

Universidad Autónoma de San Luis Potosí
Facultad de Ingeniería
Centro de Investigación y Estudios de Posgrado

Control en velocidad, basado en visión, de un robot paralelo tipo Delta

T E S I S

Que para obtener el grado de:
Maestro en Ingeniería Mecánica
Opción: Mecatrónica y Sistemas Mecánicos

Presenta:

Ing. José Antonio Loredo López

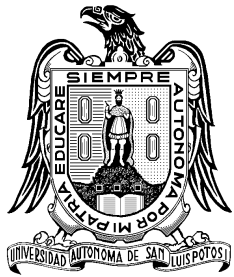
Asesor:

Dr. Mauro Eduardo Maya Méndez

Coasesor:

Dr. Juan Antonio Cárdenas Galindo





Universidad Autónoma de San Luis Potosí
Facultad de Ingeniería
Centro de Investigación y Estudios de Posgrado



Control en velocidad, basado en visión, de un robot paralelo tipo Delta

Maestro en Ingeniería Mecánica
Opción: Mecatrónica y Sistemas Mecánicos

Presenta:

Ing. José Antonio Loredo López

Sinodales:

Dr. Mauro Eduardo Maya Méndez

Asesor

Dr. Juan Antonio Cárdenas Galindo

Coasesor

Dr. Emilio Jorge González Galván

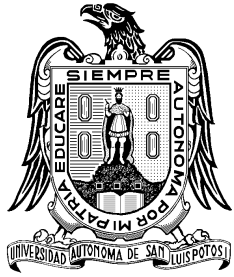
Sinodal

Dr. Héctor Méndez Azúa

Sinodal

Dr. Alejandro González de Alba

Suplente



Universidad Autónoma de San Luis Potosí
Facultad de Ingeniería
Centro de Investigación y Estudios de Posgrado



Aclaración

El presente trabajo que lleva por título:

"Control en velocidad, basado en visión, de un robot paralelo tipo Delta"

Se realizó en el periodo de enero 2016 a febrero 2018 bajo la dirección del Dr. Mauro Eduardo Maya Méndez y del Dr. Juan Antonio Cárdenas Galindo.

Originalidad

Por este medio aseguro que he realizado este documento de tesis para fines académicos sin ayuda indebida de terceros y sin utilizar otros medios más que los indicados.

Las referencias e información tomadas directa o indirectamente de otras fuentes se han definido en el texto como tales y se ha dado el debido crédito a las mismas.

El autor exime a la UASLP de las opiniones vertidas en este documento y asume la responsabilidad total del mismo.

Este documento no ha sido sometido como tesis a ningún otra institución nacional o internacional en forma parcial o total.

Se autoriza a la UASLP para que divulgue este documento de tesis para fines académicos.

José Antonio Loredo López

Agradecimientos

A mis asesores Mauro Maya y Antonio Cárdenas, que han sido mis mentores en el tema de control por visión y por ser pacientes durante todo este proceso de investigación.

A la Facultad de Ingeniería de la Universidad Autónoma de San Luis Potosí y al Consejo Nacional de Ciencia y Tecnología (CONACYT) por la beca otorgada.

A mi madre Julieta, por confiar siempre en mí y motivarme a seguir estudiando en contra de toda adversidad. A mi padre Antonio, porque gracias a él soy un hombre de bien y por apoyarme en los momentos más difíciles. A Lucía, Oscar, Lulú, Gabriel, por ser mi motivo para seguir adelante y poner el ejemplo, sí se puede hermanos.

A mis compañeros Hugo, Ana, Felipe, Robert, Rebeca, Brenda, Fanny, Carlos, porque siempre nos apoyamos mutuamente tratando de salir adelante todos juntos. Pasé buenos momentos durante estos 2 años y medio a su lado.

A Naomi, te estoy eternamente agradecido porque siempre confiaste en mí, me apoyaste en todo durante este largo proceso. Te amo.

Resumen

En este trabajo se desarrolla e implementa una nueva técnica de control basado en visión aplicada a un robot paralelo tipo Delta. Los robots paralelos son arquitecturas de creciente interés por las altas velocidades y aceleraciones permitidas a cambio de un espacio de trabajo relativamente reducido. Estos robots presentan varios problemas abiertos que dificultan que los esquemas de control tradicionales sean directamente aplicables a ellos.

La nueva técnica de control plasmada en este trabajo, está inspirada en la metodología Manipulación en Espacio de Cámara con Modelo Lineal de Cámara (LCM – CSM por sus siglas en inglés), la cual es una metodología de control por visión de robots que recientemente se ha demostrado que puede controlar robots paralelos.

La metodología LCM – CSM está basada en el paradigma de control punto a punto, mientras que la técnica desarrollada utiliza control en velocidad por lo cual llamaremos CSM en velocidades. El control en velocidad le permite hacer posicionamientos con desplazamientos suaves y rápidos, así como realizar seguimiento de trayectorias (esta capacidad permite, entre otras, realizar manipulación de objetos en movimiento).

La nueva metodología CSM en velocidades es un esquema de control de lazo cerrado, lo que permite beneficiarse de las características asociadas (por ejemplo la robustez ante errores del modelo y perturbaciones, así como ofrecer mayores garantías de convergencia y estabilidad). Como parte de este trabajo se presenta un análisis de estabilidad de sistema controlado. En contraste, la metodología LCM - CSM tradicional es un esquema en lazo abierto en el cual se ofrece una garantía formal de la estabilidad.

Se validó experimentalmente el control diseñado. Diversas pruebas experimentales arrojan un error de posicionamiento estático de 1.09 mm en promedio y con un tiempo de ciclo de control de 48 ms que son adecuados para aplicaciones industriales. Se presentan también experimentos que muestran la capacidad del sistema controlado para seguir objetos en movimiento.



UASLP

Universidad Autónoma
de San Luis Potosí

26 de octubre de 2017

**ING. JOSÉ ANTONIO LOREDO LÓPEZ
P R E S E N T E.**

En atención a su solicitud de Temario, presentada por los **Dres. Mauro Eduardo Maya Méndez y Juan Antonio Cárdenas Galindo**, Asesor y Co-asesor de la Tesis que desarrollará Usted, con el objeto de obtener el Grado de **Maestro en Ingeniería Mecánica**, me es grato comunicarle que en la Sesión del H. Consejo Técnico Consultivo celebrada el día 26 de octubre del presente, fue aprobado el Temario propuesto:

TEMARIO:

"Control en velocidad, basado en visión, de un robot paralelo tipo Delta"

Introducción.

1. Antecedentes de Robótica y Control por Visión.
 2. Implementaciones de controles basados en la metodología Manipulación en Espacio de Cámara para un robot paralelo tipo Delta.
 3. Desarrollo de una técnica de control en velocidad basado en la metodología Manipulación en Espacio de Cámara para un robot paralelo tipo Delta.
 4. Resultados experimentales.
- Conclusiones.
Bibliografía.

"MODOS ET CUNCTARUM RERUM MENSURAS AUDEBO"

A T E N T A M E N T E

**M. I. JORGE ALBERTO PÉREZ GONZÁLEZ
DIRECTOR.**



**FACULTAD DE
INGENIERÍA**

Av. Manuel Nava 8
Zona Universitaria • CP 78290
San Luis Potosí, S.L.P.
tel. (444) 826 2330 al39
fax (444) 826 2336
www.uaslp.mx

Copia. Archivo.
*etn.

Índice General

Capítulos	Página
Introducción	1
1. Antecedentes de Robótica y Control por Visión	5
1.1. Robots Industriales	5
1.2. Robots Paralelos	7
1.2.1. Robot Delta	10
1.3. Cinemática del robot paralelo tipo Delta	14
1.3.1. Modelo cinemático directo	15
1.3.2. Modelo cinemático inverso	16
1.3.3. Jacobiano	16
1.4. Problemas abiertos de los robots paralelos	17
1.5. Control por Visión	19
1.5.1. Servo Visual	20

1.5.2.	Manipulación en Espacio de Cámara	23
1.6.	Metodología de control por visión basado en LCM - CSM.	25
1.6.1.	Parámetros Intrínsecos	26
1.6.2.	Parámetros Extrínsecos	28
1.7.	Conclusiones del capítulo 1	30
2.	Implementaciones de controles basados en la metodología Manipulación en Espacio de Cámara para un robot paralelo tipo Delta	33
2.1.	Esquemas de control general de la metodología CSM	33
2.2.	Implementación de la metodología LCM - CSM para el control del robot Delta	35
2.2.1.	Determinación de los 11 parámetros de visión	35
2.2.2.	Estimación lineal de parámetros	36
2.2.3.	Determinación de la posición tridimensional	38
2.3.	Desarrollo de la metodología VW - CSM de un modelo lineal de cámara para el control del robot Delta	41
2.4.	Conclusiones del capítulo 2	43
3.	Desarrollo de una técnica de control en velocidad basado en la metodología Manipulación en Espacio de Cámara para un robot paralelo tipo Delta	45
3.1.	Matriz CSM para velocidades	45
3.2.	Ley de Control	46
3.3.	Marca visual virtual	48
3.4.	Filtro de Kalman	50
3.4.1.	Estimación de la velocidad del punto objetivo	52
3.5.	Análisis de estabilidad	52
3.6.	Singularidades en el robot Delta	53
3.7.	Esquema de Control y Diagrama de Flujo	56

3.8. Conclusiones del capítulo 3	58
4. Resultados Experimentales	59
4.1. Características de la implementación	59
4.1.1. Hardware	59
4.1.2. Software	61
4.1.3. “Pre-plan”	64
4.2. Resultados experimentales de la metodología LCM - CSM	65
4.3. Resultados experimentales de la metodología VW - CSM de un modelo lineal cámara	67
4.4. Sistema de Medición por Visión (SMV)	69
4.5. Resultados experimentales de la metodología CSM en velocidades	72
4.5.1. Sintonización de la ley de control CSM en velocidades	72
4.5.2. Resultados experimentales para objetivos estáticos	76
4.6. Resultados experimentales de la metodología CSM en velocidades para objetos en movimiento	79
4.6.1. Resultados experimentales para objetivos en movimiento con trayecto- ria lineal	79
4.6.2. Resultados experimentales para objetivos en movimiento con trayecto- ria arbitraria	85
4.7. Conclusiones del capítulo 4	87
Conclusiones	89
Bibliografía	93

Lista de Figuras

1.1. Robot Serial de 6 grados libertad ABB IRB-2400 (tomado de [2]).	6
1.2. Robot Paralelo Hexápodo de 6 grados libertad (tomado de [3]).	6
1.3. Posiblemente el primer robot paralelo (tomado de [5]).	8
1.4. El primer robot paralelo industrial patentado en 1942 (tomado de [5]). . . .	8
1.5. Plataforma de Gough de la empresa “Dunlop Tires”(tomado de [6]).	9
1.6. Boceto de la Plataforma de Stewart (tomado de [7]).	9
1.7. Boceto del robot paralelo tipo Delta de Clavel (tomado de [11]).	10
1.8. Robot Delta Demaurex (tomado de [11]).	11
1.9. Robot Delta IRB 340 FlexPicker de ABB (tomado de [11]).	12
1.10. Boceto del “ <i>New University of Western Australia Robot - NUWAR</i> ” (tomado de [17]).	13
1.11. Robot paralelo tipo Delta modelo educativo LKF-2040 (tomado de [18]). . .	13
1.12. Vista superior del robot paralelo tipo Delta (tomado de [4]).	14
1.13. Vista frontal y lateral del robot paralelo tipo Delta (tomado de [4]).	15

1.14. Configuración de las posiciones de la cámara (tomado de [29]).	20
1.15. Sistema dinámico mirar y mover basado en posición (modificado de [29]). . .	21
1.16. Sistema dinámico mirar y mover basado en imagen (modificado de [29]). . .	21
1.17. Servo visual directo basado en posición (modificado de [29]).	22
1.18. Servo visual directo basado en imagen (modificado de [29]).	22
1.19. Esquema del modelo de cámara de orificio (modificado de [43]).	25
1.20. Geometría del modelo de cámara de orificio, (modificado de [31]).	26
1.21. Sistema de coordenadas de la imagen (x_{im}, y_{im}) y el plano de imagen (\tilde{x}, \tilde{y}) (modificado de [31]).	27
1.22. Rotación y traslación entre el sistema coordenado mundial y el sistema coordenado de la cámara (modificado de [31]).	29
2.1. Esquema de control general de la metodología CSM (modificado de [45]). . .	34
2.2. Diagrama de flujo de la metodología LCM - CSM (tomado de [45]).	40
2.3. Diagrama de flujo de la metodología VW - CSM de un modelo de cámara lineal.	42
3.1. Posición de las marcas visuales físicas.	48
3.2. Marca visual virtual del efector final del robot Delta.	49
3.3. Marca visual virtual del punto objetivo.	49
3.4. Diagrama esquemático del Filtro de Kalman.	52
3.5. Posiciones singulares del robot Delta en $\theta_{i2} = 0$ o π para cualquier i	54
3.6. Posiciones singulares del robot Delta en $\theta_{i2} = 0$ o π para todo i	55
3.7. Posición en el robot Delta donde no es posible alcanzar la configuración $\theta_{i3} = 0$ o π para cualquier y todo i	55
3.8. Configuración en el robot delta donde $\theta_{i1} + \theta_{i2} = \pi$	56
3.9. Esquema de control CSM en velocidades.	56
3.10. Diagrama de flujo CSM en velocidades.	57

4.1. Cámara uEye modelo UI-2210SE-M-GL.	60
4.2. Robot Delta Parallax LKF-2040 y posición de las cámaras.	60
4.3. Banda plana con motor D.C. de armadura y driver ABB BALDOR BC 138.	60
4.4. Posición de las cámaras con respecto al robot Delta.	61
4.5. Fotografías tomadas por las dos cámaras uEye donde se observa el robot Delta y el punto objetivo.	62
4.6. Imágenes umbralizadas a partir de las fotografías mostradas en la Fig. 4.5, los puntos blancos representan las marcas visuales del efector final y del objetivo.	62
4.7. Interfaz gráfica ROBWIN CSM v2.6 para el control de la plataforma experimental.	63
4.8. Vista isométrica de la distribución de los puntos usados en el “ <i>Pre-plan</i> ”.	64
4.9. Vista superior (a) y frontal (b) de los puntos usados en el “ <i>Pre-plan</i> ”.	65
4.10. Posición inicial de la trayectoria de transición.	66
4.11. Fotografía de los dos LEDs en el efector final para calibrar el SMV.	69
4.12. Marcas visuales en espacio de cámara durante la prueba preliminar.	72
4.13. Error en pixeles y vector de velocidades angulares generados durante la prueba preliminar.	73
4.14. Fotografía del efector final después de finalizar la tarea de posicionamiento.	74
4.15. Ejemplo de una tarea de posicionamiento variando los valores de la matriz de ganancias \mathbf{G}	74
4.16. Posición de los 5 puntos objetivos para encontrar una \mathbf{G} adecuada.	75
4.17. Error en pixeles y vector de velocidades angulares generados durante una prueba experimental estática de CSM en velocidades.	76
4.18. Tamaño del intervalo para las pruebas de experimentales de objetivos con trayectoria lineal.	80
4.19. Gráficas del error durante una tarea de seguimiento de un objetivo con una velocidad de 7 cm/s.	81

4.20. Gráficas del error durante una tarea de seguimiento de un objetivo con una velocidad de 9.5 cm/s.	82
4.21. Gráficas del error durante una tarea de seguimiento de un objetivo con una velocidad de 12 cm/s.	83
4.22. Vista isométrica de la posición calculada por el SMV del objetivo con trayectorias arbitrarias.	86

Lista de Tablas

1.1. Comparativa entre las características de los robots seriales y paralelos.	7
4.1. Resultados de las pruebas realizadas en la implementación de la metodología LCM - CSM.	66
4.2. Error estadístico de la implementación de la metodología LCM - CSM.	67
4.3. Resultados de las pruebas realizadas en la implementación de la metodología VW - CSM de un modelo lineal.	68
4.4. Error estadístico de la implementación de la metodología VW - CSM de un modelo lineal.	68
4.5. Resultados de la distancia calculada entre los dos LEDs en 8 posiciones usando el SMV.	70
4.6. Incertidumbre estadística del SMV.	71
4.7. Relación mm/píxel a distintas distancias con una resolución de cámara 640 x 480 pixeles.	71
4.8. Valores de g_{ii} antes de que se presente el sobre-tiro en las 5 posiciones.	76
4.9. Error estadístico de la implementación de la metodología CSM en velocidades.	77

4.10. Resultados de las pruebas realizadas en la implementación de la metodología CSM en velocidades.	78
4.11. Error estadístico de las pruebas de seguimiento realizadas a un objetivo en diferentes velocidades	84
4.12. Error en el seguimiento y final de un objetivo con trayectorias arbitrarias. . .	85

Introducción

En la actualidad, la robótica industrial ha tenido un gran auge en la modernización y automatización de los procesos de producción. Los robots industriales, por su estructura cinemática, se pueden clasificar en seriales y paralelos. El robot serial es aquel que tiene una serie de eslabones conectados entre sí por medio de juntas (normalmente de revoluta o prismática), donde uno de sus extremos se encuentra fijo y el otro extremo se mueve con libertad. Por su parte, el robot paralelo está compuesto de una base móvil (efector final) con “ n ” grados de libertad y una base fija, unidas entre sí por al menos dos cadenas cinemáticas independientes.

El robot paralelo al tener más de una cadena cinemática, presenta ciertas ventajas frente al robot serial, como lo son una mayor velocidad y aceleración, rigidez, precisión y relación de capacidad carga/peso. Sin embargo, estas arquitecturas tienen algunas desventajas como un espacio de trabajo reducido, compleja calibración, gran número de parámetros geométricos, entre otros. Además, aún se encuentran problemas abiertos en la literatura como son: técnicas estándar para la resolución de la cinemática directa, el modelado dinámico del manipulador, análisis de las singularidades del mecanismo y cálculo inexacto del Jacobiano.

Estos problemas hacen que las técnicas de control tradicionales empleadas en los robots seriales sean difícilmente aplicables a los robots paralelos. Actualmente, el control de este tipo de robots es un campo de creciente investigación y que, comparado con los robots seriales, existe un número mucho menor de trabajos enfocados al control de mismo. Particularmente,

las técnicas basadas en el control de visión por computadora son una opción viable hacia tareas de posicionamiento “*pick and place*”.

Los sensores visuales son elementos cada vez más frecuentes en aplicaciones robóticas ya que imita el sentido de la visión de un humano y permite la percepción sin contacto de un entorno. Se han desarrollado técnicas de control basado en visión que ofrecen características como compensar errores en el modelo del robot. Entre las metodologías de control por visión más populares están Servo Visual y Manipulación en Espacio de Cámara.

Manipulación en Espacio de Cámara (CSM por sus siglas en inglés) se basa en la estimación de la relación entre la posición tridimensional de un determinado número de marcas visuales adheridas al efector final y su posición correspondiente en imágenes tomadas por al menos dos cámaras. Esta técnica no requiere de un proceso de calibración, ni la resolución del Jacobiano del Robot.

Servo Visual se basa en la integración, en tiempo real, de la información visual obtenida por una o varias cámaras en el ciclo de control del robot de lazo cerrado para realizar su movimiento. Esta técnica se puede categorizar según la señal de error con la que se maneje, Servo Visual basado en posición (también llamado 3D) donde la señal de error está expresada en espacio tridimensional y Servo Visual basado en Imagen (también llamado 2D) donde la señal de error está expresada en espacio de imagen.

Ambas técnicas (CSM y Servo Visual) cuentan con características que las hacen únicas. Por un lado Servo Visual usa un sistema de control de lazo cerrado mientras que CSM usa un control de lazo abierto con técnicas de estimación. Servo Visual hace uso del modelo cinemático y del Jacobiano del robot, además, si usa el enfoque 2D requerirá del Jacobiano de Imagen, por lo que puede controlar tanto la posición como la velocidad del efector final. Por su parte, CSM solamente requiere el modelo cinemático del robot y se enfoca en controlar la posición del mismo, por lo cual no requiere de un Jacobiano de Imagen porque no toma en cuenta el vector de velocidades expresado en el espacio de cámara.

Cabe mencionar que la metodología CSM proporciona un control punto a punto para el robot en tareas de posicionamiento en donde el objetivo se encuentra estático. Esto se vería mermado en muchas aplicaciones industriales, ya que se requiere en ocasiones del seguimiento

de objetivos en movimiento con trayectorias no definidas. Aunque ya existe un trabajo usando CSM para seguir objetos en movimiento con trayectorias definidas, este funciona bajo un modelo de extrapolación para predecir la posición futura del objetivo, lo cual no permite garantizar la convergencia del error a cero.

Con el presente trabajo, se desarrolla e implementa una variante de la metodología CSM para el control de velocidad del robot paralelo tipo Delta. Esto le permitirá hacer tareas de posicionamiento estáticas, con transiciones suaves y realizar seguimientos de objetivos con movimiento no predefinido requiriendo de la ayuda del algoritmo recursivo Filtro de Kalman. En específico, se comenzará con el desarrollo de la nueva variante a partir de la metodología Manipulación en Espacio de Cámara con Modelo Lineal de Cámara (LCM - CSM por sus siglas en inglés). Esta metodología trabaja bajo un modelo lineal que relaciona directamente la posición tridimensional de una marca visual con sus coordenadas en espacio de cámara. Esto permite una mayor facilidad para la derivación de un modelo de velocidad.

Objetivos de la Tesis

Objetivo general

Desarrollo e implementación del control CSM basado en velocidad para un robot paralelo tipo delta modelo PARALLIX LKF-2040.

Objetivos particulares

- Síntesis de un control basado en velocidad y en la metodología CSM para un robot Delta.
 - Análisis de estabilidad del control basado en velocidad y en la metodología CSM.
 - Implementación del control CSM en velocidad.
 - Validación experimental del control propuesto.
-

Descripción de los capítulos

A continuación se describe el contenido de los capítulos incluidos en este trabajo:

- **Capítulo 1:** Se presentan los antecedentes de la robótica industrial ahondando en el robot paralelo tipo Delta. Se muestra el análisis cinemático de este último así como sus problemas en la literatura que aún están abiertos. Además, se muestran los antecedentes de las metodologías de control por visión, describiendo el modelo de cámara usado en la metodología Manipulación en Espacio de Cámara.
 - **Capítulo 2:** Se describe la implementación de las metodologías de control por visión, LCM - CSM y Manipulación en Espacio de Cámara con Variación de Pesos (VW - CSM por sus siglas en inglés) a un robot paralelo tipo Delta. Esta última metodología empleándola hacia LCM - CSM, ya que originalmente fue desarrollada bajo el control Manipulación en Espacio de Cámara + Modelo Ortográfico y “*Flattening*” (OFL-CSM por sus siglas en inglés), y así verificar su precisión con esta nueva variación e incluirla en la nueva técnica desarrollada en este trabajo.
 - **Capítulo 3:** Se desarrolla la ley de control en velocidad basado en la metodología Manipulación en Espacio de Cámara a un robot paralelo tipo Delta. Se describe el uso de Filtro de Kalman para hacer una estimación de la velocidad de objetos en movimiento con el propósito de compensarla. Además, se hace un análisis de singularidades en el robot Delta y un análisis de estabilidad a la ley de control propuesta.
 - **Capítulo 4:** Se muestran y analizan los resultados de las pruebas experimentales de la implementación de las metodologías, LCM - CSM, VW - CSM bajo un modelo de cámara lineal, y la ley de control CSM basado en velocidad.
-

Antecedentes de Robótica y Control por Visión

En este capítulo se presentan dos tipos de robots industriales caracterizados por su estructura cinemática: el robot serial y el robot paralelo. Se presentan las características de cada uno de ellos, así como sus ventajas y desventajas. Se muestra el estado del arte de los robots paralelos (incluyendo los problemas abiertos que éstos presentan) hasta llegar a uno de su tipos más populares, el robot Delta. Se describe el principio del funcionamiento del robot Delta y las aportaciones que ha tenido en el paso del tiempo. Se presenta además, una breve descripción de las técnicas de visión por computadora más sobresalientes, destacando las técnicas Manipulación en Espacio de Cámara y Servo Visual.

1.1. Robots Industriales

La robótica es un campo de la ingeniería moderna, que requiere el conocimiento de mecánica, electricidad, electrónica, control, computación y matemáticas. Actualmente, la robótica es mayormente aplicada en el sector industrial, ofreciendo como ventaja automatizar procesos de producción, relevar al operario en tareas peligrosas, dar mayor precisión a tareas delicadas y repetitivas, reducir costos de producción y aumentar la flexibilidad de adaptación gracias a que se puede re-programar la serie de movimientos que los robots realizan.

Existen varias formas de clasificar los robots, pero para términos del presente trabajo se usará una clasificación basada en la estructura cinemática del robot. Una cadena cinemática es una serie de al menos dos eslabones conectados entre sí por medio de juntas, normalmente tipo revoluta o prismáticas.

De acuerdo a su estructura cinemática, los robots se pueden clasificar en seriales o paralelos [1]. Un robot serial es un mecanismo constituido por una cadena cinemática, en donde uno de los eslabones que se encuentran en los extremos está fijo (base) y el otro se encuentra libre en el espacio (efector final). En la Fig. 1.1 se observa un robot serial con seis grados de libertad usado comúnmente en la industria, tres de los grados de libertad para lograr el posicionamiento y los tres sobrantes para la orientación del efector final.



Figura 1.1. Robot Serial de 6 grados libertad ABB IRB-2400 (tomado de [2]).

Por su parte, el robot paralelo cuenta con dos o más cadenas cinemáticas que van desde un eslabón fijo (base) a un eslabón móvil (efector final) formando así una cadena cinemática cerrada. En la Fig. 1.2 se observa una robot paralelo conocido como hexápodo comúnmente usada en simulaciones de vuelo.



Figura 1.2. Robot Paralelo Hexápodo de 6 grados libertad (tomado de [3]).

Las características varían entre el robot serial y el robot paralelo [1]. En la Tabla 1.1 se muestra una comparativa cualitativa de las principales características de los robots seriales y paralelos.

Características	Robot Serial	Robot Paralelo
Espacio de Trabajo	Grande	Pequeño e irregular
Solución de la cinemática directa	Fácil	Difícil
Solución de la cinemática inversa	Difícil	Fácil
Error en posición	Acumulativo	Promediado
Error en fuerza	Promediado	Acumulativo
Rigidez con respecto al peso del robot	Baja	Alta
Modelado y solución de la dinámica	Simple	Complejo
Inercia	Grande	Pequeña
Relación carga/peso	Baja	Alta
Velocidad y Aceleración	Baja	Alta
Precisión	Baja	Alta
Calibración	Simple	Complejo

Tabla 1.1. Comparativa entre las características de los robots seriales y paralelos.

En la Tabla 1.1, se observan algunas ventajas de los robots paralelos con respecto a los robots seriales, como mayor velocidad y aceleración de operación y mayor rigidez, precisión y relación carga/peso [4]. Estas características hacen que los robots paralelos tengan diversas áreas de aplicación o interés como simuladores, aplicaciones médicas e industriales, y puedan realizar tareas de posicionamiento “*pick and place*”. Sin embargo, los robots paralelos presentan desventajas frente a los robots seriales como tener un espacio de trabajo reducido, difícil calibración mecánica, gran número de singularidades y cinemática compleja. Estas desventajas hacen que los esquemas de control tradicionales sean difícilmente aplicables hacia los robots paralelos.

1.2. Robots Paralelos

Según Ilian Bonev [5], posiblemente la plataforma paralela fue diseñada y patentada por James E. Gwinnett en Estados Unidos, 1931. Esta plataforma fue creada para la industria

del entretenimiento aunque no se sabe si se llegó a construir. La Fig. 1.3 [5] muestra que la plataforma paralela se encontraba debajo de las butacas de una sala de cine, este mecanismo movía las butacas para simular la interacción con lo proyectado en la pantalla.

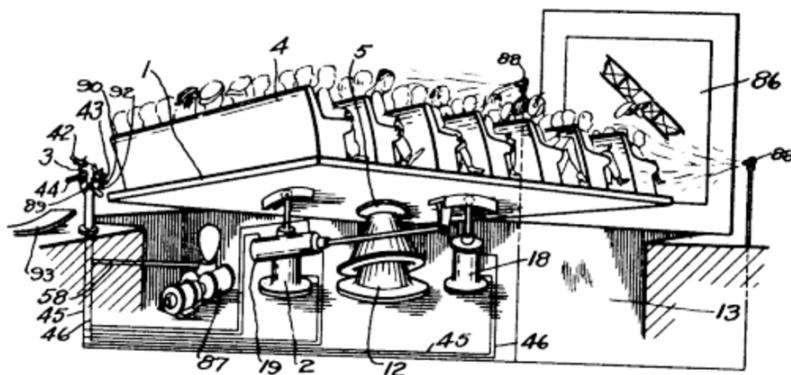


Figura 1.3. Posiblemente el primer robot paralelo (tomado de [5]).

Más tarde en 1942, Willard L. V. Pollard estableció el primer diseño de un robot paralelo industrial para una línea de pintura por aspersión automatizada, Fig. 1.4 [5]. Este robot está constituido por tres cadenas cinemáticas cerradas de dos eslabones cada una, unidas a través de una base a una plataforma móvil por medio de juntas universales dándole 5 grados de libertad.

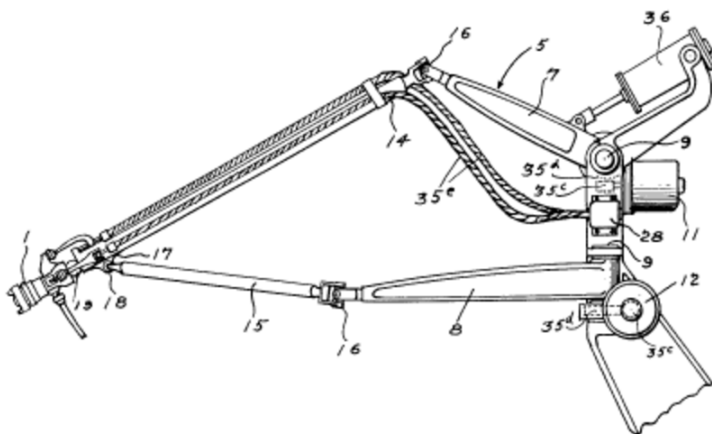


Figura 1.4. El primer robot paralelo industrial patentado en 1942 (tomado de [5]).

En 1962, Gough y Whitehall proponen el diseño de un robot paralelo con bases hexagonales para pruebas de llantas de automóvil para la empresa “Dunlop Tires”, Fig. 1.5 [6]. Este mecanismo simula el camino cambiando la orientación de la llanta por medio de actuadores lineales y juntas universales.

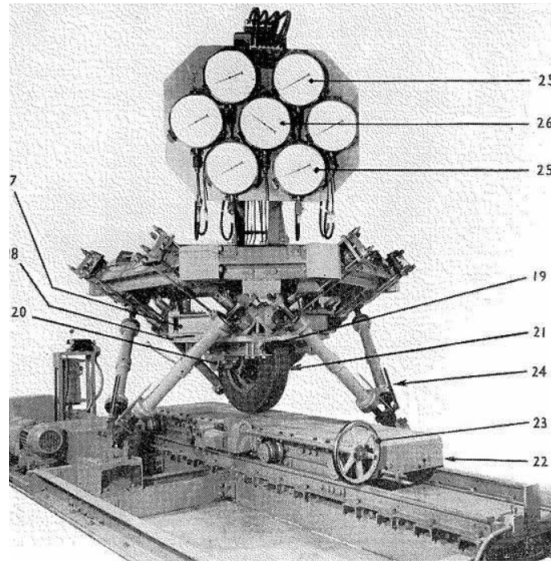


Figura 1.5. Plataforma de Gough de la empresa “Dunlop Tires”(tomado de [6]).

