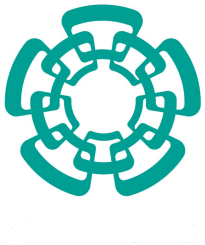


Centro de Investigación y de Estudios Avanzados
del Instituto Politécnico Nacional

Laboratorio de Tecnologías de Información,
CINVESTAV-Tamaulipas

Reconstrucción tridimensional de fachadas de edificios empleando imágenes monoculares obtenidas por un vehículo aéreo no tripulado autónomo

Tesis que presenta:

Carlos Alberto Motta Ávila

Para obtener el grado de:

Maestro en Ciencias en Computación

Dr. José Gabriel Ramírez Torres, Co-Director
Dr. Eduardo Arturo Rodríguez Tello, Co-Director

Cd. Victoria, Tamaulipas, México. Febrero, 2014

La tesis presentada por Carlos Alberto Motta Ávila fue aprobada por:

Dr. Hiram Galeana Zapién

Dr. Ezra Federico Parra González

Dr. José Gabriel Ramírez Torres, Co-Director

Dr. Eduardo Arturo Rodríguez Tello, Co-Director

Cd. Victoria, Tamaulipas, México., 28 de Febrero de 2014

«Señor, Tú nos das los dones, pero nos pides a cambio la fatiga »

-Leonardo da Vinci
(1452-1519)

«Worker bees can leave,
Even drones can fly away,
The queen is their slave. »

Agradecimientos

- Al CONACYT y su magnífico programa de becas de posgrado, mediante el cual tuve sustento económico durante dos años.
- Al CINVESTAV-Tamaulipas.
- A los doctores José Gabriel Ramírez Torres y Eduardo Rodríguez Tello por su asesoría durante el desarrollo de este trabajo de investigación.
- A los revisores, doctores Hiram Galeana Zapién y Ezra Federico Parra González por sus valiosos comentarios y correcciones.

Índice General

Índice de Figuras

Índice de Tablas

Índice de Algoritmos

Publicaciones

Resumen

Reconstrucción tridimensional de fachadas de edificios empleando imágenes monoculares obtenidas por un vehículo aéreo no tripulado autónomo

por

Carlos Alberto Motta Ávila
Laboratorio de Tecnologías de Información, CINVESTAV-Tamaulipas
Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, 2014
Dr. Eduardo Arturo Rodríguez Tello, Co-Director
Dr. José Gabriel Ramírez Torres, Co-Director

A través de la elaboración de un modelo tridimensional de la fachada de una edificación es posible describir los elementos arquitectónicos que la conforman, así como los detalles cuantitativos (sus medidas) y cualitativos (su apariencia) que la caracterizan. Dicho modelado suele ser un proceso lento y costoso, que requiere ubicar de manera precisa en distintos puntos alrededor de la construcción, un escáner láser para recuperar una nube de puntos que cubra totalmente la fachada del edificio. Sin embargo, la reconstrucción tridimensional a partir de imágenes elimina el costo de emplear un escáner tipo láser.

En este trabajo de investigación se emplea un **VANT** como plataforma de captura de imágenes, para el desarrollo de una solución novedosa al problema de modelado tridimensional de fachadas a través de imágenes tomadas desde un **VANT**. Entre las características principales de la solución propuesta, destacan la construcción de forma automática y durante el vuelo del **VANT**, de un modelo burdo de la fachada de la edificación que permita detectar los "huecos" de la digitalización para garantizar en un proceso posterior la reconstrucción completa del edificio. El enfoque de la solución propuesta desarrolla una nueva aplicación de la visión por computadora a la robótica móvil aérea, aportando al estado del arte de la fotogrametría y estrechando la brecha con la investigación con **VANT**.

El problema de determinar la posición de la plataforma aérea con respecto a la fachada fue resuelto mediante odometría, la cual se estima mediante los sensores de velocidad del **VANT**. Sin embargo, debido a que durante el vuelo existen perturbaciones del entorno la cual no es registrada por los sensores se generan errores de incertidumbres en la posición. Mediante el uso de marcadores visuales artificiales, cuya posición en la escena o fachada es fija y conocida, se determina la posición del **VANT** con precisión, aunado a lo estimado por odometría. El control de navegación junto con la odometría fue validado mediante experimentación en un escenario sin perturbaciones, obteniendo resultados satisfactorios para el desplazamiento frente a la fachada de una edificación.

El procesamiento de imágenes se realiza en pares de manera secuencial, siendo de gran importancia determinar una correcta relación de características presentes en ambas, ya que de esto depende la correcta construcción del modelo tridimensional. La construcción del modelo burdo tridimensional mediante fotografías se centró en los algoritmos más recientes disponibles en el estado del arte para procesamiento de imágenes en tiempo real. Se evaluaron detectores de puntos característicos, descriptores binarios y métodos de emparejamiento con los cuales se obtuvo un procesamiento suficiente para operar en línea.

