UNIVERSITAT POLITÈCNICA DE CATALUNYA

Programa de doctorat

AUTOMÀTICA, ROBÒTICA I VISIÓ

Tesis Doctoral

Estabilización de vídeo en tiempo real: Aplicaciones en teleoperación de micro vehículos aéreos de ala rotativa

Wilbert Geovanny Aguilar Castillo

Director de Tesis: Cecilio Angulo Bahón

2015

A mis padres, Lucía y Erasmo, a mis hermanos, Jenner, Fabián y Richard,

y a mi motivación, Vanessa.

Resumen

Los micro vehículos aéreos (MAVs), un subconjunto de vehículos aéreos no tripulados (UAVs), también llamados drones, han ganado popularidad en múltiples aplicaciones y un creciente interés debido a sus ventajas como costo de fabricación y mantenimiento, volumen, peso del vehículo, gasto energético, y maniobrabilidad de vuelo.

La destreza requerida para un teleoperador de drones es inferior a la de un piloto de aeronaves de mayor dimensión, no obstante, su proceso de entrenamiento puede durar varias semanas o incluso meses dependiendo del objetivo que se persiga. Este proceso se dificulta cuando el teleoperador no puede observar de forma directa al vehículo y depende únicamente de los sensores y cámaras a bordo del sistema.

Uno de los principales problemas con cámaras a bordo de drones es la oscilación presente en los vídeos capturados. Este inconveniente es más complejo para los

MAVs porque las perturbaciones externas provocan mayor inestabilidad. Existen dispositivos mecánicos de estabilización de vídeo que reducen las oscilaciones en la cámara. Sin embargo, estos mecanismos implican una carga adicional al sistema y aumentan el costo de producción, gasto energético y el riesgo para las personas que se encuentren cerca en caso de accidente.

En la presente tesis se propone el desarrollo de algoritmos de estabilización de vídeo por software sin elementos mecánicos adicionales en el sistema, a ser utilizados en tiempo real durante la navegación de los UAVs. En la literatura existen pocos algoritmos de estabilización de video aplicables en tiempo real, los cuales generan falsos movimientos (movimientos fantasma) en la imagen estabilizada. El algoritmo desarrollado es capaz de obtener una imagen estable y simultáneamente mantener los movimientos reales. Se han llevado a cabo múltiples experimentos con MAVs y las métricas de evaluación utilizadas evidencian el buen desempeño del algoritmo introducido.

Abstract

Micro Aerial Vehicles (MAVs), a subset of Unmanned Aerial Vehicles (UAVs), also known as drones, are becoming popular for several applications and gaining interest due to advantages as manufacturing and maintenance cost, size and weight, energy consumption, and flight maneuverability.

Required skills for drone teleoperators being lower than for aircraft pilots, however their training process can last several weeks or months depending on the target at hands. In particular, this process is harder when teleoperators cannot observe directly the vehicle, depending only on onboard sensors and cameras.

The presence of oscillations in the captured video is a major problem with cameras on UAVs. It is even more complex for MAVs because the external disturbances increase the instability. There exists mechanical video stabilizers that reduce camera oscillations, however this mechanical device adds weight and increases the

manufacturing cost, energy consumption, size, weight, and the system becomes less safe for people.

In this thesis, we propose to develop video stabilization software algorithms, without additional mechanical elements in the system, to be applied in real-time during the UAV navigation. In the literature, there are a few video stabilization algorithms able to be applied in real-time, but most of them generate false motion (phantom movements) in the stabilized image. Our algorithm represents a good tradeoff between stable video recording and simultaneously keeping UAV real motion. Several experiments with MAVs have been performed and the employed measurements demonstrate the good performance of the introduced algorithm.

Agradecimientos

Resulta difícil agradecer todo el apoyo recibido durante estos 4 años. No obstante, quiero agradecer en primer lugar a mi director Prof. Cecilio Angulo, de quién he aprendido no solo en el campo de la investigación sino a nivel personal. Mi gratitud a los revisores de tesis, Prof. José García y Prof. Sergio Escalera y miembros del tribunal.

Adicionalmente, mi agradecimiento a todos quienes contribuyeron de forma directa o indirecta a este trabajo, entre ellos profesores, estudiantes y personal de los siguientes grupos, departamentos, centros, entre otros:

- Grupo de investigación GREC
- Proyecto PATRICIA
- Departamento ESAII de la UPC-BarcelonaTECH
- Centro de investigación CETpD
- Departamento DEEE de la Universidad de las Fuerzas Armadas - ESPE

- Centro de investigación CICTE
- Becarios de la SENESCYT como Vanessa Abad, miembros del CíBEC y otros estudiantes de postgrado.

Finalmente quiero agradecer la financiación a través de una beca del Programa "Convocatoria Abierta 2011" concedida por la Secretaría de Educación Superior, Ciencia, Tecnología e Innovación SENESCYT de la República del Ecuador.

Índice de Contenido

Índice de Figuras

Capítulo 1: Introducción

La robots móviles han adquirido protagonismo en diversos campos de aplicación por su libertad de movimiento. Estas plataformas pueden desplazarse en entornos similares a aquellos en que los seres humanos llevan a cabo sus tareas, lo que facilita su interacción. No obstante, esta característica es a su vez uno de los mayores problemas a solventarse.

Los robots móviles requieren, para la navegación, sistemas de estimación de estado que permitan conocer constantemente su pose (posición y orientación) en el entorno en que se desempeñan [1]. Con base en la tecnología utilizada para la adquisición de datos, estos sistemas pueden ser clasificados en tres categorías:

- Sistemas inerciales
- Sistemas basados en visión
- Sistemas mixtos

Los sistemas inerciales por lo general se basan en una unidad de medición inercial (IMU, por sus siglas en inglés), que permiten estimar la posición, orientación y velocidad del robot sin necesidad de referencias externas [2]. Este tipo de sistemas es de uso común en la navegación de barcos, submarinos, misiles, aeronaves o en naves espaciales, pero en años recientes su uso se ha extendido hacia dispositivos móviles, weareables y distintos tipos de robots móviles. Los sistemas inerciales permiten el control y navegación de cualquier tipo de plataforma robótica en ambientes sin obstáculos.

Los sistemas mixtos utilizan una combinación de la información visual e inercial para la estimación de la pose y la navegación en entornos interiores como es el caso de [3]. Estos sistemas requieren la integración de datos de múltiples sensores (Data fusion) y uno de sus principales problemas es la sincronización de estos datos. En exteriores, independientemente del tipo de sistema, un GNSS o sistema global de navegación por satélite, como el GPS [2], [4], proporciona información de altitud y posición complementaria.

Los sistemas basados en visión [5], [6] utilizan secuencias de imágenes capturadas por una o varias cámaras a bordo del robot. Estos sistemas proporcionan mayor información del entorno de navegación, pero su fiabilidad depende directamente de la forma de movimiento de la plataforma sobre la que se encuentra instalada la cámara. Cuando se trabaja sobre plataformas estables, como robots con ruedas (Figura 1.1) o robots tipo oruga, la cámara, ubicada generalmente en la parte superior, captura imágenes del entorno de navegación con un grado de estabilidad aceptable.

Figura 1.1: Ejemplos de plataformas robóticas. Robot con ruedas: Wifibot (Izquierda superior). Robot cuadrúpedo: Aibo ERS 7 (Derecha superior). Robot bípedo: Nao (Izquierda inferior). Robot Aéreo: AR.Drone (Derecha inferior).

En el caso de robots cuya naturaleza de locomoción es de mayor complejidad, como robots con extremidades o vehículos aéreos (Figura 1.1), la estabilidad de las imágenes capturadas está más comprometida debido a la gran variabilidad en la dinámica de movimiento del robot. Ésta dinámica también depende de la ubicación de la cámara.

1.1 Marco de trabajo

En los últimos años ha habido un creciente interés en el desarrollo de vehículos aéreos no tripulados, o UAV por sus siglas en inglés. Una clase particular de UAV, que ha ganado relevancia por sus ventajas durante vuelos en espacios cerrados y reducidos, es la de los micro vehículos aéreos o MAVs (Micro Aerial Vehicles). La tesis doctoral se enmarca en los MAVs, que por sus bajos costos de adquisición y mantenimiento, combinados con su versatilidad, tienen un gran alcance en el mercado de aplicaciones comerciales de los UAV.

El MAV en el que se centra el interés es el cuadricóptero, un vehículo aéreo con una configuración de cuatro alas rotativas. Se utiliza el AR.Drone [7], un MAV de bajo costo desarrollado por la empresa francesa Parrot de código parcialmente abierto. El AR.Drone puede ser controlado desde una PC, laptop o dispositivos móviles como smartphones o tablets. Su SDK (kit de desarrollo de software) se encuentra disponible para los sistemas operativos Windows, Linux, Mac, Android e iOS. Adicionalmente existen múltiples drivers de comunicación con el AR.Drone disponibles para ROS (Robot Operative System) que pueden ser utilizados con Python, C++ o Java.

La estabilización de vídeo en la que se enmarca la investigación es en tiempo real y sin necesidad de sincronización de la información visual con los datos inerciales del vehículo. Está orientada a ser una herramienta de apoyo a la teleoperación y navegación. En este punto es importante mencionar algunas consideraciones respecto a la ubicación y naturaleza de los dispositivos de captura y procesamiento de imágenes:

- El conjunto de imágenes que se capturan y utilizan como datos de entrada son de tipo monocular.
- El dispositivo de captura de la secuencia de fotogramas se encuentra onboard (a bordo del MAV).

- El procesamiento de la información se lleva a cabo en una estación de tierra externa desde la cual se teleopera el MAV.
- Se debe tener en cuenta los posibles problemas de comunicación entre el MAV y la estación de tierra.

1.2 Motivación

La navegación es uno de los problemas fundamentales que se presentan en aplicaciones de robótica móvil, y constituye una de las principales áreas de investigación en el marco de los MAVs [1]. La complejidad aumenta cuando los datos de entrada con los que se cuenta son solo imágenes obtenidas desde el propio vehículo, es decir a través de una cámara onboard [8].

Una alternativa para abordar el problema de navegación es el sistema de Light Detection and Ranging o LIDAR [4]. La tecnología LIDAR permite obtener modelos 3D del entorno, a partir de una nube de puntos del terreno, que pueden ser utilizados para múltiples tareas de navegación como estimación de estado, percepción del entorno e incluso en la planificación de trayectorias. Sin embargo, las principales desventajas de los LIDARs son su costo económico, peso y volumen, por lo que su aplicación sobre MAVs implica una carga adicional que se traduce en costo energético, autonomía y peso del vehículo. Además implican un riesgo para las personas que se encuentren cerca del vehículo durante el vuelo. Por tanto, ya sea desde el punto de vista económico, energético o de seguridad, los sistemas de adquisición de imágenes de entrada son una propuesta más eficiente que los LIDARs.

La compensación de los efectos de traslación y rotación, producto del movimiento de la cámara, es un proceso que puede mejorar el desempeño de los sistemas de navegación basados en visión, dependiendo de la aplicación [9]. De esta forma, el conjunto de fotogramas que se capturan será reconstruido, permitiendo

una secuencia estable de imágenes. Esta problemática se conoce como estabilización de vídeo [10].

Los algoritmos de estabilización de vídeo han sido mejorados considerablemente en los últimos años al punto de constituirse en uno de los factores estándares a corregir durante el post-procesamiento de un vídeo. Forman parte fundamental de los principales software de edición de vídeo como After Effects [11] de Adobe o Youtube Editor [12] de Google.

La primera desventaja que presenta el proceso de estabilización de vídeo es respecto al tiempo de cálculo. Esto se debe en gran medida a la etapa de estimación de la intención de movimiento que es costosa a nivel computacional y será abordada más adelante. A diferencia de ello, el costo computacional que implica la detección, descripción y matching (búsqueda de correspondencias) de puntos de interés, y de la compensación de las oscilaciones en la imagen a partir del movimiento estimado, es bajo.

Una alternativa que reduce considerablemente los costos computacionales para la compensación del movimiento de la cámara, es la utilización de modelos que relacionen los movimientos dinámicos del vehículo y su cámara onboard con las acciones de control ejecutadas por el usuario. El modelo una vez estimado, puede ser aplicado en el filtro de Kalman para la estimación de la intención de movimiento. Esta intención de movimiento debe ser diferenciada de los efectos indeseados de rotación y traslación respecto a la imagen consigna. Con el propósito de obtener una secuencia estable de imágenes, únicamente se debe compensar los efectos indeseados.

1.3 Objetivos

La presente tesis doctoral tiene como objetivo contribuir en el desarrollo de nuevas algoritmos de estabilización de vídeo, capaces de ser aplicados en tiempo real sin que su desempeño se vea comprometido, para la teleoperación de micro vehículos aéreos no tripulados de ala rotativa en vuelos internos, utilizando el mínimo número de sensores a bordo y sin necesidad de sensores externos.

El desarrollo algorítmico y metodológico que se propone para lograr este objetivo general es dividido de la siguiente forma:

- Diseño de un algoritmo robusto de estabilización de vídeo capaz de ser aplicado en tiempo real en robots de dinámica compleja.

- Estudio de los movimientos fantasma, generados por los algoritmos de estabilización de vídeo y su impacto en la teleoperación.

- Diseño de un algoritmo de estabilización de vídeo que minimice el efecto de los movimientos fantasma y no dependa del modelo del robot.

- Diseño de un algoritmo de estabilización que minimice el efecto de los movimientos fantasma con un bajo costo computacional para su aplicación en tiempo real durante la teleoperación de MAVs.

1.4 Principales contribuciones

Las principales contribuciones de la presente tesis doctoral se presentan a continuación:

1. **Movimientos fantasma:** Se ha detectado la existencia de un fenómeno no estudiado en la literatura, presente en los algoritmos de estabilización de vídeo, que se genera en la fase de suavizado del movimiento durante el proceso de estabilización de vídeo, y al que se ha denominado movimientos fantasma. El impacto de este fenómeno es considerable durante la teleoperación de MAVs.

2. **Tiempo real:** El tiempo de procesamiento de la estabilización de vídeo ha sido minimizado mediante la reducción del número de iteraciones del algoritmo RANSAC y la utilización de una combinación de transformaciones geométricas para compensar esta disminución evitando comprometer la robustez del algoritmo.

3. **Modelado del MAV:** Se ha llevado a cabo el modelado del AR.Drone 1.0 y 2.0 de Parrot basándose en 3 conjuntos de datos: (a) datos inerciales provenientes de la IMU del AR.Drone, (b) acciones de control generadas por el usuario durante la captura de datos y (c) la información visual capturada por la cámara a bordo de la plataforma aérea.

4. **Dos algoritmos de estabilización de vídeo:** Se han propuesto dos enfoques capaces de minimizar el efecto del fenómeno de los movimientos fantasma:

a. El primero, independiente del modelo de la plataforma robótica, requiere un conjunto de fotogramas previos generando un retardo adicional en el proceso de estabilización de vídeo
b. El segundo que depende del modelado previo de la plataforma sobre la cual se encuentra el dispositivo de captura pero que reduce la dependencia de fotogramas previos únicamente al último fotograma.

Ambas propuestas están orientadas a minimizar el tiempo de entrenamiento de los teleoperadores.

1.5 Estimación de movimiento y movimientos fantasma

Una parte fundamental en la estabilización de vídeo es la estimación del movimiento que depende de la pose (posición y rotación) global de la cámara para cada fotograma. Entre fotogramas consecutivos es imprescindible seleccionar el modelo apropiado que describa el movimiento, generado por la cámara, entre los instantes de tiempo de captura de las imágenes. Muchos de los modelos presentes en la literatura, se basan en los modelos 2D: traslacional, semejanza, afín y proyectivo. Estos modelos 2D son una proyección truncada del movimiento tridimensional de la cámara. Una segunda alternativa son los modelos 3D, cuya exactitud es mayor al contener todos los grados de libertad del movimiento tridimensional. Estos modelos son capaces de eliminar distorsiones en la imagen [10]. La estimación de la rotación 3D a partir de la deformación de los puntos en la imagen no es un problema complicado. A diferencia de ello, la estimación de la traslación depende de datos adicionales que difícilmente pueden obtenerse usando únicamente imágenes monoculares. Es necesario conocer la profundidad de cada píxel en la imagen. Algunas propuestas usan cámaras de estéreo visión (dos cámaras), cámaras de profundidad como la kinect o cámaras plenóptica como en

[13]. Muchos enfoques basados en modelos 3D ignoran los parámetros de traslación y solo consideran la rotación.

Los métodos de estimación de movimiento basados exclusivamente en la información visual, son costosos a nivel computacional y son propensos a errores. Otros enfoques [14]–[16] basan los algoritmos de estabilización de vídeo en información inercial obtenida usando sensores como giroscopios y acelerómetros, presentes en dispositivos móviles modernos como smartphones o tablets. Con la incorporación de la información inercial, el tiempo de cómputo se reduce considerablemente, lo que permite su aplicación en tiempo real.

Según el modelo adoptado para la estabilización de vídeo, los métodos pueden clasificarse en 3 grupos:

- Estimación de movimiento basada en modelos 3D.
- Estimación de movimiento basada en modelos 2D.
- Estimación de movimiento basada en combinación de modelos.

1.5.1 Estimación de movimiento basada en modelos 3D

La estimación de movimiento basada en métodos 3D requiere la estructura tridimensional completa para la estabilización, es decir la pose. Esta estructura se estima mediante el uso de algoritmos SFM (estructura a partir de movimiento) [17]–[19], o sensores de profundidad [13]. La trayectoria oscilatoria de la cámara 3D se suaviza para la reconstrucción del vídeo estabilizado a través de esta trayectoria. Un método ampliamente usado en algoritmos de estabilización de vídeo es [10] que propone, basado en [20], la introducción de deformaciones que preserven el contenido. En [20] los autores plantean la manipulación de la apariencia de los fotogramas de la forma más rígida posible. [21] presenta una extensión de [10] mediante restricciones basadas en el plano. Los aportes de [13] son respecto al suavizado de la trayectoria para la reducción de la aceleración en los parámetros de

movimiento. Pese a que los resultados obtenidos utilizando la reconstrucción 3D son de mayor calidad, este proceso continúa siendo complicado y costoso respecto al tiempo de cómputo. Adicionalmente algunas contribuciones importantes respecto a la reconstrucción de vídeo 3D se pueden encontrar en [22]–[29].

1.5.2 Estimación de movimiento basada en modelos 2D

Los métodos 2D estiman las transformaciones entre 2 fotogramas consecutivos. La trayectoria de la cámara se obtiene mediante la concatenación de las transformaciones geométricas, ya sean estas de tipo afín, proyectivo u otras. A continuación, esta trayectoria se suaviza mediante filtros pasa-bajos [15], [30]–[32], que reducen las oscilaciones de la cámara, o empleando planificadores de trayectorias [33], [34] y se utiliza en la compensación del movimiento. Es importante mencionar que tanto en modelos 3D como en modelos 2D, el suavizado de la trayectoria, que tiene como objetivo la aproximación de los movimientos del dispositivo de captura, es conocido como estimación de la intención de movimiento en el marco de algoritmos de estabilización de vídeo.

Algunos de estos métodos funcionan únicamente bajo condiciones donde la deformación de la imagen, producto de la rotación, es considerable [35]. Trabajos basados en trayectorias envolventes de la cámara [36] y flujo óptico suavizado espacialmente [37], obtienen resultados comparables con técnicas 3D. Otra alternativa consiste en el uso de ajustes polinomiales para la estimación de la trayectoria de la cámara como es el caso de [38]. En [33], la trayectoria de la cámara se separa en segmentos y se ajusta a cada uno de los movimientos suavizados. Un filtro Gaussiano, aplicado en ventanas, se utiliza para el suavizado de la trayectoria del movimiento de la cámara en [39], [40], y se basa en los modelos traslacional y afín. [41] muestra un mejor desempeño del suavizado del movimiento global que el local, usando una función costo obtenida mediante una combinación de diferencias de primer, segundo y tercer orden del movimiento de la cámara

medido con norma L1. En [10], [14] se consigue suavizar la rotación con distancias geodésicas y un filtro pasa-bajos.

Uno de los algoritmos más populares utilizado en la estabilización de vídeo es el L1-Optimal [41], [42] que representa el movimiento de la cámara mediante la combinación de movimientos constantes, lineales y parabólicos. Esta técnica está integrada en el Youtube Editor de Google. Adicionalmente, el algoritmo L1-Optimal se mejoró en [34] aplicando reglas cinematográfica como: trayectorias constantes que representan cámaras estáticas, trayectorias con velocidad constante para simular el efecto de las cámaras panorámicas o planos de plataformas rodantes [43], y trayectorias con aceleración constante para transiciones entre cámaras estáticas y panorámicas.

La fusión de datos inerciales y visuales se utiliza ampliamente en robótica y en dispositivos móviles, para incrementar la exactitud y fiabilidad del seguimiento de movimiento [44], [45]. No obstante, la sincronización es un punto fundamental debido a que si los parámetros de movimiento del vídeo no están en fase, respecto a los sensores inerciales, los problemas de reconstrucción del vídeo podrían ser críticos.

Todos estos algoritmos se aplican en post-producción u offline (fuera de línea), es decir, se suaviza la secuencia de movimiento de la cámara luego de capturar la secuencia de vídeo completa. No obstante, para aplicaciones en tiempo real existen algunos trabajos, restringidos a modelos de movimiento 2D. En [46] se propone un filtro IIR para suavizar el movimiento en tiempo real, usando un modelo de movimiento de traslación 2D. En [47] los autores usan el filtro de Kalman para el seguimiento de puntos característicos y el modelado de parámetros de movimiento intencional (el modelo usado es la traslación 2D) . Este método basado en el filtro de Kalman se extiende al modelo de movimiento afín 2D en [48], consiguiendo un mejor desempeño. En múltiples trabajos posteriores, se usa ampliamente el enfoque de Kalman, un ejemplo de ello es [49]. En [50] se utiliza el filtro de Kalman

para la obtención, en un solo paso, de las restricciones para un modelo de movimiento traslacional 2D.

1.5.3 Estimación de movimiento basada en combinación de modelos

Las técnicas de estabilización 2D y 3D pueden combinarse para suavizar directamente las trayectorias de los puntos característicos. El enfoque de [51] se basa en la intersección de líneas epipolares calculadas entre pares de fotogramas (lo que se conoce como epipolar transfer) para evitar la inestabilidad del algoritmo en la reconstrucción 3D. Por su parte, en [52] se representa cada trayectoria como un curva de Bézier y se suaviza con optimización espacio-temporal. En [53] los autores proponen la técnica poda de morfología matemática para escoger trayectorias características robustas. En [11] las trayectorias bases del subespacio [54] se extraen a partir de las características seguidas a lo largo de al menos cincuenta fotogramas. Este método obtiene una calidad similar a los enfoques basados en modelos 3D sin la necesidad de reconstruir extensas trayectorias de características, y se encuentra disponible en el software comercial Adobe After Effects. Adicionalmente, se puede trabajar con vídeos estereoscópicos con la extensión publicada en [55]. Ya sea que se trabaje con dispositivos móviles, robots cuadrúpedos, micro vehículos aéreos o cualquier otro sistema sobre el cual se ubica la cámara, las trayectorias de movimiento de este dispositivo de captura de imágenes se extiende a lo largo de una variedad no lineal [56], [57]. Es posible aproximar esta variedad localmente mediante subespacios lineales, tomando en cuenta que la matriz de trayectoria no debe tener un rango mayor a 9 [54]. La relación entre los puntos característicos se preserva mediante la restricción del subespacio.

Otro problema de deformaciones indeseadas es el efecto rolling sutter que consiste en la curvatura de líneas rectas en la imagen capturada, causado por la

estructura paralela de lectura de los sensores CMOS. La cámara no captura la imagen de la escena completa en un solo instante de tiempo sino que realiza un barrido, ya sea a lo largo de las filas o columnas del fotograma. El efecto de rolling shutter [58] es rectificado utilizando enfoques basados en combinación de modelos como [36], [37]. También se pueden emplear modelos de traslación [59], [60] y modelos de rotación 3D [16], [61]. En [42] proponen un modelo proyectivo mixto que no requiere calibración. Las GPUs pueden compensar el efecto de rolling sutter en tiempo real utilizando tecnología de mapeado de texturas [62]. En [14], [15] se usan hardware dedicados.

1.5.4 Movimientos fantasma

Muchas de las técnicas citadas tienen un buen desempeño como algoritmo de estabilización de vídeo, pero existe un reto adicional en aplicaciones en tiempo real como la teleoperación de MAVs. Este problema, no estudiado hasta el momento en la literatura, se introduce en la presente tesis doctoral y lo hemos denominado "movimiento fantasma".

Los movimientos fantasma, se definen como falsos desplazamientos generados por el algoritmo de estabilización de vídeo en los parámetros de escala y/o traslación en los ejes. Estos movimientos fantasma se producen por la compensación de los movimientos indeseados eliminados en el proceso de suavizado. Se pueden eliminar erróneamente los movimientos reales del vehículo y/o se pueden introducir retardos. Ambos efectos dan lugar a movimientos fantasma. Este fenómeno representa un importante problema cuando se está teleoperando el MAV, y sus efectos en algoritmos del estado del arte se presentarán en la sección de resultados y discusión de los Capítulos 3 y 4.

Las técnicas estándar de estabilización de vídeo consiguen buenos resultados eliminando movimientos indeseados en imágenes capturadas con dispositivos

móviles y sistemas complejos, sin embargo generan movimientos fantasma. Este problema no es significativo para aplicaciones de post-procesamiento de vídeo. En sistemas teleoperados, los movimientos fantasma representan un problema peligroso. Una primera solución se propone es [63], basada en un filtro pasa-bajos y el uso de la acción de control como compuerta lógica con histéresis, la cual se explicará en el Capítulo 3, así como la propuesta [64] basada en el modelo del MAV que incluye la acción de control del teleoperador, y que se explicará en el Capítulo 4.

Publicamos este fenómeno por primera vez en nuestro artículo del JIVP [63] en 2014.

1.6 Estructura de la tesis

El resto de la memoria de tesis se divide de la siguiente forma:

El segundo capítulo está destinado a explicar procesos fundamentales utilizados por los algoritmos de estabilización de vídeo, incluyendo la estimación de movimiento. Adicionalmente se proponen distintas alternativas a utilizarse como fotograma consigna y respecto al cual se lleve a cabo la estabilización de vídeo.

En el tercer capítulo se propone un enfoque basado en la combinación de transformaciones geométricas para mantener la robustez en la estimación de movimiento usando un menor número de iteraciones en el algoritmo RANSAC. Se introduce los movimientos fantasma y se explica una primera propuesta basada en la combinación de un filtro pasa-bajos de segundo grado usado como técnica de suavizado de los parámetros de movimiento, optimizada para ser aplicada en tiempo real, y la acción de control introducida en el algoritmo como una histéresis para la minimización del efecto de los movimientos fantasma.

En el cuarto capítulo se presenta una segunda propuesta para la estabilización de vídeo en tiempo real, basada en el filtro de Kalman y el modelo del micro vehículo aéreo. Se explica cómo se ha llevado a cabo el modelado del MAV utilizado en la experimentación, modelo que incluye adicionalmente la acción de control realizada por el teleoperador durante la etapa de adquisición de datos fuera de línea. Finalmente, se compara los resultados obtenidos con varios de los algoritmos de estabilización de vídeo que mejor se desempeñan en esta tarea, en distintos escenarios y bajo distintas condiciones.

Las conclusiones e impacto, líneas futuras, financiamiento y publicaciones realizadas durante la investigación doctoral se presentan en el quinto y último capítulo.

Capítulo 2: Estabilización de vídeo para escenas rígidas

En el Capítulo 2 se presenta una metodología de corrección de los efectos deformatorios que la navegación del robot produce en el flujo de imágenes. Para ello, el proceso de compensación se inicia con la detección y comparación de un conjunto de puntos de interés entre fotogramas consecutivos. Su ubicación permite determinar una matriz de homografía entre fotogramas. Aplicando la inversa de dicha homografía, se obtiene la imagen reconstruida.

Se ha estructurado al capítulo de la siguiente forma: En la Sección 2.1, se explica brevemente el procedimiento seleccionado para la detección, descripción y búsqueda de correspondencia de puntos de interés, que son utilizados para

comparar dos fotogramas entre sí. La Sección 2.2 hace referencia al estudio de la homografía como modelo de compensación, cuya estimación se consigue a partir de los puntos de interés. La Sección 2.3 está destinada al comportamiento, a lo largo de todos los fotogramas en la secuencia de vídeo, de los parámetros de movimiento extraídos del modelo. La compensación del movimiento es explicada en la Sección 2.4. Finalmente los resultados y conclusiones se presentan en las Secciones 2.5 y 2.6, respectivamente.