En 1965, Stewart propone el diseño de una plataforma paralela de 6 grados de libertad para darle uso como simulador de vuelo. El mecanismo está conformado por una plataforma triangular móvil conectada por juntas esféricas a 3 actuadores lineales que se conectaban a una base fija, Fig. 1.6 [7].

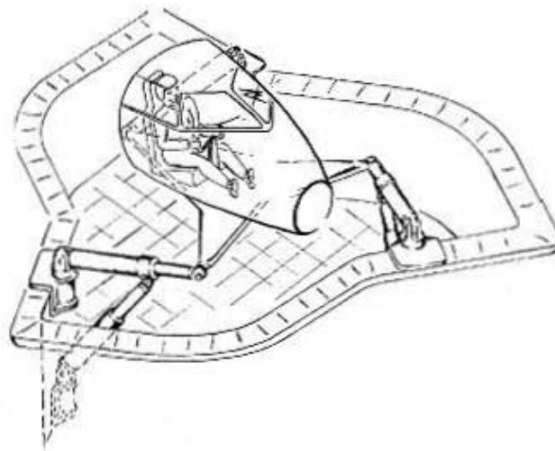


Figura 1.6. Boceto de la Plataforma de Stewart (tomado de [7]).

Más tarde en 1978, Hunt [8] propuso que los mecanismos paralelos usados en las plataformas de los simuladores de vuelo se usaran como robots manipuladores. Además, destacó que los mecanismos requerían de un estudio más detallado referente a las aplicaciones robóticas, ya que a simple vista ofrecían ventajas en cuanto a rigidez y precisión con respecto a los

robots seriales que eran muy comunes en la industria. En 1979, McCallion y Truong [9] propusieron usar la plataforma de Stewart como un manipulador paralelo dentro de una celda de armado robotizada, esto porque la posición del efector final es mucho menos sensible a perturbaciones comparado con el robot serial.

Con el paso del tiempo han surgido diversas configuraciones de plataformas paralelas, siendo una de las más populares el robot paralelo tipo Delta. Se describirá esta plataforma en la siguiente sección, así como trabajos bajo las siguientes líneas de investigación: optimizaciones para el espacio de trabajo, cinemática, dinámica, singularidades, calibración y control de las plataformas tipo Delta.

1.2.1. Robot Delta

En 1988, Clavel [10] tuvo la idea de usar paralelogramos para construir un robot paralelo de tres grados de libertad traslacionales y un grado de libertad rotacional, Fig. 1.7 [11]. Raymond Clavel llamó a su creación el Robot Delta sin sospechar que al final de siglo, sería uno de los diseños de robots paralelos más usados en el ramo industrial con una gran cantidad de estructuras activas en todo el mundo.

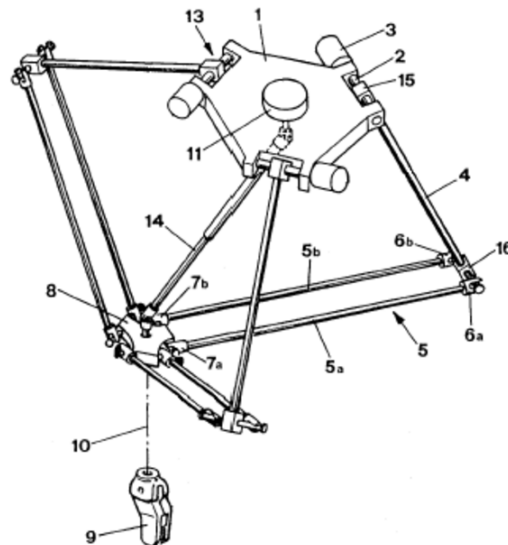


Figura 1.7. Boceto del robot paralelo tipo Delta de Clavel (tomado de [11]).

La idea básica detrás del diseño del robot delta es el uso de paralelogramos, los cuales permiten que el eslabón de salida conserve su orientación fija con respecto al eslabón de entrada. Los eslabones de entrada de los tres paralelogramos están montados sobre palancas rotatorias por medio de juntas de revoluta. Las uniones entre los eslabones de entrada y los paralelogramos son mediante juntas esféricas, de igual forma que las uniones entre los paralelogramos y la plataforma móvil. Finalmente, se utilizó un cuarto eslabón para transmitir el movimiento rotacional desde la base hacia el efector final montado en la plataforma móvil.

Varios modelos dinámicos han surgido a partir de la creación del robot Delta. En 1991, Codoney et al [12] desarrolla un modelo basado en el método Newton-Euler junto con un controlador del mismo. En 1992, Miller [13] propone otro modelo usando el enfoque Lagrangiano. Tres años después, él mismo [14] propone otro modelo ahora basándose en el método del principio de Hamilton, donde establece que ese modelo no requiere de aceleraciones como entrada para calcular el torque del motor.

En 2006, López et al. [15] emplearon el análisis cinemático para el desarrollo de las matrices jacobianas directa e inversa del robot delta, las cuales ayudan para encontrar configuraciones singulares. Además, proponen la idea de una matriz Jacobiana intermedia que relaciona la velocidad del efector final con la variación de la longitud de una cadena cinemática. En 2013, Maya et al. [4] propusieron un mecanismo de reconfiguración para aumentar o disminuir la base fija del robot delta, con esto, analizaron el espacio de trabajo y la capacidad de carga.



Figura 1.8. Robot Delta Demareux (tomado de [11]).

En cuanto a su comercialización, todo empezó en 1987 cuando los hermanos Marc-Olivier y Pascal Demareux crearon la compañía “*Demaurex*” y obtuvieron la licencia para la producción del robot Delta. El mayor objetivo de esta compañía era la incursión del robot Delta en la industria del empaquetado, donde años más tarde lograron su cometido alegando haber vendido 500 robots a nivel mundial, Fig. 1.8 [11].

Posteriormente, varias empresas han adquirido la licencia para la fabricación del robot Delta, l'École Polytechnique Fédérale de Lausanne (EPFL), DeeMed, Swedish Elekta, Elekta IGS, Hitachi Seiki, etc. Sin embargo, uno de los modelos más representativos del robot Delta es el modelo IRB 340 FlexPicker fabricado en 1999 por “*ABB Flexible Automation*”. Este robot fue implementado en los sectores alimenticio, farmacéutico y la industria electrónica, Fig. 1.9 [11].



Figura 1.9. Robot Delta IRB 340 FlexPicker de ABB (tomado de [11]).

Dentro del ramo académico, fue muy atractivo para los laboratorios de las universidades el estudio del robot Delta por las prestaciones que brindaba. Surgieron tantas variantes académicas considerables del robot Delta como en el sector industrial. Una de estas variantes fue construida en la Universidad de Maryland [16], donde se busca la maximización del espacio del trabajo.

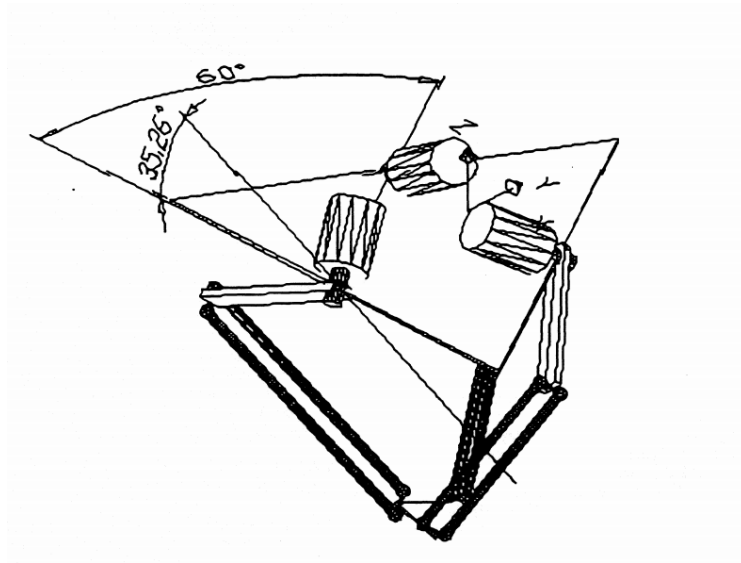


Figura 1.10. Boceto del “*New University of Western Australia Robot - NUWAR*” (tomado de [17]).

Otra versión optimizada del robot Delta es el robot NUWAR, Fig.1.10 [17], fue desarrollado en la Universidad de Australia Occidental (UWA) donde la idea detrás del diseño consiste en modificar la orientación de los motores para optimizar el espacio de trabajo y cuando esté terminado se espera que sea el más rápido del mundo ya que podrá alcanzar aceleraciones de hasta 500 m/s^2 .



Figura 1.11. Robot paralelo tipo Delta modelo educativo LKF-2040 (tomado de [18]).

Para el presente trabajo se va a utilizar la versión académica del robot Delta PARALLIX LKF-2040, desarrollada en el Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada (CICATA) del Instituto Politécnico Nacional, Querétaro. Este robot, al ser un mecanismo paralelo, comparte las mismas características como mayor aceleración y velocidad de operación, mayor relación carga/peso, rigidez y precisión. Tiene sus orígenes en la configuración del robot paralelo tipo Delta (Fig. 1.7) pero tiene articulaciones universales en lugar de esféricas, lo que le proporciona un espacio de trabajo más amplio, Fig. 1.11 [18].

1.3. Cinemática del robot paralelo tipo Delta

La Fig. 1.12 [4] y Fig. 1.13 [4] muestran los parámetros geométricos del robot Delta que están dados por: los radios de las plataformas fija (R) y móvil (r), las longitudes del eslabón actuado ($L1$) y del paralelogramo ($L2$), el ángulo constante de cada brazo con respecto al sistema coordenado del robot (α_i) y los ángulos de desplazamiento de las juntas activas (θ_{i1}) y las juntas pasivas (θ_{i2} y θ_{i3}). El subíndice i se refiere a cada cadena cinemática de robot.

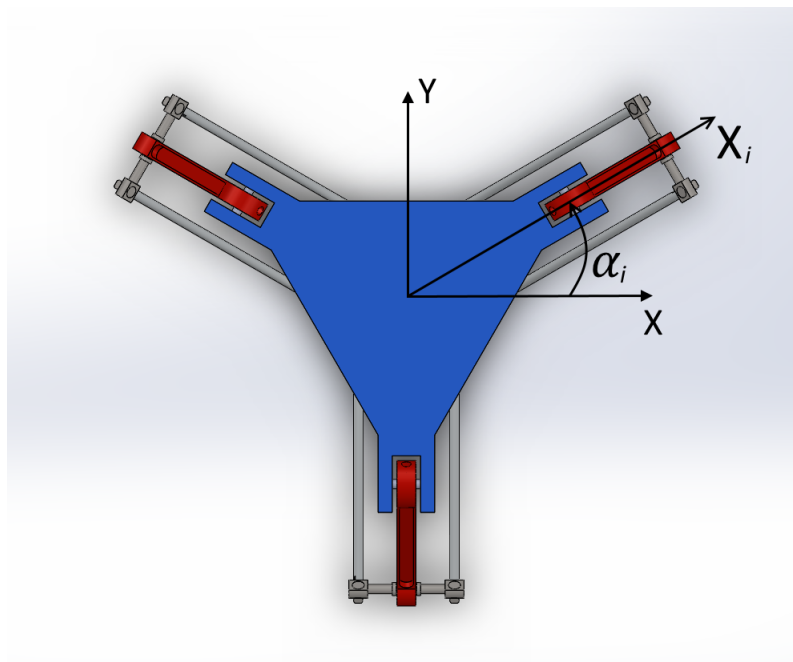


Figura 1.12. Vista superior del robot paralelo tipo Delta (tomado de [4]).

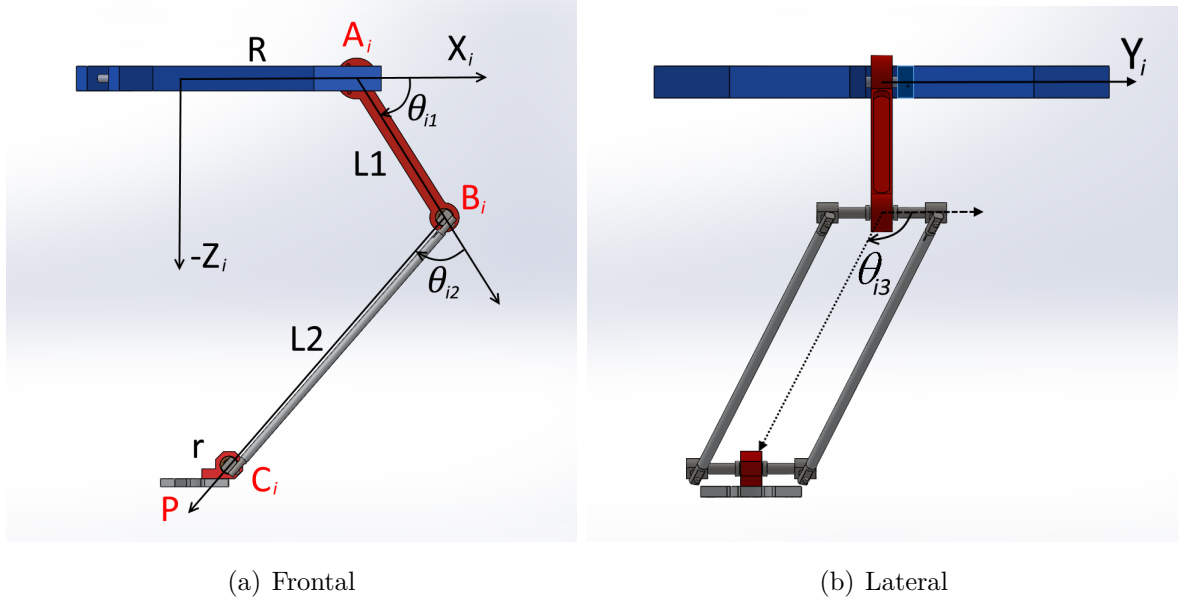


Figura 1.13. Vista frontal y lateral del robot paralelo tipo Delta (tomado de [4]).

1.3.1. Modelo cinemático directo

La cinemática directa de un robot consiste en un conjunto de ecuaciones, conformado por parámetros geométricos y el valor de las articulaciones, para determinar cuál es la posición y orientación del efector final con respecto a un marco de referencia. Maya et al. en [4] describen el modelo cinemático directo del robot Delta en las siguientes ecuaciones:

$$(X_i - X_p)^2 + (Y_i - Y_p)^2 + (Z_i - Z_p)^2 = L_2^2 \quad (1.1)$$

donde las variables X_p, Y_p, Z_p representan las coordenadas del centro de la plataforma móvil (Fig. 1.13) y las variables X_i, Y_i, Z_i están definidas por las siguientes ecuaciones:

$$\begin{aligned} X_i &= (R + L_1 \cos(\theta_{i1}) - r) \cos(\alpha_i) \\ Y_i &= (R + L_1 \cos(\theta_{i1}) - r) \sin(\alpha_i) \\ Z_i &= -L_1 \sin(\theta_{i1}) \end{aligned} \quad (1.2)$$

Para los robots paralelos, la cinemática directa puede presentar múltiples soluciones y no contar con una metodología sistemática (como en los robots seriales) para encontrar alguna de estas soluciones. En este caso, la cinemática directa del robot Delta puede presentar dos soluciones al resolver la Ec. (1.1), por lo que la plataforma móvil puede presentar dos posibles configuraciones con respecto a la plataforma fija [4].

1.3.2. Modelo cinemático inverso

La cinemática inversa de un robot consiste en un conjunto de ecuaciones, conformado por parámetros geométricos y el valor de la posición y orientación del efector final, para encontrar el valor de las articulaciones activas del robot. De igual modo que la cinemática directa, Maya et al. [4] describen el modelo cinemático inverso del robot Delta con las siguientes ecuaciones:

$$\tan\left(\frac{\theta_{i1}}{2}\right) = \frac{-2Z_p \pm \sqrt{4Z_p^2 + 4R_1^2 - S_i^2 + Q_i^2 \left(1 - \frac{R_1^2}{L_1^2}\right) + Q_i \left(\frac{-2R_1 S_i}{L_1} - R_1\right)}}{-2R_1 - Q_i \left(\frac{R_1}{L_1} - 1\right) - S_i} \quad (1.3)$$

donde:

$$\begin{aligned} Q_i &= 2X_p \cos(\alpha_i) + 2Y_p \sin(\alpha_i) \\ R_1 &= R - r \\ S_i &= \frac{1}{L_1} \left(-X_p^2 - Y_p^2 - Z_p^2 + L_2^2 - L_1^2 - R_1^2\right) \end{aligned} \quad (1.4)$$

1.3.3. Jacobiano

La matriz jacobiana proporciona una transformación de la velocidad del efector final en el espacio cartesiano ($\dot{\mathbf{x}}$), a las velocidades articulares del robot ($\dot{\mathbf{q}}$). La matriz jacobiana suele usarse para generar trayectorias, determinar las posiciones singulares y describir los límites del espacio de trabajo del manipulador [19]. La transformación de las velocidades tridimensionales del efector final a las velocidades articulares del robot Delta [15] está dada por:

$$\dot{\mathbf{q}} = J_q^{-1} J_x \dot{\mathbf{x}} \quad (1.5)$$

donde J_x y J_q son los jacobianos del robot Delta. Estos jacobianos son matrices [3x3] debido a que se tiene 3 grados de libertad en el efector final y 3 articulaciones actuadas. Los elementos de las matrices jacobianas están conformados por:

$$J_x = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad (1.6)$$

$$J_q = \text{diag}(b_{11}, b_{22}, b_{33}) \quad (1.7)$$

donde:

$$\begin{aligned}
 a_{i1} &= \text{sen}(\theta_{i3}) \cos(\theta_{i1} + \theta_{i2}) \cos(\alpha_i) + \cos(\theta_{i3}) \text{sen}(\alpha_i) \\
 a_{i2} &= \text{sen}(\theta_{i3}) \cos(\theta_{i1} + \theta_{i2}) \text{sen}(\alpha_i) - \cos(\theta_{i3}) \cos(\alpha_i) \\
 a_{i3} &= -\text{sen}(\theta_{i3}) \text{sen}(\theta_{i1} + \theta_{i2}) \\
 b_{ii} &= L1 \text{sen}(\theta_{i2}) \text{sen}(\theta_{i3})
 \end{aligned} \tag{1.8}$$

Cabe mencionar que J_x y J_q son dependientes de la configuración del robot, por lo que el valor numérico de las variables b_{ii} y a_{i1} , a_{i2} , a_{i3} dependerá de los valores instantáneos de la posición del efector final del robot. Así, la matriz jacobiana será diferente en cada uno de los puntos del espacio de trabajo.

1.4. Problemas abiertos de los robots paralelos

Los robots paralelos cuentan con problemas abiertos en la literatura [20], siendo estos: diseño mecánico, optimizaciones para el espacio de trabajo, la solución de la cinemática directa, modelado dinámico, análisis de las singularidades, calibración y control de las plataformas paralelas. En el diseño mecánico, los robots paralelos van de 2 a 6 grados de libertad (DOF por sus siglas en inglés), donde el 80% tienen 3 y 6 DOF y el resto cuenta con 2, 4 y 5 DOF [1]. No existe una metodología de síntesis ni una topología definida para la construcción de este tipo de manipuladores.

El espacio de trabajo en los robots paralelos es de forma compleja y tamaño reducido, además de que no es fácil de procesarlo en una gráfica que lo muestre de manera completa. El problema abierto consiste en diseñar un algoritmo que permita hacer un mapeo del espacio de trabajo y la posible interferencia física o colisiones entre los eslabones al momento de realizar una trayectoria deseada. Uno de los enfoques para resolver esto, es calcular una aproximación de las fronteras de todo el espacio de trabajo usando un método numérico, además de desarrollar un algoritmo que permita verificar las interferencias en los eslabones.

En la solución de la cinemática directa, se vio anteriormente que a diferencia de los robots seriales, los robots paralelos pueden tener más de una solución y que no cuenta con una metodología estándar para resolverla [20]. Además, es necesario crear un algoritmo que sea capaz de rechazar las soluciones que físicamente el robot no pueda alcanzar por interferencias en la trayectoria o por la presencia de singularidades dentro del espacio de trabajo [21].

Una de las ventajas del uso de los robots paralelos, son sus altas velocidades y aceleraciones que pueden alcanzar debido al poco peso de sus elementos móviles. Estas características, junto con las fuerzas articulares y los parámetros dimensionales, son usadas para el cálculo del modelo dinámico. Sin embargo, uno de los problemas abiertos de los robots paralelos es determinar un modelo dinámico apropiado, donde varias formulaciones se han realizado pero siendo solo modelos simplificados [22]. Otro de los problemas, es la implementación de un control basado en el modelo dinámico, ya que el tipo de solución es alto complejo para usarlo dentro de un ciclo de control de lazo cerrado [23].

En los robots paralelos el análisis de las singularidades se vuelve más complicado por su arquitectura cinemática [9]. Una singularidad ocurre cuando las matrices jacobianas son indeterminadas y por ende no se pueden invertir. Gosselin y Angeles [24] clasifican las singularidades en tres tipos: El primer tipo ocurre cuando el manipulador alcanza los límites internos o externos del espacio del trabajo, perdiendo uno o más grados de libertad en el efector final. La segunda singularidad ocurre cuando el efector final se puede mover aún cuando las juntas actuadas están bloqueadas. El tercer tipo ocurre cuando las dos anteriores están presentes. El análisis de las singularidades es un punto importante a considerar, ya que la ley de control presentada en este trabajo requiere de la inversión del Jacobiano.

Se ha descrito previamente que los robots paralelos cuentan con mayor precisión que los robots seriales debido a que cuentan con más de una cadena cinemática. Sin embargo, esta precisión depende de la correcta calibración del manipulador. La calibración consiste en determinar o conocer los parámetros geométricos o cinemáticos más cercanos al modelo del robot. La incertidumbre de estos parámetros se debe a tolerancias de fabricación, errores de montaje y desplazamientos entre los eslabones [25]. Existen dos métodos de calibración:

- **Calibración externa**, un dispositivo de medición externo es usado para determinar la diferencia entre la posición real y la posición deseada de la configuración del manipulador, esta diferencia o error puede ser usado para la calibración [26].
- **Auto-calibración**, la plataforma cuenta con sensores extras en las juntas pasivas, permitiendo tener más mediciones de todo el espacio de trabajo [27].

Para que un robot paralelo sea un manipulador efectivo, se necesita que el controlador del mismo sea capaz de lidiar con sus características antes mencionadas. Actualmente no existe una técnica de control que sea capaz de resolver todos los problemas asociados a este tipo

de plataformas. El número de trabajos reportados en la literatura dedicados al control del robot paralelo son escasos en comparación al robot serial. A continuación se presenta que es control por visión y como estas técnicas ayudan a resolver algunos de los problemas abiertos de los robots paralelos.

1.5. Control por Visión

En la actualidad, los sensores visuales son elementos cada vez más frecuentes en las aplicaciones robóticas, ya que imitan el sentido de la visión de un ser humano. Su principal ventaja es que permiten obtener una descripción del entorno bastante completa de una forma no intrusiva. La interacción entre sistemas de visión y sistemas robóticos ha sido y es objeto de gran interés en la investigación de los últimos años. Estas investigaciones tienen especial relevancia en las aplicaciones donde el manipulador tiene que interactuar con objetos estáticos o móviles y cuya localización o perfil de movimiento no sea conocido.

Las técnicas de control por visión reemplazan los esquemas tradicionales de enseñar y repetir, en la que los robots realizan una serie de movimientos pre-programados. Estas técnicas de control se caracterizan por absorber los errores geométricos del modelo cinemático del robot por lo que no es necesario que el robot se encuentre calibrado. Esto último, hace que estas técnicas sean atractivas para implementarlas tanto en robots seriales como en robots paralelos. Además, uno de los problemas abiertos de los robots paralelos es la solución de la cinemática directa, el cual puede ser resuelto con alguna de las técnicas de control por visión que se verán más adelante.

Para aumentar la precisión y robustez en las tareas de posicionamiento estático o seguimiento de objetivos en movimiento, ha sido posible el desarrollo de técnicas de control por visión que han ido evolucionando a través del tiempo. De estas técnicas, las más importantes son: Teleoperación, Técnicas de Calibración, Servo Visual y más recientemente Manipulación en Espacio de Cámara, las cuales se describen a continuación, profundizando en las dos últimas mencionadas.

1.5.1. Servo Visual

El término Servo Visual fue introducido por Agin en 1979 [28]. Esta técnica surgió como otra alternativa en los sistemas de control por visión, para aumentar la precisión de las tareas que realizaba un manipulador retroalimentando la información visual (*control lazo cerrado*). Ya que los sistemas de visión y manipulación de un robot trabajan en el modelo tradicional *control lazo abierto*, ver y luego dirigirse al punto observado. El objetivo de Servo Visual es la integración en tiempo real de las informaciones visuales obtenidas por una o varias cámaras en el ciclo de control del robot para realizar su movimiento.

Normalmente Servo Visual usa dos configuraciones de cámaras: montada en el efector final del robot, o fija en algún punto fuera de los límites de trabajo del robot. En la primera (comúnmente llamada *cámara en mano*), la cámara está montada en el efector final del robot y por lo tanto se conoce la relación entre la pose de la cámara y la pose del efector final, esta pose se encuentra representado por ${}^e\mathbf{H}_c$, Fig. 1.14. En la segunda (comúnmente llamada *cámara fija*), la cámara se encuentra estática en algún punto fuera de los límites de trabajo, por lo que se tiene que conocer la pose entre la cámara y al marco del robot representado por ${}^o\mathbf{H}_c$, Fig. 1.14 [29].

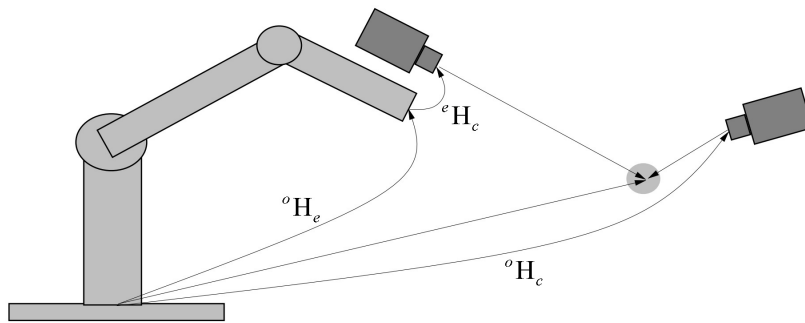


Figura 1.14. Configuración de las posiciones de la cámara (tomado de [29]).

Desde el surgimiento de Servo Visual, se desarrollaron variantes de esta técnica por lo que Sanderson y Weiss [30] propusieron una taxonomía en la cual se distinguía entre cada una de ellas. Para que estas variantes fueran categorizadas, se tomaban en cuenta estas dos cuestiones:

1. ¿Es la estructura de control jerárquica?, es decir, ¿el sistema de visión se encarga de proporcionar el valor de las juntas hacia el controlador del robot o se encarga del cálculo directo de las juntas?

2. ¿La señal de error es definida en coordenadas 3D o en términos de las características de la imagen?

La primera clasificación distingue que, si el control visual es jerárquico y usa el sistema de visión para proveer puntos de ajuste hacia el controlador de las juntas (el cual hace el uso de retroalimentación para estabilizar internamente al robot), se le conoce como *sistema dinámico mirar y mover*, Fig. 1.15 y Fig. 1.16. Contrario a lo que hace el sistema *servo visual directo*, elimina el controlador del robot reemplazándolo con el controlador del servo visual que configura directamente las juntas, Fig. 1.17 y Fig. 1.18.

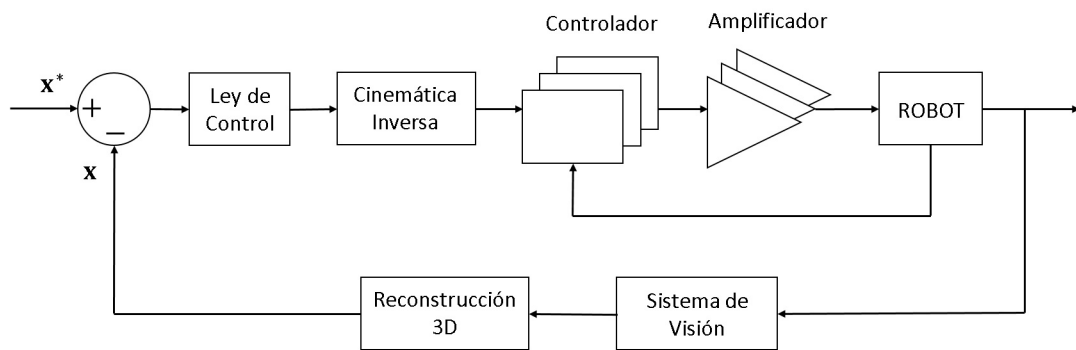


Figura 1.15. Sistema dinámico mirar y mover basado en posición (modificado de [29]).

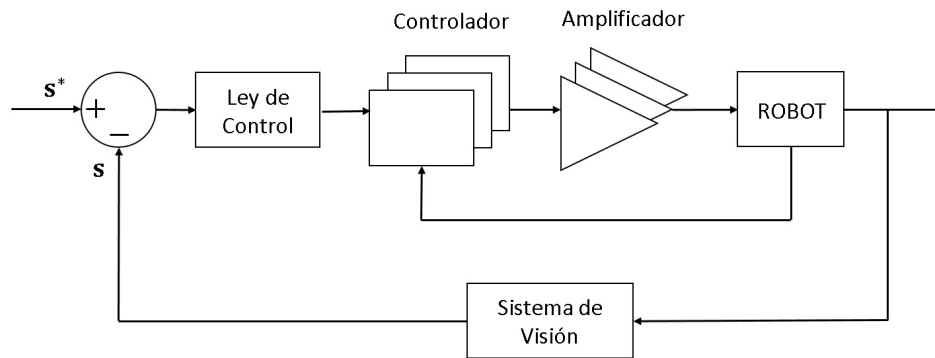


Figura 1.16. Sistema dinámico mirar y mover basado en imagen (modificado de [29]).

La segunda clasificación distingue el control basado en posición del control basado en imagen. En el primero, las características son extraídas de la imagen y usadas en conjunto con el modelo geométrico del objetivo y conociendo el modelo de la cámara estimar la pose

del objetivo con respecto de la cámara, la señal de error está dada en coordenada tridimensionales (\mathbf{x}), Fig. 1.15 y Fig. 1.17. Para trabajar con información tridimensional a partir de las características de la imagen, se requiere de un proceso de reconstrucción 3D conocido como *triangulación*. Previamente a la triangulación, se tiene que realizar la calibración de la cámara, que consiste en obtener sus parámetros intrínsecos, y la calibración estéreo, en donde se calculan los parámetros extrínsecos de la cámara derecha con respecto de la izquierda [31,32].

En el segundo, la señal de retroalimentación en el ciclo de control es con base en las características de la imagen, por lo tanto el error viene dado por pixeles (\mathbf{s}). En esta estrategia de control, se introduce el uso del Jacobiano de la Imagen, que relaciona las velocidades en el plano de imagen (pixeles/s) y las velocidades tridimensionales (mm/s) del efector final con respecto al marco de la cámara [33], Fig. 1.16 y Fig. 1.18.

