obtenidos son: posición  $(4, 39.48, 303.58)$  mm y orientación según los ángulos de Euler  $(\alpha, \beta, \gamma) = (2.98, 0.86, 1.57)$ . Estos valores son muy próximos a los obtenidos usando los métodos previos existentes (ver la calibración de la cámara usando el método de Zhang). Sin embargo, la velocidad de convergencia ha disminuido. La acción de control y la trayectoria de la cámara durante la tarea también se representan en la Figura 5-6 y 5-7 respectivamente.

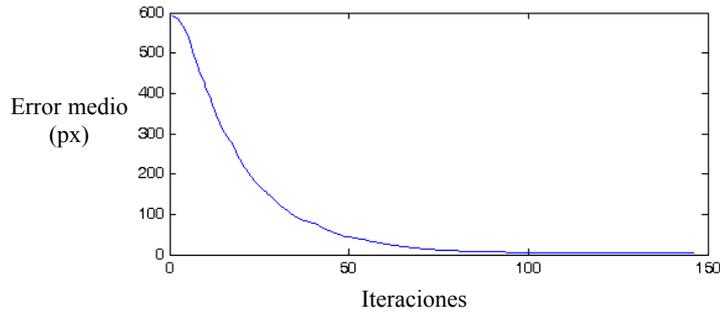


Figura 5-4. Error medio en píxeles. Prueba 1.

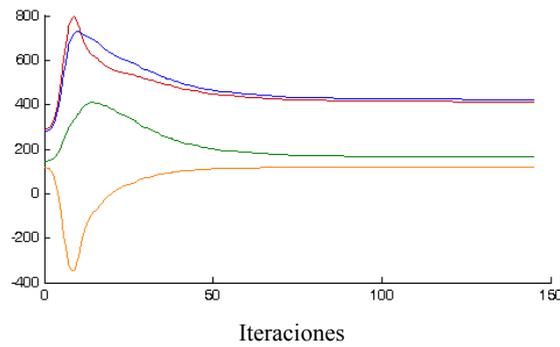


Figura 5-5. Evolución de los parámetros intrínsecos. Prueba 1.

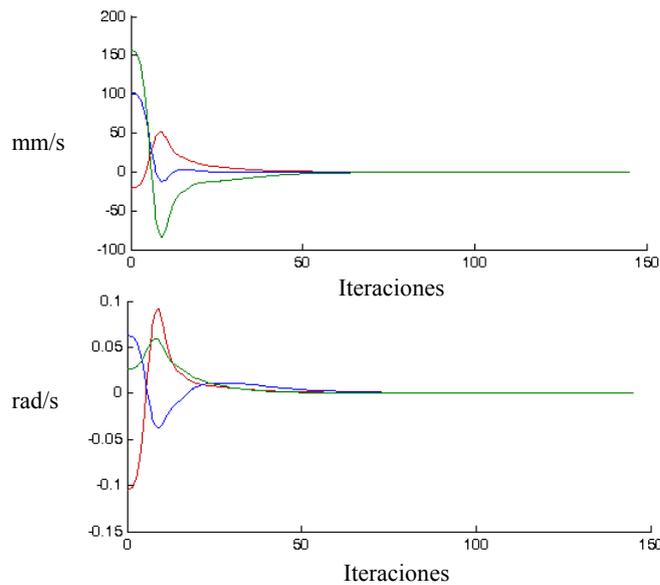


Figura 5-6. Velocidades lineal y rotacional. Prueba 1.

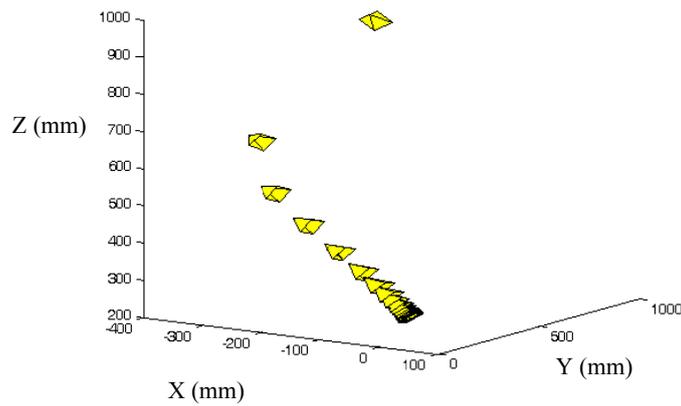


Figura 5-7. Trayectoria de la cámara. Prueba 1.

### 5.5.2. Prueba 2

La prueba anterior destaca que integrando información de diferentes puntos de vista se obtiene una calibración más precisa. Ahora, se aplica control visual virtual en cada iteración de una tarea de control visual virtual (se completa una convergencia del control visual virtual en cada iteración del control visual). De esta forma, en este caso las características deseadas del control visual virtual (es decir, las extraídas de la imagen) variarán durante la tarea. Este aspecto aporta al sistema mayor variedad de puntos de vista utilizados durante la calibración, permitiendo una calibración más precisa. Para demostrar esto último, se consideran los mismos parámetros extrínsecos e intrínsecos iniciales que en la Prueba 1 descrita en el apartado 5.5.1. Sin embargo, en este experimento, debido a la tarea de control visual, la cámara situada en configuración eye-in-hand tal y como se describe en el apartado 4-6 (ver Figura 5-8) está siguiendo la trayectoria 3D mostrada en la Figura 5-9. En esta figura se representan la trayectoria deseada y un muestreo de los parámetros extrínsecos durante la calibración.



Figura 5-8. Configuración de la cámara eye-in-hand para las pruebas.

La Figura 5-10 muestra la evolución del error cuando se consideran 4 imágenes (tamaño de la ventana de calibración) en la calibración multi-imagen durante la tarea de control visual. La Figura 5-11 muestra que durante la tarea de control visual la evolución de los parámetros intrínsecos de la cámara converge cerca de los deseados. En esta figura es posible observar que después de la primera ejecución del control visual virtual, los parámetros intrínsecos están cerca de la posición real, y luego oscilan alrededor de esta posición para cada nueva convergencia de control visual virtual. También se observa que los cambios en la posición de las características deseadas (y por lo tanto de los parámetros extrínsecos de la cámara) no producen cambios bruscos en los parámetros intrínsecos estimados. También se puede ver que los errores de calibración se han reducido en comparación con los errores de calibración obtenidos usando la misma ventana de calibración y sin desarrollar una tarea de control visual (ver Tabla 5-1). Los parámetros intrínsecos obtenidos son:  $(u_0, v_0) = (168.62, 121)$  y la distancia focal igual a  $(412.85, 423.45)$ . Los parámetros extrínsecos obtenidos con el sistema de control visual son  $(51.09, 59.85, 390.967)$  mm y la orientación según los ángulos de Euler  $(\alpha, \beta, \gamma) = (2.97, 0.73, 1.57)$ . Una vez que la tarea de control visual virtual ha terminado se obtiene un error del 4.7% en los parámetros extrínsecos. Las acciones de control durante la calibración se muestran también en la Figura 5-12. Se puede concluir que usando conjuntamente control visual y control visual virtual es posible obtener buenas calibraciones de los parámetros usando menores ventanas de calibración, por lo tanto, incrementando la velocidad de convergencia.

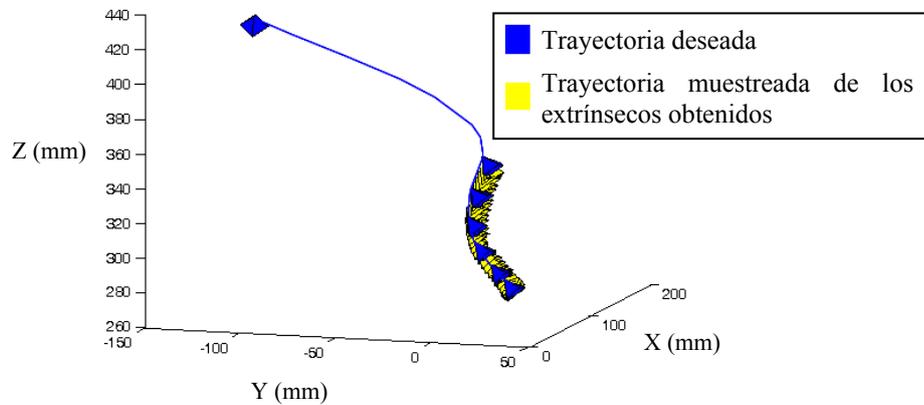


Figura 5-9. Trayectoria de la cámara. Prueba 2.

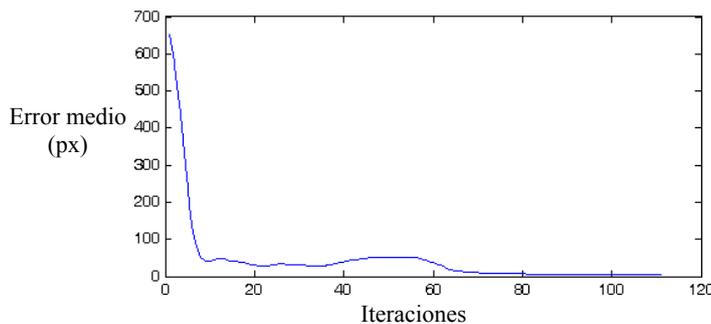


Figura 5-10. Error medio en píxeles. Prueba 2.

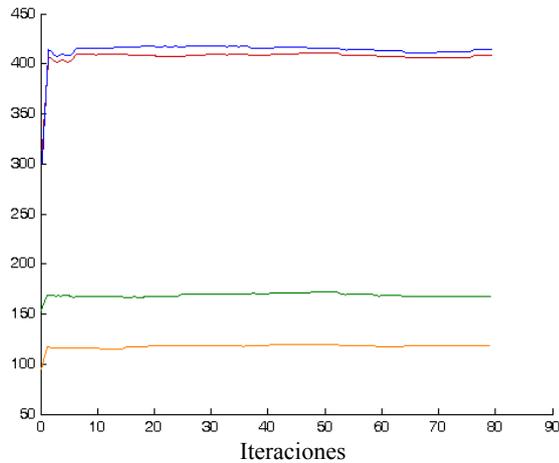


Figura 5-11. Evolución de los parámetros intrínsecos. Prueba 2.

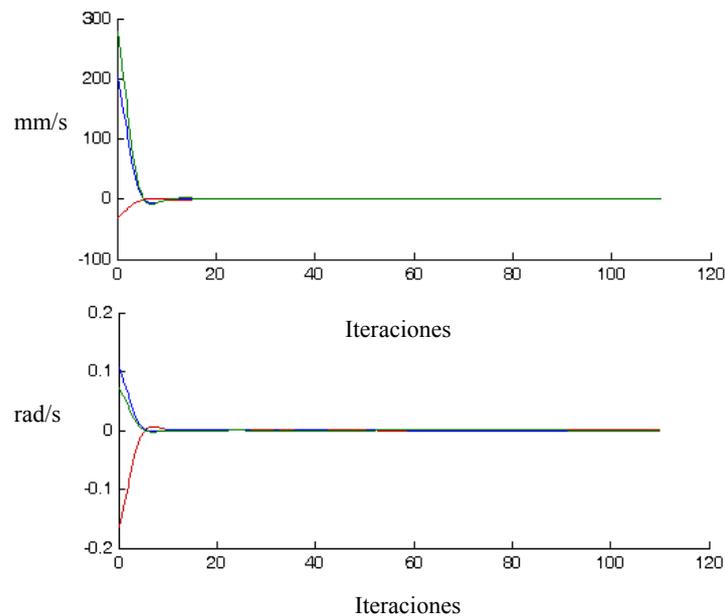


Figura 5-12. Velocidades traslacional y rotacional. Prueba 2.

### 5.5.3. Prueba 3

Finalmente se presenta un experimento en el que los parámetros intrínsecos varían durante la tarea. En este caso, el robot está ejecutando la tarea de control visual descrita en el apartado anterior. Sin embargo, durante la tarea el zoom de la cámara varía. Se ha realizado una calibración de la cámara usando el método de Zhang [ZhangZ-1999] obteniendo:  $(u_0, v_0) = (156, 124.8)$  y  $(f_u, f_v) = (526.5, 539.8)$ . Estos valores son los iniciales para el control visual virtual. En la Figura 5-13 se representa la evolución de los parámetros intrínsecos durante la tarea. Una vez que la tarea de control visual termina, los parámetros intrínsecos obtenidos son:  $(u_0, v_0) = (155.5, 125)$  and  $(f_u, f_v) = (944.9, 966.8)$

(calibrando la cámara al final de la tarea de control visual usando el método de Zhang, los parámetros intrínsecos obtenidos son:  $(u_0, v_0) = (156.2, 124.7)$  and  $(f_u, f_v) = (945.4, 967.3)$ ). El sistema es por lo tanto capaz de determinar correctamente el valor de la distancia focal cuando ésta varía durante la tarea.

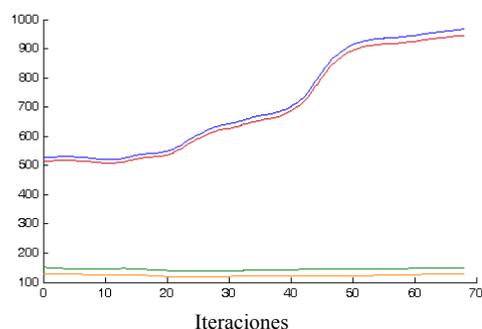


Figura 5-13. Evolución de los parámetros intrínsecos. Prueba 3.

## 5.6. Conclusiones

Para obtener un buen comportamiento en los sistemas de control visual virtual, un aspecto importante es el uso de la calibración multi-imagen. Usando únicamente calibración mono-imagen, los parámetros obtenidos en la calibración pueden diferir dependiendo de los parámetros extrínsecos iniciales. Además, se ha observado que usando sólo una imagen, el control visual virtual puede obtener parámetros en la calibración que pueden estar muy alejados de los reales. La calibración multi-imagen incrementa la robustez del sistema. Sin embargo, en esta última aproximación, un factor importante es la determinación del tamaño de la ventana de calibración. Cuanto más grande sea la ventana de calibración, más precisa será la calibración, necesitando por contra más iteraciones para la convergencia del sistema.

Este tipo de sistemas de control visual autocalibrado se puede emplear en situaciones en las que los parámetros intrínsecos varían durante la tarea. Usando esta aproximación, el sistema es capaz de adaptarse a posibles cambios en los parámetros de calibración de la cámara durante la tarea de control visual. Además, en cada momento, el sistema es capaz de determinar la posición de la cámara con respecto al objeto del que se extraen las características visuales.

Aplicando la técnica de calibración descrita en este Capítulo al seguimiento intemporal de trayectorias en la imagen basado en control visual virtual descrito en el Capítulo 4, se consigue un seguimiento mejorado al poder actualizar de forma precisa el Jacobiano de la imagen empleado en el

control visual basado en imagen. Además, permite realizar el seguimiento incluso en condiciones en las que los parámetros intrínsecos varíen durante la tarea de seguimiento, como podría ser el caso de que la cámara de seguimiento realizara un zoom sobre la escena.

En el siguiente Capítulo se describen las técnicas de fusión entre un sistema de control de fuerza y este sistema de seguimiento de trayectorias basado en control visual virtual y mejorado con la calibración multi-imagen que permiten al robot interactuar con el entorno.



# Capítulo 6

## Fusión de control visual y control de fuerza



## 6.1. Introducción

---

Los sistemas de control de fuerza se han utilizado hasta la actualidad como los mejores sistemas para controlar la interacción de un robot manipulador con el entorno. Entre los distintos esquemas de control de interacción aplicados a manipuladores propuestos hasta ahora cabe citar aquellos que hacen uso de control de fuerza, control de impedancia, control híbrido o control paralelo. Estas aproximaciones se revisan en [De Schutter-1997].

Por otro lado, en tareas en las que el robot debe posicionarse con respecto a un objeto específico del entorno, puede hacerse uso de la información proporcionada por un sistema de visión y, en la actualidad, existen una gran cantidad de controladores visuales desarrollados con este motivo. Así, en aquellas tareas en las que se debe realizar la manipulación de un determinado objeto del espacio de trabajo se hace necesario el uso de forma combinada de los controladores visuales junto con controladores de fuerza. En este caso los sistemas de visión tienen la función de establecer la trayectoria del robot durante la tarea, actuando como controladores de posición, mientras el sistema sensor de fuerza se encarga de controlar la interacción del robot con el entorno.

Atendiendo a la fusión de estos dos sistemas sensoriales el principal problema que se encuentra es que ambos sistemas proporcionan información de naturaleza distinta ya que miden fenómenos físicos distintos, aunque complementarios. Este factor condiciona los sistemas desarrollados para fusionar la información obtenida por ambos. Así, los sistemas de visión proporcionan información global del espacio de trabajo, no siendo adecuados para controlar la interacción entre el robot y los objetos manipulados. Por otro lado, los sensores de fuerza proporcionan información acerca de la geometría del contacto, pudiendo únicamente controlar el movimiento descrito por el robot a partir de información local, es decir, de las fuerzas de contacto.

Las principales aproximaciones empleadas hasta la actualidad para combinar información visual y de fuerza son control híbrido y control de impedancia (en los Apartados 6.1.1 y 6.1.2 se muestran más en detalle cada una de estas aproximaciones). En el caso de control híbrido, el espacio de control se divide en direcciones controladas por fuerza y direcciones controladas por visión siendo requisito para aplicar esta estrategia el disponer de un conocimiento previo detallado de la geometría de la tarea y el entorno en el que se va a desarrollar. En el caso de emplear control de impedancia se especifica la relación dinámica entre fuerza y movimiento. En [Deng-2005] se muestra una revisión de esta última aproximación.

Atendiendo a los distintos niveles de fusión de visión y fuerza cabe mencionar el control alternado, en el que una dirección dada es controlada alternativamente por visión y fuerza, control híbrido, es decir, control de direcciones distintas con visión y fuerza simultáneamente, y control

compartido, con el que se puede controlar una misma dirección empleando información visual y de fuerza simultáneamente.

Como se observará durante el desarrollo de este capítulo, la combinación de sistemas de control visual y fuerza constituye un campo de investigación reciente en el que la mayoría de las contribuciones se centran en control visual-fuerza híbrido. Aunque en menor proporción, también es posible encontrar estudios relativos a sistemas de control de impedancia para la combinación de estas informaciones. Es este motivo por el que en los dos siguientes apartados de este capítulo se describirán los trabajos existentes hasta la actualidad en control visual-fuerza híbrido y sistemas que combinan control visual con control de impedancia. En el Apartado 6.1.3 se muestran otros trabajos que si bien no se pueden encuadrar en ninguna de las dos aproximaciones anteriores, sí que utilizan distintas estrategias para la combinación de ambos sistemas sensoriales.

### 6.1.1. Control visual-fuerza híbrido

El esquema general de control híbrido fue propuesto originalmente por Raibert y Craig [Raibert-1981] consistente básicamente en seleccionar las direcciones del espacio de trabajo en las que debe controlarse el movimiento y las direcciones en las que debe controlarse la fuerza de interacción ejercida por el robot. De esta manera, el sistema presenta dos bucles de control separados para movimiento y fuerza, de forma que en las direcciones de control de posición se determina la posición actual y en las direcciones de control de fuerza se mide la fuerza de interacción sobre el medio. A partir de este esquema genérico de control híbrido, se obtiene un sistema control visual-fuerza híbrido cuando para las direcciones en las que debe controlarse el movimiento se emplea un sistema de control visual.

Para exponer el concepto de control híbrido se muestra a continuación la formulación de un sistema de control visual-fuerza híbrido. Esta aplicación, cuyo esquema general se muestra en la Figura 6-1, consiste en la inserción de una varilla, que se encuentra en el extremo del robot, en un orificio presente en un plano por el cual se desplaza. El subsistema de fuerza será el encargado de mantener una fuerza de interacción constante entre la varilla y el plano (eje  $z$ ), mientras que el subsistema de visión determina la trayectoria a describir para alcanzar el orificio (trayectoria en las direcciones  $x, y$ , es decir, tangente al plano).

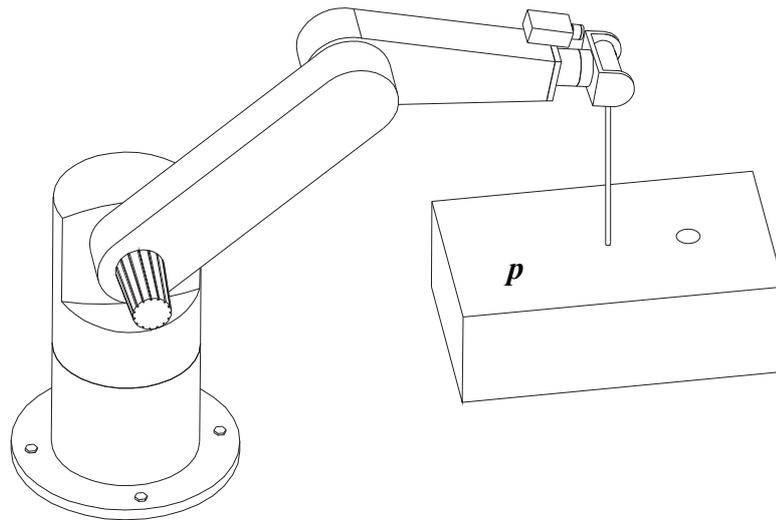


Figura 6-1. Ejemplo de tarea de control visual-fuerza híbrido.

Como se ha descrito en el Capítulo 2, la acción de control procedente de un sistema de control visual puede representarse de la siguiente manera:

$$\mathbf{v}^c = -k_v \cdot \mathbf{J}_f^+ \cdot (\mathbf{s} - \mathbf{s}_d) \quad (6.1)$$

donde  $k_v$  es la ganancia del regulador proporcional. Considérese ahora una tarea de control visual-fuerza híbrida en la que el sistema de control de fuerza debe mantener una fuerza constante con una superficie mientras que el sistema de visión se encarga de regular el movimiento del robot en una dirección tangente a la superficie (perpendicular al vector director de dicha superficie). La superficie restringe las trayectorias que puede generar el robot de forma que han de estar contenidas en el plano,  $\mathbf{p}$ . De esta manera, el eje  $z$  del sistema de coordenadas del robot habrá de permanecer perpendicular con la superficie en todo momento, de forma que:

$$\mathbf{p} \cdot \mathbf{r} = 0 \quad (6.2)$$

siendo  $\mathbf{p} = (p_1, p_2, -1, p_4)$  y  $\mathbf{r} = (x, y, z, 1)$  la posición del extremo del robot que, obviamente, deberá estar contenida en el plano. Derivando la expresión anterior respecto al tiempo se obtiene:

$$\mathbf{p}_v \cdot \dot{\mathbf{r}} = 0 \quad (6.3)$$

donde  $\mathbf{p}_v = (p_1, p_2, -1)$  y  $\dot{\mathbf{r}}$  es la velocidad descrita por el robot, tangente a la superficie, y que en este caso será establecida por un sistema de control visual.

A partir de la definición de matriz de interacción se obtiene:

$$\dot{\mathbf{s}} = \mathbf{J}_f \cdot \dot{\mathbf{r}} \quad (6.4)$$

La evolución de las características en la imagen para esta tarea de control híbrido puede obtenerse sustituyendo en (6.4) la relación obtenida en (6.3) de forma que se obtiene finalmente:

$$\dot{\mathbf{s}} = \mathbf{J}_f \cdot \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ p_1 & p_2 \end{bmatrix} \cdot \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} \quad (6.5)$$

Por lo tanto, la acción de control del sistema de control visual será:

$$\dot{\mathbf{r}}_v = -\mathbf{k}_v \cdot \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ p_1 & p_2 \end{bmatrix} \cdot \mathbf{J}_f^+ \cdot (\mathbf{s} - \mathbf{s}_d) \quad (6.6)$$

Por otro lado, el subsistema de fuerza se encarga de mantener una fuerza de interacción constante con la superficie, de forma que haciendo uso de una acción de control proporcional se obtiene:

$$\dot{\mathbf{r}}_F = \begin{bmatrix} \mathbf{0}_{2 \times 1} \\ -\mathbf{k}_F \cdot (\mathbf{F} - \mathbf{F}_d) \end{bmatrix} \quad (6.7)$$

donde  $\mathbf{F}$  y  $\mathbf{F}_d$  son la fuerzas de interacción obtenidas y deseadas respectivamente entre la superficie y el robot. Combinando esta acción de control con la obtenida en la Ecuación (6.6), se obtiene finalmente la acción de control global para el sistema de control híbrido:

$$\dot{\mathbf{r}}_H = \dot{\mathbf{r}}_F + \dot{\mathbf{r}}_v = \begin{bmatrix} \mathbf{0}_{2 \times 1} \\ -\mathbf{k}_F \cdot (\mathbf{F} - \mathbf{F}_d) \end{bmatrix} - \mathbf{k}_v \cdot \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ p_1 & p_2 \end{bmatrix} \cdot \mathbf{J}_f^+ \cdot (\mathbf{s} - \mathbf{s}_d) \quad (6.8)$$

En cuanto a sistemas híbridos control visual-fuerza destacan trabajos como [Baeten-2003, Baeten-2002b] que extienden el formalismo “task frame” [Bruyninckx-1996, De Schutter-1988] para incorporar control visual junto a control de fuerza. Haciendo uso de este formalismo, se especifica a alto nivel qué sistema sensorial es utilizado para cada dirección posible de movimiento en cada momento de la tarea. Considérese, por ejemplo, el pulido de una superficie en el que se debe seguir una arista manteniendo una fuerza constante con la misma. Haciendo uso de este formalismo, la tarea se especifica del siguiente modo:

```
TAREA pulido - seguimiento de arista: {
  x: velocidad 10 mm/seg.
  y: distancia a la característica deseada 0 mm.
  z: fuerza -15 N.
  θx: fuerza 0 Nmm.
  θy: fuerza 0 Nmm.
  θz: ángulo de la característica en la imagen 0 rad.
}
```

En esta especificación de la tarea las direcciones z,  $\theta_x$  y  $\theta_y$  son controladas empleando control de fuerza, mientras que para el resto de direcciones se hace uso de control visual.