2.1 Puntos de interés

La estimación de movimiento tiene como objetivo determinar los parámetros de movimiento entre el fotograma actual y el fotograma consigna. En este marco, los puntos de interés buscan representar las principales características de las regiones de la imagen mediante un pequeño conjunto de información [65].

Usualmente, los puntos de interés se utilizan para el proceso de comparación de imágenes, tarea en la cual se diferencian 3 pasos:

- Detección de puntos de interés.
- Descripción de puntos de interés.
- Matching o búsqueda de correspondencia de puntos de interés.

Existe una gran variedad de algoritmos que permiten llevar a cabo la detección y descripción de puntos de interés con un bajo costo computacional, entre los cuales se puede mencionar el detector de Harris [66] que, a pesar de presentar robustez ante cambios de intensidad, presenta problemas cuando se trabaja con transformaciones distintas entre imágenes.

Binary Robust Invariant Scalable Keypoints (BRISK) [67], Fast Retina Keypoint (FREAK) [68], Oriented FAST and Rotated BRIEF (ORB) [69], Scale Invariant Feature Transform [70], [71] (SIFT), y Speeded Up Robust Feature (SURF) [72], [73] son cinco algoritmos ampliamente utilizados en la solución de problemas de visión por computador. El estudio comparativo de [73] muestra que SURF tiene un menor costo computacional que SIFT. Si bien algoritmos como ORB son más rápidos que SURF, este último está disponible en un mayor número de librerías, lo que implica facilidad en la extrapolación hacia otros leguajes de programación, softwares y sistemas operativos. Adicionalmente SURF es más robusto y confiable cuando se trabaja con imágenes capturadas con MAVs, como se evidencia en la Tabla 2.1. El costo computacional de SURF es el doble que el de FAST, pero en ningún vídeo genera error en la estimación de movimiento. El proceso de estimación de movimiento se explicará más adelante.

Tabla 2.1. Algoritmos de detección y descripción de puntos de interés (Matlab)

Método	**Métrica**	**Vídeo 1**	**Vídeo 2**	**Vídeo 3**	**Vídeo 4**	**Vídeo 5**
SURF	Tiempo	0.103	0.110	0.078	0.109	0.125
FAST		**0.044**	**0.056**	**0.038**	**0.056**	**0.061**
BRISK		0.309	0.316	0.256	0.316	0.332
SURF	Pares	**153**	**160**	**212**	**116**	**131**
FAST		120	91	176	77	108
BRISK		18	12	15	8	11
SURF	Error	**0**	**0**	**0**	**0**	**0**
FAST		3	5	4	7	5
BRISK		**0**	1	**0**	2	**0**

El algoritmo SURF (por sus siglas en inglés Speeded Up Robust Features [72]), al igual que SIFT, es un algoritmo de detección, descripción y matching de puntos de interés, con un costo computacional considerablemente menor.

A continuación, por completitud, se explicará brevemente como SURF lleva a cabo los 3 procesos respecto a los puntos de interés.

2.1.1 Detección de puntos de interés

Para la detección de los puntos de interés, el algoritmo SURF utiliza una aproximación básica de la matriz Hessiana, debido al buen desempeño que presenta esta matriz con respecto a su exactitud. Para ello un concepto que resulta de gran utilidad es el de imagen integral.

La imagen integral es otra forma de representar el conjunto de píxeles de una imagen, tal que, el valor del punto P de coordenadas (x, y) representa la sumatoria de todos los píxeles de la imagen que corresponden a la región rectangular existente entre dicho punto P y el origen de coordenadas, es decir:

$$I_{\Sigma}(P) = \sum_{i=0}^{i \leq x} \sum_{j=0}^{j \leq y} I(x, y) \tag{2.1}$$

donde, $I_{\Sigma}(X)$ es la entrada de la imagen integral, y $(x, y)^T$ es la localización del punto P en la imagen integral.

Una vez que se ha determinado la imagen integral, los puntos o regiones de interés pueden ser detectados mediante un simple análisis del determinante de la matriz Hessiana, donde será máximo. Este determinante puede ser calculado de forma eficiente a nivel computacional mediante la aproximación:

$$Det(H_{aprox}) = D_{xx}D_{yy} - (0.9D_{xy})^2 \quad (2.2)$$

donde D_{xx}, D_{yy}, D_{xy} son aproximaciones a lo largo de las tres direcciones.

Utilizando la matriz Hessiana y una función denominada Espacio-escala, la localización exacta de los puntos de interés (Figura 2.1) puede ser dividida en 3 partes:

- Primeramente se desestima los valores obtenidos del determinante de la matriz Hessiana que se encuentren por debajo de un umbral establecido. Este umbral es adaptable y depende específicamente de la aplicación en la cual se está llevando a cabo el proceso de localización de puntos de interés (a mayor valor de umbral, menor número de puntos detectados).
- A continuación se realiza la selección del conjunto de puntos candidatos. Cada píxel es comparado con sus 26 vecinos en las 3 dimensiones posibles. Se dice que un píxel es máximo si es mayor que todos los píxeles que lo envuelven.
- Finalmente se localiza en Espacio-escala el píxel que corresponde al punto de interés detectado.

En la Figura 2.1 se muestra un ejemplo donde se detectaron 8 puntos característicos sobre una imagen capturada por el robot Aibo ERS 7.

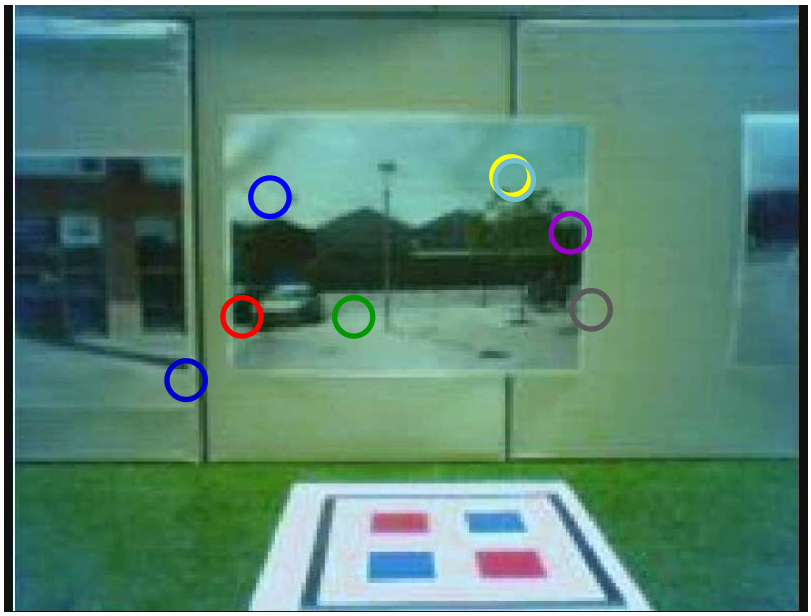

Figura 2.1. Detección de puntos de interés siguiendo el algoritmo SURF.

2.1.2 Descripción de puntos de interés

Luego de calcular los puntos de interés, SURF determina la distribución de la intensidad de los píxeles que componen las regiones cercanas a cada uno de los puntos de interés que han sido detectados. Para ello, y con el objetivo de incrementar la robustez y disminuir el tiempo de cálculo computacional respecto al descriptor SIFT, utiliza los Wavelets de Haar.

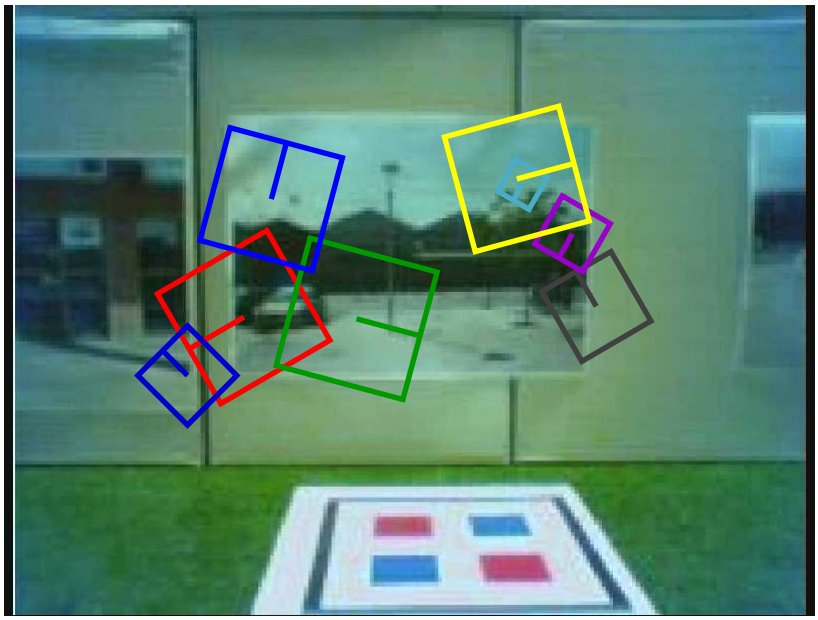

Figura 2.2. Descripción de puntos de interés siguiendo el algoritmo SURF.

Los Wavelets de Haar permiten determinar el gradiente de forma rápida en las 2 direcciones del espacio bidimensional de la imagen. La extracción del descriptor se lleva a cabo en dos partes:

- Se identifica una orientación reproducible bajo condiciones variables, para cada punto de interés, con el objetivo de conseguir invariancia en la rotación.

- Se construye una ventana que sea dependiente de la escala, de la cual se extrae un vector 64 dimensional (Figura 2.2). Con el objetivo de mantener su comportamiento invariante a la escala, todo cómputo estará basado en una medida relativa a la escala detectada.

De esta forma se obtiene un vector 64-dimensional que contiene las características diferenciables de cada punto de interés.

2.1.3 Búsqueda de correspondencias entre puntos de interés

Una vez conseguidos los vectores descriptores, el proceso de búsqueda de correspondencias o matching (Figura 2.3) de puntos de interés entre dos imágenes consiste en la selección de aquellos pares de puntos de interés que contengan la menor diferencia vectorial entre sus descriptores 64-dimensionales.

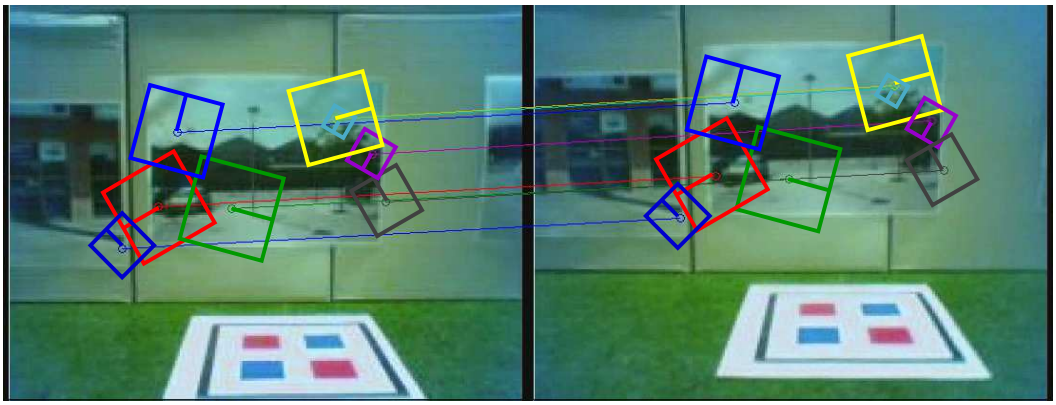

Figura 2.3. Búsqueda de correspondencias entre puntos de interés.

Para una rápida indexación durante la fase de búsqueda de correspondencias, se incluye el signo del Laplaciano, es decir, la traza de la matriz Hessiana para el punto de interés subyacente. El signo del Laplaciano permite distinguir la detección de regiones brillantes contrastadas con un fondo oscuro, de la situación contraria. Durante el matching de puntos de interés, sólo se comparan características que tengan el mismo tipo de contraste, permitiendo aumentar considerablemente la velocidad de búsqueda sin que se vea afectado el rendimiento del descriptor.

Una vez que las parejas de puntos que correlacionan dos fotogramas consecutivos han sido computados, se cuenta con la información necesaria para determinar la relación geométrica entre las dos imágenes capturadas.

2.2 Transformación geométrica

Una transformación proyectiva [35], [74] es un mapeo invertible de una imagen bidimensional hacia otra imagen de igual forma bidimensional, tal que tres puntos x_1, x_2 y x_3 pertenecen a una misma línea sí y solo sí sus proyecciones x_1', x_2' y x_3' también pertenecen a la misma línea. Es debido a esta propiedad, que la transformación proyectiva también se suele denominar colineación u homografía.

Una propiedad adicional que debe cumplir una proyectividad en un plano es el hecho de poder ser representada a través de una matriz de transformación H no singular, tal que la relación entre puntos de la imagen original y los pertenecientes a la imagen transformada, sea una aplicación lineal.

Por tanto, a partir de ahora el problema consiste en estimar la matriz de transformación que relaciona el mayor número de pares de puntos en correspondencia entre dos imágenes.

2.2.1 Solución exacta: 4 pares de puntos en correspondencia

La estimación de la matriz de homografía (determinada por la geometría proyectiva), parte del conjunto de puntos en correspondencia específico X_i y X_i', los mismos que se encuentran relacionados por esta matriz de homografía H bajo la siguiente expresión matemática:

$$X_i' = HX_i \tag{2.3}$$

donde H es una matriz cuadrada de 9 parámetros (3x3), definidos salvo un factor de escala, es decir que se tiene un conjunto de 8 incógnitas por resolver.

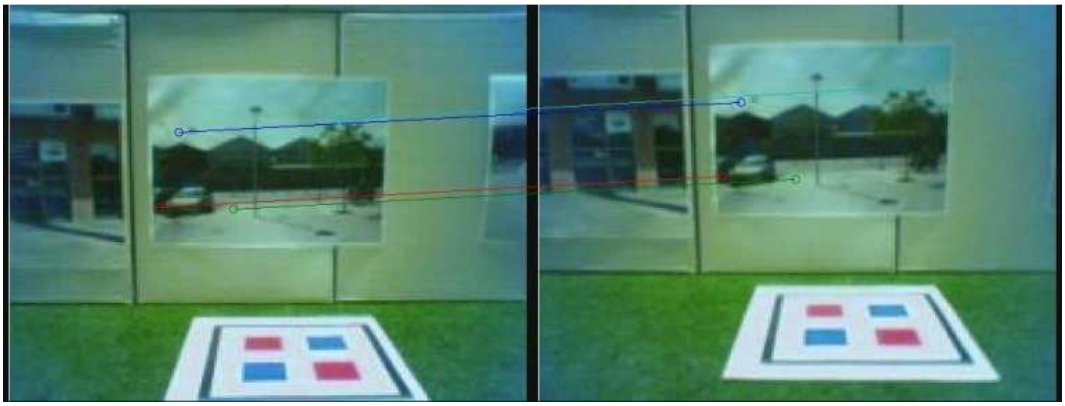

Figura 2.4 Solución exacta: 4 pares de puntos en correspondencia.

Cada par de correspondencia de puntos de interés entre las imágenes permite obtener 3 ecuaciones, de las cuales 2 son linealmente independientes. Tomando en consideración que cada par de correspondencias de puntos establece dos restricciones sobre la matriz H, el número de pares de puntos de correspondencias necesarios para obtener una solución única es 4 (Figura 2.4). Para resolver el sistema, con estos 4 pares de correspondencias, podemos utilizar el algoritmo de transformación lineal directa.

El algoritmo de transformación lineal directa o DLT es un método lineal que permite determinar la matriz de transformación H de forma simple, a partir de un conjunto de 4 correspondencias. La relación entre los puntos de las imágenes está definida en coordenadas homogéneas, por lo cual, puede interpretarse como una relación de proporcionalidad directa entre los miembros de la ecuación. Es decir,

tienen el mismo vector unitario pero distintas constantes de proporcionalidad. Por lo tanto, la relación puede ser redefinida de la siguiente forma:

$$X_i' \times HX_i = 0 \tag{2.4}$$

donde

$$H = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{bmatrix} \tag{2.5}$$

$$X_i = [x_i \quad y_i \quad w_i] \tag{2.6}$$

$$X_i' = [x_i' \quad y_i' \quad w_i'] \tag{2.7}$$

multiplicando ahora este vector X_i' con el producto obtenido entre la matriz H y el vector X_i, se da origen a la matriz:

$$X_i' \times HX_i = \begin{bmatrix} y_i'[H_{31} \quad H_{32} \quad H_{33}]^T X_i - w_i'[H_{21} \quad H_{22} \quad H_{23}]^T X_i \\ w_i'[H_{11} \quad H_{12} \quad H_{13}]^T X_i - x_i'[H_{31} \quad H_{32} \quad H_{33}]^T X_i \\ x_i'[H_{21} \quad H_{22} \quad H_{23}]^T X_i - y_i'[H_{11} \quad H_{12} \quad H_{13}]^T X_i \end{bmatrix} \tag{2.8}$$

Desarrollando las expresiones matemáticas en la matriz producto, e igualando dicha matriz a cero se obtiene un conjunto de tres ecuaciones que contiene los parámetros de H. La ecuación matricial puede expresarse de la siguiente forma:

$$\begin{bmatrix} 0^T & -w_i{X_i}^T & y_i{X_i}^T \\ w_i{X_i}^T & 0^T & -x_i{X_i}^T \\ -y_i{X_i}^T & x_i{X_i}^T & 0^T \end{bmatrix} \begin{bmatrix} H_{11} \\ H_{12} \\ H_{13} \\ H_{21} \\ H_{22} \\ H_{23} \\ H_{31} \\ H_{32} \\ H_{33} \end{bmatrix} = 0 \quad (2.9)$$

Estas ecuaciones son de la forma:

$$A_i h = 0 \quad (2.10)$$

donde A_i es una matriz de 3x9 elementos y h es un vector columna de 9 elementos constituido por los parámetros de la matriz de trasformación proyectiva H.

El sistema $A_i h = 0$ se encuentra constituido por tres ecuaciones, sin embargo, solo dos de ellas son linealmente independientes. Esto se debe a que la tercera ecuación se puede obtener de forma trivial, como combinación lineal de las otras dos ecuaciones.

En consecuencia, cada punto al que corresponde el sistema analizado sólo puede proporcionar dos ecuaciones en las entradas restrictivas para la estimación de la matriz H. El número mínimo de pares de correspondencias de puntos de interés requeridos para el cálculo de la matriz de transformación H de solución única es cuatro (cada uno aporta al sistema $A_i h = 0$ con 2 ecuaciones linealmente independientes, es decir un A_i de 2x9 elementos).

Con el objetivo de definir (x, y) como coordenadas medidas directamente sobre las imágenes, tanto para X_i como para ${X_i}'$, se iguala $w_i = {w_i}' = 1$. En caso de que w_i o ${w_i}'$ no sean 1, habría que realizar el cálculo pertinente para pasar de

coordenadas en la imagen a coordenadas de los puntos X_i o X_i'. Usando las ecuaciones de los cuatro pares de puntos, y reemplazando $w_i = w_i' = 1$ en el sistema tenemos:

$$\begin{bmatrix} 0 & 0 & 0 & -x_1 & -y_1 & -1 & y_1'x_1 & y_1'y_1 & y_1' \\ 0 & 0 & 0 & -x_2 & -y_2 & -1 & y_2'x_2 & y_2'y_2 & y_2' \\ 0 & 0 & 0 & -x_3 & -y_3 & -1 & y_3'x_3 & y_3'y_3 & y_3' \\ 0 & 0 & 0 & -x_4 & -y_4 & -1 & y_4'x_4 & y_4'y_4 & y_4' \\ x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1'x_1 & -x_1'y_1 & -x_1' \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2'x_2 & -x_2'y_2 & -x_2' \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -x_3'x_3 & -x_3'y_3 & -x_3' \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -x_4'x_4 & -x_4'y_4 & -x_4' \end{bmatrix} \begin{bmatrix} H_{11} \\ H_{12} \\ H_{13} \\ H_{21} \\ H_{22} \\ H_{23} \\ H_{31} \\ H_{32} \\ H_{33} \end{bmatrix} = 0 \qquad (2.11)$$

Una vez desarrollada la ecuación matricial, se obtiene un sistema de ecuaciones lineales de solución única. Esto se debe a que cada par de correspondencia ha dado origen a dos ecuaciones linealmente independientes y, como podemos recordar, la matriz de transformación proyectiva H es de rango 8, es decir 8 incógnitas.

El vector solución del sistema de 8 ecuaciones con 8 incógnitas que se ha generado, está dado por el vector del núcleo de la aplicación lineal definida por la matriz A. Este vector puede ser determinado, salvo factor de escala. Con el objetivo de fijar el valor del mencionado factor de escala, para llevar a cabo el cálculo del vector solución del sistema de ecuaciones, se define como condición la norma de h igualada a 1, es decir:

$$\|h\| = 1 \qquad (2.12)$$

2.2.2 Solución sobredeterminada

En el caso ideal, cuando no existe ruido en las imágenes de entrada, la matriz H estimada a partir de 4 pares de correspondencias es una solución fiable de la trasformación proyectiva que relaciona las imágenes entre sí.

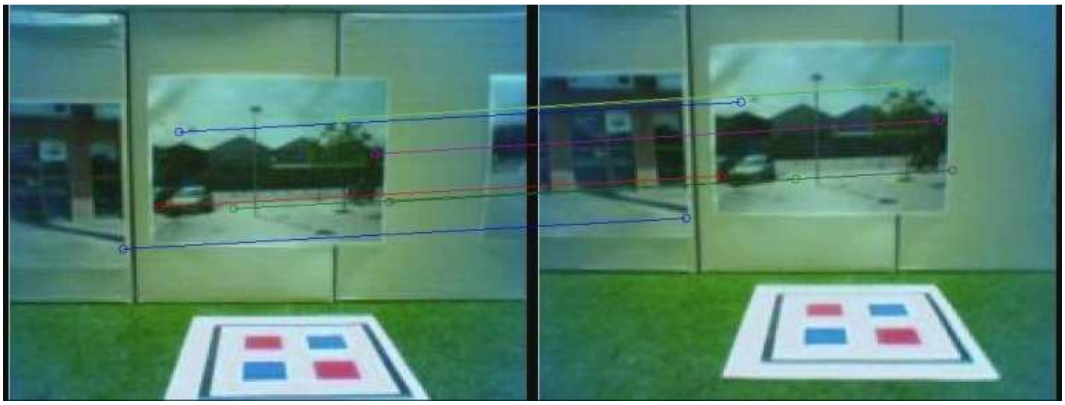

Figura 2.5. Solución sobredeterminada: más de 4 pares de puntos en correspondencia.

Sin embargo, trabajando con un conjunto real de imágenes, resulta conveniente disponer de más de cuatro pares de correspondencia en el sistema de ecuaciones, es decir que este método de estimación está sobredeterminado (Figura 2.5).

Si los puntos de entrada son exactos, matemáticamente se demuestra que el rango de la matriz A sigue siendo 8 a pesar de la presencia de un número mayor de ecuaciones lineales. Esto se debe a que en el caso de puntos exactos las ecuaciones que corresponden a los mismos son linealmente dependientes. Sin embargo, al igual que en la solución basada en 8 puntos, en el caso sobredeterminado existe un ruido en los valores de entrada y el rango de A podría ser superior, como consecuencia el sistema de ecuaciones únicamente tendría solución cuando $h = 0$.

Esta problemática se resuelve buscando, no una solución exacta, sino una aproximada, donde la función costo se minimice para el vector solución h. Por otra parte, la norma de h seguirá definiéndose como 1 para el factor de escala.

De esta forma, el problema se reduce a la minimización del cociente respecto a h:

$$\frac{\|Ah\|}{\|h\|} \tag{2.13}$$

cuya solución es el vector singular de la matriz A asociado al menor valor singular de la misma.

2.2.3 Modelos de transformación geométrica

Una vez que el conjunto de puntos de interés ha sido obtenido, los parámetros de movimiento, entre el fotograma actual y el fotograma consigna, pueden ser estimados.

El movimiento existente entre dos fotogramas específicos se puede expresar matemáticamente mediante la transformación geométrica [35], [74] que relaciona los puntos de un fotograma con sus correspondencias en el segundo fotograma,

$$X_t = HX_{t-1} \tag{2.14}$$

donde X_t es el conjunto de puntos de interés que corresponden a la imagen consigna, H es la matriz de transformación y X_{t-1} es el conjunto de puntos de interés de la imagen a compensarse.

Ésta transformación geométrica posee un modelo paramétrico de movimiento distinto, dependiendo de qué tipo de transformación se utilice, y en el que H se basa. Los 3 modelos comunes son:

Modelo de traslación

Es el más simple de los 3 modelos que se analizarán. Este modelo hace referencia a los movimientos de la imagen cuando únicamente existe traslación del dispositivo de captura en un plano paralelo al plano de la imagen (modelo geométrico de cámara pinhole).

$$X_t = X_{t-1} + \begin{bmatrix} t_y \\ t_x \end{bmatrix} \tag{2.15}$$

$$H = \begin{bmatrix} 1 & 0 & t_y \\ 0 & 1 & t_x \\ 0 & 0 & 1 \end{bmatrix} \tag{2.16}$$

Modelo afín

En el modelo o transformación afín, existen 4 parámetros: dos traslaciones en el plano paralelo al de la imagen que se describe en el modelo de traslación, una rotación en la dirección del eje roll y la escala que es proporcional al desplazamiento en la dirección del eje roll.

$$X_t = s\begin{bmatrix} cos(\emptyset) & -sin(\emptyset) \\ sin(\emptyset) & cos(\emptyset) \end{bmatrix} X_{t-1} + \begin{bmatrix} t_y \\ t_x \end{bmatrix} \quad (2.17)$$

$$H = \begin{bmatrix} s\,cos(\emptyset) & -s\,sin(\emptyset) & t_y \\ s\,sin(\emptyset) & s\,cos(\emptyset) & t_x \\ 0 & 0 & 1 \end{bmatrix} \quad (2.18)$$

Modelo proyectivo

El modelo proyectivo es el modelo completo de movimiento, en el cual se encuentran expresados matemáticamente las 3 rotaciones y traslaciones posibles. Su matriz de transformación H posee 8 parámetros linealmente independientes.

$$H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{32} \\ h_{31} & h_{32} & 1 \end{bmatrix} \quad (2.19)$$

Modelo seleccionado

Dado que el movimiento indeseado y vibraciones parásitas en la imagen únicamente se consideran significativas alrededor del eje roll, tanto en el caso de cámaras digitales como dispositivos monoculares de visión a bordo de robots de dinámica compleja, se utiliza el modelo afín para la estimación de los parámetros de movimiento.

Cabe señalar dos ventajas adicionales que se obtienen utilizando el modelo afín:

- La primera es referente al tiempo de cómputo, el cual es significativamente menor, dado que únicamente depende de 4 parámetros independientes, a diferencia del modelo proyectivo que contiene 8 parámetros independientes.

- Una segunda ventaja que se obtiene es la facilidad en la extracción directa de los cuatro parámetros de movimiento correspondientes a escala, rotación roll y traslaciones a través del plano xy.

2.3 Estimación robusta del movimiento acumulado

Se divide en 2 partes:

- Estimación robusta del movimiento en la secuencia de fotogramas completa basada en RANSAC.
- Estimación del movimiento acumulado basado en la transformación afín.

2.3.1. Estimación robusta del movimiento basada en RANSAC

Partiendo de la definición previa que establece como robustez a la asignación correcta de pares de puntos de interés en correspondencia, y luego de realizar la revisión de la literatura relativa al tema, se vuelve evidente que RANSAC [75]–[77] (Algoritmo 2.1) es una técnica iterativa fiable para la desestimación de pares de puntos de interés que no se ajustan al modelo matemático que los relaciona, en este caso al modelo afín.

