



**CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS
AVANZADOS DEL INSTITUTO POLITÉCNICO NACIONAL**

UNIDAD ZACATENCO

DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

SECCIÓN DE MECATRÓNICA

Fusión de Sensores Visuales e Inerciales para Navegación Autónoma de Cuatrirrotores

Tesis que presenta

Ángel Alejandro Gómez Casasola

Para obtener el grado de

Doctor en Ciencias

En la especialidad de

Ingeniería Eléctrica

Director de Tesis

Dr. Hugo Rodríguez Cortés

[Intencionalmente dejada en blanco]

Agradecimientos

- Al Conacyt por apoyarme con el recurso económico durante mi programa de doctorado.

[Intencionalmente dejada en blanco]

Índice general

Índice de figuras	IX
Índice de tablas	XI
Resumen	XII
Abstract	XIII
1. Introducción	1
1.1. Antecedentes	1
1.2. Estado del Arte	3
1.3. Justificación	14
1.4. Objetivos	15
1.4.1. Objetivo General	15
1.4.2. Objetivos Particulares	16
1.5. Organización de la Tesis	16
2. Materiales y Métodos	19
2.1. Dinámica del Cuadrirrotor	19
2.2. Sensores Disponibles	21
2.2.1. Posición Escalada	22
2.2.2. Aceleración Específica	22
2.2.3. Sistemas de Referencia de Orientación y Rumbo	23
2.2.4. Rapidez Vertical	24
2.3. Metodología de Inmersión e Invarianza	25
3. Diseño del Observador y Estimador	29
3.1. Problema de Observación	29

3.2. Observador de Velocidad	30
3.3. Estimador del Factor de Escala	32
4. Simulaciones Numéricas y Experimentación	39
4.1. Matlab-Simulink	39
4.2. Gazebo	41
4.2.1. Cálculo de μ	42
4.2.2. Algoritmo de Visión Monocular	42
4.2.3. Seguimiento de Trayectoria Usando el Estimador del Factor de Escala	43
4.3. Experimentos	51
4.3.1. Cuadrirrotor Experimental	51
4.3.2. Estimador con AR.Drone	52
5. Conclusiones y Trabajos a Futuro	57
5.1. Conclusiones	57
5.2. Trabajos a Futuro	58

Índice de figuras

1-1. Multirrotores de 6 hélices.	2
1-2. Multirrotores más comunes.	3
1-3. Clasificación de aeronaves según su principio de vuelo y modo de propulsión.	4
1-4. Diagrama simplificado de un acelerómetro.	6
1-5. Diagrama de bloques que muestra los componentes principales de un sistema VO.	8
1-6. Homografía entre dos imágenes en una misma superficie planar.	10
2-1. Coordenadas inerciales y coordenadas cuerpo.	20
2-2. Diferentes AHRS de la marca XSENS.	24
2-3. Sensor ultrasónico en un cuadrirrotor para medir su posición en el eje vertical.	25
4-1. Error del observador de velocidad \tilde{V}^b . \tilde{u} (línea continua), \tilde{v} (línea discontinua), \tilde{w} (línea punteada).	40
4-2. Error del estimador del factor de escala \tilde{K} . \tilde{k}_x (línea continua), \tilde{k}_y (línea discontinua), \tilde{k}_z (línea punteada).	41
4-3. Relación entre el acelerómetro y μ en el eje X^b . a_x^b (línea continua), $-\frac{\mu}{m}u$ (línea discontinua).	43
4-4. Entorno de Gazebo.	44
4-5. Velocidad observada en el eje X^b . u (línea continua), β_{1x} (línea discontinua).	45

4-6. Velocidad observada en el eje Y^b . v (línea continua), β_{1y} (línea discontinua).	46
4-7. Factor de escala estimado β_{2x} en el eje X^i	46
4-8. Factor de escala estimado β_{2y} en el eje Y^i	47
4-9. Posición del cuadrirrotor en el eje X^i . x (línea continua), $\frac{x_s}{k_x}$ (línea discontinua).	48
4-10. Posición del cuadrirrotor en el eje Y^i . y (línea continua), $\frac{y_s}{k_y}$ (línea discontinua).	48
4-11. Seguimiento de la trayectoria de lemniscata utilizando el estimador del factor de escala. Posición real (línea conti- nua), posición deseada (línea discontinua).	49
4-12. Entorno de Gazebo con diferentes condiciones iniciales. . .	50
4-13. Cuadrirrotor controlado por una computadora en placa reducida.	51
4-14. Sistema de comunicación del cuadrirrotor experimental. .	52
4-15. AR.Drone 2.0 de Parrot.	53
4-16. Posición del cuadrirrotor a lo largo de la trayectoria circular.	54
4-17. Velocidad traslacional real y estimada.	55
4-18. Factor de escala real y estimado.	55

Índice de tablas

Resumen

La navegación autónoma de vehículos aéreos no tripulados (UAV) requiere del acceso a las mediciones de velocidad traslacional y posición del mismo vehículo. Dado que no existe un único sensor para medir todos los estados del UAV, es necesario fusionar la información proveniente de diferentes tipos de sensores. Este trabajo de tesis propone un estimador determinístico para reconstruir el factor de escala de la posición determinada por un algoritmo de localización y mapeo simultáneo (SLAM) a bordo de un cuadricóptero. El factor de escala de la posición se desconoce cuando el algoritmo SLAM utiliza la información de una cámara monocular. Solo las mediciones de los sensores a bordo pueden alimentar el estimador; por lo tanto, se diseña además un observador determinístico para reconstruir la velocidad traslacional del cuadricóptero. El observador y estimador están diseñados siguiendo la metodología de inmersión e invarianza y utilizan mediciones inerciales y visuales. Argumentos de Lyapunov prueban la convergencia asintótica de los errores del observador y del estimador a cero. El desempeño del observador y estimador propuestos se validan a través de simulaciones numéricas usando un simulador basado en físicas, así como pruebas experimentales.

Abstract

Autonomous navigation of unmanned aerial vehicles (UAV) requires access to translational velocity and position measurements of the vehicle itself. Since there is no single sensor to measure all the states of the UAV, it is necessary to merge the information coming from different types of sensors. This thesis work proposes a deterministic estimator to reconstruct the position scale factor determined by a simultaneous localization and mapping (SLAM) algorithm on board a quadrotor. The position scale factor is unknown when the SLAM algorithm uses information from a monocular camera. Only measurements from on-board sensors can feed the estimator; therefore, a deterministic observer is also designed to reconstruct the translational velocity of the quadrotor. The observer and estimator are designed following the immersion and invariance methodology and use inertial and visual measurements. Lyapunov's arguments prove the asymptotic convergence of the observer and estimator errors to zero. The performance of the proposed observer and estimator is validated through numerical simulations using a physics-based simulator, as well as experimental tests.

Capítulo 1

Introducción

1.1. Antecedentes

Los vehículos aéreos no tripulados, o UAV por sus siglas en inglés (*unmanned aerial vehicle*), han pasado por un notable desarrollo tecnológico desde su concepción en la Primera Guerra Mundial, pensados como una herramienta de ataque a distancia que procuraba no poner en riesgo a los usuarios ya que éstos no se encontraban a bordo de la aeronave, hasta hoy en día en donde los podemos encontrar en aplicaciones tan variadas como entretenimiento, exploración, agricultura, logística, seguridad, etc. Este desarrollo tecnológico ha sido gracias principalmente a los múltiples esfuerzos de investigación que se han llevado a cabo a lo largo de décadas y han permitido expandir el área de conocimiento en este sector.

Dentro de toda la gama y configuraciones de vehículos aéreos no tripulados, destacan los **multirrotores**, que son aeronaves que consiguen la sustentación gracias a múltiples hélices de iguales características ubicadas en los extremos de brazos que sobresalen del centro de la aeronave y que generan el empuje por medio de motores, generalmente de corriente directa (DC) sin escobillas.

Los multirrotores son los vehículos aéreos no tripulados más utilizados actualmente debido a importantes ventajas que presentan sobre



Figura 1-1: Multirrotor de 6 hélices.

otro tipo de aeronaves. A diferencia de los aviones de ala fija y los helicópteros, los multirrotores cuentan con la mayor maniobrabilidad ya que se pueden mover y rotar en los tres ejes del espacio, su construcción es de bajo costo y a nivel de diseño de controladores de bajo nivel son de las aeronaves más fáciles de controlar. Estas ventajas los han posicionado como las aeronaves preferidas para aplicaciones como entrega de paquetes, agricultura inteligente, exploración, monitoreo, filmación de video para documentales y deportes extremos, control de incendios, rescate, etc.

Actualmente existen varias configuraciones de multirrotores dependiendo del número de rotores utilizados y la posición y sentido de giro de las hélices con respecto al frente de la aeronave. En la Figura 1-2 se muestran los multirrotores más comunes y cómo se debe construir su denominación dependiendo del número de rotores y sentido de giro.

Dentro de todas las variantes de multirrotores desarrolladas a la fecha, el cuadricóptero o **cuadrirrotor** es el más fácil de construir y controlar debido a que los 4 rotores con los que cuenta son la cantidad

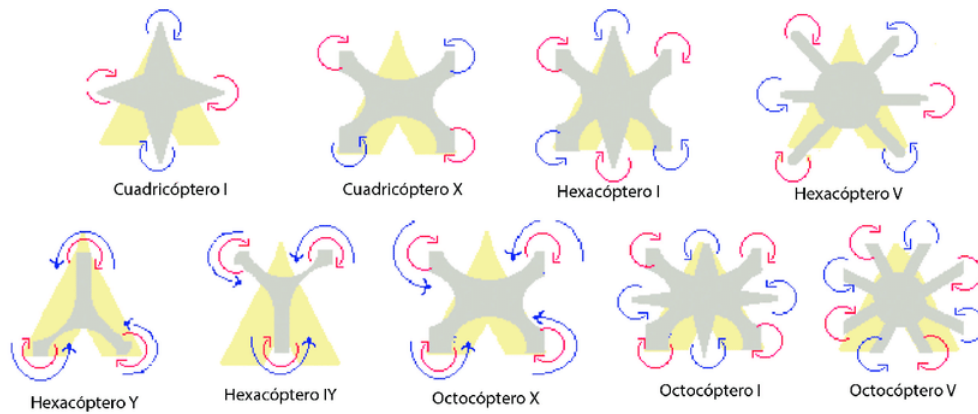


Figura 1-2: Multirrotores más comunes.

mínima requerida para que el empuje generado permita una estabilidad, maniobrabilidad y anulación del efecto contrarrotativo aplicando acciones de control equivalentes en los 4 motores. Existen multirrotores de 3 e incluso 2 rotores, pero la posición y sentido de giro de éstos no brindan estas características por lo que se necesita un esfuerzo de control extra para obtenerlas. Es por esta razón que los cuadrirrotores son los vehículos aéreos no tripulados más utilizados en las aplicaciones ya mencionadas.

Sin embargo, no todo está resuelto para los cuadrirrotores. Existen aplicaciones y áreas del conocimiento en donde se debe profundizar para plantear mejoras o soluciones a problemas presentes.

1.2. Estado del Arte

En este apartado se profundiza en las investigaciones y trabajos realizados hasta la fecha sobre cuadrirrotores en las principales áreas del conocimiento para tener un panorama completo de lo que ya se ha hecho, las oportunidades de mejora y los problemas que siguen presentes.

En el artículo (1), desarrollado por S. Gupte *et al.* en el año 2012,

se presenta un resumen de lo que es un cuadirrotor, sus principales características, los sensores y actuadores necesarios para su funcionamiento, su dinámica y métodos para controlarlo, ventajas y desventajas sobre otro tipo de aeronaves y sus principales aplicaciones. En este trabajo se presenta una clasificación de aeronaves, Figura 1-3, en donde se ubica a los cuadirrotores dentro de un grupo denominado VTOL, el cual es el acrónimo de *Vertical Take-Off and Landing*, que engloba todas aquellas aeronaves que despegan y aterrizan de forma vertical.

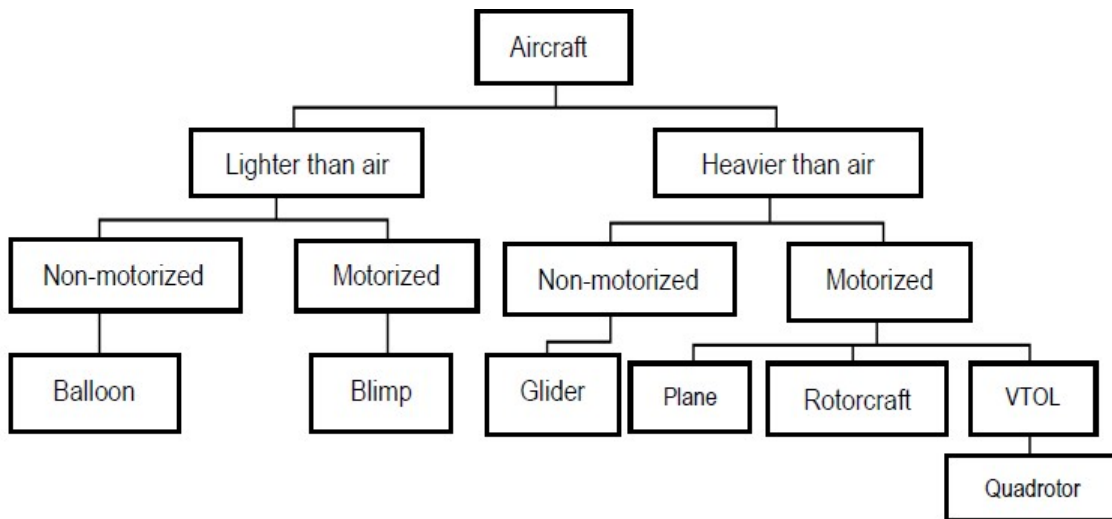


Figura 1-3: Clasificación de aeronaves según su principio de vuelo y modo de propulsión.

En este mismo artículo se hace énfasis de que en muchas de las aplicaciones de los cuadirrotores, como control de incendios, exploración o rescate, resulta impráctico e incluso peligroso que un operador humano esté dirigiendo manualmente con un control remoto al cuadirrotor, por lo que surge la importante necesidad de desarrollar algoritmos para **navegación autónoma** que le permitan al vehículo navegar por ambientes desconocidos sin la necesidad de un operador.

Para que un cuadirrotor pueda realizar navegación autónoma es

necesario que su controlador de bajo nivel tenga acceso a todos los estados del vehículo, sin embargo, actualmente no existen sensores listos para usar que puedan medir directamente todos estos estados sin tener que realizar un procesamiento previo o algún otro esfuerzo de estimación. Para poder medir la orientación del vehículo, por ejemplo, es necesario utilizar algoritmos de fusión de sensores utilizando las mediciones entregadas por una Unidad de Medición Inercial (IMU). En la referencia (2) del 2014 se presenta una comparación experimental de estos algoritmos para estimar la orientación de un cuerpo rígido a partir de una IMU. Este mismo problema de falta de sensor está presente al intentar medir la **velocidad traslacional** y la **posición** del vehículo.

Por parte de la velocidad traslacional, P. Martin *et al.* en su artículo (3) presentado en 2010, estudiaron en profundidad las fuerzas aerodinámicas presentes en la dinámica de los cuadricópteros, así como el modelo dinámico particular del acelerómetro al estar montado en estos vehículos. Tras estos estudios llegan a la conclusión de agregar un nuevo término a la dinámica del cuadricóptero que es debido a las fuerzas de arrastre ejercidas sobre las hélices del vehículo. Este nuevo término aporta información suficiente que permite observar la velocidad traslacional del cuadricóptero mediante el diseño de un observador determinístico. Con este trabajo se habilita una nueva herramienta para poder estimar este estado.

Para el año 2014, R. Beard *et al.* en (4) continúan esta línea de investigación y modelan al nuevo término propuesto por P. Martin en función de la velocidad traslacional y de un parámetro el cual denominaron como “coeficiente de arrastre” o μ , y del cual también demuestran su observabilidad. En este mismo trabajo se presenta un tutorial que explica el verdadero comportamiento de los acelerómetros en un cuadricóptero, donde se incluye un diagrama de la construcción interna de estos sensores, el cual se muestra en la Figura 1-4, con el cual se establece que los acelerómetros siempre estarán midiendo la aceleración debida a la fuerza de gravedad sin importar que el vehículo se

encuentre en reposo, por lo que esta componente se debe restar en el modelo dinámico del acelerómetro para obtener la aceleración total en el vehículo.