Gran parte de los sistemas de control visual-fuerza híbridos propuestos hasta la actualidad utilizan un sistema de cámara externo al robot. Dentro de estos trabajos cabe mencionar [Hosoda-1998] que es aplicado para el seguimiento de superficies desconocidas. Para ello incluye un sistema capaz de estimar parámetros de la cámara utilizada y de reconocer la superficie con la que interactúa el robot midiendo las fuerzas de interacción y de rozamiento que sufre durante su movimiento. En [Namiki-1999] se describe un sistema para el agarre de objetos en tiempo real que hace uso de información procedente de una cámara externa al robot y la obtenida de los sensores de fuerza introducidos en una mano robótica. Este trabajo se centra principalmente en el desarrollo de una arquitectura paralela que permita retardos del orden de 1 milisegundo en la captura y procesamiento de la información sensorial. En [Nelson-2001] se describe una aplicación de control visual-fuerza híbrido aplicado a la teleoperación en entornos virtuales. En [Pichler-2000] se muestra un sistema híbrido con cámara externa al robot que de nuevo hace uso únicamente de la información procedente del sensor de fuerza para determinar la orientación de la superficie de contacto según el robot se desplaza por la misma. Una estrategia semejante se describe en [Xiao-1998, Xiao-2000] para el seguimiento de superficies en entornos no estructurados.

De nuevo, en [Olsson-2002] se muestra un sistema de control visual-fuerza híbrido que hace uso de un par estereoscópico externo al robot para mejorar la reconstrucción del entorno. En este caso se emplea la información relativa a la cinemática del robot para reconstruir la ecuación del plano con la que interactúa el robot. Sin embargo, para ello se realiza la suposición de que ciertos parámetros de dicha ecuación son parcialmente conocidos, determinando únicamente algunos de ellos a través de la cinemática. Hassanzadeh [Hassanzadeh-2005] propone una arquitectura abierta para realizar un control híbrido de visión y fuerza en entornos no estructurados y de diferentes modos. Permite también controlar el sistema desde Internet mediante un sistema háptico.

En [Carelli-2004] se extiende el esquema general de control híbrido para considerar un banco de controladores que incluye, además del control de posición y fuerza tradicionales, controladores servovisuales y controladores de impedancia con realimentación de fuerzas físicas o ficticias.

Este tipo de sistemas híbridos requieren conocer de antemano con precisión la geometría del entorno en el que se desarrolla la tarea con el objetivo de realizar la división entre direcciones controladas por visión y direcciones controladas por fuerza. Por lo tanto, esta aproximación en general no puede aplicarse en entornos en los que el robot puede interactuar con objetos desconocidos o incluso colisionar con obstáculos. Por otro lado, la división del espacio de trabajo realizada por estos sistemas implica que ambos sistemas sensoriales no pueden emplearse simultáneamente para controlar una dirección dada. Estos dos motivos han propiciado la creación de un nuevo sistema de fusión control visual-fuerza (descrito en este Capítulo) que evita las limitaciones de los sistemas híbridos permitiendo controlar la interacción con objetos desconocidos o incluso entornos no estructurados.

Además, la estrategia propuesta permite hacer uso de control visual y fuerza simultáneamente para controlar una misma dirección dada de movimiento aprovechando la información proporcionada por cada sistema sensorial y fusionando las mismas para obtener un mejor comportamiento que el que se obtendría con la utilización de uno único.

### 6.1.2. Control de impedancia

El control de impedancia [Hogan-1985] frente al control de fuerza clásico desarrolla el control dinámico entre el movimiento y la fuerza de interacción del robot. Para conseguirlo se especifica una impedancia deseada que se obtiene mediante el filtrado de las fuerzas de interacción de la siguiente manera:

$$\mathbf{x}_i(t) = \frac{\mathbf{F}(t)}{\mathbf{I}s^2 + \mathbf{D}s + \mathbf{K}} \quad (6.9)$$

donde  $\mathbf{I} \in \mathcal{R}^{n \times n}$  es la matriz de inercia,  $\mathbf{D} \in \mathcal{R}^{n \times n}$  es la matriz de amortiguamiento y  $\mathbf{K} \in \mathcal{R}^{n \times n}$  es la matriz de elasticidad. Estas matrices son diagonales y caracterizan la función de impedancia deseada.

A partir de la expresión (6.9) se obtiene un vector de ajuste,  $\mathbf{x}_i$ , que modifica la trayectoria deseada,  $\mathbf{x}_d$ . En la Figura 6-2 se muestra el esquema general de un sistema de control de impedancia.

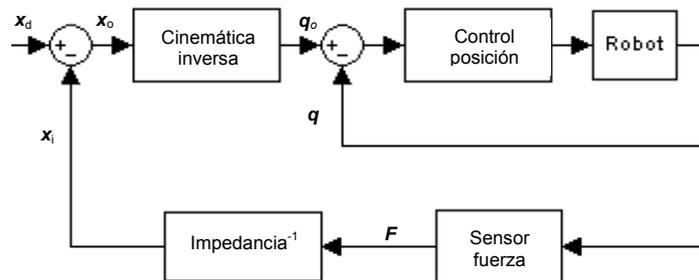


Figura 6-2. Esquema de control de impedancia.

Se puede afirmar que existen dos tipos básicos de control de impedancia: control pasivo y activo. En el control pasivo se emplean elementos mecánicos para alcanzar la impedancia deseada, mientras que en el control activo la realimentación de fuerza es utilizada para controlar el movimiento del robot. Los sistemas de control de impedancia suelen clasificarse en dos grupos principalmente:

- *Control de impedancia basado en fuerza.* En este caso el bucle de más bajo nivel es una realimentación de fuerza.
- *Control de impedancia basado en posición.* En este caso se hace uso en bajo nivel de un controlador de posición.

En general, frente a los basados en posición, los basados en fuerza son capaces de proporcionar un mejor comportamiento presentando respuestas más suaves [Lawrence-1988]. Sin embargo, se requiere un modelo dinámico del robot para garantizar su estabilidad. En las distintas estrategias control visual-fuerza propuestas principalmente se supondrá la utilización de control activo basado en posición.

A partir del esquema de control de impedancia básico mostrado en la Figura 6-2, existen modificaciones como la descrita en [Morel-1998, Pomares-2003], en la que se agrega un bucle de control externo consistente en un controlador visual que genera referencias ( $x_d$  en la Figura 6-2) para un sistema de control de impedancia. Este sistema hace uso de una cámara ubicada en el extremo del robot y se aplica a tareas de acoplamiento entre piezas. Frente a otras aproximaciones como la descrita en [Nelson-1996b] en la que los sistemas de visión y fuerza actúan en el mismo nivel de la jerarquía de control, la descrita en [Morel-1998] se caracteriza por presentar un bucle de visión externo dispuesto alrededor del bucle de control de posición. Esta misma técnica se aplica en [Pomares-2003].

En [Carelli-2004, Mut-1998, Nakabo-1998, Tsuji-1997] se propone la utilización de fuerzas ficticias aplicadas a tareas de aproximación sin contacto. En [Tsuji-1997] se hace uso de un sistema de control de impedancia sin contacto aplicado a un robot redundante. Esta aproximación emplea la información visual de una cámara externa fija para generar una fuerza virtual entre cada articulación del robot y el medio. En [Nakabo-1998] se describe el concepto de impedancia visual y se define un contacto virtual con una superficie virtual en la imagen. En este caso las fuerzas ficticias modifican las señales de control generadas por un sistema de control visual, obteniendo las correspondientes acciones de control. Por último, dentro del ámbito de la utilización de sistemas de control de impedancia visual junto con fuerzas ficticias destacan los trabajos [Carelli-2001, Mut-1998] en los que se considera una estructura de control de impedancia donde la fusión de la información visual y de fuerzas (físicas o ficticias) se realiza en el plano imagen. Por lo tanto, el objetivo de control de impedancia visual resulta definido como la relación dinámica entre las fuerzas y el error de las características en la imagen. En estos trabajos se define también distintas estrategias para la generación de las fuerzas ficticias, que en cualquier caso suele tratarse de una función de la distancia entre el extremo del robot y el objeto implicado en la tarea.

Pomares [Pomares-2005b, Pomares-2006b] propone un sistema de control de impedancia que utiliza un sistema de control visual que ofrece la posibilidad de hacer un seguimiento de una trayectoria, evitando las restricciones temporales y sin conocimiento previo de los parámetros intrínsecos de la cámara.

### 6.1.3. Otras aproximaciones

Como se ha descrito anteriormente, la combinación de diferente información sensorial ha sido estudiada a través de dos aproximaciones básicas. La primera de ellas considera que los sensores miden fenómenos de naturaleza física distinta. En este caso, a lo largo de la tarea se selecciona la información sensorial más adecuada a emplear en cada momento. La segunda aproximación considera que la información a fusionar es toda de la misma naturaleza. La primera estrategia ha sido utilizada hasta la actualidad para fusionar información visual y de fuerza, considerando las distintas características de la información obtenida de los sensores. Como ejemplo de esta aproximación cabe citar el concepto de “resolvability” [Nelson-1996a], que proporciona un indicador acerca de la “calidad” de la información obtenida a partir de cada sensor o su utilidad para guiar el robot en un instante dado de la tarea. Este indicador es seleccionado para determinar el sensor a utilizar en cada momento.

También es posible encontrar otras aproximaciones que emplean una combinación de la información de visión y fuerza, como los trabajos de Zhou, Nelson y Vikramaditya [Zhou-1998, Nelson-1997] aplicados a micro-manipulación, en los que se define un elemento supervisor que determina el sistema sensorial más adecuado a aplicar en cada momento.

Para fusionar la información visual y de fuerza también destacan la presencia de controladores borrosos. Así, Von Collani y otros [Von Collani-2000, Von Collani-1999] emplean una solución neuro-fuzzy para la aplicación del control visual y de fuerza combinado a una tarea de posicionamiento que requiere alta precisión.

Aplicando el formalismo “task frame” también es posible encontrar trabajos encaminados al control de una dirección dada de movimiento simultáneamente haciendo uso de visión y fuerza [Baeten-2002a, Baeten-2003].

En [Malis-2001] se describe la utilización de distintos sistemas sensoriales para el control de una tarea. Dentro de estos sistemas sensoriales se profundiza en el empleo de sistemas de visión y fuerza para una tarea de inserción. Esta estrategia supone una extensión a la aproximación de función de la tarea [Samson-1991].

Cabe mencionar asimismo otros trabajos como [Jörg-2000] aplicados a un sistema de ensamblaje flexible que emplea un sistema multisensorial. Para desarrollar esta tarea se hace uso de control alternado de forma que una dirección dada de movimiento es controlada alternativamente por cada sistema sensorial dependiendo del estado de la misma.

Un sistema de control visual basado en imagen no permite conocer la trayectoria 3-D seguida por el robot hasta alcanzar la configuración deseada. Así, empleando este tipo de sistemas, únicamente se dispone del conocimiento del valor de las características deseadas obtenidas en la imagen una vez que la tarea ha finalizado, pero no qué trayectoria se describirá hasta alcanzar estas características. Este aspecto se convierte en crítico en tareas en las que el robot debe interactuar con el entorno. Pomares y Torres [Pomares-2005a, Pomares-2006a] describen una aproximación, denominada control visual basado en flujo de movimiento, que permite garantizar el correcto seguimiento de una trayectoria especificada en el espacio de la imagen, presentando un correcto comportamiento tanto en la imagen como en el espacio 3-D.

Por último, basadas en una nueva formulación del principio de ortogonalización usado en control de fuerza en [Dean-León-2005a, Dean-León-2005b] se trata una aproximación llamada principio de ortogonalización visual que consigue una convergencia simultánea de la posición visual restringida y de la fuerza de contacto entre el extremo del robot y la superficie que restringe el movimiento.

## **6.2. Fusión de control visual-fuerza basado en control de impedancia**

---

En este apartado se va a describir un sistema de control visual-fuerza que permite al robot interactuar con el entorno. La Figura 6-3 muestra un esquema del sistema propuesto.

Este sistema consta de tres bloques principales:

- Un controlador de impedancia que es realimentado con información visual.
- Un sistema que permite modificar la trayectoria deseada en la imagen a partir de la información proporcionada por el sensor de fuerza.
- Un mecanismo que permite detectar cambios en la superficie de contacto a partir de la información de las fuerzas de interacción y de la información proporcionada por un laser.

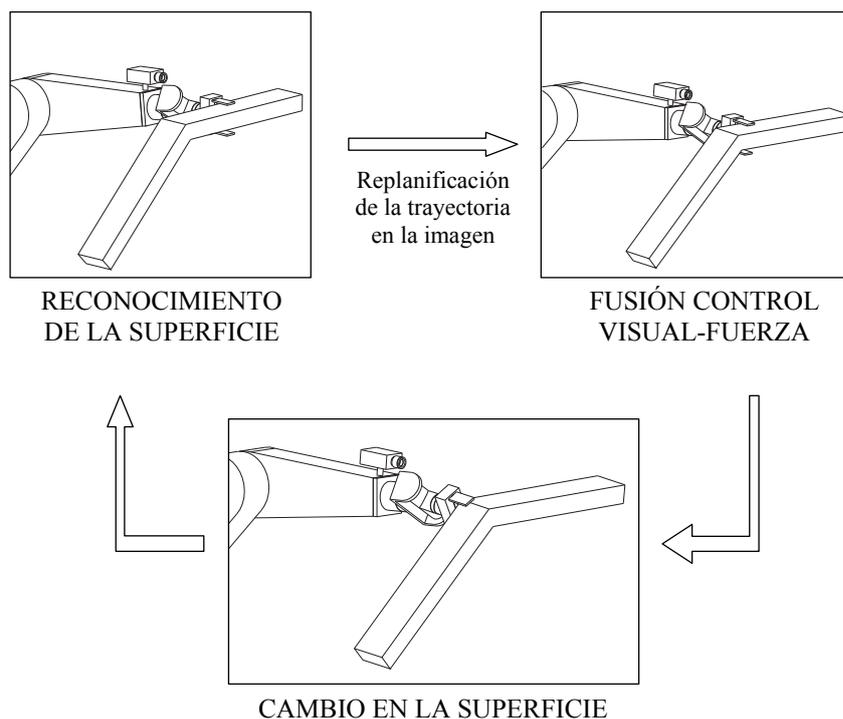


Figura 6-3. Esquema del sistema propuesto de control de la interacción con el entorno.

### 6.2.1. Esquema de control de impedancia con realimentación visual

Hasta ahora, la mayoría de las aplicaciones desarrolladas para combinar información visual y de fuerza emplean el control híbrido descrito en el apartado anterior. En estos casos, el espacio de trabajo debe ser conocido de forma muy precisa, de forma que se pueda dividir primero las direcciones controladas mediante fuerza de aquellas que lo serán por posición (es decir, por un sistema de control visual). Con la intención de incrementar la versatilidad del sistema y de mejorar su respuesta ante las incertidumbres que aparecen durante la tarea de manipulación, se ha desarrollado un sistema especial de control de impedancia visual [Pomares-2007b]. Este sistema utiliza la información obtenida a partir del seguimiento intemporal de trayectorias en la imagen descrito en el Capítulo 4. Este sistema tiene una realimentación visual externa, en contraste con otras aproximaciones [Nelson-1996b] que tienen los sistemas de visión y fuerza en el mismo nivel de la jerarquía de control.

El objetivo del control de impedancia es el de llevar el control combinado del movimiento del robot y su fuerza de interacción. El control activo basado en posición es la técnica elegida para desarrollar el controlador de visión y fuerza que se va a describir. En el control activo, la realimentación de fuerza se usa para controlar los movimientos de las articulaciones. La siguiente ecuación de impedancia establece un comportamiento en base a una determinada inercia, amortiguamiento y elasticidad para el desplazamiento en posición cuando el extremo del robot ejerce una fuerza  $F$  en el entorno:

$$F = \mathbf{I}\Delta\ddot{\mathbf{x}}_{vc} + \mathbf{D}\Delta\dot{\mathbf{x}}_{vc} + \mathbf{K}\mathbf{x}_{vc} \quad (6.10)$$

donde  $\mathbf{x}_c$  es la posición actual del extremo del robot,  $\mathbf{x}_v$  es la trayectoria de referencia,  $\Delta\mathbf{x}_{vc} = \mathbf{x}_v - \mathbf{x}_c$ ,  $\mathbf{I} \in \mathcal{R}^{n \times n}$  es la matriz de inercia,  $\mathbf{D} \in \mathcal{R}^{n \times n}$  es la matriz de amortiguamiento y  $\mathbf{K} \in \mathcal{R}^{n \times n}$  es la elasticidad. Todas son matrices diagonales y caracterizan la función de impedancia deseada.

Para implementar este controlador, se ha introducido una realimentación visual externa en el esquema de control de impedancia con una realimentación interna de movimiento. Para la implementación se ha elegido un sistema de control de impedancia basado en posición llamado control de acomodación [Whitney-1977] en el que la impedancia deseada está limitada a un amortiguamiento puro  $\mathbf{D}$  (en [Malis-2001] se muestran los principales aspectos de la estabilidad de este tipo de control). En este caso:

$$F = \mathbf{D}\Delta\dot{\mathbf{x}}_{vc} \quad (6.11)$$

Por lo tanto, la ley de control obtenida será:

$$\dot{\mathbf{x}}_c = \dot{\mathbf{x}}_v - \mathbf{D}^{-1}F \quad (6.12)$$

donde el término  $\dot{\mathbf{x}}_v$  se obtiene usando el método de seguimiento intemporal de trayectorias en la imagen basado en control visual virtual descrito en el Capítulo 4.

$$\dot{\mathbf{x}}_v = -\lambda \cdot \mathbf{C}_v \cdot \hat{\mathbf{J}}_s^+ \cdot (\mathbf{s} - \mathbf{j}\mathbf{s}) \quad (6.13)$$

donde la velocidad se transforma desde el sistema de coordenadas de la cámara al del extremo del robot usando la matriz  $\mathbf{C}_v$ .

En la Figura 6-4 se muestra el esquema del control de impedancia propuesto. En esta figura el componente de cinemática inversa implementa un algoritmo convencional pseudo inverso,  $\dot{\mathbf{q}}_c = \mathbf{J}_q^+ \dot{\mathbf{x}}_c$ , donde  $\dot{\mathbf{q}}_c$  son las correspondientes velocidades de las articulaciones y  $\mathbf{J}_q^+$  es la pseudo inversa de la matriz Jacobiana.

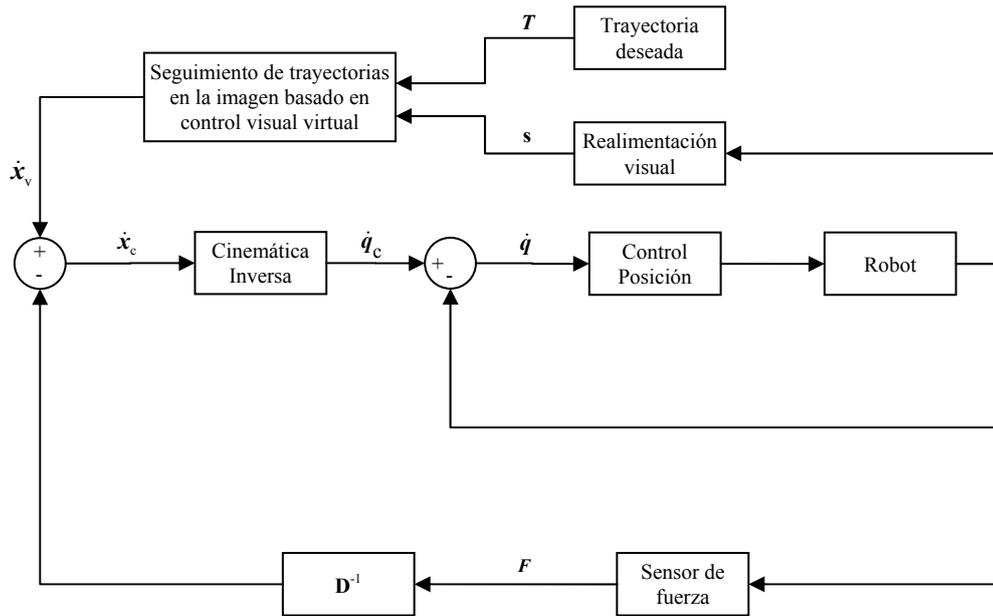


Figura 6-4. Esquema del control de impedancia con realimentación visual.

Por lo tanto, definiendo el amortiguamiento deseado  $\mathbf{D}$ , la información procedente de ambos sensores se combina como muestra la Ecuación (6.12). De esta forma, la trayectoria generada por el sistema de seguimiento intemporal de trayectorias basado en control visual virtual se corrige de acuerdo con la interacción del robot.

## 6.2.2. Modificación de la trayectoria en la imagen a partir de las fuerzas de interacción

Hasta ahora, las aproximaciones para fusionar la información visual y la de fuerza no consideran la posibilidad de que ambos sensores aporten información contradictoria en un momento dado de la tarea (es decir, el sistema de control visual establece una dirección de movimiento que es imposible de acuerdo con la información de interacción obtenida a partir del sensor de fuerza).

Para asegurar que una determinada tarea en la que se requiere una interacción con el entorno se desarrolla correctamente, el sistema debe llevar a cabo una variación de la trayectoria en la imagen, dependiendo de las restricciones espaciales impuestas por las fuerzas de interacción. Para conseguirlo, dada una colisión con el entorno de trabajo y habiendo reconocido el vector normal  $n$  a la superficie de contacto [Pomares-2005a], se determina la transformación que debe llevar a cabo la cámara para cumplir las restricciones espaciales. Esta transformación, dada por una matriz de rotación  $\mathbf{R}$  y una traslación  $t$ , se calcula de forma que represente la dirección más cercana a la obtenida a partir del seguimiento intemporal de trayectorias en la imagen basado en control visual virtual, y que está contenida en el plano de la superficie.

Por lo tanto, para garantizar que la información visual sea coherente con la obtenida del sensor de fuerza, la trayectoria en la imagen se debe modificar, de forma que la nueva trayectoria es la obtenida a partir de la transformación previamente mencionada. Para conseguirlo, se vuelve a utilizar el concepto de matriz de homografía descrito en el apartado 4.4. Esta matriz,  $\mathbf{G}$ , relaciona los puntos vistos desde una configuración de la cámara con los mismos puntos observados desde otra posición de la cámara. Si se calcula la matriz  $\mathbf{G}$  en cada iteración (es decir, cada vez que se captura una imagen) de acuerdo con la transformación obtenida tras detectar una colisión (a partir de la matriz de rotación  $\mathbf{R}$  y de la traslación  $t$ ), se podrá obtener la nueva trayectoria en el espacio imagen.

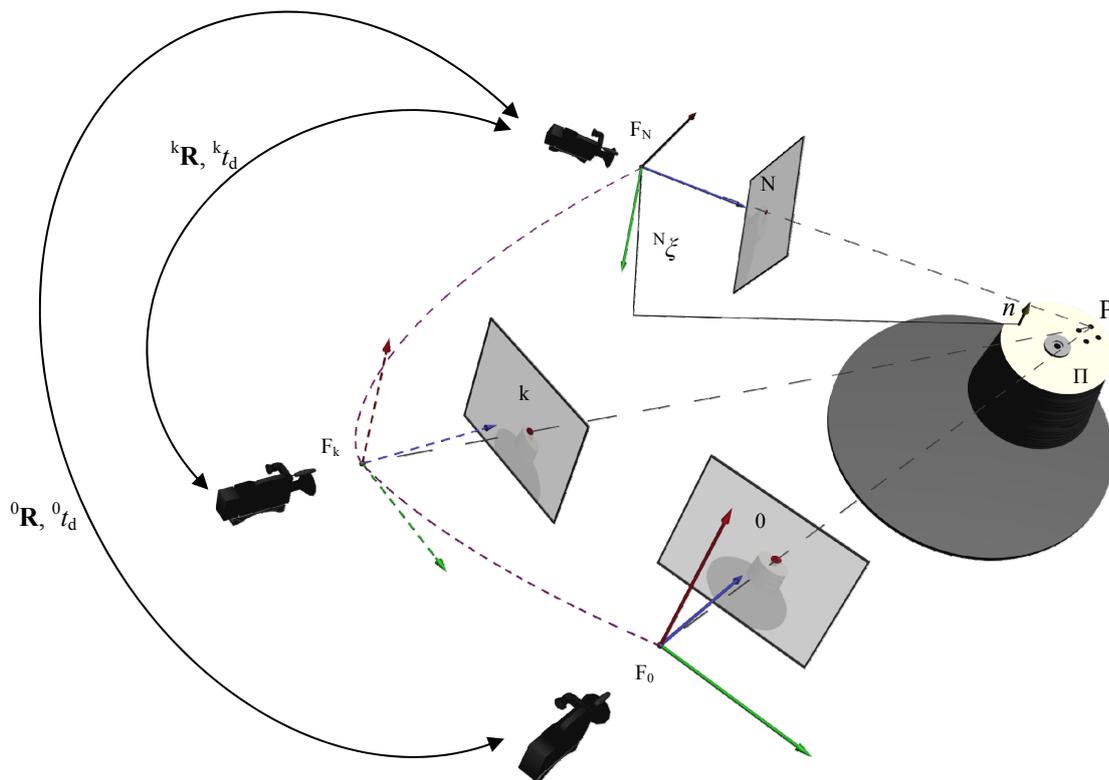


Figura 6-5. Esquema de la reconstrucción del movimiento.