Sin embargo, es importante hacer referencia a 2 aspectos en RANSAC:

- El modelo al cual debe ajustarse considera el modelo entre fotogramas consecutivos.
- La función costo que se utiliza es la diferencia absoluta de intensidad, para cada píxel, entre el fotograma compensado y el de referencia.

$$f_{cost} = |HI_{t-1} - I_t| \tag{2.20}$$

Por lo tanto el nuevo H_{new} que minimiza la función costo es:

$$H_{new} = arg\,min_A \sum_i |HI_{t-1} - I_t| \quad (2.21)$$

Algoritmo 2.1 RANSAC

1. Seleccionar aleatoriamente el mínimo número de puntos requeridos para determinar los parámetros del modelo.

2. Calcular los parámetros del modelo.

3. Determinar cuántos puntos del conjunto se ajustan al modelo con una tolerancia predefinida.

4. **Si** la fracción de números de inliers del total de números de puntos en el conjunto excede un umbral predefinido. Reestimar los parámetros del modelo usando todos los inliers identificados y terminar.

5. **Caso contrario**, repetir desde el paso 1 al 4, un número limitado de ocasiones.

2.3.2. Estimación del movimiento acumulado basado en la transformación afín

Se ha establecido que el modelo afín contiene los parámetros de movimiento entre dos fotogramas, uno a ser compensado y otro de referencia. Sin embargo, no se ha hecho mención al fotograma que se usará como consigna y al que se compensará.

El objetivo que se persigue es la estabilización del vídeo, por lo cual el fotograma a ser compensado es el fotograma actual, mientras que en el caso de la consigna tenemos 3 opciones:

- El fotograma inicial (I_0).
- El fotograma anterior (I_{t-1}).
- El fotograma anterior compensado (HI_{t-1}).

El fotograma inicial

Un candidato a fotograma es el inicial (Figura 2.6). En este punto es importante mencionar que el punto inicial de la secuencia de vídeo puede ser fijado en cualquier instante de tiempo, motivo por el cual es importante seleccionar un fotograma correspondiente al instante de tiempo en que el dispositivo se encuentra paralelo a la superficie. Si el fotograma se escoge de forma incorrecta, toda la secuencia de vídeo se ve comprometida.

La ventaja de utilizar el fotograma inicial continuamente como consigna es que, al realizar el cálculo constante de los parámetros de movimiento de H, no existe error acumulado. Sin embargo, en contraposición con esa ventaja existe una considerable desventaja respecto a la estabilidad del conjunto de fotogramas compensados que se obtiene. Dado que cada fotograma es compensado de forma independiente respecto al fotograma inicial, se genera un efecto vibratorio consecuencia de la aleatoriedad del error.

Otra desventaja importante es que, a medida que el fotograma actual se aleja del fotograma inicial, la deformación que se genera producto del movimiento se vuelve considerable, gran parte de la información se pierde, y el error en la estimación de los parámetros de movimiento se puede volver crítico.

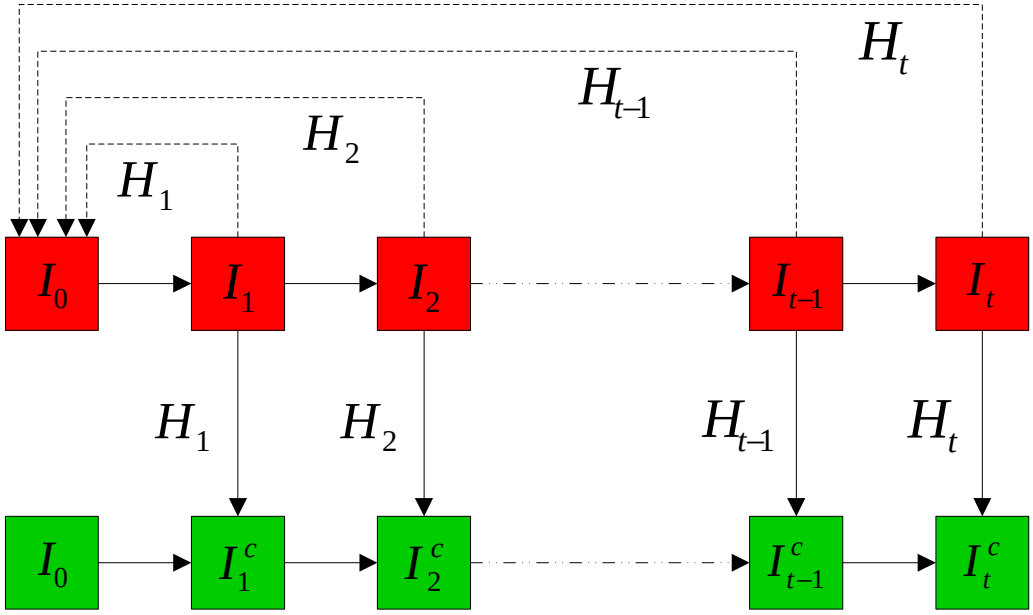

Figura 2.6. Esquema de estabilización de vídeo basada en el fotograma inicial.

El fotograma anterior

Un segundo candidato a consiga es el fotograma inmediato anterior al fotograma actual (Figura 2.7). Esta opción se basa en que la diferencia entre los instantes de tiempo, en que fotogramas adyacentes son capturados, es la mínima detectable por la cámara de vídeo. La transformación afín que se aplique al fotograma actual no es la que se compone únicamente de los parámetros de movimiento estimados entre dos fotogramas consecutivos, sino de la transformación afín acumulada entre fotogramas.

Si bien en el caso actual, los parámetros de movimiento contenidos en H son estimados a partir del frame en el instante t y $t-1$, cada H es acumulado como una transformación del fotograma inicial. Por lo tanto, al igual que en el primer caso, la selección de este fotograma inicial es fundamental.

La única desventaja que presenta es el error acumulado entre fotogramas consecutivos, problema que puede solventarse mediante técnicas de suavizado.

Nuestras propuestas para el suavizado de movimiento serán abordadas en los Capítulos 3 y 4.

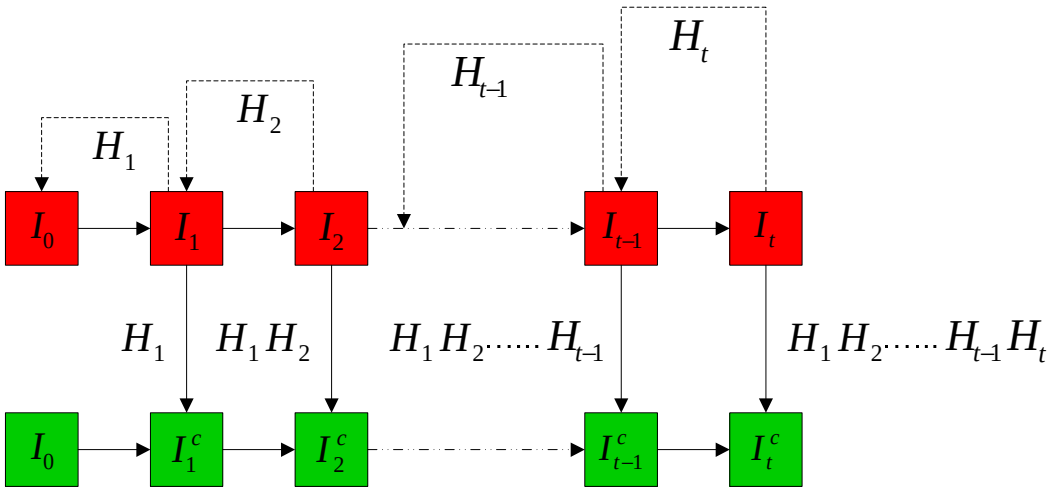

Figura 2.7. Esquema de estabilización de vídeo basada en el fotograma anterior.

El fotograma anterior compensado

Una última alternativa, que es una variante del segundo caso, es la utilización del fotograma anterior compensado (Figura 2.8). Sin embargo, esta alternativa no solo es la más costosa a nivel computacional, sino que además, utilizando como función costo base la diferencia matricial promedio por unidad de píxel entre la imagen compensada y la imagen consigna, experimentalmente es menos exacta que el enfoque basado en el fotograma anterior.

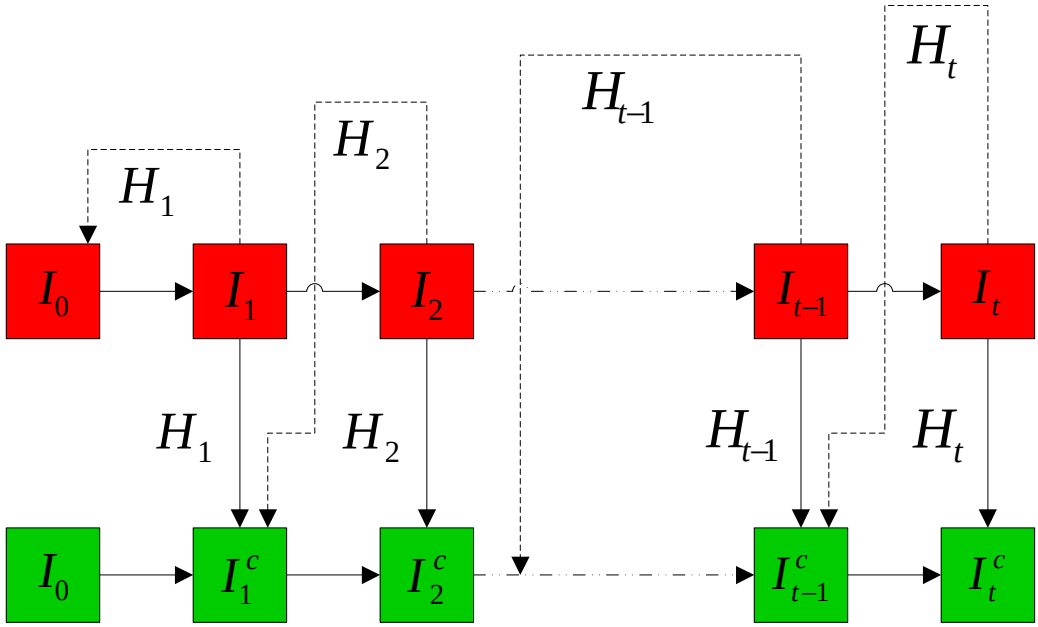

Figura 2.8. Esquema de estabilización de vídeo basada en el fotograma anterior compensado.

Opción seleccionada como fotograma consigna

Luego de llevar a cabo experimentalmente un análisis comparativo de los 3 enfoques propuestos basados en distinto candidatos a fotograma de referencia, se ha seleccionado el fotograma anterior como consigna. Las pruebas y resultados experimentales obtenidos se presentan en la Tabla 2.2.

Cabe señalar que una razón adicional por la cual se ha optado por el enfoque del fotograma anterior es que, a lo largo del tiempo, cada nuevo fotograma presenta una mayor deformación respecto a la consigna original, alcanzando un fotograma límite en el que la deformación sea tan alta que se imposibilite la estimación del movimiento. Incluso en el caso en que se utiliza el fotograma anterior o el fotograma anterior compensado como consigna, es importante que la consigna se refresque cada cierto número de fotogramas.

Como función costo se utilizó la diferencia promedio de nivel de gris normalizado de todos los fotogramas por unidad de píxel:

$$cost_0 = \frac{(\sum_i \sum_m \sum_n |HI_t - I_0|)}{i*m*n} \quad (2.22)$$

$$cost_{t-1} = \frac{(\sum_i \sum_m \sum_n |HI_t - I_{t-1}|)}{i*m*n} \quad (2.23)$$

$$cost_{(t-1)^c} = \frac{(\sum_i \sum_m \sum_n |HI_t - H_{t-1}I_{t-1}|)}{i*m*n} \quad (2.24)$$

Tabla 2.2. Función costo. Candidatos a fotograma consigna

Vídeo	$cost_0$	$cost_{t-1}$	$cost_{(t-1)^c}$
Vídeo 1	0.0528	0.0119	0.0125
Vídeo 2	0.0867	0.0198	0.0188
Vídeo 3	0.1392	0.0250	0.0219
Vídeo 4	0.1483	0.0196	0.0169
Vídeo 5	0.1050	0.0198	0.0160
Vídeo 6	0.0452	0.0113	0.0094
Vídeo 7	0.1071	0.0176	0.0169

En este punto es necesario definir las características del computador utilizado para el procesamiento de datos:

- Fabricante: Acer
- Modelo: Aspire 5951G
- Procesador: Intel® Core™ i7-2670QM 2.20GHz with Turbo Boost up to 3.1GHz
- Memoria instalada (RAM): 16,0 GB (15,9 GB utilizable)
- Tipo de sistema: Sistema operativo de 64 bits

2.3.3. Extracción de parámetros de movimientos

En la Sección 2.2.3 se describieron las ventajas que posee la selección del modelo afín, y una de ellas era justamente la simplicidad en el proceso de extracción de parámetros de movimiento. La extracción de la traslación t_x y t_y es directa. En el caso de el ángulo $\emptyset$ y la escala s, el cálculo se lo lleva a cabo de la siguiente forma:

$$R = \begin{bmatrix} s\,cos(\emptyset) & -s\,sin(\emptyset) \\ s\,sin(\emptyset) & s\,cos(\emptyset) \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} \tag{2.25}$$

$$\emptyset = tan^{-1}\left(\frac{R_{21}}{R_{11}}\right) = tan^{-1}\left(\frac{-R_{12}}{R_{22}}\right) \tag{2.26}$$

$$s = \frac{R_{11}}{cos(\emptyset)} = \frac{R_{12}}{-sin(\emptyset)} = \frac{R_{21}}{sin(\emptyset)} = \frac{R_{22}}{cos(\emptyset)} \tag{2.27}$$

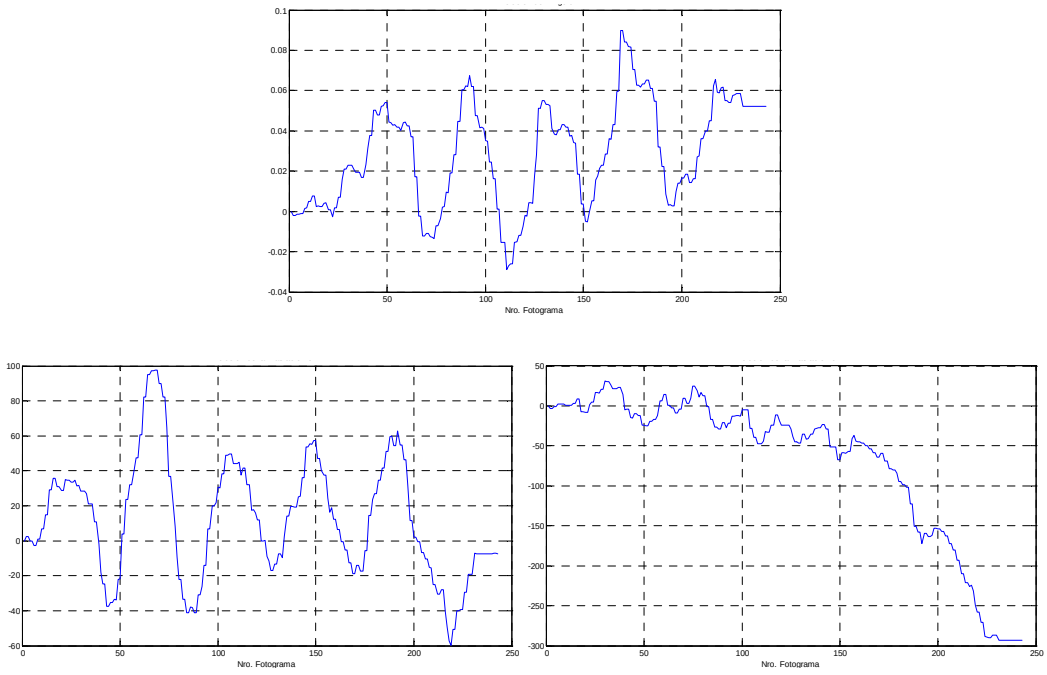

Figura 2.9. Vídeo 1. 5 pares de pasos. Superior (Ángulo). Izquierda (Traslación x). Derecha (Traslación y).

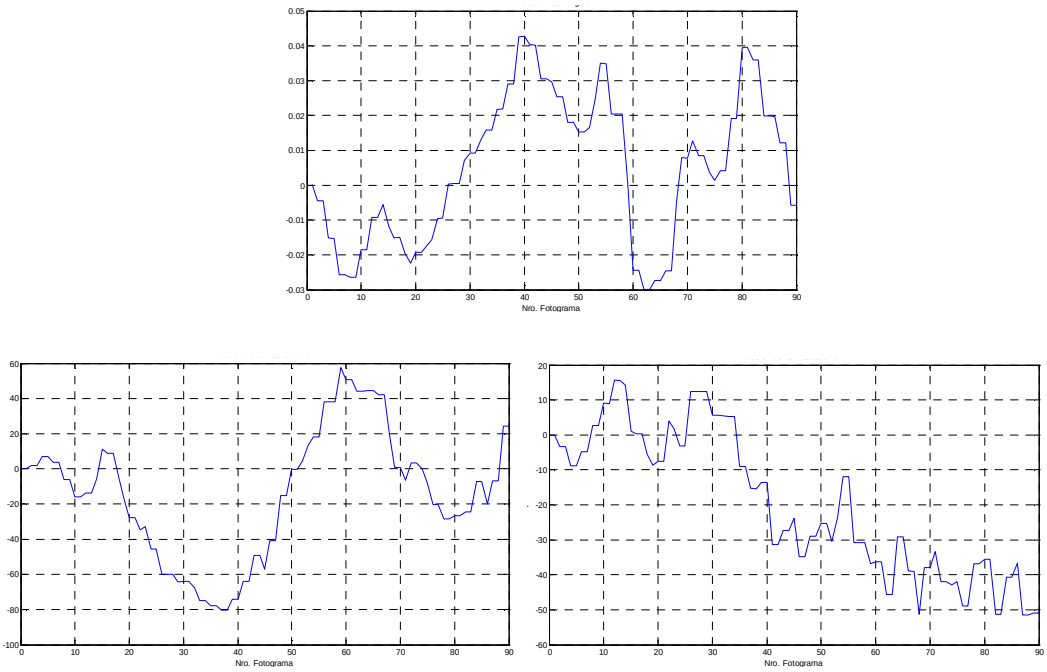

Figura 2.10. Vídeo 2. 2 pares de pasos. Superior (Ángulo). Izquierda (Traslación x). Derecha (Traslación y).

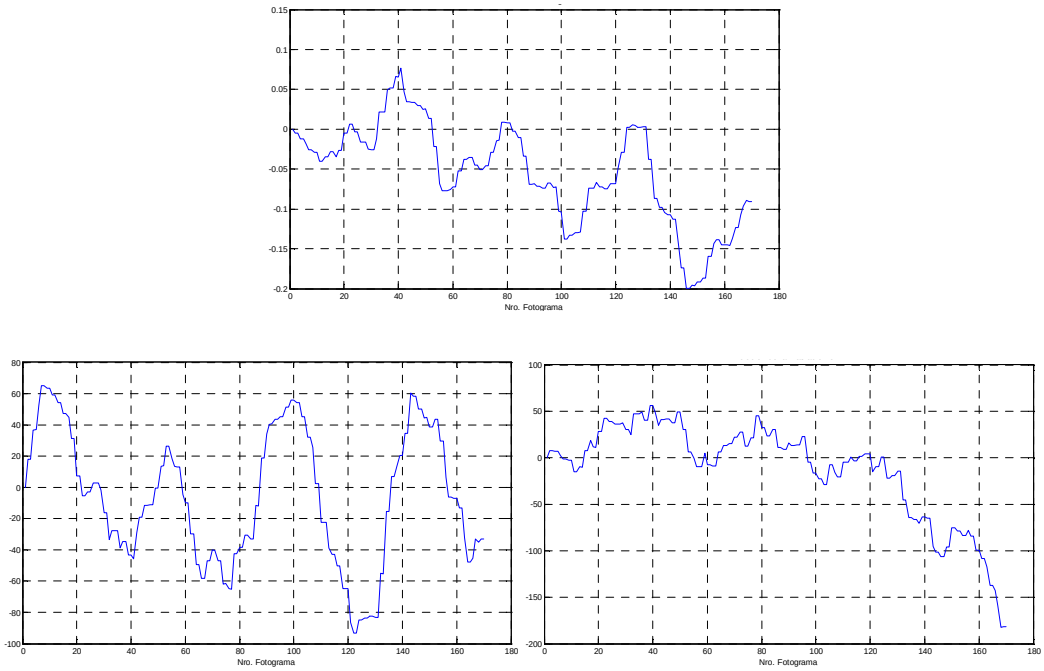

Figura 2.11. Vídeo 3. 4 pares de pasos. Superior (Ángulo). Izquierda (Traslación x). Derecha (Traslación y).

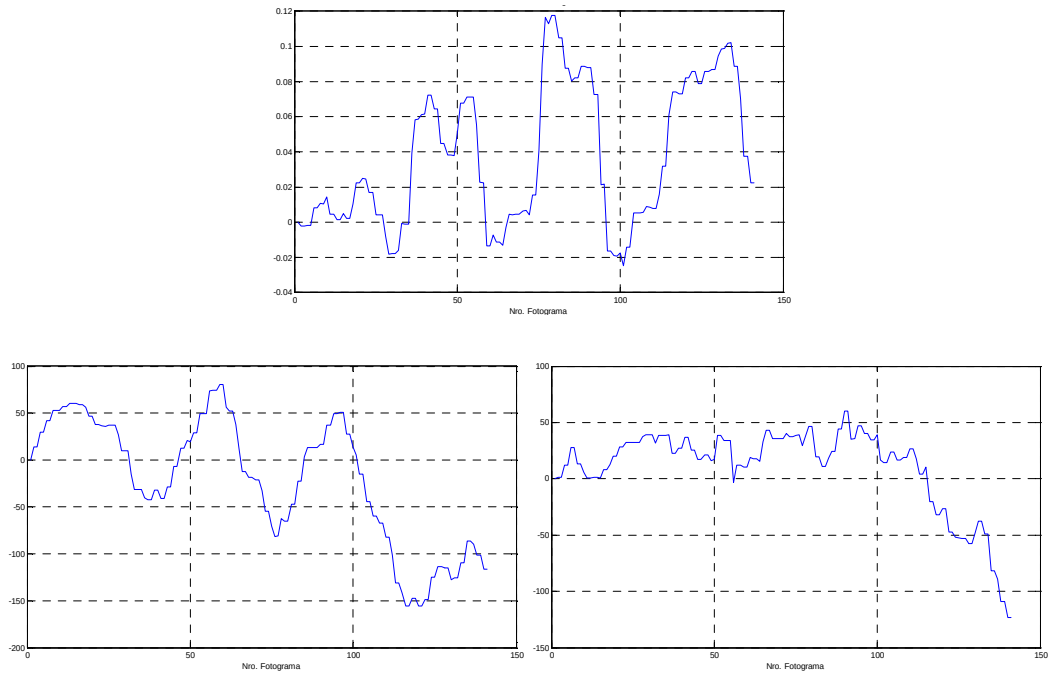

Figura 2.12. Vídeo 4. 3 pares de pasos. Superior (Ángulo). Izquierda (Traslación x). Derecha (Traslación y).

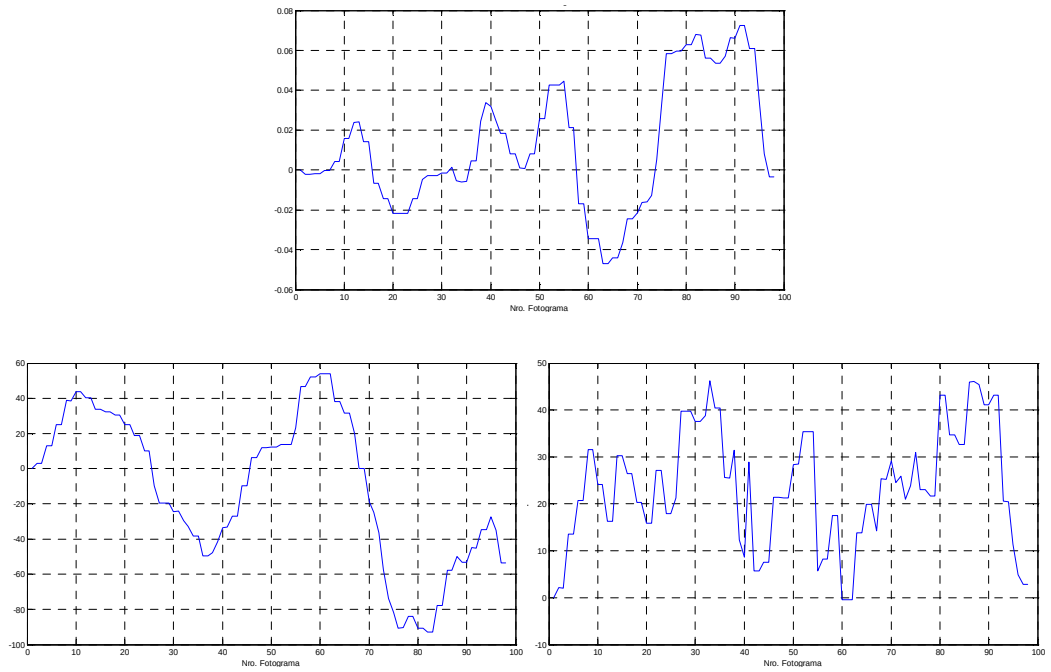

Figura 2.13. Vídeo 5. 2 pares de pasos. Superior (Ángulo). Izquierda (Traslación x). Derecha (Traslación y).

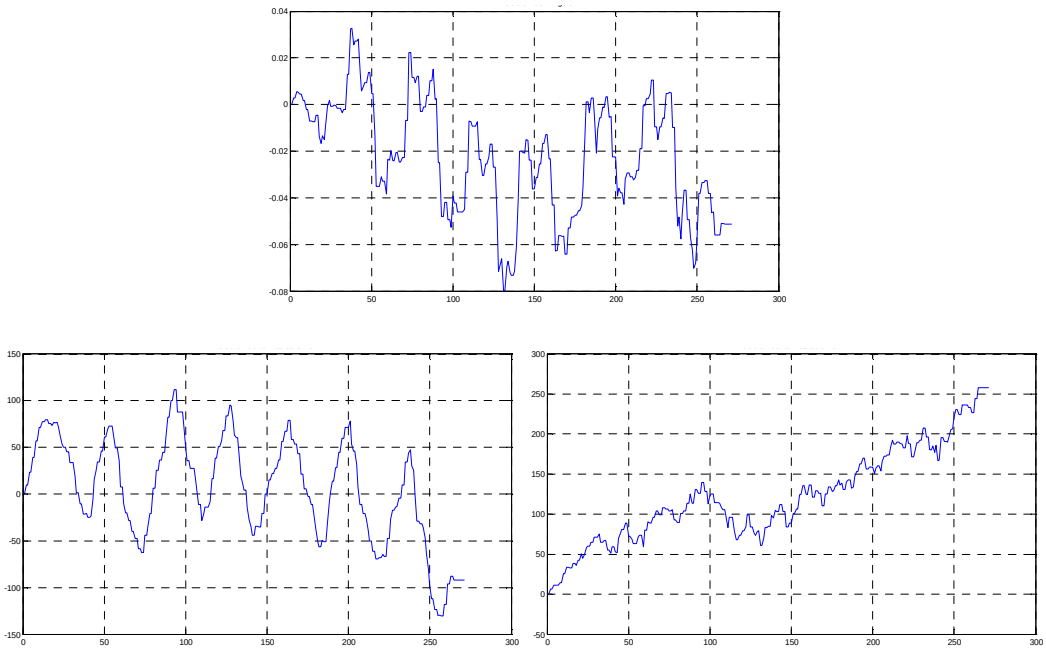

Figura 2.14. Vídeo 6. 7 pares de pasos. Superior (Ángulo). Izquierda (Traslación x). Derecha (Traslación y).