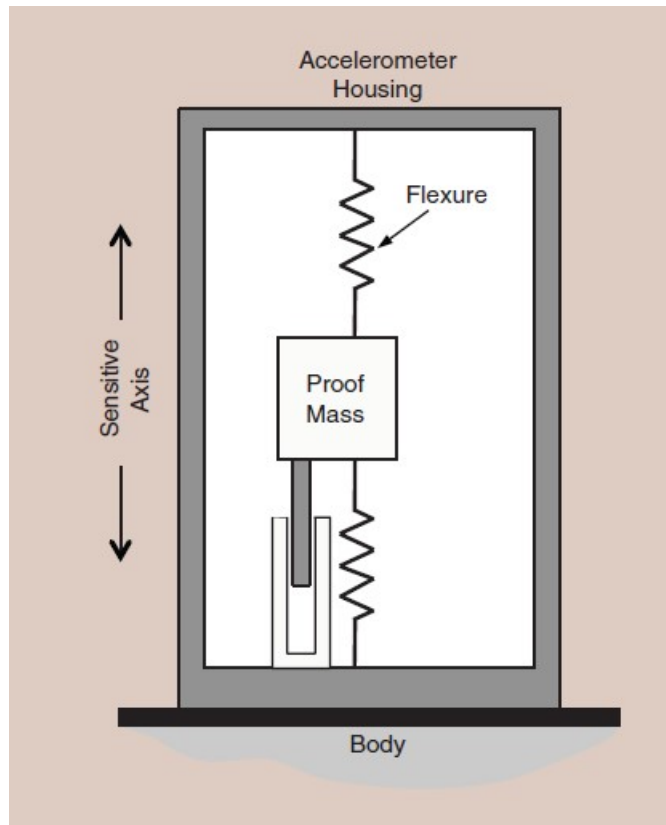


Figura 1-4: Diagrama simplificado de un acelerómetro.

Con lo que respecta a la medición de posición, el Sistema de Posicionamiento Global, o GPS por sus siglas en inglés (*Global Positioning System*), es el principal sensor utilizado para el posicionamiento de cuadrirrotores, sin embargo, este tipo de sensor presenta considerables limitaciones. No se puede usar en entornos donde la comunicación satelital está degradada, como en bosques o grandes cuerpos de agua como mares u océanos, o donde el sensor no tenga una vista directa al cielo, como en interiores de edificios o construcciones. Estos entornos donde el GPS no puede operar de forma correcta son lla-

mados “entornos denegados para GPS” (*GPS-denied environments*). El artículo (5) presentado en 2016 aborda más en profundidad este tipo de entornos. Además, los GPS de bajo costo no proporcionan una resolución suficiente para trayectorias a escala en centímetros, y el precio de un GPS con mayor resolución, como un GPS diferencial, aumenta drásticamente.

Para atacar el problema de baja resolución presente en los GPS, la referencia (6) de 2016 propuso un algoritmo de fusión de sensores basado en el filtro de Kalman para complementar las mediciones del GPS junto con un algoritmo de flujo óptico calculado a partir de la información entregada por un sensor visual. Poco después en 2018, el trabajo (7) presentó un algoritmo mejorado de fusión de sensores que agregaba además las mediciones de un Sistema de Navegación Inercial, o INS por sus siglas en inglés (*Inertial Navigation System*), basado en un filtro de Kalman extendido (EKF) y aplicado a la estimación de posición en un cuadirrotor. Ambos algoritmos de fusión de sensores mejoraron notablemente la medición de posición entregada por un GPS por sí solo, incluso para GPS de bajo costo, pero no dan solución a la medición de posición en entornos denegados para GPS.

La **visión por computadora**, o visión artificial, surge como una potente alternativa para la estimación de posición, así como complemento al fusionarla con otras técnicas de medición, como se vio en los trabajos anteriores. Los sensores visuales (cámaras) tienen muchas ventajas sobre otros sensores: son económicos, brindan información geométrica y de color para comprender la escena y consumen poca energía. Muchos algoritmos de visión por computadora están disponibles para la estimación de posición.

El algoritmo de odometría visual, o VO por sus siglas en inglés (*Visual Odometry*), presentado por D. Scaramuzza *et al.* en 2011, referencia (8), estima el movimiento de un agente (vehículo, humano, robot) utilizando únicamente la información de una cámara. VO calcula de forma incremental la posición del vehículo examinando los cambios que induce el movimiento en las imágenes de entrada. En la Figura

1-5 se muestra un diagrama a bloques de los principales componentes que forman el sistema VO. El trabajo presentado en (9) por M. L. Anderson *et al.* en 2019, mejoró el algoritmo VO al incluir una nueva técnica basada en covarianza que permite estimar la incertidumbre en la medición de la odometría dinámicamente en tiempo real, además de que esta técnica la implementaron en un cuadrirrotor logrando una navegación autónoma en interiores.

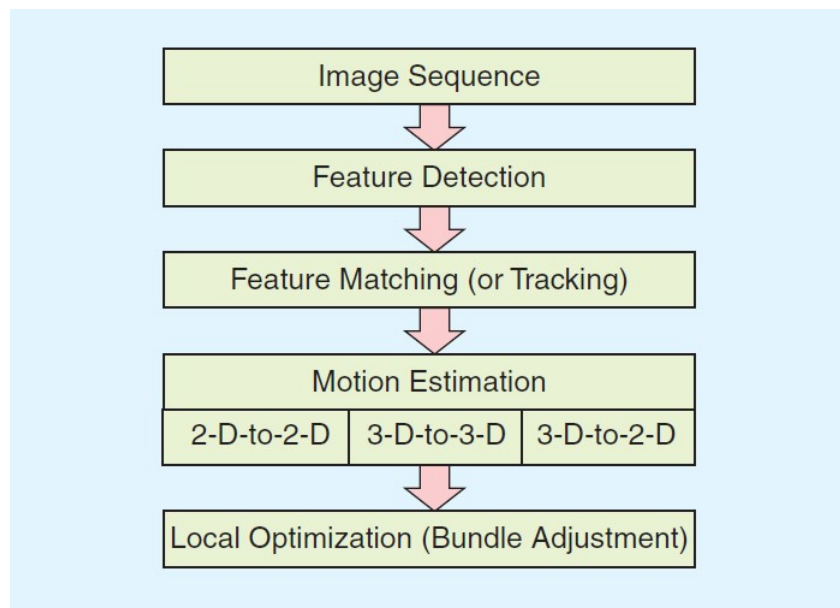


Figura 1-5: Diagrama de bloques que muestra los componentes principales de un sistema VO.

La localización y mapeo simultáneos, o SLAM por sus siglas en inglés (*Simultaneous Localization and Mapping*), proporciona una estimación de la posición del vehículo y, al mismo tiempo, construye un mapa del entorno que le rodea. Mayor referencia en (10). Existen muchas variantes del algoritmo SLAM dependiendo del método que utilicen para estimar la posición así como para construir el mapa, de entre los cuales destacan LSD-SLAM, presentado en 2014 por J. Engel *et al.* en (11), el cual utiliza un método directo (*direct method*), es decir, calcula el movimiento entre cuadros de imágenes utilizando toda la información de

la imagen, mientras que ORB-SLAM, presentado en 2015 por R. Mur-Artal *et al.* en (12), únicamente utiliza características detectadas en la imagen (*feature-based*).

El trabajo (13) de 2018 presenta una comparación de algoritmos SLAM aplicados a la navegación de robots móviles en interiores; así también, el artículo (14) de 2019 presenta la navegación autónoma en interiores de un cuadirrotor utilizando un algoritmo SLAM y sin el uso de un GPS. Con estos ejemplos se muestra que la visión por computadora ofrece una importante solución a la medición de posición, sobre todo en entornos denegados para GPS.

La odometría visual semidirecta, o SVO por sus siglas en inglés (*Semi-direct Visual Odometry*) es un algoritmo híbrido que combina métodos directos y basados en características, propuesto también por D. Scaramuzza y su equipo de investigación en 2014, (15). Esta mejora al VO convencional estima el movimiento relativo entre dos fotogramas minimizando los errores fotométricos. El error de proyección entre la ubicación de los puntos característicos y sus posiciones predichas se minimiza para obtener la posición óptima de la cámara. J. Wang *et al.* en 2017, (16), utilizan esta nueva herramienta y la aplican en la navegación de un cuadirrotor. La estrategia de navegación también incluye un esquema de estimación de posición para el control temporal del vehículo en caso de que el algoritmo SVO pierda la posición, así como también un método para corregir la escala de la escena, concepto que se abordará más adelante, sin la ayuda de otro sensor.

La homografía es la relación o correspondencia entre dos imágenes cualesquiera en una misma superficie planar. En la Figura 1-6 se muestra la relación entre las dos imágenes, la superficie planar y la matriz de homografía.

Esta matriz de correspondencia permite extraer la posición y orientación de la cámara en una imagen con respecto a la otra, con lo que se obtiene otra herramienta para estimar la posición utilizando sensores visuales. En 2017, A. Al-Kaff *et al.* en su artículo (17), pusieron en práctica esta herramienta y presentaron un sistema de navegación

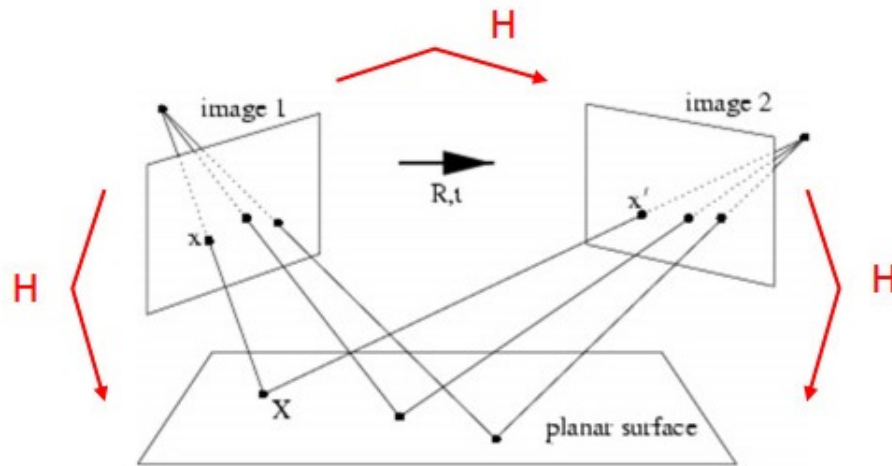


Figura 1-6: Homografía entre dos imágenes en una misma superficie planar.

para UAVs basado en homografía utilizando un cuadricóptero como plataforma experimental. El cálculo de la matriz de homografía, al ser un algoritmo basado en características, tiene múltiples métodos de ser abordado, destacando la correspondencia de cónicas (18), presentada por M. Hua *et al.* en 2017, o puntos y líneas (19), presentada por este mismo autor en 2020.

Como ejemplo final de algoritmos de visión por computadora utilizados para la estimación de la posición, la estructura a partir de movimiento, o SfM por sus siglas en inglés (*Structure from Motion*), es el proceso de reconstrucción de estructuras 3D a partir de sus proyecciones en una serie de imágenes tomadas desde diferentes puntos de vista. Como subproducto de este proceso, SfM también estima la posición de la cámara que tomó cada imagen. El artículo (20) presentado en 2016 aborda más a detalle los aspectos técnicos de este algoritmo. J. Tharp *et al.* presentaron en 2016 un algoritmo que fusiona SfM con una Unidad de Medición Inercial (IMU) para la navegación de aeronaves, (21).

Todos estos algoritmos de visión por computadora se pueden im-

plementar utilizando cámaras monoculares o estereoscópicas, con sus respectivas diferencias. Las cámaras estereoscópicas pueden capturar imágenes en 3D, lo que significa que se conoce la profundidad de las estructuras en la escena. Esta capacidad además, conduce a una mejor precisión y resolución para propósitos de estimación de posición, en comparación con las cámaras monoculares. Los algoritmos LSD-SLAM, ORB-SLAM y SVO, entre otros, fueron desarrollados primeramente utilizando cámaras monoculares, y más adelante, se mejoraron para configuraciones con varias cámaras (estereoscópicas), presentados en los artículos (22), (23) y (24), respectivamente.

Sin embargo, una **cámara monocular** se prefieren sobre las cámaras estereoscópicas para su implementación en vehículos pequeños por varias razones: una sola cámara es fácil de montar en el vehículo debido a su tamaño más pequeño, más ligero, es más barata y consume menos energía. Además, una configuración de una sola cámara está libre de la carga del proceso de calibración de varias cámaras en conjunto, y los algoritmos utilizados requieren menos potencia de procesamiento por parte de la CPU integrada en el vehículo. Una comparación de las diferencias entre estas dos implementaciones, aplicadas para un algoritmo FastSLAM, se publicó en el artículo (25) en 2016.

Las cámaras monoculares solo presentan un inconveniente: no pueden recuperar la estructura 3D y el movimiento de la cámara con información métrica completa; en otras palabras, la información de la profundidad de la escena no está disponible. Este fenómeno se conoce como **ambigüedad de similitud** (*similarity ambiguity*), o ambigüedad de escala (*scale ambiguity*). Mayores detalles técnicos de este fenómeno se pueden consultar en el libro (26).

La ambigüedad de similitud implica que los algoritmos de visión por computadora que utilizan cámaras monoculares no sean capaces de estimar la posición en escala real, es decir, la escala en la que realmente se encuentran los objetos, superficies y distancias en el mundo real, lo que también se conoce como “escala absoluta”, debido a la información métrica faltante que las cámaras monoculares no proporcionan.

Por esta razón, los algoritmos que utilizan estas cámaras trabajan en una escala diferente, llamada “escala relativa”, que es proporcional a la escala absoluta, es decir, la escala absoluta es igual a la escala relativa multiplicada por un **factor de escala**. Conociendo este factor de escala y la escala relativa que entrega el algoritmo, sería posible recuperar la posición en escala real.

Sin embargo, este factor de escala no está disponible ya que el algoritmo lo define en tiempo de ejecución y depende de muchos factores como las características físicas de la cámara (resolución, distancia focal, ángulo de visión), objetos en la escena, condiciones de luz, características a detectar por el algoritmo, condiciones iniciales, etc., por lo que este factor es diferente prácticamente para cualquier experimento, por lo que se podría definir como un parámetro casi aleatorio.

Se han propuesto varios métodos para tratar el problema de ambigüedad de similitud, ya sea estimando la escala absoluta directamente o por medio del factor de escala. Se requiere al menos una pieza de información métrica para recuperar la escala absoluta; esta señal puede provenir del conocimiento previo de la escena, como la altura de la cámara, el tamaño de un objeto, la velocidad del vehículo, u otros sensores que aporten información métrica como un LiDAR o GPS.

En los trabajos (27) de 2016 y (28) de 2018, la información métrica adicional para determinar la escala en la medición de posición la proporciona un sensor ultrasónico y un telémetro láser 1D (LRF), respectivamente. El problema de esta solución es que se requieren sensores adicionales a bordo del vehículo, y además, la escala absoluta solo se calcula en el eje donde está midiendo el sensor adicional, por lo que suponen que la escala es la misma en los dos ejes restantes, lo cual no es del todo cierto.

Otro método para resolver el problema de ambigüedad de similitud se basa en conocer la altura de la cámara a un plano de tierra (suelo). D. Zhou *et al.*, en el trabajo presentado en (29) de 2016, usaron este método para recuperar la escala absoluta para un algoritmo VO monocular, con una técnica de “divide y vencerás”, es decir, descompo-

niendo los parámetros de movimiento en la homografía a partir de los parámetros de estructura del plano de tierra. Años después en 2019, este enfoque se mejoró por los mismos autores en (30), aumentando la solidez del algoritmo. El problema de este método es que se necesita una cámara (o sensor) en la parte inferior del vehículo y una vista directa desde ella al suelo, lo que no siempre se puede lograr en todos los entornos ni para cualquier tipo de vehículo.

M. Ludhiyani *et al.* en su artículo (31) de 2019, propusieron un estimador del factor de escala para algoritmos SLAM monocular basado en el modelo dinámico del cuádrirrotor, y presentaron un análisis de observabilidad que muestra que la escala se vuelve observable cuando el modelo dinámico del cuádrirrotor y la visión monocular se usan en conjunto. Sin embargo, el uso de EKF en este enfoque hace que el estimador no sea determinístico, por lo que no es posible asegurar estabilidad en la técnica de control.

En el campo de los estimadores determinísticos, el trabajo (32) de 2018 presentó un estimador de escala para algoritmos SLAM monocular basado en la estabilidad del controlador en un cuádrirrotor. Muestra que la escala absoluta y la ganancia del controlador tienen una relación lineal única, por lo que detectando oscilaciones autoinducidas y analizando la estabilidad del sistema, se puede estimar la escala absoluta. El problema de este enfoque es que, para una estimación de la escala en línea se debe utilizar un controlador adaptativo, dejando fuera cualquier otra técnica de control, además de que los resultados experimentales obtenidos mostraron ser menos precisos que los trabajos que implementan un sensor adicional que proporciona la información métrica necesaria para calcular la escala.

L. Nieto *et al.* en su artículo (33) de 2019, propusieron un estimador para el cálculo del factor de escala en algoritmos SLAM monocular, determinístico, basado en el modelo dinámico del cuádrirrotor y los sensores, siguiendo la metodología de inmersión e invarianza (I&I). El artículo presenta un análisis de estabilidad, simulaciones numéricas y pruebas experimentales en tiempo real que validan el funcionamiento y

desempeño de la propuesta. Este estimador necesita como información de entrada las mediciones entregadas por la IMU a bordo del cuadrirrotor, la información de posición escalada entregada por el algoritmo SLAM monocular, y la velocidad traslacional del vehículo, la cual en este caso es proporcionada por un sistema de captura de movimiento. Este es el principal inconveniente del estimador propuesto ya que su uso está limitado únicamente al área que el sistema de captura de movimiento pueda cubrir, descartando completamente aplicaciones en exteriores, entre otras.