Primero, considérese  $\Pi$  el plano de la superficie de contacto. Una vez que la colisión se detecta, considerando  $P_i$  un punto 3D observado por la cámara (que representa una característica extraída del objeto), el mismo punto en el espacio imagen es  $f_i$ .  ${}^k f_i$  representa la posición de la característica en la imagen capturada en una posición intermedia  $k$  de la cámara. En la imagen de destino (que se corresponde con la imagen obtenida por la cámara colocada en la posición final de la trayectoria), el mismo punto se localizará en la posición  ${}^N f_i$  (esta notación se ilustra en la Figura 6-5).

La matriz  ${}^k\mathbf{R}$  denota la rotación entre una posición intermedia en la trayectoria y la final, mientras que  ${}^k t_d$  es la traslación escalada de la cámara entre las dos posiciones:

$${}^k t_d = \frac{{}^k t}{N \xi} \quad (6.14)$$

donde  ${}^N \xi$  es la distancia desde  $\Pi$  al origen del sistema de referencia de la cámara situado en la posición final de la trayectoria.

La homografía  ${}^k \mathbf{G}$  se define como:

$${}^k \mu_i {}^k \mathbf{f}_i = {}^k \mathbf{G} {}^N \mathbf{f}_i + \beta_i {}^k e \quad (6.15)$$

donde  ${}^k \mu$  es un factor de escala,  ${}^k e = \mathbf{A} {}^k \mathbf{R} {}^k t_d$  es la proyección en la imagen capturada por la cámara en el instante  $k$  del centro óptico cuando la cámara está en la posición final,  $\beta_i$  es un factor de escala constante que depende de la distancia  ${}^N \xi$ :

$$\beta_i = \frac{d(P_i, \Pi)}{{}^N \xi} \quad (6.16)$$

donde  $d(P_i, \Pi)$  es la distancia desde el plano de la superficie de contacto hasta el punto 3D, y  $\mathbf{A}$  es la matriz de parámetros intrínsecos:

$$\mathbf{A} = \begin{bmatrix} f \cdot p_u & -f \cdot p_u \cdot \cot(\theta) & u_0 \\ 0 & f \cdot p_v / \text{sen}(\theta) & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (6.17)$$

donde  $u_0$  y  $v_0$  son las coordenadas en píxeles del punto principal,  $f$  es la longitud focal,  $p_u$  y  $p_v$  son los factores de escalado de los píxeles respectivamente en las direcciones  $u$  y  $v$ , y  $\theta$  es el ángulo entre estos ejes.

Aunque  $\beta_i$  es desconocido, aplicando (6.15) entre la posición inicial de la cámara (la obtenida cuando se detecta la colisión y que se indica en adelante con el superíndice 0) y la posición final de la cámara (indicada con el superíndice N) se obtiene que:

$$\beta_i = \text{sign} \left( \frac{({}^0 \mu_i {}^0 \mathbf{f}_i - {}^0 \mathbf{G} {}^N \mathbf{f}_i)_1}{(\mathbf{A} {}^0 \mathbf{R} {}^0 t_d)_1} \right) \frac{\|{}^0 \mathbf{G} {}^N \mathbf{f}_i \wedge {}^0 \mathbf{f}_i\|}{\|\mathbf{A} {}^0 \mathbf{R} {}^0 t_d \wedge {}^0 \mathbf{f}_i\|} \quad (6.18)$$

donde el subíndice 1 indica el primer elemento del vector y  $\mathbf{A}$  se calcula durante el seguimiento intemporal de trayectorias en la imagen basado en control visual virtual cuando se utiliza la calibración multi-imagen definida en el Capítulo 5.

Remplazando (6.18) en (6.15) se obtienen las coordenadas en la imagen de las características simplemente dividiendo  ${}^k \mu_i {}^k \mathbf{f}_i$  por su última componente (de esta forma se obtienen las coordenadas homogéneas de  ${}^k \mathbf{f}_i$ ). Así, en el instante  $k$  se pueden calcular las coordenadas en píxeles de una característica a partir de la Ecuación:

$${}^k\mu_i {}^k f_i = {}^k\mathbf{G}^N f_i + \text{sign} \left( \frac{({}^0\mu_i {}^0 f_i - {}^0\mathbf{G}^N f_i)_1}{({}^0\mathbf{A}^0 \mathbf{R}^0 t_d)_1} \right) \frac{\|{}^0\mathbf{G}^N f_i \wedge {}^0 f_i\|}{\|{}^0\mathbf{A}^0 \mathbf{R}^0 t_d \wedge {}^0 f_i\|} \mathbf{A}^k \mathbf{R}^k t_d \quad (6.19)$$

Como se puede ver en la Ecuación (6.19), para calcular  ${}^k f_i$  se deben obtener las matrices de homografía  ${}^0\mathbf{G}$  y  ${}^k\mathbf{G}$ , al mismo tiempo que la rotación  ${}^0\mathbf{R}$  y la traslación escalada  ${}^0 t_d$ .

Si  $P_i$  está en la superficie de contacto,  $\beta_i$  es nulo. Por lo tanto, aplicando (6.15) entre las posiciones inicial y final en la trayectoria se tiene:

$${}^0\mu_i {}^0 f_i = {}^0\mathbf{G}^N f_i \quad (6.20)$$

La matriz de homografía que relaciona las posiciones inicial y final  ${}^0\mathbf{G}$  puede obtenerse a partir de la expresión (6.20) si se tienen al menos cuatro puntos situados en el plano de la superficie de contacto [Hartley-2000]. Para obtener  ${}^0\mathbf{R}$  and  ${}^0 t_d$  se introduce el concepto de matriz de homografía euclídea  $\mathbf{H}$ .

A partir de la homografía proyectiva  $\mathbf{G}$  se puede obtener la homografía Euclídea  $\mathbf{H}$  como sigue:

$$\mathbf{H} = \mathbf{A}^{-1} \mathbf{G} \mathbf{A} \quad (6.21)$$

A partir de  $\mathbf{H}$  es posible determinar el movimiento de la cámara aplicando el algoritmo propuesto en [ZhangZ-1996]. Por lo que aplicando este algoritmo entre las posiciones inicial y final se pueden calcular  ${}^0\mathbf{R}$  y  ${}^0 t_d$ .

De esta forma se han obtenido  ${}^0\mathbf{G}$ ,  ${}^0\mathbf{R}$  y  ${}^0 t_d$ . Ahora, se necesita calcular  ${}^k\mathbf{G}$  con el objetivo de obtener la posición de las características en el plano imagen de acuerdo a la trayectoria planificada. Para conseguirlo, se debe obtener previamente la matriz de homografía Euclídea  ${}^k\mathbf{H}$ . Posteriormente, es sencillo obtener  ${}^k\mathbf{G}$  a partir de la Ecuación (6.21):

$${}^k\mathbf{G} = \mathbf{A}^k \mathbf{H} \mathbf{A}^{-1} \quad (6.22)$$

$\mathbf{H}$  se puede descomponer en una matriz de rotación y una de rango 1:

$$\mathbf{H} = \mathbf{R} - \mathbf{R} t_d n \quad (6.23)$$

Aplicando la Ecuación (6.23) se puede obtener  ${}^k\mathbf{H} = {}^k\mathbf{R} - {}^k\mathbf{R}^k t_d n$  (en cualquier posición de la trayectoria planificada). Donde  ${}^k\mathbf{R}$  y  ${}^k t_d$  se obtienen del planificador de trayectorias [Pomares-2005a] como se ha descrito previamente. Los diferentes pasos descritos en este apartado para la obtención de la nueva trayectoria se muestra en la Figura 6-6.

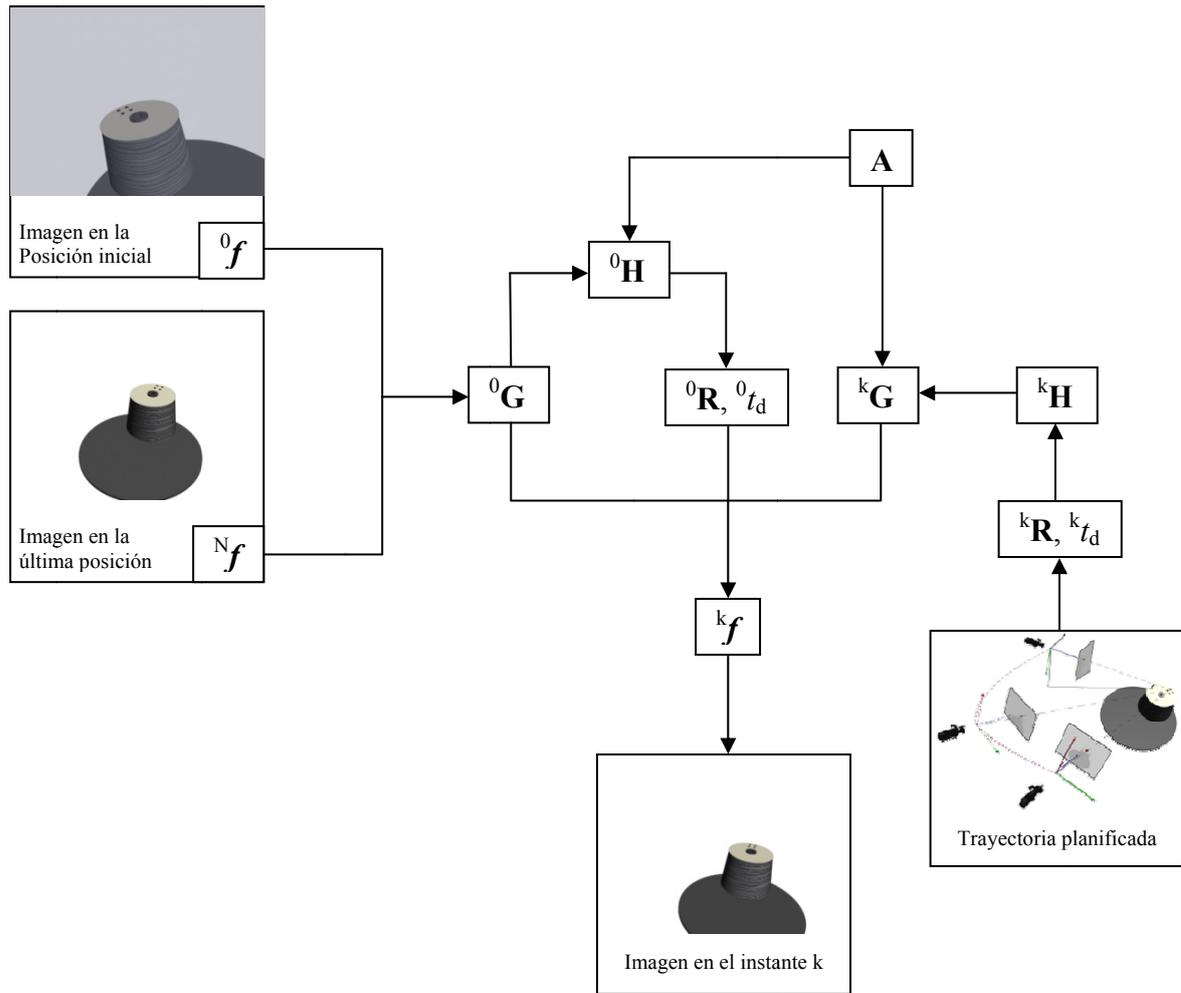


Figura 6-6. Esquema de la reconstrucción de la trayectoria en la imagen.

## 6.2.3. Detección de un cambio en la superficie

### 6.2.3.1. Detección de discontinuidad de superficies a partir de las fuerzas de interacción

Para asegurar que una tarea determinada en la que se requiere una interacción con el espacio de trabajo se desarrolla correctamente, el sistema debe llevar a cabo una variación en la trayectoria en la imagen, dependiendo de las restricciones espaciales impuestas por las fuerzas de interacción. En el apartado anterior se describió un método que permite obtener esa variación de la trayectoria en la imagen para poder seguir con la tarea a pesar de las interacciones con el entorno.

En anteriores trabajos [Pomares-2005a] se ha propuesto un método para detectar discontinuidades en la superficie de interacción basado en el GLR (Generalized Likelihood Ratio) [Willsky-1976]. Este método tiene ciertos posibles problemas por lo que se muestra un estudio más extensivo en este capítulo. Previamente, el GLR se ha aplicado en otros entornos como la estimación de movimiento 2D en una secuencia de imágenes [Faugeras-1998] y la detección de discontinuidades en curvaturas de carretera [Behringer-1995].

Para reducir el nivel de ruido en las fuerzas de interacción obtenidas cuando el robot está siguiendo una superficie, dichas fuerzas se filtran usando un filtro de Kalman cuyas principales ecuaciones son:

$${}^{k+1}x = \mathbf{F} \cdot {}^kx + {}^kv \quad (6.24)$$

$${}^kz = \mathbf{H} \cdot {}^kx + {}^kw \quad (6.25)$$

donde:

- ${}^kx \in \mathfrak{R}^n$  es el vector de estado.
- ${}^kz = \{z_i, i = 1, \dots, k\}$  es el vector de medida.
- $\mathbf{F}$  es la matriz de estado, que relaciona el estado en el paso previo  $k-1$  con el estado actual  $k$  sin ruido.
- $\mathbf{H}$  es la matriz de medida, que relaciona el estado con la medida  ${}^kz$ .
- Se asume que las variables aleatorias  ${}^kv$  y  ${}^kw$  son independientes y tienen una distribución normal, es decir,  $p(v) \sim N(0, \mathbf{Q})$ ,  $p(w) \sim N(0, \mathbf{R})$  (los valores de  $\mathbf{Q}$  y  $\mathbf{R}$  son determinados experimentalmente y se consideran iguales a  $10^{-5}$  y  $10^{-2}$  respectivamente).

Se denotan las fuerzas de interacción obtenidas como  $\mathbf{F} (f_x, f_y, f_z, n_x, n_y, n_z)$ , las cuales componen el vector de estados. Durante la tarea, las fuerzas deben permanecer constantes, y por lo tanto, el modelo de estado será:

$${}^{k+1}\mathbf{F} = \mathbf{I} \cdot {}^k\mathbf{F} + {}^kv \quad (6.26)$$

El filtro se aplica ahora para detectar cambios en la superficie para una determinada dirección, indicada por el vector  $a = [1 \ 0 \ 0 \ 0 \ 0 \ 0]^T$ . La innovación  ${}^k\gamma$  se representa como:

$${}^k\gamma = {}^k\gamma_n + g(k; \theta) \cdot \alpha \quad (6.27)$$

donde:

- ${}^k\gamma_n$  representa la innovación que se obtendrá si no hay cambio (es decir, la diferencia entre la medida actual  ${}^kz$  y una predicción de ella basada en medidas previas):

$${}^k\gamma_n = {}^kz - \mathbf{H} \cdot {}^{k|k-1}x \quad (6.28)$$

- $g(k; \theta)$  es el efecto (en la innovación medida en la iteración  $k$ ) del cambio en la superficie que se produce en la iteración  $\theta$ , en la dirección del cambio  $a$ .
- $\alpha$  es el tamaño de un cambio hipotético en el módulo de la fuerza.

Se considera  $j(k; \theta)$  como el valor del efecto (en el valor de la estimación del vector de estado medido en la iteración  $k$ ) del cambio en la superficie que se produce en la iteración  $\theta$  en la dirección del cambio  $a$ . Si  $\theta < k$ , se obtienen los siguientes valores:

$$\begin{aligned} g(k; \theta) &= a - j(k-1; \theta) \\ j(k; \theta) &= \mathbf{F} \cdot j(k-1; \theta) + {}^k \mathbf{K} \cdot g(k; \theta) \end{aligned} \quad (6.29)$$

La función de prueba para determinar un cambio en las fuerzas de interacción filtradas en la dirección  $a$  serán:

$$l(k; \theta) = \frac{d^2(k; \theta)}{c(k; \theta)} \quad (6.30)$$

donde:

$$c(k; \theta) = \sum_{j=\theta}^k \frac{g^2(j; \theta)}{j_v} \quad (6.31)$$

y:

$$d(k; \theta) = \sum_{j=\theta}^k \frac{g(j; \theta) \cdot j_\gamma}{j_v} \quad (6.32)$$

Este detector determina la presencia de un cambio en el intervalo  $k-\xi < \theta < k$ . El valor de  $\xi$  se establece lo suficientemente alto para compensar el efecto del ruido en la señal. Se ha determinado empíricamente el valor de  $\xi=10$ . Para cualquier  $\theta$  entre  $k$  y  $k-\xi$ , se determina el valor de  $\theta_m$  que maximiza  $l(k; \theta)$  y que representa el momento en el que el cambio es más probable que se produzca. Una vez definida la función de prueba,  $l(k; \theta)$ , se establece el umbral  $U$  de forma que si  $l(k; \theta_m) > U$ , entonces se ha producido un cambio en la superficie (empíricamente, este umbral se ha establecido en 2000). Más detalles en [Pomares-2004a].

En la Figura 6-7 se muestran los valores de la fuerza  $f_z$  y del parámetro  $l(k; \theta_m)$  obtenido durante el seguimiento de tres superficies diferentes (se ha empleado la configuración mostrada en la Figura 6-10). Se puede observar que usando el umbral previamente definido, el sistema puede detectar automáticamente el cambio en la superficie. Sin embargo, este método no es lo suficientemente robusto para detectar el punto exacto en el que se tiene la discontinuidad en la superficie. Por ejemplo, en la Figura 6-7.b hay varios momentos en los que se detecta la discontinuidad (es decir  $l(k; \theta)$  es mayor que 2000). Como muestra la Figura 6-7.c, se pueden obtener falsas detecciones cuando existen grandes rozamientos. Además, estas detecciones falsas pueden aparecer en otras situaciones debido al alto nivel de ruido en la señal obtenida del sensor de fuerza. Por lo tanto, se concluye que este método

es adecuado para obtener un valor acerca de la probabilidad de una discontinuidad en una superficie. Sin embargo, para mejorar la robustez en la detección se debe emplear información adicional. En la siguiente sección se muestra un método basado en luz estructurada que combinado con el propuesto hasta ahora permite obtener un método robusto para la detección de discontinuidades en las superficies de contacto.

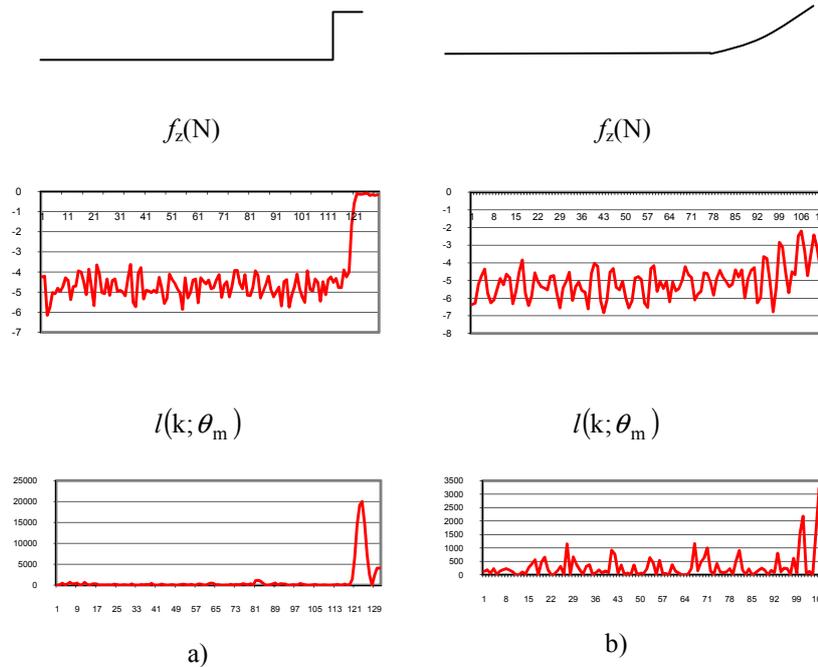


Figura 6-7. Forma de la superficie, evolución de  $f_z$  y parámetro  $l(k; \theta_m)$ . a) Seguimiento de una superficie con un salto. b) Seguimiento de una superficie con un cambio suave.

### 6.2.3.2. Luz estructurada para la detección de zonas de discontinuidades en el espacio imagen

En [Gil-2006] se describe una técnica para estimar zonas de discontinuidad en una superficie determinada en el espacio imagen. Esta aproximación está basada en el uso de patrones de luz estructurada que son proyectados en las superficies de los objetos de la escena. La proyección del haz de luz sobre las distintas superficies determina ciertas discontinuidades debido al cambio de dirección de los planos de proyección. Las características 3D del objeto, como los valores de profundidad y las orientaciones de la superficie producen una función diferente de distribución de la reflexión y, de esta forma, distintos planos de proyección. El sistema empleado para el proceso de detección se compone de una cámara CCD, un láser LASIRIS SNF con un ancho de onda de 660nm, una potencia de 20mW, un ángulo de aproximadamente 15° y 7 círculos concéntricos con un ángulo entre haz de 0.77°. Este laser está localizado cerca de la cámara en el extremo del robot. De esta forma, el sistema de cámara-laser escanea la escena en cada una de las trayectorias ejecutadas por el robot.

Cuando más de un haz de luz golpea diferentes superficies, es decir, superficies con diferentes orientaciones o distinto valor de profundidad de un mismo objeto (o superficies u objetos distintos), se

observa un efecto de discontinuidad en los planos de proyección. La discontinuidad es causada por la rotura de la proyección del haz de luz; así, se pueden observar varios planos de proyección desplazados en el espacio. El patrón empleado consiste en círculos concéntricos. Este método proporciona una línea donde se tiene una discontinuidad en la superficie. Este proceso se detalla en [Pomares-2006e].

### **6.2.3.3. Luz estructurada y fuerzas de interacción para la detección conjunta de cambios en las superficies**

Hasta el momento se han mostrado dos técnicas para la detección de cambios en una superficie. Sin embargo, ambos métodos independientemente no son lo suficientemente robustos para detectar con gran precisión el momento en el que la superficie cambia. Para impedir la limitación de ambos métodos se va a describir a continuación el uso de los dos sensores conjuntamente. Para conseguirlo, la luz estructurada se usa para determinar las zonas de discontinuidad en las que el método descrito en la Sección 6.2.3.1 se aplica. De esta forma, no se obtienen las falsas detecciones.

Cuando el sistema descrito en el apartado 6.2.3.2 basado en luz estructurada determina una zona de discontinuidad, se obtiene una línea en el espacio imagen que representa esta discontinuidad  ${}^0\mathbf{l}_d$ . Como muestra la Figura 6-8, a partir de la información visual es posible determinar no sólo las características visuales empleadas por el sistema de control visual, sino también el punto,  $\mathbf{p}_c$ , en el que la herramienta situada en el extremo del robot colisiona con la superficie. En consecuencia, el algoritmo descrito en el apartado 6.2.3.1 se activará en el momento en el que  $\mathbf{p}_c$  llegue a la línea de discontinuidad  ${}^0\mathbf{l}_d$ , evitando de esta forma las falsas detecciones. La estimación de la iteración en la que el punto  $\mathbf{p}_c$  esta cerca de la línea recta  ${}^0\mathbf{l}_d$  es el objeto del presente apartado.

Una vez que se detecta la línea  ${}^0\mathbf{l}_d$ , su posición en futuras iteraciones se estima utilizando la información sobre la posición de las características en la imagen que emplea el sistema de control visual. En una determinada iteración, se conocen la posición actual de las características  ${}^k\mathbf{s}$  y las deseadas  ${}^{k+1}\mathbf{s}$ , obtenidas a partir del método de seguimiento intemporal de trayectorias basado en control visual virtual descrito en el Capítulo 4. Usando esta información se puede calcular la matriz de homografía que relaciona estas dos posiciones  $\mathbf{G}_k$  tal y como se describe en [Hartley-2000]. La Figura 6-8 muestra las posiciones actual y deseada de las características visuales en una determinada iteración de la tarea de control visual.

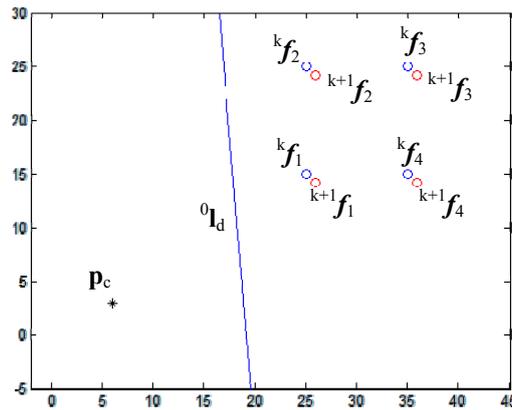


Figura 6-8. Características extraídas en la imagen y línea de discontinuidad  ${}^0l_d$ .