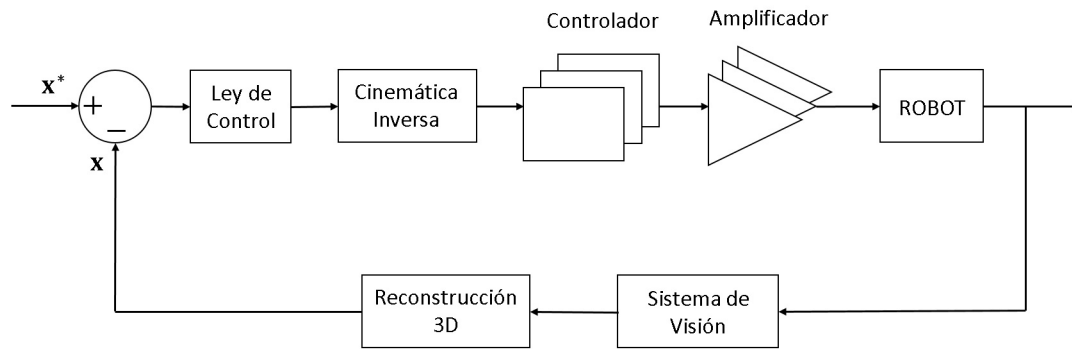


Figura 1.17. Servo visual directo basado en posición (modificado de [29]).

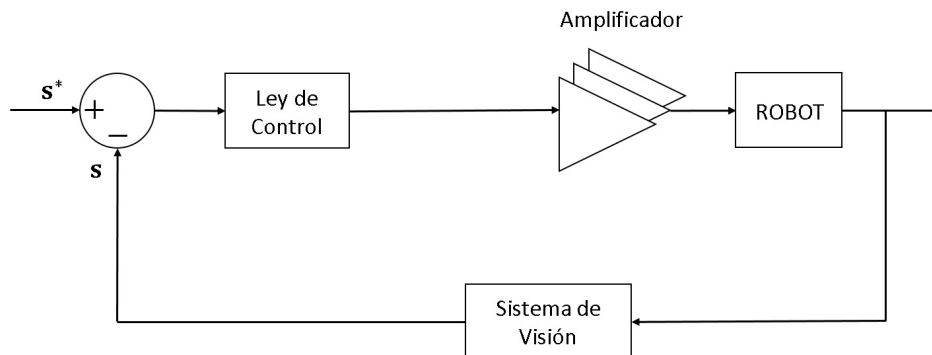


Figura 1.18. Servo visual directo basado en imagen (modificado de [29]).

Para mayores detalles acerca de los controles de Servo Visual y algunos ejemplos de cómo realizar tareas de posicionamiento, ver por ejemplo [29].

1.5.2. Manipulación en Espacio de Cámara

La técnica de control basada en visión Manipulación en Espacio de Cámara (CSM por sus siglas en inglés) trabaja estableciendo y definiendo una relación entre las proyecciones de marcas visuales (adheridas al efector final del robot) vistas por al menos dos cámaras y la configuración interna de las juntas del robot. La técnica CSM emplea un modelo simplificado de cámara denominado modelo ortográfico [34]. Este modelo es válido cuando las cámaras están colocadas a una distancia relativamente alejada del área de trabajo del robot donde el efecto de perspectiva sea mínimo.

En 1987, Skaar [35] desarrolló esta técnica de control y realizó un experimento donde un brazo robótico de un grado de libertad, tenía que atrapar una pelota de ping-pong con movimiento de tiro parabólico en un ambiente de dos dimensiones. Tres años después, Skaar [36] llevó su técnica a un ambiente de tres dimensiones. Además en 1992 [37], Skaar lo implementó a un sistema No-holonómico (sobre ruedas), con un brazo de 1 grado de libertad montado sobre el robot móvil y las cámaras fijas en algún punto en el espacio. Posteriormente, Cárdenas et al. [38] lo implementa a un sistema No-holonómico, pero con las cámaras montadas sobre el robot.

El esquema clásico de CSM o OFL - CSM (“*Orthographic + Flattening Camera Space Manipulation*”) [34,39] trabaja en dos etapas. En la primera etapa, el método usa el modelo de cámara ortográfico [31,40] y el modelo de la cinemática directa del manipulador. En la segunda etapa, usa el ajuste conocido como “*flattening*”, que consiste en incrementar la precisión del modelo ortográfico considerando el efecto de perspectiva [41].

El mapeo que realiza OFL - CSM en la primera etapa está definido por un conjunto de 6 *parámetros de visión*, representados por un sistema de ecuaciones no lineales, lo que conlleva usar métodos iterativos para encontrarlos. Estos parámetros de visión representan la combinación del modelo de cámara ortográfico y la cinemática directa del manipulador. Para determinar los 6 parámetros de visión, se debe minimizar la siguiente ecuación:

$$J(\mathbf{C}) = \sum_{i=1}^m \left\{ \left[u_i - f_x(x'_i, y'_i, z'_i; \mathbf{C}) \right]^2 + \left[v_i - f_y(x'_i, y'_i, z'_i; \mathbf{C}) \right]^2 \right\} \mathbf{w}_i \quad (1.9)$$

donde:

m = número de muestras.

$u_i, v_i =$ coordenadas en el plano de imagen (píxeles) de la i -ésima marca visual.

$x'_i, y'_i, z'_i =$ coordenadas tridimensionales de la i -ésima marca visual al marco de referencia del robot.

$\mathbf{C} = [C_1, \dots, C_6] =$ parámetros de visión.

$\mathbf{W}_i =$ matriz de pesos.

Las relaciones f_x y f_y de la Ec. (1.9) están definidas como:

$$\begin{aligned} f_x(x'_i, y'_i, z'_i; \mathbf{C}) &\equiv b_1(\mathbf{C})x'_i + b_2(\mathbf{C})y'_i + b_3(\mathbf{C})z'_i + b_4(\mathbf{C}) \\ f_y(x'_i, y'_i, z'_i; \mathbf{C}) &\equiv b_5(\mathbf{C})x'_i + b_6(\mathbf{C})y'_i + b_7(\mathbf{C})z'_i + b_8(\mathbf{C}) \end{aligned} \quad (1.10)$$

donde:

$$b_1(\mathbf{C}) = (C_1^2 + C_2^2 - C_3^2 - C_4^2)$$

$$b_2(\mathbf{C}) = 2(C_2C_3 + C_1C_4)$$

$$b_3(\mathbf{C}) = 2(C_2C_4 - C_1C_3)$$

$$b_4(\mathbf{C}) = C_5$$

$$b_5(\mathbf{C}) = 2(C_2C_3 - C_1C_4)$$

$$b_6(\mathbf{C}) = (C_1^2 - C_2^2 + C_3^2 - C_4^2)$$

$$b_7(\mathbf{C}) = 2(C_3C_4 + C_1C_2)$$

$$b_8(\mathbf{C}) = C_6$$

En la segunda etapa se emplea el procedimiento de “*flattening*”, en donde se incrementa la precisión del modelo ortográfico, reduciendo el efecto de perspectiva. El funcionamiento de esta etapa consiste en modificar las muestras en el plano de imagen, de tal manera que sean proyectadas en el plano de la muestra donde se quiere aumentar la precisión.

En 2010 Rendón-Mancha et al [42], crean una variante del esquema clásico de CSM sustituyendo el modelo de cámara ortográfico por el modelo lineal de cámara llamado de orificio (“*Pin-hole*”, Fig. 1.19), con lo que también se omite el proceso de “*flattening*”, dándole el nombre de LCM - CSM (“*Linear Camera Model Camera Space Manipulation*”). La gran diferencia entre OFL - CSM y LCM - CSM, es que este último cuenta con *11 parámetros de visión*, pero al ser un modelo lineal de cámara, el costo computacional para encontrarlos es menor a comparación de OFL - CSM que usa métodos iterativos.

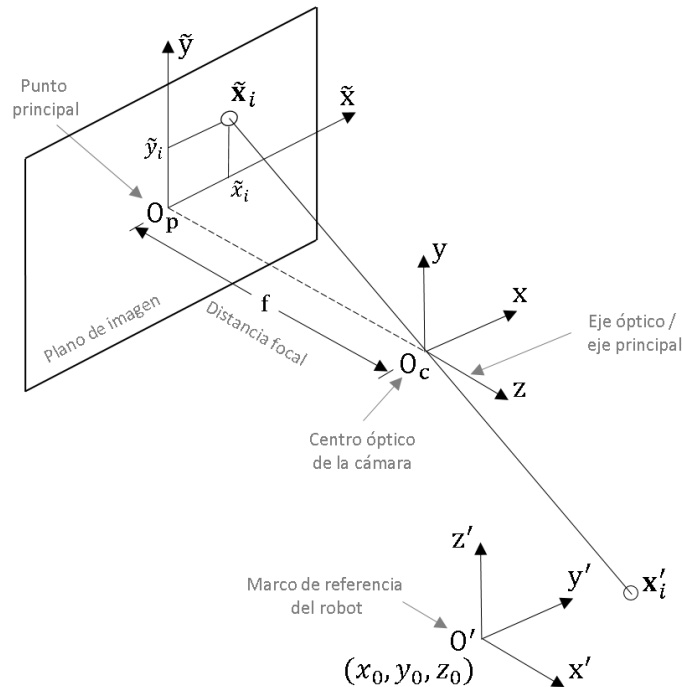


Figura 1.19. Esquema del modelo de cámara de orificio (modificado de [43]).

El presente trabajo se basa en la metodología LCM - CSM, para crear una nueva variante donde se puedan relacionar las variaciones tridimensionales que hay de una marca visual y sus variaciones en el plano de imagen. Se eligió esta metodología por hacer una relación directa entre la posición tridimensional y la posición en el plano de imagen de una marca visual y se facilita una derivación de un modelo de velocidad. Por lo que en la siguiente sección, se describirá la metodología LCM - CSM y de donde surgen sus *11 parámetros de visión*.

1.6. Metodología de control por visión basado en LCM - CSM.

La metodología LCM - CSM (comúnmente llamada CSM Lineal) es una variante de la metodología original OFL - CSM, la cual, sustituye el modelo de cámara ortográfico y el proceso de “*flattening*” por un modelo lineal de cámara conocido como de orificio (“*Pin-hole*”). Además, ambas técnicas fueron inicialmente desarrolladas para manipular robots seriales y no requieren de la calibración de los parámetros geométricos del robot ni la resolución del Jacobiano como Servo Visual, por lo que ambas son atractivas como metodologías de control de robots.

La metodología LCM - CSM establece una relación de la posición tridimensional de varias marcas visuales referidas al marco del robot, con su correspondencia en imágenes adquiridas por al menos dos cámaras. Esta relación está compuesta por un conjunto de parámetros intrínsecos, relacionados al modelo de la cámara, y por parámetros extrínsecos, que relacionan al marco de referencia del robot con el de la cámara. A continuación se muestra como están definidos esta serie de parámetros.

1.6.1. Parámetros Intrínsecos

Los parámetros intrínsecos de un modelo de cámara son aquellos que definen la geometría interna y la óptica de una cámara. Estos determinan la proyección de puntos de un mundo tridimensional (3D) al plano de imagen de una cámara (2D) [31, 32]. Estos son constantes debido a que dependen de la fabricación de los componentes, sin embargo, pueden cambiar por el uso de lentes u objetivos con distancia focal variable o con efectos de ojo de pez.

Para la determinación de los parámetros intrínsecos, se partirá del modelo de cámara de orificio, Fig. 1.20 [31]. El marco de referencia de la cámara está adherido al centro óptico de la misma (O_c, x, y, z), en donde a una distancia focal (f) sobre el eje óptico (z) existe un plano de imagen en el cual se proyecta un punto tridimensional ($\mathbf{x} = [x, y, z]^T$) visto por la cámara.

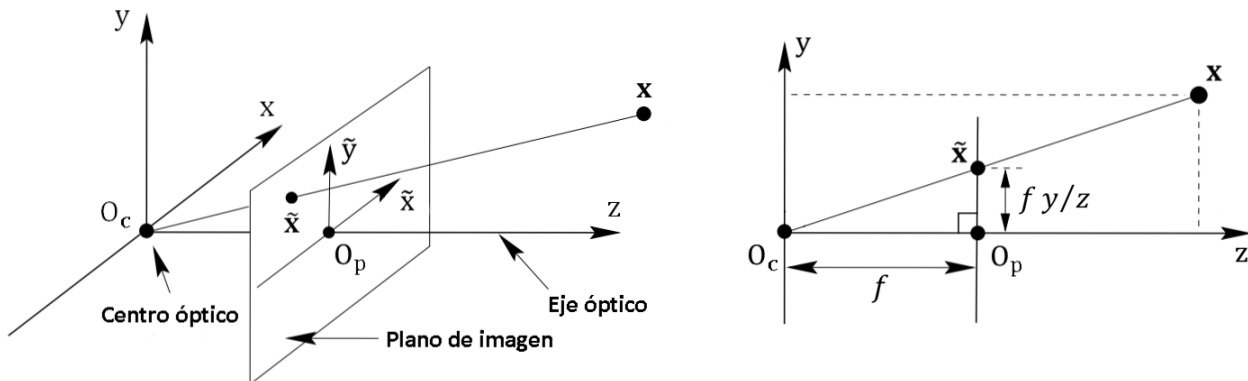


Figura 1.20. Geometría del modelo de cámara de orificio, (modificado de [31]).

Para encontrar el mapeo de un punto tridimensional referido al marco de referencia de la cámara (\mathbf{x}) hacia el plano de imagen ($\tilde{\mathbf{x}} = [\tilde{x}, \tilde{y}]^T$), se usa la regla de los triángulos semejantes quedando la siguiente ecuación:

$$\tilde{x}_i = f \frac{x_i}{z_i}, \quad \tilde{y}_i = f \frac{y_i}{z_i} \quad (1.11)$$

donde i es el i -ésimo punto en el espacio tridimensional con su correspondencia en el plano de imagen. Representando la Ec.1.11 en su forma matricial homogénea se tiene:

$$\rho \begin{bmatrix} \tilde{x}_i \\ \tilde{y}_i \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix} \quad (1.12)$$

donde ρ representa un factor de escala ($1/z_i$). La Ec. (1.12) está dada a un marco de referencia adherido al punto O_p como se observa en la Fig. 1.21 [31].

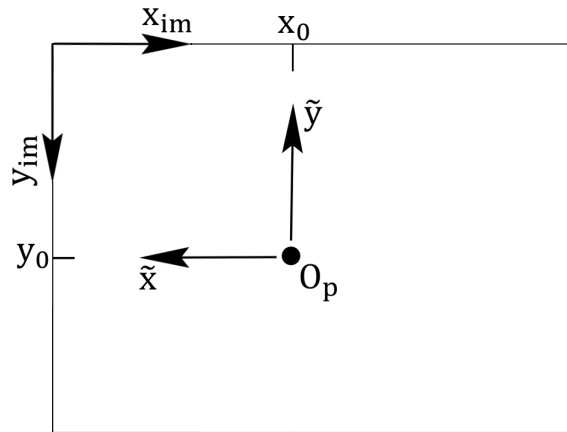


Figura 1.21. Sistema de coordenadas de la imagen (x_{im}, y_{im}) y el plano de imagen (\tilde{x}, \tilde{y}) (modificado de [31]).

Sin embargo, cuando se trabaja con imágenes digitales, el conteo de píxeles comienza en la esquina superior izquierda, por lo que se tiene que hacer un cambio de marco de referencia en el plano de imagen, Fig. 1.21 [31]. Para esto, se debe incluir una compensación x_0, y_0 y queda expresado como:

$$\rho \begin{bmatrix} \tilde{x}_i \\ \tilde{y}_i \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & x_0 & 0 \\ 0 & f & y_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix} \quad (1.13)$$

Si los parámetros f , x_0 y y_0 están expresados en unidades de longitud, entonces se requieren factores (m_x, m_y) que permitan la conversión de unidades de longitud a pixeles. Los factores m_x y m_y representan el número de pixeles por unidad de longitud en las direcciones x y y respectivamente. Incluyendo estos factores en la Ec. (1.13) se tiene:

$$\rho \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_x & 0 & u_0 & 0 \\ 0 & \alpha_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix} \quad (1.14)$$

donde u_i y v_i son las coordenadas del punto proyecto en pixeles, $\alpha_x = fm_x$, $\alpha_y = fm_y$ representan la distancia focal de la cámara en términos de pixeles en la dirección x y y respectivamente. Similarmente, $u_0 = x_0m_x$ y $v_0 = y_0m_y$ son las coordenadas en pixeles del punto O_p , que es la intersección del eje óptico con el plano de imagen en el marco de referencia correspondiente. De esta forma, la Ec. (1.14) representa el mapeo de un punto tridimensional, expresado en el marco de referencia de la cámara, a su proyección en la imagen expresada en pixeles.

La submatriz 3x3 de 1.14 es denominada matriz de calibración (K). En las cámaras CCD, puede llegar el caso en que los pixeles no sean totalmente rectangulares, por lo que se agrega un término s el cual toma valores de 0 o 1. Incluyendo s , la matriz K viene dada por 1.15 y contiene los 5 parámetros intrínsecos de la cámara.

$$K = \begin{bmatrix} \alpha_x & s & u_0 \\ 0 & \alpha_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1.15)$$

1.6.2. Parámetros Extrínsecos

Los parámetros extrínsecos relacionan un sistema de coordenadas arbitrario (O', x', y', z') con el sistema coordenado de la cámara (O_c), donde se incluyen la rotación (\mathbf{R}) y traslación (\mathbf{t}) entre los mismos, Fig. 1.16 [31].

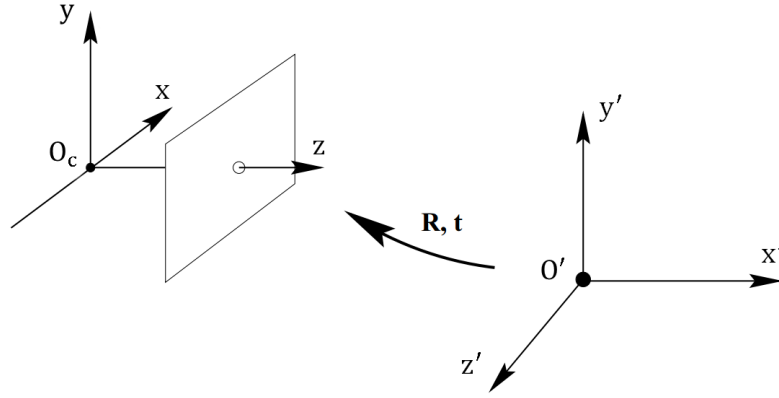


Figura 1.22. Rotación y traslación entre el sistema coordenado mundial y el sistema coordenado de la cámara (modificado de [31]).

Para lograr el cambio entre ambos sistemas, se utiliza una matriz de transformación homogénea (\mathbf{H}) que contiene la matriz de rotación (\mathbf{R}) y el vector de traslación (\mathbf{t}). Esta matriz homogénea va a estar conformada por 6 parámetros, 3 de ellos relacionados a los ángulos de rotación sobre cada eje y los otros 3 relacionados con la traslación. Así, la matriz de transformación homogénea quedaría como:

$$\mathbf{H} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \quad (1.16)$$

donde r_{ij} representan los elementos de la matriz \mathbf{R} y t_i los elementos del vector de traslación \mathbf{t} . Es importante mencionar, que a pesar de que la matriz \mathbf{R} está constituida por 9 elementos, estos no son independientes y solamente se requieren 3 variables para determinarlos. De esta manera la Ec. (1.14), incluyendo el factor s , queda definida como:

$$\rho \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_x & s & u_0 \\ 0 & \alpha_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} x'_i \\ y'_i \\ z'_i \\ 1 \end{bmatrix} \quad (1.17)$$

donde u_i, v_i representan las coordenadas en píxeles de un punto visto en el plano de imagen y $[x', y', z']$ representan un punto en el espacio tridimensional visto por la cámara y referenciado a un sistema de coordenadas global (en este caso, referido al sistema coordenado del robot).

La Ec. (1.17) representa el modelo completo de cámara de orificio, compuesto por 5 parámetros intrínsecos (matriz 3x3) y 6 parámetros extrínsecos (matriz 3x4). Teniéndose en total 11 parámetros o grados libertad. Al multiplicar la matriz de parámetros intrínsecos y extrínsecos se obtiene una matriz 3x4 conocida como *matriz de proyección de cámara* [42]:

$$\rho \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} x'_i \\ y'_i \\ z'_i \\ 1 \end{bmatrix} \quad (1.18)$$

1.7. Conclusiones del capítulo 1

En este capítulo se mostraron los dos tipos de robots industriales más comunes, el robot serial y el robot paralelo. Este último robot cuenta con características favorables que lo hacen atractivo como precisión, rigidez, altas velocidades y aceleraciones y mayor relación carga/peso para su implementación en sectores industriales, medicinales y educativos. Sin embargo, el robot paralelo cuenta aún con problemas abiertos identificados en la literatura como es la obtención de su cinemática directa, dinámica, análisis de singularidades, entre otros. Una forma para la solución de los problemas anteriores, es la implementación de técnicas de control basadas en visión por computadora.

Las técnicas de control por visión sustituyen los esquemas tradicionales de control de robots, en donde previamente se cargan una serie de movimientos pre-programados. Además, ayudan a aumentar la precisión en tareas de posicionamiento estático y en seguimiento de objetivos en movimiento. Existen varias técnicas de control por visión, entre las más comunes se encuentran Manipulación en Espacio de Cámara (CSM) y Servo Visual.

Servo Visual se basa en el control de lazo cerrado y la integración en tiempo real de las informaciones visuales obtenidas de una a varias cámaras, además, le permite controlar la posición y velocidad del robot por lo que requiere el uso de la Cinemática y el Jacobiano del robot. CSM se basa en el control de lazo abierto con técnicas de estimación de parámetros para estimar la relación tridimensional de varias marcas visuales con su correspondencia en imágenes tomadas por al menos dos cámaras, sin embargo, al hacer control punto a punto no requiere del Jacobiano del robot y evita el manejo de las singularidades del mismo.

Ambas técnicas de control por visión pueden ser implementadas en el robot paralelo tipo Delta. Son escasos los trabajos que reportan el desarrollo de controles basados en las técnicas Servo Visual y CSM aplicados a robots paralelos en comparación a robots seriales. Además, no existe alguna variante de CSM que controle la posición robot usando comandos de velocidad. Cabe resaltar que un control basado en velocidad, realiza movimientos suaves en la trayectoria de posicionamiento en objetos estáticos o en movimiento. Con esto, se puede justificar uno de los aportes del presente trabajo, con el desarrollo de una nueva variante de CSM para el control en velocidad de un robot paralelo tipo Delta.

Implementaciones de controles basados en la metodología Manipulación en Espacio de Cámara para un robot paralelo tipo Delta

En este capítulo se describe la implementación de la metodología LCM - CSM, que ya había sido reportado en trabajos previos [44,45], pero modificando la trayectoria pre-planeada (conjunto de n puntos previo a una tarea de posicionamiento para inicializar los parámetros de visión) a un volumen más extenso dentro del espacio de trabajo del robot Delta. Esto último para corroborar si se presenta alguna mejora en la precisión en tareas de posicionamiento. Además, se desarrolla una variante de la metodología VW - CSM (“*Varying Weight Camera Space Manipulation*”) que originalmente fue desarrollada bajo el concepto de OFL - CSM [46] descrito en el capítulo anterior, pero ahora usando el modelo lineal de cámara de orificio.

2.1. Esquemas de control general de la metodología CSM

Como se explicó anteriormente, la metodología de control CSM y sus variantes tradicionales definen una relación entre las proyecciones de marcas visuales (adheridas al efector final del robot) vistas por al menos dos cámaras y la configuración interna de las juntas del robot. La metodología clásica OFL - CSM [34] está basada en el modelo cinemático del robot, en el modelo de cámara ortográfico y en una etapa posterior de ajuste conocida como “*flattening*”. Este esquema de CSM trabaja con un modelo no lineal de cámara, por lo que se requiere de métodos iterativos para encontrar sus 6 parámetros de visión. El ajuste conocido como “*flattening*”, considera el efecto de perspectiva para aumentar la precisión del modelo de cámara ortográfico.

Antes de realizar una tarea de posicionamiento, se tiene que inicializar los 6 parámetros de visión en una etapa previa conocida como “*Pre-plan*”. Esta etapa consiste en tomar al menos 6 muestras aleatorias dentro del espacio de trabajo del robot (muestras globales). Además del uso de “*flattening*”, se toman muestras alrededor del punto objetivo (muestras locales) para darles mayor importancia y así estimar unos parámetros de visión más confiables [34].

La metodología LCM - CSM [42] está basada en el modelo cinemático y el modelo de cámara orificio. Como se mencionó anteriormente, este esquema de control trabaja bajo un modelo lineal de cámara, por lo que requiere de menor costo computacional para calcular sus 11 parámetros de visión. Al igual que OFL - CSM, LCM - CSM requiere de tomar muestras locales cuando el robot se va acercando al punto objetivo para aumentar la precisión en el posicionamiento.

La metodología VW - CSM fue desarrollada por Liu et al en 2014 [46] y es una variante directa de la técnica OFL - CSM. La metodología VW - CSM consiste en variar la importancia de las muestras globales obtenidas en el “*Pre-plan*” en relación a la distancia euclidiana con el punto objetivo en el plano de imagen. A pesar de que las metodologías OFL - CSM, VW - CSM y LCM - CSM cuentan con su propio algoritmo para la determinación del movimiento requerido del robot, comparten el mismo esquema de control como se muestra en la siguiente figura.

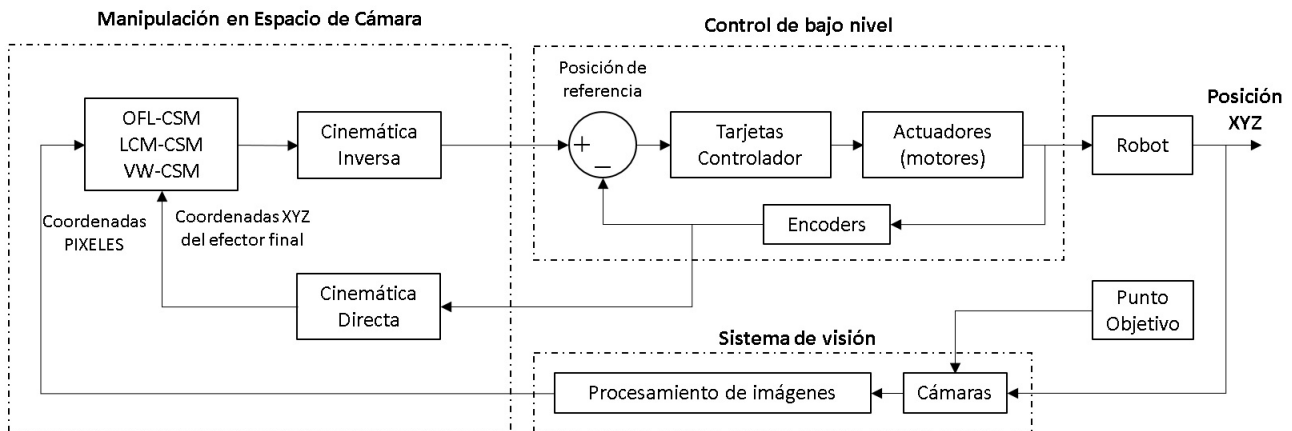


Figura 2.1. Esquema de control general de la metodología CSM (modificado de [45]).

En la Fig. 2.1 se observa que todo el esquema de control está compuesto por tres secciones:

- **Control de bajo nivel**, está compuesto por los encoders, los motores y el controlador. Este último contiene internamente un control PID para asegurar que se cumpla con la posición de referencia determinada por la metodología CSM.
- **Sistema de visión**, está compuesto por 2 o más cámaras que observan tanto el robot como el objetivo. También, está constituido por el procesamiento de las imágenes capturadas, para extraer las características necesarias para el uso en el algoritmo de CSM.
- **Manipulación en Espacio de Cámara**, esta compuesto por el algoritmo de OFL - CSM, VW - CSM o LCM - CSM según sea el caso a utilizar, además se incluye la cinemática directa e inversa del robot que son necesarias para el uso de estas mismas metodologías.

2.2. Implementación de la metodología LCM - CSM para el control del robot Delta

En [45] se implementó un control LCM - CSM para el robot Delta. En esta sección se expone una implementación de este control para el robot Delta, en donde el “*Pre-plan*” se extiende a casi todo el espacio de trabajo del robot y con un número mayor de muestras. Esta implementación servirá de referencia para múltiples propósitos: por un lado, permite utilizar prácticamente todo el espacio de trabajo del robot para tareas de posicionamiento, lo cual es consistente con el diseño de cualquier robot. Por otro lado, la implementación servirá de referencia para las metodologías VW - CSM y la metodología de control objeto de este trabajo.

2.2.1. Determinación de los 11 parámetros de visión

Para la determinación de los 11 parámetros de visión, existen varias metodologías como por ejemplo la estimación lineal de parámetros [47] o SVD (“*Singular Value Decomposition*”) [31] por sus siglas en inglés. Para fines de este trabajo se usará la metodología de estimación lineal, que se describirá más adelante.

A partir de la Ec. (1.18), se puede despejar u_i y v_i de la siguiente manera:

$$\begin{aligned} u_i &= \frac{p_{11}x'_i + p_{12}y'_i + p_{13}z'_i + p_{14}}{p_{31}x'_i + p_{32}y'_i + p_{33}z'_i + p_{34}} \\ v_i &= \frac{p_{21}x'_i + p_{22}y'_i + p_{23}z'_i + p_{24}}{p_{31}x'_i + p_{32}y'_i + p_{33}z'_i + p_{34}} \end{aligned} \quad (2.1)$$

Como se mencionó anteriormente, se tienen 11 grados de libertad dentro de los parámetros intrínsecos y extrínsecos, sin embargo, la matriz de proyección de cámara cuenta con 12 elementos (matriz 3x4). En este caso, se puede elegir entre cualquiera de estos elementos para representarlo como un factor de escala. Por comodidad se elige el último elemento p_{34} como factor de escala y la matriz de parámetros de visión podrá dividirse entre este factor. De esta manera, se obtiene un sistema lineal de la siguiente forma:

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} x'_i & y'_i & z'_i & 1 & 0 & 0 & 0 & 0 & -u_i x'_i & -u_i y'_i & -u_i z'_i \\ 0 & 0 & 0 & 0 & x'_i & y'_i & z'_i & 1 & -v_i x'_i & -v_i y'_i & -v_i z'_i \end{bmatrix} \begin{bmatrix} p_{11} \\ \vdots \\ p_{33} \end{bmatrix} \quad (2.2)$$

donde los elementos $p_{11} \dots p_{33}$ ya se encuentran normalizados con respecto a p_{34} , por lo que a partir de aquí se conocerán como los 11 parámetros de visión. Para encontrar la solución de los parámetros de visión en la Ec. (2.2), se requieren de al menos 6 muestras de un punto asociado el efector final en posiciones arbitrarias y conocidas, ya que cada una proporciona 2 ecuaciones al sistema debido al uso de dos cámaras. En la práctica, se recomienda utilizar más de 6 muestras para tener un sistema sobre-determinado y lograr así una mejor estimación de los parámetros de visión para cada cámara.