Estos trabajos dejan en evidencia que resolviendo el problema de ambigüedad de escala, se obtendrían todos los beneficios que ofrece una cámara monocular, junto con la posibilidad de realizar estimaciones métricas en escala absoluta, por lo que utilizando cualquier algoritmo de visión monocular para estimar la posición, como los mencionados anteriormente, se podría obtener de ellos directamente la posición real.

1.3. Justificación

Analizando y resumiendo las investigaciones y trabajos realizados hasta la fecha en torno a los cuadrirrotores, presentados en el Estado del Arte, se destacan las siguientes observaciones:

- Para que un cuadrirrotor pueda navegar de forma autónoma, en un ambiente desconocido y sin la necesidad de un operador humano o una unidad de control remota, el cuadrirrotor debe ser capaz de medir todos sus estados utilizando únicamente sus sensores y procesador a bordo.
- Actualmente, no existen sensores listos para usar que puedan medir o estimar la velocidad traslacional y posición de vehículos aéreos.

- El coeficiente de arrastre μ permite observar la velocidad traslacional de un cuádrirrotor mediante un método determinístico y sin la necesidad de sensores extra.
- La visión por computadora se ha establecido como una potente alternativa para estimar la posición de vehículos aéreos en ambientes desconocidos y con una amplia variedad de algoritmos disponibles para ello.
- Es indispensable el uso de un sensor visual, o cámara, para la ejecución de un algoritmo de visión por computadora. Las cámaras monoculares presentan considerables ventajas sobre las cámaras estereoscópicas en la mayoría de aplicaciones, salvo por un inconveniente: ambigüedad de similitud.
- Conocer el factor de escala que utilizan los algoritmos de estimación de posición con visión monocular permitiría medir en escala real la posición del vehículo.
- Fusionar algoritmos de estimación basados en visión por computadora con mediciones provenientes de sensores inerciales enriquece notablemente la estimación final del algoritmo, mejorando precisión, exactitud y tiempo de respuesta.
- Un método de estimación determinístico permite demostrar, matemáticamente, convergencia en la estimación así como estabilidad en la técnica de control, lo cual no es posible con métodos probabilísticos.

1.4. Objetivos

1.4.1. Objetivo General

- Diseñar un observador de la velocidad traslacional, así como un estimador del factor de escala para algoritmos de estimación de

posición con visión monocular; ambos determinísticos y basados en los modelos dinámicos del cuadrirrotor y los sensores a bordo disponibles, fusionados en un único estimador en cascada.

1.4.2. Objetivos Particulares

- Diseñar un observador de velocidad traslacional, determinístico, basado en modelo, para un cuadrirrotor, utilizando el coeficiente de arrastre μ y demostrar su convergencia.
- Diseñar un estimador del factor de escala, determinístico, basado en modelo, para algoritmos de estimación de posición con visión monocular y demostrar su convergencia.
- Fusionar el observador de velocidad traslacional y el estimador del factor de escala diseñados, en un único estimador en cascada y demostrar su convergencia.
- Validar mediante simulaciones numéricas la posibilidad de navegación autónoma, realizando un seguimiento de trayectoria en lazo cerrado utilizando únicamente los sensores a bordo del cuadrirrotor así como las mediciones de velocidad y posición provenientes del estimador en cascada diseñado.

1.5. Organización de la Tesis

El resto de este trabajo de tesis está organizado de la siguiente forma. En el capítulo 2 se presenta la dinámica del cuadrirrotor, los modelos de los sensores utilizados, y la metodología de inmersión e invarianza, con lo que se definen todas las herramientas matemáticas a utilizar. El capítulo 3 describe los desarrollos matemáticos para diseñar el observador de velocidad así como el estimador de factor de escala, y resume la principal contribución de este trabajo. En el capítulo 4 se muestra el desempeño del estimador en cascada a través

de simulaciones numéricas en diferentes plataformas y condiciones de simulación. Finalmente, el capítulo 5 finaliza este documento con algunos comentarios finales y trabajos a futuro.

Capítulo 2

Materiales y Métodos

2.1. Dinámica del Cuadrirrotor

R. Mahony *et al.* describen detalladamente la dinámica de un cuadrirrotor, así como las principales técnicas de control, en su artículo (34). De esta referencia se toman la mayoría de ecuaciones para establecer los fundamentos matemáticos de este trabajo, junto con el término en la dinámica traslacional debido a las fuerzas de arrastre introducido en (4), utilizando la notación particular del autor de esta tesis.

Se necesita un marco de coordenadas inerciales y un marco de coordenadas no inerciales, o coordenadas cuerpo, unido al centro de gravedad del cuadrirrotor como se muestra en la Figura 2-1, para describir su dinámica.

Las siguientes ecuaciones, expresadas en coordenadas mixtas, inerciales y cuerpo, describen la dinámica de traslación y rotación del cuadrirrotor:

$$\begin{aligned}\dot{X} &= RV^b \\ m\dot{V}^b &= mgR^\top e_3 - T_T e_3 - \mu HV^b - mS(\Omega)V^b \\ \dot{R} &= RS(\Omega) \\ J\dot{\Omega} &= -S(\Omega)J\Omega + M^b\end{aligned}\tag{2.1}$$

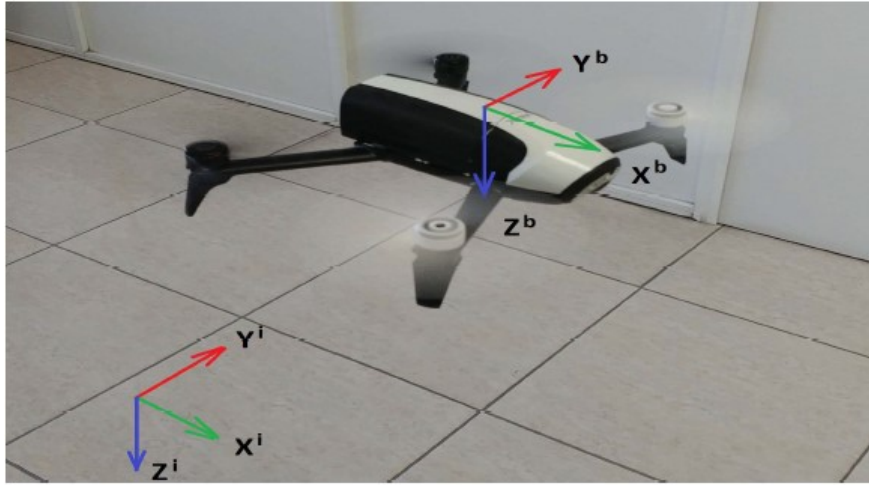


Figura 2-1: Coordenadas inerciales y coordenadas cuerpo.

$X = \begin{bmatrix} x & y & z \end{bmatrix}^\top$ es la posición del cuadrirrotor expresada en ejes inerciales.

$R \in SO(3)$ es la matriz de rotación que permite trasladar estados o parámetros de coordenadas cuerpo a coordenadas inerciales. Esta matriz pertenece al grupo especial ortogonal de 3 dimensiones el cual está definido como

$$SO(3) = \{R \in \mathbb{R}^{3 \times 3} \mid R^\top R = I, \det(R) = 1\}$$

siendo I la matriz identidad.

En función de los ángulos de Euler en la secuencia de rotación ZYX , $\Phi = \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}^\top$, la matriz de rotación está definida como

$$R = \begin{pmatrix} c_\theta c_\psi & c_\psi s_\theta s_\phi - s_\psi c_\phi & c_\psi s_\theta c_\phi + s_\psi s_\phi \\ c_\theta s_\psi & s_\psi s_\theta s_\phi + c_\psi c_\phi & s_\psi s_\theta c_\phi - c_\psi s_\phi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{pmatrix} \quad (2.2)$$

siendo las abreviaciones $s_\sigma = \sin(\sigma)$ y $c_\sigma = \cos(\sigma)$ para cualquier ángulo σ .

$V^b = \begin{bmatrix} u & v & w \end{bmatrix}^\top$ es la velocidad traslacional del cuadrirrotor expresada en coordenadas cuerpo.

m es la masa del cuadrirrotor, g es la aceleración debido a la fuerza de gravedad y $e_3 = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^\top$ es un vector unitario utilizado para eliminar las componentes en el plano horizontal de una magnitud.

T_T es el empuje total generado por los rotores del vehículo, cuya forma general es

$$T_T = \sum_{i=1}^N T_i$$

donde T_i el empuje de cada rotor y N el número total de rotores, siendo $N = 4$ para el caso del cuadrirrotor.

$$H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

es una matriz utilizada para eliminar la componente del eje vertical del término debido a las fuerzas de arrastre, y μ es el coeficiente de arrastre presentado en (4).

$\Omega = \begin{bmatrix} p & q & r \end{bmatrix}^\top$ es la velocidad angular expresada en coordenadas cuerpo, con $S(\Omega)$ una matriz antisimétrica definida como

$$S(\Omega) = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix}$$

J es la matriz de inercias del cuadrirrotor y M^b el vector de momentos generados por el empuje diferencial y el momento de reacción entre los cuatro rotores.

2.2. Sensores Disponibles

Se supone que el cuadrirrotor lleva a bordo un conjunto de sensores que proporcionan las mediciones necesarias para su control. A continuación, se especifican todas estas mediciones que están disponibles para el diseño del estimador, y que serán denotadas como y_i .

2.2.1. Posición Escalada

El vehículo lleva a bordo una cámara monocular orientada hacia el plano horizontal y la potencia computacional necesaria (CPU) para implementar un algoritmo de visión monocular para determinar su posición inercial. Por lo tanto, la siguiente medición está disponible:

$$y_1 = X_s = \begin{bmatrix} \hat{k}_x x \\ \hat{k}_y y \\ \hat{k}_z z \end{bmatrix} = \begin{bmatrix} \hat{k}_x & 0 & 0 \\ 0 & \hat{k}_y & 0 \\ 0 & 0 & \hat{k}_z \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \text{diag}(K)X \quad (2.3)$$

X_s es la posición escalada entregada por el algoritmo de visión monocular debido al fenómeno de ambigüedad de similitud, y $K = \begin{bmatrix} \hat{k}_x & \hat{k}_y & \hat{k}_z \end{bmatrix}^\top$ es el factor de escala desconocido en los ejes $X^i Y^i Z^i$, respectivamente. El operador $\text{diag}(A)$ es una matriz diagonal cuyos elementos en su diagonal principal son los elementos del vector $A \in \mathbb{R}^3$. Este operador satisface las siguientes identidades:

$$\text{diag}(A + B) = \text{diag}(A) + \text{diag}(B) \quad (2.4)$$

$$\text{diag}(A)B = \text{diag}(B)A \quad (2.5)$$

con $A, B \in \mathbb{R}^3$ vectores.

2.2.2. Aceleración Específica

Comúnmente, los cuadrirrotores llevan una Unidad de Medición Inercial (IMU) que proporciona, en coordenadas cuerpo, la aceleración de traslación, la velocidad angular y la intensidad del campo magnético terrestre. De acuerdo con el “tutorial de acelerómetro” reportado en (4), la aceleración específica medida por un acelerómetro montado en un cuadrirrotor viene dada por

$$a^b = \frac{1}{m} F_T^b - gR^\top e_3 \quad (2.6)$$

donde $a^b = \begin{bmatrix} a_x^b & a_y^b & a_z^b \end{bmatrix}^\top$ con a_x^b , a_y^b y a_z^b las aceleraciones específicas a lo largo de los ejes $X^b Y^b Z^b$, y F_T^b son las fuerzas externas totales que actúan sobre el cuadirrotor. Del modelo dinámico del cuadirrotor en la ecuación (2.1) se tiene que

$$F_T^b = mgR^\top e_3 - T_T e_3 - \mu H V^b \quad (2.7)$$

Es importante notar que en esta ecuación (2.7) no está incluido el término $-mS(\Omega)V^b$, ya que la fuerza de Coriolis no es una fuerza real, en el sentido de que no hay nada que la produzca. Es una fuerza ficticia que se introduce para explicar, desde el punto de vista del sistema en rotación, la aceleración del cuerpo, cuyo origen está en realidad, en el hecho de que el sistema de observación está rotando.

Sustituyendo (2.7) en (2.6) se obtiene el modelo dinámico de un acelerómetro montado en un cuadirrotor:

$$a^b = -\frac{T_T}{m} e_3 - \frac{\mu}{m} H V^b \quad (2.8)$$

Por lo tanto, la aceleración específica es una salida disponible, leída elemento a elemento como:

$$y_2 = a^b = \begin{bmatrix} a_x^b \\ a_y^b \\ a_z^b \end{bmatrix} = -\frac{1}{m} \begin{bmatrix} \mu u \\ \mu v \\ T_T \end{bmatrix} \quad (2.9)$$

2.2.3. Sistemas de Referencia de Orientación y Rumbo

El dispositivo que calcula la orientación del cuadirrotor mediante un algoritmo de fusión de sensores a partir de las mediciones entregadas por la IMU se llama Sistema de Referencia de Orientación y Rumbo, o AHRS por sus siglas en inglés. Ejemplos de estos dispositivos se muestran en la Figura 2-2.

Suponiendo que el cuadirrotor lleva un AHRS a bordo, las siguiente señal está disponible:



Figura 2-2: Diferentes AHRS de la marca XSENS.

$$y_3 = R = \begin{bmatrix} r_1^T \\ r_2^T \\ r_3^T \end{bmatrix} \quad (2.10)$$

donde $r_i, i = 1, \dots, 3$ son las columnas de la matriz de rotación transpuesta, desglosadas en (2.2).

Además, un AHRS realiza un filtrado a las mediciones del giroscopio de la IMU para eliminar el ruido debido a los rotores de vehículo, proporcionando una medición más limpia y libre de este factor, por lo que la velocidad angular está disponible:

$$y_4 = \Omega = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (2.11)$$

2.2.4. Rapidez Vertical

Uno de los aspectos en los que se busca innovar en este trabajo de tesis, es en siempre considerar la fuerza de Coriolis a lo largo de los cálculos, la cual en trabajos ya presentados siempre es considerada lo suficientemente pequeña como para eliminarla.

Mantener este término implica la necesidad de poder medir la rapidez vertical w , lo cual se puede realizar a partir de la medición de la posición vertical del cuadrirrotor mediante un sensor láser, un sensor

ultrasónico, como se muestra en la Figura 2-3, o un altímetro, normalmente incluido en un AHRS.

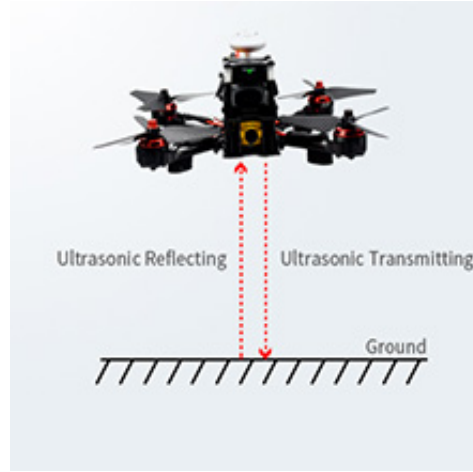


Figura 2-3: Sensor ultrasónico en un cuadrirrotor para medir su posición en el eje vertical.

Por lo tanto, se supone que la siguiente medición está disponible:

$$y_5 = w \quad (2.12)$$

Finalmente, tener en cuenta que la dinámica de traslación del cuadrirrotor, la segunda ecuación en (2.1), expresada en términos de los estados medidos ya definidos y dividiendo todo por la masa m , se puede expresar como:

$$\dot{V}^b = gy_3^\top e_3 + y_2 - S(y_4)V^b \quad (2.13)$$

2.3. Metodología de Inmersión e Invarianza

El método de Inmersión e Invarianza (I&I), presentado por A. Astolfi y R. Ortega en el artículo (35) en 2003, incorpora la idea de inmersión de estados, con la idea de variedades invariantes, cuya dimensión está relacionada con la dimensión de los sistemas principal y objetivo.

Considere el siguiente sistema no lineal, determinístico e invariante en tiempo

$$\begin{aligned}\dot{\eta} &= f_1(\eta, y) \\ \dot{y} &= f_2(\eta, y)\end{aligned}\tag{2.14}$$

donde $\eta \in \mathcal{R} \subset \mathbb{R}^n$ y $y \in \mathcal{Y} \subset \mathbb{R}^m$ son los estados no medibles y medibles, respectivamente. Se supone que los campos vectoriales $f_1(\eta, y)$ y $f_2(\eta, y)$ son completos hacia adelante (*forward complete*).