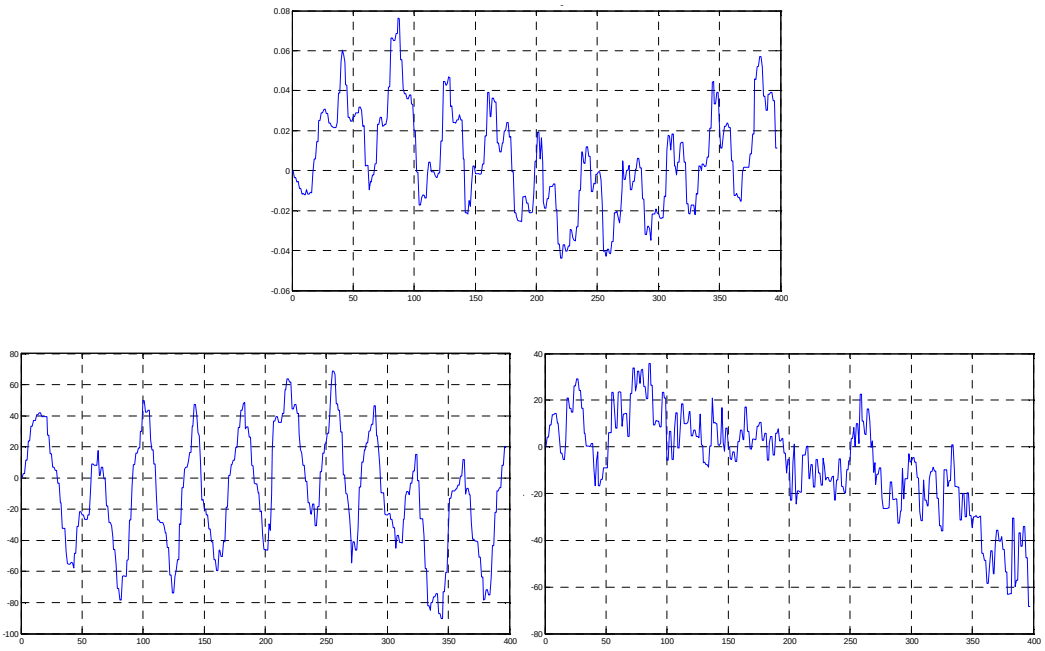

Figura 2.15. Vídeo 7. 10 pares de pasos. Superior (Ángulo). Izquierda (Traslación x). Derecha (Traslación y).

En las Figuras 2.9,10-2.15 se grafican los parámetros de traslación t_x, t_y y ángulo, extraídos del movimiento acumulado de un vídeo capturado por una persona al caminar. La cámara se ubica en la cintura de la persona con el objetivo de obtener un vídeo altamente inestable. Como resultado se puede percibir la presencia de mínimos y máximos locales a lo largo de una curva con tendencia sinusoidal.

Durante el movimiento de una persona que transporta la cámara, los instantes de tiempo, en que la pierna derecha del sujeto que transporta el dispositivo toca el suelo al caminar, coinciden con los mínimos locales de la curva, mientras que los instante de tiempo de la pierna izquierda coinciden con los máximos locales.

Estos datos abren un alternativo enfoque respecto a la detección de caminata. Utilizando como base la curvas sinusoidales, se puede determinar el número de pasos por unidad de fotograma, que en combinación con la frecuencia del vídeo, se puede determinar por unidad de tiempo. Asimismo, esta información puede ser complementada con la distancia recorrida en cada paso. La unificación de esta información permite determinar de forma indirecta la velocidad de caminata.

Sin considerar los tiempos muertos inicial y final, en la Figura 2.9 se aprecia con facilidad los instantes de tiempo en que los pies derecho e izquierdo alcanzan el suelo. La gráfica que representa con mayor evidencia la dinámica de movimiento, del individuo portador de la cámara, corresponde a la evolución de x, en la cual se distinguen claramente 5 pasos con la pierna derecha y 5 con la izquierda. Lo mismo sucede en las Figuras 2.10-2.15.

2.4 Compensación del movimiento en escenas rígidas

Una vez calculada la matriz de homografía, ya sea utilizando 4 o más puntos de interés entre dos fotogramas consecutivos capturados durante la navegación del robot, ésta se usa para compensar los efectos de rotación y traslación del

dispositivo de captura y reconstruir la imagen deseada, tal como se aprecia en la Figura 2.16, 2.17. Este proceso de reconstrucción se lleva a cabo de forma iterativa sobre una secuencia de fotogramas capturados por la plataforma robótica en la cual se está aplicando el método, en este caso el robot Aibo.

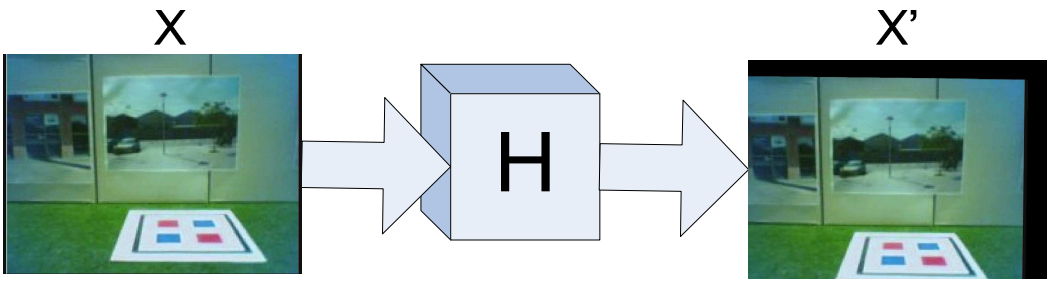

Figura 2.16. Aplicación de la homografía sobre la imagen deformada.

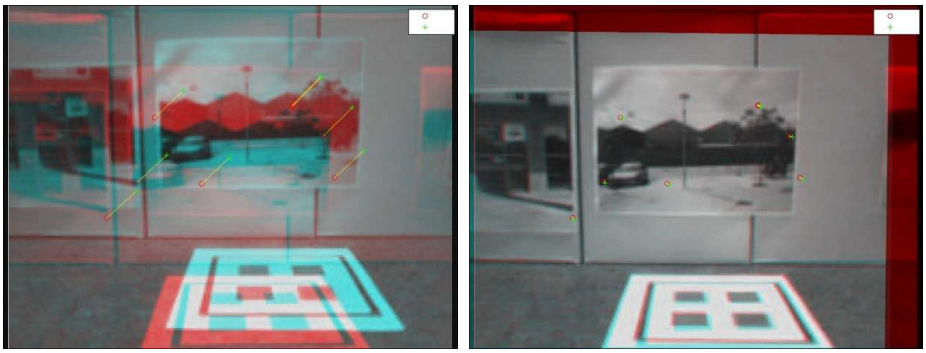

Figura 2.17. Puntos en correspondencia en la imagen deformada y compensada.

Sin embargo, regresando al objetivo original del trabajo de obtener una secuencia de imágenes estables que compense el efecto deformatorio de la traslación y rotación, generado como producto del movimiento de la cámara. Es necesario establecer un fotograma como imagen original que será el que sirva de referencia de navegación. Esta imagen original será la consigna que se pretende reconstruir a partir del conjunto de imágenes rotadas.

Para determinar el fotograma consigna es necesario conocer las características tanto del robot como de la cámara con la cual se adquieren las imágenes. Cuando el robot se encuentra en una posición en que la cámara se halla paralela al terreno de movimiento, se la considera como una posición original. La imagen capturada por la cámara será el fotograma que se utilice como consigna (Figura 2.18).

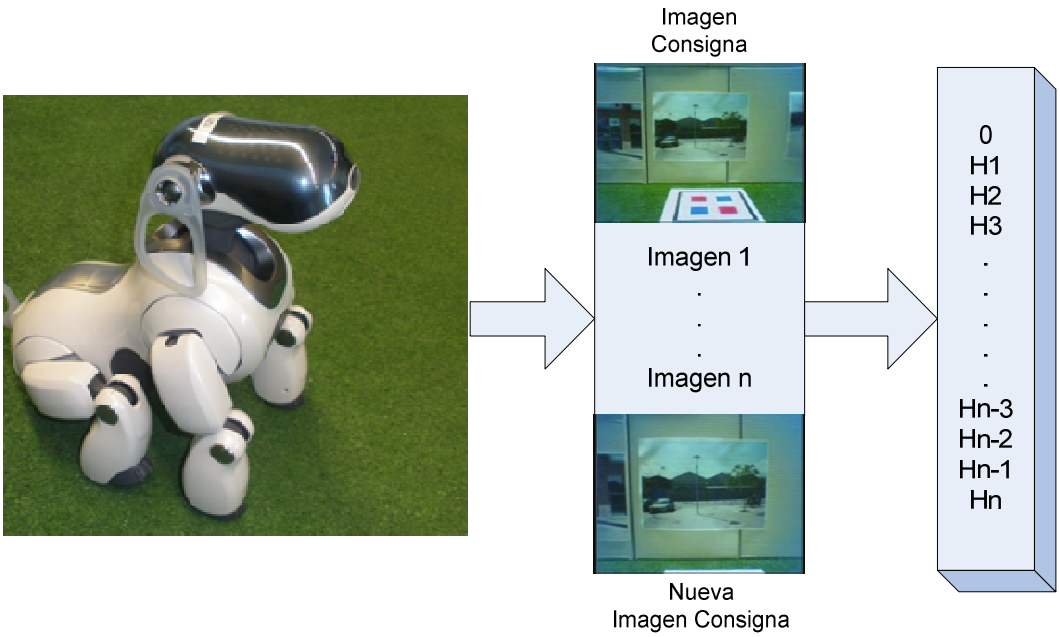

Figura 2.18. Conjunto de matrices de transformación generadas en la secuencia de imágenes.

A medida que se capturan nuevas imágenes, en la secuencia de vídeo de entrada del Aibo, cada nuevo fotograma sufre una mayor deformación de rotación y traslación respecto al fotograma original. Sin embargo, por la naturaleza cíclica del movimiento del Aibo al caminar, luego de un determinado periodo, los nuevos fotogramas tienden a asemejarse al primer fotograma hasta que la única deformación que se presenta entre el nuevo fotograma y el original sea una traslación en la dirección de avance, la misma que se traduce como escalado tal como lo muestran las Figuras 2.19, 2.20.

Figura 2.19. Reconstrucción de la imagen original a partir de la imagen deformada y de la matriz de homografía estimada.

Figura 2.20. Efecto deformatorio de la traslación única en la dirección de avance del robot.

Cuando se ha alcanzado un fotograma cuya deformación se debe casi exclusivamente a la traslación en la dirección de avance, se utiliza este fotograma como nueva consigna y se itera en el proceso de navegación.

2.5 Resultados y discusión

Finalmente, se puede obtener una perspectiva visual de los resultados obtenidos para cada uno de los escenarios sobre los cuales se ha llevado a cabo la experimentación. En las Figuras 2.21-2.27 se puede constatar la compensación del movimiento, basado en un modelo de transformación afín, en el cual, existe traslación a lo largo de los 2 ejes paralelos al plano de la imagen en el modelo de cámara pin-hole, rotación única alrededor del eje perpendicular a los 2 ejes anteriores y escalado. Con motivos ilustrativos se ha compensado la escala, de tal forma que se pueda apreciar con facilidad la relación entre la imagen, en un instante de tiempo específico, y la consigna. Sin embargo, como se ha mencionado se puede obviar la compensación del parámetro de escala, con el objetivo de evitar una secuencia de vídeo estática.

Figura 2.21. Vídeo 1: Fotograma 1, 30, 60 y 90.

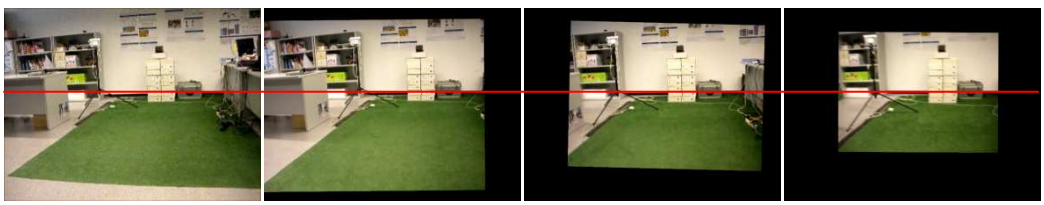

Figura 2.22. Vídeo 2: Fotograma 1, 30, 60 y 90.

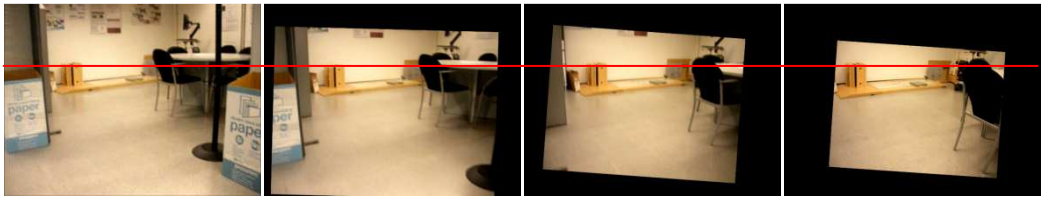

Figura 2.23 Vídeo 3: Fotograma 1, 30, 60 y 90.

Figura 2.24. Vídeo 4: Fotograma 1, 30, 60 y 90.

Figura 2.25. Vídeo 5: Fotograma 1, 30, 60 y 90.

Figura 2.26. Vídeo 6: Fotograma 1, 30, 60 y 90.

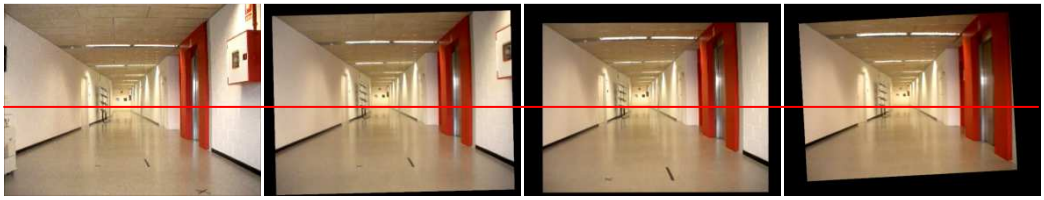

Figura 2.27. Vídeo 7: Fotograma 1, 30, 60 y 90.

Es importante mencionar algunas de las limitaciones que eventualmente se pueden presentar:

- Presencia de objetos cercanos
- Escenas con objetos en movimiento
- Desplazamientos significativos
- Vídeos de baja frecuencia
- Desplazamientos de alta velocidad

Estas limitaciones se solventan en los capítulos posteriores.

2.6 Conclusiones

Aunque el mínimo número de puntos de interés requeridos para estimar la homografía es 8, una solución sobredeterminada permite obtener una estimación más fiable que será usada en el modelo de compensación.

El modelo matemático que mejor se relaciona con movimientos indeseados típicos es el modelo afín; esto se debe a que las rotaciones pich y yaw son mínimas, y su impacto sobre la deformación final de la imagen es casi imperceptible.

El mejor candidato a consigna es el fotograma anterior, tanto desde el punto de vista de la función costo planteada, como de tiempo de cómputo. A pesar que el uso de este candidato como consigna minimiza el efecto vibratorio de fotograma a fotograma, se puede generar un error acumulado. Este última problemática puede ser solventada utilizando técnicas de suavizado y segmentación.

Los resultados obtenidos para la estimación de movimiento a partir de vídeos capturados con cámara ubicadas a la altura de la cintura de una persona abren un enfoque alternativo respecto a la detección de caminata y movimientos humanos.

Capítulo 3: Intención de movimiento basada en la acción de control

En el presente capítulo se propone una combinación del modelo proyectivo y afín para obtener una transformación de confianza (robustez) con un bajo costo computacional (rapidez) y una baja deformación (calidad). Adicionalmente se introduce una propuesta que utiliza, para estimar la intención de movimiento, una combinación de un filtro pasa-bajos e información de la acción de control.

El Capítulo 3 está organizado de la siguiente forma: Primero se explica brevemente las fases en las que se dividen la mayoría de los algoritmos de estabilización de vídeo. Luego se detalla la estimación de los parámetros de movimiento inter-fotograma en nuestro método. Adicionalmente, se describe una

combinación de RANSAC y una función costo basada en la diferencia de nivel de gris para la estimación robusta del movimiento inter-fotogramas. En la sección de estimación de la intención de movimiento, se presenta una técnica de suavizado del movimiento basado en un filtro pasa-bajos. La sección de estabilización de vídeo en tiempo real se enfoca en la optimización del algoritmo con el mínimo número de fotogramas para estimar la intención de movimiento. Se propone un nuevo enfoque para solucionar el problema de los movimiento fantasma. Finalmente, se presentan resultados experimentales y conclusiones del capítulo.

3.1 Introducción

Como se mencionó en el Capítulo 1, la robustez de los sistemas de control, navegación y guiado para MAVs [1] dependen de la información de entrada obtenida de sensores y cámaras onboard (a bordo del vehículo). Los movimientos indeseados se generan usualmente durante el vuelo como resultado de las complejas características aerodinámicas del UAV. Rotaciones y translaciones indeseadas de la imagen se presentan en la secuencia de imágenes, aumentando la dificultad de control del vehículo.

Existen múltiples técnicas en la literatura diseñadas para compensar los movimientos indeseados de la cámara [70], [78], [79]. No obstante, dos de los algoritmos que mejor se desempeñan en post-procesamiento son: (a) El algoritmo de estabilización de vídeo L1 Optimal proporcionado por el Editor de Youtube que se introdujo en [12] y [41], y (b) el algoritmo Subspace vídeo Stabilization publicado en [11] y utilizado en el software comercial Adobe After Effects. El Director Mode de Parrot es una aplicación disponible para iOS y Android, para el post-procesamiento de vídeos capturados con los AR.Drones de Parrot.

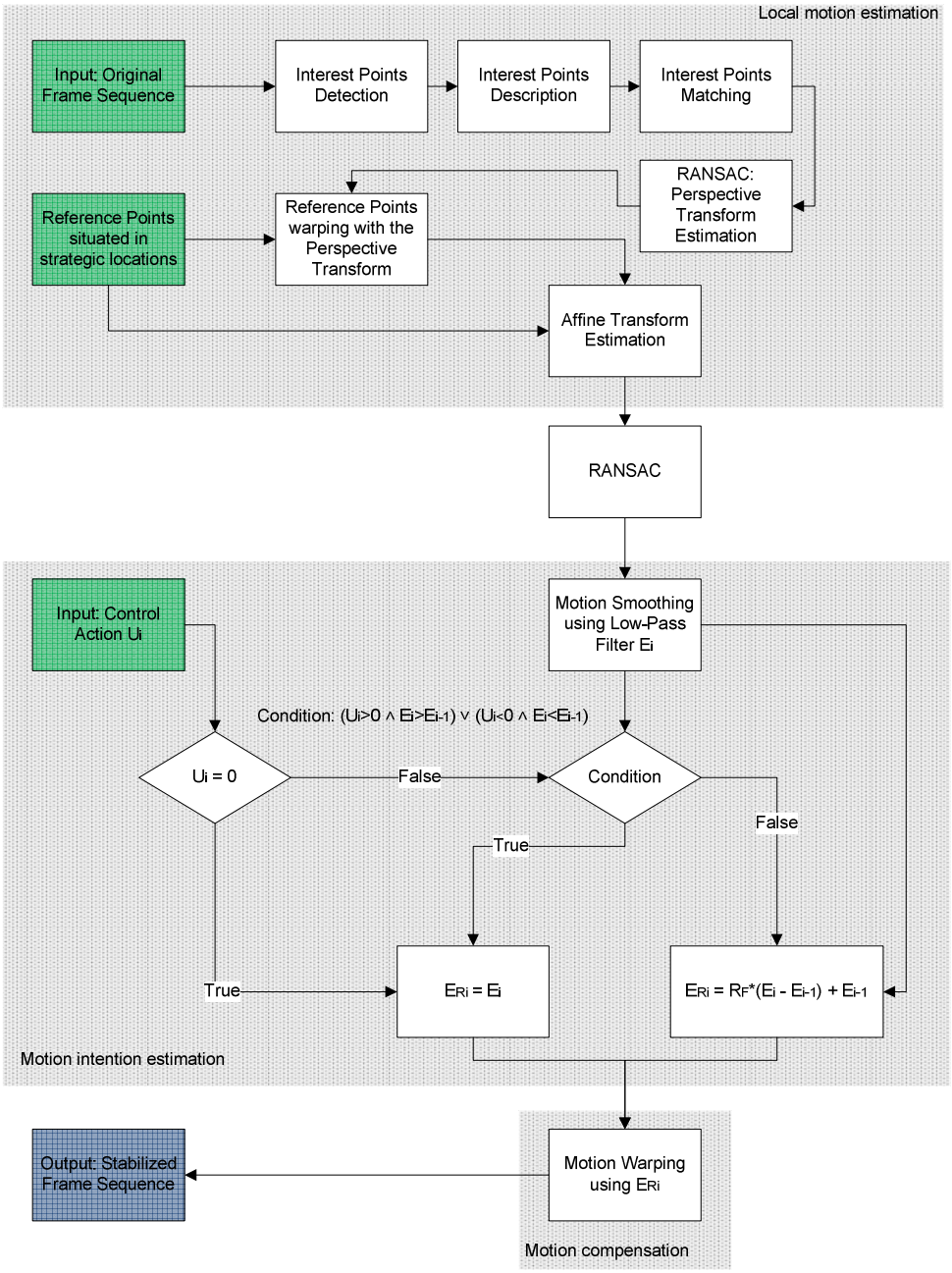

Figura 3.1. Diagrama de flujo. Propuesta para estabilización de vídeo. Uso de una combinación del suavizado de movimiento y la acción de control de entrada.

Una apuesta reciente de la empresa francesa Parrot es el Bebop, un micro vehículo aéreo basado en la tecnología de las versiones previas del AR.Drone 1.0 y

2.0. Este nuevo MAV de Parrot tiene la capacidad de estabilizar el vídeo capturado por la cámara en tiempo real mediante el uso del un microcontrolador destinado exclusivamente a esta función. Evidentemente esta tecnología ha incrementado considerablemente el costo del drone en más de 200 euros adicionales.

La mayor parte de estas técnicas de estabilización de vídeo son desarrolladas en tres fases:

- Estimación de movimiento inter-fotograma
- Estimación de la intención de movimiento
- Compensación del movimiento

Nuestro algoritmo completo se presenta en la Figura 3.1.

3.1.1 Estimación del movimiento inter-fotograma

Los dos enfoques estándares utilizados para estimar los parámetros en la compensación de movimientos que relacionan dos fotogramas consecutivos son el flujo óptico [80], [81] y los modelos de transformación geométrica [49], [53], [82]. Los modelos de transformación geométrica están basados en la estimación de los parámetros de movimiento. Para esta estimación es necesario que los puntos de interés sean detectados y descritos. Una lista de técnicas para llevar a cabo esta tarea se puede encontrar en la literatura [83]–[85], además de los cinco algoritmos mencionados en el capitulo anterior. Nuestra contribución no está enfocada en la reducción de los retardos que son consecuencia de la estimación de los puntos de interés. El retardo debido a las técnicas de suavizado es considerablemente mayor. Se continua utilizando SURF como detector y descriptor de puntos de interés del estado del arte pese a que ORB presenta un mejor desempeño como se comentó en el capítulo anterior.

La segunda parte del proceso de estimación de movimiento es la búsqueda de correspondencia de puntos de interés entre fotogramas consecutivos. Esta parte es crítica debido a que los parámetros de movimiento estimados dependen directamente de la confiabilidad de los puntos en correspondencia. Las falsas correspondencias se removerán usando la técnica iterativa conocida como RANdom SAmple Consensus (RANSAC), a la cual empleamos en el Capítulo 2 y que se basa en el modelo que se obtiene de los pares de puntos en correspondencia [75]–[77], [86]. La función de costo que se usa para RANSAC solo se basa en la diferencia del nivel de gris, lo que minimiza el retardo.

3.1.2 Estimación de la intención de movimiento

Con el objetivo de obtener vídeos estables, pero no estáticos como en el Capítulo 2, los parámetros de movimiento inter-fotogramas se acumulan a lo largo de la secuencia de vídeo completa. Este movimiento acumulativo incluye movimientos indeseados y deseados, es decir, aquellos realizados por el teleoperador y los que son producto de otros factores. Los movimientos intencionales se estiman mediante la eliminación de las señales de alta frecuencia del movimiento acumulado completo. Múltiples métodos de suavizado de movimiento se encuentran disponibles para la estimación de la intención de movimiento como particle filter [49], Kalman filter[82], Gaussian filter [40], [30]. adaptive filter [87], [88], spline smoothing [89], [90], o point feature trajectory smoothing [91], [92]. Estos algoritmos se enfocan en el seguimiento de puntos de interés, respecto a los cuales se compensa el movimiento. Con base en esta idea, el objetivo del suavizado de movimiento es obtener el movimiento intencional sobre los puntos característicos y no sobre los parámetros de movimiento inter-fotograma.

Alternativamente, la intención de movimiento puede ser estimada a partir de los parámetros de movimiento en lugar de los puntos de característicos [63], como se lo planteó en el capítulo anterior. En nuestro enfoque se presenta una nueva

metodología respecto a las de la literatura, donde la señal de control enviada al MAV se procesa como información conocida. Una combinación de un filtro pasa-bajos de segundo orden, usando el mínimo número de fotogramas posibles, y la entrada de la acción de control se emplea para estimar la intención de movimiento con mayor confiabilidad. Se consiguió reducir el número de fotogramas (ventanas temporales) requeridos para el suavizado de la imagen usando un proceso de optimización, lo que se explicará en la Sección 3.4.1.

3.1.3 Compensación de movimiento

Finalmente y al igual que en la metodología empleada en el Capítulo 2, se deforma el fotograma actual usando los parámetros de movimiento obtenidos de la intención de movimiento para generar una secuencia de vídeo estable.

3.2 Propuesta para estimación del movimiento inter-fotograma

La transformación geométrica [35], [74], [93] se usa para describir la relación matemática entre dos imágenes consecutivas en la secuencia de vídeo. Una imagen es la referencia y otra el fotograma a ser procesado. Esta relación matemática puede ser representada como:

$$I_{sp} = H_t \cdot I_t \tag{3.1}$$

donde $I_{sp} = [x_{sp}, y_{sp}, 1]^T$ y $I_t = [x_t, y_t, 1]^T$ son las coordenadas de los puntos de interés en la imagen referencia y la imagen no compensada, respectivamente, y H_t es la matriz de transformación geométrica.

Esta matriz contiene los parámetros de movimiento que dependen del modelo usado para representar el efecto de deformación generado entre dos fotogramas consecutivos durante el movimiento de la cámara. Los modelos paramétricos de movimiento pueden ser 2D o 3D. Los modelos 2D son ampliamente utilizados en algoritmos de estabilización de vídeo y los más comunes son los mencionados en el Capítulo 2: modelo de traslación, afín, semejanza no reflectiva y proyectivo u homografía. En nuestro algoritmo se utilizó la transformación afín para obtener una estimación robusta del movimiento inter-fotogramas.

3.2.1 Usando la transformación proyectiva

Como se mencionó, existen múltiples enfoques para el cálculo de los puntos de interés. Nuestro algoritmo de estabilización de vídeo se basa en SURF por lo que cada punto característico detectado tiene un descriptor 64-dimensional asociado. La transformación geométrica inter-fotograma se basa en los puntos característicos calculados, representados en el espacio 64-dimensional de los descriptores SURF, para cada imagen. Los puntos de un fotograma se deben emparejar con sus correspondencias en el otro fotograma. El proceso de búsqueda de correspondencias localiza los vecinos más cercanos, es decir, el par de puntos característicos con la distancia euclidiana mínima en este espacio 64-dimensional.

Para imágenes capturadas en condiciones no controladas, el proceso de emparejamiento genera inevitablemente falsas correspondencias. El algoritmo RANSAC es comúnmente usado para la desestimación de los pares de puntos incorrectamente emparejados. Por consiguiente, se propone un enfoque similar al del Capítulo 2, usando en este caso la transformación proyectiva en lugar de la

transformación afín como modelo matemático del algoritmo RANSAC. La transformación afín se utilizará en la siguiente fase.

La transformación proyectiva, también llamada homografía, contiene seis parámetros, tres de rotación y tres de traslación. La matriz de transformación se encuentra compuesta por ocho parámetros linealmente independientes,

$$H_t = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{32} \\ h_{31} & h_{32} & 1 \end{bmatrix} \tag{3.2}$$

El algoritmo RANSAC (Algoritmo 3.1) se aplica luego del proceso de búsqueda de correspondencia de puntos de interés. En cada iteración, la transformación proyectiva H_t se estima con base en cuatro pares de puntos seleccionados aleatoriamente y usados para deformar al fotograma H_t. Como función costo, se usa la diferencia de nivel de gris entre el fotograma de referencia y el fotograma actual deformado por H_t.

Finalmente, se seleccionan los parámetros de la transformación proyectiva que minimicen la función costo:

$$arg\, min_{h_{ij}} \sum_j \left| Frame_t^{'} - Frame_{sp} \right| \tag{3.3}$$

donde $Frame_t^{'}$ y $Frame_{sp}$ son los fotogramas deformado y de referencia, respectivamente.