2.2.2. Estimación lineal de parámetros

Sea un conjunto de m valores medidos de un modelo descrito por un comportamiento lineal [47], cuyo modelo matemático puede ser por ejemplo la Ec. (2.2) y está compuesto por n parámetros ($n = 11$ para la Ec. (2.2)). Sin embargo, al sustituir los parámetros al modelo previamente descrito, éste no producirá un acuerdo exacto con las mediciones ($[u_i, v_i, \dots, u_n, v_n]^T$ para la Ec. (2.2)) por lo que generará una diferencia. Además, esta diferencia o error también puede originarse debido a una selección errónea de los parámetros, a errores en las mediciones o a errores en el modelo. Este modelo en forma matricial queda representado por la siguiente ecuación:

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e} \quad (2.3)$$

donde:

$\mathbf{y} = [y_1, \dots, y_m]^T$ = son los valores medidos u observados.

$\mathbf{e} = [e_1, \dots, e_m]^T$ = son los errores residuales.

$\mathbf{x} = [x_1, \dots, x_n]^T$ = son los parámetros incluidos en el modelo.

$$\mathbf{A} = \begin{bmatrix} a_1^1 & \cdots & a_n^1 \\ \vdots & \ddots & \vdots \\ a_1^m & \cdots & a_n^m \end{bmatrix}$$

Con base en el principio de mínimos cuadrados de Gauss, se encuentra un valor particular o estimado de los parámetros ($\hat{\mathbf{x}}$) que minimice la suma de los cuadrados de los errores residuales, donde:

$$\text{mín } J = \mathbf{e}^T \mathbf{e} \quad (2.4)$$

De la Ec. (2.3), se obtiene $\mathbf{e} = \mathbf{y} - \mathbf{A}\hat{\mathbf{x}}$ y entonces:

$$J = \mathbf{y}^T \mathbf{y} - 2\mathbf{y}^T \mathbf{A}\hat{\mathbf{x}} + \hat{\mathbf{x}}^T \mathbf{A}^T \mathbf{A}\hat{\mathbf{x}} \quad (2.5)$$

La minimización de esta función requiere que se satisfagan las siguientes condiciones:

o Condición necesaria:

$$\nabla_{\hat{\mathbf{x}}} J = -2\mathbf{A}^T \mathbf{y} + 2\mathbf{A}^T \mathbf{A}\hat{\mathbf{x}} = 0 \quad (2.6)$$

o Condición suficiente:

$$\nabla_{\hat{\mathbf{x}}}^2 J = 2\mathbf{A}^T \mathbf{A} \quad (2.7)$$

A partir de la condición necesaria (Ec. (2.6)), se puede obtener una solución para el valor óptimo de los parámetros:

$$\hat{\mathbf{x}} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{y} \quad (2.8)$$

La Ec. (2.8) puede determinar los valores óptimos para $\hat{\mathbf{x}}$, dándole la misma importancia a los valores medidos $[y_1, \dots, y_m]^T$. Sin embargo esto puede diferir en un ámbito real, ya que las mediciones pueden no tener el mismo grado de precisión. Para seleccionar el peso que tendrá cada muestra, se define una matriz diagonal \mathbf{W} que se muestra a continuación:

$$\mathbf{W} = \begin{bmatrix} w_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & w_m \end{bmatrix} \quad (2.9)$$

en donde el peso w_i , de la i -ésima observación, dependerá de la importancia que le asignemos a ese punto (por ejemplo, todos los puntos para la determinación de parámetros a través del “Pre-plan” en la metodología CSM). Si \mathbf{W} es incluido en la Ec. (2.8) se obtiene:

$$\hat{\mathbf{x}} = (\mathbf{A}^T \mathbf{W} \mathbf{A})^{-1} \mathbf{A}^T \mathbf{W} \mathbf{y} \quad (2.10)$$

2.2.3. Determinación de la posición tridimensional

Para determinar la posición tridimensional de un punto a partir su proyección en la imagen de al menos dos cámaras y sus respectivos parámetros de visión, se puede obtener un sistema lineal de la Ec. (2.1) para calcular un vector de posición (x'_i, y'_i, z'_i) de la siguiente manera:

$$\begin{bmatrix} u1_i - p1_{14} \\ v1_i - p1_{24} \\ u2_i - p2_{14} \\ v2_i - p2_{24} \end{bmatrix} = \begin{bmatrix} p1_{11} - p1_{31}u_i & p1_{12} - p1_{32}u_i & p1_{13} - p1_{33}u_i \\ p1_{21} - p1_{31}v_i & p1_{22} - p1_{32}v_i & p1_{23} - p1_{33}v_i \\ p2_{11} - p2_{31}u_i & p2_{12} - p2_{32}u_i & p2_{13} - p2_{33}u_i \\ p2_{21} - p2_{31}v_i & p2_{22} - p2_{32}v_i & p2_{23} - p2_{33}v_i \end{bmatrix} \begin{bmatrix} x'_i \\ y'_i \\ z'_i \end{bmatrix} \quad (2.11)$$

donde $p1$ y $p2$ son los parámetros de visión de la cámara 1 y cámara 2 respectivamente. Para la solución de la Ec. (2.11), se usará la Pseudo-inversa de Moore-Penrose para la inversión de matrices no cuadradas [48].

Para realizar el posicionamiento del robot en un punto objetivo, se obtiene la posición en pixeles del objetivo para hacer el cálculo de un vector de posición tridimensional (Ec. (2.11)). Como se mencionó anteriormente, si queremos aumentar la precisión en el posicionamiento se tendrá que tomar muestras alrededor del punto objetivo (muestras locales) para incluirlas en el modelo (Ec. (2.2)) y así poder actualizar y/o refinar los parámetros de visión .

Para la adquisición de las muestras locales, se propone crear una *trayectoria de transición* discretizada en n puntos, entre el objetivo y un punto relativamente cercano a éste. En cada aproximación o punto de la discretización, se tiene que hacer el cálculo de la configuración articular del robot y de la localización del centroide de las marcas visuales (LEDs en este trabajo) asociadas al efector final. En cada una de estas muestras, se le da mayor importancia o peso modificando su respectivo valor en la matriz diagonal w_i de la Ec. (2.10). Con esto, en cada punto de la discretización o paso de la trayectoria de transición se va aumentando la confiabilidad de los parámetros de visión hasta llegar al punto objetivo.

Todo lo visto en esta sección está plasmado en el diagrama de flujo de la Fig. 2.2. Lo primero que se realiza es la etapa de “*Pre-plan*” para calcular los parámetros iniciales de visión, Ec. (2.2).

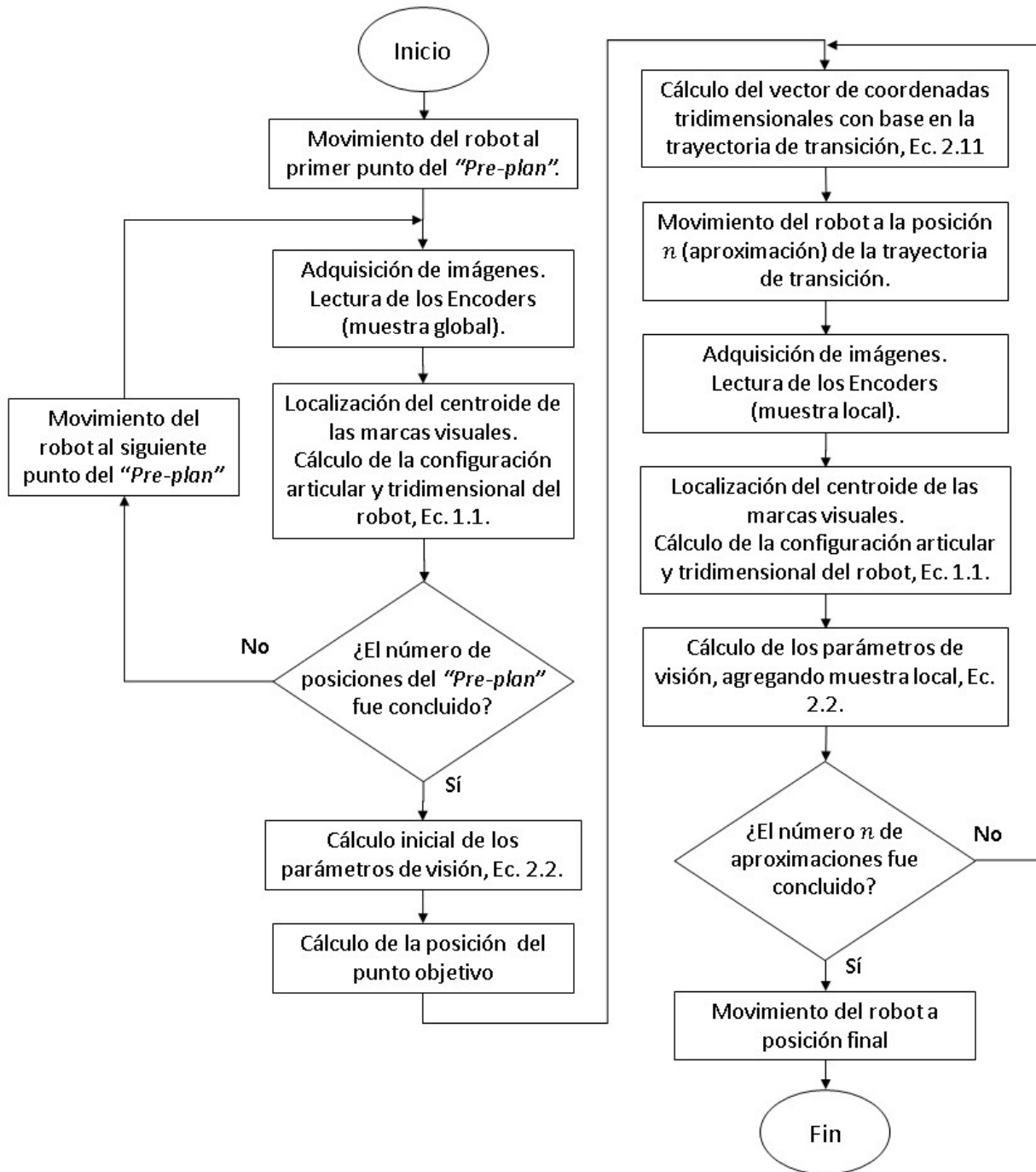


Figura 2.2. Diagrama de flujo de la metodología LCM - CSM (tomado de [45]).

2.3. Desarrollo de la metodología VW - CSM de un modelo lineal de cámara para el control del robot Delta

VW - CSM es una metodología para el control de robots que originalmente fue desarrollado usando el esquema OFL - CSM. La idea detrás de esta metodología, es variar el peso o importancia de las muestras adquiridas en el “*Pre-plan*” (muestras globales), con relación a la distancia euclidiana hacia el punto objetivo. Esta forma de variar el peso en las muestras, le permite al esquema VW - CSM acelerar la convergencia del algoritmo para realizar tareas de posicionamiento más rápidas.

En el trabajo de Liu et al [46], se proponen 4 fórmulas (w_1, w_2, w_3, w_4) para variar el peso las cuales se enuncian a continuación:

$$w_1 = k/l \tag{2.12}$$

$$w_2 = k + 1/l \tag{2.13}$$

$$w_3 = 1/(l)^k \tag{2.14}$$

$$w_4 = k/(l)^k \tag{2.15}$$

donde l es la distancia euclidiana calculada y k es un parámetro de variación que agrega mayor peso a los puntos del “*Pre-plan*” más cercanos al punto objetivo.

Para implementar la metodología VW - CSM usando el esquema LCM -CSM, se usará la Ec. (2.12) dado que, de acuerdo con [46], presenta un error de posicionamiento más estable al variar el valor del parámetro k . Los elementos de la matriz \mathbf{W} de la Ec. (2.9), se calculan usando la Ec. (2.16) para todo $i = 1, \dots, m$, donde m es el número de muestras tomadas en el “*Pre-plan*”:

$$w_i = k/l_i \tag{2.16}$$

A pesar de que en [46] agregan nuevas muestras en una trayectoria pre-planeada antes de llegar al objetivo, en la presente implementación solo se tomarán las muestras globales dando lugar a que el robot se posicione directamente desde una posición arbitraria al objetivo.

El algoritmo de esta metodología (VW - CSM de un modelo de cámara lineal) se puede ver en un diagrama de flujo mostrado en la Fig. 2.3. Donde lo primero que se realiza es la etapa de “Pre-plan” para calcular los parámetros iniciales de visión 2.2.

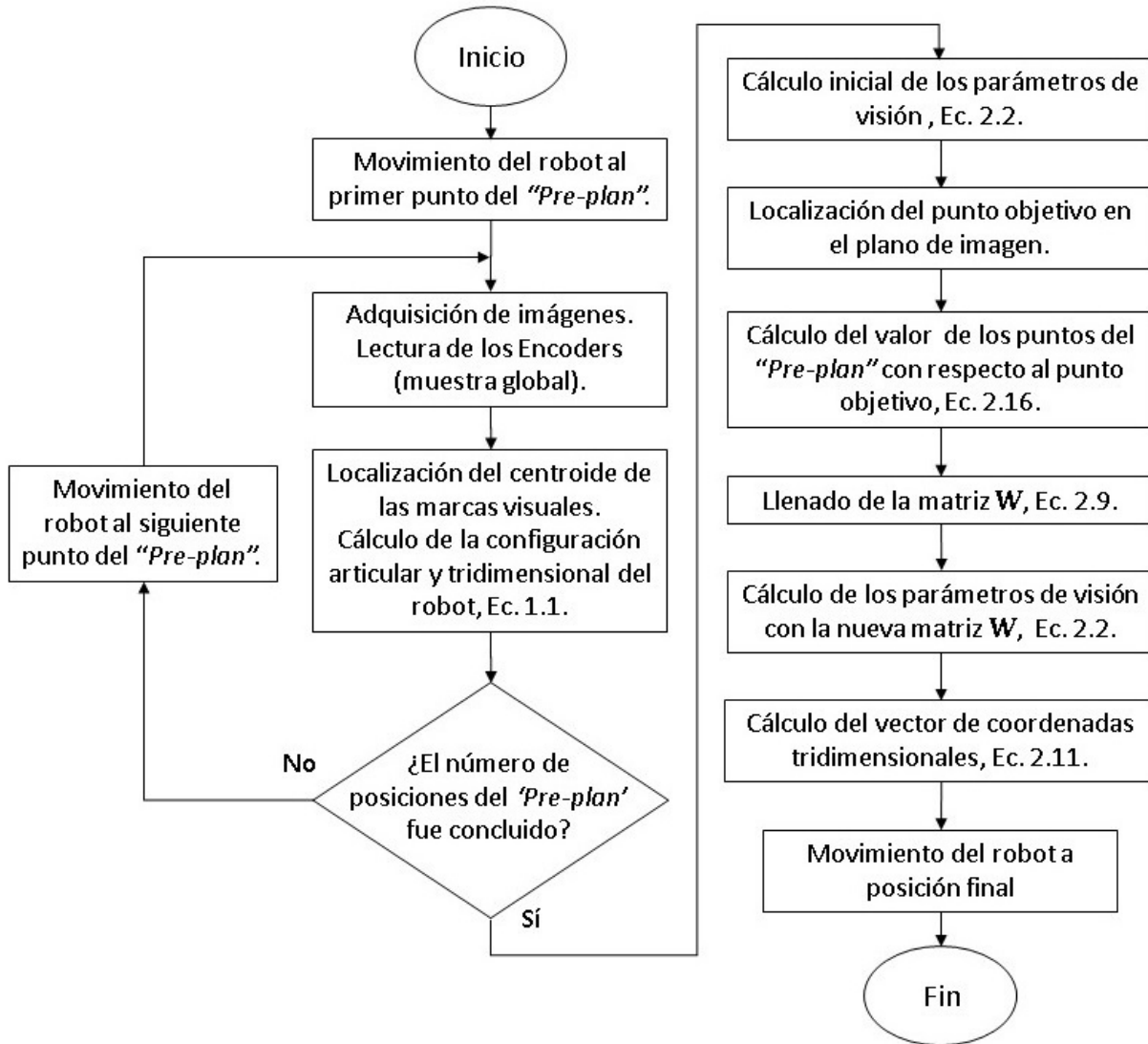


Figura 2.3. Diagrama de flujo de la metodología VW - CSM de un modelo de cámara lineal.

Para realizar una tarea de posicionamiento, se obtiene la posición en pixeles del punto objetivo para el cálculo de la distancia euclidiana con cada una de las muestras globales. Posteriormente, se evalúa la Ec. (2.16) para rellenar los elementos de la matriz \mathbf{W} . Con esto, se vuelven a calcular los parámetros de visión (Ec. (2.2)) y así encontrar el vector de coordenadas tridimensionales del punto objetivo (Ec. (2.11)). Por último, se posiciona el robot desde cualquier punto arbitrario en el espacio de trabajo hacia la posición calculada.

2.4. Conclusiones del capítulo 2

En este capítulo se describe la implementación de la metodología LCM - CSM. Además, se desarrolló la metodología VW - CSM extendiéndola a un modelo lineal de cámara orificio, ya que originalmente fue desarrollada para la metodología OFL - CSM para el control de un robot serial. Ambas metodologías se aplicarán para el control de un robot Delta, ayudando a resolver algunos problemas aún abiertos en la literatura para el control de los robots paralelos.

La metodología LCM - CSM usa un modelo de cámara lineal conocido como orificio, el cual hace una transformación lineal entre un punto tridimensional referenciado a un marco global con su respectiva proyección en el plano de imagen de una cámara. Para aumentar la precisión en las tareas, toma muestras locales (muestras adicionales tomadas conforme el robot se acerca al punto objetivo) y las cuales son utilizadas para mejorar la estimación de los parámetros de visión. Esto se logra asignando un peso o importancia mayor a las muestras locales utilizando una matriz de pesos.

En cambio, la implementación de VW - CSM de un modelo lineal de cámara elimina esta necesidad de tomar muestras locales y toma los puntos previamente agregados en el “*Pre-plan*” (muestras globales) y varía su importancia o peso según su distancia euclidiana respecto al punto objetivo. Las metodologías LCM - CSM y VW - CSM de un modelo lineal de cámara, se implementaron en el robot paralelo tipo Delta, en el que se realizaron tareas de posicionamiento dentro del espacio de trabajo efectivo cubierto por el “*Pre-plan*” propuesto.

Desarrollo de una técnica de control en velocidad basado en la metodología Manipulación en Espacio de Cámara para un robot paralelo tipo Delta

En este capítulo se desarrolla una nueva técnica basada en visión, para el control en velocidad de robots, aplicado al robot paralelo tipo Delta. La formulación se basa en la metodología de LCM - CSM partiendo de la matriz de proyección de cámara. La metodología CSM y sus variantes, son técnicas de lazo abierto diseñadas para el control de robots, para manipular objetos estáticos usando comandos de posición con técnicas de estimación. La técnica propuesta trabaja bajo el esquema de control de un sistema de lazo cerrado y puede funcionar tanto para objetos estáticos como en movimiento usando comandos de velocidad. Se presenta una forma de estimar la velocidad de objetos en movimiento con ayuda del Filtro de Kalman lo cual permitirá compensar esa velocidad en la ley de control propuesta. Además, se aborda un análisis de estabilidad del sistema controlado.

3.1. Matriz CSM para velocidades

Para el desarrollo de la ley de control CSM basado en velocidad, se va a partir del modelo de LCM - CSM. Derivando la Ec. (1.18) se puede obtener una matriz 2×3 que relaciona las velocidades de un punto que se desplaza en el espacio euclidiano tridimensional (mm/s) y las velocidades en el plano de imagen (píxeles/s) asociadas a la proyección del punto en la imagen:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \end{bmatrix} \begin{bmatrix} \dot{x}' \\ \dot{y}' \\ \dot{z}' \end{bmatrix} \quad (3.1)$$

En la Ec. (3.1), se observa que es un sistema de ecuaciones indeterminado, por lo que se si se usan al menos dos cámaras (condición necesaria para usar CSM y sus variantes) queda un sistema sobre-determinado:

$$\begin{bmatrix} \dot{u}1 \\ \dot{v}1 \\ \dot{u}2 \\ \dot{v}2 \end{bmatrix} = \begin{bmatrix} p1_{11} & p1_{12} & p1_{13} \\ p1_{21} & p1_{22} & p1_{23} \\ p2_{11} & p2_{12} & p2_{13} \\ p2_{21} & p2_{22} & p2_{23} \end{bmatrix} \begin{bmatrix} \dot{x}' \\ \dot{y}' \\ \dot{z}' \end{bmatrix} \quad (3.2)$$

donde $p1$ y $p2$ son los parámetros de visión de la cámara 1 y la cámara 2 respectivamente. La matriz resultante la llamaremos *CSM para velocidades* (J_{csm}):

$$J_{csm} = \begin{bmatrix} p1_{11} & p1_{12} & p1_{13} \\ p1_{21} & p1_{22} & p1_{23} \\ p2_{11} & p2_{12} & p2_{13} \\ p2_{21} & p2_{22} & p2_{23} \end{bmatrix} \quad (3.3)$$

3.2. Ley de Control

Consideremos una tarea donde se desea que el efector final de un robot se posicione en un punto objetivo (por ejemplo tareas “*pick and place*”). Esta tarea se puede expresar en términos de la distancia o error de posicionamiento de la siguiente manera:

$$\mathbf{e} = \mathbf{s}^* - \mathbf{s} \quad (3.4)$$

donde \mathbf{e} es el vector de error de posicionamiento, $\mathbf{s}^* = [u1^* \ v1^* \ u2^* \ v2^*]$ representa el punto objetivo (puntero laser, LEDs, etc.) y $\mathbf{s} = [u1 \ v1 \ u2 \ v2]$ representa un punto adherido al efector final con las componentes de todos los vectores expresados en pixeles. Derivando la Ec. (3.4) y considerando la matriz CSM para velocidades (Ec. (3.3)) se tiene:

$$\dot{\mathbf{e}} = \dot{\mathbf{s}}^* - \dot{\mathbf{s}} = \dot{\mathbf{s}}^* - J_{csm}\dot{\mathbf{x}} \quad (3.5)$$

donde $\dot{\mathbf{e}}$ es la variación del error en pixeles con respecto al tiempo (pixeles/s), $\dot{\mathbf{s}}^*$ es la velocidad del punto objetivo en el plano de imagen (pixeles/s), $\dot{\mathbf{s}}$ es la velocidad del efector final en el plano de imagen (pixeles/s).

Considerando el Jacobiano del robot Delta (Ec. (1.5) - 1.7), la Ec. (3.5) se puede reescribir como:

$$\dot{\mathbf{e}} = \dot{\mathbf{s}}^* - J_{csm}J_x^{-1}J_q\dot{\mathbf{q}}. \quad (3.6)$$

Definimos como comportamiento deseado del sistema controlado en lazo cerrado:

$$\dot{\mathbf{e}} = -\mathbf{G}\mathbf{e} \quad (3.7)$$

donde \mathbf{G} es una matriz de ganancias. Si \mathbf{G} es una matriz definida positiva, entonces el error decrecerá exponencialmente a cero. Lo anterior implica que el vector que denota la posición del efector final en la imagen tenderá exponencialmente al vector que representa el punto objetivo. De manera similar, en el espacio euclidiano, el efector final alcanzará la posición deseada.

Sustituyendo el sistema controlado $\dot{\mathbf{e}} = -\mathbf{G}\mathbf{e}$ en la Ec. (3.6), se tiene:

$$-\mathbf{G}\mathbf{e} = \dot{\mathbf{s}}^* - J_{csm}J_x^{-1}J_q\dot{\mathbf{q}}. \quad (3.8)$$

A partir de la Ec. (3.4) y (3.6), se puede definir la ley de control buscada despejando $\dot{\mathbf{q}}$:

$$\dot{\mathbf{q}} = J_q^{-1}J_xJ_{csm}^\dagger [\mathbf{G}(\mathbf{s}^* - \mathbf{s}) + \dot{\mathbf{s}}^*] \quad (3.9)$$

donde la matriz J_{csm}^\dagger es la Pseudoinversa de Moore-Penrose de la Ec. (3.3). Esta nueva ley de control permite al robot realizar posicionamiento o seguimiento de un objetivo con trayectorias suaves y rápidas, esto debido a que el comportamiento deseado del sistema en lazo cerrado (Ec. (3.7)) decrecerá exponencialmente a cero el error en pixeles.

Si se propone una tarea de posicionamiento estático el término $\dot{\mathbf{s}}^*$ de la Ec. (3.9) se vuelve 0. Cabe mencionar que si no se conoce este último término, la ley de control propuesta (Ec. (3.9)) puede hacer que el robot siga a un objeto en movimiento pero con un cierto desfase. Por lo anterior, se requiere de alguna técnica de estimación para calcular el término $\dot{\mathbf{s}}^*$ en trayectorias definidas que involucran movimiento, ya que el sensor (cámara CCD) usado no nos puede dar ese tipo de información.

3.3. Marca visual virtual

En la sección previa se determinó la nueva ley de control CSM en velocidades, donde su premisa es controlar un robot hacia la posición de un punto objetivo usando comandos de velocidad (Ec. (3.4)). Para la implementación en el robot Delta, esto se puede volver complicado ya que las marcas visuales, tanto en el punto objetivo como en el efector final del robot Delta, se encuentran en posiciones (Fig. 3.1) donde al realizar una tarea de posicionamiento puede provocar *colisiones*.

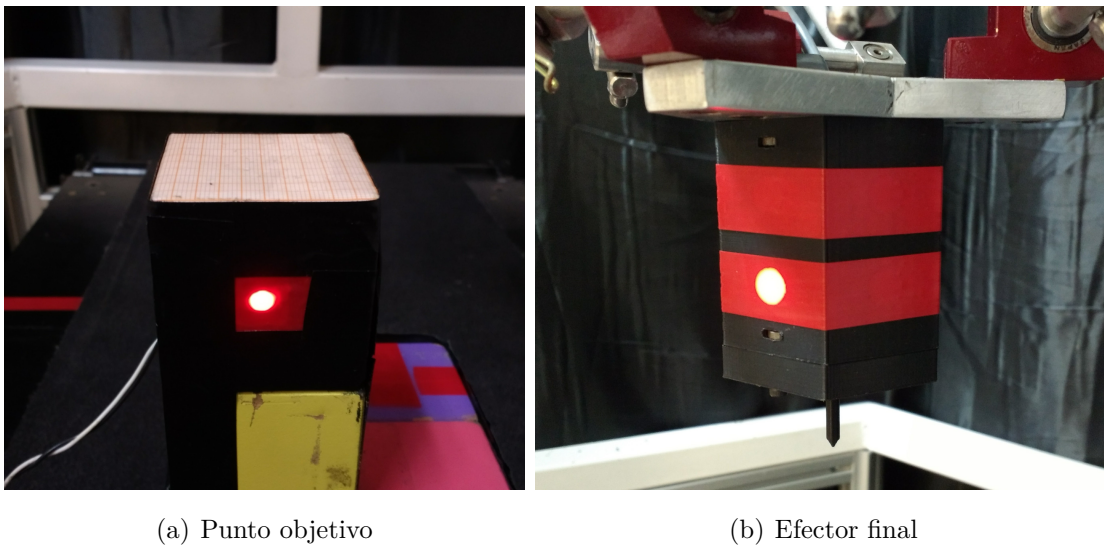


Figura 3.1. Posición de las marcas visuales físicas.

Para dar solución a la problemática de la colisión del efector final con el punto objetivo, se propone crear marcas visuales virtuales. Esto consiste en crear marcas con un cierto desfase en el espacio de cámara a partir de las marcas físicas. Para fines de este trabajo, la marca visual del efector final se moverá hacia el puntero del mismo como se muestra en la Fig. 3.2.

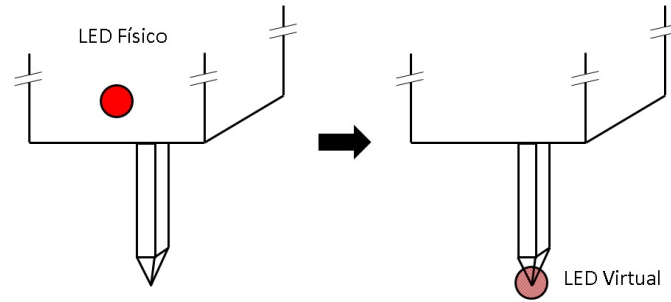


Figura 3.2. Marca visual virtual del efector final del robot Delta.

La marca visual virtual del punto objetivo, se desplazará hacia la parte superior del cubo como se muestra en la Fig. 3.3. Con esto, se podrá posicionar el robot por encima del punto objetivo y así evitar cualquier tipo de colisiones.

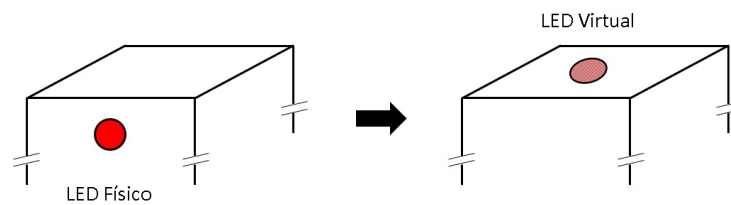


Figura 3.3. Marca visual virtual del punto objetivo.

Para lograr el traslado de la marca visual física a la virtual, se hará uso de la metodología VW - CSM siguiendo esta serie de pasos:

1. Obtener las coordenadas u y v de la marca visual física del objetivo y/o efector final (según sea el caso).
 2. Hacer uso de la metodología VW - CSM para obtener la matriz de pesos \mathbf{W}_i de la Ec. (2.16), esto a modo de actualiza/refinar los parámetros de visión (Ec. (2.2)) y aumentar la precisión del traslado.
 3. Obtener la posición tridimensional de la marca visual en cuestión (Ec. (2.11)) y aplicarle el desfase a donde se requiera.
 4. Con la posición tridimensional ya desplazada, hacer uso de la Ec. (2.1) para el cálculo de la posición en plano de imagen.
-

3.4. Filtro de Kalman

El Filtro de Kalman fue creado por R. Kalman en 1960 [49, 50] y es normalmente descrito como *un algoritmo recursivo para el procesamiento óptimo de datos de un sistema dinámico lineal*. El Filtro de Kalman toma en cuenta toda la información disponible, así como el ruido blanco o gaussiano que se pueda presentar tanto en el estado del sistema como en las mediciones. Como se explicó anteriormente, si se realizan tareas de seguimiento de trayectorias para manipular objetos en movimiento (Ec. (3.9)), para lograr un buen seguimiento se requiere conocer la velocidad del objeto (\dot{s}^*), por lo que se propone el uso del Filtro de Kalman para estimarla.

El Filtro de Kalman consiste en estimar el estado $\mathbf{x} \in \mathfrak{R}^n$ de un proceso en tiempo discreto que es representado por la ecuación diferencial lineal, [49]:

$$\mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_k + \mathbf{w}_k \quad (3.10)$$

donde \mathbf{A} es una matriz $n \times n$ que relaciona el estado previo (\mathbf{x}_{k-1}) con el estado actual (\mathbf{x}_k), \mathbf{B} es una matriz $n \times l$ que relaciona la entrada de control opcional $\mathbf{u} \in \mathfrak{R}^l$ con \mathbf{x}_k y \mathbf{w}_k representa el ruido originado en el proceso. Además, se tiene que incluir la medición $\mathbf{Y} \in \mathfrak{R}^m$, descrita como:

$$\mathbf{Y}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k \quad (3.11)$$

donde \mathbf{H} es una matriz $m \times n$ que relaciona el estado (\mathbf{x}_k) con la medición (\mathbf{Y}_k) y \mathbf{v}_k es el ruido generado en esta última.