Definición 1 *El sistema dinámico*

$$\dot{\hat{\eta}} = \varphi(\hat{\eta}, y)\tag{2.15}$$

con $\hat{\eta} \in \mathbb{R}^p$, $p \geq n$ es un observador del estado no medible η si existe un mapeo $\beta : \mathbb{R}^p \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ tal que la variedad

$$\mathcal{M} = \{(\eta, \hat{\eta}, y) \in \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^m \mid \beta(\hat{\eta}, y) = \eta\}$$

tenga las siguientes propiedades:

- \mathcal{M} es positivamente invariante.
- Todas las trayectorias de (2.14), (2.15) que empiecen en una vecindad de \mathcal{M} convergen asintóticamente a \mathcal{M} .

La construcción del observador de la forma dada en la Definición 1 requiere propiedades adicionales en el mapeo $\beta(\hat{\eta}, y)$ como se establece en el siguiente resultado.

Teorema 1 *Considere el sistema (2.14). Suponga que existe un mapeo diferenciable $\beta : \mathbb{R}^p \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ tal que*

A1 Para toda $\hat{\eta}$ y y el mapeo $\beta(\hat{\eta}, y)$ satisface

$$\det\left(\frac{\partial\beta}{\partial\hat{\eta}}\right) \neq 0$$

A2 El sistema dinámico

$$\dot{\tilde{\eta}} = f_1(\tilde{\eta} + \beta(\hat{\eta}, y), y) - f_1(\beta(\hat{\eta}, y), y) - \frac{\partial \beta}{\partial y} (f_2(\tilde{\eta} + \beta(\hat{\eta}, y), y) - f_2(\beta(\hat{\eta}, y), y)) \quad (2.16)$$

tiene un equilibrio asintóticamente (globalmente) estable en $\tilde{\eta} = 0$ uniformemente en $\hat{\eta}$ y y .

Entonces el sistema (2.15) con

$$\varphi(\hat{\eta}, y) = \left(\frac{\partial \beta}{\partial \hat{\eta}} \right)^{-1} \left(f_1(\beta(\hat{\eta}, y), y) - \frac{\partial \beta}{\partial y} f_2(\beta(\hat{\eta}, y), y) \right) \quad (2.17)$$

es un observador (global) para (2.14).

Demostración del Teorema 1 La distancia de las trayectorias de los sistemas (2.14) y (2.15) a la variedad \mathcal{M} está caracterizada por

$$\tilde{\eta} = \eta - \beta(\hat{\eta}, y)$$

La dinámica de $\tilde{\eta}$ está dada por

$$\dot{\tilde{\eta}} = f_1(\eta, y) - \frac{\partial \beta}{\partial \hat{\eta}} \varphi(\hat{\eta}, y) - \frac{\partial \beta}{\partial y} f_2(\eta, y) \quad (2.18)$$

entonces, sustituyendo $\varphi(\hat{\eta}, y)$ de la ecuación (2.17) en (2.18) se obtiene que la dinámica de $\tilde{\eta}$ se vuelve igual a (2.16). Finalmente, la suposición A2 implica que (globalmente)

$$\lim_{t \rightarrow \infty} \beta(\hat{\eta}, y) = \eta \quad (2.19)$$

El resultado expresado en el Teorema 1 es el que se sigue para diseñar el observador de velocidad y el estimador del factor de escala.

Capítulo 3

Diseño del Observador y Estimador

Esta sección aborda el diseño del observador para reconstruir la velocidad traslacional V^b y el estimador para el factor de escala K a partir de las mediciones de posición escalada, aceleración, orientación, velocidad angular y velocidad vertical, descritas en el capítulo anterior.

3.1. Problema de Observación

Los siguientes términos establecen el problema de observación. Suponga que las salidas y_i , $i = 1, \dots, 5$ son medibles. Diseñar dos sistemas dinámicos de la forma

$$\begin{aligned}\dot{\hat{V}}^b &= \varphi_1(\hat{V}^b, y_2, y_3, y_4, y_5) \\ \dot{\hat{K}} &= \varphi_2(\hat{K}, \hat{V}^b, y_1, y_2, y_3, y_4, y_5)\end{aligned}$$

donde $\hat{V}^b \in \mathbb{R}^3$ y $\hat{K} \in \mathbb{R}^3$, tal que existen dos funciones $\beta_1 \in \mathbb{R}^3$ y $\beta_2 \in \mathbb{R}^3$ que dependen de la información disponible, y las siguientes identidades se mantienen asintóticamente

$$\begin{aligned}\lim_{t \rightarrow \infty} \beta_1(\hat{V}^b, y_2, y_3, y_4, y_5) &= V^b \\ \lim_{t \rightarrow \infty} \beta_2(\hat{K}, \hat{V}^b, y_1, y_2, y_3, y_4, y_5) &= K\end{aligned}\tag{3.1}$$

3.2. Observador de Velocidad

De acuerdo con la técnica de Inmersión e Invarianza, el error de observación se define de la siguiente manera:

$$\tilde{V}^b = V^b - \beta_1(\hat{V}^b, \sigma) \quad (3.2)$$

con β_1 leído elemento por elemento como

$$\beta_1 = \begin{bmatrix} \beta_{1x} \\ \beta_{1y} \\ \beta_{1z} \end{bmatrix} \quad (3.3)$$

donde

$$\dot{\sigma} = \begin{bmatrix} -\frac{\mu}{m}u \\ \frac{\mu}{m}v \\ -w \end{bmatrix} = -\bar{H}V^b \quad (3.4)$$

y

$$\bar{H} = \begin{bmatrix} \frac{\mu}{m} & 0 & 0 \\ 0 & \frac{\mu}{m} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

La ecuación (3.2) modela la distancia a la variedad \mathcal{M} de la Definición 1, donde la velocidad V^b es igual a $\beta_1(\hat{V}^b, \sigma)$. Esta distancia debe converger asintóticamente a cero para completar el diseño del observador. Tenga en cuenta que la salida y_2 no se usa directamente ya que la derivada temporal de \tilde{V}^b requerirá conocer \dot{y}_2 , la cual no está disponible; esta es la razón por la que se introduce el nuevo estado σ .

La derivada con respecto al tiempo de \tilde{V}^b está dada por

$$\dot{\tilde{V}}^b = \dot{V}^b - \frac{\partial \beta_1}{\partial \hat{V}^b} \dot{\hat{V}}^b - \frac{\partial \beta_1}{\partial \sigma} \dot{\sigma} \quad (3.5)$$

Sustituyendo $\dot{\hat{V}}^b$, la velocidad traslacional en coordenadas cuerpo V^b y la derivada de σ de las ecuaciones (2.13), (3.2) y (3.4), respectivamente, se tiene

$$\dot{\hat{V}}^b = gy_3^\top e_3 + y_2 - S(y_4)(\tilde{V}^b + \beta_1) - \frac{\partial \beta_1}{\partial \hat{V}^b} \dot{\hat{V}}^b + \frac{\partial \beta_1}{\partial \sigma} \bar{H}(\tilde{V}^b + \beta_1) \quad (3.6)$$

Ahora, la dinámica del observador de velocidad $\dot{\hat{V}}^b$ se define en función de las señales conocidas, como

$$\dot{\hat{V}}^b = \frac{\partial \beta_1}{\partial \hat{V}^b}^{-1} \left(gy_3^\top e_3 + y_2 - S(y_4)\beta_1 + \frac{\partial \beta_1}{\partial \sigma} \bar{H}\beta_1 \right) \quad (3.7)$$

Sustituyendo (3.7) en (3.6), se tiene

$$\dot{\hat{V}}^b = -S(y_4)\tilde{V}^b + \frac{\partial \beta_1}{\partial \sigma} \bar{H}\tilde{V}^b \quad (3.8)$$

Considere la siguiente función de Lyapunov para definir la función $\beta_1(\hat{V}^b, \sigma)$

$$\nu_V = \frac{1}{2} \tilde{V}^{b\top} \tilde{V}^b \quad (3.9)$$

cuya derivada es

$$\dot{\nu}_V = \tilde{V}^{b\top} \frac{\partial \beta_1}{\partial \sigma} \bar{H} \tilde{V}^b \quad (3.10)$$

Por tanto, para garantizar que la derivada con respecto al tiempo de ν_V es definida negativa, la matriz $\frac{\partial \beta_1}{\partial \sigma} \bar{H}$ también debe ser definida negativa. Por otro lado, la ecuación (3.7) requiere que la matriz $\frac{\partial \beta_1}{\partial \hat{V}^b}$ sea invertible. Note que seleccionando

$$\beta_1(\hat{V}^b, \sigma) = \hat{V}^b - \Gamma_v \sigma \quad (3.11)$$

con

$$\Gamma_v = \begin{bmatrix} \gamma_{vx} & 0 & 0 \\ 0 & \gamma_{vy} & 0 \\ 0 & 0 & \gamma_{vz} \end{bmatrix} \quad (3.12)$$

siendo γ_{vx} , γ_{vy} y γ_{vz} ganancias positivas, se tiene que

$$\dot{\nu}_V = -\tilde{V}^{b\top} \Gamma_v \bar{H} \tilde{V}^b = -\tilde{V}^{b\top} \text{diag}(h) \tilde{V}^b, \quad \frac{\partial \beta_1}{\partial \hat{V}^b} = I_3 \quad (3.13)$$

con $h = \left[\frac{\gamma_{vx}\mu}{m} \quad \frac{\gamma_{vy}\mu}{m} \quad \gamma_{vz} \right]^T$ y I_3 la matriz identidad de 3×3 . Como resultado, $\frac{\partial \beta_1}{\partial \sigma} \bar{H}$ es definida negativa y $\frac{\partial \beta_1}{\partial \hat{V}^b}$ invertible.

3.3. Estimador del Factor de Escala

En la referencia (33), el estimador del factor de escala necesita la velocidad traslacional como salida medible, pero en este trabajo está disponible a través del observador diseñado (3.11) según (3.1). Es importante notar también en esta ecuación que β_2 depende de estados expresados en coordenadas inerciales y cuerpo, a diferencia de β_1 que depende solo de estados expresados en coordenadas cuerpo. Para tener todos los estados expresados en coordenadas inerciales, β_1 debe trasladarse con la matriz de rotación. Por tanto, la velocidad inercial se introduce de la siguiente manera:

$$V^i = y_3 V^b \quad (3.14)$$

Además, el error del observador de velocidad inercial se define como

$$\tilde{V}^i = V^i - \hat{V}^i \quad (3.15)$$

con

$$\hat{V}^i = y_3 \beta_1 \quad (3.16)$$

Ahora, el error de estimación del factor de escala se define de la siguiente forma

$$\tilde{K} = K - \beta_2(\hat{K}, y_1, \hat{V}^i) \quad (3.17)$$

con β_2 leída elemento a elemento como

$$\beta_2 = \begin{bmatrix} \beta_{2x} \\ \beta_{2y} \\ \beta_{2z} \end{bmatrix} \quad (3.18)$$

La derivada con respecto al tiempo del error de estimación es

$$\dot{\hat{K}} = -\frac{\partial\beta_2}{\partial\hat{K}}\dot{\hat{K}} - \frac{\partial\beta_2}{\partial y_1}\dot{y}_1 - \frac{\partial\beta_2}{\partial\hat{V}^i}\dot{\hat{V}}^i \quad (3.19)$$

De las ecuaciones (2.1) y (2.3), se obtiene que

$$\begin{aligned} \dot{y}_1 &= \text{diag}(K)y_3V^b = \text{diag}(K)V^i \\ \dot{y}_3 &= y_3S(y_4) \end{aligned} \quad (3.20)$$

Al combinar las ecuaciones (3.2) y (3.16), se obtiene

$$\hat{V}^i = y_3(V^b - \tilde{V}^b) \quad (3.21)$$

entonces,

$$\dot{\hat{V}}^i = ge_3 + y_3y_2 - \Gamma_v\bar{H}\tilde{V}^i \quad (3.22)$$

Substituyendo (3.20) y (3.21) en (3.19), se obtiene

$$\dot{\hat{K}} = -\frac{\partial\beta_2}{\partial\hat{K}}\dot{\hat{K}} - \frac{\partial\beta_2}{\partial y_1}\text{diag}(K)V^i - \frac{\partial\beta_2}{\partial\hat{V}^i}\left(ge_3 + y_3y_2 - \Gamma_v\bar{H}\tilde{V}^i\right) \quad (3.23)$$

Ahora, substituyendo K de (3.17) y V^i de (3.15), el error de estimación del factor de escala se convierte en

$$\dot{\hat{K}} = -\frac{\partial\beta_2}{\partial\hat{K}}\dot{\hat{K}} - \frac{\partial\beta_2}{\partial y_1}\text{diag}(\tilde{K} + \beta_2)\left(\tilde{V}^i + \hat{V}^i\right) - \frac{\partial\beta_2}{\partial\hat{V}^i}\left(ge_3 + y_3y_2 - \Gamma_v\bar{H}\tilde{V}^i\right) \quad (3.24)$$

La dinámica del estimador del factor de escala se define en términos de las señales conocidas de la siguiente manera

$$\dot{\hat{K}} = \frac{\partial\beta_2}{\partial\hat{K}}^{-1}\left(-\frac{\partial\beta_2}{\partial y_1}\text{diag}(\beta_2)\hat{V}^i - \frac{\partial\beta_2}{\partial\hat{V}^i}(ge_3 + y_3y_2)\right) \quad (3.25)$$

Después de substituir $\dot{\hat{K}}$ en el error del estimador del factor de escala (3.24), se obtiene que

$$\dot{\hat{K}} = -\frac{\partial\beta_2}{\partial y_1}\left(\text{diag}(\tilde{K})\hat{V}^i + \text{diag}(\tilde{K} + \beta_2)\tilde{V}^i\right) + \frac{\partial\beta_2}{\partial\hat{V}^i}\Gamma_v\bar{H}\tilde{V}^i \quad (3.26)$$

Una vez más, la función β_2 necesita ser definida para asegurar que el error de estimación \tilde{K} converja a cero con $\frac{\partial \beta_2}{\partial \tilde{K}}$ una matriz invertible. Por lo tanto, se propone la siguiente función vectorial

$$\beta_2(\hat{K}, y_1, \hat{V}^i) = \hat{K} + \Gamma_k \text{diag}(y_1) \hat{V}^i = \hat{K} + \Gamma_k \text{diag}(\hat{V}^i) y_1 \quad (3.27)$$

con

$$\Gamma_k = \begin{bmatrix} \gamma_{kx} & 0 & 0 \\ 0 & \gamma_{ky} & 0 \\ 0 & 0 & \gamma_{kz} \end{bmatrix} \quad (3.28)$$

y $\gamma_{ki}, i = x, y, z$ las ganancias del estimador del factor de escala.

Reemplazando (3.27) en (3.26) se tiene

$$\dot{\tilde{K}} = -\Gamma_k \text{diag}(\hat{V}^i) \left(\text{diag}(\hat{V}^i) \tilde{K} + \text{diag}(\tilde{V}^i) K \right) + \Gamma_k \text{diag}(y_1) \Gamma_v \bar{H} \tilde{V}^i \quad (3.29)$$

donde (2.4), (2.5) y (3.17) han sido consideradas. De (3.15), se obtiene que

$$\begin{aligned} \dot{\tilde{K}} &= -\Gamma_k \left(\text{diag}(V^i) \text{diag}(V^i) + \text{diag}(\tilde{V}^i) \text{diag}(\tilde{V}^i) \right) \tilde{K} \\ &\quad + 2\Gamma_k \text{diag}(V^i) \text{diag}(\tilde{V}^i) \tilde{K} - \Gamma_k \left(\text{diag}(V^i) - \text{diag}(\tilde{V}^i) \right) \text{diag}(\tilde{V}^i) K \\ &\quad + \Gamma_k \text{diag}(y_1) \Gamma_v \bar{H} \tilde{V}^i \end{aligned} \quad (3.30)$$

Se considera que los siguientes suposiciones establecen el resultado principal de este trabajo de tesis.

Suposición 1 *Se cumple la siguiente igualdad:*

$$\lim_{t \rightarrow \infty} \int_0^t \text{diag}(V^i(\tau)) \text{diag}(V^i(\tau)) d\tau = \infty I$$

con $I \in \mathbb{R}^{3 \times 3}$ la matriz identidad.

Suposición 2 *Existen entradas de control T_T y M^b tales que los siguientes estados del cuadrirrotor pueden ser acotados en un límite superior, esto es*

$$\|V^i\| \leq \kappa_0, \quad \|y_1\| \leq \kappa_1 \quad (3.31)$$

*para algunas no necesariamente constantes κ_0 y κ_1 .*¹

Observación 1 *La Suposición 1 es la persistencia de la condición de excitación; en este caso, el cumplimiento de esta condición implica que el cuadrirrotor debe moverse para estimar el factor de escala con éxito. La Suposición 2 significa que un circuito de control permite que el cuadrirrotor vuele de manera estable; en consecuencia, la dinámica del cuadrirrotor es completa hacia adelante.*

La siguiente Proposición resume el principal resultado de este trabajo de tesis.