Se determinó que la calidad visual del modelo obtenido en línea depende del par inicial de imágenes, de la cual se calcula el *baseline* u origen para procesar la secuencia de imágenes. La obtención de nube de puntos tridimensionales a partir de las características de las imágenes se realizó por triangulación iterativa, la cual permite reducir el error de reproyección. Las nubes de puntos tridimensionales obtenidas de cada par de imágenes se incrementan al modelo burdo, estimando su relación con el *baseline* desde antes de ser procesadas, esto elimina el procesamiento directo con puntos tridimensionales. Los resultados obtenidos en la experimentación permiten demostrar la efectividad de la propuesta de solución para la construcción de un modelo tridimensional de forma automática, empleando imágenes obtenidas por un **VANT**. Se concluye este trabajo resumiendo áreas de oportunidad para la propuesta de solución.

Abstract

Building façace 3D reconstruction from monocular images obtained by an autonomous unmanned aerial vehicle

by

Carlos Alberto Motta Ávila
Information Technology Laboratory, CINVESTAV-Tamaulipas
Research Center for Advanced Study from the National Polytechnic Institute, 2014
Dr. Eduardo Arturo Rodríguez Tello, Co-advisor
Dr. José Gabriel Ramírez Torres, Co-advisor

The 3D reconstruction of a building facade allows to describe the architectural elements that compose it as well as its quantitative (the measures) and cualitative (the appearance) details. This modeling process, which is slow and demands a high computational resources, requires the right ubication of a laser scanner in order to be able to recover the point cloud of the whole facade. However, image-based 3D modeling requires just a digital camera, being a accessible and simpler solution than the laser-based one.

In this research work, a novel solution to the tridimensional facade modelling problem has been developed using an **UAV** as the platform to obtain the images. The on-line automatic reconstruction of a superficial facade model, that will allow to detect the digitalizing errors and will guarantee the complete building reconstruction in an later process, stands out among the main characteristics of the proposed solution.

The proposed solution developed a state of the art novel computer vision application and mobile robotics, contributing to the fairly new **UAV** research field.

Odometry, which was calculated through the UAV's velocity sensors, was used to estimate the position of the aerial platform in reference to the facade. However, during flight, errors in position estimation are produced due to perturbances. Artificial visual markers with known position, fixed in the facade, are used to help the navigation of the **UAV** and altogether with odometry, estimate the real position the aerial platform.

Images are processed sequentially in pairs, being a must to estimate the correct relation between them which finally will produce a visually appealing 3D model. State of the art algorithms in computer vision were evaluated to achieve online image matching.

The quality of the generated 3d model relies on the initial pair of images, which are the baseline of the model. The point cloud was produced with iterative triangulation, reducing the reprojection error of the whole model. After estimating image relation with the baseline, non baseline point clouds are produced and added to model.
Experimentation shows that the current proposed solution can achieve automatic 3D model online reconstruction with images obtained by an **UAV**. This work concludes by summarizing possible further improvements for the current proposed solution.

Nomenclatura

SANT	Sistema Aéreo No Tripulado
IBR	Renderización basada en imágenes, (*Image-based rendering*)
IBM	Modelado basado en imágenes, (*Image-based Modelling*)
SLAM	Localización y modelado simultáneos, (*Simultaneous Localization and Mapping*)
MAV	Vehículos Aéreos Miniatura, (*Miniature Air Vehicles*)
VTOL	Vehículos de despegue y aterrizaje vertical, (*Vertical Take-Off & Landing*)
LASE	Vehículos de baja altitud y corta duración, (*Low Altitude, Short-Endurance*)
LALE	Vehículos de baja altitud y larga duración, (*Low Altitude, Long Endurance*)
MALE	Vehículos de mediana altitud y larga duración, (*Medium Altitude, Long Endurance*)
HALE	Vehículos de gran altitud y larga duración, (*High Altitude, Long Endurance*)
SfM	Estructuras a partir de movimiento, (*Structure from Motion*(
LIDAR	Laser Imaging Detection and Ranging
RANSAC	Random Sampling Consensus
VTOL	Vertical Take Off and Landing
GCS	Ground Control Station
VANT	Vehículo Aéreo No Tripulado
BRISK	(*Binary Robust Invariant Scalable Keypoints*)
FREAK	(*Fast Retina Keypoints*)
SURF	(*Speeded Up Robust Features*)
SIFT	(*Scale-Invariant Feature Transform*)
LOD	Nivel de detalle, (*Level of Detail*)

1

Introducción

A través de la elaboración de un modelo tridimensional de la fachada de una edificación es posible describir los elementos arquitectónicos que la conforman, así como los detalles cuantitativos (sus medidas) y cualitativos (su apariencia) que la caracterizan. Esta reconstrucción permite desarrollar aplicaciones muy interesantes, resaltando especialmente el registro histórico-temporal de la condición de la fachada de monumentos y edificaciones históricas.

En la práctica, la obtención de un modelo 3D de un edificio suele ser un proceso lento y costoso, que requiere ubicar de manera precisa, en distintos puntos alrededor de la construcción, un escáner láser para recuperar una nube de puntos que cubra totalmente la fachada del edificio. La nube de puntos se construye al concluir el proceso de digitalización, por lo que esta tecnología no permite detectar, al momento de la captura, los "huecos" en la nube de puntos debido a oclusiones, ni tampoco recuperar información sobre el color y la textura del edificio.