Algoritmo 3.1 Algoritmo RANSAC basado en la función costo

for $j = 1$ to N **do**

Estimación de la transformación proyectiva j-ésima: H_j

j-ésima deformación del i-ésimo fotograma: $Frame_t^{'}$

Cálculo de la función costo j-ésima: $J_j = \left|Frame_t^{'} - Frame_{sp}\right|$

end for

Selección de los parámetros de H_{opt} que minimicen la función costo: $arg\ min_{h_{ij}} \sum_j \left|Frame_t^{'} - Frame_{sp}\right|$

3.2.2 Definiendo el fotograma de referencia

Es importante especificar el fotograma a ser compensado y el fotograma a ser usado como referencia en el algoritmo. El fotograma actual será deformado mediante compensación del movimiento, obteniendo una secuencia estable de movimiento en el vídeo de salida. Por otra parte, para el fotograma de referencia existen diferentes alternativas.

Un estudio llevado a cabo en [94] y explicado en el Capítulo 2, compara tres candidatos a fotograma de referencia: el fotograma inicial ($Frame_{sp} = Frame_0$), el fotograma previo ($Frame_{sp} = Frame_{t-1}$) y el fotograma previo compensado ($Frame_{sp} = Frame_{t-1}^{'}$). El análisis de los tres enfoques propuesto se replicó con datos obtenidos de la cámara a bordo de micro vehículos aéreos, basándose en el

error cuadrático medio (MSE) entre las imágenes monocromáticas de dimensión $M \cdot N$,

$$MSE(k) = \frac{1}{M \cdot N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \|I_k(i,j) - I_{k-1}(i,j)\|^2 \quad (3.4)$$

Nuevamente el fotograma previo $Frame_{t-1}$ es el mejor candidato a referencia.

3.2.3 Usando la transformación afín

Para cámaras y dispositivos de visión monocular a bordo de MAVs, muchos de los movimiento indeseados y vibraciones parásitas en la imagen se consideran significativos únicamente alrededor del eje roll. El modelo afín es el modelo geométrico seleccionado para describir estos movimientos por tres razones:

- El modelo afín es capaz de representar los principales movimientos indeseados de cámaras a bordo de micro vehículos aéreos.
- Pese a que la fiabilidad de la transformación proyectiva es mayor como modelo de RANSAC, la deformación del fotograma compensado con la transformación afín es menor y el vídeo final es más estable.
- Los parámetros relevantes de movimiento se pueden extraer directamente de la matriz de transformación. Estos parámetros son esenciales para estimar la intención de movimiento.

Los parámetros de movimiento del modelo son: dos traslaciones en el plano paralelo a la imagen, rotación roll ϕ alrededor del eje perpendicular al plano xy, y la escala s que es proporcional al movimiento en la orientación del eje roll,

$$H_t = \begin{bmatrix} s\cos(\phi) & -s\sin(\phi) & t_y \\ s\sin(\phi) & s\cos(\phi) & t_x \\ 0 & 0 & 1 \end{bmatrix} \quad (3.5)$$

En el modelo afín, se pueden considerar dos posibles ángulos: $tan^{-1}\frac{H_t(2,1)}{H_t(1,1)}$ y $tan^{-1}\frac{H_t(1,2)}{H_t(2,2)}$. Se estima el ángulo medio ajustable a estos valores. Este modelo es conocido como semejanza no reflectiva y es un caso particular del modelo afín.

3.2.4 Usando una combinación de transformaciones

Algunas técnicas de estabilización de vídeo rápida emplean técnicas de suavizado de la trayectoria de los puntos característicos, sin embargo, este enfoque requiere una continua estimación de la pose 3D. Para aplicaciones en tiempo real, este método no es recomendable debido al alto costo computacional requerido para la estimación de la pose 3D en cada punto.

En lugar de ello, nuestro enfoque usa la transformación afín basada en la transformación proyectiva.

Por un lado, la homografía es más confiable en RANSAC que el modelo afín. Sin embargo, el valor ITF (Fidelidad inter-fotograma) medido entre fotogramas consecutivos estabilizados usando la homografía, es menor que usando el modelo afín. Esto se debe a que el modelo proyectivo contiene tres rotaciones que incrementan la deformación en la imagen. Adicionalmente, nuestra propuesta optimiza la matriz homográfica usando RANSAC. A continuación, se seleccionan tres puntos de referencia para computar el ángulo y la escala, y uno adicional para obtener la traslación en el eje 2D. La transformación proyectiva usada para estimar el modelo afín permite calcular el ángulo de rotación medio de la imagen y reduce el error acumulado.

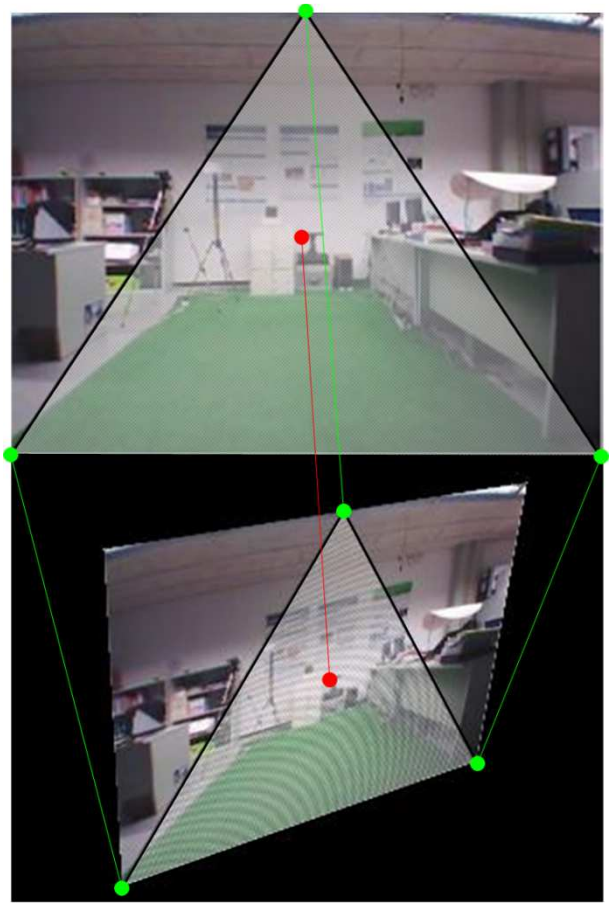

Figura 3.2. Puntos de referencia y deformados.

Se sitúan tres puntos de referencia (puntos en los bordes de la Figura 3.2) en ubicaciones estratégicas con el objetivo de maximizar el área capturada. Si los puntos se encuentran cerca, la región encerrada por ellos es pequeña, por consiguiente se buscará la mayor distancia posible entre los puntos. Existen varias opciones para ubicar tres puntos en una imagen rectangular con la distancia máxima entre ellos.

Considerando que se usa una cámara a bordo de un micro vehículo aéreo, se seleccionó la opción triangular de la Figura 3.3. La razón de esta decisión es que la cámara a bordo del MAV se debe enfocar principalmente en la región frontal inferior de la escena durante el vuelo. Otra opción es calcular el valor medio entre

los ángulos de la transformación afín estimada con cada distribución de puntos característicos. Sin embargo, dependiendo de la secuencia, esto podría generar un efecto de oscilación en la salida.

Figura 3.3. Área de interés.

Luego de calcular los parámetros afín sin error acumulado usando la homografía como referencia, se estima el modelo de traslación a partir del cuarto punto localizado en el centro de la imagen como referencia y su correspondencia estimada con la transformación geométrica. El uso de este cuarto punto de referencia garantiza la estabilización respecto al centro de la imagen. Mediante la unificación de las transformaciones, se obtiene la matriz de compensación. Cuando se aplica en

el fotograma actual, se genera un fotograma compensado similar al fotograma de referencia. Por tanto, los movimientos inter-fotograma son minimizados a partir de la secuencia de vídeo completa para obtener una escena tan similar al fotograma de referencia como sea posible, compensando los movimiento indeseados un fotograma a la vez.

3.3 Estimación de la intención de movimiento

El algoritmo RANSAC, basado en la minimización de la diferencia de nivel de gris, es suficiente para la obtención de robustez en la compensación de la imagen para escenas estáticas (sin movimientos intencionales en la imagen) [94], [95]. Nuestro objetivo es alcanzar una estabilización robusta de secuencias de vídeo capturadas con cámara a bordo de micro vehículos aéreos. Por lo general, los vídeos capturados tanto con robots aéreos como con dispositivos móviles corresponden a escenas dinámicas, es decir, escenas que contienen movimientos intencionales. Bajo esta perspectiva, algunos movimientos del dispositivo de captura no se deben eliminar, sino compensar suavemente generando un vídeo estable en lugar de una escena estática.

Múltiples algoritmos de estabilización de vídeo usan métodos de suavizado para la estimación de la intención de movimiento, tales como el filtro de Kalman, filtro Gaussiano y filtro de partículas. Nuestro enfoque se basa en el filtro Butterworth de segundo orden, un filtro ampliamente utilizado en el suavizado de señales [96].

Nuestra plataforma de experimentación es un MAV de bajo costo, cuyo comportamiento durante vuelos internos presenta una dinámica compleja. Consecuentemente los vídeos capturados con la cámara, a bordo de vehículo, usualmente contienen desplazamientos significativos y movimientos de alta frecuencia en el plano perpendicular al eje roll. Se debe considerar los efectos

provenientes de problemas de comunicación wireless, tales como vídeos de baja frecuencia o el congelado de la imagen.

Usando un filtro pasa-bajos como estimador de la intención de movimiento, se pueden evitar varios problemas asociados a vuelos internos. Los efectos de congelado pueden eventualmente continuar presentes debido a comunicaciones de baja calidad. Aquellos parámetros de movimiento, estimados a partir de los fotogramas congelados, se deben descartar antes de continuar con el proceso de estimación.

Una vez que se extraen los parámetros de la transformación afín (escala rotación, translación x, y), y se eliminan los valores de los parámetros que corresponden a pantallas congeladas, el filtro pasa-bajos calcula la intención de movimiento como una salida sin señales de alta frecuencia. Las bajas frecuencias se encuentran asociadas al movimiento intencional, mientras que las frecuencias altas son referidas a los movimiento indeseados. Bajo esta perspectiva, la frecuencia de corte dependerá de las características del sistema y de la aplicación. Un valor alto en la frecuencia de corte implica una salida de vídeo similar a los movimientos originales que incluye a los movimientos indeseados, es decir, un vídeo de baja estabilidad. Un valor bajo implica una salida de vídeo que elimina los movimientos intencionales, es decir, un vídeo menos fiel al movimiento real de la cámara. En consecuencia, es importante encontrar un compromiso entre la estabilidad del vídeo y su congruencia con el movimiento deseado. Se usó un filtro de segundo orden con una frecuencia de corte de 66.67 Hz para el suavizado de la señal de los cuatro parámetros de movimiento. Una opción de alternativa es usar un filtro diferente para cada parámetro de movimiento.

Un movimiento indeseado puede ser estimado mediante substracción de la intención de movimiento, obteniendo una señal de alta frecuencia. Esta señal es usada en la deformación de la imagen para compensar las vibraciones y simultáneamente mantener los movimientos intencionales. En la Figura 3.4, se

puede observar la señal de intención de movimiento estimada con un filtro pasa-bajos (gráfica superior), y la señal de alta frecuencia a ser compensada (gráfica inferior) para el parámetro de deformación, en este caso el ángulo. Gráficas similares pueden ser observadas en las Figuras 3.5, 3.6 y 3.7 para la escala, y las traslaciones en los ejes x, y respectivamente.

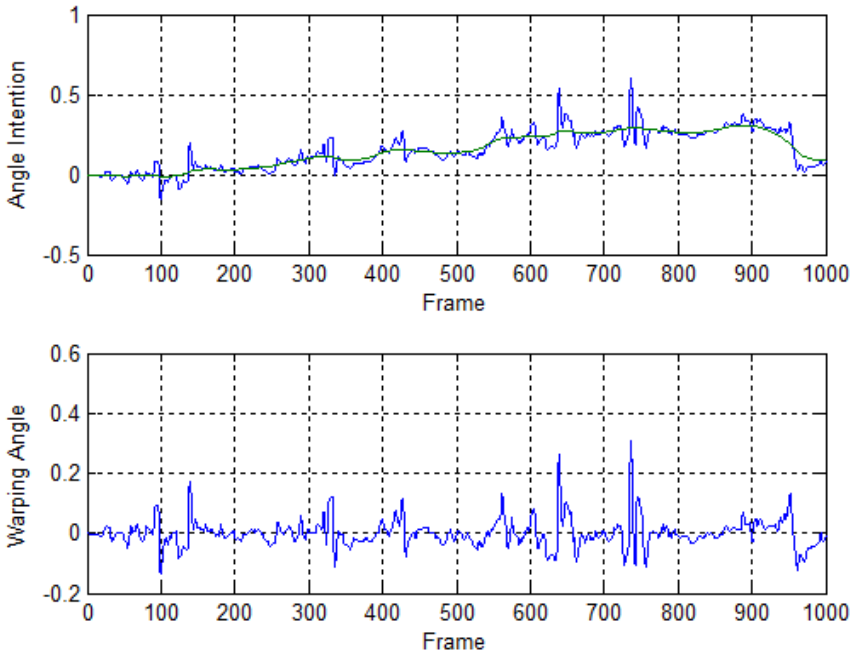

Figura 3.4. Ángulo. Superior: Señales de movimiento acumulado (azul) e intencional (verde) estimados con un filtro pasa-bajos. Inferior: Señal de alta frecuencia a ser compensada.

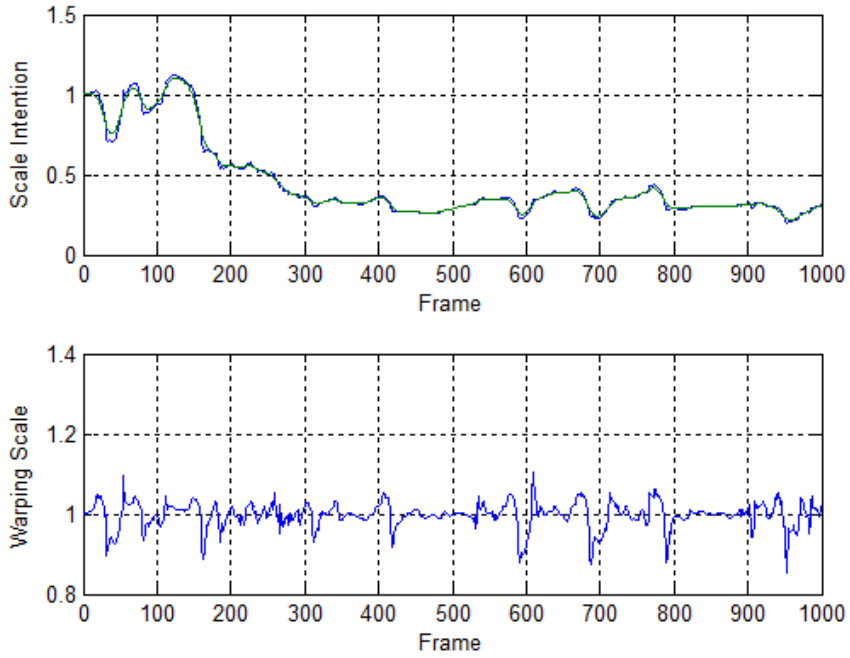

Figura 3.5. Escala. Superior: Señales de movimiento acumulado (azul) e intencional (verde) estimados con un filtro pasa-bajos. Inferior: Señal de alta frecuencia a ser compensada.

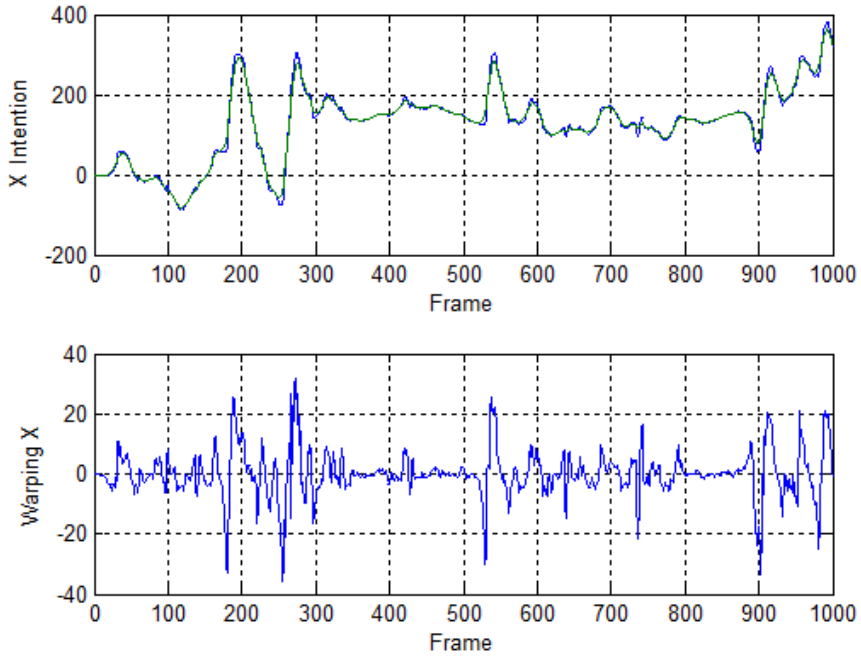

Figura 3.6. Traslación en el eje-x. Superior: Señales de movimiento acumulado (azul) e intencional (verde) estimados con un filtro pasa-bajos. Inferior: Señal de alta frecuencia a ser compensada.

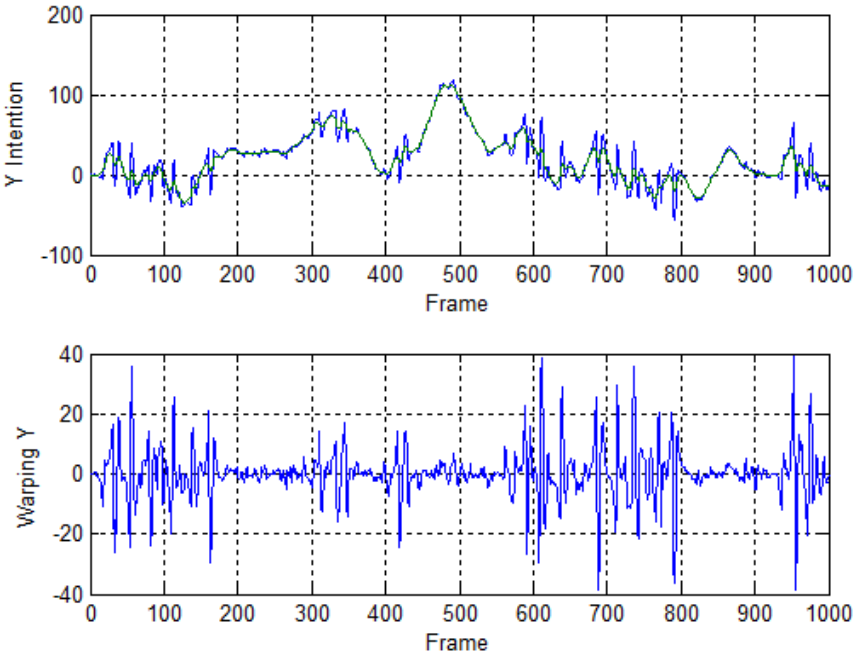

Figura 3.7. Traslación en el eje-y. Superior: Señales de movimiento acumulado (azul) e intencional (verde) estimados con un filtro pasa-bajos. Inferior: Señal de alta frecuencia a ser compensada.

3.4 Estabilización de vídeo en tiempo real

Se detalló un algoritmo robusto de post-procesamiento para la estabilización de vídeo, no obstante, el objetivo es conseguir una versión capaz de aplicarse en tiempo real. En este contexto y aunque existen varias técnicas de estabilización de vídeo en tiempo real, un problema a solventar es el tiempo de cómputo. Existen algunas alternativas para minimizar este tiempo de cómputo, como en [97] mediante el uso de algoritmos óptimos de detección y descripción de puntos de interés. Este método reduce el tiempo de estimación de la intención de movimiento sin necesidad de acumular el movimiento global por medio de un filtro Gaussiano y del uso de los fotogramas estabilizados además de los fotogramas originales. Nuestra propuesta usa un proceso de optimización offline para determinar el mínimo número de fotogramas que se pueden aplicar, en tiempo real, en el sistema

sin comprometer el desempeño inicial de la estabilización de vídeo offline. Este filtro se combinará con la señal de la acción de control, con el objetivo de eliminar los movimiento fantasma en el vídeo compensado.

3.4.1 Estimación de la intención de movimiento optimizado

Para minimizar el número de fotogramas requeridos en el proceso de estabilización de vídeo, se llevó a cabo una búsqueda exhaustiva mediante un algoritmo que iterativamente incrementa el número de fotogramas usados para estimar la intención de movimiento.

Para el proceso de optimización es necesario definir una métrica de evaluación del desempeño de la estabilización del vídeo. Mean Opinion Score o MOS es una métrica de evaluación subjetiva que se utiliza ampliamente en la evaluación de la calidad de compresión multimedia [98]. La otra posibilidad es el uso de métricas de evaluación como bounding boxes, líneas de referencia o secuencias sintéticas [99]. La fidelidad de la transformación entre fotogramas (ITF) [97] se emplea frecuentemente como método para medir la eficacia y el rendimiento de la estabilización de vídeo. Su expresión matemática es:

$$ITF = \frac{1}{N_f - 1} \sum_{k=1}^{N_f - 1} PSNR(k) \tag{3.6}$$

donde N_f es el número de fotogramas del vídeo y

$$PSNR(k) = 10\, log_{10} \frac{Ip_{MAX}}{MSE(k)} \tag{3.7}$$

es la relación de señal ruido de pico entre dos fotogramas consecutivos con

$$MSE(k) = \frac{1}{M \cdot N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \| Frame_k(i,j) - Frame_{k-1}(i,j) \|^2 \quad (3.8)$$

siendo el error cuadrático medio entre imágenes monocromáticas de dimensión $M \cdot N$ e Ip_{MAX} la intensidad de píxel máxima en el fotograma.

Los resultados obtenidos basados en la optimización de la métrica de evaluación objetivo ITF se presentan en la Figura 3.8.

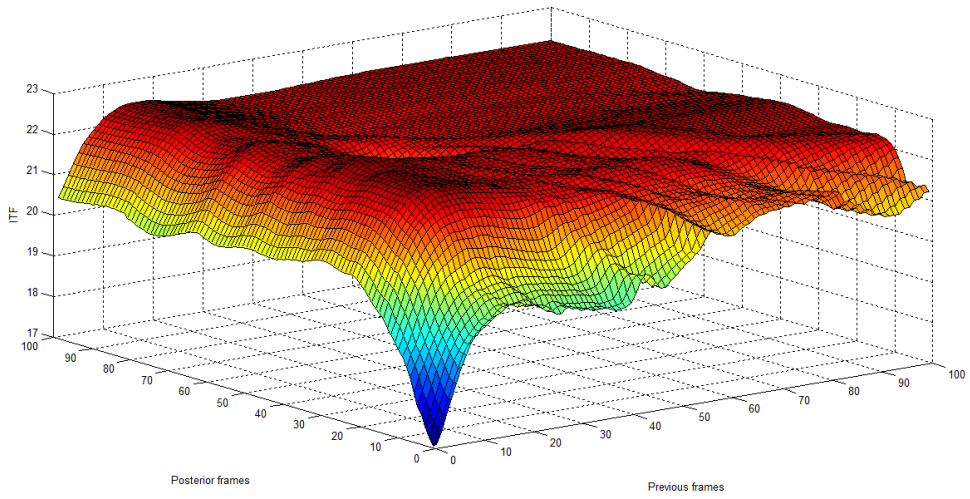

Figura 3.8. Minimización de la fidelidad de la transformación inter-fotograma (ITF).

En la Figura 3.8 se presentan 3 ejes. Uno corresponde a los fotogramas previos utilizados, el segundo a los fotogramas posteriores utilizados y el último al valor ITF. Experimentalmente se determinó que al estimar la intención de movimiento usando a) cuatro fotogramas previos y cuatro posteriores o b) únicamente seis

fotogramas previos, se obtiene una secuencia resultante de fotogramas con un ITF similar.

Nuestro trabajo se enfoca en aplicaciones en tiempo real, por lo que es importante analizar los efectos que las dos opciones generan respecto al costo computacional: a) Para el primer caso, usar cuatro fotogramas previos y cuatro fotogramas posteriores implica que el algoritmo será ejecutado cuatro fotogramas después de la inicialización de la secuencia de movimiento, y la secuencia estabilizada estará lista después de cuatro fotogramas. b) En el segundo caso, el algoritmo comienza a ejecutarse seis fotogramas luego de la inicialización de la secuencia de vídeo, es decir, un retardo de dos fotogramas respecto al primero, pero el resto de la secuencia se puede aplicar sin retardos adicionales. Considerando que la frecuencia de muestreo fue de 10 Hz para el sistema, un retardo de 0.4 segundos se introduce en el tiempo de cómputo total cuando se utiliza la primera opción. En nuestro algoritmo se optó por la segunda opción que solo depende de la información precedente.

3.4.2 Movimientos fantasma

Previos métodos de estabilización de vídeo han tenido buen desempeño eliminando movimientos indeseados en imágenes capturadas con dispositivos móviles y sistemas de dinámica compleja, pero cabe mencionar que estos algoritmos se han evaluado utilizando al ITF como función costo. Aunque el vídeo final alcanza un alto ITF, es decir un vídeo estable, el proceso de suavizado de movimiento genera movimientos fantasma.

Para aplicaciones de post-procesamiento, el principal objetivo es estabilizar el vídeo, donde los movimientos fantasma no representan un problema mayor. No obstante, para aplicaciones en tiempo real, el objetivo es obtener un vídeo estable con el mayor realismo de movimiento posible. En este sentido, es importante

disminuir la diferencia entre la intención de movimiento real y la estimada, preservando el rendimiento del ITF.

La raíz cuadrada del error cuadrático medio, o RMSE por sus siglas en inglés [100] se adopta con el objetivo de evaluar la confiabilidad del movimiento estimado respecto al movimiento observado. Un RMSE bajo implica que la intención de movimiento estimada es similar a la intención de movimiento real.

La métrica de evaluación objetivo, propuesta para su optimización, es la diferencia entre el movimiento global estimado y el movimiento observado, medida como un RMSE:

$$RMSE = \frac{1}{2F}\left(\sqrt{\sum_{j=0}^{F}\left(E_{x,j} - T_{x,j}\right)^2} + \sqrt{\sum_{i=0}^{F}\left(E_{y,i} - T_{y,i}\right)^2}\right) \quad (3.9)$$

donde $E_{x,j}$ y $E_{y,i}$ son los movimientos acumulados globales del fotograma j-ésimo, en x el eje-x y el eje-y, respectivamente, $T_{x,j}$ y $T_{y,i}$ son los movimientos observados del fotograma j-ésimo en el eje-x y en el eje-y, respectivamente, y F denota el número de fotogramas en la secuencia.

Existen dos fuentes de información para la estimación intención de movimiento ($T_{x,j}$ y $T_{y,i}$): los datos obtenidos desde la unidad de medida inercial a bordo del sistema (IMU) y la información de la acción de control. La elección depende de la exactitud del modelo. En nuestro algoritmo, la acción de control se emplea debido a que la información IMU se encuentra desincronizada respecto a la imagen. El movimiento observado se define como una combinación entre la acción de control y la señal de los parámetros de movimientos suavizados.

Nuestro algoritmo de estimación de movimiento (Algoritmo 3.2) usa la acción de control como compuerta lógica permitiendo la ejecución de filtros pasa-bajos solo

cuando una intención de movimiento tele-operada está presente. Nuestro algoritmo inserta una histéresis posterior a la ejecución de la acción de control. El objetivo de esta histéresis es que el sistema alcance su posición máxima (o mínima, de acuerdo a la señal de la acción de control) como efecto posterior a una nueva acción de control.