Una vez que se ha calculado la matriz de homografía, es posible obtener la rotación  $\mathbf{R}_k$  y la traslación  $\mathbf{t}_k$  que la cámara ha realizado a partir de  $\mathbf{G}_k$ , aplicando el algoritmo mostrado en [ZhangZ-1996]. Cuando se ha obtenido la rotación y traslación de la cámara, se puede localizar en la siguiente iteración  $i$  la línea recta  ${}^i l_d$  dada por el algoritmo descrito en 6.2.3.2. Para conseguirlo, se toman dos puntos de la recta. Aplicando la siguiente ecuación se determina la posición de cada uno de estos puntos a partir del movimiento de la cámara mencionado anteriormente:

$$\mathbf{x}_1 = \mathbf{R}_k^{-1}(\mathbf{x}_0 - \mathbf{t}_k) \quad (6.33)$$

donde  $\mathbf{x}_0$  y  $\mathbf{x}_1$  indican las coordenadas del punto en la imagen en la iteración actual y los obtenidos después de aplicar el movimiento de la cámara respectivamente. Calculando el movimiento de los dos puntos, se puede reconstruir la posición de la línea virtual  ${}^i l_d$  en el plano imagen.

El algoritmo calcula la posición de la línea recta  ${}^i l_d$  en cada iteración de la tarea de control visual hasta que el extremo del robot,  $\mathbf{p}_c$ , está cerca de esta línea. En este momento se emplea el algoritmo descrito en la Sección 6.2.3.1 para determinar con precisión el punto exacto de discontinuidad. En la Figura 6-9 se puede ver la simulación del cálculo de la posición de la línea virtual  ${}^0 l_d$  durante la tarea de control visual. El extremo del robot,  $\mathbf{p}_c$ , tiene una posición fija en el plano imagen (hay una distancia constante entre él y la cámara). Las posición inicial se indica con  ${}^0 l_d$ . Las líneas representadas con  ${}^i l_d$  son las sucesivas estimaciones obtenidas por el algoritmo descrito en este apartado. Finalmente, la línea representada por  ${}^f l_d$  ilustra la obtenida en la iteración en la que el algoritmo detecta que la herramienta situada en el extremo del robot está suficientemente cerca de esta línea. En este punto, se activa el método descrito en el apartado 6.2.3.1.

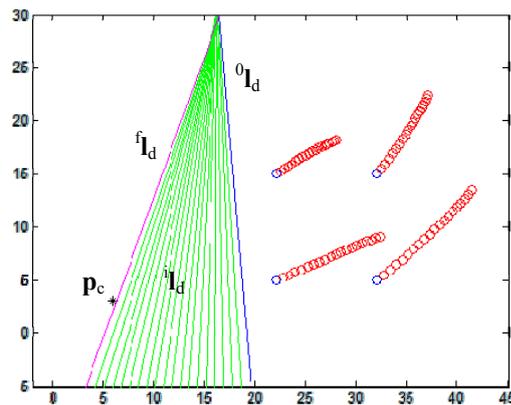


Figura 6-9. Proyección de la discontinuidad durante la tarea de control visual.

## 6.2.4. Resultados

### 6.2.4.1. Arquitectura del sistema

La arquitectura del sistema está compuesta de una cámara en configuración eye-in-hand PHOTONFOCUS MV-D752-160-CL-8 en el extremo de un robot de 7 grados de libertad Mitsubishi PA-10 (ver Figura 6-10) también equipado con un sensor de fuerza (67M25A-I40 de JR3. Inc.) y un laser LASIRIS SNF con un ancho de onda de 660nm. Se usa una Coreco X64CL-Pro como tarjeta de adquisición y procesamiento de imágenes. La cámara es capaz de adquirir y procesar hasta 100 frames/s usando una resolución de imagen de 320x240. Al no estar interesados en tareas de proceso de imágenes en estos experimentos, la trayectoria en la imagen se genera usando un patrón con cuatro puntos que serán las características extraídas para realizar el control visual.

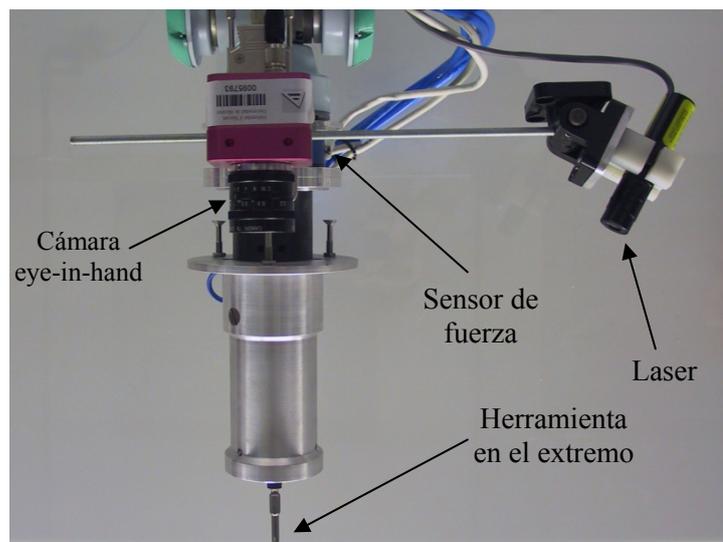


Figura 6-10. Arquitectura del sistema para la experimentación.

### 6.2.4.2. Detección de cambios en la superficie

En este apartado se presenta un experimento para seguir una superficie curvada que valide los algoritmos propuestos en la sección 6.2.3. En este caso, una vez que la discontinuidad se detecta y la nueva superficie se ha reconocido [Pomares-2005a], el robot se reorienta con respecto a esta nueva superficie. Aplicando el algoritmo para fusionar la información visual y de fuerza descrito en el apartado 6.2.1, se obtiene la trayectoria 3D representada en la Figura 6-11. Como se describe en el apartado 6.2.2, una vez que la nueva superficie se ha reconocido, la trayectoria en la imagen se modifica de forma que sea posible utilizar la información de imagen y de fuerza para controlar la tarea. En esta figura se puede observar que el robot permanece perpendicular con respecto a la superficie durante la trayectoria.

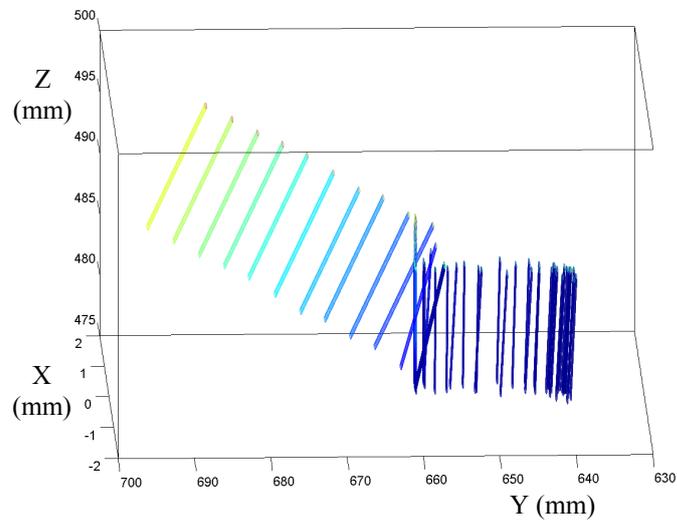


Figura 6-11. Seguimiento de una superficie curva.

Para verificar que el sistema puede detectar el cambio en la superficie usando únicamente la información del sensor de fuerza, en la Figura 6-12 se representan las fuerzas de interacción obtenidas (en la dirección z) y el valor de  $l(k; \theta)$ . Como se indicaba anteriormente, el umbral usado para la detección de los cambios en la superficie es 2000, de forma que el sistema es capaz de detectar el cambio.

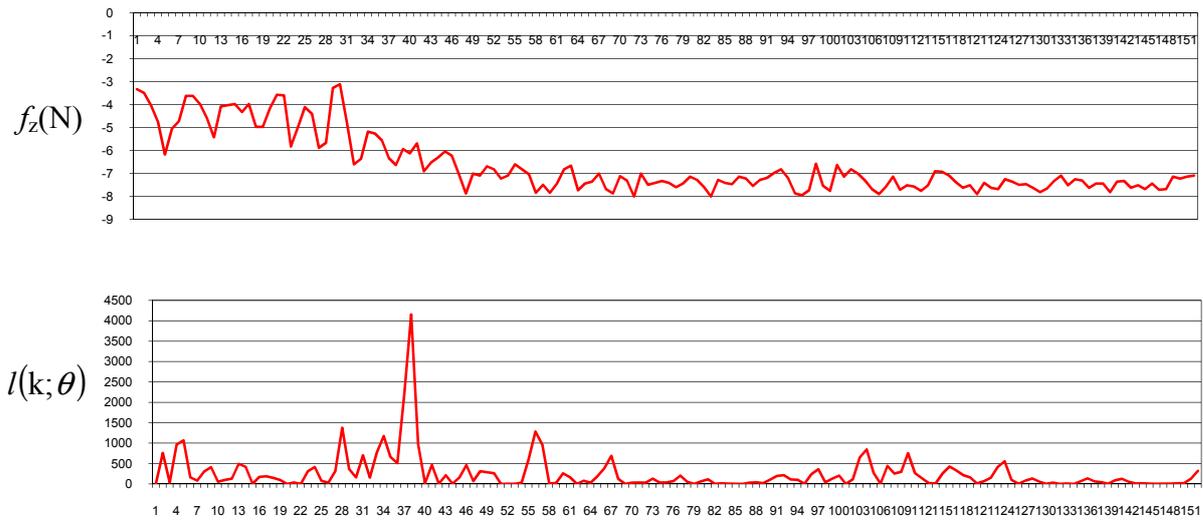


Figura 6-12. Evolución de la fuerza  $f_z$  and  $l(k; \theta)$ .

Sin embargo, tal y como se ha mostrado en la sección 6.2.3 este método puede proporcionar falsas detecciones de discontinuidades. En el siguiente apartado, se describe un nuevo experimento para mostrar cómo el uso de luz estructurada puede mejorar el comportamiento del sistema.

#### 6.2.4.3. Mejora de la detección de cambios en la superficie usando el sistema multisensorial

Ahora, se describe un experimento en el que se generan más fricciones durante el seguimiento de la superficie. En este caso, la discontinuidad de la superficie se muestra en la Figura 6-13. Las fuerzas de interacción y el parámetro  $l(k; \theta)$  obtenidos en el seguimiento de la superficie usando la técnica descrita en el apartado 6.2.3.1 se muestran en la Figura 6-14. Se puede observar que el sistema es capaz de detectar la zona de discontinuidad, pero también aporta falsas detecciones. Para mejorar el comportamiento el siguiente paso es limitar las zonas donde la discontinuidad puede aparecer usando el método basado en luz estructurada descrito en la Sección 6.2.3.2. Usando este método, se obtiene la zona de discontinuidad representada por dos líneas rectas en la Figura 6-15.a. El último paso es determinar cuándo se encuentra el extremo cerca de la zona de discontinuidad de forma que el algoritmo descrito en la Sección 6.2.3.1 se ejecute sólo en esta zona, evitando las detecciones falsas. Siguiendo el proceso descrito en 6.2.3.3, una vez que la línea de discontinuidad se ha detectado  $\mathbf{l}_d$ , la rotación y traslación de la cámara se determina en cada iteración de la tarea de control visual. Esta información se emplea para proyectar la posición de la línea de discontinuidad en cada iteración,  ${}^k\mathbf{l}_d$ . De esta forma, cuando la línea esté lo suficientemente cerca de la posición del extremo del robot, se ejecuta el algoritmo descrito en la sección 6.2.3.1 para determinar el punto exacto de la discontinuidad (este proceso se ilustra en la Figura 6-15.b).

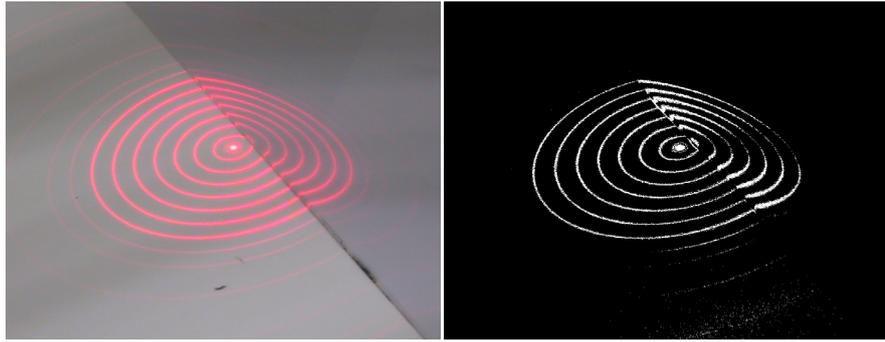


Figura 6-13. Discontinuidad de la superficie.

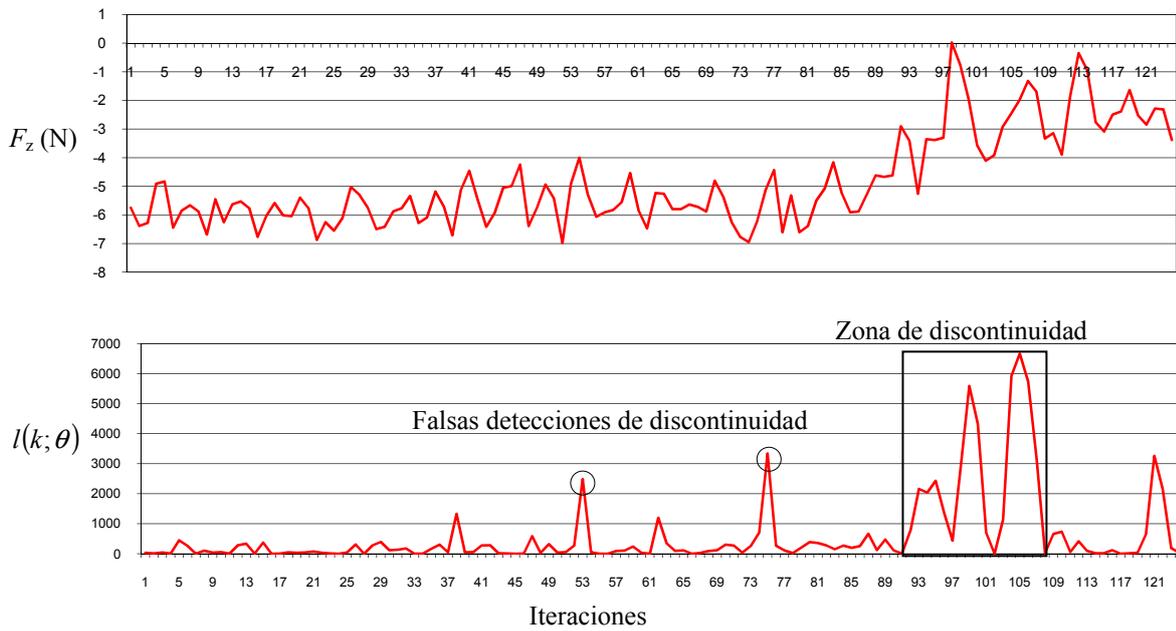


Figura 6-14. Evolución de la fuerza  $f_z$  and  $l(k; \theta)$ .

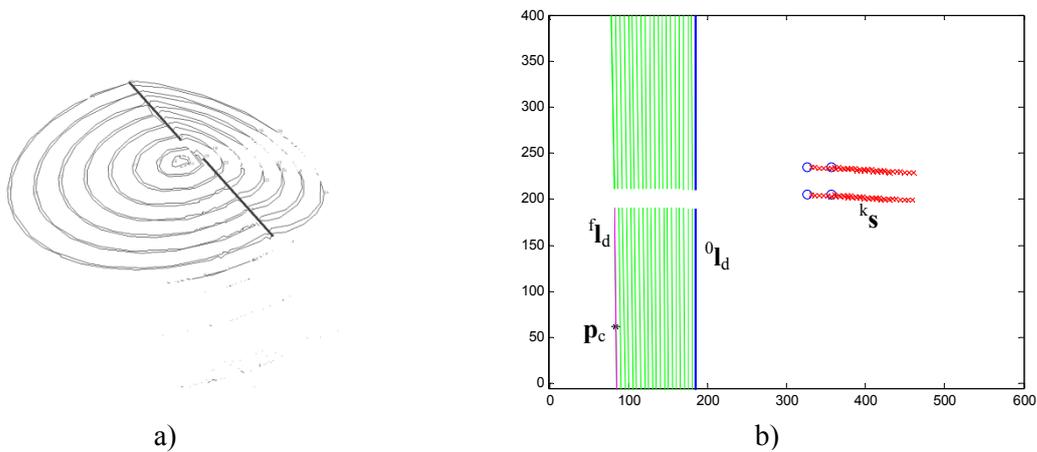


Figura 6-15.a) Zona de discontinuidad usando el método descrito en el apartado 6.2.3.2. b) Proceso llevado a cabo para determinar el punto en el que el extremo llega a la discontinuidad.

#### 6.2.4.4. Tarea de inserción que requiere la fusión de visión y fuerza

El siguiente experimento utiliza la configuración descrita en la Figura 6-10 para insertar el extremo de la herramienta en el objeto representado en la Figura 6-16.



Figura 6-16. Objeto a seguir por el sistema de seguimiento de trayectorias en la imagen basado en control visual virtual.

Debido a la gran precisión que se requiere para la inserción, una inserción perfecta es imposible de garantizar si se usa únicamente información visual. Como se muestra en la Figura 6-17, sin control de fuerza, el más leve contacto con el borde de la apertura causa que las fuerzas se incrementen rápidamente y que la inserción no se lleve a cabo. Para demostrar el comportamiento del sistema utilizando información visual y de fuerza de forma simultánea, se describen a continuación dos experimentos, en los que la inserción se ejecuta en diferentes posiciones. En la Figura 6-18, se muestra la trayectoria de la herramienta situada en el extremo del robot durante las dos inserciones. La apertura en la que se realiza la inserción se muestra para verificar el correcto comportamiento.

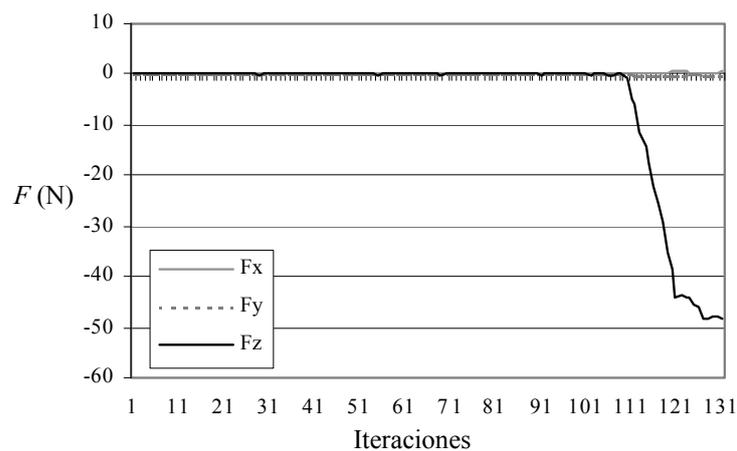


Figura 6-17. Fuerzas obtenidas sin control de fuerzas.

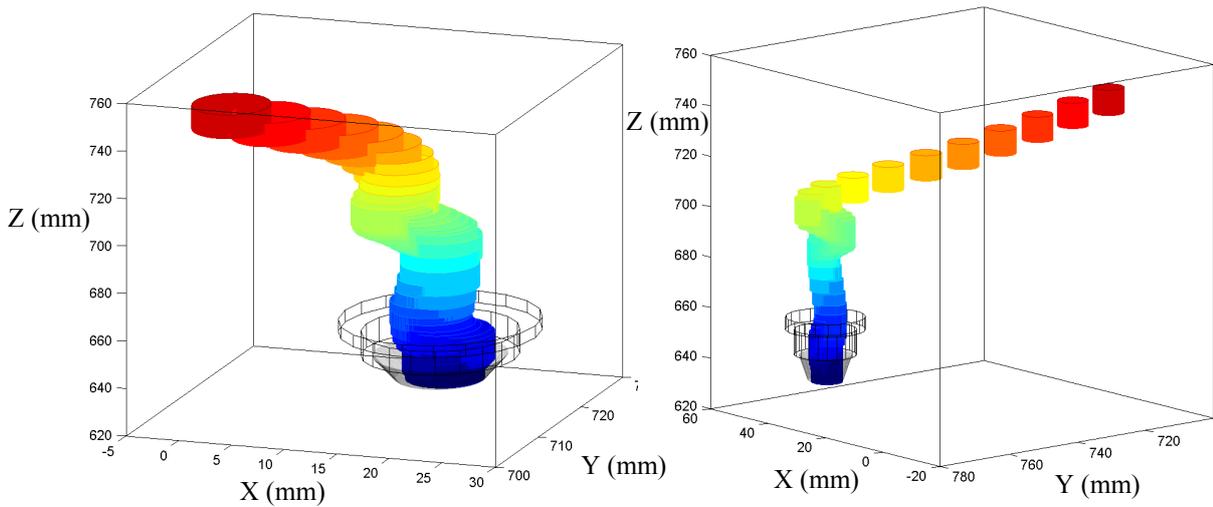


Figura 6-18. Trayectoria seguida por el robot en dos experimentos de inserción distintos.

En la Figura 6-19, se observa que al fusionar la información visual y de fuerza, las fuerzas se mantienen bajas durante la trayectoria. Las fuerzas se compensan rápidamente, y la trayectoria se corrige de forma precisa, permitiendo de esta forma la inserción.

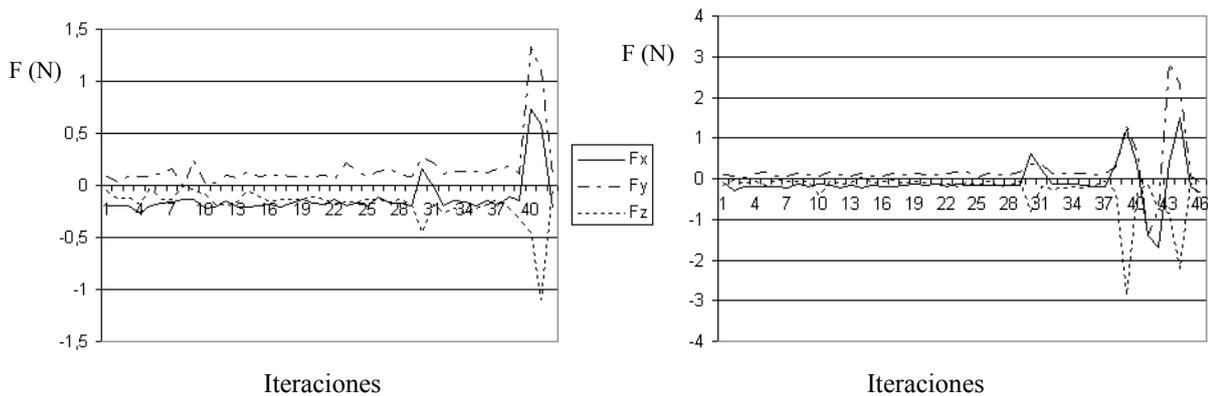


Figura 6-19. Evolución de la fuerza de interacción del robot con su entorno de trabajo en los dos experimentos.

Si la trayectoria no se genera de forma correcta, la inserción no se puede conseguir. Por eso, la trayectoria en la imagen se debe modificar dependiendo de las fuerzas de interacción. Este aspecto queda ilustrado en la Figura 6-20. En este caso, el robot golpea la superficie. Una vez que se ha reconocido esta superficie con los mecanismos descritos en el apartado 6.2.3, la trayectoria en la imagen se modifica (tal y como se explicó en el apartado 6.2.2) permitiendo el uso conjunto de ambos sensores para controlar la tarea.

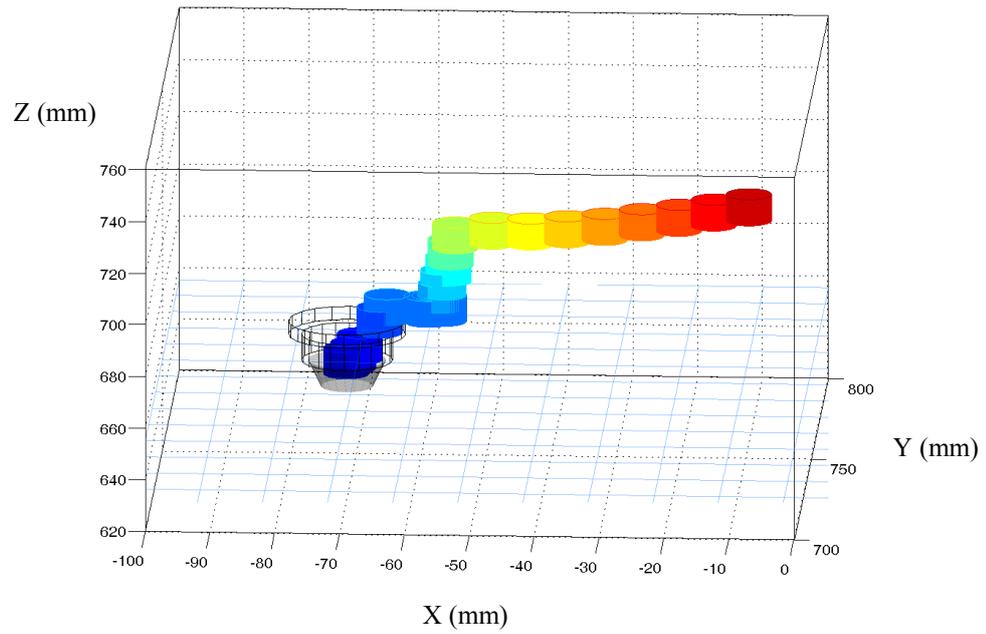


Figura 6-20. Trayectoria en el espacio Cartesiano 3D durante la tarea de inserción cuando el robot colisiona con la superficie.