Una alternativa interesante para cumplir con esta tarea es el empleo de vehículos autónomos. Los **VANT** son aeronaves que carecen de piloto humano a bordo y que vuelan con un cierto grado de autonomía. Usualmente, su navegación es observada y/o controlada por un equipo humano en tierra a través de una estación de radio control. En la última década, el desarrollo tecnológico de los **VANT** ha progresado significativamente, lo que se refleja en un incremento importante en sus aplicaciones militares, civiles y científicas.

En este trabajo de investigación se propone el empleo de un **VANT** como plataforma de captura de imágenes, para el desarrollo de una solución novedosa al problema de modelado tridimensional de fachadas a través de imágenes tomadas desde el **VANT**. Entre las particularidades de la solución propuesta, destaca la construcción de forma automática y durante el vuelo del **VANT**, de un modelo burdo de la fachada de la edificación que permita detectar los "huecos" de la digitalización para garantizar que la reconstrucción fina del edificio, realizada al finalizar el proceso de captura de imágenes, sea completa. Asimismo, el uso del **VANT** permite prescindir de andamiajes para realizar la captura de imágenes, y la reconstrucción tridimensional a partir de imágenes elimina el costo de emplear un escáner tipo láser.

1.1 Antecedentes

Un modelo digital tridimensional es una representación virtual de un objeto, el cual puede ser manipulado por medio de una computadora. En función de su nivel de detalle y fidelidad, dicho modelo permite comprender la estructura del objeto original, analizar sus características y estudiar el impacto de modificaciones estructurales. Cuando se debe construir el modelo a partir de un objeto real, como es el caso de un edificio o fachada, es necesario contar con los medios que permitan adquirir una cantidad de información suficiente para la construcción del modelo y para el análisis de

su estructura.

A través de los últimos años se ha visto crecer el interés en el desarrollo de soluciones para el modelado y la reconstrucción tridimensionales de edificios, fachadas y monumentos, en campos como la ingeniería civil, la preservación digital de edificios históricos y, con la llegada del turismo digital, el entretenimiento. Existe también un gran potencial de aplicación de los modelos 3D de ciudades en planificación de desarrollo urbano, la evaluación de daños en en situaciones de desastres naturales y la inspección y monitores de instalaciones y servicios urbanos.

Para algunas de las aplicaciones mencionadas anteriormente, no se cuenta con un registro digital actualizado de la estructura tridimensional de la edificación. Esta situación puede explicar principalmente por la época histórica en que fue construido el edificio, pero incluso las construcciones más recientes pueden haber sufrido transformaciones o alteraciones a través del tiempo, tanto por razones climáticas como sociales, por lo que puede ser necesario actualizar su registro digital.

La obtención del modelo digital de un edificio puede realizarse a través de técnicas muy diversas, que emplean distintos equipos según el nivel de detalle o la precisión deseados. Estas técnicas van desde los métodos totalmente manuales elaborados por expertos en ingeniería civil, hasta las técnicas con escáner digital, como el caso de los escáner láser (**LIDAR**[1]).

El nivel de detalle está sujeto principalmente a la aplicación para la que se destine el modelo. Por ejemplo, si se considera el caso de planeamiento de desarrollo urbano, resulta de mayor utilidad un modelo general del edificio sin sus elementos arquitectónicos individuales (como son las ventanas o las puertas), ya que se analizan las edificaciones en un contexto de utilidad urbana. En contraste, para construir el modelo digital de un edificio histórico, para su estudio estructural o para el turismo

[1] *Light Detection and Ranging*

digital, se da mayor importancia a las características que definan su estilo arquitectónico, como es el caso de los elementos que componen las fachadas.

Al estudiar el modelo virtual de la fachada de un edificio histórico es posible valorar los daños y el desgaste, contribuyendo a la conservación del mismo; también, al recuperar la información de la arquitectura y estética, se rescata el conocimiento aplicado durante la construcción del edificio.

1.2 Motivación

Por el gran valor histórico, social y económico del estudio de la arquitectura y fachada de edificios, monumentos e instalaciones en general, el problema de la obtención de un modelo tridimensional y de la reconstrucción de fachadas de una edificación resulta de un enorme interés académico, científico y tecnológico. A partir de la digitalización de la fachada de una construcción es posible aproximar su aspecto inicial mitigando el desgaste, haciendo posible apreciar las características cualitativas y cuantitativas originales. En el contexto de la preservación histórica, este tipo de modelos permite contextualizar el estilo arquitectónico del edificio y valorar el desgaste ocasionado durante su historia.

En la actualidad, la elaboración de un modelo digital a partir de una edificación física se realiza a través de un proceso lento y complejo, que requiere de sensores especializados y un personal altamente capacitado. Pero los progresos recientes en el desarrollo de VANT permite considerar utilizar el punto de vista privilegiado de estos vehículos para, a partir del análisis de imágenes aéreas, construir un modelo tridimensional preciso y detallado de un edificio.

1.2.1 Reconstrucción tridimensional de edificios

La fachada de un edificio está compuesta de elementos estéticos y estructurales que la distinguen, por lo que es importante considerarlos en la recuperación del modelo tridimensional, ajustando, según el propósito final, el nivel de detalle del modelo.