Las variables \mathbf{w}_k de la Ec. (3.10) y \mathbf{v}_k de la Ec. (3.11) se asume que son independientes y cuentan con una distribución de probabilidad normal, esto se puede mostrar de la siguiente manera:

$$\begin{aligned} p(\mathbf{w}) &\sim N(0, \mathbf{Q}) \\ p(\mathbf{v}) &\sim N(0, \mathbf{R}) \end{aligned} \quad (3.12)$$

donde (\mathbf{Q}) es la matriz de covarianza del error en el proceso y (\mathbf{R}) es la matriz de covarianza del error en la medición.

El Filtro de Kalman deduce el proceso usando una forma de control de lazo cerrado, donde primero estima el estado y después obtiene una retroalimentación en forma de mediciones. Con esto, el Filtro de Kalman se divide en dos etapas: predicción y corrección, que se describen a continuación.

Predicción

Esta etapa es la responsable de proyectar el estado y la covarianza del error para obtener un estimado *a priori* para la siguiente etapa. Las ecuaciones que conforman esta etapa son las siguientes:

$$\hat{\mathbf{x}}_k^- = \mathbf{A}\hat{\mathbf{x}}_{k-1} + \mathbf{B}\mathbf{u}_{k-1} \quad (3.13)$$

$$\mathbf{P}_k^- = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q} \quad (3.14)$$

donde $\hat{\mathbf{x}}_k^-$ representa el estado estimado y \mathbf{P}_k^- la covarianza del error, ambos *a priori*. El término $\hat{\mathbf{x}}_{k-1}$ es el estado previo estimado [49].

Corrección

Esta etapa es la responsable por la retroalimentación, incorporando una nueva medición en el estimado del estado *a priori* para obtener un estimado del estado *a posteriori*. Las ecuaciones que conforman esta etapa son las siguientes:

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}^T (\mathbf{H} \mathbf{P}_k^- \mathbf{H}^T + \mathbf{R})^{-1} \quad (3.15)$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{Y}_k - \mathbf{H} \hat{\mathbf{x}}_k^-) \quad (3.16)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}) \mathbf{P}_k^- \quad (3.17)$$

donde $\hat{\mathbf{x}}_k$ representa el estado estimado *a posteriori*, \mathbf{P}_k la covarianza del error *a posteriori* e \mathbf{I} es una matriz identidad. El término \mathbf{K}_k de la Ec. (3.15), que es conocido como la *Ganancia de Kalman*, representa el peso que se le debe de dar a la medición tomada [49]. En la Fig. 3.4 se aprecia de manera esquemática el algoritmo del Filtro de Kalman.

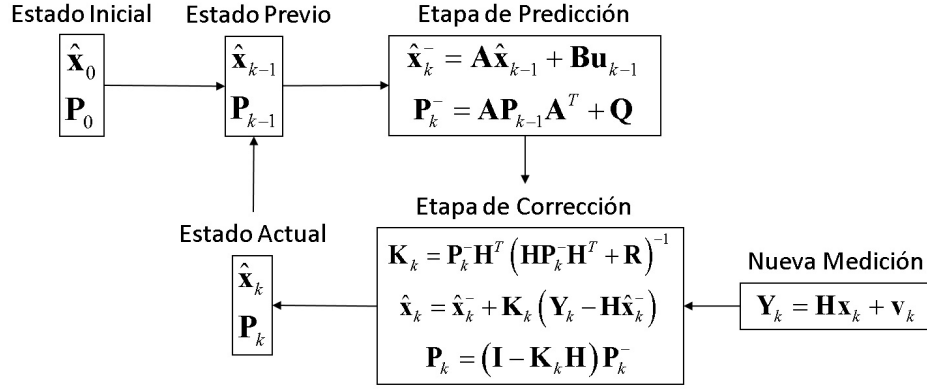


Figura 3.4. Diagrama esquemático del Filtro de Kalman.

3.4.1. Estimación de la velocidad del punto objetivo

Como lo describió López-Lara en [45], se supondrá el comportamiento del punto objetivo como un movimiento rectilíneo uniformemente acelerado dentro del espacio de cámara. El modelo que representa este comportamiento puede definirse como un movimiento rectilíneo uniforme con aceleración constante, descrito de la siguiente manera:

$$u = u_0 + \dot{u}t + \frac{1}{2}\ddot{u}t^2 \quad (3.18)$$

donde u_0 es la posición actual del punto objetivo con una velocidad(\dot{u}) y una aceleración(\ddot{u}) expresadas en pixeles/s y pixeles/s² respectivamente. Para el presente trabajo, se usará un Filtro de Kalman para cada uno de los ejes de ambas cámaras (u_1, v_1, u_2, v_2). Esto debido a los resultados presentados en [45], en donde se presentan mejores precisiones usando 4 Filtros de Kalman en lugar de 2 (un Filtro de Kalman para dos ejes).

3.5. Análisis de estabilidad

La ley de control propuesta (Ec. (3.9)) está bien definida si las inversas de los Jacobianos (Ec. (1.6), (1.7)) existen y son conocidos a lo largo del movimiento del robot. En este caso, el Jacobiano del robot estará bien definido evitando un número finito de posiciones singulares. Además, el Jacobiano de visión estará bien definido cuidando que las cámaras no tengan la misma orientación o posición. El Jacobiano del robot estará bien definido evitando un número finito de posiciones singulares. Una matriz esta bien definida siempre y cuando sea de rango pleno o su determinante sea diferente de 0.

Dado que el sistema controlado es un sistema lineal, es conocido que el sistema es estable si $-\mathbf{G}$ es Hurwitz. Como la matriz \mathbf{G} se puede elegir libremente (es una matriz de ganancias de control), la estabilidad del sistema controlado se garantiza definiendo $-\mathbf{G}$ Hurwitz. Por otro lado, la obtención de este sistema de lazo cerrado depende de conocer perfectamente los Jacobianos involucrados en la ley de control lo cual es poco realista. Sin embargo, el análisis de estabilidad para el sistema controlado considerando aproximaciones de los términos que intervienen en la ley de control requiere un desarrollo significativo por lo que se deja como objeto de estudio de un trabajo posterior.

3.6. Singularidades en el robot Delta

Con el fin de asegurar que la ley de control propuesta (Ec. (3.9)) se ejecute de manera correcta, se analizarán las singularidades que pueda presentar el robot Delta y cómo evitarlas. Una forma de encontrar estas singularidades es por medio del jacobiano del robot, en donde, si el determinante de la matriz jacobiana es igual a cero significa que el robot entró en singularidad.

Analizando las dos matrices que conforman el jacobiano del robot Delta (Ec. (1.6) y (1.7)):

$$J_x = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

$$J_q = \text{diag}(b_{11}, b_{22}, b_{33})$$

donde:

$$a_{i1} = \text{sen}(\theta_{i3}) \cos(\theta_{i1} + \theta_{i2}) \cos(\alpha_i) + \cos(\theta_{i3}) \text{sen}(\alpha_i)$$

$$a_{i2} = \text{sen}(\theta_{i3}) \cos(\theta_{i1} + \theta_{i2}) \text{sen}(\alpha_i) - \cos(\theta_{i3}) \cos(\alpha_i)$$

$$a_{i3} = -\text{sen}(\theta_{i3}) \text{sen}(\theta_{i1} + \theta_{i2})$$

$$b_{ii} = L_1 \text{sen}(\theta_{i2}) \text{sen}(\theta_{i3})$$

para $i = 1, 2, 3$ (numero de extremidades en el robot Delta). Se pueden presentar cinco casos en donde se puede entrar en singularidad [15]. A continuación se presentan estos casos y un modo de evitarlos.

1. **Para $\theta_{i2} = 0$ o π para cualquier i , singularidad en matriz J_q .** Esta singularidad ocurre cuando el brazo actuado se encuentra extendido (Fig. 3.5(a)) o contraído (Fig. 3.5(b)) con el eslabón pasivo. Esta última condición no ocurre debido a colisiones con el robot Delta.
2. **Para $\theta_{i2} = 0$ o π para todo i , singularidad en matriz J_q .** Esta singularidad ocurre cuando las 3 extremidades del robot se encuentran totalmente extendidas (Fig. 3.6(a)) o se encuentran contraídas como se observa en la Fig. 3.6(b). Esta última condición no ocurre debido a colisiones con el robot Delta.
3. **Para $\theta_{i3} = 0$ o π para cualquier i , singularidad en la matriz J_q y J_x .** Esta singularidad ocurre cuando el eslabón pasivo se encuentra totalmente perpendicular al eslabón activo. Sin embargo, no ocurre debido a colisiones con este último eslabón.
4. **Para $\theta_{i3} = 0$ o π para todo i , singularidad en la matriz J_q y J_x .** Esta singularidad ocurre cuando los tres eslabones pasivos del robot se encuentran totalmente perpendiculares a los eslabones activos con los que se encuentran unidos. Sin embargo, no puede llegar a esta configuración por colisiones con los eslabones activos. Además, solo dos brazos pueden alcanzar esa condición, el tercero se mantiene extendido, Fig. 3.7.

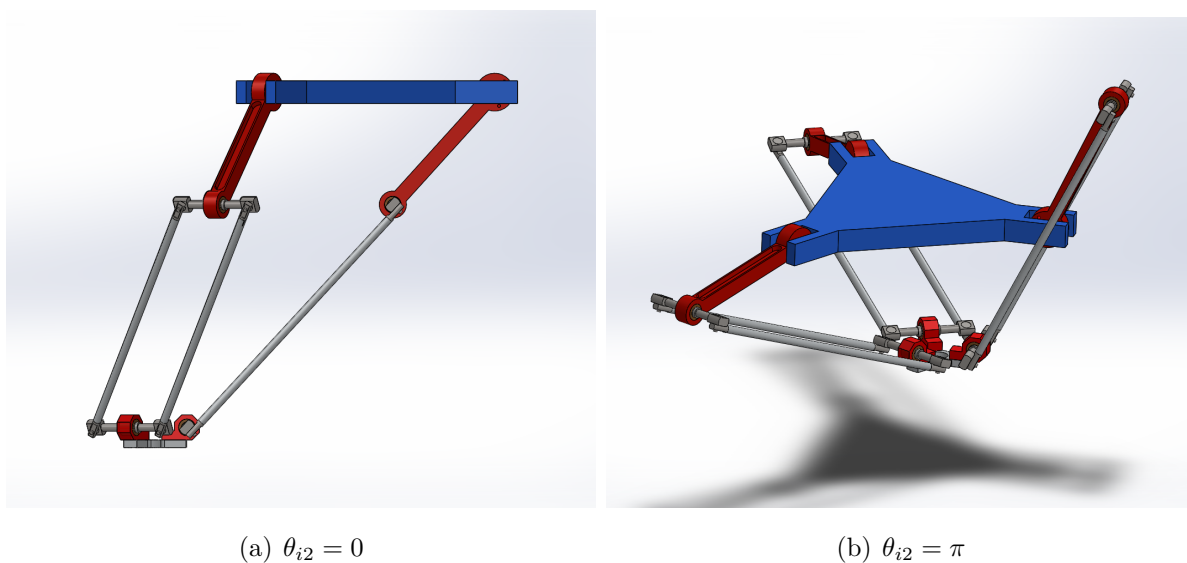
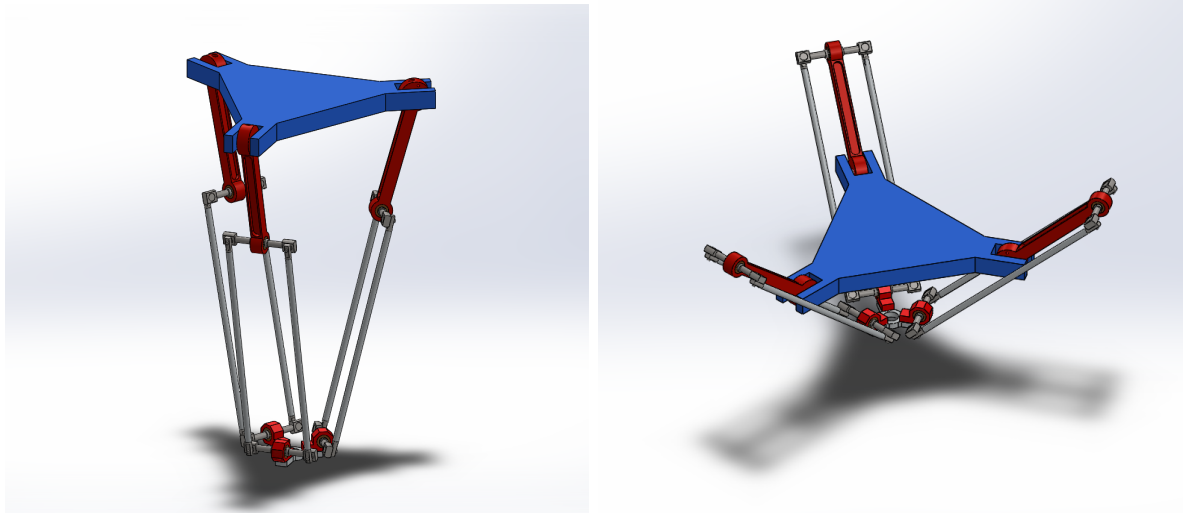


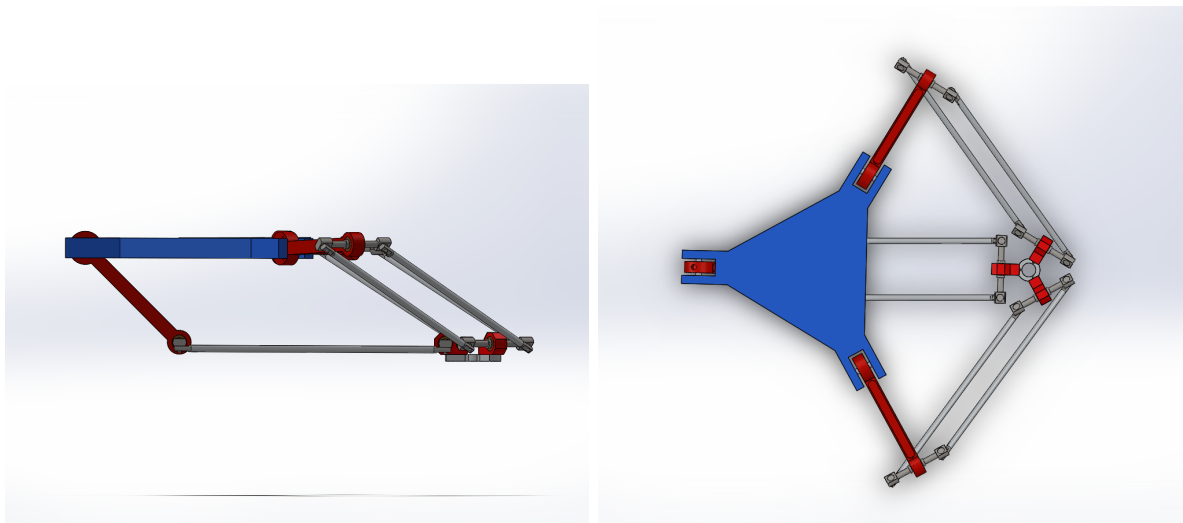
Figura 3.5. Posiciones singulares del robot Delta en $\theta_{i2} = 0$ o π para cualquier i .



(a) $\theta_{i2} = 0$

(b) $\theta_{i2} = \pi$

Figura 3.6. Posiciones singulares del robot Delta en $\theta_{i2} = 0$ o π para todo i .

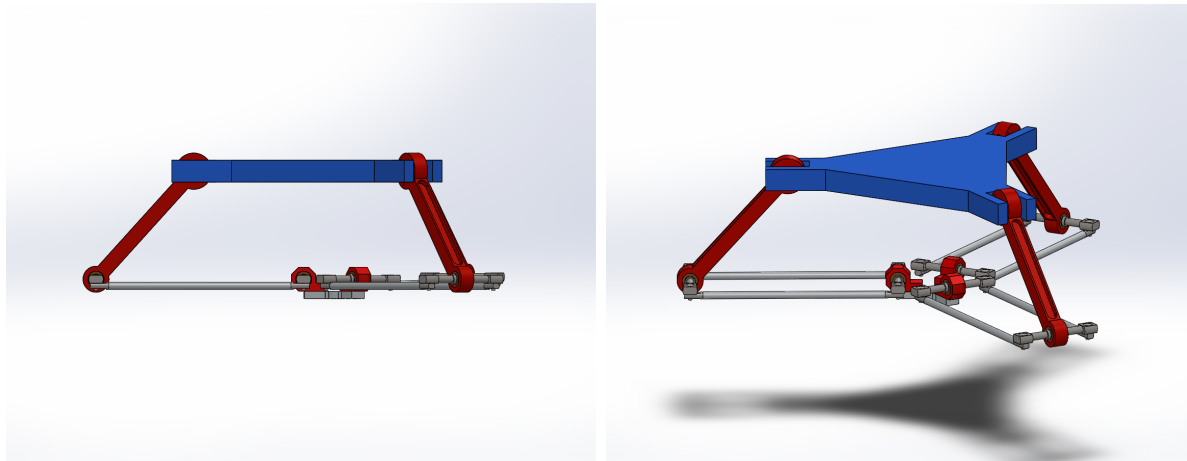


(a) Vista lateral

(b) Vista superior

Figura 3.7. Posición en el robot Delta donde no es posible alcanzar la configuración $\theta_{i3} = 0$ o π para cualquier y todo i

5. **Para $\theta_{i1} + \theta_{i2} = 0$ o π para todo i , singularidad en la matriz J_x .** Esta singularidad ocurre cuando el eslabón pasivo se alinea con la plataforma móvil, esto se cumple bajo la condición $R+L1 > r+L2$, Fig. 3.8. Sin embargo, no ocurre porque las medidas geométricas del robot PARALLIX LKF-2040 son: $R = 200$ mm, $r = 65$ mm, $L1 = 200$ mm, $L2 = 400$ mm.



(a) Vista lateral

(b) Vista isométrica

Figura 3.8. Configuración en el robot delta donde $\theta_{i1} + \theta_{i2} = \pi$

Los casos 1, 2 y 3 ocurren en la frontera del espacio de trabajo del robot. Con el fin de asegurar el desempeño requerido en la ley de control y de cuidar la integridad del robot Delta, se proponen dos soluciones para prevenir estos dos casos: el primero es trabajar dentro del espacio efectivo del robot Delta, evitando trayectorias que se acerquen o alcancen los límites. El segundo, como medida de seguridad adicional por software, se puede poner una condición en el algoritmo de control para que al momento de que detecte que cualquier θ_{i2} se vuelva 0, detenga el robot de manera inmediata y envíe mensaje explicando el error.

3.7. Esquema de Control y Diagrama de Flujo

La Fig. 2.1 ilustra la ley de control CSM en velocidades (Ec. (3.9)) de forma esquemática.

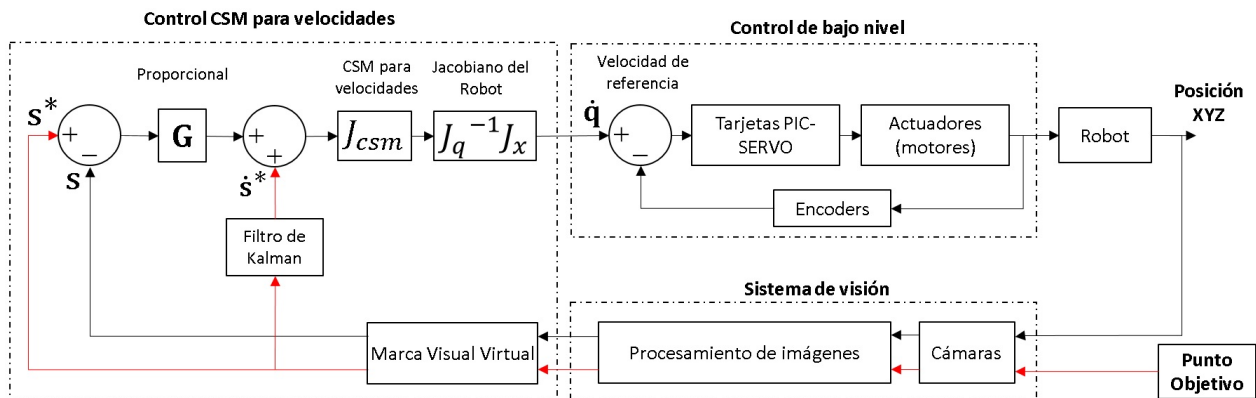


Figura 3.9. Esquema de control CSM en velocidades.

La Fig. 3.10, plasma en diagrama de flujo el algoritmo programado para la ejecución de una tarea de posicionamiento estática o en movimiento. En la implementación práctica se introdujo una condición de convergencia en términos de la magnitud del error. Cuando se cumple la condición de convergencia se da por terminada la tarea.

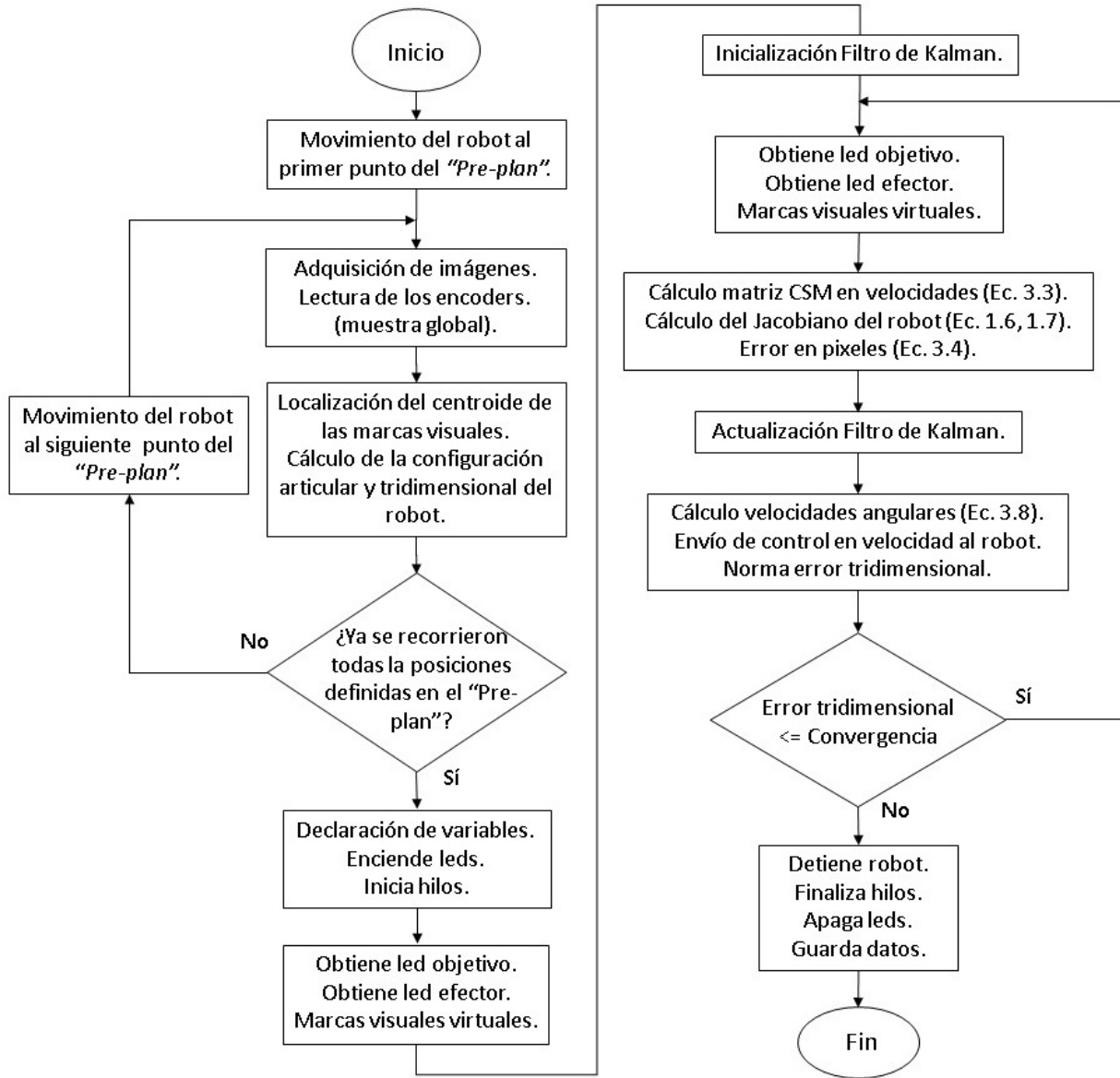


Figura 3.10. Diagrama de flujo CSM en velocidades.

3.8. Conclusiones del capítulo 3

En este capítulo se desarrolló una nueva técnica basada en visión, para el control de un robot paralelo tipo Delta. Esta técnica de control, llamada CSM en velocidades, esta compuesta por los parámetros de visión de la metodología LCM - CSM, estos son usados para relacionar las velocidades en el plano de imagen con las velocidades tridimensionales del efector final.

La técnica de CSM en velocidades, requiere de la integración en tiempo real de la información visual, para conocer en todo momento el error existente entre las marcas visuales adheridas al efector final y el punto objetivo. Esta nueva técnica, al ser en velocidad, permite al robot realizar trayectorias más suaves y rápidas, además de hacer seguimiento de objetos en movimiento con trayectorias arbitrarias.

En este trabajo, el Filtro de Kalman es usado para la estimación de la velocidad del objetivo en el plano de imagen, ya que se requiere en la ley control CSM en velocidades. Como se presenta en [45], se propone usar un Filtro de Kalman para cada eje de cada cámara, por presentar una estimación más confiable del término de velocidad.

Para asegurar el correcto funcionamiento de la ley de control presentada en este capítulo, se presentó un análisis de estabilidad del mismo así como las singularidades que pueda presentar el robot Delta. En el siguiente capítulo se mostrarán los resultados experimentales de tareas de posicionamiento tanto estáticas como en movimiento haciendo uso de esta nueva técnica propuesta.

Resultados Experimentales

En este capítulo se presentan los resultados obtenidos en la implementación de las metodologías de control LCM - CSM y VW - CSM bajo un modelo lineal de cámara en el robot paralelo tipo Delta, descritos en la Sección 2.2 y 2.3 respectivamente. Además, se muestran las pruebas realizadas del posicionamiento de un objetivo estático y de un objetivo en movimiento con trayectoria lineal y arbitraria usando la nueva ley de control propuesta, vista en el Capítulo 3. Se describe las características de implementación de las pruebas realizadas.

4.1. Características de la implementación

4.1.1. Hardware

La plataforma experimental es un robot paralelo tipo Delta de 3 grados de libertad modelo PARALLIX LKF-2040, Fig. 1.11. Este robot es de tipo académico y cuenta con 3 motores Maxtor de D.C., cada uno con un reductor planetario de 14:1. La lectura de la posición angular de los motores es realizada por encoders incrementales CUI AMT102-V, con una resolución de 400 cuentas/vuelta. El control de los motores y la lectura de los encoders es gobernado por tarjetas PIC-SERVO SC, [18].

Para el sistema de visión, se usaron dos cámaras uEye modelo UI-2210SE-M-GL con una resolución de 640 x 480 pixeles e integrando un sensor CCD de 1/2" a 75 cuadros por segundo, Fig. 4.1. La computadora personal utilizada cuenta con un procesador Intel® Core™ i5-6600HQ a 2.4 GHz, 8 GB de memoria RAM. El sistema cuenta con una banda plana con motor D.C. de armadura y un driver ABB BALDOR BC 138, Fig. 4.3.



Figura 4.1. Cámara uEye modelo UI-2210SE-M-GL.

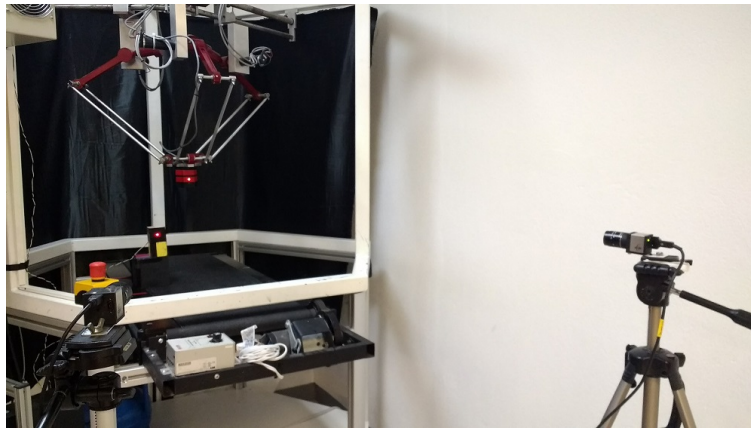


Figura 4.2. Robot Delta Parallax LKF-2040 y posición de las cámaras.

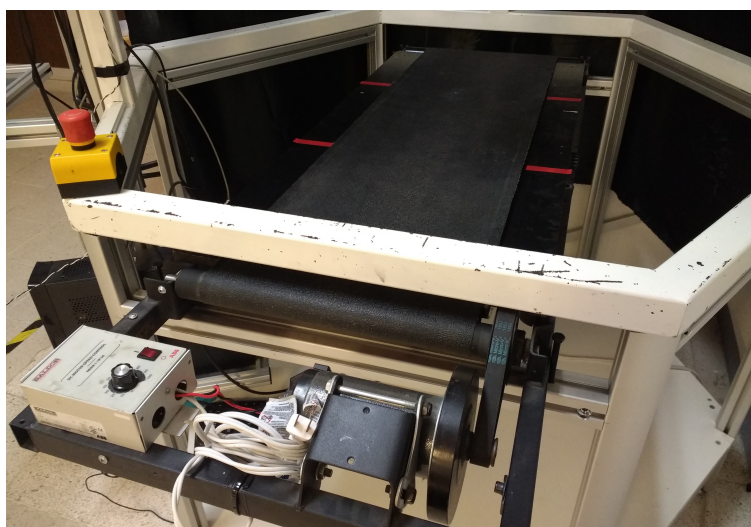


Figura 4.3. Banda plana con motor D.C. de armadura y driver ABB BALDOR BC 138.

En la Fig. 4.4 se aprecia la posición de las cámaras con respecto al robot Delta. Se colocaron de esta manera con el fin de que en la imagen se pueda apreciar a la perfección todo el espacio de trabajo del robot. Las cámaras se posicionaron a 1500 mm aproximadamente con respecto al marco del robot y una distancia entre ellas de 900 mm.