Algoritmo 3.2 Algoritmo de minimización de movimientos fantasma usando la acción de control

---

{U_i es la acción de control actual, E_{R_i} es el movimiento actual estimado sin movimientos fantasma, E_i es el movimiento actual estimado usando el filtro}

if $U_i \neq 0$ **then**

$$E_{R_i} = E_i$$

else if $\big((U_{i-1} > 0) \wedge (E_i > E_{i-1}) \vee (U_{i-1} < 0) \wedge (E_i < E_{i-1})\big)$

then

$$E_{R_i} = E_i$$

else

$$E_{R_i} = R_F \cdot (E_i - E_{i-1}) + E_{i-1}$$

end if

Se definió un parámetro de confiabilidad 0 < RF < 1. Un valor de RF cercano a uno implica alcanzar un valor de ITF alto, es decir, un vídeo de mayor estabilidad con

movimientos fantasma. En el caso contrario, usando un valor de RF cercano a cero se obtiene un vídeo de menor estabilidad sin movimientos fantasma.

3.5 Resultados y discusión

Esta sección se divide en tres partes: diseño experimental, desempeño de la estabilización de vídeo y comparativa con otros algoritmos.

3.5.1 Diseño experimental

El AR.Drone 1.0, un MAV, construido por la compañía francesa PARROT (Paris, Francia) se usó como una plataforma experimental por múltiples razones: bajo costo, ahorro energético, seguridad de vuelo y tamaño del vehículo. La metodología propuesta se implementó en un computador portátil con las siguientes características: Procesador Inter Core i7-2670QM, 2.20 GHz con Turbo Boost up, a 3.1 HZ y RAM 16.0 Gb. Las imágenes de cuatro diferentes escenarios se obtienen con una cámara a bordo (frecuencia de muestreo = 10 Hz) y se envían vía wifi al computador portátil para su procesamiento. Se grabó un vídeo con una cámara cenital para capturar el movimiento del robot volador, en el plano-xy, desde una perspectiva fija y con mayor objetividad. RMSE se selecciona como medida objetivo de confiabilidad de movimiento, comparando el movimiento estimado con el movimiento observado. Con el objetivo de obtener la posición se usa un seguidor basado en optical flow [101] y un método de calibración de cámara para la distorsión radial [102]. Luego de ello, el RMSE se calcula comparando el movimiento estimado con el observado (en la cámara cenital).

3.5.2 Desempeño en la estabilización de vídeo

Una percepción visual de las resultados obtenidos para ambientes experimentales se muestran en las Figuras 3.9, 3.10, 3.11 y 3.12. Los experimentos demuestran que nuestro enfoque basado en la estimación de la intención de movimiento es robusto ante la presencia de objetos cercanos, escenas con objetos en movimiento, y problemas comunes descritos en secciones pasadas de cámaras a bordo de vehículo aéreos de micro-escala durante vuelos internos.

Figura 3.9. Escena 1. Superior: Vídeo original. Inferior: Vídeo estabilizado.

Figura 3.10. Escena 2. Superior: Vídeo original. Inferior: Vídeo estabilizado.

Figura 3.11. Escena 3. Superior: Vídeo original. Inferior: Vídeo estabilizado.

Figura 3.12. Escena 4. Superior: Vídeo original. Inferior: Vídeo estabilizado.

Presencia de objetos cercanos

La presencia de objetos cercanos en las escenas representan uno de los principales problemas de la estabilización de vídeo. Esto se debe a que muchos de los puntos de interés se generan en torno a las regiones de la imagen en la que se encuentran ubicados los objetos. La compensación de la imagen se computa usando el movimiento de los objetos en lugar del movimiento de la escena. Nuestro proceso de búsqueda de correspondencias de puntos de interés se basa en el algoritmo RANSAC y la diferencia del nivel de gris entre fotogramas consecutivos como función costo. En consecuencia, el proceso de estimación de movimiento no se ejecuta en torno a los puntos de interés del objeto, sino respecto a la escena completa.

Escenas con objetos en movimiento

Los objetos en movimiento constituyen otro problema común. Algunos objetos con muchos puntos de interés ocasionan, durante la estimación del movimiento,

seguimientos indeseado de estos objetos. Una vez más, el proceso de búsqueda de correspondencias basado en RANSAC no solo se lleva a cabo con referencia a los objetos en movimiento sino a la imagen completa.

Problemas de cámaras a bordo de MAVs

Los desplazamientos considerables entre fotogramas consecutivos son un problema común en imágenes capturadas con cámaras a bordo y se deben principalmente a la dinámica compleja del MAV durante vuelos internos. En todos ellos, los cambios entre fotogramas consecutivos son importantes, produciendo un problema crítico en la estabilización de vídeo. En nuestro enfoque, la estimación de la intención de movimiento resuelve ese problema, y una desestimación previa de datos mayores que un umbral preestablecido, proporcionan de robustez adicional.

Movimientos fantasma

Independientemente del enfoque utilizado, la estimación de la intención de movimiento es una parte indispensable del proceso de estabilización de vídeo para evitar que el resultado sea una secuencia de imágenes estáticas. Los movimientos fantasma se generan por la eliminación de los movimientos de alta frecuencia durante esta parte del proceso.

En los algoritmos presentes en la literatura, este problema es recurrente y considerable como se puede observar en la Figura 3.14. El parámetro de movimiento escala del algoritmo L1-Optimal no es tan fiel, respecto al movimiento capturado con una cámara cenital fija desde un posición más objetiva, como el estimado por nuestra propuesta.

Nuestra propuesta elimina los movimientos fantasma utilizando una combinación del filtro pasa-bajos como estimador de la intención de movimiento,

en combinación con la acción de control. Este método reduce levemente el valor ITF.

3.5.3 Comparación con otros algoritmos

Nuestra propuesta se comparó con el método offline L1-Optimal [12], el cual está disponible en el editor de Youtube como una opción de estabilización de vídeo. Resultados en cuatro diferentes escenas se presentan en la Tabla 3.1 usando dos métricas de evaluación: ITF y RMSE.

El resultado obtenido muestra que nuestro algoritmo es comparable con el método L1-Optimal. El rendimiento de nuestro propuesta respecto a la métrica ITF es ligeramente menor que el L1-Optimal, lo que implica que el vídeo obtenido es un poco menos estable. No obstante, el rendimiento respecto a la métrica RMSE, que presenta la similitud entre el movimientos del vídeo estabilizado y el movimiento intencional, es mayor en nuestro enfoque.

Tabla 3.1. Métricas de evaluación.

Algoritmo	Métrica	Vídeo 1	Vídeo 2	Vídeo 3	Vídeo 4
Original	ITF (dB)	14.09	13.43	14.65	16.96
L1-Optimal		**19.62**	**19.57**	**20.16**	**20.24**
Smoothing		19.48	19.52	19.89	21.12
Subspace		19.58	19.59	20.12	20.91
L1-Optimal	RMSE	0.046	0.051	0.047	0.036
Smoothing		**0.028**	**0.023**	**0.029**	**0.017**
Subspace		0.049	0.053	0.048	0.034

Cabe señalar que la medida ITF se podría incrementar variando el factor de confiabilidad RF $\in$[0,1], pero RMSE inevitablemente incrementaría. Para aplicaciones de post-producción, el valor del RF puede ser igual a uno, pero en aplicaciones de control de movimiento basado en la información de la cámara, el realismo del movimiento es esencial, por lo que el factor de confiabilidad debería tender a cero. En la Figura 3.13, la escala observada se comparan gráficamente con la estimada usando la técnica L1-Optimal y nuestro enfoque.

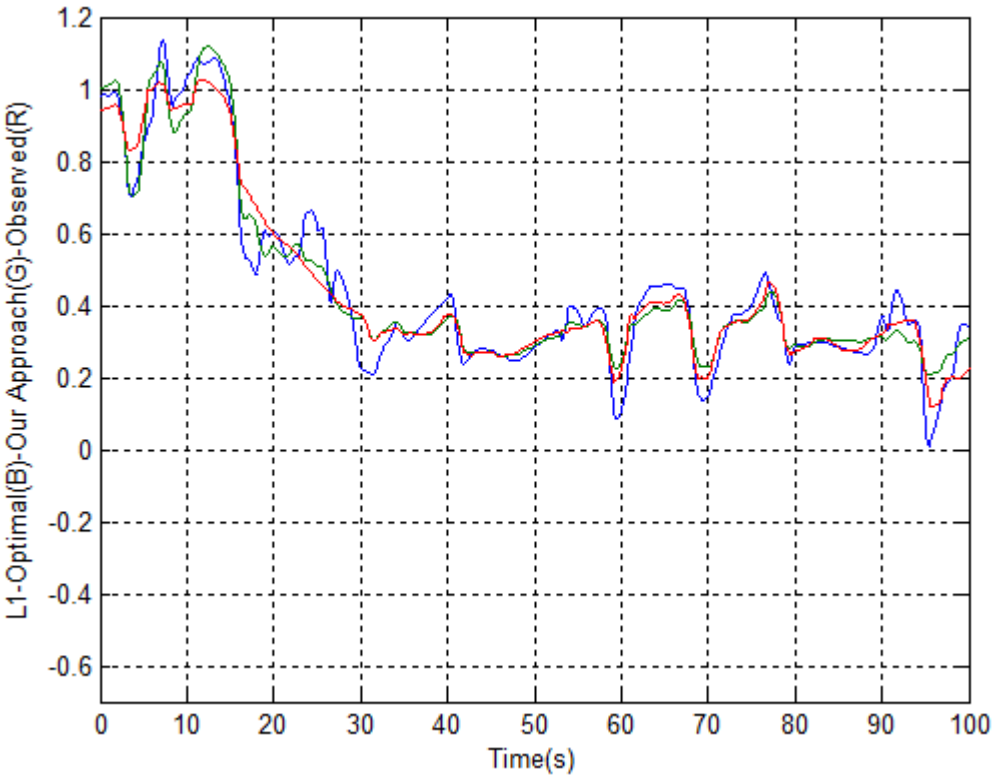

Figura 3.13. Comparación de escalas. L1-Optimal (azul), nuestro enfoque (verde), y el observado (rojo).

3.6 Conclusiones

Se constató experimentalmente que el filtro pasa-bajos posee un buen desempeño como algoritmo de estimación de la intención de movimiento, eliminando movimientos indeseados. Sin embargo, este método puede ser optimizado usando un menor número de fotogramas sin disminuir el valor de ITF. La frecuencia de corte depende de las características del modelo; por lo tanto, la información del sistema de captura implica una considerable contribución en una fase previa de configuración.

Los movimientos fantasma son fenómenos que aún no se han estudiado en la literatura sobre estabilización de vídeo, sin embargo, constituyen un punto fundamental en el control y teleoperación de sistemas de dinámica compleja como vehículos aéreos de micro-escala, donde el realismo en los movimientos podría significar la diferencia que prevenga un accidente. El factor de confiabilidad se adapta al propósito de la aplicación. Esta aplicación puede ser una post-producción con un valor alto de ITF y poca fidelidad al movimiento real, o la situación opuesta para estabilización de vídeo en tiempo real de sistemas teleoperados.

Capítulo 4: Intención de movimiento basada en modelo

Pocos algoritmos de estabilización presentes en la literatura pueden ser aplicados en tiempo real. Además, no discriminan entre movimientos intencionales del operador y los indeseados debidos a factores externos. En el presente capítulo, una nueva técnica es introducida para estabilización de vídeo en tiempo real con bajo costo computacional, sin generar movimiento falso o reducir la calidad de la secuencia de vídeo estabilizada. Nuestra propuesta se basa en el enfoque de [63] para estimar el movimiento intencional a partir de los parámetros y no de los puntos característicos. A diferencia de la propuesta en [63], se usa el filtro de Kalman y el modelo del MAV, donde las acciones intencionales de control están desacopladas de los movimientos no intencionales.

El modelo del MAV incluye la acción de control, solventando el problema de movimientos fantasma y, al mismo tiempo, minimizando el número de fotogramas previos que se utilizarán. El algoritmo depende únicamente del último fotograma de la secuencia de vídeo, por lo que se puede aplicar en tiempo real sin retardos o reducción del desempeño.

El modelo del MAV se estima en offline a partir de datos recolectados en vuelos internos, pero nuestro algoritmo propuesto se aplica en tiempo real con base en el mencionado modelo. Para el modelado en vuelos externos se debe tomar en cuenta movimientos indeseados adicionales como turbulencia o viento, lo que queda fuera del alcance de la propuesta.

El resto del capítulo ha sido organizado de la siguiente forma: El modelado del MAV es explicado en las tres primeras secciones. Luego, en la Sección 4.4, se introduce una nueva técnica de estimación de la intención de movimiento basada en el modelo del MAV que incluye la entrada de la acción de control. Resultados experimentales son presentados en la Sección 4.5. Finalmente, se brindan algunas conclusiones y líneas futuras en la Sección 4.6.

En la Figura 4.1 se presenta nuestro algoritmo de estabilización de vídeo completo.

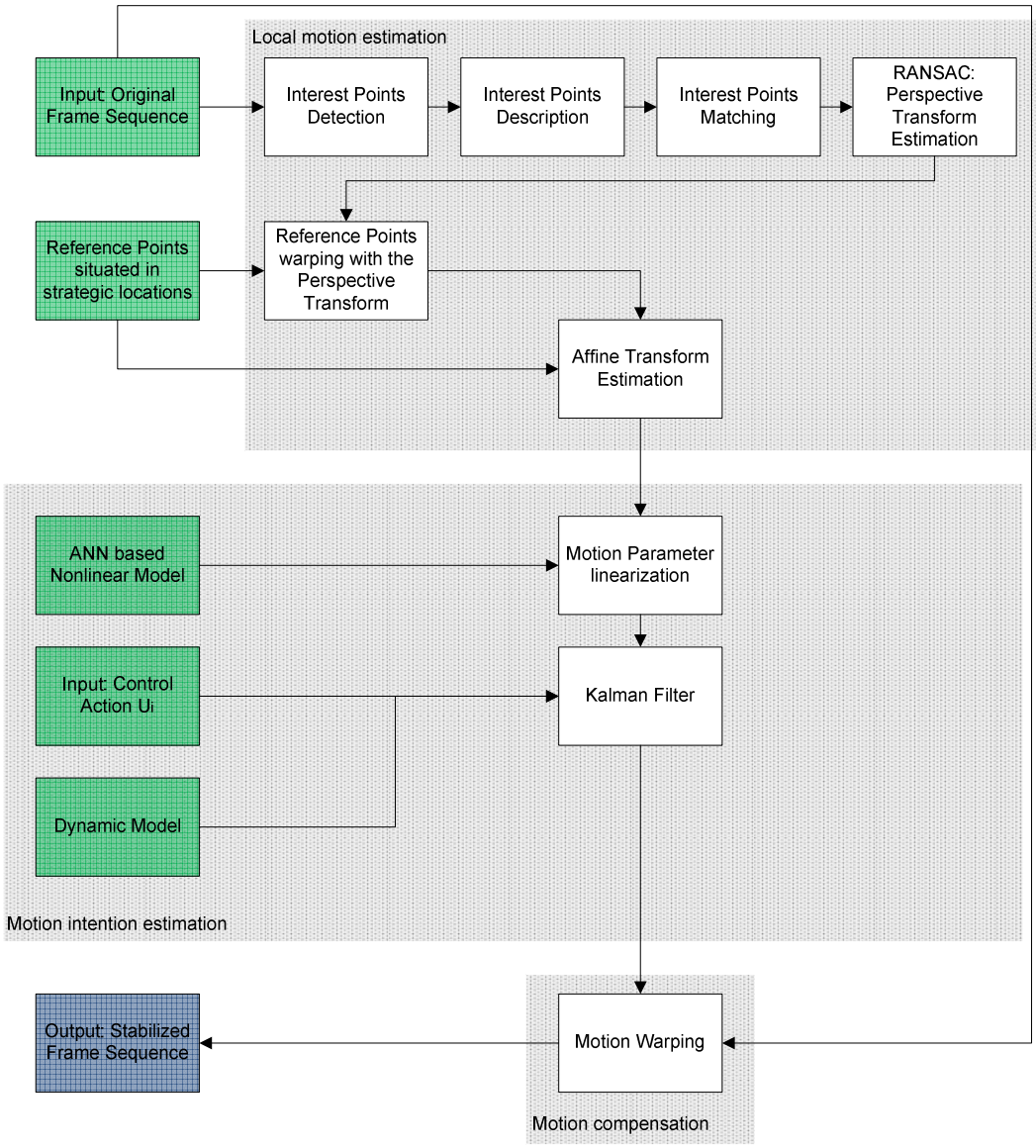

Figura 4.1. Diagrama de flujo. Propuesta para estabilización de vídeo basada en el modelo.

4.1 Estimación de la intención de movimiento basada en el modelo

Nuestro enfoque basado en una combinación de transformaciones geométricas obtiene una estimación de movimiento inter-fotograma con un alto rendimiento

como estabilizador en escenas estáticas [94], [95]. Sin embargo, nuestro objetivo es conseguir una estabilización robusta de vídeo en tiempo real para micro vehículos aéreos.

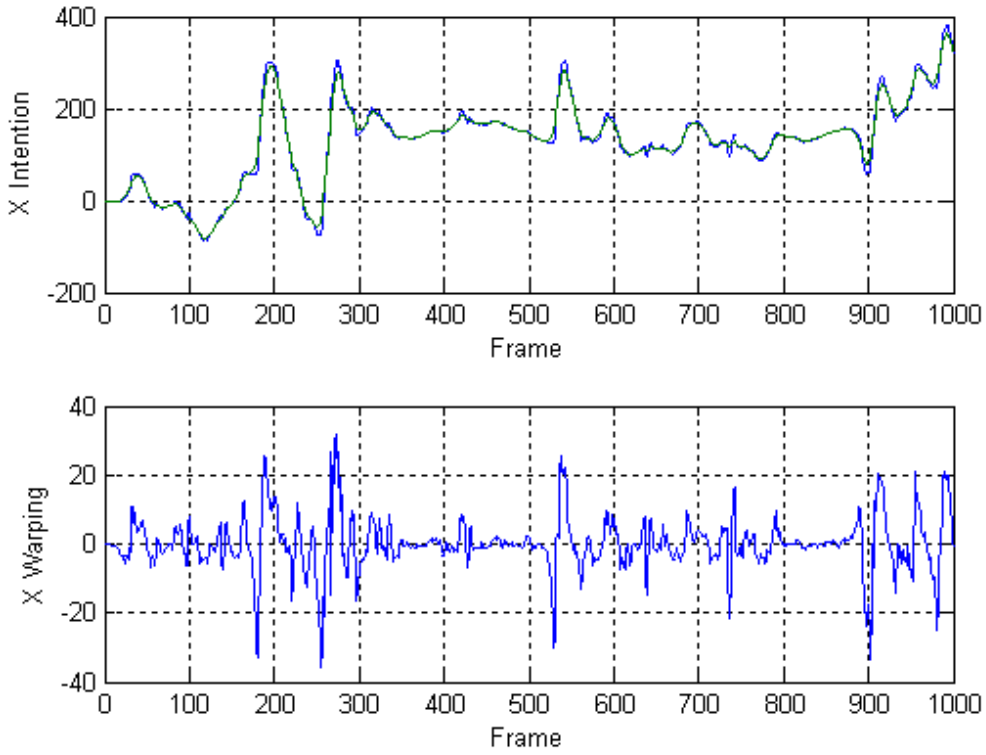

Figura 4.2. Superior: Señal de la intención de movimiento (verde), parámetro de movimiento acumulado (azul). Inferior: Diferencia entre la señal de la intención de movimiento y la señal del parámetro acumulado.

Durante vuelos tele-operados, el campo visual de la cámara a bordo se encuentra en continuo desplazamiento y algunos movimientos del dispositivo de captura no deberían eliminarse, sino compensarse suavemente, generando un vídeo estable en lugar de una escena estática. Los movimientos intencionales de la cámara deben ser estimados y eliminados de los parámetros de movimiento acumulados, obteniendo un señal de alta frecuencia. Se usa esta señal resultante

para compensar vibraciones y al mismo tiempo mantener el movimiento intencional. La imagen superior en la Figura 4.2 muestra el parámetro de movimiento acumulado (azul) y el movimiento intencional (verde). La diferencia entre las señales (imagen inferior en la Figura 4.2) se utiliza para compensar la secuencia entera.

Existen distintos algoritmos de estabilización de vídeo, como se mencionó en la Sección 1, que usan métodos de suavizado para la estimación de la intención de movimiento. En un artículo reciente [63], una propuesta nueva para la estimación de la intención de movimiento se introdujo, la cual se basa en un filtro Butterworth [96] de segundo orden. Esta técnica permite compensar señales de alta frecuencia de los parámetros de movimiento acumulado sin disminuir la calidad del vídeo o generar señales de alta frecuencia de movimiento acumulado

A pesar de los avances significativos mostrados por los algoritmos en la literatura como estabilizadores de vídeo, el uso de cualquier filtro siempre genera un retardo en la salida, y el filtro Butterworth de segundo orden no es la excepción. Con el objetivo de evitar el uso de una técnica de suavizado que adicione un retardo indeseado en el proceso de estabilización de vídeo, se propone el uso de un modelo matemático que relacione la acción de control con los parámetros de movimiento. Este modelo puede ser obtenido fuera de línea a través de experimentación con el MAV y su aplicación en la arquitectura del algoritmo de estabilización de vídeo en tiempo real no representa mayor problema.

4.1.1 Estimación del modelo del MAV

La plataforma utilizada en la experimentación es el AR.Drone 2.0, seleccionada por múltiples razones:

- Bajo costo: menos de 260 euros actualmente.
- Bajo consumo de energía.

- Seguridad de vuelo.
- Dimensiones físicas del vehículo.

El AR.Drone puede ser controlado con un dispositivo móvil como un smartphone o una tablet con sistema operativo iOS o Android. Adicionalmente, Parrot ha abierto el SDK (kit de desarrollo de software por sus siglas en inglés de Software Develop Kit) para los sistemas operativos Linux, Mac y Windows, por lo que puede ser controlado con un computador portátil o de escritorio. El sistema de control del drone permite manipular cuatro diferentes acciones de control: pitch, roll, yaw y altitud.

Para llevar a cabo el modelado del robot aéreo, se recopilaron los datos de la IMU (unidad inercial de medida por sus siglas en inglés) del AR.Drone y sus acciones de control. El modelo directo se estimó considerando como entrada las acciones de control, y como salida la posiciones y velocidades del MAV (la experimentación completa y los resultados asociados pueden revisarse en [103]). Finalmente nuestro interés es la estimación del modelo que relaciona las acciones de control con los parámetros de movimiento en la imagen, basados en el enfoque de [103].

4.1.2 Hipótesis en el modelo

Con el objetivo de resolver algunos problemas de modelado, se consideraron dos hipótesis [103] basadas en los datos:

- Los modelos están desacoplados y definidos por la siguiente relación: La escala depende del control pitch, la translación en el eje y depende del control de altitud, y la traslación en el eje x depende del control roll y yaw.

- La relación entre la acción de control y los parámetros de movimiento es un modelo no lineal en estado estable combinado con un modelo lineal en estado transitorio.

En la primera hipótesis, el parámetro de movimiento ángulo no se considera debido a que se está estimando la intención de movimiento, en cuya dinámica éste parámetro no tiene incidencia. De la misma forma, existen movimientos en el eje y que dependen del control pitch, y que en el eje x depende del control roll. No obstante, en ambos casos los movimientos son indeseados. La Tabla 4.1 muestra la intencionalidad de movimiento de los parámetros que dependen de cada acción de control.

En la Figura 4.3, se puede apreciar una fuerte relación entre el ángulo pitch con la velocidad en x, así como entre el ángulo roll y la velocidad en y.

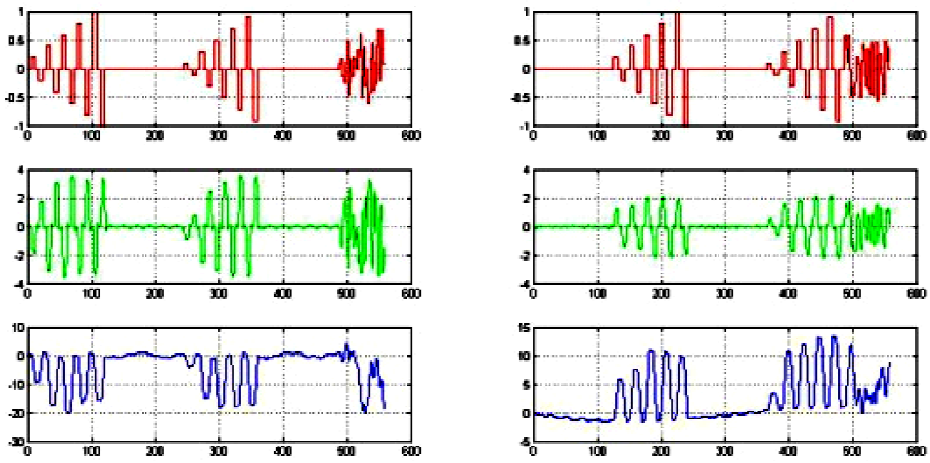

Figura 4.3. Derecha: pitch (superior), velocidad x, (central), posición x (inferior). Izquierda: roll (superior), velocidad y, (central), posición y (inferior).

Tabla 4.1. Movimientos intencionales e indeseados de los parámetros debidos al control.

Acción	Ángulo	Escala	X	Y
Roll	No deseado	No deseado	**Intencional**	No deseado
Pitch	No deseado	**Intencional**	No deseado	No deseado
Yaw	No deseado	No deseado	**Intencional**	No deseado
Altitud	No deseado	No deseado	No deseado	**Intencional**

Considerando la segunda hipótesis, el proceso de modelado se separó en dos partes:

- Estimación del modelo en estado estable no lineal.
- Estimación del modelo en estado transitorio lineal.

4.2 Estimación del modelo en estado estable no lineal basada en redes neuronales

La no linealidad en el modelo en estado estable se debe a un efecto de saturación en el sistema de control del ángulo. Para ángulos mayores que el límite de saturación, la velocidad es constante, pese a que los datos muestran disminuciones ocasionales que no pueden ser invertidas. Uno de los parámetros de configuración del AR.Drone es el ángulo máximo para cada rotación. En este sentido, es importante explicar que el modelo no lineal solo es necesario para aplicaciones donde la aceleración de la acción de control no es constante.

Es imprescindible definir los valores estacionarios aproximados de las velocidades para cada valor de ángulo como entrada. En la Tabla 4.2 se encuentran especificados dichos valores.

Tabla 4.2. Ángulos vs valores estacionarios de velocidades.

Pitch	Vx (m/s)	Roll	Vy (m/s)
-1.00	2.70	-1.00	-1.91
-0.90	2.80	-0.90	-2.14
-0.80	2.85	-0.80	-2.06
-0.70	3.29	-0.70	-2.10
-0.60	3.29	-0.60	-2.07
-0.50	3.03	-0.50	-1.87
-0.40	3.01	-0.40	-1.50
-0.30	2.47	-0.30	-1.58
-0.20	1.78	-0.20	-1.26
-0.10	0.82	-0.10	-0.59
0.00	0.00	0.00	0.00
0.10	-0.82	0.10	0.59
0.20	-1.78	0.20	1.26
0.30	-2.47	0.30	1.58
0.40	-3.01	0.40	1.50
0.50	-3.03	0.50	1.87
0.60	-3.29	0.60	2.07
0.70	-3.29	0.70	2.10
0.80	-2.85	0.80	2.06
0.90	-2.80	0.90	2.14
1.00	-2.70	1.00	1.91

En [103], la parte no lineal del modelo es estimada como un sistema en estado estable, usando un polinomio de quinto grado que relaciona la acción de control con los parámetros de movimiento.

Para compensar esta no linealidad es necesario calcular el polinomio inverso, polinomio en cuya estimación nuevamente se hace uso del ajuste polinomial, utilizando como datos de entrada en este caso los valores estimados de velocidad estacionaria y como datos de salida la referencia Pitch.

Tabla 4.3. Valores estacionarios de velocidades vs ángulos.

Vx (m/s)	Pitch	Vy (m/s)	Roll
		-2.108	-0.70
3.287	-0.60	-2.086	-0.60
3.215	-0.50	-1.969	-0.50
2.926	-0.40	-1.743	-0.40
2.42	-0.30	-1.414	-0.30
1.728	-0.20	-0.9963	-0.20
0.8995	-0.10	-0.5148	-0.10
0.00	0.00	0.00	0.00
-0.8995	0.10	0.5148	0.10
-1.728	0.20	0.9963	0.20
-2.42	0.30	1.414	0.30
-2.926	0.40	1.743	0.40
-3.215	0.50	1.969	0.50
-3.287	0.60	2.086	0.60
		2.108	0.70

Este polinomio es no invertible, ya que para un mismo valor de Pitch tiene distintos valores de velocidad estacionaria, por lo cual truncaremos los valores entre -0.6 y 0.6 para pitch y -0.7 y 0.7 para roll (Tabla 4.3) de forma que se puede trabajar en la zona biyectiva. Estos valores corresponden al máximo y mínimo valor de velocidad estacionaria alcanzable para en x e y respectivamente.