Las técnicas empleadas para la adquisición automática de datos para la elaboración del modelo tridimensional pueden clasificarse en dos tipos: activas y pasivas. La adquisición activa se caracteriza por la medición del reflejo sobre el objeto de algún tipo de onda emitida desde el propio sensor (luz estructurada, ultrasonido, láser, microondas, etc.) mientras que en las técnicas pasivas se mide el brillo emitido o reflejado por el propio objeto para inferir su forma, por medio de fotografías (luz visible, luz infrarroja, espectrografía).

Los sensores activos obtienen directamente las coordenadas tridimensionales necesarias para la generación de una rejilla (*mesh*) que modele la superficie de un objeto, mientras que los sensores pasivos proveen imágenes que requieren un procesamiento para derivar en un modelo tridimensional. Para la selección del tipo de sensores a emplear, Pop *et al.* en el artículo *"3D buildings modelling based on a combination of techniques and methodologies"* [41] sugieren considerar como factores de decisión: el costo o presupuesto, la aplicación, la complejidad del objeto y el tiempo de procesamiento. En el año 2008, Strecha *et al.* [57] realizaron una comparativa entre técnicas basadas en imágenes y escáner láser (**LIDAR**) para la adquisición de exteriores, demostrando que el procesamiento de fotografías es capaz de lograr resultados comparables a los obtenidos por el **LIDAR** a un costo sensiblemente menor.

La fotogrametría es el conjunto de técnicas que abarcan el arte, la ciencia y tecnología para obtener información cuantitativa confiable acerca de objetos a través de captura, medición e

interpretación de imágenes fotográficas [33]. Las técnicas pasivas, en las cuales se considera el uso de cámaras, pueden clasificarse de acuerdo al número de cámaras empleado: monocular, binocular o estéreo y multicámara. La visión monocular, definida como la visión por medio de una sola cámara, presenta retos más complejos y distintos a aquellos de la visión estéreo o de los sistemas multicámara. Para estos últimos usualmente se conoce la configuración de posición y distancia entre las cámaras, lo cual facilita la recuperación de características tridimensionales, mientras que en la visión monocular la diferencia de posición desde la cual las distintas imágenes fueron obtenidas debe ser estimada. La estructura por movimiento (**SfM**) es un subcampo de la visión por computadora el cual ha tenido éxito en crear reconstrucciones tridimensionales densas a partir de visión monocular, como lo muestran Pollefeys *et al.* en el artículo [39].

Entre las principales dificultades para la adquisición de información tridimensional en un ambiente real, destaca el manejo de equipos especializado, además de que se requiere situar al sensor en diferentes posiciones respecto al objeto o fachada para obtener la información necesaria. Autores como Blaer y Allen en [5], Nüchter *et al.* en [38] y Thrun *et al* en [60] han propuesto emplear robots móviles para facilitar el proceso de reconstrucción de entornos exteriores.

1.2.2 Sistemas aéreos no tripulados

Un vehículo aéreo no tripulado (**VANT**) es un aeronave que realiza el vuelo sin tripulación humana a bordo y que puede ser controlado remotamente o volar de forma autónoma [68]. Los **VANT**, junto con la estación de control remoto y la tripulación en tierra forman un Sistema Aéreo No Tripulado (**SANT**), como se muestra en la Figura 1.1. Durante el vuelo, el **VANT** envía a la estación de control remoto los datos de vuelo en tiempo real como son la posición, velocidad, altitud, distancia, batería o combustible, entre otros. Para realizar el vuelo autónomo, las trayectorias se pueden definir mediante coordenadas **GPS**, puntos de control en tierra (**GCP**) o navegación visual.

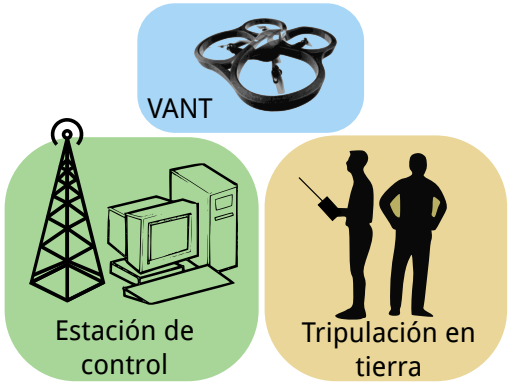


Figura 1.1: Sistema Aéreo No Tripulado (**SANT**)

De acuerdo a la Asociación Europea de Vehículos aéreos no tripulados (***EUROUVS***) y la organización *UVS International*, los **VANT** se pueden clasificar de acuerdo a la altura a la que operan, el tiempo que pueden permanecer en vuelo debido al combustible o fuente de energía, su velocidad, el peso máximo que pueden cargar durante el despegue y por tamaño, entre otros factores. En la Figura 1.2 se muestra la clasificación por la altura y duración de vuelo, definiendo los tipos:

MAV Vehículos Aéreos Miniatura

VTOL Vehículos de despegue y aterrizaje vertical

LASE Vehículos de baja altitud y vuelo de corta duración

LALE Vehículos de baja altitud y vuelo de larga duración

MALE Vehículos de mediana altitud y vuelo de larga duración

HALE Vehículos de gran altitud y vuelo de larga duración

Los **VANT** desde su origen han tenido un gran atractivo para la investigación en el dominio de la robótica móvil. Sin embargo, la complejidad y costos de los sistemas de estabilización, control y comunicación en las aeronaves, las convertían en una plataforma de desarrollo accesible únicamente

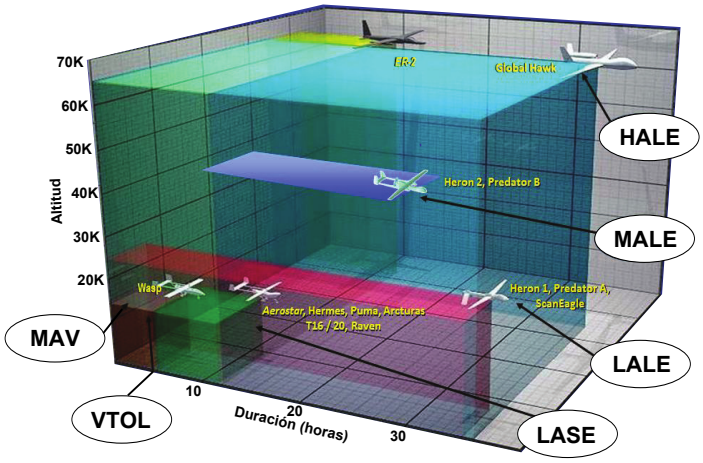


Figura 1.2: Clasificación de los **VANT**, de acuerdo a Polski [40]

para su uso militar. En años recientes, la miniaturización de los semiconductores y los mecanismos micro-electrónicos (**MEMS**), han permitido el desarrollo de unidades de estabilización accesibles y precisas, con las cuales se han desarrollado **VANT** de menor costo, ampliando el horizonte de aplicación, investigación y desarrollo hacia las aplicaciones civiles.

El Grupo Teal realizó en el año 2013 un estudio de mercado [59] en el cual estima que las inversiones en **VANT** se duplicará durante la próxima década. Dicho estudio indica que el gasto actual mundial anual en **VANT** es de *USD*$5.2 billones y se espera que en los próximos diez años alcance *USD*$89 billones. *Strategic Defence Intelligence* [56] estima para ese mismo periodo que el mercado de **VANT** crecerá hasta los *USD*$114.7 billones.

Watts *et al.* mencionan en el artículo [62] que los **VANT** se están convirtiendo rápidamente en la plataforma preferida para desarrollo de instrumentos y aplicaciones de teledetección. Entre las ventajas más importantes de los **VANT** que enumera el autor, podemos resaltar la capacidad de adquisición de datos en tiempo real, rápidamente disponible para su transmisión y procesamiento. Colomina *et al.* [9] afirma que los **VANT** son el nuevo paradigma de la fotogrametría de alta resolución y bajo costo. La mayoría de loa autores coinciden en que los **VANT** combinan las

ventajas de los robots terrestres y los sensores aerotransportados, explotando el espacio tridimensional completo. En el caso particular de los **VANT** tipo **VTOL**, su mayor ventaja consiste en la facilidad de despliegue en áreas remotas, sin la necesidad de pistas de aterrizaje y su capacidad de vuelo estacionario, ideal para tareas de monitoreo y vigilancia.

Everaerts *et al.* [14] realizaron un reporte acerca de las regulaciones, clasificación y aplicaciones de **VANT** en la realización de mapas. Así, Niranjan *et al.* [37] recopilan las aplicaciones civiles más importantes en las que se emplean **VANT**, siendo las aplicaciones más comunes silvicultura y agricultura, arqueología y patrimonio ([50]), topografía ambiental, monitoreo de tráfico y reconstrucción tridimensional de estructuras hechas por el hombre ([61], [22]). Entre las aplicaciones con fines bélicos se puede citar las tareas de reconocimiento, espionaje y los bombarderos ([11]); mientras que las aplicaciones civiles más recurrentes son el monitoreo de vegetación y de tipo de suelo ([31]), el estudio de condiciones climáticas ([36]), la vigilancia de costas en zonas protegidas ([49]), la observación de tráfico terrestre ([17]), la filmación de eventos deportivos ([47]), la fotogrametría remota ([67]), el acceso a zonas o comunidades aisladas por algún fenómeno meteorológico o desastre natural ([32]), la supervisión de ductos de petróleo ([44]), la elaboración de planos arqueológicos ([50]), por sólo mencionar algunas.

1.3 Planteamiento del problema

El modelo tridimensional de la fachada de un edificio obtiene por medio del análisis de datos tomados desde distintos puntos alrededor de la edificación. De este análisis se extrae la información sobre la geometría y, en el caso del análisis de imagen, la textura de la fachada del edificio. Este proceso se realiza fuera de línea, al concluir la captura de la información, y no garantiza una reconstrucción completa del edificio, ya que pueden existir áreas de la fachada que hayan quedado

ocultas por otros elementos de la misma (oclusión).