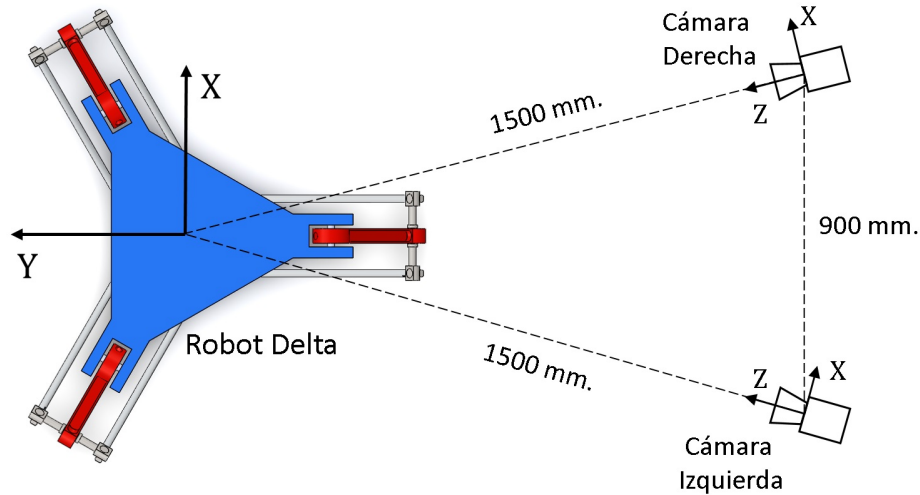
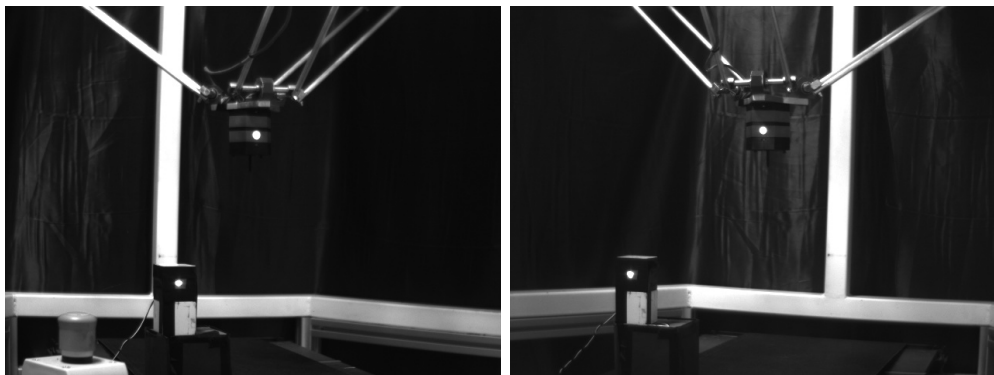


Figura 4.4. Posición de las cámaras con respecto al robot Delta.

4.1.2. Software

La interfaz gráfica junto con el algoritmo de control fue desarrollada con el programa “Microsoft® Visual Studio Express™ 2012 for Desktop”. El lenguaje de programación usado es C++, se eligió debido a su simplicidad en la programación y porque esta basado en código fuente. Esto significa que su tiempo de ejecución es menor comparado con lenguajes interpretados como Matlab™. Las librerías usadas en la programación son: PIC-SERVO [51] para la conexión y manipulación de las tarjetas del mismo nombre y GSL para el uso de álgebra lineal [52]. El sistema operativo usado es Windows 10 Single-Language™.

Para poder hacer uso de las metodologías de control por visión, se requiere de obtener el centroide de las marcas visuales adheridas al efector final y al punto objetivo. Para ello se usó OpenCV [53], que es un conjunto de bibliotecas genéricas para la adquisición y tratamiento de imágenes. En la Fig. 4.5, se presentan dos fotografías tomadas por las cámaras uEye, en donde se puede observar el robot Delta y el punto objetivo con un LED adherido a cada uno de ellos.

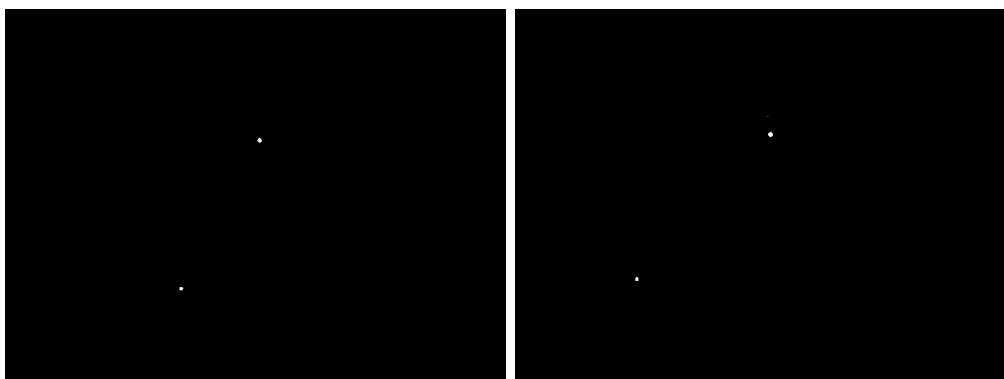


(a) Izquierda

(b) Derecha

Figura 4.5. Fotografías tomadas por las dos cámaras uEye donde se observa el robot Delta y el punto objetivo.

Las fotografías mostradas en la Fig. 4.5, no nos dicen por sí solas cual es el centroide en píxeles de las marcas visuales, por lo que se requiere de un tratamiento de imagen para obtener la información que se requiere. En este caso, se usará un tratamiento conocido como *umbralizado*, que consiste en hacer de una imagen que está en escala de grises a una en blanco y negro. Esto se logra poniendo un umbral y todos los píxeles que estén abajo de éste (píxeles con grises más oscuros) se vuelvan negros, y los que estén arriba (píxeles con grises más claros) se vuelvan blancos. Así, con menos iluminación de la escena y haciendo este proceso de umbralizado, se obtiene en la imagen los píxeles en blanco donde se encuentran las marcas visuales (Fig. 4.6), en donde es más fácil el cálculo de los centroides.



(a) Izquierda

(b) Derecha

Figura 4.6. Imágenes umbralizadas a partir de las fotografías mostradas en la Fig. 4.5, los puntos blancos representan las marcas visuales del efector final y del objetivo.

Se desarrolló una interfaz gráfica basada en una preexistente, se le dio el nombre de ROBWIN CSM v2.6 (Fig. 4.7) y en ella se puede manipular el robot Delta, adquirir en tiempo real lo que observan las cámaras y el umbralizado, así como realizar las pruebas para cualquiera de las metodologías de control por visión discutidas en este trabajo.

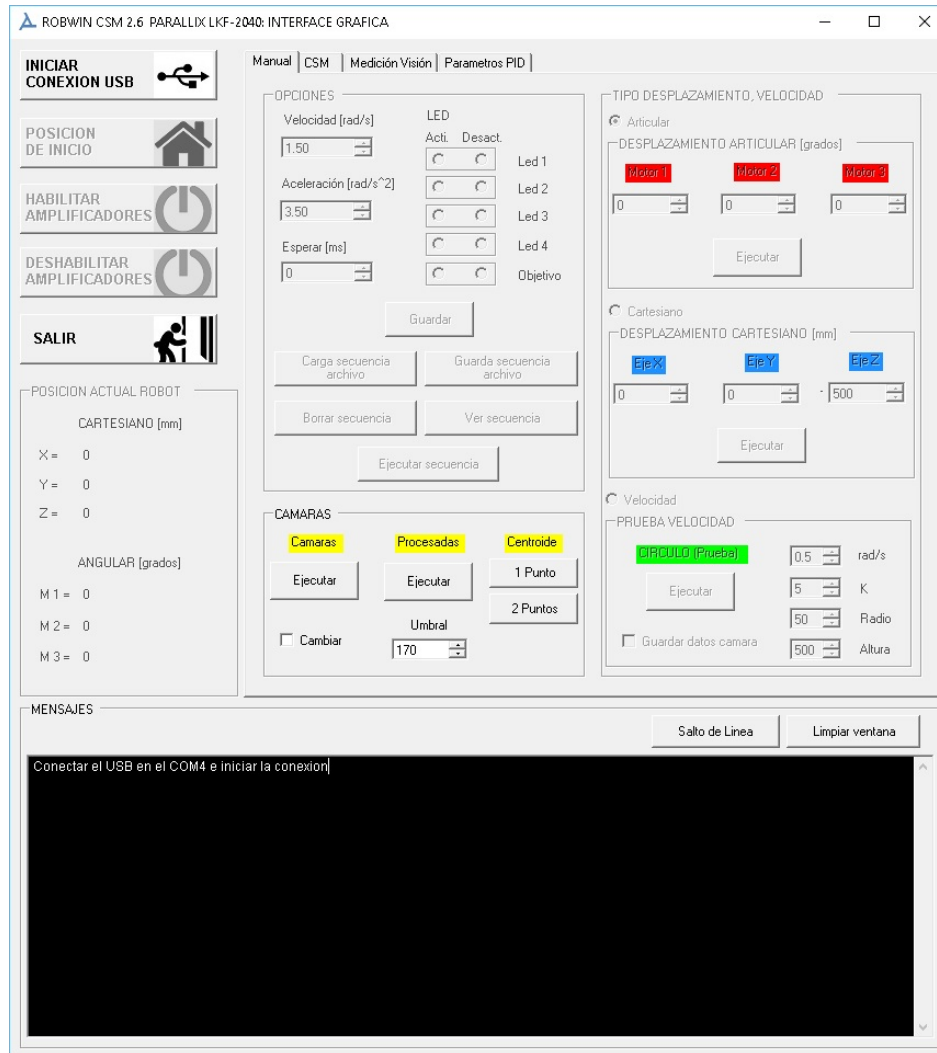


Figura 4.7. Interfaz gráfica ROBWIN CSM v2.6 para el control de la plataforma experimental.

Para optimizar el cálculo de la ley de control de CSM en velocidades en el tiempo de ejecución por la adquisición/tratamiento de las imágenes y el cálculo de los centroides, se hizo uso de la *programación multi-hilos*. Esto es, en el hilo principal se realiza el cálculo de la metodología de control por visión, mientras que en hilos secundarios se hará la adquisición de las fotografías de ambas cámaras, el umbralizado de estas mismas y el cálculo de los centroides.

4.1.3. “Pre-plan”

Para las pruebas de las metodologías LCM - CSM, VW - CSM de un modelo lineal de cámara, y la nueva ley de control propuesta en este trabajo se usa el mismo “Pre-plan”, como una de las condiciones de referencia para que los resultados obtenidos sean lo más comparable posible. Se tomaron 410 puntos distribuidos en todo el espacio efectivo de trabajo del robot Delta Parallax LKF-2040 [54]. Los puntos del “Pre-plan” forman un prisma cuadrangular mostrado en la Fig. 4.8, 4.9, donde los asteriscos azules representan el punto asociado al efector final en coordenadas tridimensionales con respecto al marco de referencia del robot Delta.

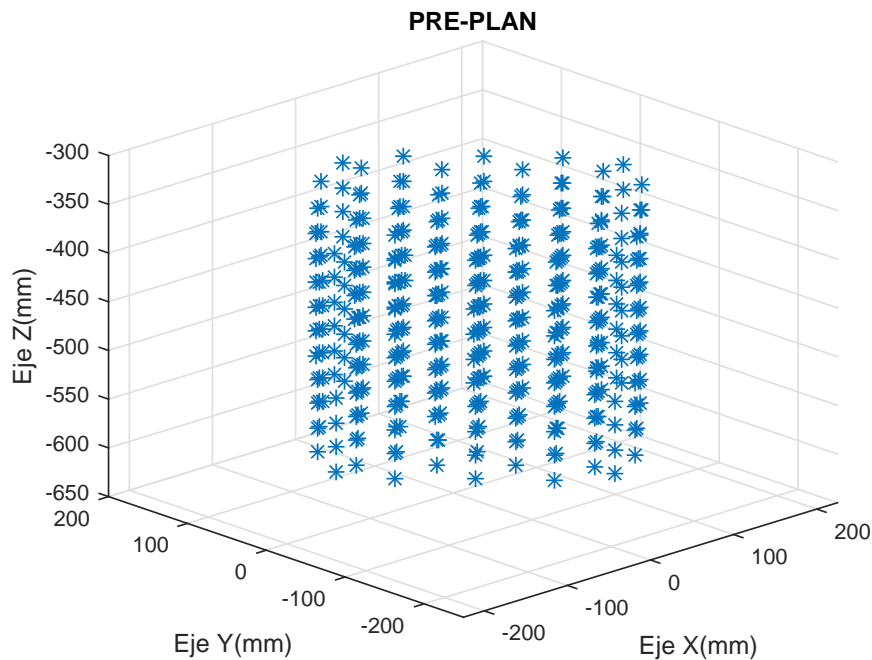


Figura 4.8. Vista isométrica de la distribución de los puntos usados en el “Pre-plan”.

Las dimensiones del cuadrado son de 350 mm en sus líneas diagonales, mientras que la altura del prisma (altura del espacio de trabajo considerado) es de 250 mm. La distancia entre cada punto es de 50 mm en el eje x y el eje y , mientras que en el eje z es de 25 mm. Estas medidas se pueden apreciar en la Fig. 4.9, donde se observan la vista superior y vista frontal del “Pre-plan”.

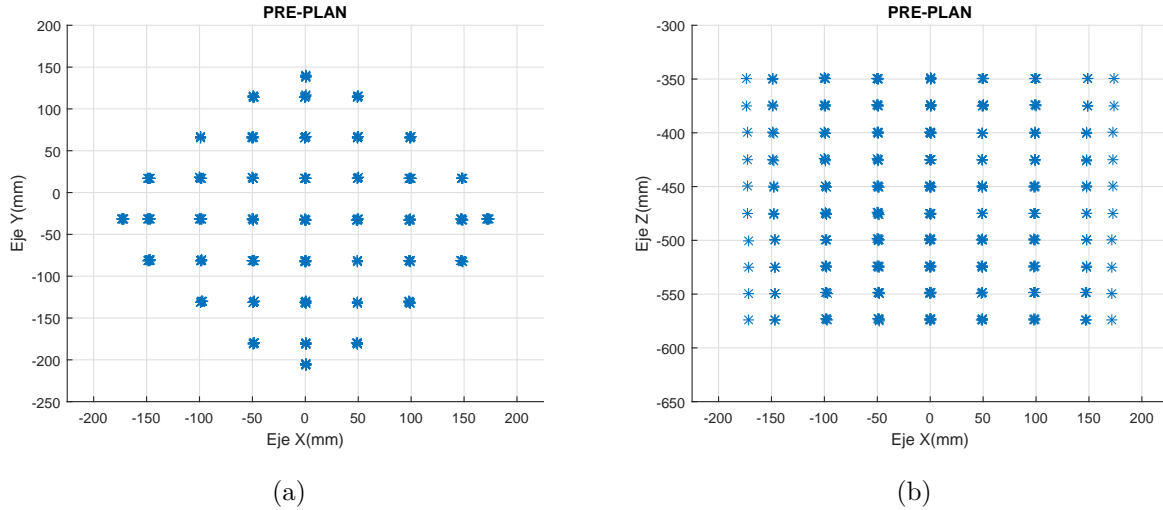


Figura 4.9. Vista superior (a) y frontal (b) de los puntos usados en el “*Pre-plan*”.

4.2. Resultados experimentales de la metodología LCM - CSM

En la Sección 2.2, se vio el algoritmo conformado por la metodología LCM - CSM. Para aumentar la precisión en cualquier tarea de posicionamiento, se requiere de tomar muestras locales cerca del punto objetivo para darles mayor importancia. Para esto, se propone una trayectoria de transición discretizada en cierto número de elementos que representan las muestras locales a considerar. Las pruebas realizadas fueron hechas bajo las siguientes características:

- La trayectoria de transición empieza a partir de la posición tridimensional calculada del punto objetivo (usando los parámetros de visión del “*Pre-plan*”), con 50 mm de desfase en la dirección $Z+$ del marco del robot, como se muestra en la Fig. 4.10.
- La trayectoria de transición se dividió en 10 partes, por ende se obtendrán 10 muestras locales.
- El valor en la matriz de pesos (Ec. (2.9)), es de 10 para cada muestra local, dejando el valor de 1 para todos los puntos tomados en el “*Pre-plan*” (Fig. 4.8).

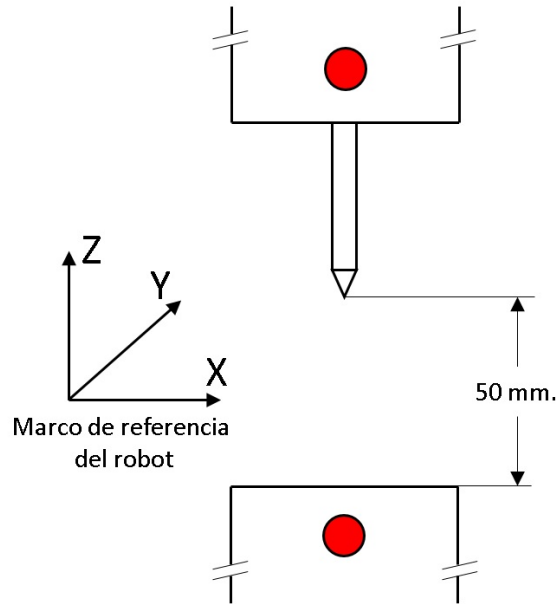


Figura 4.10. Posición inicial de la trayectoria de transición.

Posición	Repetición	Error (mm)	Posición	Repetición	Error (mm)
1	1	1.25	6	1	1.21
	2	1.12		2	1.03
	3	1.14		3	1.30
2	1	0.83	7	1	0.95
	2	0.83		2	0.95
	3	0.88		3	0.84
3	1	1.09	8	1	0.74
	2	0.97		2	0.83
	3	1.16		3	0.95
4	1	1.06	9	1	0.81
	2	1.01		2	1.04
	3	1.01		3	0.86
5	1	1.23	10	1	0.97
	2	1.10		2	0.79
	3	1.10		3	0.92

Tabla 4.1. Resultados de las pruebas realizadas en la implementación de la metodología LCM - CSM.

Se realizaron 10 tareas de posicionamiento a un punto objetivo, en posiciones arbitrarias dentro del espacio de trabajo del robot Delta. Se repitió 3 veces cada posicionamiento dando un total de 30 pruebas, donde se midió con un Calibrador Vernier el error (en la dirección X , Y , Z del robot Delta) entre el objetivo y el puntero del efector final, los errores en norma euclidiana se muestran en la Tabla 4.1. La Tabla 4.2 muestra las estadísticas (promedio, valor máximo, valor mínimo, desviación estándar) de los datos que se describen en este párrafo.

Error (mm)	
Promedio	0.997
Máximo	1.30
Mínimo	0.74
Desv. Estándar	0.151

Tabla 4.2. Error estadístico de la implementación de la metodología LCM - CSM.

En cuanto al tiempo de ejecución, el tiempo total que le tomó al robot realizar la tarea desde su posición de inicio hasta el objetivo es de **6.5 segundos**. Esto se debe a que se incluyen las 10 posiciones locales dentro de la trayectoria de transición para el refinamiento de los parámetros de visión. El cálculo de los parámetros de visión (Ec. (2.2)) para cada nueva muestra local tarda aproximadamente **48 milisegundos**. El cálculo de la posición tridimensional (Ec. (2.11)), toma un tiempo aproximado de **1 milisegundo**.

4.3. Resultados experimentales de la metodología VW - CSM de un modelo lineal cámara

En la Sección 2.3, se planteó el algoritmo para la metodología de VW - CSM bajo un modelo lineal de cámara. Al igual que en las pruebas realizadas bajo el esquema LCM - CSM, se ejecutaron 10 tareas de posicionamiento en ubicaciones arbitrarias (mismas que en la implementación de LCM - CSM) con 3 repeticiones cada una. En la Tabla 4.3, se muestra la norma euclidiana del error en X , Y , Z medido con un Calibrador Vernier. La Tabla 4.4 muestra las estadísticas (promedio, valor máximo, valor mínimo, desviación estándar) de los datos que se describen en este párrafo.

Posición	Repetición	Error (mm)	Posición	Repetición	Error (mm)
1	1	0.84	6	1	1.16
	2	0.97		2	1.19
	3	1.03		3	1.14
2	1	1.29	7	1	0.79
	2	1.11		2	0.94
	3	1.19		3	0.85
3	1	0.97	8	1	0.95
	2	1.16		2	1.07
	3	0.99		3	1.04
4	1	0.95	9	1	1.06
	2	1.06		2	1.18
	3	1.16		3	1.27
5	1	0.84	10	1	0.82
	2	0.75		2	0.91
	3	0.83		3	0.75

Tabla 4.3. Resultados de las pruebas realizadas en la implementación de la metodología VW - CSM de un modelo lineal.

Error (mm)	
Promedio	1.010
Máximo	1.29
Mínimo	0.75
Desv. Estándar	0.154

Tabla 4.4. Error estadístico de la implementación de la metodología VW - CSM de un modelo lineal.

Bajo esta metodología, el tiempo de posicionamiento hacia el objetivo es cerca de los **1.3 segundos**. Donde el cálculo de la matriz de pesos (Ec. (2.16)) de ambas cámaras le toma un tiempo aproximado de **4 milisegundos**, el cálculo de los parámetros de visión (Ec. (2.2)) le toma alrededor de **40 milisegundos** y la posición tridimensional (Ec. (2.11)) se calcula en **1 milisegundo**. A pesar de que la exactitud de posicionamiento de esta metodología con respecto a LCM - CSM es prácticamente la misma, el tiempo total de ejecución de la tarea es significativamente menor.

4.4. Sistema de Medición por Visión (SMV)

En la práctica, es muy difícil realizar alguna medición entre dos objetos en movimiento (efector final y punto objetivo) con un dispositivo mecánico (Calibrador Vernier, regla, etc.). Debido a lo anterior, para poder medir la exactitud de seguimiento de objetos móviles, se optó por desarrollar un sistema de medición por visión (SMV). El sistema de medición por visión nos permite medir, sin contacto, distancias en su campo de visión, incluyendo longitudes entre objetos en movimiento. El instrumento de medición del SMV son dos cámaras y el procedimiento para realizar esta medición se hará bajo la metodología VW - CSM de un modelo lineal de cámara. Se decidió usar esta metodología debido a que se encuentra implementada y su costo computacional es bajo. Posiblemente este sistema de medición no es óptimo, sin embargo no constituye el objetivo de este trabajo y es meramente una herramienta que permite validar el concepto del trabajo desarrollado.

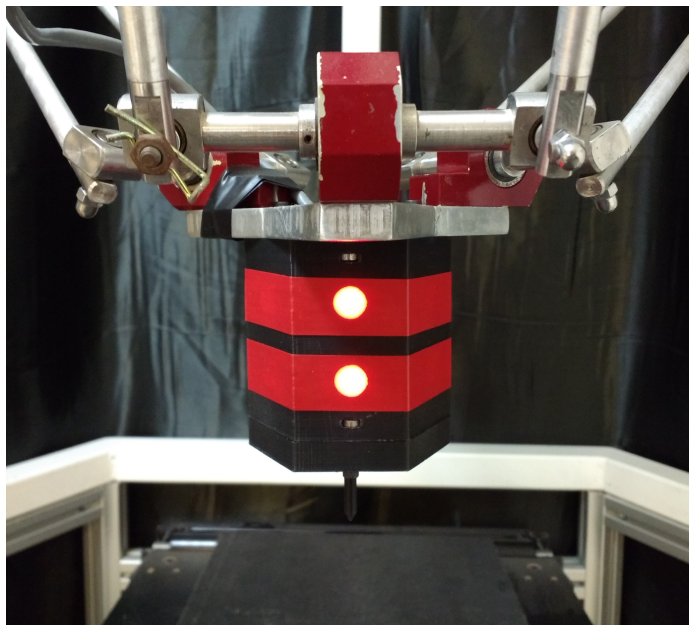


Figura 4.11. Fotografía de los dos LEDs en el efector final para calibrar el SMV.

Para poder realizar una medición del error de seguimiento usando SMV, primero hay que tener una idea de las características de la incertidumbre asociada a este sistema de medición que existe en todo el espacio de trabajo del robot Delta. La incertidumbre es una estimación del posible error en una medida, también representa la probabilidad de que el valor verdadero este dentro de un rango de valores indicado (mínimo y máximo). El patrón de referencia para calibrar el SMV es la distancia que hay entre dos LEDs (**25 mm** de centro a centro) adheridos al efector final, Fig 4.11.

Se midió la distancia que hay entre estos dos LEDs en 205 posiciones distribuidas uniformemente en todo el espacio efectivo de trabajo del robot Delta Parallax LKF-2040, [54]. Para obtener la distancia, se calcula la posición tridimensional de ambos LEDs usando la metodología VW - CSM de un modelo lineal de cámara, recalando que se calcula una matriz de pesos \mathbf{W} (Ec. (2.9)) y una nueva estimación de los parámetros de visión (Ec. (2.2)) para cada LED y en cada una de las 205 posiciones. En cada posición, se toma la distancia entre los LEDs 50 veces para aplicar la media cuadrática. La media cuadrática (x_{RMS}) es una medida estadística de la magnitud de una cantidad variable, se suele utilizar cuando el signo de la variable no es importante y lo que interesa es el valor absoluto del elemento, y se describe de la siguiente manera en su forma discreta [55]:

$$x_{RMS} = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2} \quad (4.1)$$

donde N es el número de muestras, en este caso $N = 50$, y x es el valor de la muestra (distancia entre los LEDs). Por cuestiones de espacio, en la Tabla 4.5 se observan los resultados de 10 de las 205 mediciones, mostrando el valor medio cuadrático en su respectiva posición, y el error con respecto a la distancia real.

Posición del manipulador (mm)			Distancia entre LEDs con media cuadrática (mm)	Error (mm)
X	Y	Z		
-100	0	-400	25.0008	0.0008
-150	0	-300	25.0014	0.0014
-150	-50	-400	24.8816	0.1184
-100	-50	-500	24.8814	0.1186
-100	-100	-450	24.7406	0.2594
-50	-100	-350	25.2612	0.2612
0	150	-400	24.4574	0.5426
150	50	-300	24.4548	0.5452
175	0	-300	23.7485	1.2515
100	50	-500	23.7421	1.2579

Tabla 4.5. Resultados de la distancia calculada entre los dos LEDs en 8 posiciones usando el SMV.

En la Tabla 4.6 se muestra las estadísticas del error entre la medición realizada por el SMV y la distancia real entre dos LEDs adheridos al efector final. El error promedio es 0.37833 mm

con una desviación estándar de 0.26058 mm. Además, conociendo el error mínimo y máximo se determina que en cualquier medición que se realice usando el SMV dentro del espacio de trabajo del robot Delta, tendrá una incertidumbre entre 0.37833 a 1.25793 mm. Esto último quiere decir, por ejemplo, que si la distancia real que hay entre un objetivo y el efector final en una tarea de posicionamiento es de 2 mm, usando el SMV para medir esta distancia, los valores oscilarán entre 0.7421 a 1.9991 mm y 2.0008 a 3.2579 mm.

Error (mm)	
Promedio	0.37833
Máximo	1.25793
Mínimo	0.00082
Desv. Estándar	0.26058

Tabla 4.6. Incertidumbre estadística del SMV.

El origen de esta incertidumbre se debe a la naturaleza discreta del sensor de la cámara donde un píxel se relaciona directamente con las distancias tridimensionales según la profundidad del objeto que se observa (sobre el eje Z de la cámara). En este caso, la resolución de la cámara es de 640 x 480 píxeles y la Tabla 4.7 muestra la relación mm/píxel entre el rango de distancias que se hicieron las mediciones con respecto a la cámara. Se observa que, entre mayor sea la profundidad al medir un objeto, mayor es la relación mm/píxel.

Distancia cámara (mm)	Relación mm/píxel
1000	0.833
1500	1.190
2000	1.562

Tabla 4.7. Relación mm/píxel a distintas distancias con una resolución de cámara 640 x 480 píxeles.

Para mejorar el Sistema de Medición por Visión se pueden usar más cámaras, usar una mayor resolución del sensor óptico, acercar más las cámaras (con el inconveniente de no poder cubrir el espacio de trabajo completo) para reducir la relación mm/píxel o usar marcas visuales con menor diámetro y mayor luminosidad para no dar lugar a una mala estimación del centroide. Sin embargo, en este caso no es el objetivo central del presente trabajo, y no

se tiene a la mano los recursos, por lo que se dejará así y se propone mejorarlo como trabajo a futuro.

4.5. Resultados experimentales de la metodología CSM en velocidades

4.5.1. Sintonización de la ley de control CSM en velocidades

En el Capítulo 3 se desarrolló la metodología CSM para el control de velocidades aplicada a un robot paralelo tipo Delta (Ec. (3.9)). Se realizó una prueba preliminar de posicionamiento dentro del espacio de trabajo del robot Delta, para verificar el funcionamiento de la ley de control bajo las siguientes condiciones: posición inicial del robot ($X = 0$ mm, $Y = 0$ mm, $Z = -450$ mm), valores de los elementos diagonales (g_{ii}) de la matriz de ganancias \mathbf{G} igual a 1.

La metodología CSM en velocidades requiere de la integración en tiempo real de la información visual. El algoritmo de control ejecuta n iteraciones hasta que termine la tarea de posicionamiento. En cada iteración, se guarda la posición en píxeles que toma el efector final y el punto objetivo en ambas cámaras. Con esto, se puede graficar cómo se va moviendo el efector final durante toda la tarea de posicionamiento.

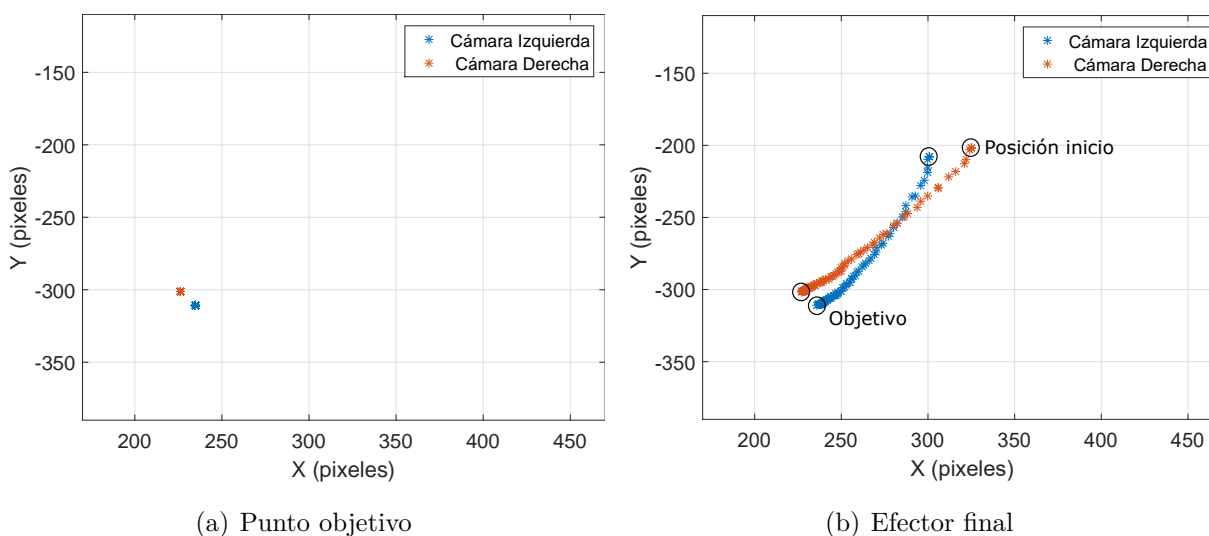


Figura 4.12. Marcas visuales en espacio de cámara durante la prueba preliminar.

En la Fig. 4.12 se aprecia lo que observan la cámara izquierda (azul) y la cámara derecha (rojo) durante esta prueba. Los asteriscos representan las posiciones en el plano de imagen (píxeles) tomadas en cada iteración. Se observa en la Fig. 4.12(a) que el punto objetivo solo se muestra como un asterisco (debido a que se mantiene estático), mientras que en el efector final, se forma una línea casi recta con una mayor acumulación de puntos cerca de la finalización de la tarea.

Si se tiene la posición del punto objetivo y cómo se comporta el robot a lo largo de la prueba preliminar, se puede saber también el comportamiento del error entre ellos (Ec. (3.4)). En la Fig. 4.13(a) se observa este comportamiento, donde se plasman los 4 errores (u y v de ambas cámaras) y se aprecia que estos van decreciendo de manera exponencial hasta llegar a 0. Con esto se corrobora que el sistema controlado en lazo cerrado ($\dot{\mathbf{e}} = -\mathbf{G}\mathbf{e}$) se comporta de la manera requerida.