Proposición 1 *Bajo las Suposiciones 1 y 2, existen matrices de ganancias Γ_v y Γ_k tales que los sistemas dinámicos (3.7) y (3.25) son localmente un observador de la velocidad traslacional y un estimador del factor de escala, respectivamente. La velocidad traslacional y el factor de escala se reconstruyen en (3.11) y (3.27).*

Demostración de la Proposición 1 *La demostración se divide en los siguientes pasos. En el primer paso, se muestra que el error del observador de velocidad converge exponencialmente a cero. Luego, se prueba que las trayectorias del error de estimación no explotan en tiempo finito. Finalmente, se verifica que el término de interconexión entre la dinámica del observador y del estimador satisface una condición de crecimiento lineal al error del estimador.*

Note que al considerar (3.11), la derivada con respecto al tiempo de la función de Lyapunov (3.10) se puede escribir como

¹A continuación, la notación $\|(\cdot)\|$ representa la norma euclidiana para una matriz o vector (\cdot) .

$$\dot{\nu}_V = -\tilde{V}^b \Gamma_v \bar{H} \tilde{V}^b \quad (3.32)$$

como resultado, se tiene que \tilde{V}^b converge exponencialmente a zero.

Ahora, para probar que las trayectorias del error de estimación no explotan en un tiempo finito, considere la siguiente función de Lyapunov

$$\nu_K = \frac{1}{2} \tilde{K}^\top \tilde{K} \quad (3.33)$$

cuya derivada con respecto al tiempo a lo largo de la dinámica del estimador (3.30) es

$$\begin{aligned} \dot{\nu}_K = & -\tilde{K}^\top \Gamma_k \left(\text{diag}(V^i) \text{diag}(V^i) + \text{diag}(\tilde{V}^i) \text{diag}(\tilde{V}^i) \right) \tilde{K} \\ & + 2\tilde{K}^\top \Gamma_k \text{diag}(V^i) \text{diag}(\tilde{V}^i) \tilde{K} \\ & - \tilde{K}^\top \Gamma_k \left(\text{diag}(V^i) - \text{diag}(\tilde{V}^i) \right) \text{diag}(\tilde{V}^i) K \\ & + \tilde{K}^\top \Gamma_k \text{diag}(y_1) \Gamma_v \bar{H} \tilde{V}^i \end{aligned} \quad (3.34)$$

entonces,²

$$\begin{aligned} \dot{\nu}_K \leq & \lambda_M(\Gamma_k) \left(\|\text{diag}(V^i)\|^2 + \|\text{diag}(\tilde{V}^i)\|^2 \right) \|\tilde{K}\|^2 \\ & + 2\lambda_M(\Gamma_k) \|\text{diag}(V^i)\| \|\text{diag}(\tilde{V}^i)\| \|\tilde{K}\|^2 \\ & + \lambda_M(\Gamma_k) \left(\|\text{diag}(V^i)\| + \|\text{diag}(\tilde{V}^i)\| \right) \|\text{diag}(\tilde{V}^i)\| \|K\| \|\tilde{K}\| \\ & + \lambda_M(\Gamma_k) \|\text{diag}(y_1)\| \lambda_M(\Gamma_v) \lambda_M(\bar{H}) \|\tilde{V}^i\| \|\tilde{K}\| \end{aligned} \quad (3.35)$$

Por lo tanto, dado que la dinámica del cuadrirrotor es completa hacia adelante, la relación en (3.35) implica que la dinámica de \tilde{K} no explota en un tiempo finito. Tenga en cuenta que si la señal de error \tilde{V}^b es igual a cero, el error dinámico de estimación (3.30) se reduce a

$$\dot{\tilde{K}} = -\Gamma_k \text{diag}(V^i) \text{diag}(V^i) \tilde{K} \quad (3.36)$$

y la derivada con respecto al tiempo de la función de Lyapunov (3.33) a lo largo de las trayectorias de (3.36) se convierte en

² $\lambda_M(\cdot)$ representa el valor propio máximo de la matriz \cdot .

$$\dot{v}_K = -\tilde{K}^\top \Gamma_k \text{diag}(V^i) \text{diag}(V^i) \tilde{K} \quad (3.37)$$

entonces, de la Suposición 1 y seleccionando Γ_k como matriz definida positiva implica que el error de estimación \tilde{K} converge asintóticamente a cero.

El último paso en esta prueba es probar que el término de interconexión entre la dinámica del estimador y del observador dado por

$$\begin{aligned} \Psi &= -\Gamma_k \text{diag}(\tilde{V}^i) \text{diag}(\tilde{V}^i) \tilde{K} + 2\Gamma_k \text{diag}(V^i) \text{diag}(\tilde{V}^i) \tilde{K} \\ &\quad -\Gamma_k \left(\text{diag}(V^i) - \text{diag}(\tilde{V}^i) \right) \text{diag}(\tilde{V}^i) K \\ &\quad +\Gamma_k \text{diag}(y_1) \Gamma_v \tilde{H} \tilde{V}^i \end{aligned} \quad (3.38)$$

crece linealmente con respecto a \tilde{K} . De la Suposición 2 se tiene

$$\begin{aligned} \Psi &\leq \lambda_M(\Gamma_k) \left(\|\tilde{V}^i\|^2 + 2\kappa_0 \|\tilde{V}^i\| \right) \|\tilde{K}\| + \lambda_M(\Gamma_k) \left(\kappa_0 + \|\tilde{V}^i\| \right) \|\tilde{V}^i\| \|K\| \\ &\quad +\lambda_M(\Gamma_k) \kappa_1 \lambda_M(\Gamma_v) \lambda_M(\tilde{H}) \|\tilde{V}^i\| \end{aligned} \quad (3.39)$$

La desigualdad (3.39) muestra que el término de interconexión Ψ crece linealmente con respecto al error de estimación \tilde{K} ; por lo tanto, se cumple la Suposición 4.5 de la referencia (36). Como resultado, el sistema en cascada (3.30) y (3.8) es asintóticamente estable. Por lo tanto, la prueba está concluida.

Capítulo 4

Simulaciones Numéricas y Experimentación

Se realizó un estudio de simulación numérica en diferentes plataformas para evaluar el desempeño del observador y el estimador diseñados. Además, se realizaron algunos experimentos con cuadricopteros reales para tener un mayor marco de validación en situaciones prácticas.

4.1. Matlab-Simulink

La primer plataforma de simulación fue Matlab-Simulink, esto con el fin de evitar problemas como el ruido en los sensores y perturbaciones externas, y así poder evaluar los estimadores por sí mismos y sin interferencias no controlables. Se diseñó un programa para simular el control de un cuadricóptero en lazo cerrado, siguiendo la técnica de control desarrollada en (37), para seguir una trayectoria circular en el plano cartesiano y a la vez una forma sinusoidal en el plano vertical. Para cumplir con la Suposición 1, las trayectorias deseadas fueron $x_d = A\cos(\omega t)$, $y_d = A\sin(\omega t)$ y $z_d = A\cos(\omega t)$. Para el observador de velocidad, las condiciones iniciales utilizadas fueron $V^b(0) = \begin{bmatrix} -0,2 & 0,3 & 0,2 \end{bmatrix}^T$ con una $\mu = 0,6$ propuesta y ganancias

$\gamma_{vx} = 1,2$, $\gamma_{vy} = 1,2$ y $\gamma_{vz} = 1,2$. Con respecto al algoritmo de posicionamiento de visión monocular para esta simulación numérica, se puede usar cualquier valor real para el factor de escala K ; sin embargo, en simulaciones más realistas como las del siguiente apartado, se observa que la posición escalada siempre es menor que la posición real, $X_s < X$, por lo que $K < 1$; por lo tanto, los valores reales del factor de escala para esta simulación se fijaron en $K = [0,65 \ 0,7 \ 0,55]^T$. Las ganancias utilizadas para el estimador fueron $\gamma_{kx} = 2,0$, $\gamma_{ky} = 2,0$ y $\gamma_{kz} = 2,0$.

La Figura 4-1 muestra el error del observador de velocidad \tilde{V}^b para esta simulación, donde se puede ver que las velocidades convergen a cero correctamente. Tenga en cuenta que el error de velocidad en el eje Z , \tilde{w} , siempre es cero porque la velocidad en este eje se supone medible, (2.12).

La Figura 4-2 muestra el error del estimador del factor de escala \tilde{K} , que prueba la convergencia correcta del estimador. En esta gráfica también se puede observar el comportamiento en cascada de (3.30) con (3.8), lo que significa que \tilde{K} siempre convergerá después de \tilde{V}^b debido al término de interconexión Ψ .

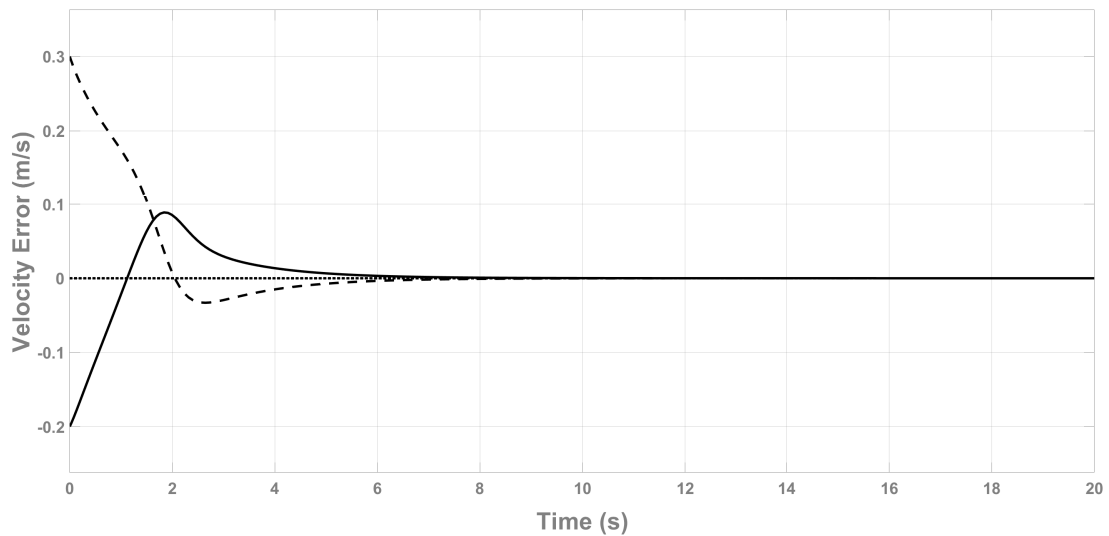


Figura 4-1: Error del observador de velocidad \tilde{V}^b . \tilde{u} (línea continua), \tilde{v} (línea discontinua), \tilde{w} (línea punteada).

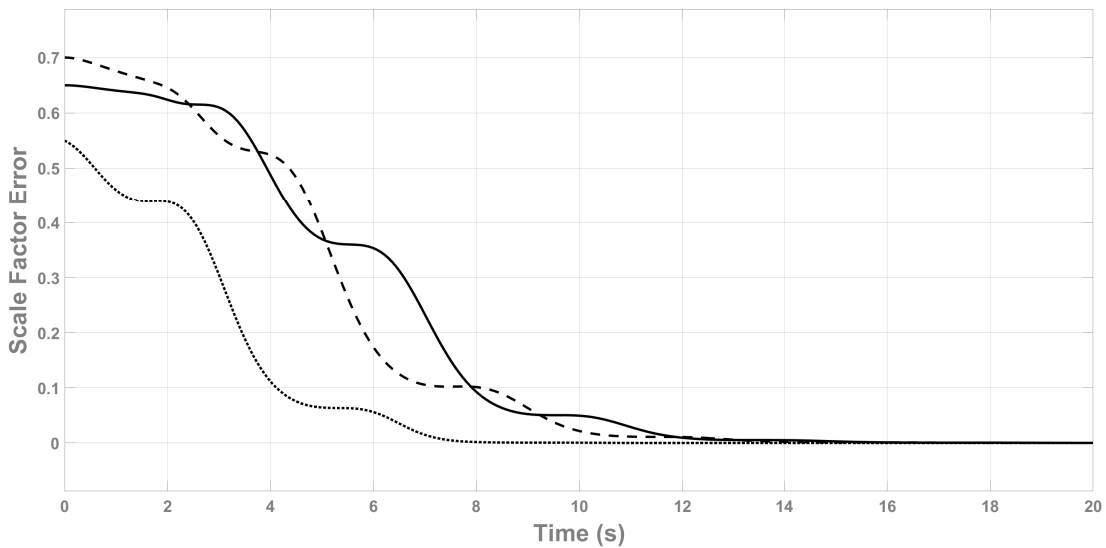


Figura 4-2: Error del estimador del factor de escala \tilde{K} . \tilde{k}_x (línea continua), \tilde{k}_y (línea discontinua), \tilde{k}_z (línea punteada).

4.2. Gazebo

Tras comprobar el correcto funcionamiento del observador y estimador en un entorno controlado, es hora de ponerlos a prueba en un simulador que se asemeje más a la realidad. Gazebo es un simulador de robótica 3D de código abierto. Incorpora el Motor de Dinámica Abierta, u ODE por sus siglas en inglés, como motor de físicas, OpenGL para gráficos avanzados, código de soporte para simulación de sensores y control de actuadores. El Sistema Operativo Robot, o ROS por sus siglas en inglés, y el simulador Gazebo están totalmente vinculados.

En la simulación anterior, fue fácil establecer valores de simulación para la constante μ y la posición escalada X_s proporcionada por un algoritmo de visión monocular. En la simulación de Gazebo, dichos valores deberán tratarse con más rigor, como se haría en un experimento real.

4.2.1. Cálculo de μ

μ es un término relacionado con el coeficiente de arrastre, una constante positiva que representa una combinación del perfil y las fuerzas de arrastre inducidas en los rotores, conocido en la literatura de helicópteros como “arrastre en el rotor” (*rotor drag*), referencias (4; 38). Como cualquier otro parámetro del cuadirrotor, al igual que su peso (m), por ejemplo, debe ser medido antes de realizar experimentos. Debido a las unidades de este parámetro, kg/s , se le conoce como “tasa de flujo másico” (*mass flow rate*).

De la ecuación (2.9), se puede ver que

$$\begin{aligned} a_x^b &= -\frac{\mu}{m}u \\ a_y^b &= -\frac{\mu}{m}v \end{aligned} \quad (4.1)$$

De cualquiera de estas dos ecuaciones, se puede medir la constante μ al tener acceso al acelerómetro y mediciones de velocidades traslacionales, por ejemplo

$$\mu = -\frac{a_x^b}{u}m \quad (4.2)$$

Se tiene acceso a ambas mediciones en Gazebo, por lo que se realizó una simulación de seguimiento de trayectoria circular para medir estos estados. El valor calculado para el cuadirrotor utilizado en Gazebo es $\mu = 0,18$.

Con la μ calculada, la Figura 4-3 muestra la ecuación (4.1) en el eje X con el cuadirrotor siguiendo una trayectoria circular, donde se puede ver que la relación se cumple, por lo que el cálculo de la μ es correcto. Tenga en cuenta que las lecturas del acelerómetro (a_x^b) tienen ruido agregado debido a los rotores.

4.2.2. Algoritmo de Visión Monocular

Como se mencionó anteriormente, el estimador del factor de escala diseñado funciona con cualquier algoritmo de visión por computadora

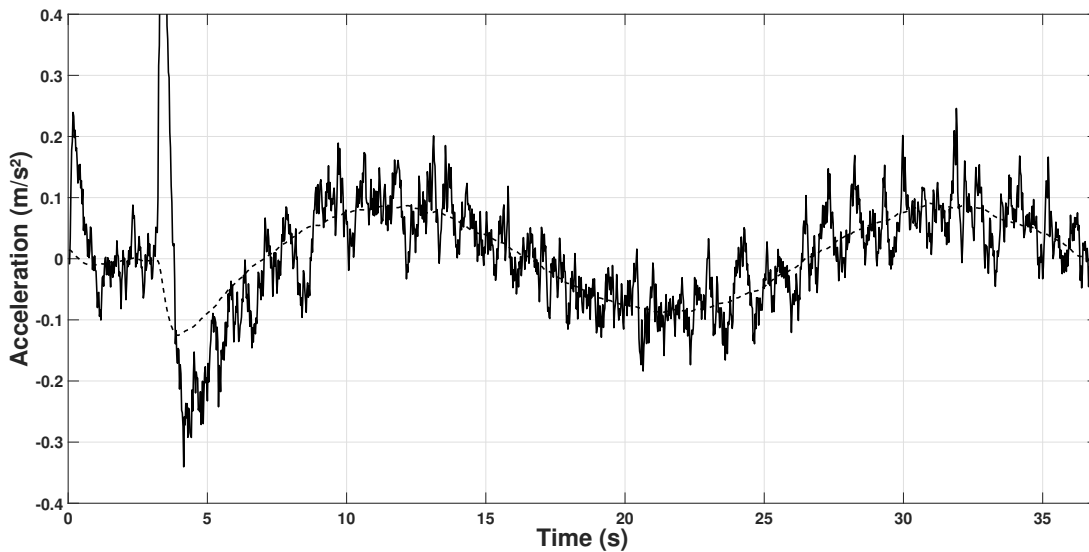


Figura 4-3: Relación entre el acelerómetro y μ en el eje X^b . a_x^b (línea continua), $-\frac{\mu}{m}u$ (línea discontinua).

que implemente una cámara monocular y proporcione mediciones de posición. Para esta simulación, usamos ORB-SLAM2, referencia (23), debido a su precisión y exactitud, es fácil de instalar y está bien documentado, además de que tiene integración con ROS. Por lo tanto, es fácil de agregar al entorno de Gazebo.