Este problema resulta más evidente cuando la fuente de información son imágenes capturadas con cámaras, ya que la reconstrucción tridimensional se basa en el análisis de correspondencias en un conjunto de imágenes, del mismo elemento de la fachada, tomadas desde puntos de vista diferentes. Si dicho elemento aparece únicamente en una o dos de las imágenes, su reconstrucción tridimensional no es posible.

Es ente punto donde un sistema que permita detectar, al momento de la captura de la información, aquellos puntos que no han sido suficientemente fotografiados para realizar la reconstrucción tridimensional a detalle, resulta sumamente interesante. Contar con un sistema flexible de toma de imágenes y con un sistema de reconstrucción rápido, en línea y al momento de fotografiar el edificio, representa una ventaja importante sobre los demás trabajos existentes en la literatura.

La investigación y desarrollo de técnicas de fotogrametría que emplean **VANT** consideran aeronaves que son construidas *ex profeso* o bien se tratan de plataformas comercialmente disponibles con costos elevados, mientras que las aeronaves de bajo costo son muy inestables en condiciones de viento adversas.

Existen además otras problemáticas a considerar: la recuperación del modelo tridimensional se realiza de forma incremental. lo que puede llevar a la acumulación de errores de deriva (Steffen *et al.* en el artículo [55]). Además, por la naturaleza propia del sistema, la información **GPS** de la que se dispone cuando se explora una amplia extensión vertical (como es el caso de las fachadas) no permite determinar de forma precisa la altitud de vuelo del **VANT**, como puede verificarse en los trabajos de Püschel *et al.* [42] y en Scaioni *et al.*[51]. Se requiere entonces de otro medio para determinar la posición del **VANT**, por lo que es necesario recurrir a la propia información visual que se está

capturando.

Durante este trabajo de investigación se ha pretendido contestar a la siguiente pregunta: A partir de un conjunto de fotografías monoculares de la fachada de un edificio, obtenidas por un **VANT**, así como la información de su orientación y considerando las restricciones de tiempo de vuelo de la aeronave, ¿Es posible desarrollar un proceso de imágenes que permita obtener, de forma automática, un modelo tridimensional incremental de bajo detalle en línea (mientras que el **VANT** se encuentre en vuelo) de tal manera que sea posible, para el operador humano, detectar las zonas de la fachada que no han sido suficientemente fotografiadas?

A partir de esta pregunta, es posible formular el problema de investigación de la siguiente manera: *Dada la fachada de un edificio con elementos estructurales y estéticos característicos que es fotografiada por un **VANT** equipado con una cámara y manipulado remotamente, como se muestra en la Figura 1.3, se desarrollará un procesamiento de imágenes que genere, de manera automática durante el vuelo del **VANT**, un modelo tridimensional de bajo detalle a partir de las imágenes transmitidas inalámbricamente por el **VANT** y recibidas en una computadora.*

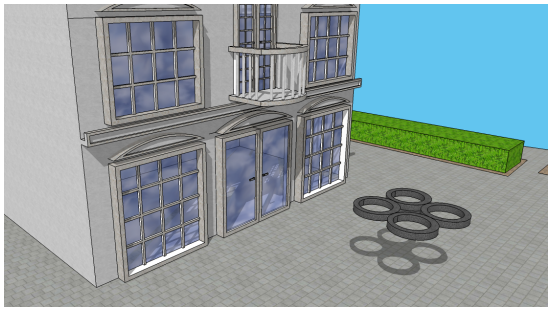


Figura 1.3: Fotografiando la fachada de un edificio con un **VANT** tipo cuadricóptero.

Si bien la construcción del modelo tridimensional propiamente dicho, con un alto nivel de detalle, es un trabajo que debe hacerse fuera de línea y con todo el conjunto de fotografías obtenido, la

construcción de un modelo tridimensional burdo, al mismo tiempo que el **VANT** ejecuta la tarea de fotografiar la fachada del edificio, permitirá localizar regiones sobre la misma para las cuales no se cuenta con la suficiente información para la construcción del modelo tridimensional final. De esta manera será posible corregir *in situ* el proceso de fotografía, evitando los defectos que de otra manera sería notorios durante la reconstrucción del modelo de mayor detalle, evitando retrasos y gastos suplementarios.

1.4 Hipótesis

La hipótesis de trabajo de la presente investigación es la siguiente: *Considerando un **VANT** tipo **VTOL** con tiempo de vuelo limitado que adquiere fotografías monoculares de una fachada, es posible desarrollar un procesamiento de imágenes rápido y ligero que permita la construcción de un modelo tridimensional burdo del edificio, en línea e incremental, para detectar las zonas de las cuales no existe suficiente información para construir un modelo con un mayor nivel de detalle, es decir, zonas que requieren ser fotografiadas nuevamente.*

1.5 Objetivos

1.5.1 General

Aportar al estado del arte de la fotogrametría y la robótica móvil de **VANT** por medio de un sistema de procesamiento de imágenes que genere, de manera automática y en línea, un modelo tridimensional burdo de la fachada de un edificio, empleando las imágenes transmitidas por la cámara fotográfica del **VANT** que realiza la tarea de captura de imágenes.