La trayectoria 3D que el robot debe seguir durante la tarea de inserción representada en la Figura 6-20 se puede ver en el espacio imagen en la Figura 6-21.

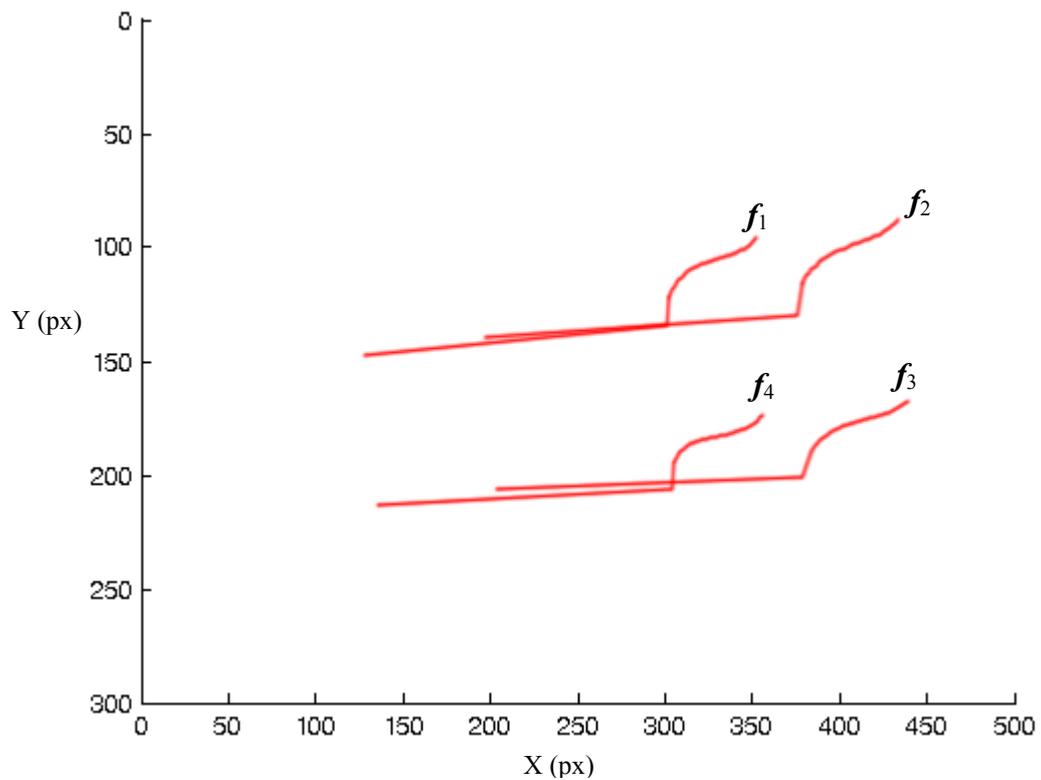


Figura 6-21. Trayectoria deseada en imagen durante la tarea de inserción.

Esta trayectoria necesita ser modificada por el hecho de que el robot colisiona con la superficie del objeto antes de que llegue a su posición final. La Figura 6-22 muestra la trayectoria en el espacio imagen de las características en el experimento representado en la Figura 6-21. El sistema detecta la colisión y la trayectoria se replanifica de forma que la inserción pueda completarse siguiendo la trayectoria más cercana a la deseada permitida por las restricciones (ver apartado 6.2.2).

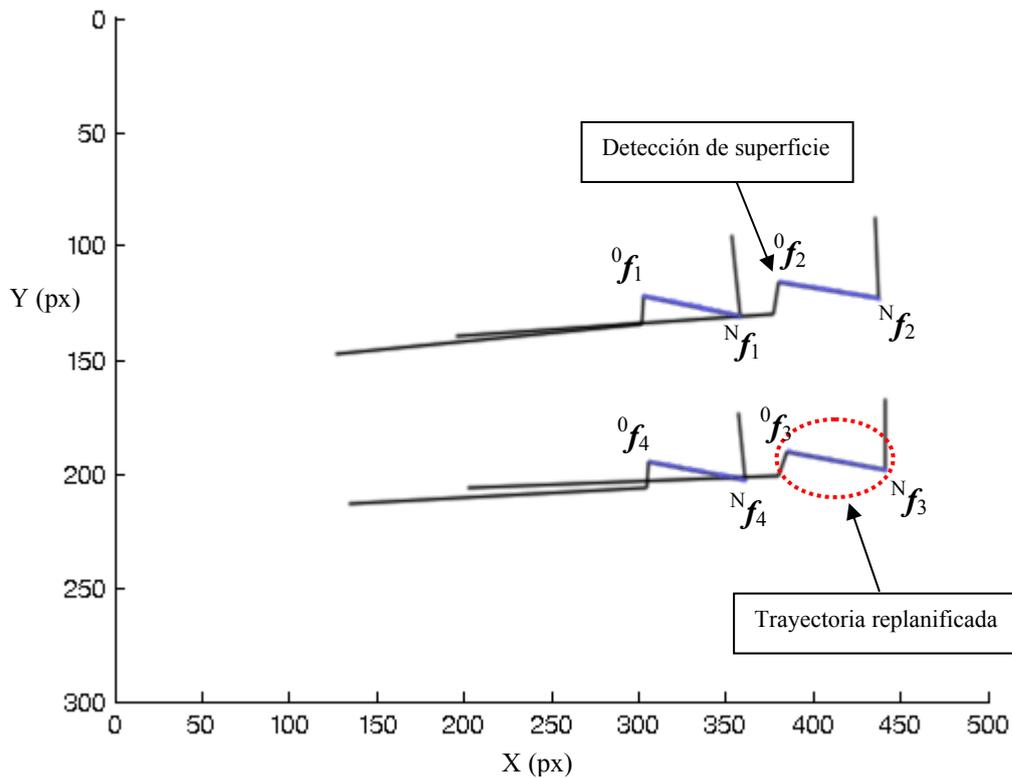


Figura 6-22. Trayectoria seguida por el robot durante la tarea de inserción cuando colisiona con la superficie.

### 6.3. Fusión de control visual-fuerza basado en la matriz de interacción Fuerza-Imagen

La necesidad de realizar una calibración previa se hace patente en sistemas de control visual-fuerza. En este tipo de sistemas de control el robot realiza simultáneamente una tarea de control visual y control de la interacción con el entorno. En tareas de control visual y fuerza aparecen parámetros desconocidos a priori tales como rozamientos, relaciones entre sistemas de coordenadas de los sensores, características de la superficie de contacto, etc.

En este apartado se propone un sistema de control visual-fuerza que no requiere una calibración previa entre el sistema de visión y el sistema sensor de fuerza [GarcíaGJ-2007a]. Para combinar la información sensorial se realiza una modificación de la trayectoria en la imagen seguida por el sistema de control visual en base a la información procedente del sensor de fuerza. Para ello se ha definido el concepto de matriz de interacción fuerza-imagen. Esta matriz determina la relación entre variaciones en la imagen y variaciones en las fuerzas de interacción. Este método es capaz de realizar una estimación online de dicha matriz sin necesidad de disponer de conocimiento previo de los parámetros anteriormente mencionados. El método realiza una estimación en base a la información obtenida de ambos sensores. En concreto, el sistema empleado para la estimación se basa en el método recursivo Gauss-Newton y usa una optimización no lineal de mínimos cuadrados. Este método se basa en trabajos previos como [Hosoda-1994, Jagersand-1997, Piepmeier-2003] dirigidos al desarrollo de un método de Gauss-Newton dinámico de control visual para el seguimiento de objetos en movimiento. En concreto, estos métodos se encargan de realizar una estimación robusta del Jacobiano de la imagen que relaciona las características visuales con la variación en las coordenadas articulares del sistema.

### 6.3.1. Control Visual-Fuerza mediante la matriz de interacción fuerza-imagen

En este apartado se describe el significado de la matriz de interacción fuerza-imagen,  $\mathbf{L}_{FI}$ , que se pretende estimar. Para ello, considerando  $\mathbf{F}$  las fuerzas de interacción obtenidas respecto al extremo del robot y  $\mathbf{r}$  la localización del objeto, se define la matriz de interacción para las fuerzas de interacción,  $\mathbf{L}_F$ , de la siguiente manera:

$$\mathbf{L}_F = \frac{\partial \mathbf{F}}{\partial \mathbf{r}} \rightarrow \mathbf{L}_F^+ = (\mathbf{L}_F^T \mathbf{L}_F)^{-1} \mathbf{L}_F^T \frac{\partial \mathbf{r}}{\partial \mathbf{F}} \quad (6.34)$$

Por otro lado, se considera  $\mathbf{q}$  las coordenadas articulares del robot y  $\mathbf{J}$  el Jacobiano del robot, por lo tanto:

$$\mathbf{v}_e = \dot{\mathbf{r}} = \mathbf{J} \cdot \dot{\mathbf{q}} \rightarrow \dot{\mathbf{q}} = \mathbf{J}^+ \cdot \dot{\mathbf{r}} \quad (6.35)$$

donde  $\mathbf{v}_e = [\dot{\mathbf{p}}_e \quad \boldsymbol{\omega}_e]$ ,  $\dot{\mathbf{p}}_e$  es el vector 3x1 de la velocidad lineal del extremo del robot, y  $\boldsymbol{\omega}_e$  la velocidad angular del extremo.

Con  $\mathbf{J}_I$  se representa el Jacobiano de la imagen [Hutchinson-1996] que relaciona las variaciones en la imagen con la variación en el espacio articular, es decir,  $\dot{\mathbf{s}} = \mathbf{J}_I \cdot \dot{\mathbf{q}}$ , donde  $\mathbf{s} = [f_1, f_2, \dots, f_M]^T$  es el conjunto de características extraídas de la imagen. A partir de esta última relación y aplicando (6.35) se tiene que:

$$\begin{aligned}\dot{\mathbf{s}} &= \mathbf{J}_1 \cdot \mathbf{J}^+ \cdot \frac{\partial \mathbf{r}}{\partial t} = \mathbf{J}_1 \cdot \mathbf{J}^+ \cdot \frac{\partial \mathbf{r}}{\partial \mathbf{F}} \cdot \frac{\partial \mathbf{F}}{\partial t} \\ \dot{\mathbf{s}} &= \mathbf{J}_1 \cdot \mathbf{J}^+ \cdot \mathbf{L}_F^+ \cdot \dot{\mathbf{F}} \rightarrow \dot{\mathbf{s}} = \mathbf{L}_{FI} \cdot \dot{\mathbf{F}}\end{aligned}\quad (6.36)$$

siendo  $\mathbf{L}_{FI}$  la matriz de interacción que se desea estimar empleando mínimos cuadrados ponderados exponencialmente.

En una aplicación en la que sea necesario mantener una fuerza constante con el entorno, la trayectoria en imagen se debe modificar dependiendo de las fuerzas de interacción (véase apartado 6.2.2). Para ello, usando la matriz  $\mathbf{L}_{FI}$ , las nuevas características deseadas (empleadas por el sistema de seguimiento intemporal de trayectorias basado en control visual virtual descrito en el Capítulo 4) durante el contacto serán:

$${}^j\mathbf{s}' = {}^j\mathbf{s} + \mathbf{L}_{FI} \cdot (\mathbf{F} - \mathbf{F}_d) \quad (6.37)$$

### 6.3.2. Estimación de la matriz de interacción fuerza-imagen

En esta sección se va a describir el método empleado para la estimación de la matriz de interacción  $\mathbf{L}_{FI}$ .

Considerando un sistema MIMO en el que las entradas son variaciones en las fuerzas de interacción y salidas variaciones en las características visuales, la ecuación de medida puede representarse de la siguiente manera:

$$\Delta \mathbf{s} = \mathbf{L}_{FI} \cdot \Delta \mathbf{F} + \mathbf{e} \quad (6.38)$$

donde:

- $\Delta \mathbf{s} = (\Delta s_1 \quad \dots \quad \Delta s_m)^T$  son las salidas del sistema, es decir, la variación en las características visuales.
- $\Delta \mathbf{F} = (\Delta F_1 \quad \dots \quad \Delta F_n)^T$  son las entradas al sistema, es decir, la variación en las fuerzas de interacción.
- $\mathbf{e}$  es un vector de error de medida (ruido blanco).
- $\mathbf{L}_{FI}$  es la matriz de interacción fuerza-imagen anteriormente definida cuyo valor estimado se va a representar por  $\hat{\mathbf{L}}_{FI}$  (por simplicidad en la notación a partir de ahora se representa  $\mathbf{L} = \mathbf{L}_{FI}$ ):

$$\mathbf{L} = \begin{pmatrix} \mathbf{L}_{11} & \dots & \mathbf{L}_{1n} \\ \dots & \dots & \dots \\ \mathbf{L}_{m1} & \dots & \mathbf{L}_{mn} \end{pmatrix} \quad (6.39)$$

$$\hat{\mathbf{L}}_j = (\hat{\mathbf{L}}_{j1} \quad \dots \quad \hat{\mathbf{L}}_{jn}), j=1\dots m$$

Por lo tanto, el valor para cada una de las  $m$  salidas o características visuales será:

$$\Delta s_j = \hat{\mathbf{L}}_j^T \cdot \Delta \mathbf{F} + e_j \quad (6.40)$$

Despejando el error en la expresión (6.40) y considerando  $k$  medidas se obtiene la siguiente ecuación para el error ponderado exponencialmente:

$$\begin{pmatrix} \lambda^{(k-1)/2} e_j(1) \\ \lambda^{(k-2)/2} e_j(2) \\ \vdots \\ e_j(k) \end{pmatrix} = \begin{pmatrix} \lambda^{(k-1)/2} \Delta s_j(1) \\ \lambda^{(k-2)/2} \Delta s_j(2) \\ \vdots \\ \Delta s_j(k) \end{pmatrix} - \begin{pmatrix} \lambda^{(k-1)/2} (\Delta \mathbf{F}(1))^T \\ \lambda^{(k-2)/2} (\Delta \mathbf{F}(2))^T \\ \vdots \\ (\Delta \mathbf{F}(k))^T \end{pmatrix} \cdot \hat{\mathbf{L}}_j \quad (6.41)$$

Donde el parámetro  $\lambda \in [0,1]$  indica el peso de las nuevas medidas frente a estimaciones anteriores. Cuanto mayor es el valor de  $\lambda$  más peso se le asigna a las nuevas medidas. La ecuación (6.41) también puede representarse de la siguiente manera:

$$\bar{e}_{jk} = \bar{\mathbf{S}}_{jk} - \bar{\mathbf{F}}_k \cdot \hat{\mathbf{L}}_j \quad (6.42)$$

Con el objetivo de estimar la matriz de interacción fuerza-imagen se define la siguiente función objetivo a minimizar:

$$E_k = \sum_{j=1}^m \bar{e}_{jk}^T \bar{e}_{jk} = \sum_{j=1}^m (\bar{\mathbf{S}}_{jk} - \bar{\mathbf{F}}_k \cdot \hat{\mathbf{L}}_j)^T (\bar{\mathbf{S}}_{jk} - \bar{\mathbf{F}}_k \cdot \hat{\mathbf{L}}_j) \quad (6.43)$$

La estimación de la matriz de interacción fuerza-imagen puede representarse de la siguiente manera:

$$\hat{\mathbf{L}} = (\hat{\mathbf{L}}_1^T \quad \dots \quad \hat{\mathbf{L}}_m^T)^T \quad (6.44)$$

El valor de  $\hat{\mathbf{L}}_j(k+1)$  puede ser deducido recursivamente a partir de la medida  $(k+1)$  y el valor de  $\hat{\mathbf{L}}_j(k)$ . Para  $(k+1)$  medidas el error ponderado exponencialmente será el siguiente:

$$\begin{pmatrix} \sqrt{\lambda} \bar{e}_{jk} \\ e_j(k+1) \end{pmatrix} = \begin{pmatrix} \sqrt{\lambda} \bar{\mathbf{S}}_{jk} \\ \Delta s_j(k+1) \end{pmatrix} - \begin{pmatrix} \sqrt{\lambda} \bar{\mathbf{F}}_k \\ (\Delta \mathbf{F}(k))^T \end{pmatrix} \cdot \hat{\mathbf{L}}_j \quad (6.45)$$

Además, la siguiente condición deberá cumplirse si se desea reducir en la iteración  $k+1$  completamente el error representado en la Ecuación (6.43):

$$\hat{\mathbf{L}}_j(k+1) = (\bar{\mathbf{F}}_{k+1}^T \bar{\mathbf{F}}_{k+1})^{-1} \bar{\mathbf{F}}_{k+1}^T \bar{\mathbf{S}}_{jk+1} \quad (6.46)$$

Supóngase ahora que:

$$\mathbf{P}(k) = (\bar{\mathbf{F}}_k^T \bar{\mathbf{F}}_k)^{-1} \quad (6.47)$$

Por lo tanto:

$$\mathbf{P}(k+1) = (\bar{\mathbf{F}}_{k+1}^T \bar{\mathbf{F}}_{k+1})^{-1} = \left[ \begin{array}{cc} \sqrt{\lambda} \bar{\mathbf{F}}_k^T & \Delta \mathbf{F}(k+1) \end{array} \right] \left[ \begin{array}{c} \sqrt{\lambda} \bar{\mathbf{F}}_k \\ (\Delta \mathbf{F}(k+1))^T \end{array} \right]^{-1} \quad (6.48)$$

Multiplicando los términos de la Ecuación (6.48) se obtiene:

$$\mathbf{P}(k+1) = \left[ \lambda \bar{\mathbf{F}}_k^T \bar{\mathbf{F}}_k + \Delta \mathbf{F}(k+1) (\Delta \mathbf{F}(k+1))^T \right]^{-1} = \frac{1}{\lambda} \left[ \mathbf{P}^{-1}(k) + \Delta \mathbf{F}(k+1) \lambda^{-1} (\Delta \mathbf{F}(k+1))^T \right]^{-1} \quad (6.49)$$

Finalmente, desarrollando esta expresión:

$$\mathbf{P}(k+1) = \frac{1}{\lambda} \left[ \mathbf{P}(k) - \mathbf{P}(k) \Delta \mathbf{F}(k+1) \cdot \left[ \lambda + (\Delta \mathbf{F}(k+1))^T \mathbf{P}(k) \Delta \mathbf{F}(k+1) \right]^{-1} \cdot (\Delta \mathbf{F}(k+1))^T \mathbf{P}(k) \right] \quad (6.50)$$

Con el objetivo de obtener una representación más compacta de la expresión anterior se define:

$$\mathbf{K}(k+1) = \mathbf{P}(k) \Delta \mathbf{F}(k+1) \cdot \left[ \lambda + (\Delta \mathbf{F}(k+1))^T \mathbf{P}(k) \Delta \mathbf{F}(k+1) \right]^{-1} \quad (6.51)$$

Con lo que sustituyendo (6.51) en (6.50) se llega a:

$$\mathbf{P}(k+1) = \frac{1}{\lambda} \left[ \mathbf{P}(k) - \mathbf{K}(k+1) (\Delta \mathbf{F}(k+1))^T \mathbf{P}(k) \right] \quad (6.52)$$

Por otro lado, a partir de la Ecuación (6.46) se puede deducir que:

$$\begin{aligned} \hat{\mathbf{L}}_j(k+1) &= \mathbf{P}(k+1) \left[ \begin{array}{cc} \sqrt{\lambda} \bar{\mathbf{F}}_k^T & \Delta \mathbf{F}(k+1) \end{array} \right] \left[ \begin{array}{c} \sqrt{\lambda} \bar{\mathbf{S}}_{jk} \\ \Delta s_j(k+1) \end{array} \right] = \\ &= \mathbf{P}(k+1) \left[ \lambda \bar{\mathbf{F}}_k^T \bar{\mathbf{S}}_{jk} + \Delta \mathbf{F}(k+1) \Delta s_j(k+1) \right] \end{aligned} \quad (6.53)$$

Sustituyendo (6.52) en (6.53):

$$\hat{\mathbf{L}}_j(k+1) = \hat{\mathbf{L}}_j(k) + \mathbf{K}(k+1) \cdot \left[ \Delta s_j(k+1) - (\Delta \mathbf{F}(k+1))^T \hat{\mathbf{L}}_j(k) \right] \quad (6.54)$$

A partir de (6.54) se puede obtener:

$$\hat{\mathbf{L}}_j^T(k+1) = \hat{\mathbf{L}}_j^T(k) + \left[ \Delta s_j(k+1) - \hat{\mathbf{L}}_j^T(k) \Delta \mathbf{F}(k+1) \right] \mathbf{K}^T(k+1) \quad (6.55)$$

Agrupando las estimaciones de las m filas de la matriz de interacción se obtiene finalmente:

$$\hat{\mathbf{L}}_j^T(k+1) = \hat{\mathbf{L}}_j^T(k) + \left[ \Delta s_j(k+1) - \hat{\mathbf{L}}_j^T(k) \Delta \mathbf{F}(k+1) \right] \mathbf{K}^T(k+1) \quad (6.56)$$

donde el valor de la matriz  $\mathbf{K}$  se corresponde con la expresión obtenida en (6.51). Finalmente, usando la Ecuación (6.56) de manera recursiva, se obtiene una estimación de la matriz de interacción fuerza-imagen.

Para resumir los diferentes pasos que se requieren para determinar el valor de la matriz de interacción fuerza-imagen, se puede seguir el siguiente algoritmo:

$$\begin{aligned} & \Delta \mathbf{s} : \mathfrak{R}^n \rightarrow \mathfrak{R}^m ; \Delta \mathbf{F}_0, \Delta \mathbf{F}_1 \in \mathfrak{R}^n ; \hat{\mathbf{L}}(0) \in \mathfrak{R}^{m \times n} ; \hat{\mathbf{P}}(0) \in \mathfrak{R}^{n \times n} ; \lambda \in (0,1) \\ & \text{For } k=0, 1, \dots \text{ do} \\ & \quad \Delta \mathbf{s}(k+1) = \mathbf{s}_{k+1} - \mathbf{s}_k \\ & \quad \Delta \mathbf{F}(k+1) = \mathbf{F}_{k+1} - \mathbf{F}_k \\ & \quad \hat{\mathbf{L}}(k+1) = \hat{\mathbf{L}}(k) + \left[ \Delta \mathbf{s}(k+1) - \hat{\mathbf{L}}(k) \Delta \mathbf{F}(k+1) \right] \mathbf{K}^T(k+1) \\ & \quad \mathbf{K}(k+1) = \mathbf{P}(k) \Delta \mathbf{F}(k+1) \cdot \left[ \lambda + (\Delta \mathbf{F}(k+1))^T \mathbf{P}(k) \Delta \mathbf{F}(k+1) \right]^{-1} \\ & \quad \mathbf{P}(k+1) = \frac{1}{\lambda} \left[ \mathbf{P}(k) - \mathbf{K}(k+1) (\Delta \mathbf{F}(k+1))^T \mathbf{P}(k) \right] \\ & \text{end} \end{aligned}$$

### 6.3.3. Resultados

#### 6.3.3.1. Resultados de simulación

Con el objetivo de demostrar el correcto comportamiento del sistema en primer lugar se va a describir un experimento simulado en el cual las características visuales van variando según aumenta la fuerza de interacción en  $z$ . En concreto, se trata de una simulación de un descenso en profundidad en el que la fuerza de interacción en el eje  $z$  aumenta linealmente (1 N. en cada iteración) y las características siguen la evolución que se muestra en Figura 6-23 (un píxel en  $(x,y)$  por cada iteración).

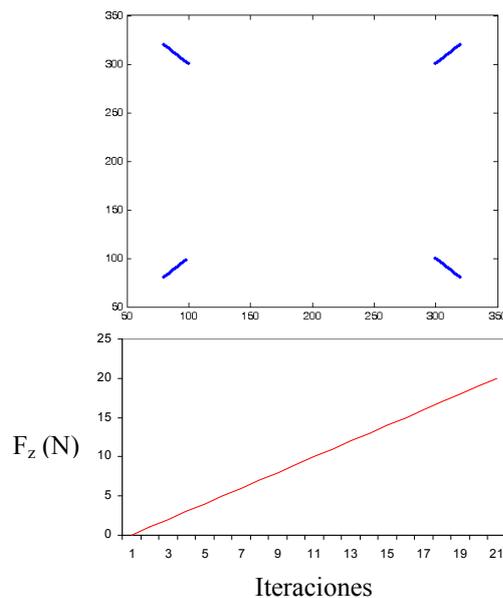


Figura 6-23. Evolución de las características en la imagen y de las fuerzas de interacción durante el experimento.