4.2.3. Seguimiento de Trayectoria Usando el Estimator del Factor de Escala

La simulación consiste en hacer volar el cuadricóptero que ejecuta ORB-SLAM2, a partir de la información visual obtenida por su cámara monocular a bordo, sin control de posición (lazo abierto) siguiendo una trayectoria diagonal en el plano horizontal para obtener la información requerida para mantener las identidades en (3.1). Después de este paso, el factor de escala K se conocerá a través de β_2 , por lo que el cuadricóptero podrá medir su posición real con y_1 , (2.3). Entonces, el cuadricóptero seguirá una trayectoria de lemniscata de forma autónoma

(lazo cerrado).

El controlador de bajo nivel del cuadrirrotor está manejado por (39), que es un controlador para interactuar con los cuadrirrotores Parrot AR-Drone a través de ROS. Toma velocidades de traslación como entradas de control.

La Figura 4-4 muestra el entorno de Gazebo con el algoritmo ORB-SLAM ejecutándose. La ventana de la izquierda muestra lo que ve la cámara integrada con las características (puntos) detectadas, y la ventana de la derecha muestra la construcción del mapa.

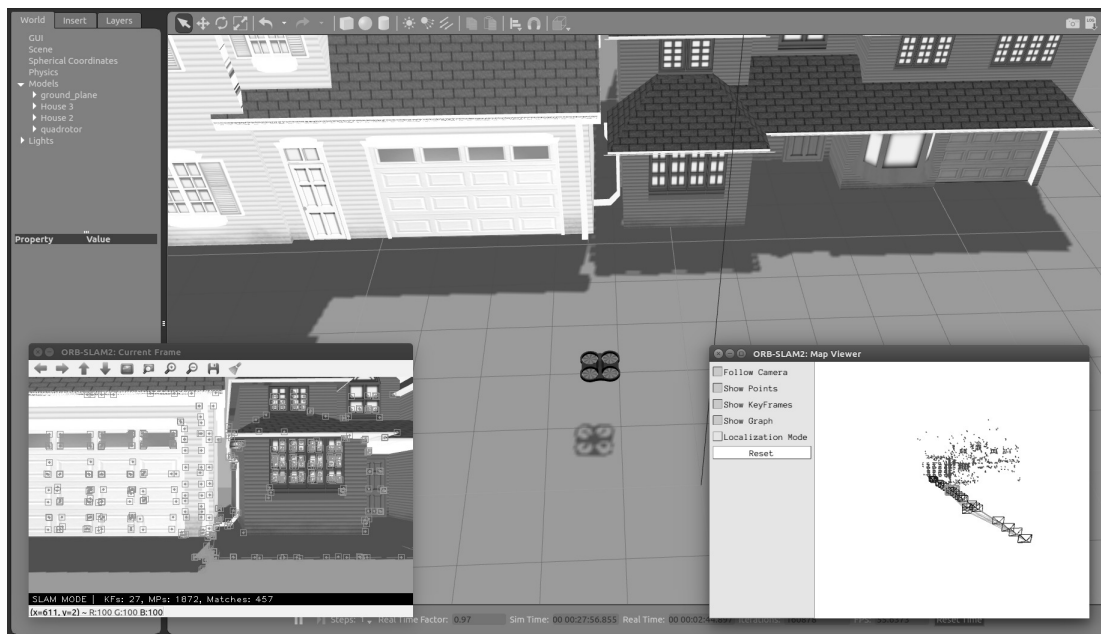


Figura 4-4: Entorno de Gazebo.

El primer paso es volar el cuadrirrotor en lazo abierto sobre una trayectoria diagonal desde la posición $(0, 0)$ a $(3, 3)$, luego a la posición $(-3, -3)$ y finalmente de regreso al origen $(0, 0)$, manteniendo siempre una velocidad constante de $u_d = 0,2$ y $v_d = 0,2$.

Las Figuras 4-5 y 4-6 muestran la velocidad real y la estimada por el observador. Siendo iguales las condiciones iniciales de la velocidad real y del observador, $V^b(0) = [0 \ 0 \ 0]^T$, y las altas ganancias $\gamma_{vx} = 30$,

$\gamma_{vy} = 30$ y $\gamma_{vz} = 30$, hacen que el observador converja inmediatamente.

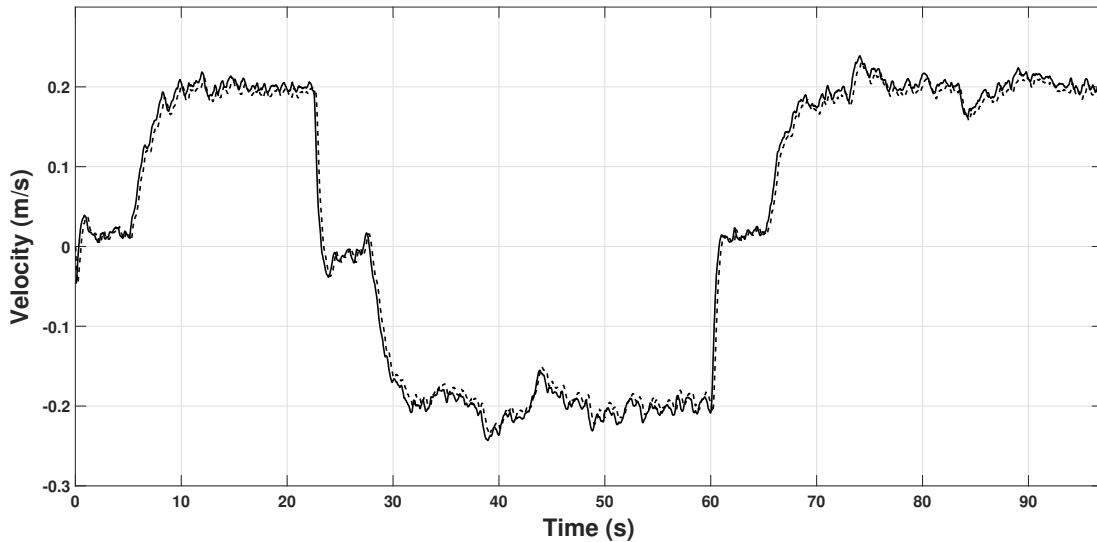


Figura 4-5: Velocidad observada en el eje X^b . u (línea continua), β_{1x} (línea discontinua).

Las Figuras 4-7 y 4-8 muestran la estimación del factor de escala en los ejes X y Y , respectivamente. Obsérvese que desde el tiempo 0s hasta el 6s, el estimador permanece en cero porque, en ese período, el cuadricóptero despegó y se mantiene en vuelo estacionario durante unos segundos, por lo que la Suposición 1 no se cumple. En el tiempo 22s y 60s aparecen unos picos porque el cuadricóptero alcanza las coordenadas $(3, 3)$ y $(-3, -3)$, respectivamente. Por lo tanto, cambia de velocidad abruptamente para cambiar su dirección de movimiento para llegar al siguiente punto. Estos cambios en la velocidad también se pueden ver en las Figuras 4-5 y 4-6 en los mismos instantes de tiempo.

Después de la rutina de lazo abierto, se calcula que β_2 converge en $k_x = 0,175$ y $k_y = 0,13$. También se puede ver en las Figuras 4-7 y 4-8 que \tilde{K} converge exponencialmente, como se anticipó en (3.36). A partir de este punto, conociendo el factor de escala K , el cuadricóptero podrá medir su propia posición real a través de y_1 en (2.3) de la siguiente manera:

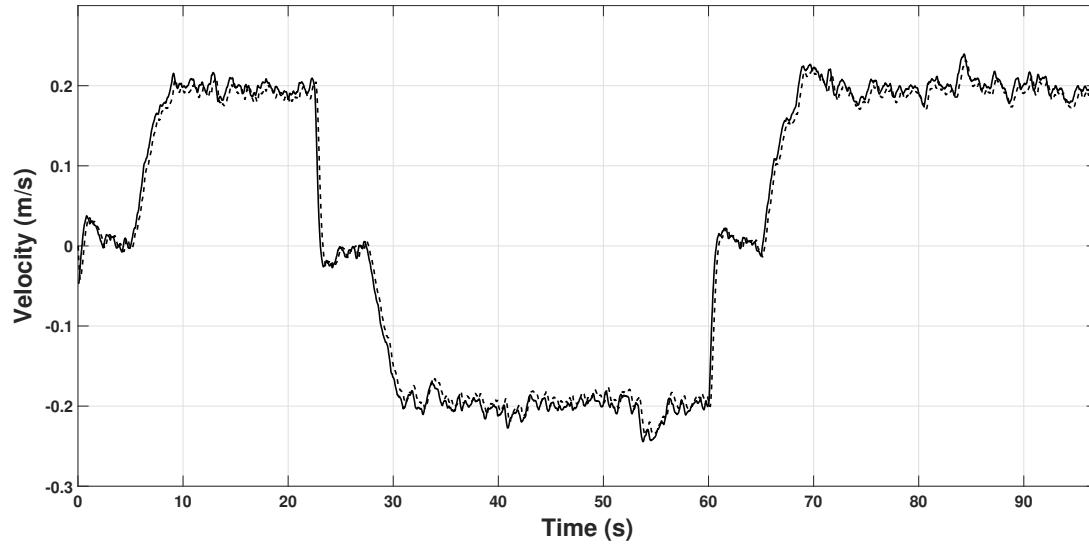


Figura 4-6: Velocidad observada en el eje Y^b . v (línea continua), β_{1y} (línea discontinua).

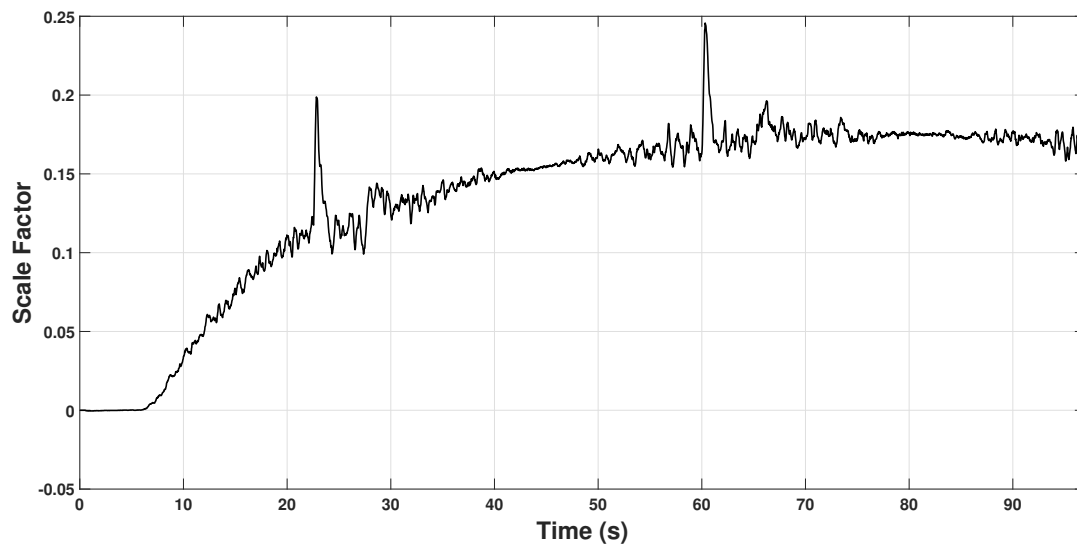


Figura 4-7: Factor de escala estimado β_{2x} en el eje X^i .

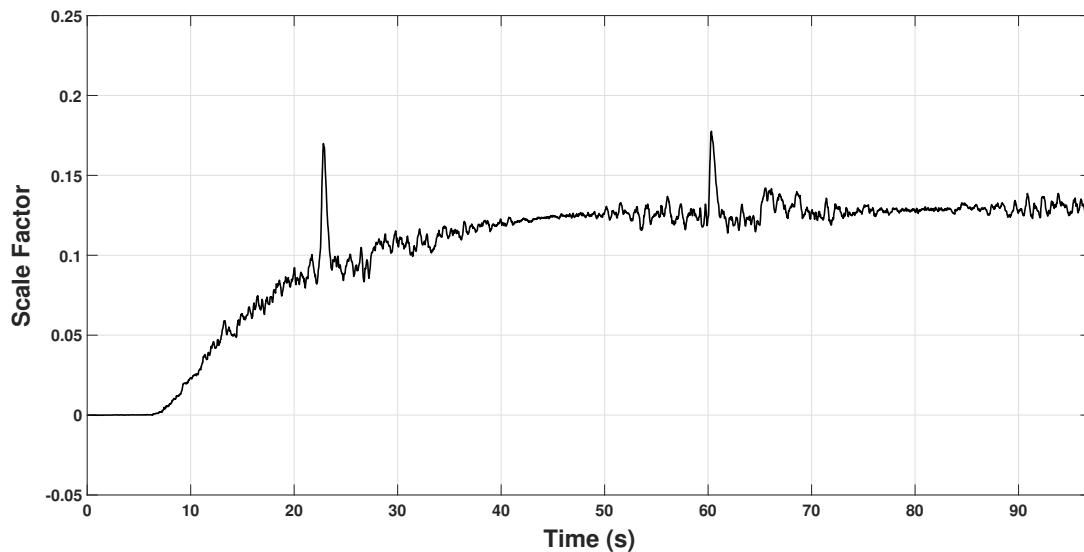


Figura 4-8: Factor de escala estimado β_{2y} en el eje Y^i .

$$X = \text{diag}(K)^{-1}y_1 \quad (4.3)$$

Finalmente, utilizando la información estimada, el cuadrirrotor seguirá una trayectoria de lemniscata de forma autónoma en lazo cerrado; es decir, $x_d = 1,5\cos(\omega t)$ y $y_d = \text{sen}(2\omega t)$. Las Figuras 4-9 y 4-10 muestran la posición real y la posición medida por el cuadrirrotor en el eje X y Y , respectivamente, a lo largo de la trayectoria de la lemniscata.

La Figura 4-11 muestra el gráfico XY de la trayectoria que valida el control de lazo cerrado del cuadrirrotor usando el estimador de factor de escala propuesto. El cuadrirrotor completó la trayectoria completa de la lemniscata dos veces con el tiempo de simulación dado. La flecha grande indica la dirección en la que mira la cámara a bordo durante toda la trayectoria, la línea de la izquierda representa la ubicación de los edificios y las flechas pequeñas indican la dirección del movimiento del cuadrirrotor.

Como se muestra en la Figura 4-11, el algoritmo SLAM tiene mejor precisión en la parte izquierda de la lemniscata porque la cámara está más cerca de los edificios, por lo que más características de la imagen

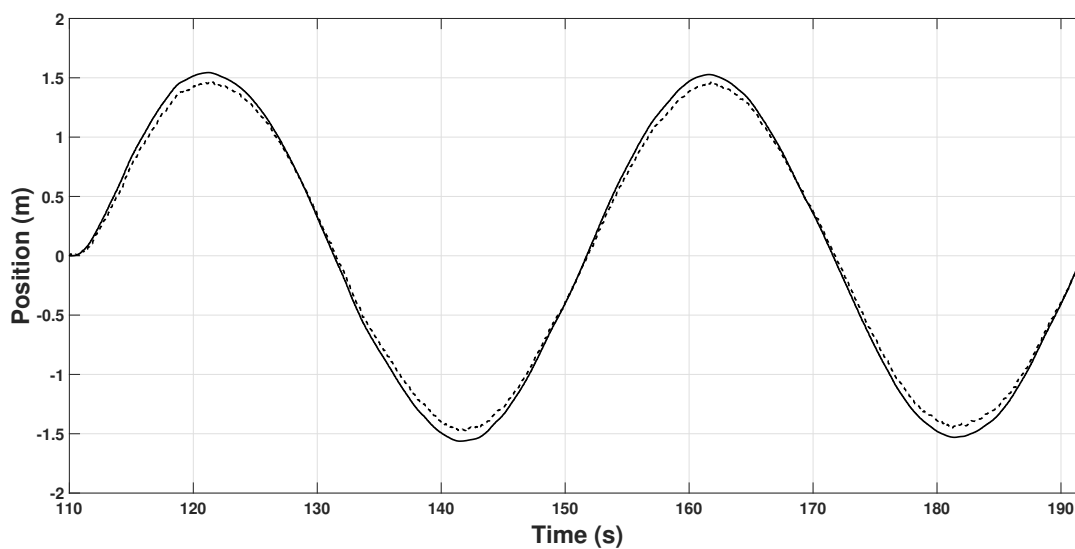


Figura 4-9: Posición del cuadrirrotor en el eje X^i . x (línea continua), $\frac{x_s}{k_x}$ (línea discontinua).