1.5.2 Particulares

- **Contar con un sistema de obtención de características tridimensionales burdo el cual opera a la par que se adquieren las fotografías con el VANT considerando un tiempo de vuelo máximo de 15 minutos**.

- Lograr un sistema de navegación que permita al ***VANT*** ubicarse frente a una fachada y realizar trayectorias predefinidas con un margen de error menor a 40cm entre la posición deseada u la real.

- Obtener la reconstrucción tridimensional automática de una estructura basada en imágenes obtenidas mediante un **VANT**. El sistema realizará sin intervención de un usuario u operador la recuperación de las características de la fachada para la generación de un modelo tridimensional en línea.

- Desarrollar una nueva aplicación de la visión por computadora a la robótica móvil aérea.

1.6 Enfoque propuesto

El procedimiento propuesto para lograr la reconstrucción de la fachada de un edificio con una cámara monocular montada en un **VANT** se muestra en la Figura 1.4. La propuesta se basa en los algoritmos de reconstrucción tridimensional por medio de fotografías presentados por Barazzetti *et al.* en su artículo [2]. Estos algoritmos fueron desarrollados en el campo de la visión por computadora, y apenas recientemente comienzan a ser integrados a la robótica móvil. La innovación de nuestra propuesta es el empleo de un **VANT** para realizar la captura de las imágenes para la reconstrucción de un edificio. Para lograrlo, se propone integrar el sistema de visión por computadora junto con técnicas de odometría visual y odometría inercial para lograr la generación de un modelo en línea.

El proceso de captura comienza con la estimación de la posición del vehículo frente a la fachada del edificio. El origen del sistema coordenado de reconstrucción será dado por el usuario, empleando un marcador artificial que será colocado sobre la edificación, que el **VANT** podrá ubicar visualmente y utilizar como referencia. Este mismo marcador permitirá retomar la reconstrucción del modelo en caso de que sea necesario detener el proceso de captura de imágenes, por ejemplo, al cambiar las baterías del vehículo.

Al seguir la trayectoria de vuelo, el **VANT** realiza la adquisición de fotografías en distintas posiciones y orientaciones. Al desplazarse, el sistema utiliza la información inercial y las mismas imágenes para estimar la posición y orientación (pose) de la cámara con respecto al edificio y a la posición inicial. Esta información queda asociada a las imágenes capturadas para realizar la reconstrucción tridimensional del edificio.

Cada nueva imagen es analizada y filtrada para detectar únicamente los puntos más sobresalientes que estén presentes en las imágenes anteriores, para efectuar la triangulación de las coordenadas y

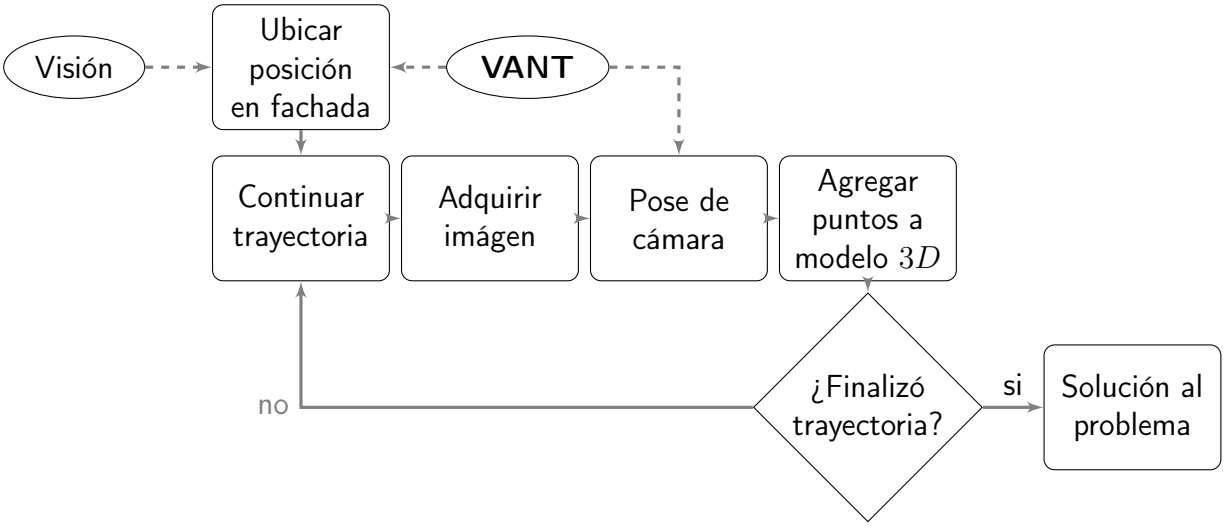


Figura 1.4: Procedimiento propuesto para la construcción en línea de un modelo tridimensional de bajo detalle, empleando un **VANT**.

obtener una nube de puntos tridimensionales, las cuales conforman el modelo tridimensional de la fachada. El procesamiento y análisis de imágenes y la incorporación de los puntos tridimensionales al modelo es un proceso que se repite durante la trayectoria de vuelo del **VANT**. Se finaliza una vez que se termine la trayectoria de vuelo, y por lo tanto la adquisición de las imágenes, dando por terminado la construcción del modelo tridimensional de la fachada.