Los valores iniciales considerados para la matriz de interacción fuerza-imagen  $\hat{\mathbf{L}}(0)$  y  $\hat{\mathbf{P}}(0)$  son respectivamente:

$$\hat{\mathbf{L}}(0) = \begin{bmatrix} 0 & 0 & 0,001 \\ 0 & 0 & 0,001 \\ 0 & 0 & 0,001 \\ 0 & 0 & 0,001 \\ 0 & 0 & 0,001 \\ 0 & 0 & 0,001 \\ 0 & 0 & 0,001 \\ 0 & 0 & 0,001 \\ 0 & 0 & 0,001 \end{bmatrix} \quad \hat{\mathbf{P}}(0) = \begin{bmatrix} 9 & 0 & 0 \\ 0 & 9 & 0 \\ 0 & 0 & 9 \end{bmatrix} \quad (6.57)$$

En 3 iteraciones ya se ha obtenido el muy próximo al adecuado para la matriz de interacción ( $\lambda = 0,9$ ):

$$\hat{\mathbf{L}}(3) = \begin{bmatrix} 0 & 0 & -0.9709 \\ 0 & 0 & -0.9709 \\ 0 & 0 & 0.9710 \\ 0 & 0 & -0.9709 \\ 0 & 0 & 0.9710 \\ 0 & 0 & 0.9710 \\ 0 & 0 & -0.9709 \\ 0 & 0 & 0.9710 \end{bmatrix} \quad (6.58)$$

La matriz de interacción real empleada en la simulación es:

$$\mathbf{L} = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & -1 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \\ 0 & 0 & 1 \end{bmatrix} \quad (6.59)$$

### 6.3.3.2. Resultados experimentales

En estos experimentos se ha realizado el montaje experimental representado en la Figura 6-10. La mesa con la que colisiona el robot presenta cierto grado de compresibilidad. De esta forma dependiendo de la fuerza con la que interacciona el robot el sistema es capaz de observar claramente una modificación de las características en la imagen. A partir de esta información se determina el valor de la matriz de interacción fuerza-imagen según se describió en el apartado 6.3.2. Esta matriz es a su vez empleada para modificar la trayectoria deseada en la imagen de forma que se consiga mantener una fuerza de interacción constante entre el robot y la superficie. Como se ha descrito en trabajos

anteriores [Pomares-2005a] la modificación de la información visual a partir de la información de interacción con el entorno permite garantizar la coherencia entre ambos sistemas sensoriales. De esta forma también se evita que se produzcan acciones de control contradictorias entre ambos sistemas sensoriales. En estos experimentos se ilustran dos ejemplos sencillos pero en los que se observa con claridad el correcto comportamiento del sistema propuesto.

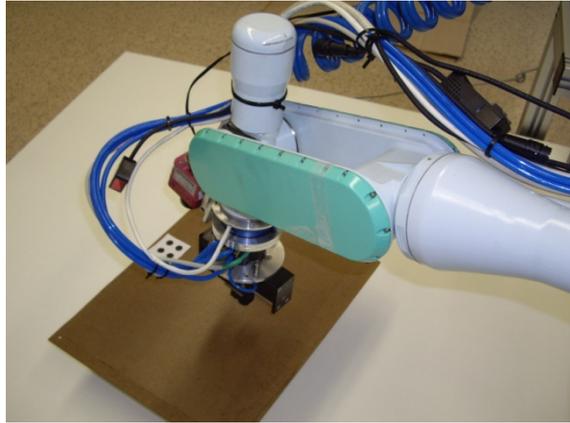


Figura 6-24. Montaje experimental.

#### 6.3.3.2.1 Experimento 1

En este experimento el robot lleva a cabo el seguimiento de una trayectoria usando el sistema descrito en el Capítulo 4. Sin embargo, durante la trayectoria el robot golpea una mesa tal y como ilustra la Figura 6-24. Usando el algoritmo descrito en el apartado anterior, el sistema determina el valor de la matriz de interacción fuerza-imagen, obteniendo la matriz representada en la Ecuación (6.60) ( $\lambda = 0.95$ ). Finalmente, considerando como fuerza de contacto deseada en la dirección z 20 N y la estrategia de control visual-fuerza propuesta, la evolución de las fuerzas de interacción se representa en la Figura 6-25. Esta figura muestra el correcto comportamiento del sistema.

$$\hat{\mathbf{L}} = \begin{bmatrix} 0.6079 & -0.6347 & -0.0469 \\ -0.6079 & 0.6347 & 0.0469 \\ -0.6079 & 0.6347 & 0.0469 \\ -0.6079 & 0.6347 & 0.0469 \\ -0.6079 & 0.6347 & 0.0469 \\ 0.6079 & -0.6347 & -0.0469 \\ 0.6079 & -0.6347 & -0.0469 \\ 0.6079 & -0.6347 & -0.0469 \end{bmatrix} \quad (6.60)$$

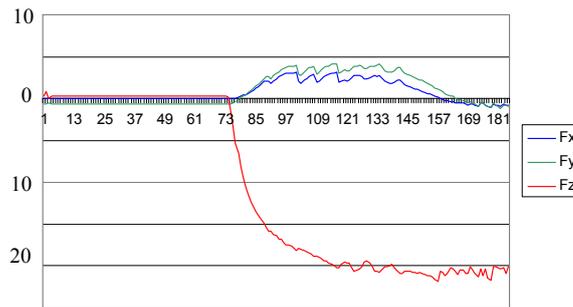


Figura 6-25. Evolución de las fuerzas de interacción durante el experimento. Experimento 1.

### 6.3.3.2.2 Experimento 2

En este experimento el robot también está siguiendo una trayectoria en la imagen usando el controlador descrito en el Capítulo 4. Sin embargo, en este caso durante la interacción el robot debe mantener una fuerza constante de 40 N en la dirección z contra la superficie de contacto.

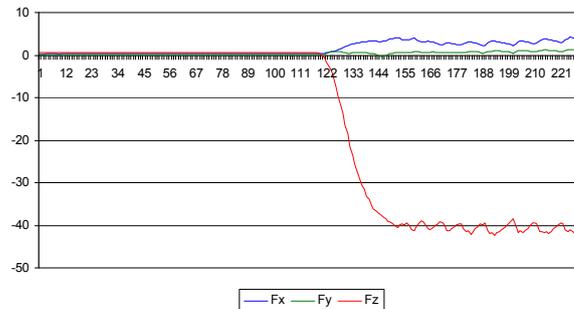


Figura 6-26. Evolución de las fuerzas de interacción durante el experimento. Experimento 2.

Se usa la misma superficie de interacción, por eso se vuelve a considerar la matriz de interacción fuerza-imagen indicada en la Ecuación (6.60) Con estas consideraciones, se obtienen las fuerzas de interacción representadas en la Figura 6-26 cuando el robot golpea la superficie de contacto. En esta figura es posible observar que el sistema es capaz de mantener una fuerza constante de 40 N entre el robot y la superficie en la dirección z. Por lo tanto, la información visual se emplea no sólo para controlar el seguimiento de la trayectoria, sino también para controlar la interacción del robot con el entorno del trabajo.

## 6.4. Conclusiones

Las diferentes propiedades de la información obtenida de los sensores visual y fuerza, pueden provocar que, si no se toman las medidas oportunas, las acciones de control procedentes de ambos sensores puedan llegar a ser contradictorias. Así, el sistema de visión puede establecer una trayectoria

que no sea posible en base a la información de interacción procedente del sensor de fuerza. En este Capítulo se han descrito dos nuevos métodos para fusionar la información sensorial procedente de un sistema de visión por computador y de un sensor de fuerza.

El primer método descrito hace uso del control de impedancia para permitir al robot interactuar con su entorno. La fusión de esta información de fuerza con la visual se consigue mediante la realimentación visual a través del controlador descrito en el Capítulo 4 que permite seguir trayectorias en la imagen de una manera intemporal. Cuando se detecta una interacción con el entorno, se modifica la trayectoria deseada en la imagen para que el método de seguimiento de trayectorias en la imagen basado en control visual virtual tenga en cuenta esta interacción. El método se ha aplicado a un proceso de inserción que requiere gran precisión. En esta aplicación, se realiza un seguimiento de una trayectoria deseada en la imagen. El sistema de seguimiento ha mostrado un correcto comportamiento tanto en el espacio imagen como en el espacio 3D. La trayectoria deseada en la imagen se modifica para permitir al sistema alcanzar la posición deseada cuando se detecta una colisión.

El método descrito necesita recalcular la trayectoria en la imagen cada vez que la superficie de contacto se modifica. Debido a ello, se ha propuesto un método robusto de detección de cambios en las superficies basado en las fuerzas de interacción y en luz estructurada. Primero, se ha probado que únicamente con la información de las fuerzas de interacción no se consigue detectar las discontinuidades en las superficies. Esta aproximación es sensible a fricciones o incluso el alto nivel de ruido de la señal del sensor de fuerza. Para mejorar las detecciones se ha incluido información sensorial adicional a partir de un laser. Usando luz estructurada, el sistema es capaz de automáticamente detectar las zonas en las que puede haber una discontinuidad. De esta forma, únicamente se consideran correctas las discontinuidades determinadas por el sensor de fuerza que están cerca de las determinadas usando luz estructurada. Se ha probado que el sistema de detección es robusto incluso en situaciones en las que la superficie tiene discontinuidades muy suaves.

El segundo método propuesto de fusión de control visual-fuerza está basado en la estimación de la matriz de interacción fuerza-imagen. Este método no requiere conocimiento a priori del modelo cinemático del robot, la calibración de la cámara o de calibración sensorial entre la cámara y el sensor de fuerza. La matriz de interacción fuerza-imagen relaciona los cambios en el espacio imagen con cambios en las fuerzas de interacción. Para estimar el valor de esta matriz se aplica un método basado en el algoritmo de Gauss-Newton.



# Capítulo 7

## Conclusiones

## 7.1. Conclusiones

---

A lo largo de la memoria de investigación se han desarrollado distintas aportaciones dentro del campo del guiado de robots empleando información visual, centrándose los desarrollos fundamentalmente en sistemas de seguimiento de trayectorias y su aplicación a tareas de manipulación en las que se requiere un control combinado de la información visual y la de fuerza.

El primer aspecto que cabría reseñar es la propuesta de un nuevo sistema de seguimiento de trayectorias en el espacio imagen. Este sistema aporta una serie de mejoras respecto a los métodos de seguimiento en imagen propuestos hasta la fecha. En primer lugar, presenta un comportamiento intemporal. Esta propiedad es vital en la posterior fusión del sistema con información de fuerza para permitir obtener un controlador capaz de interactuar con el entorno. En segundo lugar, se trata de un sistema que es capaz de mantener una velocidad constante durante el seguimiento de la trayectoria. Los métodos intemporales definidos hasta ahora no permiten ajustar de manera constante esta velocidad, lo que hace que no sean válidos en aplicaciones en las que sea crucial mantener esta velocidad constante para abordar con éxito la tarea. Gracias a esta última característica, se consigue obtener un resultado en imagen y en 3D mucho menos oscilante que el obtenido con sistemas intemporales previos. Además, el método propuesto permite realizar el seguimiento a partir del conjunto de características visuales, asegurando en todo momento que se tiene una posición 3D válida del robot durante la tarea de seguimiento.

Para conseguir esto último se propone el uso de control visual virtual para calcular la posición 3D de la cámara respecto del objeto, es decir, los parámetros extrínsecos. Sin embargo, empleando control visual virtual con una única imagen no es posible obtener precisión en la calibración de los parámetros intrínsecos. Por ello, se ha propuesto un método llamado calibración multi-imagen que permite a partir de más de una imagen obtener de forma precisa los parámetros intrínsecos de la cámara. De esta forma, se han ampliado las características del método de seguimiento propuesto aportándole la posibilidad de obtener online el valor de los parámetros intrínsecos necesarios en el cálculo del Jacobiano de la imagen que se utiliza en la ley de control.

Los sistemas de control visual-fuerza hacen posible la interacción con el entorno mientras se posiciona el robot siguiendo una trayectoria predefinida. La mayoría de los desarrollos existentes hasta la actualidad para la combinación de ambas informaciones sensoriales emplean estrategias híbridas en las que el entorno es perfectamente conocido y se realiza una división del espacio de trabajo en direcciones controladas por visión y direcciones controladas por fuerza. Las estrategias control visual-fuerza desarrolladas presentan un marco general para su utilización en tareas de manipulación dentro del cual se han desarrollado distintas aportaciones para su aplicación a entornos no estructurados. La

última de las propuestas de fusión de control visual-fuerza hace uso de la matriz que relaciona directamente las variaciones de las características en el espacio imagen con las variaciones de las fuerzas de interacción. Con la estimación de esta matriz se consigue un control visual-fuerza que no requiere conocimiento previo ni de la cinemática del robot, ni de la calibración de la cámara, ni de la calibración previa entre el sensor de fuerza y la cámara.

Así, se ha llegado a un sistema final capaz de interactuar con objetos del entorno, pudiendo manipularlos o evitar dichos obstáculos empleando información visual y procedente del sensor de fuerza, con un correcto comportamiento en el espacio 3D.

## 7.2. Publicaciones

---

Seguidamente se citan las distintas publicaciones generadas a lo largo de estos dos años de investigación, desglosadas entre revistas impactadas, congresos internacionales y congresos nacionales.

### 7.2.1. Revistas

- **“A Robust Approach to Control Robot Manipulators by Fusing Visual and Force Information”**. Pomares, J.; García, G.J.; Torres, F. *Journal of Intelligent and Robotic Systems*. Vol. 48. Num. 4. Pp. 437-456. 2007.

El artículo muestra un sistema de control visual-fuerza para controlar el movimiento de un robot. La trayectoria deseada en la imagen se modifica a partir de la información del sensor de fuerza utilizando la matriz de homografía. Además, este método es robusto ante errores en la estimación de los parámetros intrínsecos.

- **“Adaptive visual servoing and force control fusion to track surfaces”**. Pomares, J.; García, G.J.; Payá, L.; Torres, F. Ed. *WSEAS Transactions on Systems*. Vol. 5. Num. 1. pp. 25-32. 2006.

En este artículo se presenta un método de fusión de control visual-fuerza. El controlador visual propuesto presenta un comportamiento intemporal, y además es autocalibrado, de forma que es capaz de obtener los parámetros intrínsecos durante la tarea. Su fusión con un control de impedancia permite obtener el control visual-fuerza propuesto.

- **“An uncalibrated approach to track trajectories using visual–force control”**. Pomares, J.; García, G.J.; Payá, L.; Torres, F. *Informatics in Control, Automation and Control II*. Ed. Springer. 2006.

El control de impedancia permite controlar las fuerzas de interacción mientras se utiliza un controlador de posición. En el presente artículo se realimenta con información visual al reemplazar el controlador de posición por un control visual basado en flujo de movimiento.

- **“Image motion estimator to track trajectories specified with respect to moving objects”**. Pomares, J.; García, G.J.; Payá, L.; Torres, F. *Informatics in Control, Automation and Robotics III*. Ed. Springer. 2008. (Pendiente de publicación).

En tareas de control visual en las que se debe seguir un objeto en movimiento se necesita estimar este movimiento del objeto de alguna forma para evitar que el error se mantenga durante todo el seguimiento. Para estimar el movimiento del objeto, es necesario calcular la velocidad de la cámara en la iteración anterior de la tarea de control visual. En este artículo se presenta una técnica para calcular esta velocidad.

- **“Improving detection of surface discontinuities in visual-force control systems”**. Pomares, J.; Gil, P.; García, G.J.; Torres, F. *International Journal of Computer Vision*. Ed. Springer. (Artículo enviado en proceso de revisión).

En este artículo se describe un método que fusiona información de fuerza y de luz estructurada para detectar posibles cambios en la superficie con la que el robot está interactuando. Mediante patrones de luz se consigue una línea que marca un posible cambio en la superficie. El movimiento virtual de la posible línea de discontinuidad aportada por la luz estructurada permite afinar el momento en el que activar el detector basado en fuerza. De esta forma se evitan las detecciones falsas.

## 7.2.2. Congresos Internacionales

- **“A New Non-Time Dependent Image Path Tracker to Guide Robots Using Visual Servoing”**. García, G.J.; Pomares, J.; Torres, F. *12th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2007)*. Patras, Grecia. 2007.

El método de seguimiento intemporal de trayectorias en la imagen basado en control visual virtual se presenta en este artículo. Se compara con un método previo de seguimiento intemporal de trayectorias en la imagen (el control visual basado en flujo de movimiento), remarcando las mejoras del método propuesto.

- **“Robot guidance by estimating the force-image interaction matrix”**. García, G.J.; Pomares, J.; Torres, F. *IFAC International Workshop on Intelligent Manufacturing Systems 2007*. Alicante, España. 2007.

Este artículo presenta un método de fusión de control visual-fuerza que utiliza la matriz de interacción fuerza-imagen para controlar a un robot en tareas donde haya interacción con el entorno. La matriz de interacción se estima a partir de un método basado en el algoritmo de Gauss-Newton.

- **“Visual–Force Control and Structured Light Fusion to Improve Recognition of Discontinuities in Surfaces”**. Pomares, J.; Gil, P.; García, G.J.; Torres, F. 11th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2006). Praga (República Checa). 2006.

Este artículo describe un método robusto de detección de cambios en las superficies a partir de la información obtenida del sensor de fuerza y de luz estructurada. Los fallos en la detección de estos cambios usando únicamente información de fuerza se consiguen reparar con la información obtenida a partir de la proyección sobre el objeto de haces de luz de un laser.

- **“Multi-sensorial system for the generation of disassembly trajectories”**. Pomares, J.; Puente, S.T.; García, G.J.; Torres, F. World Automation Congress (WAC 2006). Budapest (Hungría). 2006.

En este artículo se presenta un método para generar trayectorias para el desensamblado automático de distintos tipos de artículos. El sistema utiliza información visual y de un sensor de fuerza para evitar los factores desconocidos en el plan de desensamblado y corregir las trayectorias generadas que se basan en el modelo del producto.

- **“Improving tracking trajectories with motion estimation”**. Pomares, J.; García, G.J.; Torres, F. 3rd International Conference on Informatics in Control, Automation and Robotics (ICINCO 2006). Setubal, Portugal. 2006.

El cálculo de la velocidad de la cámara durante una tarea de control visual permite mejorar el comportamiento de los algoritmos de control visual aplicados al seguimiento de objetos en movimiento. En este artículo se hace uso de la matriz de homografía 2D para calcular la velocidad de la cámara a partir de la imagen previa y la actual.

- **“Impedance control for fusing multisensorial systems in robotic manipulation tasks”**. Pomares, J.; García, G.J.; Torres, F. Proc. De la WSEAS International Conference on Dynamical Systems and Control. 2005.

En este artículo se presenta un control de impedancia para guiar un robot en entornos donde hay interacción. Este control de impedancia se realimenta con información visual al reemplazar el controlador de posición por un control visual basado en flujo de movimiento. De esta forma se puede realizar el seguimiento de una trayectoria mientras se tiene en cuenta las fuerzas de interacción con el entorno.

### 7.2.3. Congresos Nacionales

- **“Sistema de simulación y evaluación de tareas de control visual basado en control visual virtual”**. García, G.J.; Payá, L.; Pomares, J.; Torres, F. XXVII Jornadas de Automática. Almería. 2006.

ViSeC es una herramienta presentada en un artículo previo. En este artículo se amplían las capacidades de esta herramienta aplicando control visual virtual para conseguir mover la cámara virtual del simulador de ViSeC y, de esta forma, realizar un control visual simulado. El control visual virtual permite obtener los parámetros extrínsecos de la cámara durante el control visual simulado.

- **“Control visual flexible empleando momentos invariantes”**. García, G.J.; Pomares, J.; Torres, F. XXVI Jornadas de Automática. Alicante. 2005.

El artículo presenta la formulación necesaria para realizar el control visual con momentos invariantes. Los momentos invariantes permiten realizar el control visual con objetos en los que no es sencillo obtener puntos característicos.

- **“Control visual de robots manipuladores. Una herramienta para su diseño y aprendizaje”**. García, G.J.; Pomares, J.; Torres, F. Proc. de las XXV Jornadas de Automática. 2004.

En este artículo se presentó ViSeC, una herramienta que permite realizar tareas de control visual. ViSeC permite obtener todo tipo de gráficas de la tarea de control realizada. El resultado es una herramienta flexible, fácil de manejar y perfecta para el diseño e implementación de nuevas estrategias de control visual.

## 7.3. Trabajos Futuros

---

Durante estos dos años de trabajo se han abierto ciertas líneas de investigación en el campo del guiado de robots empleando información visual, dentro de las cuales cabría destacar la posibilidad de emplear características más realistas que puedan extraerse de un objeto. Así, los algoritmos propuestos emplean 4 puntos dispuestos en un patrón. Sin embargo, de cara a su implementación en entornos reales sería interesante la extensión de los algoritmos propuestos para permitir que estas características sean, por ejemplo, líneas, círculos, elipses, texturas o color.

La matriz de interacción fuerza-imagen ha sido probada hasta el momento en muy pocas superficies. El siguiente paso es aplicar el método propuesto para estimar la matriz de interacción fuerza-imagen en distintas superficies con distintas características.

En cuanto a la calibración multi-imagen, se ha podido observar en los resultados obtenidos que el número de imágenes para realizar la calibración, lo que se ha venido a llamar el tamaño de la ventana de calibración, varía dependiendo de muchos factores para obtener el resultado óptimo. Sería interesante determinar automáticamente este tamaño de ventana para obtener un sistema que de forma automática se ajuste en cada momento para conseguir los mejores resultados.

También se pretende integrar el control visual-fuerza a control compartido e integrar todos los desarrollos.

Por último, un aspecto muy importante es el de garantizar la robustez del sistema de control visual-fuerza ante incertidumbre en los parámetros del sistema y ante interacción con objetos desconocidos, así como demostrar su estabilidad.



# Anexo 1

## Cuaternios

## A1.1. Definición

Los cuaternios son una forma de representar giros y orientaciones entre dos sistemas de referencia. Un cuaternio  $Q$  está constituido por cuatro componentes  $(q_0, q_1, q_2, q_3)$  que representan las coordenadas del cuaternio en una base  $\{\mathbf{w}, \mathbf{i}, \mathbf{j}, \mathbf{k}\}$ . Normalmente, se denomina parte escalar a la componente en  $\mathbf{w}$ , y parte vectorial al resto. Así, un cuaternio se puede representar como:

$$Q = q_0 \mathbf{w} + q_1 \mathbf{i} + q_2 \mathbf{j} + q_3 \mathbf{k} = [\mathbf{c}, \mathbf{b}] \quad (\text{A1.1})$$

donde  $\mathbf{c}$  representa la parte escalar y  $\mathbf{b}$  representa la parte vectorial.

Para entender qué es un cuaternio se asocia el giro de un ángulo  $\theta$  sobre el vector  $\mathbf{h}$  al cuaternio definido por:

$$Q = \text{Rot}(\mathbf{h}, \theta) = \left( \cos \frac{\theta}{2}, \mathbf{h} \sin \frac{\theta}{2} \right) \quad (\text{A1.2})$$

## A1.2. Transformaciones básicas

Tal como se define en [Beggs-1983] la matriz de rotación de una matriz homogénea se puede obtener a partir de un cuaternio de la siguiente manera:

$$\mathbf{R} = \begin{pmatrix} 2q_0^2 + 2q_1^2 - 1 & 2q_1q_2 - 2q_3q_0 & 2q_1q_3 + 2q_2q_0 \\ 2q_1q_2 + 2q_3q_0 & 2q_0^2 + 2q_2^2 - 1 & 2q_2q_3 - 2q_1q_0 \\ 2q_1q_3 - 2q_2q_0 & 2q_2q_3 + 2q_1q_0 & 2q_0^2 + 2q_3^2 - 1 \end{pmatrix} \quad (\text{A1.3})$$

La matriz de rotación  $\mathbf{R}$  queda definida mediante sus elementos de la siguiente manera:

$$\mathbf{R} = \begin{pmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{pmatrix} \quad (\text{A1.4})$$

A partir de esta matriz de rotación, para calcular el cuaternio asociado se utiliza la relación indicada en [Barrientos-1997]:

$$\begin{aligned}
q_0 &= \frac{1}{2} \sqrt{(n_x + o_y + a_z + 1)} \\
q_1 &= \frac{1}{2} \sqrt{(n_x - o_y - a_z + 1)} \\
q_2 &= \frac{1}{2} \sqrt{(-n_x + o_y - a_z + 1)} \\
q_3 &= \frac{1}{2} \sqrt{(-n_x - o_y + a_z + 1)}
\end{aligned} \tag{A1.5}$$

### A1.3. Interpolación de la Orientación

Como es conocido, la especificación de la posición por parte del usuario se realiza habitualmente, y salvo escasas excepciones, en coordenadas cartesianas. Sin embargo, la especificación de la orientación puede realizarse mediante diferentes herramientas, como: matrices de rotación, ángulos de Euler, par de rotación o cuaternios.

Para la posición, la interpolación lineal es la evolución lineal, a velocidad constante, de cada coordenada cartesiana desde su valor inicial hasta el final, resultando un movimiento sencillo, fácilmente interpretable por el usuario y de calidad.

Sin embargo, para el caso de la orientación, esta evolución lineal desde el valor inicial hasta el final puede ser planteado en términos de matrices de rotación, ángulos de Euler, par de rotación o cuaternios, resultando en cada caso trayectorias diferentes.

La utilización de las matrices de rotación lleva a resultados inconsistentes. Como es sabido, las matrices de rotación deben ser necesariamente matrices ortonormales. La interpolación lineal entre una matriz de rotación inicial y otra final lleva a matrices intermedias no ortonormales y, por lo tanto, que no corresponden a matrices de rotación.

La utilización de ángulos de Euler, en cualquiera de sus formas, además de ser la representación más compacta no presenta este inconveniente. Así, para pasar de una orientación inicial  $(\alpha_i, \beta_i, \gamma_i)$  a una final  $(\alpha_f, \beta_f, \gamma_f)$  se podrían utilizar las funciones lineales:

$$\begin{aligned}
\alpha(t) &= (\alpha_f - \alpha_i) \frac{t - t_i}{t_f - t_i} + \alpha_i \\
\beta(t) &= (\beta_f - \beta_i) \frac{t - t_i}{t_f - t_i} + \beta_i \\
\gamma(t) &= (\gamma_f - \gamma_i) \frac{t - t_i}{t_f - t_i} + \gamma_i
\end{aligned} \tag{A1.6}$$

Donde  $t_i$  y  $t_f$  son los instantes de tiempo en los que se pretende estar en la orientación inicial y final respectivamente. El inconveniente de esta trayectoria es que desde el punto de vista del usuario es una trayectoria no intuitiva, con extrañas evoluciones de la orientación.