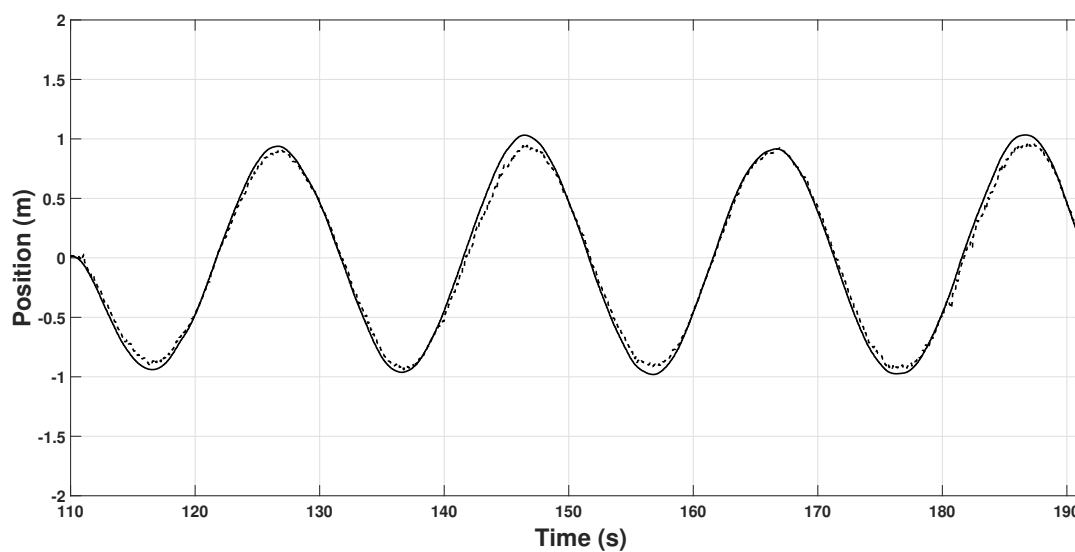


Figura 4-10: Posición del cuadrirrotor en el eje Y^i . y (línea continua), $\frac{y_s}{k_y}$ (línea discontinua).

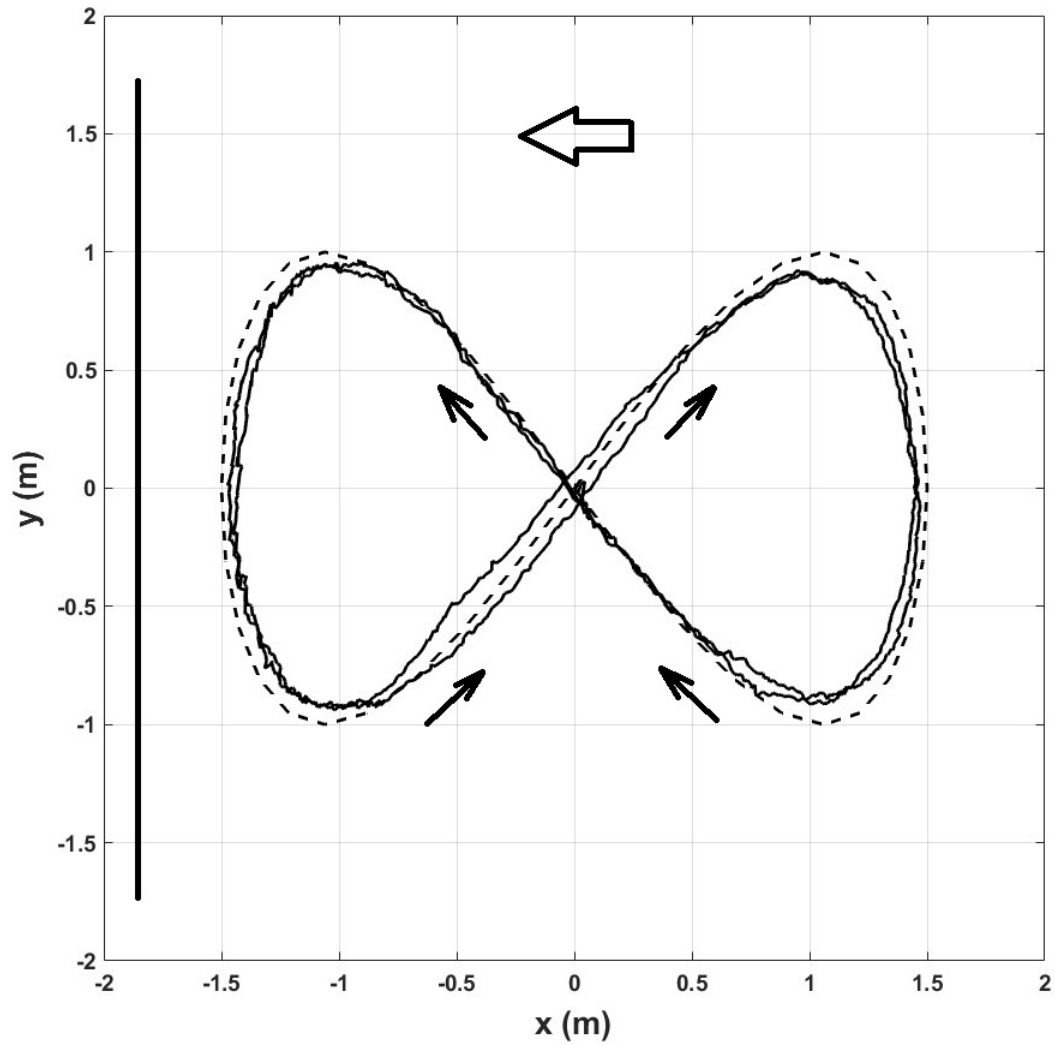


Figura 4-11: Seguimiento de la trayectoria de lemniscata utilizando el estimador del factor de escala. Posición real (línea continua), posición deseada (línea discontinua).

50CAPÍTULO 4. SIMULACIONES NUMÉRICAS Y EXPERIMENTACIÓN

alimentaron el algoritmo. En las diagonales, el algoritmo SLAM funciona mejor cuando el cuadirrotor se acerca a los edificios que cuando se aleja. Aunque el estimador del factor de escala recupera la dimensión de la posición real, no ayuda en ningún sentido a mejorar el rendimiento del algoritmo SLAM.

Cambiando la posición de la casa de la derecha en el entorno de Gazebo, se realizó una segunda simulación para comprobar si pequeños cambios en las condiciones iniciales del algoritmo de visión monocular influyen en el valor final del factor de escala. En este caso, la casa de la derecha está más cerca del cuadirrotor, como se muestra en la Figura 4-12. Al final de la simulación, los valores del factor de escala calculados fueron $k_x = 0,232$ y $k_y = 0,166$, lo que demuestra que el factor de escala es diferente incluso para el mismo entorno pero con cambios menores en las condiciones iniciales.

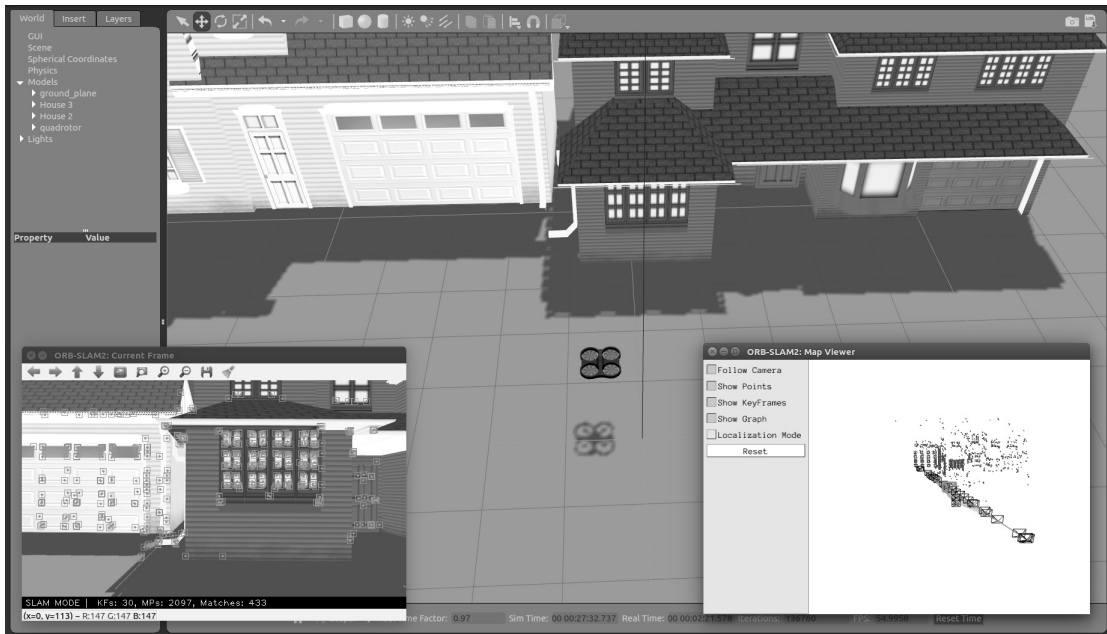


Figura 4-12: Entorno de Gazebo con diferentes condiciones iniciales.

4.3. Experimentos

4.3.1. Cuadrirrotor Experimental

En el primer año del desarrollo de este trabajo tesis se estuvo trabajando la idea de construir desde cero un cuadrirrotor con las características técnicas necesarias para poder llevar a cabo todas las pruebas experimentales requeridas para la validación de esta investigación.

Al ser necesaria la implementación de un algoritmo de visión por computadora, se decidió por utilizar una computadora en placa reducida, o SBC por sus siglas en inglés, como procesador central del cuadrirrotor, y de entre las cuales se seleccionó la Raspberry Pi por su popularidad en el mercado, bajo costo y amplia documentación disponible.

En la Figura 4-13 se muestra el cuadrirrotor diseñado, y en la Figura 4-14 un diagrama de sus principales elementos e interconexión entre ellos.

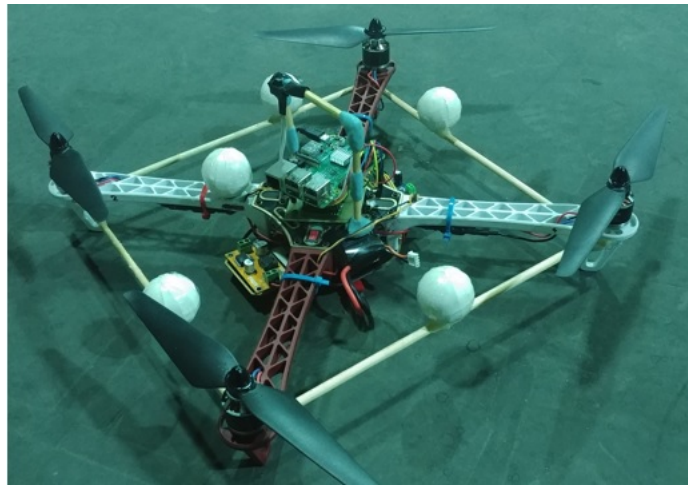


Figura 4-13: Cuadrirrotor controlado por una computadora en placa reducida.

Sin embargo, las primeras pruebas de control dejaron en evidencia abundantes problemas técnicos que era necesario resolver, como ruido

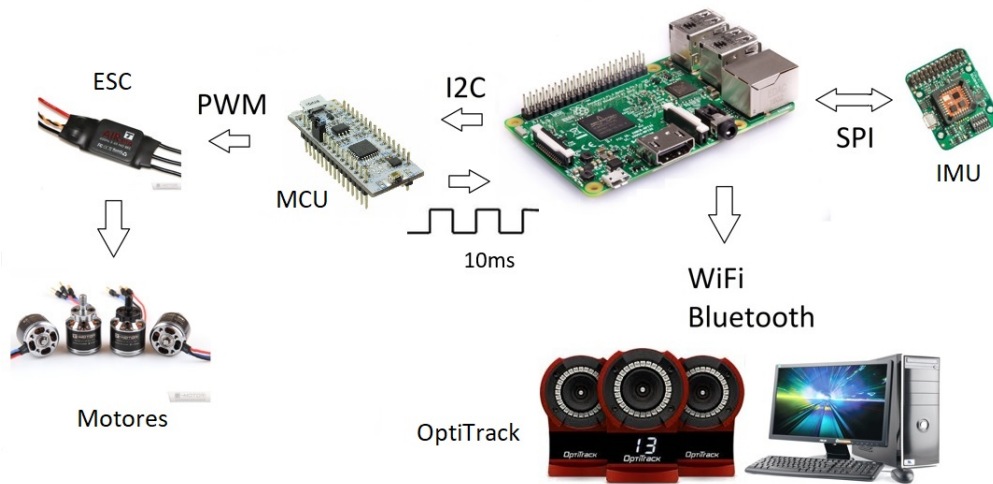


Figura 4-14: Sistema de comunicación del cuadricóptero experimental.

abundante en los sensores, caracterización de los controladores electrónicos de velocidad (ESC) y los motores, ejecución en tiempo real del CPU, implementación de paros de emergencia, entre otros. Por lo que se decidió descartar esta idea y utilizar un cuadricóptero comercial como plataforma experimental ya que el solucionar los problemas presentes se salía del objetivo principal de este trabajo.

En el artículo (40) de 2018 se presenta el desarrollo de este cuadricóptero, así como una prueba experimental de vuelo suspendido (*hover*) y consideraciones técnicas a tener en cuenta en el diseño y construcción de un cuadricóptero basado en una SBC.

4.3.2. Estimador con AR.Drone

La finalidad de este experimento es validar en un cuadricóptero real el correcto funcionamiento y convergencia del observador y estimador diseñados, utilizando únicamente como entradas los sensores a bordo del mismo cuadricóptero. Para ello, se utilizó el cuadricóptero comercial AR.Drone 2.0 de Parrot, junto con un sistema de captura de movimiento, llamado OptiTrack, el cual proporcionará los valores de referencia

de velocidad y posición. La comunicación entre el cuadrirrotor, la interfaz de usuario y el sistema de captura de movimiento están gestionado por ROS. Este cuadrirrotor tiene un a bordo cámara, como se puede ver en la Figura 4-15, y transmite la señal de video a una computadora remota en el que se estará ejecutando el algoritmo ORB-SLAM.



Figura 4-15: AR.Drone 2.0 de Parrot.

Es importante mencionar que este experimento se llevó a cabo en una etapa de diseño más temprana a la presentada en el Capítulo 3, en donde básicamente la diferencia radica en que el observador y estimador se habían diseñado considerando únicamente el plano horizontal XY , es decir, excluyendo el eje Z , ya que entonces no se había encontrado una forma de manejar las ecuaciones considerando los 3 ejes. Información más detallada de esta etapa de diseño se puede consultar en el artículo (41) publicado en 2020.

En este experimento, el cuadrirrotor realiza un seguimiento de trayectoria circular en el plano cartesiano mientras calcula en tiempo real el observador y estimador propuestos, utilizando la misma técnica de control que en las simulaciones numéricas. La Figura 4-16 muestra la posición del cuadrirrotor medida por el sistema de captura de movimiento. La Figura 4-17 muestra con línea azul la velocidad traslacional del cuadrirrotor calculada a partir de la posición entregada por el sistema Optitrack, y con línea roja la velocidad calculada por el observador. La Figura 4-18 presenta en línea azul los factores de escala estimados,

y con línea roja los valores reales de k_x y k_y , calculados con la ecuación (2.3), tomando la posición real X del sistema de captura de movimiento y X_s del algoritmo ORB-SLAM.

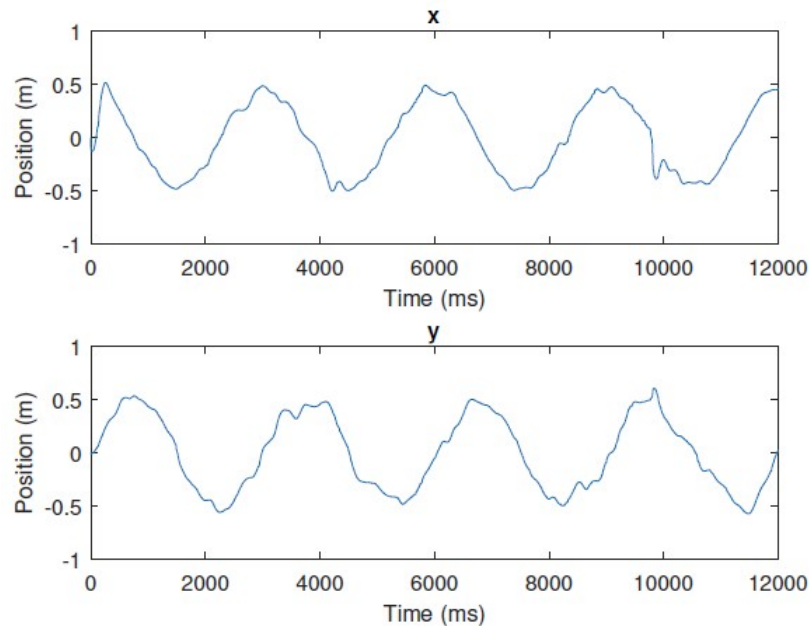


Figura 4-16: Posición del cuadrirrotor a lo largo de la trayectoria circular.