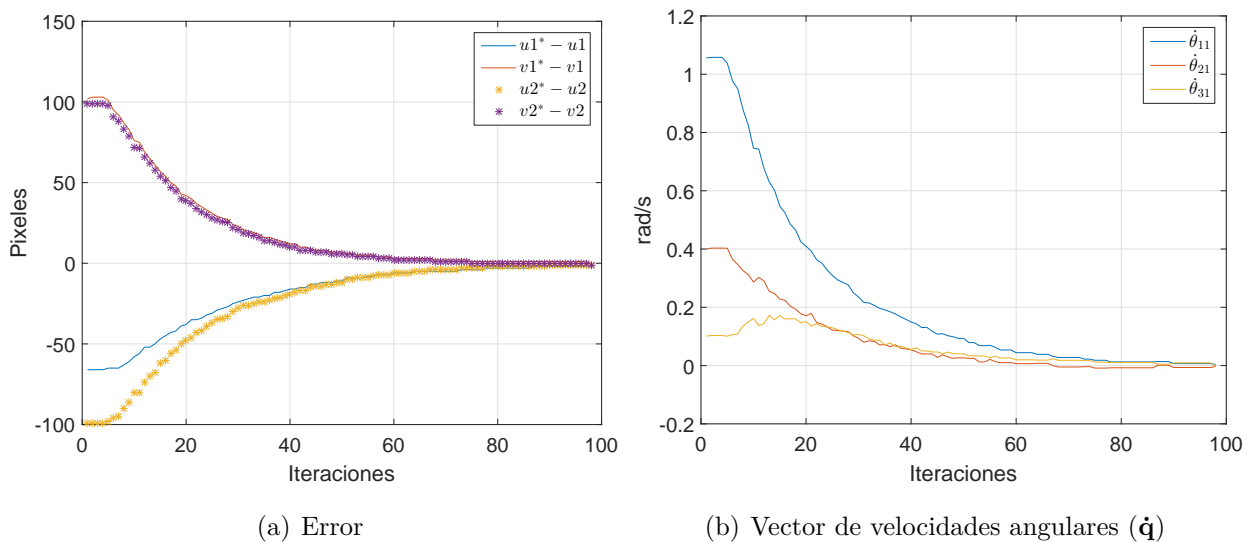


Figura 4.13. Error en píxeles y vector de velocidades angulares generados durante la prueba preliminar.

Se guarda el vector de velocidades angulares $\dot{\mathbf{q}} = \begin{bmatrix} \dot{\theta}_{11} & \dot{\theta}_{21} & \dot{\theta}_{31} \end{bmatrix}$ que arroja la ley CSM en velocidades (Ec. (3.9)) en cada iteración. Posteriormente, estas velocidades son inyectadas a las tarjetas controladoras PIC-SERVO del robot Delta. En la Fig. 4.13(b), se aprecia como el vector $\dot{\mathbf{q}}$ va decreciendo de manera exponencial hacia 0 mientras el robot va llegando a la posición objetivo. En la Fig. 4.14, se observa una fotografía tomada después de que el robot Delta se posicionó en el punto objetivo (marca visual virtual). Con esto se deduce que la ley de control CSM en velocidades en primera instancia es **funcional**.

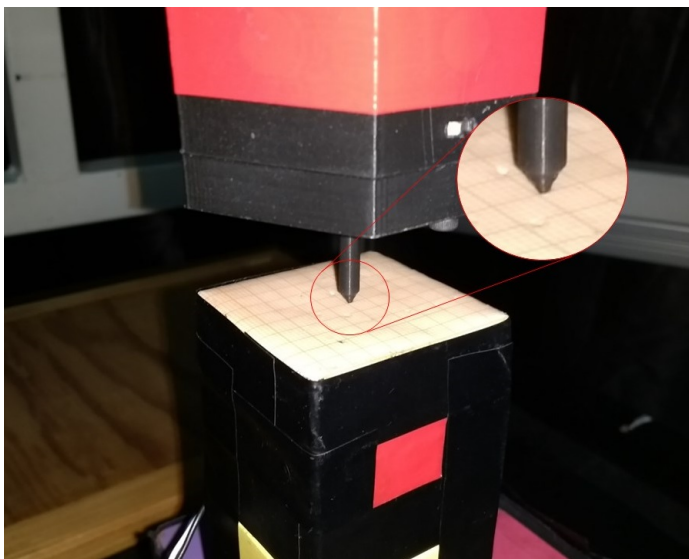


Figura 4.14. Fotografía del efector final después de finalizar la tarea de posicionamiento.

La ley de control CSM en velocidades simbolizada por la Ec. (3.9), contiene una matriz diagonal de ganancias \mathbf{G} cuyos valores (g_{ii}) están asociados con la velocidad de convergencia del error (Ec. (3.4)). Entre mayor sean los valores de la matriz, más rápido el error convergerá hacia 0. Sin embargo, puede presentar un efecto conocido en los sistemas de control como *sobre-tiro*. El efecto *sobre-tiro* consiste en una respuesta rápida del sistema sobrepasando el valor de referencia a converger (en este caso $\mathbf{e} = 0$), lo que provoca oscilaciones.

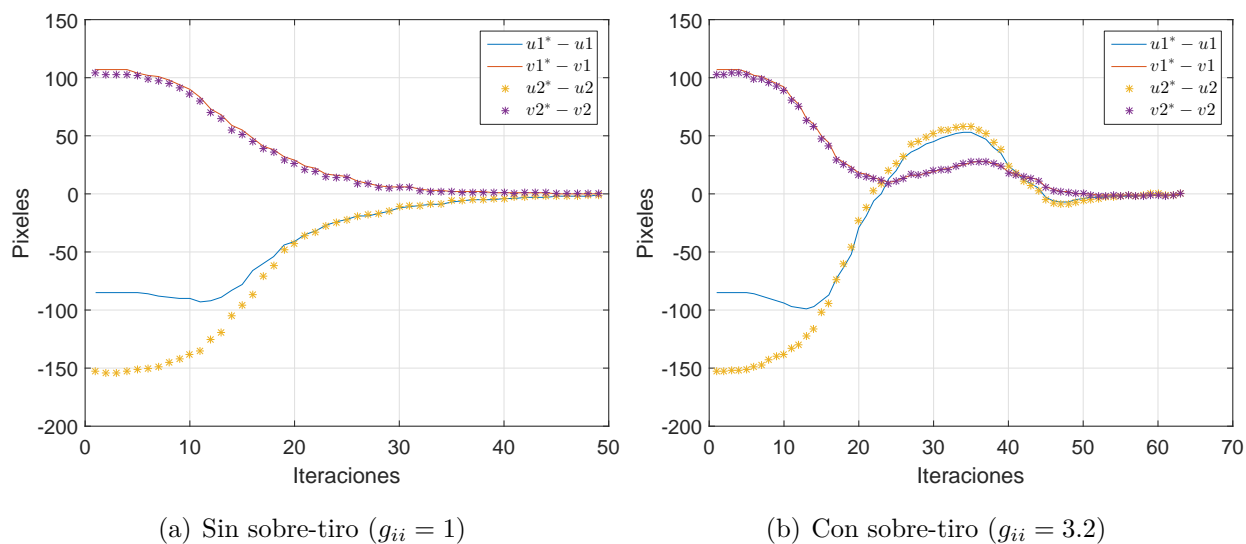


Figura 4.15. Ejemplo de una tarea de posicionamiento variando los valores de la matriz de ganancias \mathbf{G} .

Para verlo de manera gráfica, en la Fig. 4.15 se plasman dos tareas de posicionamiento con valores diferentes de \mathbf{G} . En la primera, el valor de g_{ii} es igual a 1 y se observa que el error converge de manera suave hasta llegar a 0. En la segunda, el valor de \mathbf{G} es de 3.2 y se aprecia claramente que tiene un efecto de sobre-tiro. Pero a pesar de contar con él, cumple con su convergencia a 0 aunque en mayor número de iteraciones.

Para poder realizar las pruebas experimentales de manera exitosa en todo el espacio de trabajo, se tiene que encontrar una matriz \mathbf{G} adecuada con la finalidad de que no se presente el efecto sobre-tiro. Se propone colocar el punto objetivo en 5 posiciones como se muestran en la Fig. 4.16, donde las posiciones 2 a 4 se encuentran cerca a la frontera de trabajo. Para cada una de las posiciones, se realizó tareas de posicionamiento variando \mathbf{G} desde 0.1 hasta que se presente el efecto sobre-tiro.

En la Tabla 4.8, se presentan los valores de g_{ii} antes de que se presentara el sobre-tiro, así como las iteraciones que les tomó en llegar al punto objetivo. Se aprecia que el menor de los valores de g_{ii} es de **2.3** en la posición 2 (la más cercana a las cámaras). Además para cada caso, el número de iteraciones plasmado en la Tabla 4.8 es el menor en todas las variaciones desde 0.1. Si se elige este valor, se asegura que dentro del espacio de trabajo efectivo del robot se realicen las pruebas experimentales sin sobre-tiro.

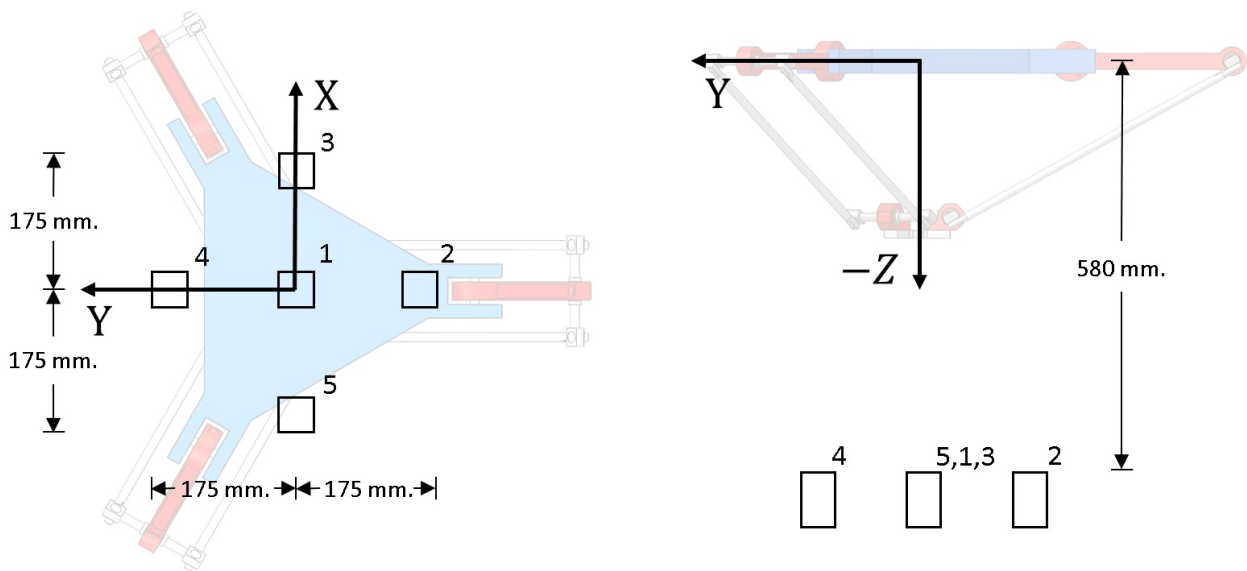


Figura 4.16. Posición de los 5 puntos objetivos para encontrar una \mathbf{G} adecuada.

Posición	g_{ii}	Iteraciones
1	3.1	26
2	2.3	44
3	2.4	49
4	2.9	36
5	2.6	32

Tabla 4.8. Valores de g_{ii} antes de que se presente el sobre-tiro en las 5 posiciones.

4.5.2. Resultados experimentales para objetivos estáticos

Al igual que en las pruebas realizadas bajo el esquema LCM - CSM y VW - CSM, se ejecutaron 10 tareas de posicionamiento en ubicaciones arbitrarias con 3 repeticiones cada una. Las condiciones para realizar estas pruebas son: posición inicial del robot ($X = 0$ mm, $Y = 0$ mm, $Z = -450$ mm), valor de la matriz \mathbf{G} de **2.3** y criterio de convergencia (Fig. 3.10) de 2 mm.

Al cumplirse el criterio de convergencia en cada tarea de posicionamiento, se mantuvo activa la ley de control durante 50 iteraciones para obtener el error medio cuadrático (Ec. (4.1)) usando el SMV, esto último para verificar el funcionamiento del SMV en la práctica.

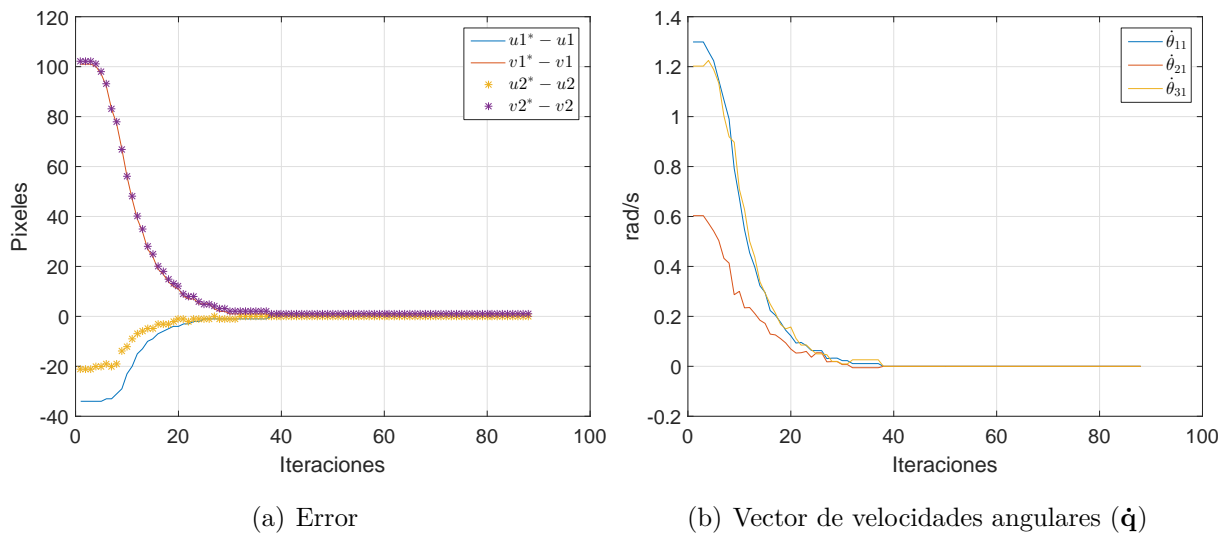


Figura 4.17. Error en pixeles y vector de velocidades angulares generados durante una prueba experimental estática de CSM en velocidades.

En la Fig. 4.17 se muestra cómo se comporta el error en pixeles y el vector de velocidades angulares ($\dot{\mathbf{q}}$) de una de las tareas de posicionamiento estático. Se observa en ambas gráficas, en la iteración 38 se cumple el criterio de convergencia y el robot mantiene su posición durante 50 iteraciones.

En la Tabla 4.10, se muestra la norma euclidiana del error en X , Y , Z del robot Delta medido por un Calibrador Vernier (tercera columna) y por el error medio cuadrático usando el SMV (cuarta columna) de las 50 iteraciones cuando se cumple el criterio de convergencia. Si se comparan los errores calculados por el SMV con su correspondiente error medido con el Calibrador Vernier, se observa que están cercanos entre sí sin sobrepasar la incertidumbre previamente conocida en la calibración del SMV (Tabla 4.6). La Tabla 4.9 muestra las estadísticas (promedio, valor máximo, valor mínimo, desviación estándar) de los errores medidos por el Calibrador Vernier (cuarta columna de la Tabla 4.10).

Error (mm)	
Promedio	1.086
Máximo	1.36
Mínimo	0.76
Desv. Estándar	0.195

Tabla 4.9. Error estadístico de la implementación de la metodología CSM en velocidades.

En esta metodología, el tiempo de posicionamiento promedio hacia el objetivo es de **1.79 segundos**. Donde cada iteración de control tiene un tiempo aproximado de **48 milisegundos**. Los resultados de precisión obtenidos con esta metodología son competentes frente a las metodologías LCM -CSM y VW - CSM de un modelo lineal de cámara por lo que la hace una opción más como técnica de control por visión para posicionamiento estático.

Posición	Repetición	Error con Vernier (mm)	Error con SMV (mm)
1	1	1.28	1.754
	2	1.32	1.65
	3	1.36	1.96
2	1	1.33	0.996
	2	1.28	1.158
	3	1.32	0.952
3	1	1.25	1.493
	2	1.22	1.499
	3	1.31	1.499
4	1	1.06	1.48
	2	1.14	1.783
	3	1.32	1.577
5	1	0.81	1.576
	2	0.88	1.957
	3	0.76	1.834
6	1	1.04	1.771
	2	1.12	1.767
	3	1.03	1.721
7	1	1.21	1.57
	2	1.12	1.608
	3	1.31	1.575
8	1	0.81	1.261
	2	0.93	1.317
	3	0.96	1.286
9	1	0.86	1.306
	2	0.91	1.451
	3	0.79	1.451
10	1	0.95	1.672
	2	0.88	1.661
	3	1.09	1.658

Tabla 4.10. Resultados de las pruebas realizadas en la implementación de la metodología CSM en velocidades.

4.6. Resultados experimentales de la metodología CSM en velocidades para objetos en movimiento

Una de las bondades de la metodología CSM en velocidades (Ec. (3.9)) es el seguimiento de objetos en movimiento, aún si no se conoce la trayectoria de este mismo. En esta sección, se muestran los resultados de pruebas realizadas con objetos en movimiento, con trayectorias lineales (objeto sobre una banda plana) y con trayectorias arbitrarias (círculo, cuadrado, ocho, espiral decreciente, zigzag).

4.6.1. Resultados experimentales para objetivos en movimiento con trayectoria lineal

Se realizaron pruebas de movimiento de un objetivo con trayectoria lineal sobre una banda transportadora (Fig. 4.3). La banda se encuentra paralela a las plataformas fija y móvil, su dirección de movimiento esta alineado aproximadamente con el eje Y y está a 800 mm del marco de referencia del robot Delta. El seguimiento del objeto se hace con 3 velocidades de la banda: 7 cm/s, 9.5 cm/s y 12 cm/s. Además, para cada velocidad de la banda se realizaron 10 tareas de seguimiento para cada una de las siguientes configuraciones:

- Valores diagonales (g_{ii}) matriz \mathbf{G} : 2.3
- Valores diagonales (g_{ii}) matriz \mathbf{G} : 2.7
- Valores diagonales (g_{ii}) matriz \mathbf{G} : 3.1

Se eligieron los valores 2.3, 2.7, 3.1 de la matriz de ganancias diagonal \mathbf{G} por lo reportado en la Tabla 4.8, donde explica los valores g_{ii} en diferentes posiciones antes de que se presente el efecto sobre-tiro. Para cada prueba de seguimiento (con su respectiva configuración de velocidad de banda, aceleración del robot y matriz \mathbf{G}) se hace con y sin la compensación de velocidad ($\dot{\mathbf{s}}^*$) usando el Filtro de Kalman, lo que da un total de 180 tareas de seguimiento. Las matrices de covarianza del error \mathbf{Q} y \mathbf{R} (Ec. (3.12)) del Filtro de Kalman se mantuvieron constantes durante todas las pruebas de movimiento y se ajustaron previamente de manera “*off-line*” [49].

La finalidad de realizar estas pruebas en movimiento, es medir el error de seguimiento en pixeles (Ec. (3.4)) y tridimensional (usando el SMV) del robot con respecto a un objetivo en un intervalo dado. Este intervalo es de 350 mm sobre la banda como se observa en la Fig. 4.18.

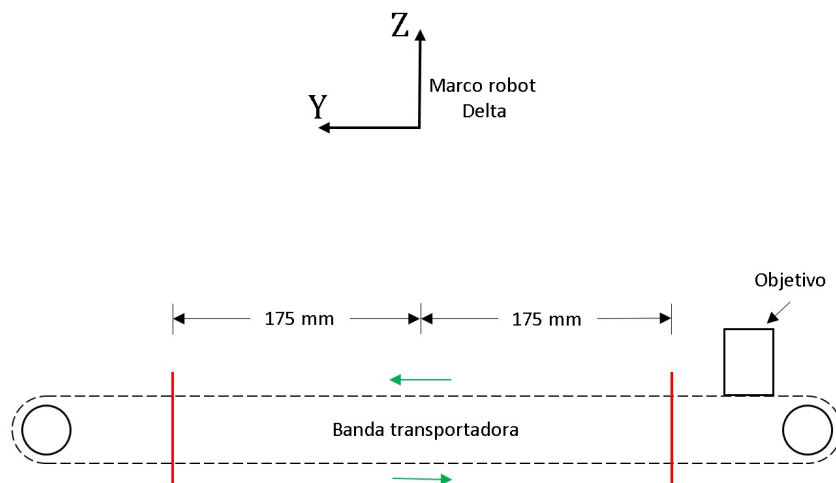
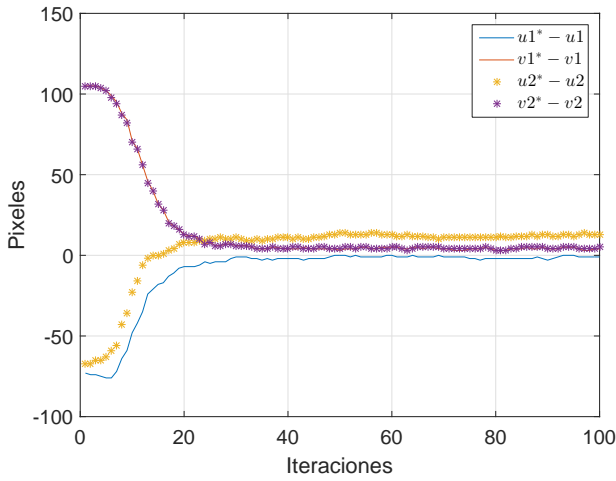


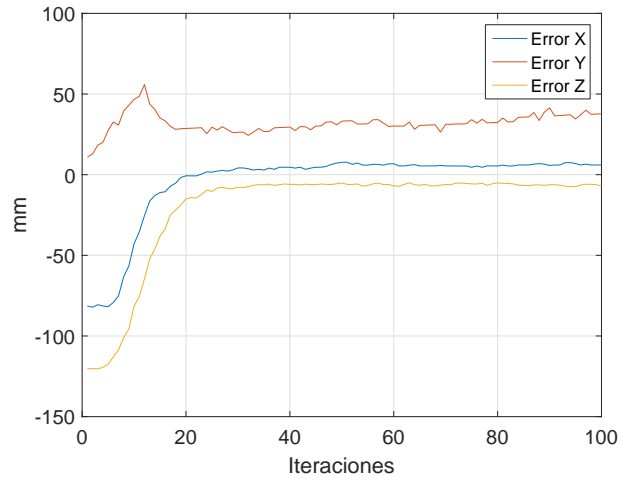
Figura 4.18. Tamaño del intervalo para las pruebas de experimentales de objetivos con trayectoria lineal.

La posición inicial del robot Delta para las pruebas experimentales es $X = 0$ mm, $Y = -175$ mm, $Z = -450$ mm. De manera automática, el algoritmo de control inicia el movimiento del robot Delta cuando el punto objetivo haya cruzado el inicio del intervalo ($Y = -175$ mm) y detiene el robot cuando el punto objetivo haya alcanzado el valor de $Y = 175$ mm. Se decidió hacerlo de esta manera debido a la pericia requerida para iniciar y detener la banda justo en el intervalo indicado, y al hacerlo de manera automática, se asegura que en todas las pruebas se tengan la mismas condiciones.

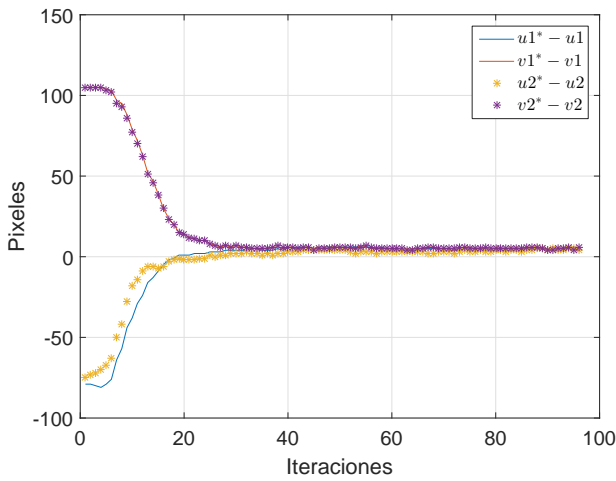
En la Fig. 4.19 se muestra el error en pixeles y tridimensional (medido por el SMV) durante una tarea de seguimiento con las siguientes condiciones: $g_{ii} = 2.3$, velocidad de la banda = 7 cm/s, sin y con la compensación del Filtro de Kalman.



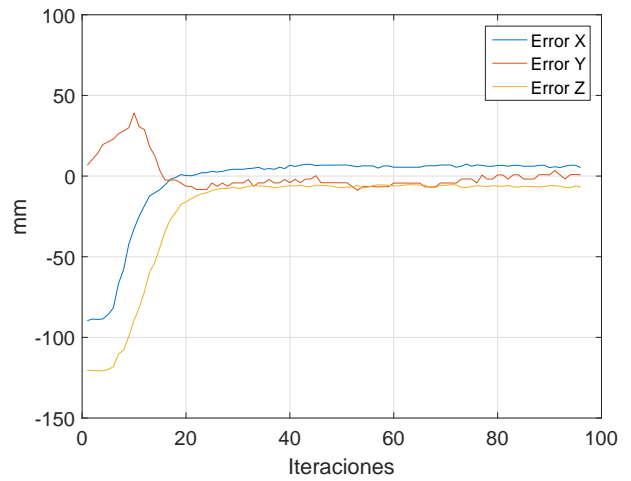
(a) pixeles sin compensación de velocidad



(b) Tridimensional (SMV) sin compensación de velocidad



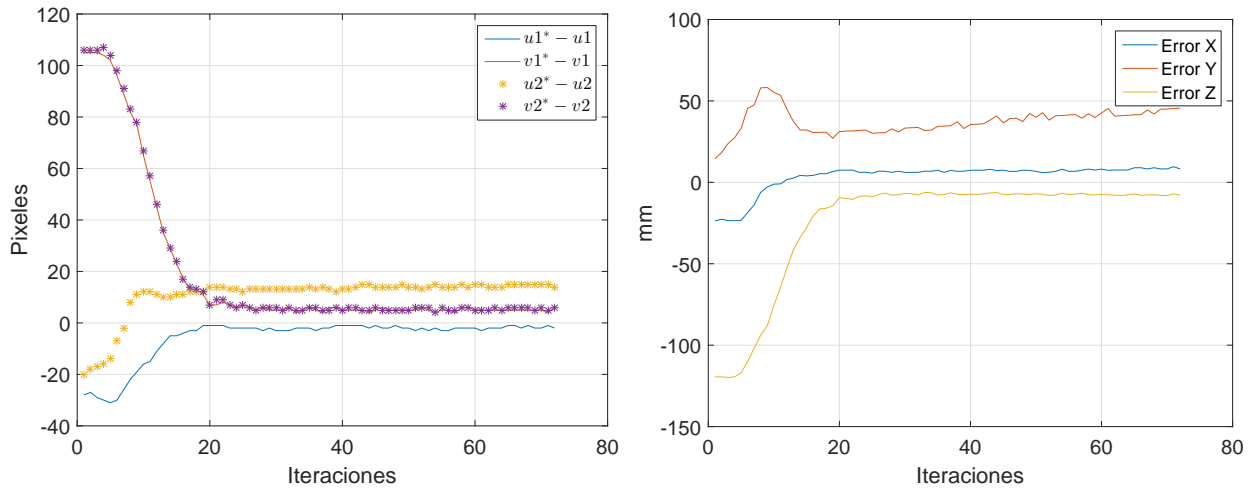
(c) pixeles con Filtro de Kalman



(d) Tridimensional (SMV) con Filtro de Kalman

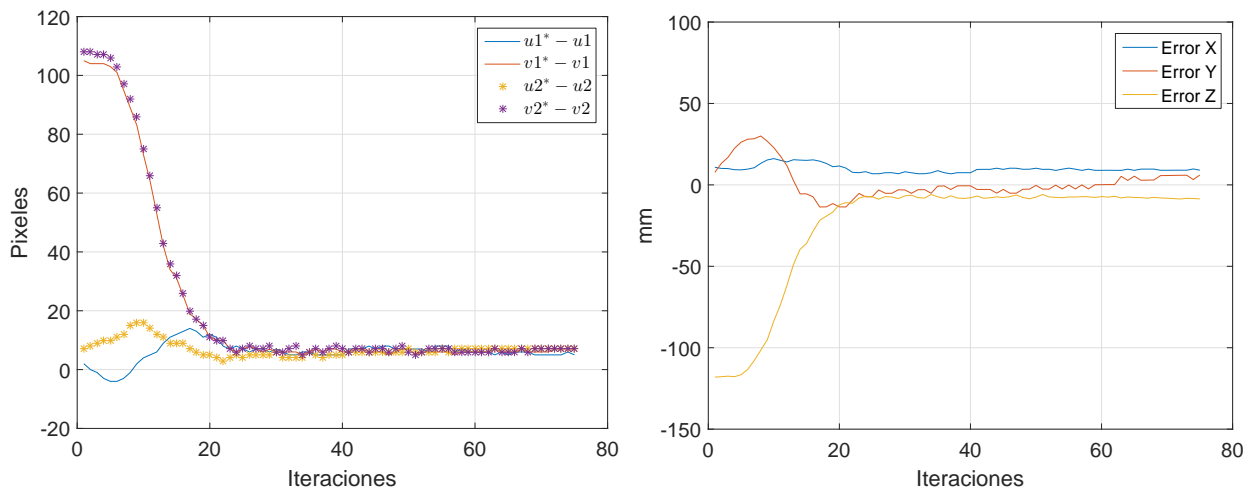
Figura 4.19. Gráficas del error durante una tarea de seguimiento de un objetivo con una velocidad de 7 cm/s.

En la Fig. 4.20 se muestra el error en pixeles y tridimensional (medido por el SMV) durante una tarea de seguimiento con las siguientes condiciones: $g_{ii} = 2.7$, velocidad de la banda = 9.5 cm/s, sin y con la compensación del Filtro de Kalman.