La evolución más natural desde una orientación inicial hasta otra final, sería aquella que hace girar de manera progresiva al efector final (u objeto manipulado por el robot) desde su orientación inicial hasta la final en torno a un eje de giro fijo. Por este motivo, la utilización del par de rotación, o su equivalente, los cuaternios, es la manera más adecuada para generar la trayectoria cartesiana de orientación. La Figura A1-1 muestra la esfera del espacio de cuaternios, indicando un esquema de la interpolación entre dos cuaternios:  $Q_a$  a  $Q_b$ .

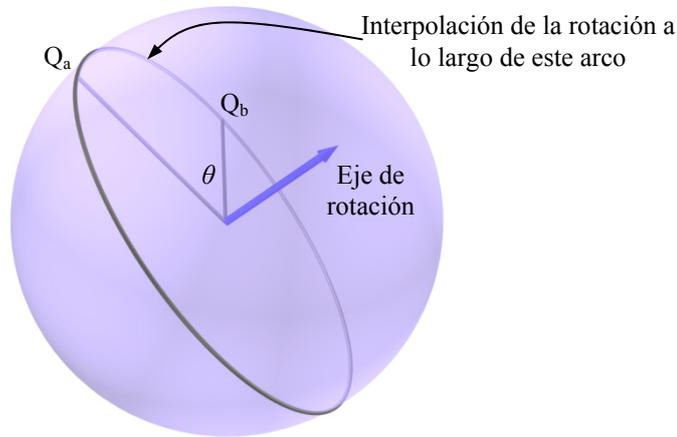


Figura A1-1. Interpolación de cuaternios.

Si  $Q$  es el cuaternio que rota desde  $Q_a$  a  $Q_b$ , entonces:

$$Q_b = Q_a \cdot Q \tag{A1.7}$$

Multiplicando ambos lados por el conjugado de  $Q_a$  permite obtener:

$$Q = \text{conj}(Q_a) \cdot Q_b \tag{A1.8}$$

La parte real de la multiplicación es:

$$\begin{aligned} \text{real}((Q_a \cdot w + iQ_a \cdot x + jQ_a \cdot y + kQ_a \cdot z) \cdot (Q_b \cdot w + iQ_b \cdot x + jQ_b \cdot y + kQ_b \cdot z)) = \\ = Q_a \cdot w \cdot Q_b \cdot w - Q_a \cdot x \cdot Q_b \cdot x - Q_a \cdot y \cdot Q_b \cdot y - Q_a \cdot z \cdot Q_b \cdot z \end{aligned} \tag{A1.9}$$

Entonces, usando el conjugado de  $Q_a$  proporciona:

$$\begin{aligned} \text{real}((Q_a \cdot w - iQ_a \cdot x - jQ_a \cdot y - kQ_a \cdot z) \cdot (Q_b \cdot w + iQ_b \cdot x + jQ_b \cdot y + kQ_b \cdot z)) = \\ = Q_a \cdot w \cdot Q_b \cdot w + Q_a \cdot x \cdot Q_b \cdot x + Q_a \cdot y \cdot Q_b \cdot y + Q_a \cdot z \cdot Q_b \cdot z \end{aligned} \tag{A1.10}$$

Como:

$$\text{real}(Q) = \cos\left(\frac{t}{2}\right) \tag{A1.11}$$

Por lo tanto:

$$\cos\left(\frac{\theta}{2}\right) = Q_a \cdot w \cdot Q_b \cdot w + Q_a \cdot x \cdot Q_b \cdot x + Q_a \cdot y \cdot Q_b \cdot y + Q_a \cdot z \cdot Q_b \cdot z \quad (\text{A1.12})$$

Como puede apreciarse, tiene el aspecto de un producto escalar de vectores. Para obtener la ecuación que permite calcular el cuaternio interpolado se usa la siguiente ecuación:

$$Q_m = \frac{Q_a \cdot \text{sen}((1-t)\theta) + Q_b \cdot \text{sen}(t\theta)}{\text{sen}\theta} \quad (\text{A1.13})$$

donde:

- $Q_m$  es el cuaternio interpolado.
- $Q_a$  es el primer cuaternio en la trayectoria de interpolación.
- $Q_b$  es el último cuaternio en la trayectoria de interpolación.
- $T$  es un escalar entre 0.0 (en  $Q_a$ ) y 1.0 (en  $Q_b$ ).
- $\theta$  es la mitad del ángulo entre  $Q_a$  y  $Q_b$ .

Para convertir de  $\cos(\theta)$  a  $\text{sen}(\theta)$  la mejor opción es usar:

$$\text{sen}(\theta) = \text{sen}(\arccos(\cos(\theta))) \quad (\text{A1.14})$$

O también:

$$\text{sen}(\theta) = \text{sqrt}(1 - \cos^2(\theta)) \quad (\text{A1.15})$$



Anexo 2

# Método de optimización de Levenberg - Marquardt

## A2.1. Introducción

El algoritmo de Levenberg-Marquardt es una técnica iterativa que obtiene el mínimo de una función que se expresa como la suma de los cuadrados de funciones reales no lineales [Levenberg - 1944, Marquardt-1963]. Se ha convertido en una técnica estándar para resolver problemas de mínimos cuadrados no lineales, ampliamente adoptados en un amplio espectro de disciplinas. El algoritmo de Levenberg-Marquardt puede entenderse como una interpolación entre el descenso por gradiente y el método de Gauss-Newton. Cuando la solución actual está muy alejada de la correcta, el algoritmo se comporta como un método de descenso por gradiente: lento, pero garantizando la convergencia. Cuando la solución actual se acerca a la correcta, se convierte en un método de Gauss-Newton. A continuación se presenta una breve descripción del algoritmo de Levenberg-Marquardt.

## A2.2. El Algoritmo de Levenberg-Marquardt

Considérese  $f$  una relación funcional que relaciona un vector de parámetros  $\mathbf{p} \in \mathfrak{R}^m$  con un vector de medida estimado  $\hat{x} = f(\mathbf{p})$ ,  $x \in \mathfrak{R}^n$ . Inicialmente se dispone de una estimación inicial del parámetro  $\mathbf{p}_0$  y un vector de medida  $x$  y se desea encontrar el vector que mejor satisface la relación funcional  $f$ , es decir minimizar la distancia al cuadrado  $\varepsilon^T \varepsilon$  con  $\varepsilon = x - \hat{x}$ . La base del algoritmo de Levenberg-Marquardt es una aproximación lineal a  $f$  en la vecindad de  $\mathbf{p}$ . Para una pequeña variación de  $\mathbf{p}$ ,  $\|\delta_p\|$ , la expansión de la serie de Taylor lleva a la aproximación:

$$f(\mathbf{p} + \delta_p) \approx f(\mathbf{p}) + \mathbf{J} \delta_p \quad (\text{A1.1})$$

donde  $\mathbf{J}$  es la matriz Jacobiana  $\frac{\partial f(\mathbf{p})}{\partial \mathbf{p}}$ . Como todos los métodos de optimización no lineal, el algoritmo de Levenberg-Marquardt es iterativo: iniciado en el punto inicial  $\mathbf{p}_0$ , el método produce una serie de vectores  $\mathbf{p}_1, \mathbf{p}_2, \dots$ , que convergen hacia un minimizador local  $\mathbf{p}^+$  para  $f$ . Así, en cada paso, se requiere encontrar el  $\delta_p$  que minimiza la cantidad  $\|x - f(\mathbf{p} + \delta_p)\| \approx \|x - f(\mathbf{p}) - \mathbf{J} \delta_p\| = \|\varepsilon - \mathbf{J} \delta_p\|$ . El  $\delta_p$  buscado es por tanto la solución al problema de mínimos cuadrados lineal. Se logra el mínimo cuando  $\mathbf{J} \delta_p - \varepsilon$  es ortogonal a las columnas de  $\mathbf{J}$ . Esto lleva a  $\mathbf{J}^T (\mathbf{J} \delta_p - \varepsilon) = 0$ , lo que permite obtener  $\delta_p$  como la solución de las llamadas ecuaciones normales:

$$\mathbf{J}^T \mathbf{J} \delta_p = \mathbf{J}^T \varepsilon \quad (\text{A2.2})$$

La matriz  $\mathbf{J}^T \mathbf{J}$  en el lado izquierdo de la Ecuación (A2.2) es la Hessiana aproximada, es decir, una estimación de la matriz de las derivadas de segundo orden. El método de Levenberg-Marquardt realmente soluciona una ligera variación de la Ecuación (A2.2), conocida como las ecuaciones normales aumentadas:

$$\mathbf{N} \delta_p = \mathbf{J}^T \boldsymbol{\varepsilon} \quad (\text{A2.3})$$

donde los elementos no diagonales de la matriz  $\mathbf{N}$  son idénticos a los elementos correspondientes de  $\mathbf{J}^T \mathbf{J}$  y los elementos diagonales se obtienen con  $\mathbf{N}_{ii} = \mu + [\mathbf{J}^T \mathbf{J}]_{ii}$  para valores de  $\mu > 0$ . La estrategia de modificar los elementos diagonales de  $\mathbf{J}^T \mathbf{J}$  se llama *damping* y al término  $\mu$  se le llama *término de damping*. Si el vector actualizado  $\mathbf{p} + \delta_p$  con  $\delta_p$  calculado a partir de la Ecuación (A2.3) conduce a una reducción del error  $\varepsilon$ , la actualización se acepta y el proceso se repite con un término de damping menor. En caso contrario, se incrementa el término de damping, se resuelven las ecuaciones normales aumentadas de nuevo y el proceso itera hasta que se encuentre un valor de  $\delta_p$  que disminuya el error. El proceso de resolver repetidamente la Ecuación (A2.3) para distintos valores del término de damping hasta que se encuentra una actualización aceptable del vector de parámetro se corresponde con una iteración del algoritmo de Levenberg-Marquardt. En el algoritmo de Levenberg-Marquardt, el término de damping se ajusta en cada iteración para asegurar una reducción en el error  $\varepsilon$ . Si el damping se establece como un valor muy alto, la matriz  $\mathbf{N}$  de la Ecuación (A2.3) es prácticamente diagonal y el paso de actualización del algoritmo de Levenberg-Marquardt es muy cercano a la dirección del descenso por gradiente. Además, el valor de  $\delta_p$  se reduce en este caso. El damping también maneja situaciones donde el Jacobiano tiene rango deficiente y  $\mathbf{J}^T \mathbf{J}$  es por lo tanto singular. En este caso, el algoritmo de Levenberg-Marquardt puede buscar efectivamente en una región del espacio de parámetros en el que el modelo es altamente no lineal. El algoritmo de Levenberg-Marquardt es adaptativo porque controla su propio damping: sube el damping si un paso falla en la reducción de  $\varepsilon$ ; en caso contrario reduce el damping. De esta forma el algoritmo es capaz de alternar entre una aproximación de descenso lento cuando se encuentra lejos del mínimo, y una rápida convergencia cuando se encuentra en un entorno cercano del mínimo. El algoritmo de Levenberg-Marquardt termina cuando al menos se cumple una de las siguientes condiciones:

- El valor del gradiente de  $\boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon}$ , es decir,  $\mathbf{J}^T \boldsymbol{\varepsilon}$  en el lado derecho de la Ecuación (A2.2), se sitúa por debajo de un umbral  $\varepsilon_1$ .
- El cambio relativo en el valor de  $\delta_p$  disminuye por debajo de un umbral  $\varepsilon_2$ .
- El error  $\boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon}$  disminuye por debajo de un umbral  $\varepsilon_3$ .
- Se completa un número máximo de iteraciones  $k_{\max}$ .

Si está disponible una matriz de covarianza  $\Sigma_x$  para el vector de medida, se puede incorporar en el algoritmo de Levenberg-Marquardt minimizando la norma al cuadrado  $\boldsymbol{\varepsilon}^T \Sigma_x^{-1} \boldsymbol{\varepsilon}$  en lugar de la Euclídea  $\boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon}$ . Por consiguiente, el mínimo se encuentra resolviendo un problema de mínimos cuadrados ponderados definido por las ecuaciones normales ponderadas:

$$\mathbf{J}^T \Sigma_x^{-1} \mathbf{J} \boldsymbol{\delta}_p = \mathbf{J}^T \Sigma_x^{-1} \boldsymbol{\varepsilon} \quad (\text{A2.4})$$

El resto del algoritmo permanece sin cambios.

# Referencias

- [Abolmaesumi-2002] Abolmaesumi, P.; Salcudean, S. E.; Zhu, W.H.; Sirouspour, M.R.; DiMaio, S.P. "Image Guided Control of a Robot for Medical Ultrasound," IEEE Transactions on Robotics and Automation. Vol. 18. No. 1. Pp. 11-23. 2002
- [Allen-1992] Allen, P. K.; Timcenko, A.; Yoshimi, B.; Michelman, P. "Real-time visual servoing". IEEE International Conference Robotics and Automation. Pp. 1850-1856. 1992.
- [Angel-2004] Angel, L.; Sebastian, J.M.; Saltaren, R.; Aracil, R. "RoboTennis: Parallel robot with visual control". Robotics: Trends, Principles, and Applications - Proceedings of the Sixth Biannual World Automation Congress, WAC, pp. 405-412. 2004.
- [Baeten-2002a] Baeten, J.; Bruyninckx, H.; De Schutter, J. "Shared Control in Hybrid Vision/Force robotic Servoing using the Task Frame", Proceedings of the 2002 ". IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'2002). Lausanne, Suiza. Pp. 2128-2133. 2002.
- [Baeten-2002b] Baeten, J.; De Schutter, J. "Hybrid Vision/Force Control at Corners in Planar Robotic-Contour Following" IEEE/ASME Transactions on Mechatronics. Vol. 7. No 2. Pp. 143 – 151. 2002.
- [Baeten-2003] Baeten, J.; Bruyninckx, H, De Schutter, J. "Integrated vision/force robotic servoing in the task frame formalism". International Journal of Robotics Research. Vol. 22. No. 10-11. Pp. 941-954. 2003.
- [Barrientos-1997] Barrientos, A.; Peñin, L.F.; Balaguer, C.; Aracil, R. "Fundamentos de Robótica". McGraw-Hill. 1997.
- [Beggs-1983] Beggs, J.S. "Kinematics". Springer Verlag, Berlin. 1983.
- [Behringer-1995] Behringer, R. "Detection of discontinuities of road curvature by glr methods". Proceedings of International Symposium on Intelligent Vehicles. Detroit, USA. 1995.
- [Blanc-2005] Blanc, G.; Mezouar, Y.; Martinet, P. "Indoor navigation of a wheeled mobile robot along visual routes". *Proceedings - IEEE International Conference on Robotics and Automation*, 2005, art. no. 1570628, pp. 3354-3359. 2005.
- [Bourquardez-2007] Bourquardez, O.; Chaumette, F. "Visual Servoing of an Airplane for Alignment with respect to a Runway". In IEEE Int. Conf. on Robotics and Automation, ICRA'07, Roma, Italia, April 2007.
- [Bruyninckx-1996] Bruyninckx, H, De Schutter, J.. "Specification of force-controlled actions in the task frame formalism-A synthesis". IEEE Transactions on Robotics and Automation. Vol. 12. No. 4. Pp. 581-589. 1996.
- [Carelli-2004] Carelli, R.; Oliva, E.; Soria, C.; Nasisi, O. "Combined force and visual control of an industrial robot". *Robotica* 22 (2), pp. 163-171, 2004.

- [Cervera-2003] Cervera, E.; del Pobil, A. P.; Berry, F.; Martinet, P. "Improving image-based visual servoing with three-dimensional features". International Journal of Robotics Research. Vol. 22. No. 10-11. Pp. 821-840. 2003
- [Chaumette-1990] Chaumette, F. "La relation vision-commande: théorie et application à des tâches robotiques" Tesis. Universidad de Rennes. 1990.
- [Chaumette-1997] Chaumette, F.; Malis, E.; Boudet, S. "2d 1/2 visual servoing with respect to a planar object". Workshop on New trends in Image-based robot servoing. Pp. 45-52. 1997.
- [Chaumette-1998] Chaumette, F. "Potential problems of convergence in visual servoing". Int. Symposium on Mathematical Theory of Networks and Systems, (MTNS'98). Padoue, Italia. 1998.
- [Chaumette-2002] Chaumette, F. "A first step toward visual servoing using image moments". IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'2002). Lausanne, Suiza. Pp. 378-383. 2002.
- [Chaumette-2004] Chaumette, F. "Image moments: A general and useful set of features for visual servoing". IEEE Trans. Robotics and Automation, vol. 20, no 4, pp.713-723. Aug. 2004.
- [Chaumette-2006] Chaumette, F.; S. Hutchinson. "Visual Servo Control, Part I: Basic Approaches". IEEE Robotics and Automation Magazine, 13(4):82-90, December 2006.
- [Chaumette-2007] Chaumette, F.; S. Hutchinson. "Visual Servo Control, Part II: Advanced Approaches". IEEE Robotics and Automation Magazine, 14(1), March 2007.
- [Chen-2005] Chen, J.; Dawson, D.; Dixon, W.; Behal, A. "Adaptive homography-based visual servo tracking for fixed and camera-in-hand configurations". IEEE Trans. Control Syst. Technol., vol. 13, no. 9, pp. 814-825, Sept. 2005.
- [Chesi-2005] Chesi, G.; Prattichizzo, D.; Vicino, A. "Visual Servoing: Reaching the Desired Location Following a Straight Line via Polynomial Parameterizations". ICRA 2005. Proceedings of the 2005 IEEE International Conference on Robotics and Automation, 18-22 April 2005 Page(s):2559 - 2564. 2005.
- [Chriette-2001] Chriette, A.; Hamel, T.; Mahony, R. "Visual servoing for a scale model autonomous helicopter". IEEE International Conference on Robotics and Automation, Seoul, Korea. Pp. 21-26. 2001.
- [Collewet-2000] Collewet, C.; Chaumette, F. "A contour approach for image-based control of objects with complex shape" IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'2002). Takamatsu, Japón. Vol. 1. Pp. 751-756. 2000.

- [Collewet-2004] Collewet, C.; Alhaj, A.; Chaumette, F. “Model-free visual servoing on complex images based on 3D reconstruction”. ICRA'04, vol. 1, pages 751-756, New Orleans, LA, April 2004.
- [Corke-1994] Corke, P. I. “Visual control of robot manipulators – a review”. Visual servoing. K. Hashimoto, Ed. Singapore: World Scientific. Pp. 1-31. 1994.
- [Corke-1996a] Corke, P. I. “Visual control of robots” Research studies press ltd. 1996.
- [Corke-2001] Corke, P. I.; Hutchinson, S. A. “A new partitioned approach to image-based visual servo control”. IEEE Transactions on Robotics and Automation. Vol. 17. No. 4. Pp. 507-515. 2001.
- [Cowan-1999] Cowan, N. J.; Koditschek, D. E. “Planar image based visual servoing as a navigation problem”. IEEE International Conference on Robotics and Automation, Detroit, Michigan. Pp. 611–617. 1999.
- [Cowan-2002] Cowan, N. J.; Weingarten, J. D.; Koditschek, D.E. “Visual servoing via navigation functions”. IEEE Transactions on Robotics and Automation. Vol. 18. No. 4. Pp. 521-533. 2002.
- [Crétual-1998] Crétual, A.; Chaumette, F.; Bouthemy, P. “Complex object tracking by visual servoing based on 2D image motion”. Proceedings of the IAPR International Conference on Pattern Recognition. (ICPR'98). Brisbane, Australia. Vol. 2. Pp. 1251-1254. 1998.
- [David-2002] David, P.; DeMenthon, D.; Duraiswami, R.; Samet, H. “SoftPOSIT: Simultaneous Pose and Correspondence Determination”. Proceedings of ECCV 02 European Conference of Computer Vision. Copenhagen. 2002.
- [Dean-León-2005a] Dean-León, E.G.; García, L.G.; Parra-Vega, V.; Espinosa-Romero, A. “Experimental study on image-based position-force adaptive visual servoing for constrained robots under jacobian and dynamic friction uncertainties”. IEEE International Conference on Mechatronics and Automation, ICMA 2005, pp. 7-12, 2005.
- [Dean-León-2005b] Dean-León, E.G.; García, L.G.; Parra-Vega, V.; Espinosa-Romero, A. “Visual servoing for constrained robots: A new complete theoretical framework and its experimental validation”. IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM 2, pp. 1445-1452, 2005.
- [De Schutter-1988] De Schutter, J.; Van Brussel, H. “Compliant robot motion. I. A formalism for specifying compliant motion tasks. II. A control approach based on external control loops”. International Journal of Robotics Research. Vol. 7. No. 4. Pp. 3-33. 1988.

- [De Schutter-1997] De Schutter, J.; Bruyninckx, H.; Spong, M. "Force control : a bird's eye view". IEEE International Workshop on Control Problems in Robotics and Automation : Future Directions. San Diego, CA, 1997.
- [DeMenthon-1995] DeMenthon, D.; Davis, L. S. "Model based object pose in 25 lines of code". International Journal of Computer Vision, Kluwer Academic Publishers. Vol. 12. No 1-2. Pp. 123-141. 1995.
- [Deng-2005] Deng, L.; Janabi-Sharifi, F.; Hassanzadeh, I. "Comparison of combined vision and force control strategies for robot manipulators". Proceedings of SPIE - The International Society for Optical Engineering 6052, art. no. 605202, 2005.
- [Dickmanns-1988] Dickmanns, E. D.; Graefe, V. "Applications of dynamic monocular machine vision". Machine Vision and Applications. Vol 1. Pp. 241-261. 1988.
- [Dornaika-1996] Dornaika, F. "Contributions à l'intégration vision/robotique: calibrage, localisation et asservissement". University of Illinois at Urbana-Champaign, Illinois, USA. 1996.
- [Faugeras-1998] Faugeras, O.; Lustman, F. "Motion and structure from motion in a piecewise planar environment". Int. Journal of Pattern Recognition and Artificial Intelligence, vol. 2, no. 3. pp. 485 – 508. 1998.
- [Gangloff-1998] Gangloff, J.; Mathelin, M.; Abba, G. "DOF High Speed Dynamic Visual Servoing Using GPC Controllers". IEEE International Conference on Robotics and Automation, Leuven, Bélgica. Pp. 2008-2013. 1998.
- [GarcíaGJ-2004] García, G.J.; Pomares, J.; Torres, F. "Control visual de robots manipuladores. Una herramienta para su diseño y aprendizaje". Proc. de las XXV Jornadas de Automática. 2004.
- [GarcíaGJ-2005] García, G.J.; Pomares, J.; Torres, F. "Control visual flexible empleando momentos invariantes". XXVI Jornadas de Automática. Alicante. 2005.
- [GarcíaGJ-2006] García, G.J.; Payá, L.; Pomares, J.; Torres, F. "Sistema de simulación y evaluación de tareas de control visual basado en control visual virtual". XXVII Jornadas de Automática. Almería. 2006.
- [GarcíaGJ-2007a] García, G.J.; Pomares, J.; Torres, F. "Robot guidance by estimating the force-image interaction matrix". IFAC International Workshop on Intelligent Manufacturing Systems 2007. Alicante, Spain. 2007.
- [GarcíaGJ-2007b] García, G.J.; Pomares, J.; Torres, F. "A New Non-Time Dependent Image Path Tracker to Guide Robots Using Visual Servoing". 12th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2007). Patras, Grecia. 2007.

- [GarcíaN-2002] N.García, G. Mamani, O. Reinoso, O. Nasisi, R. Aracil, R. Carelli. “Visual Servo Control of Industrial Robot Manipulator”. IFAC 2002 World Congress. Barcelona. 2002.
- [Gil-2006] Gil, P.; Torres, F.; Reinoso, O. “A detection method of intersections for determining overlapping using active vision”. International Conference on Computer Vision Theory and Applications, VISAPP 2006, vol. 1, pp. 501-507, Setubal, Portugal, 2006.
- [Ginhoux-2005] Ginhoux, R.; Gangloff, J.; de Mathelin, M.; Soler, L.; Sanchez, M. A.; Marescaux, J. “Active Filtering of Physiological Motion in Robotized Surgery Using Predictive Control”. In IEEE Transactions on Robotics, Vol. 21(1):67--79, February, 2005.
- [Grosso-1996] Grosso, E.; Meta, M.; Andrea, A.; Sandini, G. “Robust visual servoing in 3-d reaching tasks”. IEEE Transactions on Robotics and Automation. Vol 12. No. 5. Pp. 732-742. 1996.
- [Harrell-1989] Harrell, R. C.; Slaughter, D. C.; Adsit, P. D. “A fruit-tracking system for robotic harvesting”. Machine Vision and Applications. Pp. 69-80. 1989.
- [Hartley-2000] Hartley, R.; Zisserman, A. “Multiple view geometry in computer vision”. Cambridge University Press. 2000.
- [Hashimoto-1994] Hashimoto, K. (Ed.) “Visual Servoing : Real Time Control of Robot Manipulators Based on Visual Sensory Feedback”. World Scientific Series in Robotics and Automated Systems. Vol. 7. World Singapore: Scientific Press. 1994.
- [Hashimoto-1996] Hashimoto, K.; Ebine, T.; Kimura, H. “Visual servoing with hand-eye manipulator-optimal control approach”. IEEE Transactions on Robotics and Automation. Vol. 12, No. 5. Pp. 766-774. 1996.
- [Hassanzadeh-2005] Hassanzadeh, I.; Janabi-Sharifi, F. “An open architecture for hybrid force-visual servo control of robotic manipulators in unstructured environments”. Proceedings of SPIE - The International Society for Optical Engineering 6052, art. no. 605205, 2005.
- [Hemayed-2003] Hemayed, E. E. “A survey of camera self-calibration,” in Proc. IEEE Conference on Advanced Video and Signal Based Surveillance, Miami, USA, pp. 351 – 357. 2003.
- [Hill-1979] Hill, J.; Park, W. T. “Real time control of a robot with a mobile camera” Proceedings 9th.International Symposium on Industrial Robots. Washington, D.C. Pp. 233-246. 1979.
- [Hogan-1985] Hogan, H. “Impedance control: An Approach to Manipulation: Part I, Part II, Part III”. ASME Journal of Dynamic Systems Measurement Control. Vol. 107. Pp. 1-24. 1985.