Puede observarse que las velocidad traslacional estimada converge a los valores reales casi inmediatamente, al igual que en la simulación numérica. Note que en el tiempo 10000ms, el factor de escala estimado en el eje X , ver Figura 4-18, hizo un cambio abrupto; esto se debe a que el cuadrirrotor perdió la comunicación con el sistema de captura de movimiento por un instante e hizo una movimiento repentino hacia otra dirección. Esto se debe a que la principal fuente de información para el estimador del factor de escala es la velocidad traslacional, como establece la Suposición 1. Este comportamiento de cambios abruptos en la estimación en función de cambios abruptos en la velocidad traslacional también se vio en las simulaciones numéricas.

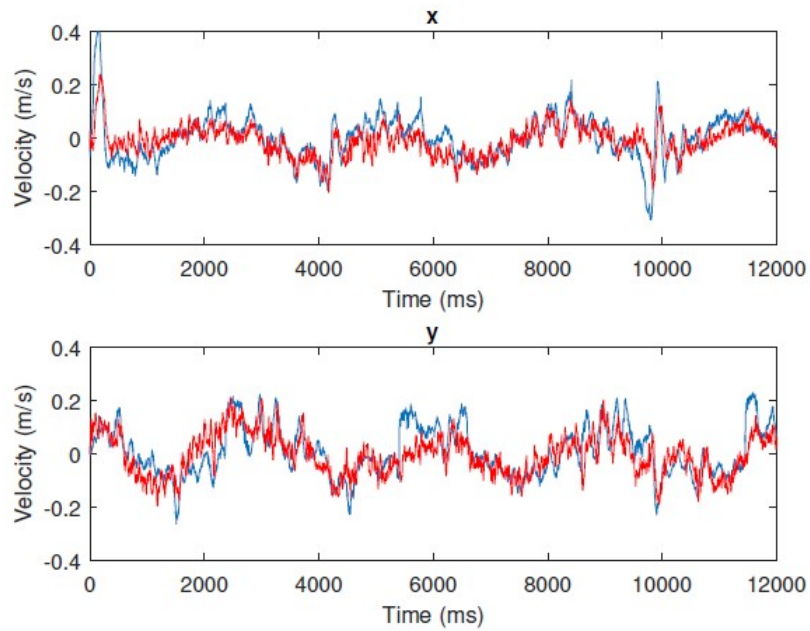


Figura 4-17: Velocidad traslacional real y estimada.

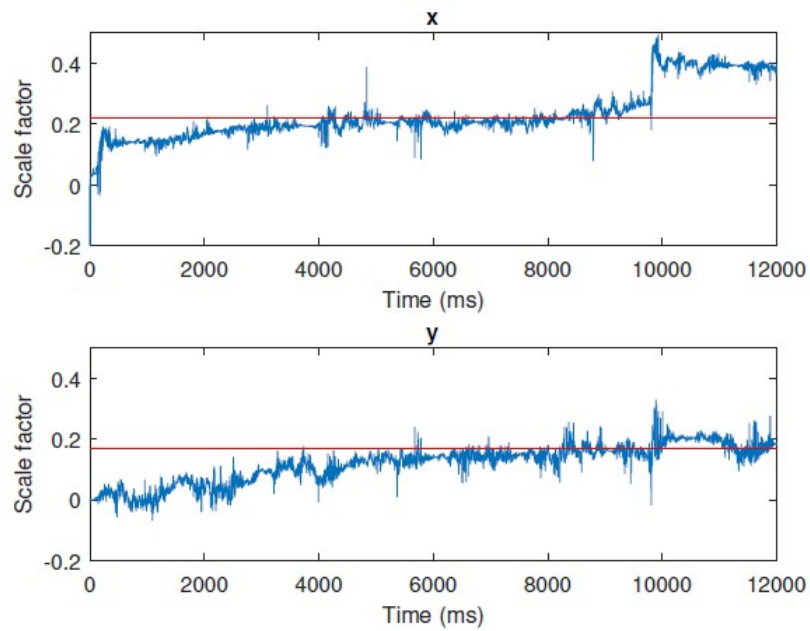


Figura 4-18: Factor de escala real y estimado.

Capítulo 5

Conclusiones y Trabajos a Futuro

Este trabajo de tesis propuso un observador de velocidad traslacional en cascada con un estimador del factor de escala para algoritmos de visión monocular. Las contribuciones más significativas de este trabajo se enumeran a continuación, así como las oportunidades que abre para futuros desarrollos y trabajos de investigación.

5.1. Conclusiones

- El observador de velocidad traslacional no descarta el término debido a la fuerza de Coriolis, a diferencia de la mayoría de trabajos sobre este tema, ofreciendo mayor precisión en vuelos a altas velocidades y trayectorias complicadas.
- El estimador de factor de escala permite aprovechar todas las ventajas de las cámaras monoculares, obteniendo la precisión de una cámara estereoscópica sin tener que aumentar la potencia de procesamiento a bordo.
- Los argumentos de Lyapunov demuestran la convergencia asintótica a cero de los errores del observador y del estimador, y las

simulaciones numéricas y experimentales validan el correcto desempeño y uso de la teoría propuesta.

- Se ilustra que el factor de escala no es igual en todos los ejes, como suponen algunos autores. Incluso es diferente al realizar experimentos en el mismo entorno si las condiciones iniciales cambian.
- El enfoque propuesto permite un seguimiento de trayectoria utilizando directamente las mediciones de un algoritmo de posicionamiento de visión monocular, eliminando las limitantes de un GPS o un sistema de captura de movimiento.

5.2. Trabajos a Futuro

- Navegación autónoma de un cuadirrotor real con el suficiente poder computacional como para ejecutar el algoritmo de visión monocular a bordo.
- Implementación de otros tipos de algoritmos de visión por computadora, como reconocimiento de personas o evasión de obstáculos, para robustecer la navegación autónoma.
- Diseño y validación de un estimador para el parámetro μ .
- Retroalimentación del factor de escala una vez calculado, en el algoritmo de visión monocular para mejorar su exactitud y precisión.
- Navegación autónoma utilizando diferentes técnicas de control, además de la presentada en este trabajo.
- Implementación de diferentes algoritmos de visión monocular, como SVO, homografía o SfM, para enriquecer la validación del estimador propuesto.

Bibliografía

- [1] S. Gupte, Paul Infant Teenu Mohandas, and J. M. Conrad, “A survey of quadrotor unmanned aerial vehicles,” in *2012 Proceedings of IEEE Southeastcon*, pp. 1–6, 2012.
- [2] A. Cavallo, A. Cirillo, P. Cirillo, G. De Maria, P. Falco, C. Natale, and S. Pirozzi, “Experimental comparison of sensor fusion algorithms for attitude estimation,” *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 7585–7591, 2014. 19th IFAC World Congress.
- [3] P. Martin and E. Salaün, “The true role of accelerometer feedback in quadrotor control,” in *2010 IEEE International Conference on Robotics and Automation*, pp. 1623–1629, 2010.
- [4] R. C. Leishman, J. C. Macdonald, R. W. Beard, and T. W. McLain, “Quadrotors and accelerometers: State estimation with an improved dynamic model,” *IEEE Control Systems Magazine*, vol. 34, no. 1, pp. 28–41, 2014.
- [5] G. Balamurugan, J. Valarmathi, and V. P. S. Naidu, “Survey on uav navigation in gps-denied environments,” in *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, pp. 198–204, 2016.
- [6] N. Xie, X. Lin, and Y. Yu, “Position estimation and control for quadrotor using optical flow and gps sensors,” in *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pp. 181–186, 2016.

-
- [7] L. Arreola, A. Montes de Oca, A. Flores, J. Sanchez, and G. Flores, "Improvement in the uav position estimation with low-cost gps, ins and vision-based system: Application to a quadrotor uav," in *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 1248–1254, 2018.
- [8] D. Scaramuzza and F. Fraundorfer, "Visual odometry [tutorial]," *IEEE Robotics Automation Magazine*, vol. 18, no. 4, pp. 80–92, 2011.
- [9] M. L. Anderson, K. M. Brink, and A. R. Willis, "Real-time visual odometry covariance estimation for unmanned air vehicle navigation," *Journal of Guidance, Control, and Dynamics*, vol. 42, no. 6, pp. 1272–1288, 2019.
- [10] Y. Chen, Y. Zhou, Q. Lv, and K. K. Deveerasetty, "A review of v-slam," in *2018 IEEE International Conference on Information and Automation (ICIA)*, pp. 603–608, 2018.
- [11] J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam," in *Computer Vision – ECCV 2014* (D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, eds.), (Cham), pp. 834–849, Springer International Publishing, 2014.
- [12] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "Orb-slam: A versatile and accurate monocular slam system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.
- [13] M. Filipenko and I. Afanasyev, "Comparison of various slam systems for mobile robot in an indoor environment," in *2018 International Conference on Intelligent Systems (IS)*, pp. 400–407, 2018.
- [14] B. Mouaad, B. Z. Razika, R. H. Ramzi, and C. Karim, "Control design and visual autonomous navigation of quadrotor," in *2019 International Conference on Advanced Electrical Engineering (ICAEE)*, pp. 1–7, 2019.

-
- [15] C. Forster, M. Pizzoli, and D. Scaramuzza, “Svo: Fast semi-direct monocular visual odometry,” in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 15–22, 2014.
- [16] Y. Lin, J. Wang, Z. Shi, and Y. Zhong, “Reinitializable and scale-consistent visual navigation for uavs,” in *2017 36th Chinese Control Conference (CCC)*, pp. 5871–5876, 2017.
- [17] A. Al-Kaff, A. de La Escalera, and J. M. Armingol, “Homography-based navigation system for unmanned aerial vehicles,” in *Advanced Concepts for Intelligent Vision Systems* (J. Blanc-Talon, R. Penne, W. Philips, D. Popescu, and P. Scheunders, eds.), (Cham), pp. 288–300, Springer International Publishing, 2017.
- [18] M. Hua, T. Hamel, R. Mahony, and G. Allibert, “Explicit complementary observer design on special linear group $sl(3)$ for homography estimation using conic correspondences,” in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pp. 2434–2441, 2017.
- [19] M.-D. Hua, J. Trumpf, T. Hamel, R. Mahony, and P. Morin, “Non-linear observer design on $sl(3)$ for homography estimation by exploiting point and line correspondences with application to image stabilization,” *Automatica*, vol. 115, p. 108858, 2020.
- [20] J. L. Schonberger and J.-M. Frahm, “Structure-from-motion revisited,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
- [21] J. Tharp and B. G. Woolley, *Enhancing Inertial Navigation with Structure from Motion Trajectory Estimates*.
- [22] J. Engel, J. Stückler, and D. Cremers, “Large-scale direct slam with stereo cameras,” in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1935–1942, 2015.

-
- [23] R. Mur-Artal and J. D. Tardós, “Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras,” *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.
- [24] C. Forster, Z. Zhang, M. Gassner, M. Werlberger, and D. Scaramuzza, “Svo: Semidirect visual odometry for monocular and multicamera systems,” *IEEE Transactions on Robotics*, vol. 33, no. 2, pp. 249–265, 2017.
- [25] R. Giubilato, M. Pertile, and S. Debei, “A comparison of monocular and stereo visual fastslam implementations,” 06 2016.
- [26] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. USA: Cambridge University Press, 2 ed., 2003.
- [27] O. Esrafilian and H. D. Taghirad, “Autonomous flight and obstacle avoidance of a quadrotor by monocular slam,” in *2016 4th International Conference on Robotics and Mechatronics (ICROM)*, pp. 240–245, 2016.
- [28] Z. Zhang, R. Zhao, E. Liu, K. Yan, and Y. Ma, “Scale estimation and correction of the monocular simultaneous localization and mapping (slam) based on fusion of 1d laser range finder and vision data,” *Sensors*, vol. 18, no. 6, 2018.
- [29] D. Zhou, Y. Dai, and H. Li, “Reliable scale estimation and correction for monocular visual odometry,” in *2016 IEEE Intelligent Vehicles Symposium (IV)*, pp. 490–495, 2016.
- [30] D. Zhou, Y. Dai, and H. Li, “Ground-plane-based absolute scale estimation for monocular visual odometry,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 2, pp. 791–802, 2020.
- [31] M. Ludhiyani, V. Rustagi, R. Dasgupta, and A. Sinha, “Multirotor dynamics based online scale estimation for monocular slam,” in

- 2019 International Conference on Robotics and Automation (ICRA)*, pp. 6475–6481, 2019.
- [32] S. H. Lee and G. de Croon, “Stability-based scale estimation for monocular slam,” *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 780–787, 2018.
- [33] L. Nieto-Hernández, A. A. Gómez-Casasola, and H. Rodríguez-Cortés, “Monocular slam position scale estimation for quadrotor autonomous navigation,” in *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 1359–1364, 2019.
- [34] R. Mahony, V. Kumar, and P. Corke, “Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor,” *IEEE Robotics Automation Magazine*, vol. 19, no. 3, pp. 20–32, 2012.
- [35] A. Astolfi and R. Ortega, “Immersion and invariance: a new tool for stabilization and adaptive control of nonlinear systems,” *IEEE Transactions on Automatic Control*, vol. 48, no. 4, pp. 590–606, 2003.
- [36] R. Sepulchre, M. Jankovic, and P. V. Kokotovic, *Constructive Nonlinear Control*. Springer London, 2012.
- [37] H. Rodríguez-Cortés, “Aportaciones al control de vehículos aéreos no tripulados en México,” *Revista Iberoamericana de Automática e Informática industrial*, jun. 2022.
- [38] A. R. S. Bramwell, G. Done, and D. Balmford, *Bramwell’s Helicopter Dynamics*. Oxford: Butterworth-Heinemann, 2nd ed., 2001.
- [39] M. Monajjemi, “ardrone_autonomy: A ros driver for ar-drone 1.0 & 2.0,” 2012.
- [40] A. Gómez-Casasola and H. Rodríguez-Cortés, “Control en tiempo real de un cuatrirotor empleando una computadora en placa reducida,” in *Congreso Nacional de Control Automático (CNCA)*, pp. 418–423, 2018.

-
- [41] A. Gómez-Casasola and H. Rodríguez-Cortés, “Sensor fusion for quadrotor autonomous navigation,” in *2020 American Control Conference (ACC)*, pp. 5219–5224, 2020.
- [42] R. Ganesan, X. M. Raajini, A. Nayyar, P. Sanjeevikumar, E. Hosain, and A. H. Ertas, “Bold: Bio-inspired optimized leader election for multiple drones,” *Sensors*, vol. 20, no. 11, 2020.
- [43] A. Astolfi, D. Karagiannis, and R. Ortega, *Nonlinear and Adaptive Control with Applications*, vol. 187. Springer, 2008.
- [44] T. Hamel, R. Mahony, J. Trumpf, P. Morin, and M. Hua, “Homography estimation on the special linear group based on direct point correspondence,” in *2011 50th IEEE Conference on Decision and Control and European Control Conference*, pp. 7902–7908, 2011.
- [45] Q. Lv, J. Ma, G. Wang, and H. Lin, “Absolute scale estimation of orb-slam algorithm based on laser ranging,” in *2016 35th Chinese Control Conference (CCC)*, pp. 10279–10283, 2016.
- [46] E. Kobzili, C. Larbes, and A. Allam, “Multi-rate robust scale estimation of monocular slam,” in *2017 6th International Conference on Systems and Control (ICSC)*, pp. 1–5, 2017.
- [47] E. Sucar and J. Hayet, “Bayesian scale estimation for monocular slam based on generic object detection for correcting scale drift,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5152–5158, 2018.
- [48] Y. Lin, F. Gao, T. Qin, W. Gao, T. Liu, W. Wu, Z. Yang, and S. Shen, “Autonomous aerial navigation using monocular visual-inertial fusion,” *Journal of Field Robotics*, vol. 35, no. 1, pp. 23–51, 2018.
- [49] Zhaolin Yang, Feng Lin, and B. M. Chen, “Survey of autopilot for multi-rotor unmanned aerial vehicles,” in *IECON 2016 -*

-
- 42nd Annual Conference of the IEEE Industrial Electronics Society*, pp. 6122–6127, 2016.
- [50] A. Garcia, S. S. Mittal, E. Kiewra, and K. Ghose, “A convolutional neural network vision system approach to indoor autonomous quadrotor navigation,” in *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 1344–1352, 2019.
- [51] J. J. Corona-Sánchez and H. Rodríguez-Cortés, “Trajectory tracking control for a rotary wing vehicle powered by four rotors,” *Journal of Intelligent & Robotic Systems*, vol. 70, pp. 39–50, Apr 2013.