El ajuste se consigue con un polinomio de grado 5:

$$P(q) = aq^5 + bq^4 + cq^3 + dq^2 + eq + f \tag{4.1}$$

Para la mejora del modelo entrada-salida del MAV, se usa una red neuronal feedforward constituida por una capa oculta con 5 neuronas. La red neuronal artificial entrenada permite reducir el error cuadrático medio (RMSE). Se ha obtenido un RMSE = 0.0251 con la red neuronal, el cual es considerablemente menor al obtenido usando la aproximación polinomial (RMSE = 0.2072).

En la Figura 4.4, el polinomio y los valores estacionarios aproximados son graficados y comparados.

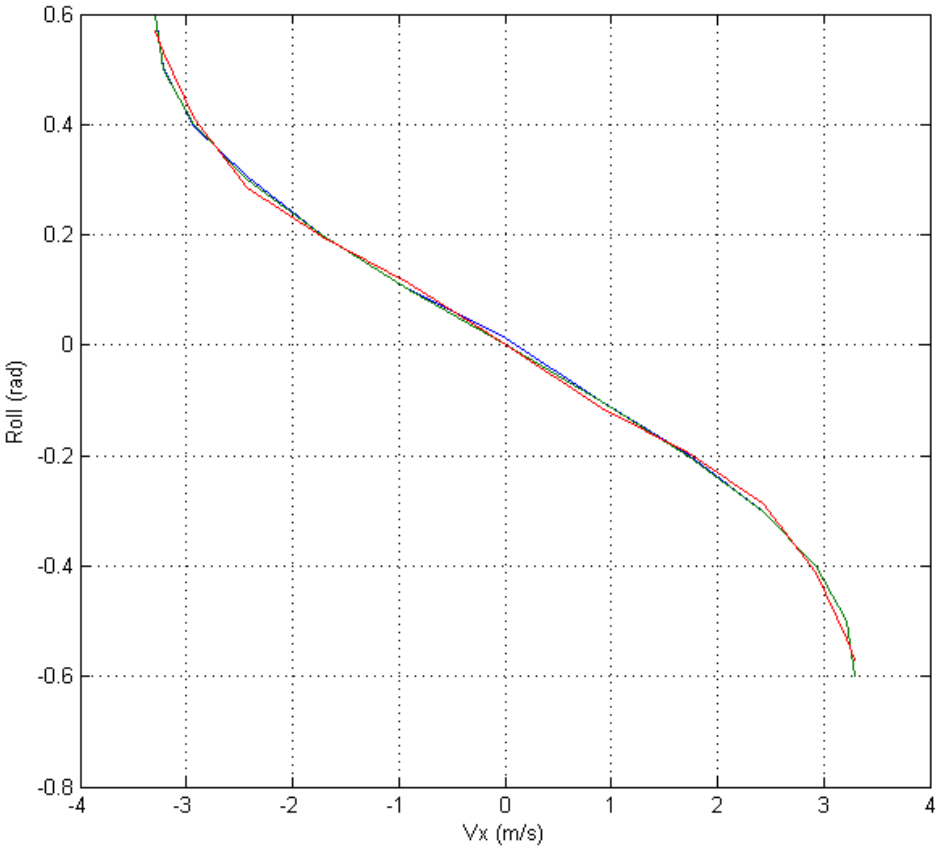

Figura 4.4. Estimación del modelo no lineal en estado estable. Datos reales (verde), función polinomial (rojo) RMSE = 0.2072, red neuronal (azul) RMSE = 0.0251.

4.3 Identificación del modelo lineal en estado transitorio

Una vez que la no linealidad se estima, se identifica el modelo lineal que relaciona la acción de control con los parámetros de movimiento para una velocidad constante. En la Figura 4.5 se presenta un gráfico con los datos de acción de control para roll (gráfico superior) y los parámetros de movimiento filtrado para la traslación x (gráfico inferior).

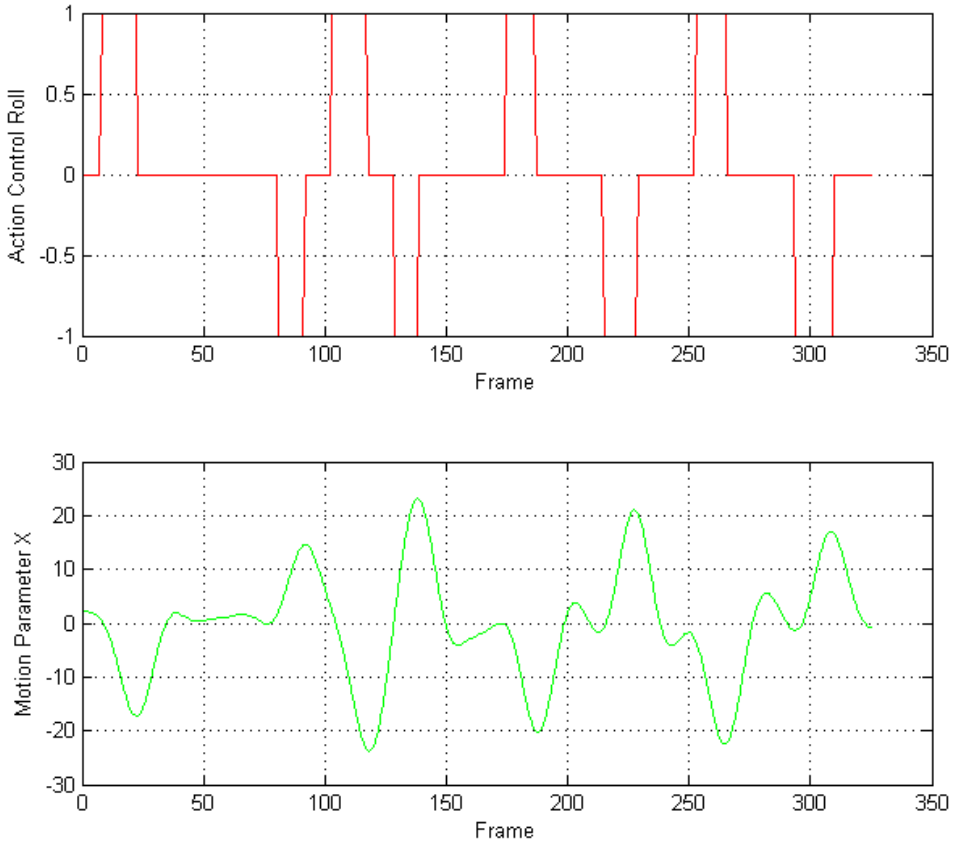

Figura 4.5. Identificación del modelo en estado transitorio. Superior: Entrada. Inferior: Salida.

Usando una herramienta de identificación de modelos para sistemas lineales en estado transitorio, se obtienen los parámetros de la función de transferencia,

$$G(s) = \frac{K_p}{1+Tp*s} * exp^{-(Td*s)} \quad (4.2)$$

con ganancia K_p, constante de tiempo T_p y retardo T_d.

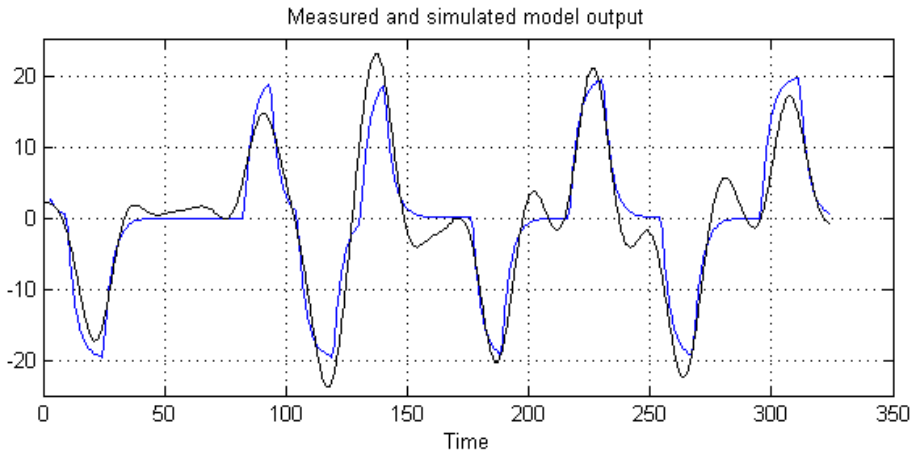

Figura 4.6. Resultados de la identificación del modelo en estado transitorio lineal.

En la Figura 4.6, se presenta la posición filtrada (negro) y el movimiento estimado con base en el modelo (azul). El movimiento estimado, mediante el modelo, se acopla mejor al movimiento real que el estimado en offline usando un filtro.

Tabla 4.4. Rendimiento del modelo

Variable	GPP: 0.90108		
	Valor (m)	**Calificación**	**Porcentaje**
Dist. x Media	0.06272	AD	31.3598%
Dist. y Media	0.016007	AD	8.0035%
Dist. x Máx.	0.99527	D	38.0977%
Dist. y Máx.	0.21274	AD	42.5478%

Para la simulación y evaluación del modelo estimado se utiliza las herramientas facilitadas por el Comité español de automática CEA [104]. El modelo obtenido permitió ganar el primer lugar en la fase 1 del concurso de ingeniería de control

2013 del CEA, y el segundo lugar en la fase 2. En la Tabla 4.4 se encuentran especificados los resultados de la evaluación del modelo estimado. Las siglas AD hacen referencia a Altamente-Deseable, mientras que D significa Deseable. Los criterios de evaluación se explican con detalle en [104].

4.4 Filtro de Kalman

En la literatura se pueden encontrar algoritmos que usan el filtro de Kalman para la estimación de la intención de movimientos. Sin embargo, estos algoritmos emplean el filtro de Kalman para el seguimiento de puntos característicos como paso previo a la estimación de los parámetros de movimiento.

Nuestro enfoque utiliza el filtro de Kalman como técnica de suavizado del movimiento. El filtro de Kalman es aplicado luego de calcular los parámetros de movimiento y se fundamenta en el modelo matemático del MAV. El modelo nuevamente es dividido en dos partes: un modelo no lineal en estado estable y un modelo lineal en estado transitorio.

Estimando la inversa del modelo no lineal en estado estable, y aplicándolo a la entrada del sistema, se obtiene la entrada que se usa en el modelo lineal en estado transitorio. En el filtro de Kalman se usa la representación en espacio de estados del modelo lineal en estado transitorio. Por consiguiente, la función de transferencia discreta se representa como:

$$x_k = Ax_{k-1} + Bu_{k-1}$$

$$z_k = Cx_{k-1} + D \qquad (4.3)$$

Muchos de los algoritmos de estabilización de vídeo que usan el filtro de Kalman no consideran la entrada u_{k-1}. Nuestro algoritmo se basa en la entrada acción de control para la eliminación de los movimientos fantasma.

Una de la ventajas del filtro de Kalman es que puede ser aplicado en tiempo real porque depende únicamente del último fotograma. En la Figura 4.7 se puede observar una comparación gráfica de los parámetros acumulados de traslación en x, movimiento intencional en x basado en el filtro pasa-bajos (usando seis fotogramas previos) y el movimiento intencional en x basado en el filtro de Kalman.

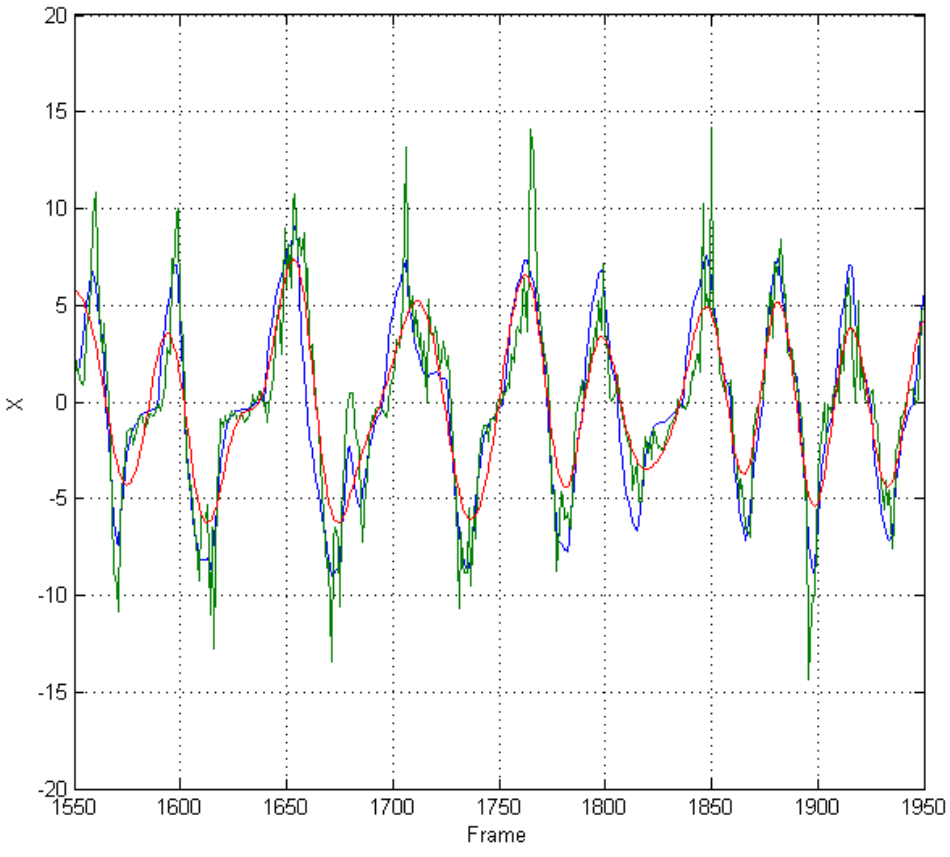

Figura 4.7. Movimiento intencional. Parámetro acumulado de traslación en x (verde), filtro pasa-bajos (rojo), filtro de Kalman (azul).

4.5 Resultados y discusión

Se llevó a cabo experimentos empleando un MAV con una cámara a bordo en cuatro diferentes escenarios, todos ellos en ambientes internos. El MAV que se utilizó es el AR.Drone, descrito en la Sección 4.1.1. Nuestro algoritmo de estabilización de vídeo es implementado en la estación de tierra, un computador portátil con un Procesador Core i7-2670QM 2.20GHz, Turbo Boost up a 3.1GHz y RAM 16.0 Gb y RAM 16 Gb. Se usó ROS (Robot Operative System) para comunicar la estación de tierra con el MAV. La cámara a bordo tiene una resolución de 1280x720 (720p), y es capaz de grabar 30 fotogramas monoculares por segundo, es decir una frecuencia de muestreo de 30 Hz.

4.5.1 Métricas de evaluación

En la literatura, los algoritmos de estabilización de vídeo usan métricas tanto subjetivas (Mean Opinion Score o MOS [98]) como objetivas (bounding boxes, líneas de referencia y secuencias sintéticas [99]) para evaluar el desempeño de sus métodos. Centrándose en la calidad del vídeo estabilizado final, se recurrió a la Inter-frame Transformation Fidelity (ITF) [97], una métrica de evaluación de eficiencia y desempeño ampliamente utilizada,

$$ITF = \frac{1}{N_f - 1} \sum_{k=1}^{N_f - 1} PSNR(t) \tag{4.4}$$

donde N_f es el número de fotogramas del vídeo y

$$PSNR(k) = 10\, log_{10} \frac{Ip_{MAX}{}^2}{MSE(k)} \tag{4.5}$$

es la relación de señal a ruido de pico entre dos fotogramas consecutivos, donde Ip_{MAX} es la máxima intensidad de píxel en el fotograma y MSE es el error cuadrático medio mencionado anteriormente.

Enfocándose en el realismo del movimiento del vídeo estabilizado final, usamos el error cuadrático medio (RMSE) [100],

$$RMSE = \frac{1}{2N_f}\left(\sqrt{\sum_{j=0}^{N_f}\left(E_{x,j} - T_{x,j}\right)^2} + \sqrt{\sum_{i=0}^{N_f}\left(E_{y,i} - T_{y,i}\right)^2}\right) \quad (4.6)$$

donde $E_{x,t}$, $E_{y,t}$ son los movimientos estimados, y $T_{y,t}$, $T_{y,t}$ son los observados en los ejes para el fotograma t-ésimo. N_f denota el número de fotograma en la secuencia. RMSE evalúa la diferencia entre el movimiento estimado a partir del vídeo estabilizado y el movimiento real del robot aéreo en el plano-xy. Un RMSE bajo implica una intención de movimiento estimada con una mayor similitud al movimiento real.

Un seguidor basado en flujo óptico [101] y un algoritmo de calibración de cámara para la distorsión radial [102], se utilizan para calcular el movimiento real a partir del vídeo capturado en un plano cenital.

4.5.2 Comparación con otros algoritmos

Nuestro enfoque se comparó con tres algoritmos de la literatura:

- El método fuera de línea L1-Optimal [12], aplicado en el YouTube Editor como una opción de estabilización de vídeo.

- Nuestro algoritmo anterior basado en un filtro pasa-bajos como técnica de suavizado de movimiento [63].
- Subspace vídeo stabilization [11], utilizado en el software comercial Adobe After Effects.

La evaluación de desempeño de estos enfoques de estabilización de vídeo, enfocada en el ITF y RMAS, se llevó a cabo usando cada técnica para estabilizar diferentes vídeos. Se utilizó los siguientes tipos de vídeos:

- Cuatro vídeos sin objetos en movimiento (30fps).
- Cuatro vídeos sin objetos en movimiento (10fps).
- Cuatro vídeos con objetos en movimiento (30fps).

Tabla 4.5. Métricas de evaluación. Vídeos sin objetos en movimiento.

Algoritmo	Métrica	Vídeo 1	Vídeo 2	Vídeo 3	Vídeo 4
Original	ITF (dB)	14.09	13.43	14.65	16.96
MAV-Model		19.49	19.55	19.89	21.20
L1-Optimal		**19.62**	**19.57**	**20.16**	**20.24**
Smoothing		19.48	19.52	19.89	21.12
Subspace		19.58	19.59	20.12	20.91
MAV-Model	RMSE	**0.021**	**0.018**	**0.024**	**0.015**
L1-Optimal		0.046	0.051	0.047	0.036
Smoothing		0.028	0.023	0.029	0.017
Subspace		0.049	0.053	0.048	0.034
MAV-Model	TIEMPO	0.035	0.032	0.034	0.038

Figura 4.8. Vídeos sin objetos en movimientos.

En la Tabla 4.5 se presentan resultados de cuatro vídeos grabados en escenarios sin objetos en movimiento, y estabilizados con cuatro distintos métodos, incluyendo nuestro algoritmo. Adicionalmente, se presenta el tiempo computacional por fotograma requerido por nuestro enfoque, pero el tiempo depende del número de pares de puntos emparejados usados por RANSAC y la resolución de cada fotograma. Se redimensionó todos los fotogramas a 640x360 y se usó 50 pares de puntos sin disminuir el ITF o el RMSE.

En la Figura 4.8, se pueden ver fotogramas de los vídeos 1-4 sin objetos en movimiento. Los resultados evidencian que nuestro enfoque, aplicado en tiempo real, alcanza valores de ITF tan altos como los obtenidos usando otros enfoque de la literatura aplicados fuera de línea.

Adicionalmente, el RMSE de nuestro algoritmo es menor que el obtenido por los otros algoritmos gracias a que los movimientos fantasma no son generados, es decir, el movimiento del vídeo procesado con nuestra técnica posee mayor realismo

sin disminuir la estabilidad del vídeo. El algoritmo publicado en [63] también es capaz de compensar la imagen sin generar movimientos fantasma, sin embargo los seis últimos fotogramas de la secuencia de vídeo son requeridos para llevar a cabo la estabilización. Nuestro enfoque actual depende solo del último fotograma, y el tiempo computacional presentado en la Tabla 4.5 muestra que no existe inconveniente para aplicar nuestro algoritmo en tiempo real para estabilizar vídeos a 30fps.

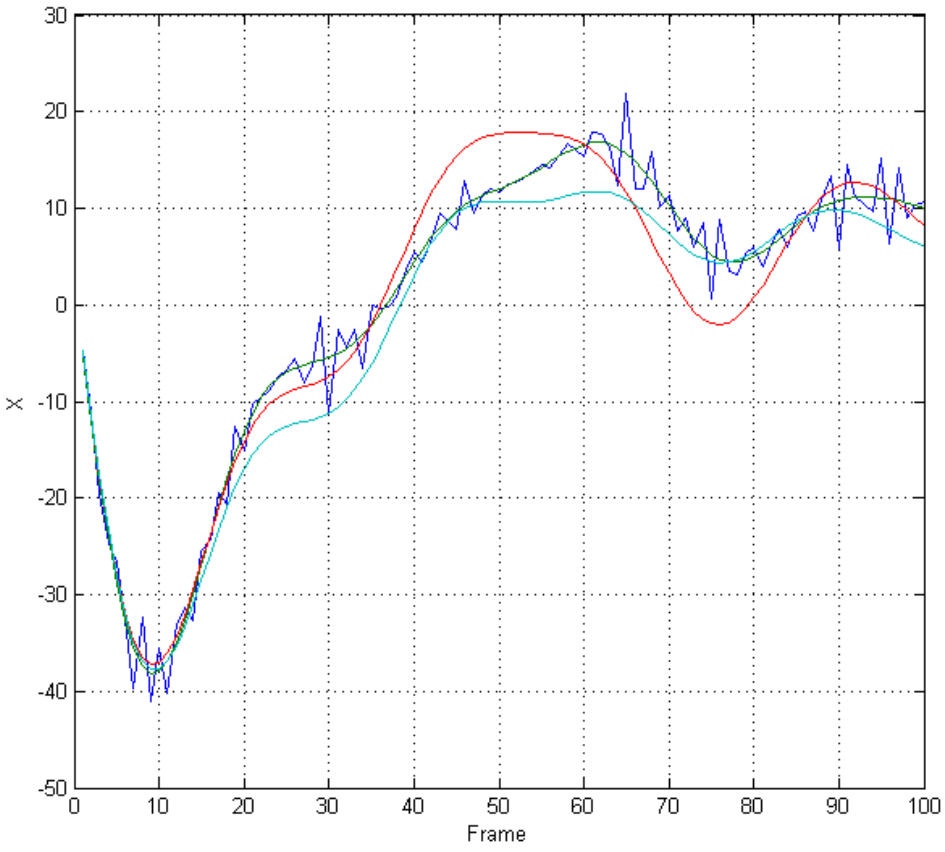

Figura 4.9. Comparación de la traslación X. L1-Optimal (celeste), Subspace (rojo), Nuestro enfoque (verde), Observado (azul).

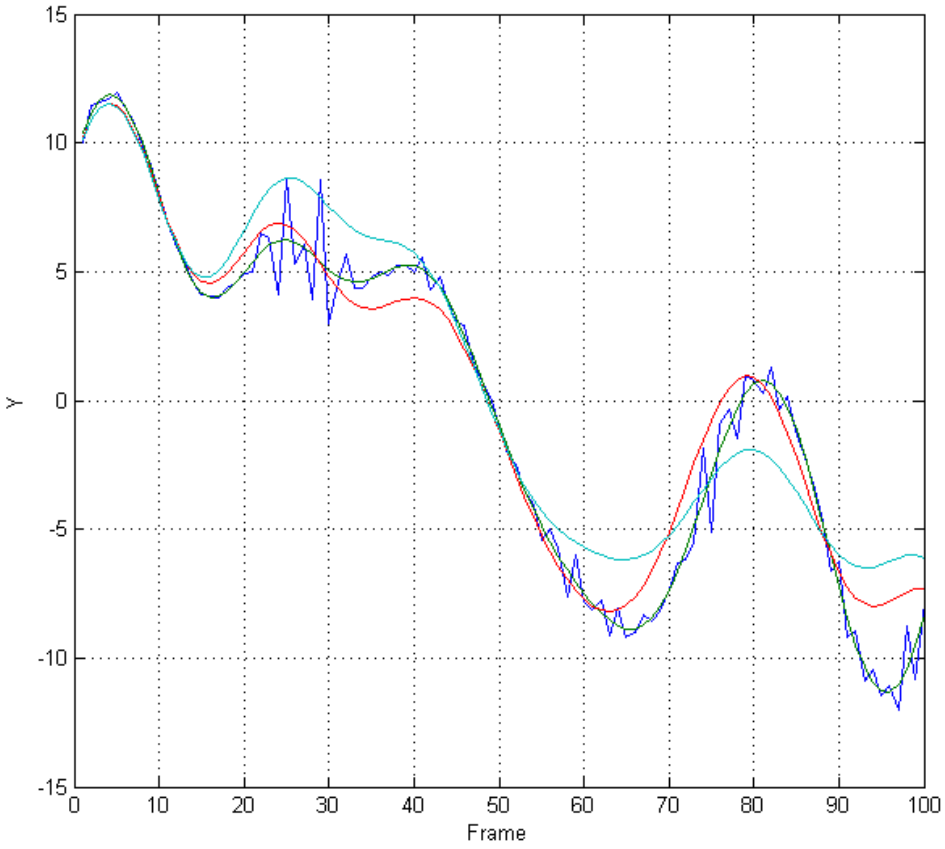

Figura 4.10. Comparación de la traslación Y. L1-Optimal (celeste), Subspace (rojo), Nuestro enfoque (verde), Observado (azul).

El efecto de los movimientos fantasma es gráficamente comparado en la Figura 4.9 y 4.10 entre los algoritmos L1-Optimal, Subspace y nuestro enfoque. Nuestro algoritmo reduce los movimientos fantasma en tiempo real con eficiencia similar a [63].

Tabla 4.6. Métricas de evaluación. Vídeos sin objetos en movimiento.

Algoritmo	Métrica	Vídeo 1	Vídeo 2	Vídeo 3	Vídeo 4
Original	ITF (dB)	12.27	12.12	12.28	13.43
MAV-Model		17.47	17.08	17.96	18.20
L1-Optimal		**17.64**	**17.42**	**17.15**	**18.22**
Smoothing		17.40	17.07	17.92	18.14
Subspace		17.57	17.39	17.11	18.12
MAV-Model	RMSE	**0.022**	**0.020**	**0.023**	**0.019**
L1-Optimal		0.057	0.059	0.054	0.046
Smoothing		0.023	**0.020**	0.025	**0.019**
Subspace		0.057	0.060	0.058	0.049

Figura 4.11. Vídeos con objetos en movimientos.

Tabla 4.7. Métricas de evaluación. Vídeos con objetos en movimiento

Algoritmo	Métrica	Vídeo 1	Vídeo 2	Vídeo 3	Vídeo 4
Original	ITF (dB)	12.46	12.27	12.82	14.48
MAV-Model		17.31	17.03	17.95	18.43
L1-Optimal		**17.49**	**17.44**	**17.94**	**18.47**
Smoothing		17.21	16.96	17.90	18.34
Subspace		17.42	17.39	17.51	18.40
MAV-Model	RMSE	0.026	**0.022**	**0.025**	**0.018**
L1-Optimal		0.060	0.057	0.059	0.046
Smoothing		**0.024**	**0.022**	0.027	**0.018**
Subspace		0.059	0.058	0.063	0.047

El enfoque presentado en este capítulo es robusto ante vídeos de baja frecuencia. En la Tabla 4.6 y 4.7, se presentan resultados experimentales de los cuatro últimos vídeos capturados a 10fps.

De igual forma, el algoritmo mantiene su buen desempeño ante la presencia de objetos en movimiento. La Tabla 4.7 corresponde a resultados obtenidos a partir de vídeos en escenarios con objetos en movimiento. En la Figura 4.11, se pueden ver fotogramas de los vídeos 1-4 con objetos en movimiento.

L1-Optimal y Subspace son dos de los mejores algoritmos de estabilización de vídeo, y son aplicados fuera de línea en dos de los más famosos softwares de edición de vídeo. Nuestro algoritmo es capaz de trabajar en tiempo real mostrando una robustez, ante objetos en movimiento y frecuencia baja (Tabla 4.8 y Tabla 4.9), tan buena como la obtenida por los algoritmos L1-Optimal y Subspace.