- [**Hollinghurst-1993**] **Hollinghurst, N.; Cipolla, R.** “Uncalibrated stereo hand-eye coordination”. Informe técnico número CUED/F-INFENG/TR126. Department of Engineering, University of Cambridge. 1993.
- [**Horaud-1997**] **Horaud, R.; Dornaika, F.; Lamiroy, B.; Christy, S.** “Object pose: The link between weak perspective, paraperspective, and full perspective”. *International Journal of Computer Vision*. Vol. 22, No. 2. Pp. 173-189. 1997.
- [**Hosoda-1994**] **Hosoda, K.; Asada, M.** “Versatile visual servoing without knowledge of true jacobian”. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots Systems*, pp. 186–193. Munchen, Germany, Sept. 1994.
- [**Hosoda-1995**] **Hosoda, K.; Sakamoto, K.; Asada, M.** “Trajectory generation for obstacle avoidance of uncalibrated stereo visual servoing without 3-D reconstruction”. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS’1995)*. Pittsburgh, USA. Vol. 1. No. 3. Pp. 29–34. 1995.
- [**Hosoda-1998**] **Hosoda, K.; Igarashi, K.; Asada, M.** “Adaptive hybrid control for visual and force servoing in an unknown environment”. *IEEE Robotics and Automation Magazine*. Vol. 5. No. 4. Pp. 39–43. 1998.
- [**Hutchinson-1996**] **Hutchinson, S.; Hager, G. D.; Corke, P.I.** “A tutorial on visual servo control”. *IEEE Transactions on Robotics and Automation*, Vol. 12, No. 5. Pp. 651-670, 1996.
- [**Iwatsuki-2005**] **Iwatsuki, M.; Okiyama, N.** “A new formulation of visual servoing based on cylindrical coordinate system”. *IEEE Trans. Rob. Automat.* vol. 21, no 2, pp. 266-273, April 2005.
- [**Jagersand-1997**] **Jagersand, M.; Fuentes, O.; Nelson, R.** “Experimental evaluation of uncalibrated visual servoing for precision manipulation”. *IEEE International Conference on Robotics and Automation*. Albuquerque, USA. Vol. 3. Pp. 2874-2880. 1997.
- [**Janabi-Sharifi-1997**] **Janabi-Sharifi, F.; Wilson, W. J.** “Automatic selection of image features for visual servoing”. *IEEE Transactions on Robotics and Automation*. Vol. 13. No. 6. Pp. 890-914. 1997.
- [**Jörg-2000**] **Jörg, S.; Langwald, J.; Stelter, J.; Hirzinger, G.; Natale, C.** “Flexible robot-assembly using a multi-sensory approach”. *IEEE International Conference on Robotics and Automation*. San Francisco, CA. Pp. 3687-3694. 2000.
- [**Kelly-2000**] **Kelly R.; Carelli, R.; Nasisi, O.; Kuchen, B.; Reyes, J. F.** "Stable visual servoing of camera-in-hand robotic systems". *IEEE/ASME Transactions on Mechatronics*. Vol. 5. No. 1. Pp. 39-48. 2000.

- [Kinoshita-1994] Kinoshita, K.; Deguschi, K. "Simultaneous determination of camera pose and intrinsic parameters by visual servoing", in Proc. 12th International Conference on Pattern Recognition, Jerusalem, Israel, vol. A, 1994, pp. 285-289.
- [Kragic-2001] Kragic, D.; Christensen, H. I. "Cue integration for visual servoing". IEEE Transactions on Robotics and Automation. Vol 17. No. 1. Pp. 18-27. 2001.
- [Kragic-2002] Kragic, D.; Peterson, L.; Christensen, H. I. "Visually guided manipulation tasks". Robotics and Autonomous Systems. Vol. 40, No. 2-3. Pp.193-203. 2002.
- [Kragic-2003] Kragic, D.; Christensen, H. I. "Robust visual servoing". International Journal of Robotics Research. Vol. 22. No. 10-11. Pp. 923-940. 2003.
- [Lawrence-1988] Lawrence, D. "Impedance control stability properties in common implementations". IEEE International Conference on Robotics and Automation, Philadelphia, USA. Pp. 1185-1192. 1988.
- [Levenberg -1944] Levenberg, K. "A Method for the Solution of Certain Non-linear Problems in Least Squares". Quarterly of Applied Mathematics, 2(2):164-168, Jul. 1944.
- [Li-2001] Li, P. Y.; Horowitz, R. "Passive Velocity Field Control (PVFC): Part II – Application to Contour Following". IEEE Transaction on Automatic Control, Vol. 46, No 9. pp. 1360-1371, Sept. 2001.
- [Lippiello-2003] Lippiello, V.; Siciliano, B. Villani, L. "Robust visual tracking using a fixed multi-camera system", IEEE International Conference on Robotics and Automation, Taipei, Taiwan. 2003.
- [Lots-2001] Lots, J. F.; Lane, D. M.; Trucco, E.; Chaumette, F. "A 2-D Visual Servoing for Underwater Vehicle Station Keeping" Proceedings of the 2001 IEEE Conference on Robotics and Automation. Seoul, Korea. Pp. 2767-2772. 2001.
- [Lotufo-2001] Lotufo R.; Zampiroli, F. "Fast multidimensional parallel euclidean distance transform based on mathematical morphology". Proceedings of SIBGRAPI 2001, XIV Brazilian Symposium on Computer Graphics and Image Processing, IEEE Computer Society. Pp. 100-105. 2001.
- [Malis-1999] Malis, E.; Chaumette, F.; Boudet, S. "2D 1/2 visual servoing". IEEE Transaction on Robotics and Automation, Vol. 15. No. 2. Pp. 234-246. 1999.
- [Malis-2001] Malis, E.; Morel, G.; Chaumette, F. "Robot control from disparate multiple sensors". International Journal of Robotics Research. Vol. 20. No. 5. Pp. 364-378. 2001.
- [Malis-2003] Malis, E.; Rives, P. "Uncalibrated Active Affine Reconstruction closing the loop by Visual Servoing". IEEE/RSJ International Conference on Intelligent Robots Systems, (IROS'2003) Las Vegas, Nevada, U.S.A. 2003.

- [Malis-2004a] Malis, E. “Visual servoing invariant to changes in camera-intrinsic parameters”. IEEE Transactions on Robotics and Automation, 20 (1), pp. 72-81. 2004.
- [Malis-2004b] Malis, E. “Improving vision-based control using efficient second-order minimization techniques”. In Proc. IEEE International Conference Robot. Automat., pp. 1843-1848. April 2004.
- [Marchand-1996] Marchand, E.; Rizzo, A.; Chaumette, F. “Avoiding robot joint limits and kinematic singularities in visual servoing”. International Conference in Pattern Recognition (ICPR’96). Vol. A. Pp. 297-301. 1996.
- [Marchand-2001] Marchand, E.; Chaumette, F. “A new formulation for non-linear camera calibration using virtual visual servoing”. Rapport de Recherche IRISA, No 1366, 2001.
- [Marchand-2005] Marchand, E.; Chaumette, F. “Feature tracking for visual servoing purposes”. Robotics and Autonomous Systems, 52(1):53-70, July 2005.
- [Marquardt-1963] Marquardt, D.W. “An Algorithm for the Least-Squares Estimation of Nonlinear Parameters”. SIAM Journal of Applied Mathematics, 11(2):431–441, Jun. 1963.
- [Martinet-1999] Martinet, P.; Gallice J. “Position based visual servoing using a nonlinear approach” Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, (IROS’1999). Kyongju, Korea. Vol. 1. Pp 531-536. 1999.
- [Martinet-2001] Martinet, P.; Cervera, E. “Stacking Jacobians Properly in Stereo Visual Servoing”. IEEE International Conference on Robotics and Automation. Seoul, Korea. Pp. 717-722. 2001.
- [Mezouar-2002a] Mezouar, Y.; Chaumette, F. “Path Planning For Robust Image-based Control”. IEEE Transactions on Robotics and Automation, Vol. 18, No. 4. Pp. 534-549. 2002.
- [Mezouar-2002b] Mezouar, Y.; Chaumette, F. “Avoiding self-occlusions and preserving visibility by path planning in the image”. Robotics and Autonomous Systems. Vol. 41. No. 2. Pp. 77-87. 2002.
- [Mezouar-2003] Mezouar, Y.; Chaumette, F. “Optimal camera trajectory with image-based control”. International Journal of Robotics Research. Vol. 22. No. 10-11. Pp. 781-804. 2003.
- [Morel-1998] Morel, G.; Malis, E.; Boudet, S. “Impedance based combination of visual and force control”. IEEE International Conference on Robotics and Automation, Leuven, Bélgica. Pp. 1743-1748. 1998.

- [Mut-1998] Mut, V.; Nasisi, O.; Carelli, R.; Kuchen, B. "Tracking adaptive impedance robot control with visual feedback". IEEE International Conference on Robotics and Automation, Leuven, Bélgica. Pp. 2002-2008. 1998.
- [Nakabo-1998] Nakabo, Y.; Ishikawa, M. "Visual impedance using 1ms visual feedback system". IEEE International Conference on Robotics and Automation, Leuven, Bélgica. Pp. 2333-2338. 1998.
- [Namiki-1999] Namiki, A.; Nakabo, I.; Ishikawa, M. "High speed grasping using visual and force feedback". IEEE International Conference on Robotics and Automation. Detroit, MI. Pp. 3195-3200. 1999.
- [Nasisi-2003] Nasisi, O.; Carrelli, R. "Adaptive servo visual robot control". Robotics and Autonomous Systems, Vol. 43. Pp. 51-78. 2003.
- [Nelson-1993] Nelson, B. J.; Papanikolopoulos, N. P.; Khosla, P. K. "Visual servoing for robotic assembly" in Visual Servoing-Real-Time Control of Robot Manipulators Based on Visual Sensory Feedback. Ed. K. Hashimoto, World Scientific Publishing Co. Pte. Ltd., River Edge, NJ, Pp. 139-164. 1993.
- [Nelson-1996a] Nelson, B. J.; and Khosla, P. K. "Force and vision resolvability for assimilating disparate sensory feedback". IEEE Transaction on Robotics and Automation, Vol. 12, N° 5. Pp. 714-731, 1996.
- [Nelson-1996b] Nelson, B. J.; Morrow, J.; Khosla, P. K. "Robotic manipulation using high bandwidth force and vision feedback". Mathl. Computer Modelling, Vol. 24. No. 5/6. Pp. 11-29. 1996.
- [Nelson-1997] Nelson, B. J.; Zhou, Y.; Vikramaditya, B. "Integrating force and vision feedback for microassembly". Proceedings of SPIE: Microrobotics and Microsystems Fabrication. Pittsbrugh, PA. Pp. 30-41. 1997.
- [Nelson-2001] Nelson, B. J. "Assimilating disparate sensory feedback within virtual environments for telerobotic systems". Robotics and Autonomous Systems. Vol. 36. No. 1. Pp. 1-10. 2001.
- [Oberkampf-1996] Oberkampf, D.; DeMenthon, F.; Davis, L. S. "Iterative pose estimation using coplanar feature points". Computer Vision and Image Understanding, CVIU. Academic Press INC. Vol 63. No. 3. Pp. 495-511. 1996.
- [Oku-2000] Oku, H.; Ishii, I.; Ishikawa, M. "Tracking a Protozoon Using High-Speed Visual Feedback" Proceedings of 1st Annual Int. IEEE-EMBS Special Topic Conference on Microtechnologies in Medicine & Biology. Lyon, France. Pp.156-159. 2000.
- [Olsson-2002] Olsson, T.; Bengtsson, J.; Johansson, R.; Malm, H. "Force Control and Visual Servoing Using Planar Surface Identification" IEEE International Conference Robotics and Automation. Washington, DC. Pp. 4211-4216. 2002.

- [Papanikolopoulos-1992] Papanikolopoulos, N. P.; Khosla, P. K. "Shared and traded telerobotic visual control". IEEE International Conference Robotics and Automation. Pp. 878-885. 1992.
- [Park-2003] Park, J.; Chung, M. "Path planning with uncalibrated stereo rig for image-based visual servoing under large pose discrepancy". IEEE Transactions on robotics and automation. Vol. 19. No. 2. Pp. 250-258. 2003.
- [Pavlidis-1996] Pavlidis, I.; Papanikolopoulos, N. "Automatic selection of control points for deformable model based target tracking". IEEE International Conference on Robotics and Automation. No. 4. Minneapolis, Minnesota, USA. Pp. 2915-2920. 1996.
- [Perrin-2003] Perrin, D. P.; Kadioglu, E.; Stoeter, S. A. "Grasping and tracking using constant curvature dynamic contours". International Journal of Robotics Research. Vol. 22. No. 10-11. Pp. 855-872. 2003.
- [Pichler-2000] Pichler, J. A.; Jägersand, M. "Uncalibrated hybrid force-vision manipulation". IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'2000). Takamatsu, Japón. Pp. 1866-1871. 2000.
- [Piepmeier-2003] Piepmeier, J.; Lipkin, H. "Uncalibrated eye-in-hand visual servoing". International Journal of Robotics Research. 22(10-11), 805-819. 2003.
- [Piepmeier-2004] Piepmeier, J.; Murray, G.M.; Lipkin, H. "Uncalibrated dynamic visual servoing," IEEE Trans. Robot. Automat., vol. 20, no. 1, pp. 143-147, Feb. 2004.
- [Pomares-2003] Pomares, J.; Torres, F.; Gil, P. "Visual servoing and force control fusion for complex insertion tasks". International Conference on Advanced Robotics. Coimbra (Portugal). Vol. 1. Pp. 357-362. 2003.
- [Pomares-2004a] Pomares, J. "Control visual 2D indirecto para el seguimiento intemporal de trayectorias y su aplicación a estrategias de control visual-fuerza en manipuladores". Tesis Doctoral. Universidad de Alicante. 2004.
- [Pomares-2004b] Pomares, J.; Torres, F. "Control Visual Basado en Flujo de Movimiento para el Seguimiento de Trayectorias con Oclusiones". Revista IEEE América Latina. Vol. 2. Num. 2. 2004.
- [Pomares-2004c] Pomares, J.; Puente, S.T.; Torres, F.; Candelas, F.A.; Gil P. "Virtual disassembly of products based on geometric models". Ed. Elsevier. Computers in Industry. Vol. 55. Num. 1. pp. 1-14. 2004.
- [Pomares-2005a] Pomares, J.; Torres, F. "Movement-Flow-Based Visual Servoing and Force Control Fusion for Manipulation Tasks in Unstructured Environments". IEEE Transactions on Systems, Man, and Cybernetics. Vol. 35. Num. 1. pp. 4-15. 2005.

- [Pomares-2005b] Pomares, J.; García, G.J.; Torres, F. “Impedance control for fusing multisensorial systems in robotic manipulation tasks”. Proc. De la WSEAS International Conference on Dynamical Systems and Control. 2005.
- [Pomares-2006a] Pomares, J.; García, G.J.; Payá, L.; Torres, F. “An uncalibrated approach to track trajectories using visual–force control”. Informatics in Control, Automation and Control II. Ed. Springer. 2006.
- [Pomares-2006b] Pomares, J.; García, G.J.; Payá, L.; Torres, F. “Adaptive visual servoing and force control fusion to track surfaces”. Ed. WSEAS Transactions on Systems. Vol. 5. Num. 1. pp. 25-32. 2006.
- [Pomares-2006c] Pomares, J.; García, G.J.; Torres, F. "Improving tracking trajectories with motion estimation". 3rd International Conference on Informatics in Control, Automation and Robotics (ICINCO 2006). Setubal, Portugal. 2006.
- [Pomares-2006d] Pomares, J.; Puente, S.T.; García, G.J.; Torres, F. "Multi-sensorial system for the generation of disassembly trajectories". World Automation Congress (WAC 2006). Budapest (Hungria). 2006.
- [Pomares-2006e] Pomares, J.; Gil, P.; García, G.J.; Torres, F. “Visual–Force Control and Structured Light Fusion to Improve Recognition of Discontinuities in Surfaces”. 11th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2006). Praga (República Checa). 2006.
- [Pomares-2007a] Pomares, J.; Chaumette, F.; Torres, F. “Adaptive Visual Servoing by Simultaneous Camera Calibration”. IEEE International Conference on Robotics and Automation. ICRA 2007. Roma, Italia. 2007.
- [Pomares-2007b] Pomares, J.; García, G.J.; Torres, F. “A Robust Approach to Control Robot Manipulators by Fusing Visual and Force Information”. Journal of Intelligent and Robotic Systems. Vol. 48. Num. 4. Pp. 437-456. 2007.
- [Raibert-1981] Raibert, M.; Craig, J. “Hybrid position/force control of manipulators”. Journal of Dynamics Systems, Measurement and Control. Vol. 102. Pp. 126-133. 1981.
- [Remaizelles-2006] Remaizelles, A.; Chaumette, F. “Image-based robot navigation from an image memory”. Robotics and Autonomous Systems, doi:10.1016/j.robot.2006.10.002. 2006.
- [Samson-1991] Samson, C.; Espiau, B.; Le Borgne, M. “Robot Control: the Task Function Approach”. Clarendon Press. Oxford, England, 1991.
- [Sanderson-1987] Sanderson, A. C.; Weiss, L. E.; Neuman, C. P. “Dynamic sensor-based control of robots with visual feedback”. IEEE Transactions on Robotics and Automation. Vol. 3. Pp. 404-417. 1987.

- [Sanderson-1980] Sanderson, A. C.; Weiss, L. E. "Image-based visual servo control using relational graph error signals". Proceedings IEEE International Conference on Cybernetics and Society. Pp. 1074-1077. 1980.
- [Schramm-2004] Schramm, F.; Morel, G.; Micaelli, A.; Lottin, A. "Extended-2D visual servoing," in Proc. IEEE Int. Conf. Robotics Automation, New Orleans, LA, pp. 267-273, April 2004.
- [Schramm-2006] Schramm, F.; Morel, G. "Ensuring visibility in calibration-free path planning for image-based visual servoing". IEEE Transactions on Robotics 22 (4), pp. 848-854. 2006.
- [Sebastián-2005] Sebastián, J.M.; Pari, L.; González, C.; Ángel, L. "A new method for the estimation of the image jacobian for the control of an uncalibrated joint system". Lecture Notes in Computer Science, 3522 (I), pp. 631-638. 2005.
- [Sequeira-2004] Sequeira Gonçalves P.J.; Mendonça L.F.; Sousa J.M.; Caldas Pinto J.R. "Fuzzy Model Based Control Applied to Path Planning Visual Servoing". In Progress in Pattern Recognition, Image Analysis and Applications, Alberto Sanfeliu, José F. Martínez Trinidad, Jesús A. Carrasco Ochoa (Eds.), Lecture Notes in Computer Science, Volume 3287, Pages 224 - 231, Springer-Verlag Berlin Heidelberg, 2004.
- [Serra-1981] Serra, J. Image Analysis and Mathematical Morphology. Academic Press. 1981.
- [Shirai-1973] Shirai, Y.; Inoue, H. "Guiding a robot by visual feedback in assembling tasks". Pattern Recognition. Vol. 5. Pp. 99-108. 1973.
- [Shoemake-1985] Shoemake, K. Animating Rotation with Quaternion Curves," Computer Graphics V.19 N.3, 1985.
- [Soatto-1996] Soatto, S.; Frezza, R.; Perona, P. "Motion estimation via dynamic vision". IEEE Transactions on Automatic Control. Vol. 41. No. 3. Pp. 393-413. 1996.
- [Suh-1993] Suh, I. H. "Visual servoing of robot manipulators by fuzzy membership function based neural networks". Robotics and Automated Systems. World Scientific Publishing Co. Pp. 285-315. 1993.
- [Tahri-2005] Tahri, O.; Chaumette, F. "Point-based and región-based image moments for visual servoing of planar objects". IEEE Trans. Robot., vol. 21, n0. 6. Pp. 1116-1127. Dec. 2005.
- [Tatsambon-2007] Tatsambon Fomena, R.; Chaumette, F. "Visual Servoing from Spheres using a Spherical Projection Model". In IEEE Int. Conf. on Robotics and Automation, ICRA'07, Roma, Italia, April 2007.

- [Torres-2002] Torres, F.; Pomares, J.; Gil, P.; Puente, S.; Aracil, R. "Robots y Sistemas Sensoriales". Prentice-Hall, 2002.
- [Torres-2004] Torres, F.; Gil, P.; Puente, S.; Pomares, J.; Aracil, R. "Automatic PC disassembly for component recovery". International Journal of Advanced Manufacturing Technology. Vol. 23. No. 1-2. Pp. 39-46. 2004.
- [Tsai-1987] Tsai, R. Y. "A versatile camera calibration technique for high accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses". IEEE J. Robotics Automat., vol. RA-3, no. 4, pp. 323-344, 1987.
- [Tsuji-1997] Tsuji, T.; Hiromasa, A.; Kaneko, M. "Non-contact impedance control for redundant manipulators using visual information". IEEE International Conference on Robotics and Automation. Albuquerque, USA. Vol. 3. Pp. 2571-2576. 1997.
- [Vikramaditya-1997] Vikramaditya, B.; Nelson, B. J. "Visually Guided Microassembly Using Optical Microscopes and Active Vision Techniques". IEEE International Conference on Robotics and Automation. Albuquerque, USA. Pp. 3172-3177. 1997.
- [Von Collani-1999] Von Collani, Y.; Scheering, C.; Zhang, J.; Knoll, A. "A neuro-fuzzy solution for integrated visual and force control". IEEE International Conference on Robotics and Automation. Detroit, MI. Pp. 2965-2970. 1999.
- [Von Collani-2000] Von Collani, Y.; Ferch, M.; Zhang, J.; Knoll, A. "A general learning approach to multisensor based control using statistic index". IEEE International Conference on Robotics and Automation. San Francisco, CA. Pp. 3221-3226. 2000.
- [Wei-1997] Wei, G. Q., Arbter, K., Hirzinger G. "Real-Time Visual Servoing for Laparoscopic Surgery". IEEE Engineering in Medicine and Biology. Vol. 16, No. 1 Pp. 40-45. 1997
- [Weiss-1984] Weiss, L.E. "Dynamic visual servo control of robots: an adaptative image-based approach". Tesis, CMU-RI-TR-84-16, The Robotics Institute, Carnegie Mellon Univerity. 1984.
- [Whitney-1977] Whitney, D. E. "Force feedback control of manipulator fine motions". Journal of Dynamic Systems, Measurement and Control. Pp. 97-97. 1977.
- [Willsky-1976] Willsky, A. S.; Jones, H. L. "A generalized likelihood ration approach to the detection and estimation of jumps in linear systems". IEEE Transactions on Automatic Control. Vol. 21. N. 1. Pp. 108-112. 1976.
- [Wilson-1996] Wilson, W. J.; Williams Hulls, C. C.; Bell, G. S. "Relative end-effector control using cartesian position based visual servoing". IEEE Transactions on Robotics and Automation. Vol. 12. No. 5. Pp. 684-696. 1996.

- [Wunsch-1997] Wunsch, P.; Hirzinger, G. "Real-time visual tracking of 3-d objects with dynamic handling of occlusions". International Conference on Robotics and Automation. Albuquerque, NM. Pp. 2868-2873. 1997.
- [Xiao-1998] Xiao, D.; Gosh, B. K.; Xi, N.; Tarn, T. J. "Intelligent robotics manipulation with hybrid position/force control in an uncalibrated workspace". IEEE International Conference on Robotics and Automation. Leuven, Bélgica. Pp. 1671-1676. 1998.
- [Xiao-2000] Xiao, D.; Gosh, B. K.; Xi, N.; Tarn, T. J. "Sensor-based hybrid position/force control of a robot manipulator in an uncalibrated environment". IEEE Transactions on Control Systems Technology. Vol 8. No. 4. Pp. 635-645. 2000.
- [Zhang-2002] Zhang, H.; Ostrowski, J. P. "Visual Motion Planning for Mobile Robots". IEEE Transactions on Robotics and Automation. Vol. 18. No. 2. Pp. 199-208. 2002.
- [ZhangZ-1996] Zhang, Z.; Hanson, A.R. "Three-dimensional reconstruction under varying constraints on camera geometry for robotic navigation scenarios", Ph.D. Dissertation. 108-140. 1996.
- [ZhangZ-1999] Zhang, Z. "Flexible camera calibration by viewing a plane from unknown orientations". International Conference on Computer Vision. Corfu, Greece, pp. 666-673. 1999. 23.
- [Zhou-1998] Zhou, Y.; Nelson, B. J.; Vidramaditya, B. "Fusing force and vision feedback for micromanipulation". IEEE International Conference on Robotics and Automation. Leuven, Bélgica. Pp. 1220-1225. 1998.







**Universitat d'Alacant**  
**Universidad de Alicante**