(a) pixeles sin compensación de velocidad

(b) Tridimensional (SMV) sin compensación de velocidad

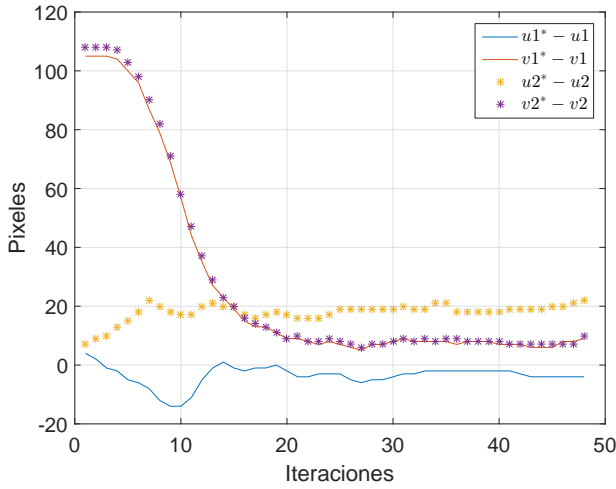


(c) pixeles con Filtro de Kalman

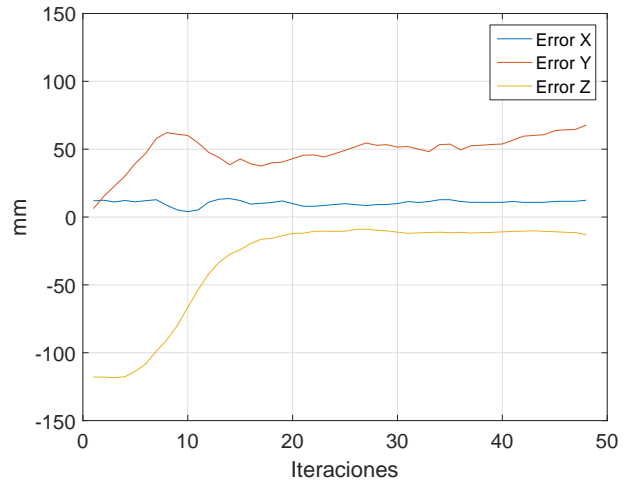
(d) Tridimensional (SMV) con Filtro de Kalman

Figura 4.20. Gráficas del error durante una tarea de seguimiento de un objetivo con una velocidad de 9.5 cm/s.

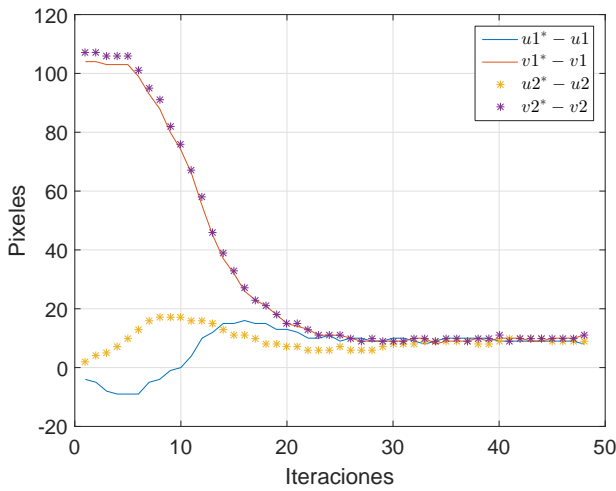
En la Fig. 4.21 se muestra el error en pixeles y tridimensional (medido por el SMV) durante una tarea de seguimiento con las siguientes condiciones: $g_{ii} = 3.1$, velocidad de la banda = 12 cm/s, sin y con la compensación del Filtro de Kalman.



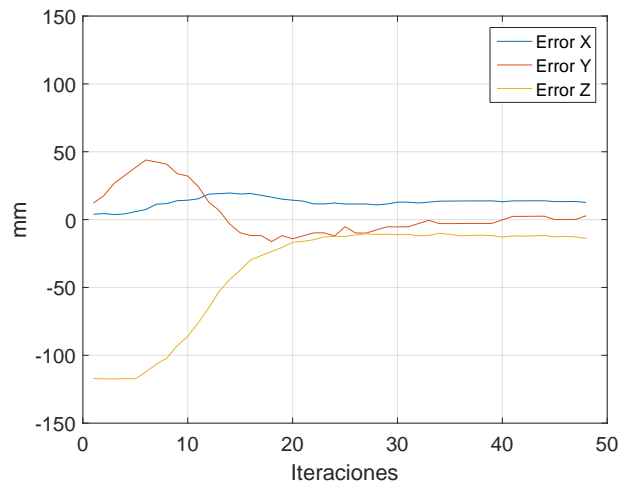
(a) pixeles sin compensación de velocidad



(b) Tridimensional (SMV) sin compensación de velocidad



(c) pixeles con Filtro de Kalman



(d) Tridimensional (SMV) con Filtro de Kalman

Figura 4.21. Gráficas del error durante una tarea de seguimiento de un objetivo con una velocidad de 12 cm/s.

Como se observa en las Fig. 4.19 - 4.21, el uso de la compensación de velocidad con el Filtro de Kalman ayuda considerablemente a disminuir el error (píxeles, tridimensional) en la tarea de seguimiento, sin embargo, conforme se aumenta la velocidad del objetivo también crece el error de seguimiento. Además, el error cuenta con un periodo de transición en las primeras iteraciones hasta alcanzar un estado estable. Este periodo de transición es debido a que el robot parte de una posición inicial y comienza moverse hasta llegar al objetivo en movimiento aún cuando lo llegue o no a alcanzar.

A continuación se presenta el promedio del error medio cuadrático (Ec. (4.1)) de las 180 pruebas de seguimiento realizadas con sus respectivas configuraciones. El error medio cuadrático se obtiene de la mitad de iteraciones totales hacia adelante (error en estado estable) que le tomó a la tarea de seguimiento en el intervalo dado (Fig. 4.18).

Velocidad banda	g_{ii}	Error seguimiento (mm)	
		Sin compensación	Con compensación (FK)
7 cm/s	2.3	34.3807	11.2137
	2.7	29.8393	9.6279
	3.1	26.0212	8.8899
9.5 cm/s	2.3	45.6601	14.8628
	2.7	40.7828	12.3435
	3.1	37.9386	11.7613
12 cm/s	2.3	65.6163	27.8823
	2.7	61.5556	20.9840
	3.1	58.1373	18.6458

Tabla 4.11. Error estadístico de las pruebas de seguimiento realizadas a un objetivo en diferentes velocidades

En las Tabla 4.11, se observan los errores de seguimiento promedio variando la velocidad del objetivo de 7 a 12 cm/s. Se aprecia que para todos los casos disminuye el error por lo menos un 60 % al aplicarle la compensación de velocidad usando el Filtro de Kalman. También se distingue que al aumentar los valores g_{ii} de la matriz de ganancias \mathbf{G} (2.3 a 3.1) se obtiene una mejora de mínimo 6 % en el error.

Cuando se aumenta la velocidad del objetivo también incrementa el error de seguimiento sobre éste. Esto se le atribuye probablemente a una mala estimación del término \dot{s}^* (Ec. (3.9)) por parte del Filtro de Kalman debido a la presencia de demasiado ruido en la obtención del centroide en pixeles del punto objetivo. El desarrollo, implementación y evaluación de otras técnicas de estimación de velocidad se dejan para un trabajo a futuro.

4.6.2. Resultados experimentales para objetivos en movimiento con trayectoria arbitraria

Se realizaron pruebas de movimiento de un objetivo con trayectorias arbitrarias como: círculo, cuadrado, espiral decreciente, ocho y zigzag, inscritas en una superficie cuadrada de 350 mm de lado en el plano X-Y del robot Delta con distancia en el eje -Z de 600 mm. La velocidad del objetivo es relativamente baja debido a que se moverá a mano alzada para formar las trayectorias deseadas. Las condiciones para realizar estas pruebas son: posición inicial del robot ($X = 0$ mm, $Y = 0$ mm, $Z = -450$ mm) y valor de la matriz \mathbf{G} de 2.7 .

En la Tabla 4.12 se muestran los resultados experimentales de las 5 tareas de seguimiento de un objetivo con trayectoria arbitraria. En la primera columna de la tabla, se observa el error medio cuadrático en el seguimiento en estado estable calculado por el Sistema de Medición por Visión. En la segunda columna de la tabla, se observa el error en norma euclidiana medido en X , Y , Z del robot por un Calibrador Vernier cuando el objetivo se detuvo. En la tercera columna de la tabla se muestra la medición por parte del SMV cuando el objetivo se detuvo, estas mediciones se encuentran dentro de la incertidumbre conocida del SMV.

	Error (mm)		
	Seguimiento	Final	
	SMV	Vernier	SMV
Círculo	7.521	1.44	1.56
Cuadrado	10.471	1.41	1.039
Espiral	9.021	1.23	1.436
Ocho	9.661	1.16	0.942
Zig-zag	10.788	1.29	2.052

Tabla 4.12. Error en el seguimiento y final de un objetivo con trayectorias arbitrarias.

En la Fig. 4.22, se muestra la posición tridimensional del objetivo, calculada por el SMV, en las 5 tareas de seguimiento con trayectoria arbitraria.

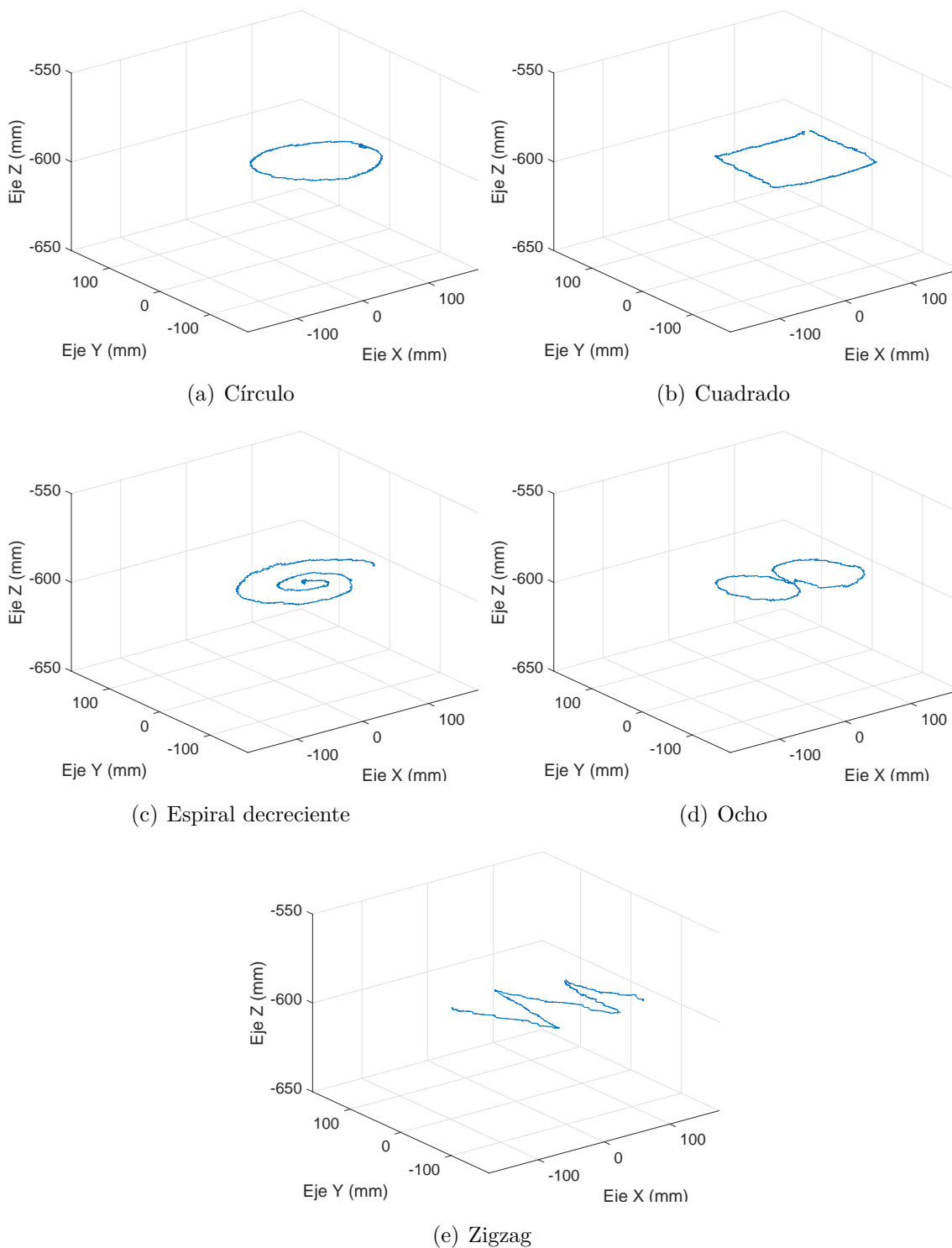


Figura 4.22. Vista isométrica de la posición calculada por el SMV del objetivo con trayectorias arbitrarias.

4.7. Conclusiones del capítulo 4

En este capítulo se mostraron los resultados de las implementaciones de las metodologías LCM - CSM, VW - CSM de un modelo lineal de cámara y la nueva técnica de control CSM en velocidades para manipular el robot paralelo tipo Delta hacia un objetivo estático. Además de implementar el control CSM en velocidad para el seguimiento de objetivos con trayectorias lineal y arbitraria.

Los resultados obtenidos por las 3 metodologías para posiciones estáticas fueron muy similares, en la metodología LCM - CSM se obtuvo un error promedio de 0.997 mm con una desviación estándar de 0.151 mm. Este resultado es una ligera mejora con respecto a lo reportado por Lopez-Lara en [45]. Esto se debe al mayor número de muestras (410 con respecto a 42) recopiladas en el “*Pre-plan*”. En la metodología VW - CSM de un modelo lineal de cámara se obtuvo un error promedio de 1.010 con una desviación estándar de 0.154 mm, resultado muy similar a lo obtenido con LCM - CSM. En la metodología CSM en velocidades se obtuvo un error promedio en el posicionamiento estático de 1.086 mm con una desviación estándar 0.195 mm. Este resultado es compatible con lo obtenido en las metodologías LCM - CSM y VW - CSM de un modelo lineal de cámara.

Los resultados de la metodología CSM en velocidades para seguimiento de objetivos son mayores con respecto al posicionamiento estático. Los errores de seguimiento para trayectorias lineales son mayores con respecto a los de trayectorias arbitrarias, esto debido a que los seguimientos lineales se hicieron con velocidades mayores. Sin embargo, CSM en velocidades demostró que puede seguir cualquier trayectoria sin ningún problema y que al detenerse el objetivo, el robot puede llegar con la misma precisión como si hubiera sido una tarea de posicionamiento estático.

Una de las causas a que se deban los resultados en seguimiento de objetivos, es una mala estimación de la velocidad por parte del Filtro de Kalman debido al ruido generado por el cálculo del centroide del objetivo, además, de los límites del prototipo (Parallax LKF-2040) y ajuste de ganancias en las tarjetas de control (PIC-SERVO). A pesar de los errores, los resultados muestran la factibilidad de CSM en velocidades y, más importante, se demostró la estabilidad del sistema controlado.

Conclusiones

Se cumplieron con los objetivos propuestos de desarrollar e implementar el control CSM basado en velocidades para un robot paralelo tipo Delta modelo Parallax LKF-2040. El nuevo esquema CSM en velocidades es una metodología de control basado en visión innovadora inspirada en LCM -CSM. Mientras LCM - CSM está basada en control punto a punto, la nueva metodología utiliza control en velocidad lo cual representa la principal innovación y aportación de este trabajo. CSM en velocidades tiene la capacidad de manipular objetivos estáticos y en movimiento (aún cuando no se conozca la trayectoria de éste) con un desplazamiento suave del robot. Además, CSM en velocidades es una metodología de lazo cerrado que permite ofrecer garantías de estabilidad lo cual no es posible en los demás desarrollos de CSM.

Cabe destacar, que la técnica de control Servo Visual (basada en imagen) tiene la misma esencia que CSM en velocidades, sin embargo, Servo Visual usa el Jacobiano de Imagen para la relación de velocidades previamente descrita. La diferencia entre el Jacobiano de Imagen con respecto a la matriz CSM en velocidades, es que en la primera se tiene que volver a calcular en cada ciclo de control porque depende de la posición en plano de imagen y la distancia en el eje Z de cámara del punto objetivo, mientras que la segunda es constante si el posicionamiento es dentro del volumen de puntos tomados en el “*Pre-plan*”.

Los esquemas de control tradicionales son difícilmente aplicables al robot Delta (y a los robots paralelos en general) debido a problemas abiertos en la literatura como: la obtención

de su modelo cinemático directo y dinámico, análisis de singularidades, entre otros. Comúnmente el robot Delta es usado en aplicaciones industriales para tareas “*pick and place*”, sin embargo, este tipo de aplicaciones llegan a tener cambios debido a diferente manejo de productos sobre la misma línea, especificaciones por parte del cliente y desplazamiento de los objetivos. Las técnicas de control por visión reemplazan los esquemas de control tradicionales de robots, en donde previamente se cargan una serie de movimientos pre-programados y ayudan a resolver problemas abiertos en la literatura para el control del robot (cinemática directa, dinámica, etc).

Se implementaron las metodologías LCM -CSM y VW - CSM de un modelo lineal de cámara en el robot Delta. Para evaluar la precisión de estas implementaciones se realizaron pruebas de posicionamiento estático. El error promedio en la aplicación de LCM -CSM es de 0.997 mm con una desviación estándar de 0.151 mm y una duración promedio de 6.5 segundos. El error promedio en la aplicación de VW - CSM de un modelo lineal de cámara es de 1.010 mm con una desviación estándar de 0.154 mm y una duración promedio de 1.3 segundos. La diferencia de tiempos entre ambas metodologías es debido a que LCM - CSM requiere de adquirir muestras adicionales (locales) cerca del punto objetivo para aumentar la precisión en el posicionamiento. Mientras que VW - CSM de un modelo lineal de cámara usa las muestras globales para hacer una correlación con el objetivo, por lo que no requiere de muestras locales.

El error promedio en la implementación de CSM en velocidades para manipular objetivos estáticos es de 1.083 mm con una desviación estándar de 0.195 mm y una duración de posicionamiento promedio de 1.79 milisegundos. Estos resultados están a la par de las metodologías LCM - CSM y VW - CSM de un modelo lineal de cámara y puede ser competitivo en un buen número de aplicaciones industriales (por ejemplo “*pick and place*”). En cuanto a tareas de seguimiento de objetos móviles, los resultados arrojan un error de seguimiento de 8.889 mm para objetivos con trayectoria lineal y 7.521 mm con trayectoria arbitraria. A pesar de que el error de seguimiento es relativamente grande, CSM en velocidades demostró que al detenerse el objetivo después de realizar una serie de movimientos aleatorios puede llegar con la misma precisión como si fuera una tarea de posicionamiento estática.

El sistema comercial “*Pick Master*” de ABB es capaz de identificar y transferir objetos con posiciones aleatorias sobre una banda transportadora (es decir para trayectorias rectas a velocidad constante y conocida) en 50 a 200 milisegundos. Si la metodología CSM se compara con este sistema, puede llegar a ser competitivo debido a que puede identificar y

transferir objetos con posiciones aleatorias, sin embargo, el desempeño de la metodología dependerá del hardware a implementar (robot, computadora, cámaras, etc). Por ejemplo, los resultados mostrados en la implementación de CSM en velocidades mejorarían si se aplica a una plataforma comercial (IRB 360 FlexPicker de ABB por ejemplo) ya que en este trabajo se aplicó a una plataforma académica (Parallix LKF-2040). Además, CSM en velocidades tiene la ventaja sobre el sistema “*Pick Master*” que es el seguimiento de objetivos en movimiento con trayectorias aleatorias tanto en posición como en velocidad, adicionalmente, CSM en velocidades no requiere el conocimiento previo de la velocidad del objeto.

Contribuciones

Enseguida se enlistan las principales contribuciones obtenidas durante el desarrollo del presente trabajo.

1. Desarrollo de una nueva variante de la metodología Manipulación en Espacio de Cámara de un Modelo Lineal de Cámara (LCM - CSM) para el control en velocidad de un robot paralelo tipo Delta (CSM en velocidades). *Contribución central del presente trabajo.*
 2. Implementación y validación experimental de la metodología CSM en velocidades a la plataforma Parallix LKF-2040 para la manipulación de objetivos estáticos y en movimiento.
 3. Publicación del trabajo: “Control en velocidad, basado en visión, de un robot paralelo tipo Delta”, **Antonio Loredo**, Mauro Maya, Antonio Cárdenas, Emilio González-Galván, Alejandro González. XIX Congreso Mexicano de Robótica. Mazatlán, Sinaloa, México, Noviembre 2017.
 4. Implementación de la metodología VW - CSM para el control del robot paralelo tipo Delta usando un modelo lineal de cámara (LCM - CSM). Originalmente VW-CSM fue desarrollada a partir del esquema OFL - CSM a un robot serial.
 5. Desarrollo de un sistema de medición tridimensional basado en visión usando la metodología VW - CSM de un modelo lineal de cámara.
-

Trabajo a futuro

Al término de este trabajo se identifican algunas líneas en la que se puede mejorar la implementación de las técnicas y métodos desarrollados en este trabajo.

- Llevar la implementación del control CSM en velocidades a robots seriales.
 - Utilizar otras técnicas, aparte del Filtro de Kalman, para la estimación de la velocidad del objetivo.
 - Implementar otro esquema de control, con su respectivo análisis de estabilidad, que tome en cuenta la variación de los parámetros de visión.
-

Bibliografía

- [1] Z. Pandilov y V. Dukovski, “Comparison of the characteristics between serial and parallel robots,” *Acta Technica Corviniensis-Bulletin of Engineering*, vol. 7, no. 1, p. 143, 2014.
- [2] ABB Company. <http://new.abb.com/>, 2017. [En línea; acceso Diciembre-2017].
- [3] Sun Aerosys Co. Ltd. <https://eomyj.en.ec21.com/>, 2017. [En línea; acceso Diciembre-2017].
- [4] M. Maya, E. Castillo, A. Lomelí, E. González-Galván, y A. Cárdenas, “Workspace and payload-capacity of a new reconfigurable delta parallel robot,” *International Journal of Advanced Robotic Systems*, vol. 10, no. 1, p. 56, 2013.
- [5] I. Bonev, “The true origins of parallel robots.” <http://www.parallemic.org/Reviews/Review007.html>, 2003. [En línea; acceso Agosto-2017].
- [6] V. Gough, “Universal tyre test machine,” *Proc. 9th Int. Technical Congr. FISITA, London, 1962*, 1962.
- [7] D. Stewart, “A platform with six degrees of freedom,” *Proceedings of the institution of mechanical engineers*, vol. 180, no. 1, pp. 371–386, 1965.
- [8] L. Ángel Silva, *Control visual de robots paralelos: análisis, desarrollo y aplicación a la plataforma RoboTennis*. Tesis de Doctorado, Universidad Politécnica de Madrid, 2005.
- [9] J.-P. Merlet, *Parallel robots*. Springer Science & Business Media, 2006.

- [10] R. Clavel, “Delta, a fast robot with parallel geometry,” in *Proceedings of the 18th International Symposium on Industrial Robots* (C. W. Burckhardt, ed.), (New York), pp. 91–100, Springer-Verlag, 1988.
- [11] I. Bonev, “Delta parallel robot—the story of success.” <http://www.parallemic.org/Reviews/Review002.html>, 2001. [En línea; acceso Agosto-2017].
- [12] A. Codourey, R. Clavel, y C. Burckhardt, “Control algorithm and controller for the direct drive delta robot,” *IFAC Proceedings Volumes*, vol. 24, no. 9, pp. 543–549, 1991.
- [13] K. Miller y R. Clavel, “The lagrange-based model of delta-4 robot dynamics,” *Robotersysteme*, vol. 8, no. 1, pp. 49–54, 1992.
- [14] K. Miller, “Experimental verification of modeling of delta robot dynamics by direct application of hamilton’s principle,” in *Robotics and Automation, 1995. Proceedings., 1995 IEEE International Conference on*, vol. 1, pp. 532–537, IEEE, 1995.
- [15] M. López, E. Castillo, G. García, y A. Bashir, “Delta robot: inverse, direct, and intermediate jacobians,” *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 220, no. 1, pp. 103–109, 2006.
- [16] R. E. Stamper, L.-W. Tsai, y G. C. Walsh, “Optimization of a three dof translational platform for well-conditioned workspace,” in *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*, vol. 4, pp. 3250–3255, IEEE, 1997.
- [17] K. Miller, “Synthesis of a manipulator of the new uwa robot,” in *Proc. of Australian Conf. on Robotics and Automation, Brisbane, Australian*, pp. 228–233, 1999.
- [18] E. C. Castañeda, A. O. V. Rodríguez, y A. G. Hernández, “Desarrollo del Robot Paralelo Parallax LKF-2400,” *Ingenio Magno*, vol. 3, no. 1, 2013.
- [19] L. A. Silva, *Control visual de robots paralelos: análisis, desarrollo y aplicación a la plataforma RoboTennis*. Tesis de Doctorado, Universidad Politécnica de Madrid, 2005.
- [20] Z. Pandilov y V. Dukovski, “Several open problems in parallel robotics,” *Acta Technica Corviniensis-Bulletin of Engineering*, vol. 4, no. 3, p. 77, 2011.
- [21] A. K. Dash, I.-M. Chen, S. H. Yeo, y G. Yang, “Workspace generation and planning singularity-free path for parallel manipulators,” *Mechanism and Machine Theory*, vol. 40, no. 7, pp. 776–805, 2005.

- [22] J. Kovecses, J.-C. Piedbœuf, y C. Lange, “Methods for dynamic models of parallel robots and mechanisms,” in *Proceedings of the WORKSHOP on Fundamental Issues and Future Research Directions for Parallel Mechanisms and Manipulators October*, pp. 3–4, 2002.
- [23] J. Gallardo, J. Rico, A. Frisoli, D. Checcacci, y M. Bergamasco, “Dynamics of parallel manipulators by means of screw theory,” *Mechanism and machine theory*, vol. 38, no. 11, pp. 1113–1131, 2003.
- [24] C. Gosselin y J. Angeles, “Singularity analysis of closed-loop kinematic chains,” *IEEE transactions on robotics and automation*, vol. 6, no. 3, pp. 281–290, 1990.
- [25] J. Wang y O. Masory, “On the accuracy of a stewart platform. i. the effect of manufacturing tolerances,” in *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*, pp. 114–120, IEEE, 1993.
- [26] P. Renaud, N. Andreff, F. Marquet, y P. Martinet, “Vision-based kinematic calibration of a h4 parallel mechanism,” in *Robotics and Automation, 2003. Proceedings. ICRA’03. IEEE International Conference on*, vol. 1, pp. 1191–1196, IEEE, 2003.
- [27] A. Nahvi, J. M. Hollerbach, y V. Hayward, “Calibration of a parallel robot using multiple kinematic closed loops,” in *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, pp. 407–412, IEEE, 1994.
- [28] G. J. Agin, *Real time control of a robot with a mobile camera*. SRI International, 1979.
- [29] S. Hutchinson, G. D. Hager, y P. I. Corke, “A tutorial on visual servo control,” *IEEE transactions on robotics and automation*, vol. 12, no. 5, pp. 651–670, 1996.
- [30] L. E. Weiss, A. C. Sanderson, y C. P. Neuman, “Dynamic visual servo control of robots: an adaptive image-based approach,” in *Robotics and Automation. Proceedings. 1985 IEEE International Conference on*, vol. 2, pp. 662–668, IEEE, 1985.
- [31] R. Hartley y A. Zisserman, *Multiple view geometry in computer vision*. Cambridge University Press, 2003.
- [32] Y. Ma, S. Soatto, J. Kosecka, y S. S. Sastry, *An invitation to 3-d vision: from images to geometric models*, vol. 26. Springer Science & Business Media, 2012.
- [33] P. Pinto y P. Lizardo, *Control visual basado en características de un sistema articulado. Estimación del jacobiano de la imagen utilizando múltiples vistas*. Tesis de Doctorado, Universidad Politécnica de Madrid, 2007.

- [34] E. J. Gonzalez-Galvan y S. B. Skaar, “Efficient camera-space manipulation using moments,” in *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, vol. 4, pp. 3407–3412, IEEE, 1996.
- [35] S. B. Skaar, W. H. Brockman, y R. Hanson, “Camera-space manipulation,” *The International Journal of Robotics Research*, vol. 6, no. 4, pp. 20–32, 1987.
- [36] S. B. Skaar, W. H. Brockman, y W. Jang, “Three-dimensional camera space manipulation,” *The International Journal of Robotics Research*, vol. 9, no. 4, pp. 22–39, 1990.
- [37] S. B. Skaar, I. Yalda-Mooshabad, y W. H. Brockman, “Nonholonomic camera-space manipulation,” *IEEE Transactions on Robotics and Automation*, vol. 8, no. 4, pp. 464–479, 1992.
- [38] A. Cárdenas, B. Goodwine, S. Skaar, y M. Seelinger, “Vision-based control of a mobile base and on-board arm,” *The International Journal of Robotics Research*, vol. 22, no. 9, pp. 677–698, 2003.
- [39] E. J. González-Galván, S. R. Cruz-Ramírez, M. J. Seelinger, y J. J. Cervantes-Sánchez, “An efficient multi-camera, multi-target scheme for the three-dimensional control of robots using uncalibrated vision,” *Robotics and Computer-Integrated Manufacturing*, vol. 19, no. 5, pp. 387–400, 2003.
- [40] B. Horn, *Robot vision*. MIT press, 1986.
- [41] E. J. Gonzalez-Galvan, S. B. Skaar, U. A. Korde, y W. Chen, “Application of a precision-enhancing measure in 3d rigid-body positioning using camera-space manipulation,” *The International Journal of Robotics Research*, vol. 16, no. 2, pp. 240–257, 1997.
- [42] J. M. Rendón-Mancha, A. Cárdenas, M. A. García, y B. Lara, “Robot positioning using camera-space manipulation with a linear camera model,” *IEEE Transactions on Robotics*, vol. 26, no. 4, pp. 726–733, 2010.
- [43] E. J. González-Galván, F. Pazos-Flores, S. B. Skaar, y A. Cardenas-Galindo, “Camera pan/tilt to eliminate the workspace-size/pixel-resolution tradeoff with camera-space manipulation,” *Robotics and Computer-Integrated Manufacturing*, vol. 18, no. 2, pp. 95–104, 2002.
- [44] E. Coronado, M. Maya, A. Cardenas, O. Guarneros, y D. Piovesan, “Vision-based control of a delta parallel robot via linear camera-space manipulation,” *Journal of Intelligent & Robotic Systems*, vol. 85, no. 1, pp. 93–106, 2017.

- [45] G. J. López-Lara, “Control basado en visión de un robot paralelo tipo delta por medio del método manipulación en espacio de cámara,” Tesis de Maestría, Universidad Autónoma de San Luis Potosí, 2017.
- [46] Y. Liu, D. Shi, y S. Baard Skaar, “Robust industrial robot real-time positioning system using vw-camera-space manipulation method,” *Industrial Robot: An International Journal*, vol. 41, no. 1, pp. 70–81, 2014.
- [47] E. J. González-Galván, “Notas sobre teoría de estimación aplicada a la robótica.” Universidad Autónoma de San Luis Potosí, Notas de curso, 2017.
- [48] S. R. Buss, “Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods,” *IEEE Journal of Robotics and Automation*, vol. 17, no. 1-19, p. 16, 2004.
- [49] G. Bishop y G. Welch, “An introduction to the kalman filter,” *Proc of SIGGRAPH, Course*, vol. 8, no. 27599-23175, p. 41, 2001.
- [50] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Journal of basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.
- [51] L. Jeffrey Kerr, “Pic-servo motion control.” <http://www.jrkerr.com/>, 2017. [En línea; acceso Diciembre-2017].
- [52] M. Galassi y J. Theiler, “Gnu scientific library.” <https://www.gnu.org/software/gsl/>, 1996. [En línea; acceso Junio-2017].
- [53] I. Intel Corporation, Willow Garage, “Open source computer vision library.” <http://opencv.org/>, 2000. [En línea; acceso Julio-2016].
- [54] J. A. Lomelí-Rodríguez, “Análisis y diseño de un mecanismo para la reconfiguración de un robot paralelo tipo delta,” Tesis de Maestría, Universidad Autónoma de San Luis Potosí, 2011.
- [55] J. F. Kenney y E. S. Keeping, *Mathematics of Statistics: Part One*. D. Van Nostrand Company, 3er ed., 1962.