4.6 Conclusiones

En este capítulo, se presentó un nuevo algoritmo de estabilización de vídeo capaz de ser aplicado en tiempo real, robusto a escenas con objetos en desplazamiento y movimientos de dinámica compleja generados por cámaras a bordo de micro vehículos aéreos. Nuestra propuesta puede computar una secuencia estable de vídeo sin generar movimientos fantasma para compensar el movimiento involuntario. Este algoritmo proporciona una herramienta confiable para sistemas teleoperados.

La técnica está basada en la estimación del modelo del MAV, usando una red neuronal feedforward, que incluye la acción de control y la aplicación de este modelo en el filtro de Kalman para el suavizado de movimiento sin generar movimientos falsos. Los algoritmos de la literatura son suficientes para aplicaciones de post-procesamiento, pero nuestro objetivo se enfoca en tareas autónomas y de teleoperación de MAVs.

Nuestro algoritmo obtiene un alto desempeño para vuelos internos. En el futuro, se planea evaluar el método de estabilización de vídeo mediante la comparación del desempeño de personas sin experiencia en la teleoperación del MAV utilizando la imagen estabilizada respecto a su desempeño usando la imagen original.

Capítulo 5: Conclusiones e impacto, líneas futuras, publicaciónes y financiación

5.1 Conclusiones e impacto

El algoritmo desarrollado permite obtener una imagen estable de la cámara frontal del drone de micro-escala, sin eliminar movimientos intencionales significativos o generar movimientos inexistentes.

Cabe mencionar que el algoritmo está siendo ejecutado en tiempo real en un computador portátil con un procesador Intel Core i5, y es capaz de ser

implementador a 30 fotogramas de 720p por segundo. Todo el proceso de estabilización de vídeo se consigue por software sin necesidad de dispositivos mecánicos externos que aumente la carga del drone de micro-escala, lo que implica un menor consumo energético y una significativa disminución en los costos de producción y mantenimiento.

El algoritmo incorpora una valiosa herramienta de apoyo al teleoperador, que contribuye a minimizar tiempo y costos en el entrenamiento del personal, y, simultáneamente, aumenta la seguridad y confianza en este tipo de plataformas. El soporte del algoritmo se encuentra presente en las múltiples aplicaciones de teleoperación que los MAVs ofrecen como: filmaciones, vigilancia, búsqueda, rescate, levantamientos cartográficos en urbanismo y como método menos invasivo en zonas protegidas, reconstrucción 3D de modelos en zonas de difícil acceso, transporte, etc.

En el mercado, durante el CES (Consumer Electronics Show) 2015 hace menos de 4 meses, se lanzó de forma oficial el bebop, un drone de micro-escala construido por la empresa francesa Parrot. Una de las nuevas características que el drone presenta respecto a sus versiones anteriores es la estabilización de vídeo en tiempo real, pero su costo asciende a 500 euros, mientras que utilizando nuestro algoritmo se puede dotar de la misma funcionalidad al AR.Drone 1.0 y 2.0, plataformas que actualmente se consiguen por menos de 260 euros.

5.2 Líneas futuras

Las investigaciones futura a desarrollarse se enmarcará en tres tópicos:

Evaluación de teleoperadores

Esta primera línea de investigación futura se encuentra en proceso. Es la evaluación experimental del desempeño de teleoperadores sin experiencia en tareas de seguimiento utilizando las imágenes inestables y las estabilizadas mediante nuestra propuesta. Se comparará el desempeño de los teleoperadores en cada caso, respecto al tiempo que les tome completar la tarea. Se llevó a cabo un experimento piloto con cinco voluntarios donde se observó indicios de mejora en el desempeño, utilizando las imágenes estabilizadas, y en los próximos meses se replicará la experiencia con un mayor número de voluntarios.

Base de datos

Otra de las líneas futuras en las que ya se está trabajando es la construcción de una base de datos con vídeos capturados desde distintos MAVs, en varios escenarios con y sin objetos en movimiento, y diferentes frecuencias de captura. Esta base de datos incluirá además de las imágenes capturadas onboard, la información inercial a partir de la IMU del MAV, la información de posición capturada con un cámara cenital y las acciones de control ejecutadas por el teleoperador. Existen bases de datos de vídeos inestables que pueden utilizarse para la evaluación de los algoritmos de estabilización de vídeo. Escasas bases de datos incluyen la información de la IMU, las cuales se suelen capturar con dispositivos móviles, en especial smartphones. Pero uno de los principales inconvenientes que se presentó durante el desarrollo de la investigación doctoral, fue la ausencia de bases de datos que incluyan la acción de control del teleoperador y la información de posición desde una perspectiva cenital más objetiva.

Aplicaciones en otros algoritmos

Una tercera línea en desarrollo es la aplicación de nuestra propuesta de estabilización de vídeo para mejorar el desempeño de algoritmos de visión por computador que puedan asistir a la navegación autónoma. Un ejemplo de ellos es la detección de caras, con el que se han obtenido resultados preliminares que muestran una mejora en la detección, usando imágenes capturadas desde el MAV y previamente estabilizadas [105].

Modelo

Nuestra propuesta de estabilización del Capítulo 4 se basa en el filtro de Kalman para la estimación del movimiento a partir de los parámetros extraídos y del modelo previamente estimado que incluye la acción de control. Cuando la dinámica de movimiento es altamente impredecible, como es el caso de los vuelos exteriores, donde las corrientes de aíre, condiciones de presión y temperatura son muy difíciles de modelar, el algoritmo puede presentar problemas. No obstante se está explorando una posible solución que incorpore al modelo usado en el filtro de Kalman la información inercial sincronizada. La sincronización con la información inercial es otro tópico de investigación en la estabilización de vídeo, en el cual existen soluciones propuestas, como se mencionó en el Capítulo 1, con sus respectivas limitaciones.

Fisheye

La incorporación de una lente tipo fisheye a la cámara onboard del MAV, permitiría aumentar el ángulo de visión y solventar el problema de la pérdida de información que es inevitable durante la estabilización de vídeo. La combinación de nuestra propuesta con algoritmos de compensación del efecto

de fisheye de la literatura, comprende una propuesta para un trabajo futuro que aún no se ha explorado.

5.3 Publicaciones

Journal Papers

- **W. G. Aguilar** and C. Angulo, "Real-time model-based video stabilization for micro aerial vehicles," *Neural Process Letters*, p. 1--19, 2015.

  2013 Impact Factor: 1.237

- **W. G. Aguilar** and C. Angulo, "Real-time video stabilization without phantom movements for micro aerial vehicles," *EURASIP Journal on Image and Video Processing*, vol. 2014, no. 1, p. 46, 2014.

  2013 Impact Factor: 0.662

Conference Papers

- **W. G. Aguilar** and C. Angulo, "Robust video stabilization based on motion intention for low-cost micro aerial vehicles," in *Multi-Conference on Systems, Signals Devices (SSD), 2014 11th International*, 2014, pp. 1–6.

- **W. G. Aguilar** and C. Angulo, "Estabilización de vídeo en micro vehículos aéreos y su aplicación en la detección de caras," in *Memorias del IX Congreso de Ciencia y Tecnología ESPE 2014*, 2014.

- **W. G. Aguilar**, C. Angulo, R. Costa, and L. Molina, "Control autónomo de cuadricopteros para seguimiento de trayectorias," in *Memorias del IX Congreso de Ciencia y Tecnología ESPE 2014*, 2014.

- **W. G. Aguilar** and C. Angulo, "Estabilización robusta de vídeo basada en diferencia de nivel de gris," in *Memorias del VIII Congreso de Ciencia y Tecnología ESPE 2013*, 2013.

- **W. G. Aguilar** and C. Angulo, "Compensación de los Efectos Generados en la Imagen por el Control de Navegación del Robot Aibo ERS 7," in *Memorias del VII Congreso de Ciencia y Tecnología ESPE 2012*, 2012.

- **W. G. Aguilar** and C. Angulo, "Compensación y Aprendizaje de Efectos Generados en la Imagen durante el Desplazamiento de un Robot," in *X Simposio CEA de Ingeniería de Control*, 2012.

5.4 Financiación

- **Proyecto de Investigación:** Tratamiento del dolor y la ansiedad basado en la interacción de robots sociales con niños para mejorar la experiencia del paciente - PATRICIA.

  Institución: Ministerio de Economía y Competitividad. PROYECTOS COORDINADOS DE I+D+i.

  Referencia: TIN2012-38416-C03-01.

  Fecha: De 01.01.2013 a 31.12.2015.

- **Beca de Doctorado:** Programa “Convocatoria Abierta 2011”.

  Institución: Secretaría de Educación Superior, Ciencia, Tecnología e Innovación SENESCYT de la República del Ecuador.

  Fecha: De 31.08.2011 a 31.08.2015.

Referencias

[1] F. Kendoul, "Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems," *Journal of Field Robotics*, vol. 29. pp. 315–378, 2012.

[2] J. Wendel, O. Meister, C. Schlaile, and G. F. Trommer, "An integrated GPS/MEMS-IMU navigation system for an autonomous helicopter," *Aerosp. Sci. Technol.*, vol. 10, no. 6, pp. 527–533, Sep. 2006.

[3] "Autonomous Flight in GPS-Denied Environments Using Monocular Vision and Inertial Sensors," *J. Aerosp. Inf. Syst.*, vol. 10, no. 4, pp. 172–186, Apr. 2013.

[4] A. Fernandez, J. Diez, D. de Castro, P. F. Silva, I. Colomina, F. Dovis, P. Friess, M. Wis, J. Lindenberger, and I. Fernandez, "ATENEA: Advanced techniques for deeply integrated GNSS/INS/LiDAR navigation," in *2010 5th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC)*, 2010, pp. 1–8.

[5] J. Engel, J. Sturm, and D. Cremers, “Camera-based navigation of a low-cost quadrocopter,” in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 2815–2821.

[6] J. Engel and D. Cremers, “Accurate Figure Flying with a Quadrocopter Using Onboard Visual and Inertial Sensing,” in *IMU*, 2012.

[7] P. B. François, C. David, and D. Jemmapes, “The Navigation and Control technology inside the AR . Drone micro UAV,” in *18th IFAC World Congress*, 2011, pp. 1477–1484.

[8] J. Engel, J. Sturm, and D. Cremers, “Scale-aware navigation of a low-cost quadrocopter with a monocular camera,” *Rob. Auton. Syst.*, vol. 62, no. 11, pp. 1646–1656, Nov. 2014.

[9] A. Ollero, J. Ferruz, F. Caballero, S. Hurtado, and L. Merino, “Motion compensation and object detection for autonomous helicopter visual navigation in the COMETS system,” in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA ’04. 2004*, 2004, pp. 19–24 Vol.1.

[10] F. Liu, M. Gleicher, H. Jin, and A. Agarwala, “Content-preserving warps for 3D video stabilization,” *ACM Trans. Graph.*, vol. 28, no. 3, p. 1, Jul. 2009.

[11] F. Liu, M. Gleicher, J. Wang, H. Jin, and A. Agarwala, “Subspace video stabilization,” *ACM Transactions on Graphics*, vol. 30. pp. 1–10, 2011.

[12] M. Grundmann, V. Kwatra, and I. Essa, “Auto-directed video stabilization with robust L1 optimal camera paths,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2011, pp. 225–232.

[13] B. M. Smith, L. Zhang, H. Jin, and A. Agarwala, “Light field video stabilization,” in *2009 IEEE 12th International Conference on Computer Vision*, 2009, pp. 341–348.

[14] G. Hanning, N. Forslow, P.-E. Forssen, E. Ringaby, D. Tornqvist, and J. Callmer, “Stabilizing cell phone video using inertial measurement sensors,” in *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2011, pp. 1–8.

[15] A. Karpenko, D. Jacobs, and M. Levoy, “Digital Video Stabilization and Rolling Shutter Correction using Gyroscopes,” 2011.

[16] P.-E. Forssen and E. Ringaby, “Rectifying rolling shutter video from hand-held devices,” in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 507–514.

[17] G. Zhang and H. Bao, “Keyframe-based real-time camera tracking,” in *2009 IEEE 12th International Conference on Computer Vision*, 2009, pp. 1538–1545.

[18] Z. Dong, G. Zhang, and H. Bao, “Robust monocular SLAM in dynamic environments,” in *2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2013, pp. 209–218.

[19] G. Zhang, X. Qin, W. Hua, T.-T. Wong, P.-A. Heng, and H. Bao, “Robust Metric Reconstruction from Challenging Video Sequences,” in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[20] T. Igarashi, T. Moscovich, and J. F. Hughes, “As-rigid-as-possible shape manipulation,” in *ACM SIGGRAPH 2005 Papers on - SIGGRAPH ’05*, 2005, p. 1134.

[21] Z. Zhou, H. Jin, and Y. Ma, “Plane-Based Content Preserving Warps for Video Stabilization,” in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2299–2306.

[22] S. Agarwal, Y. Furukawa, N. Snavely, I. Simon, B. Curless, S. M. Seitz, and R. Szeliski, “Building Rome in a day,” *Commun. ACM*, vol. 54, no. 10, p. 105, Oct. 2011.

[23] C. Buehler, M. Bosse, and L. McMillan, “Non-metric image-based rendering for video stabilization,” in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 2, pp. II–609–II–614.

[24] S. Cho, J. Wang, and S. Lee, “Video deblurring for hand-held cameras using patch-based synthesis,” *ACM Trans. Graph.*, vol. 31, no. 4, pp. 1–9, Jul. 2012.

[25] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski, “Towards Internet-scale multi-view stereo,” in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 1434–1441.

[26] N. Jiang, Z. Cui, and P. Tan, “A Global Linear Method for Camera Pose Registration,” in *2013 IEEE International Conference on Computer Vision*, 2013, pp. 481–488.

[27] J. Nianjuan, T. Ping, and C. Loong-Fah, "Seeing double without confusion: Structure-from-motion in highly ambiguous scenes," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1458–1465.

[28] D. Nistér, "An efficient solution to the five-point relative pose problem.," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 6, pp. 756–77, Jun. 2004.

[29] C. Wu, "Towards Linear-Time Incremental Structure from Motion," in *2013 International Conference on 3D Vision*, 2013, pp. 127–134.

[30] Y. Matsushita, E. Ofek, W. Ge, X. Tang, and H.-Y. Shum, "Full-frame video stabilization with motion inpainting.," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 7, pp. 1150–63, Jul. 2006.

[31] Y. Matsushita and E. Ofek, "Full-Frame Video Stabilization," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, pp. 50–57.

[32] C. Morimoto and R. Chellappa, "Evaluation of image stabilization algorithms," in *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP '98 (Cat. No.98CH36181)*, vol. 5, pp. 2789–2792.

[33] M. L. Gleicher and F. Liu, "Re-cinematography: improving the camera dynamics of casual video," in *Proceedings of the 15th international conference on Multimedia - MULTIMEDIA '07*, 2007, p. 27.

[34] M. L. Gleicher and F. Liu, "Re-cinematography: Improving the camerawork of casual video," *ACM Trans. Multimed. Comput. Commun. Appl.*, vol. 5, no. 1, pp. 1–28, Oct. 2008.

[35] A. Hartley, Richard and Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[36] S. Liu, L. Yuan, P. Tan, and J. Sun, "Bundled camera paths for video stabilization," *ACM Trans. Graph.*, vol. 32, no. 4, p. 1, Jul. 2013.

[37] S. Liu, L. Yuan, P. Tan, and J. Sun, "SteadyFlow: Spatially Smooth Optical Flow for Video Stabilization," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 4209–4216.

[38] B.-Y. Chen, K.-Y. Lee, W.-T. Huang, and J.-S. Lin, "Capturing Intention-based Full-Frame Video Stabilization," *Comput. Graph. Forum*, vol. 27, no. 7, pp. 1805–1814, Oct. 2008.

[39] S. Ertürk and T. J. Dennis, “Image sequence stabilisation based on DFT filtering,” *IEE Proc. - Vision, Image, Signal Process.*, vol. 147, no. 2, p. 95, 2000.

[40] Y. Matsushita, E. Ofek, W. Ge, X. Tang, and H. Y. Shum, “Full-frame video stabilization with motion inpainting.,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, pp. 1150–1163, 2006.

[41] M. Grundmann, V. Kwatra, and I. Essa, “Auto-directed video stabilization with robust L1 optimal camera paths,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2011, pp. 225–232.

[42] M. Grundmann, V. Kwatra, D. Castro, and I. Essa, “Calibration-free rolling shutter removal,” in *2012 IEEE International Conference on Computational Photography (ICCP)*, 2012, pp. 1–8.

[43] M. Grundmann, “Computational video: post-processing methods for stabilization, retargeting and segmentation,” Georgia Institute of Technology, 2013.

[44] A. I. Mourikis and S. I. Roumeliotis, “A Multi-State Constraint Kalman Filter for Vision-aided Inertial Navigation,” in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 3565–3572.

[45] D. Strelow, “Motion Estimation from Image and Inertial Measurements,” *Int. J. Rob. Res.*, vol. 23, no. 12, pp. 1157–1195, Dec. 2004.

[46] S. Erturk, “Image sequence stabilisation: motion vector integration (MVI) versus frame position smoothing (FPS),” in *ISPA 2001. Proceedings of the 2nd International Symposium on Image and Signal Processing and Analysis. In conjunction with 23rd International Conference on Information Technology Interfaces (IEEE Cat. No.01EX480)*, pp. 266–271.

[47] S. Ertürk, “Real-Time Digital Image Stabilization Using Kalman Filters,” *Real-Time Imaging*, vol. 8, no. 4, pp. 317–328, Aug. 2002.

[48] A. Litvin, J. Konrad, and W. C. Karl, “Probabilistic video stabilization using Kalman filtering and mosaicing,” 2003, pp. 663–674.

[49] J. Yang, D. Schonfeld, and M. Mohamed, “Robust video stabilization based on particle filter tracking of projected camera motion,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, pp. 945–954, 2009.

[50] M. Tico and M. Vehvilainen, “Constraint motion filtering for video stabilization,” in *IEEE International Conference on Image Processing 2005*, 2005, pp. III–569.

[51] A. Goldstein and R. Fattal, “Video stabilization using epipolar geometry,” *ACM Trans. Graph.*, vol. 31, no. 5, pp. 1–10, Aug. 2012.

[52] Y.-S. Wang, F. Liu, P.-S. Hsu, and T.-Y. Lee, “Spatially and temporally optimized video stabilization.,” *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 8, pp. 1354–61, Aug. 2013.

[53] K. Lee, Y. Chuang, B. Chen, and M. Ouhyoung, “Video stabilization using robust feature trajectories,” *2009 IEEE 12th Int. Conf. Comput. Vis.*, pp. 1397–1404, 2009.

[54] M. Irani, “Multi-Frame Correspondence Estimation Using Subspace Constraints,” vol. 48, no. 153, pp. 173–194, 2002.

[55] F. Liu, Y. Niu, and H. Jin, “Joint Subspace Stabilization for Stereoscopic Video,” in *2013 IEEE International Conference on Computer Vision*, 2013, pp. 73–80.

[56] C. Tomasi and T. Kanade, “Shape and motion from image streams under orthography: a factorization method,” *Int. J. Comput. Vis.*, vol. 9, no. 2, pp. 137–154, Nov. 1992.

[57] A. Goh and R. Vidal, “Segmenting Motions of Different Types by Unsupervised Manifold Clustering,” in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–6.

[58] J. NAKAMURA, *Image sensors and signal processing for digital still cameras*. CRC press, 2005.

[59] C.-K. Liang, L.-W. Chang, and H. H. Chen, “Analysis and compensation of rolling shutter effect.,” *IEEE Trans. Image Process.*, vol. 17, no. 8, pp. 1323–30, Aug. 2008.

[60] S. Baker, E. Bennett, S. B. Kang, and R. Szeliski, “Removing rolling shutter wobble,” in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2392–2399.

[61] P. E. FORSSÉN and E. RINGABY, “Efficient video rectification and stabilization of cell-phones,” *Int. J. Comput. Vis.*, vol. 2, no. 7, 2011.

[62] P. Heckbert, “Survey of Texture Mapping,” *IEEE Comput. Graph. Appl.*, vol. 6, no. 11, pp. 56–67, 1986.

[63] W. G. Aguilar and C. Angulo, “Real-time video stabilization without phantom movements for micro aerial vehicles,” *EURASIP J. Image Video Process.*, vol. 2014, no. 1, p. 46, 2014.

[64] W. G. Aguilar and C. Angulo, “Real-time model-based video stabilization for micro aerial vehicles,” *Neural Process. Lett.*, 2015.

[65] K. Mikolajczyk and C. Schmid, “Scale & affine invariant interest point detectors,” *Int. J. Comput. Vis.*, vol. 60, pp. 63–86, 2004.

[66] C. Harris and M. Stephens, “A Combined Corner and Edge Detector,” in *Procedings of the Alvey Vision Conference 1988*, 1988, pp. 23.1–23.6.

[67] S. Leutenegger, M. Chli, and R. Y. Siegwart, “BRISK: Binary Robust invariant scalable keypoints,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2011, pp. 2548–2555.

[68] A. Alahi, R. Ortiz, and P. Vandergheynst, “FREAK: Fast retina keypoint,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2012, pp. 510–517.

[69] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “ORB: An efficient alternative to SIFT or SURF,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2011, pp. 2564–2571.

[70] S. Battiato, G. Gallo, G. Puglisi, and S. Scellato, “SIFT features tracking for video stabilization,” in *Proceedings - 14th International conference on Image Analysis and Processing, ICIAP 2007*, 2007, pp. 825–830.

[71] D. G. Lowe, “Object recognition from local scale-invariant features,” *Proc. Seventh IEEE Int. Conf. Comput. Vis.*, vol. 2, 1999.

[72] H. Bay, T. Tuytelaars, and L. Van Gool, “SURF: Speeded up robust features,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2006, vol. 3951 LNCS, pp. 404–417.

[73] L. Juan and O. Gwun, “A comparison of sift, pca-sift and surf,” *Int. J. Image Process.*, vol. 3, pp. 143–152, 2009.

[74] J. Ponce and D. Forsyth, *Computer vision: a modern approach*. 2012, p. 793.

[75] M. A. Fischler and R. C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” *Communications of the ACM*, vol. 24. pp. 381–395, 1981.

[76] S. Choi, T. Kim, and W. Yu, "Performance Evaluation of RANSAC Family," *Procedings Br. Mach. Vis. Conf. 2009*, pp. 81.1–81.12, 2009.

[77] K. G. Derpanis, "Overview of the RANSAC Algorithm," *Image Rochester NY*, vol. 4, pp. 2–3, 2010.

[78] Y.-F. Hsu, C.-C. Chou, and M.-Y. Shih, "Moving camera video stabilization using homography consistency," in *2012 19th IEEE International Conference on Image Processing*, 2012, pp. 2761–2764.

[79] C. Song, H. Zhao, W. Jing, and H. Zhu, "Robust video stabilization based on particle filtering with weighted feature points," *IEEE Trans. Consum. Electron.*, vol. 58, pp. 570–577, 2012.

[80] H. Changl, S. Lai, and U. Systems, "A robust and efficient video stabilization algorithm," in *2004 IEEE International Conference on Multimedia and Expo (ICME) (IEEE Cat. No.04TH8763)*, 2004, vol. 20, pp. 29–32.

[81] R. Strzodka and C. Garbe, "Real-time motion estimation and visualization on graphics cards," *IEEE Vis. 2004*, 2004.

[82] C. Wang, J. Kim, K. Byun, J. Ni, and S. Ko, "Robust digital image stabilization using the Kalman filter," *IEEE Trans. Consum. Electron.*, vol. 55, no. 1, pp. 6–14, Feb. 2009.

[83] J. Canny, "A Computational Approach to Edge Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.

[84] C. Harris and M. Stephens, "A Combined Corner and Edge Detector," in *Procedings of the Alvey Vision Conference 1988*, 1988, pp. 23.1–23.6.

[85] O. Miksik and K. Mikolajczyk, "Evaluation of local detectors and descriptors for fast feature matching," *Pattern Recognit. (ICPR), 2012 21st …*, pp. 2681–2684, 2012.

[86] B. Tordoff and D. Murray, "Guided sampling and consensus for motion estimation," *Lect. Notes Comput. Sci.*, 2002.

[87] J. Rawat, Paresh and Singhai, "Adaptive motion smoothening for video stabilization," *Int. J. Comput.*, vol. 72, no. 20, pp. 14–20, 2013.

[88] S. Wu, D. C. Zhang, Y. Zhang, J. Basso, and M. Melle, "Adaptive smoothing in real-time image stabilization," in *Visual Information Processing XXI*, 2012, p. 83990L.

[89] Y. Wang, Z. Hou, K. Leman, and R. Chang, "Real-Time Video Stabilization for Unmanned Aerial Vehicles.," *MVA*, pp. 2–5, 2011.

[90] Y. Wang, R. Chang, and T. Chua, "Video stabilization based on high degree b-spline smoothing," *… (ICPR), 2012 21st …*, no. Icpr, pp. 3152–3155, 2012.

[91] Y. G. Ryu, H. C. Roh, and M. J. Chung, "Long-time video stabilization using point-feature trajectory smoothing," in *Digest of Technical Papers - IEEE International Conference on Consumer Electronics*, 2011, pp. 189–190.

[92] D. Jing and X. Yang, "Real-Time Video Stabilization Based on Smoothing Feature Trajectories," *Appl. Mech. Mater.*, vol. 519–520, no. 1662–7482, pp. 640–643, 2014.

[93] Q.-T. Faugeras, Olivier and Luong, *The geometry of multiple images: the laws that govern the formation of multiple images of a scene and some of their applications*. MIT press, 2004.

[94] W. G. Aguilar and C. Angulo, "Robust video stabilization based on motion intention for low-cost micro aerial vehicles," in *Multi-Conference on Systems, Signals Devices (SSD), 2014 11th International*, 2014, pp. 1–6.

[95] W. G. Aguilar and C. Angulo, "Estabilización robusta de vídeo basada en diferencia de nivel de gris," in *Memorias del VIII Congreso de Ciencia y Tecnología ESPE 2013*, 2013.

[96] B. Bailey, Stephen W and Bodenheimer, "A comparison of motion capture data recorded from a Vicon system and a Microsoft Kinect sensor," in *Proceedings of the ACM Symposium on Applied Perception*, 2012, pp. 121–121.

[97] J. Xu, H. Chang, S. Yang, and M. Wang, "Fast feature-based video stabilization without accumulative global motion estimation," *IEEE Trans. Consum. Electron.*, vol. 58, no. 3, pp. 993–999, Aug. 2012.

[98] M. Niskanen, O. Silven, and M. Tico, "Video Stabilization Performance Assessment," *2006 IEEE Int. Conf. Multimed. Expo*, 2006.

[99] S.-J. Kang, T.-S. Wang, D.-H. Kim, A. Morales, and S.-J. Ko, "Video stabilization based on motion segmentation," *2012 IEEE Int. Conf. Consum. Electron.*, pp. 416–417, 2012.

[100] C.-L. Fang, T.-H. Tsai, and C.-H. Chang, "Video stabilization with local rotational motion model," in *2012 IEEE Asia Pacific Conference on Circuits and Systems*, 2012, pp. 551–554.

[101] A. Yilmaz, O. Javed, and M. Shah, “Object tracking: A survey,” *ACM Comput. Surv.*, vol. 38, p. 13, 2006.

[102] Z. Zhang, “A flexible new technique for camera calibration,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, pp. 1330–1334, 2000.

[103] W. G. Aguilar, C. Angulo, R. Costa, and L. Molina, “Control autónomo de cuadricopteros para seguimiento de trayectorias,” in *Memorias del IX Congreso de Ciencia y Tecnología ESPE 2014*, 2014.

[104] X. Blasco, S. García-Nieto, and G. Reynoso-Meza, “Control autónomo del seguimiento de trayectorias de un vehículo cuatrirrotor. Simulación y evaluación de propuestas,” *Rev. Iberoam. Automática e Informática Ind. RIAI*, vol. 9, no. 2, pp. 194–199, Apr. 2012.

[105] W. G. Aguilar and C. Angulo, “Estabilización de vídeo en micro vehículos aéreos y su aplicación en la detección de caras,” in *Memorias del IX Congreso de Ciencia y Tecnología ESPE 2014*, 2014.