La Figura 1.5 nuestra los diversos componentes funcionales en los que se dividió el enfoque propuesto. Cada uno de estos bloques realiza una tarea en específico:

- Control de desplazamiento y Estimación de posición del **VANT**.
- Detección visual de un marcador artificial.
- Seguimiento de trayectorias del **VANT**.
- Reconstrucción de un modelo tridimensional burdo en línea.
- Visualización del modelo burdo.

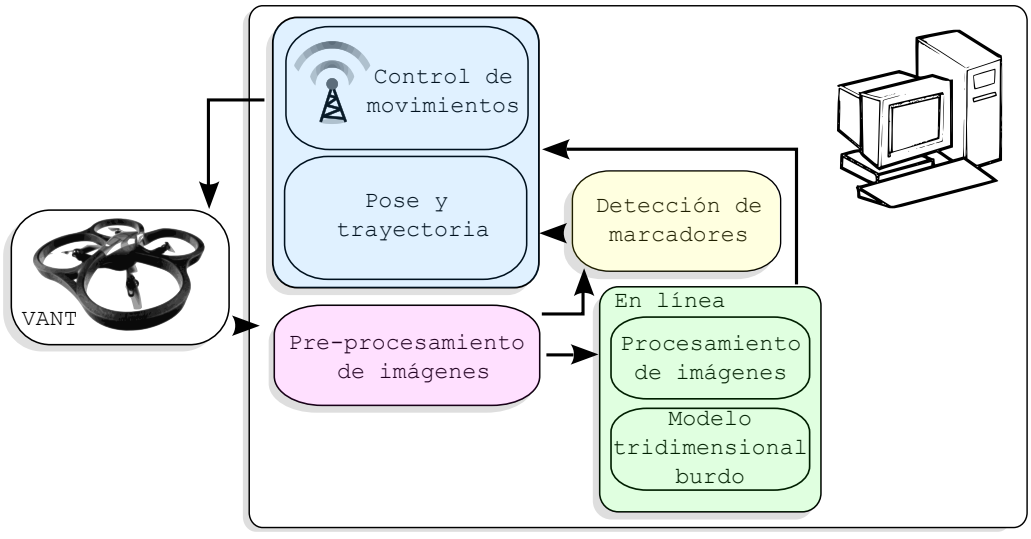


Figura 1.5: Diagrama general de la solución propuesta

1.6.1 Control de desplazamiento y Estimación de posición del VANT

Los dos pilares en los que se apoya la propuesta de solución son el control del **VANT** y la visión por computadora. Como el modelo tridimensional burdo que deseamos obtener es procesado en línea, la plataforma aérea debe tener la capacidad de transmitir imágenes vía inalámbrica a una estación en tierra. Otra de las características del **VANT** a considerar es el control de desplazamientos y trayectorias predeterminadas al volar frente a la fachada.

Con un control de desplazamiento adecuado se garantiza un seguimiento apropiado de la trayectoria deseada. Dicho control debe ser capaz de realizar una estimación correcta de la posición del **VANT** en el aire, utilizando la información proporcionada por la estación inercial a bordo, así como la información visual proporcionada por la misma cámara.

La misma estimación de posición es necesaria para realizar la triangulación de puntos durante la reconstrucción tridimensional de la fachada del edificio.

1.6.2 Detección visual de un marcador artificial.

El sistema se apoya en marcadores visuales artificiales para determinar el inicio de la trayectoria y del proceso de modelado tridimensional por medio del **VANT**. Un marcador visual artificial es una figura de forma conocida y fácilmente identificable en una escena, como el que se muestra en la Figura 1.6. La navegación de robots móviles empleando marcadores visuales para determinar su posición se ha investigado exitosamente, como lo reportado por Lim y Lee en el artículo [28] y, en el caso particular de aplicación a un **VANT**, por Lamberti *et al.* en el artículo [26].

El usuario debe colocar físicamente un marcador en la fachada del edificio. Por tratarse de un marcador plano con dimensiones conocidas, podemos determinar la posición del **VANT** con respecto

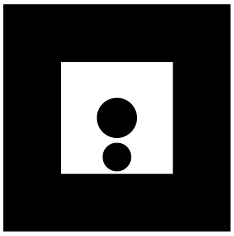


Figura 1.6: Marcador visual artificial

a éste y acoplar inicialmente la información de odometría a la visual. En el caso de que se agote la batería del **VANT**, el marcador permite retomar el procesamiento, recuperando una referencia conocida. También es posible utilizarla como una referencia común cuando se desee emplear más de un **VANT**.

1.6.3 Seguimiento de trayectorias del VANT

El **VANT** sigue una trayectoria definida *a priori*, sobre un plano paralelo a la fachada. El origen de la trayectoria es el marcador visual. Durante el vuelo a lo largo de la trayectoria predefinida se adquieren y procesan las imágenes.

La trayectoria predefinida contempla posiciones y poses en las cuales el **VANT** adquiere fotografías que aporten información suficiente para el proceso de modelado tridimensional. El sistema se apoya en lo propuesto por Rachmielowski *et al.* en el artículo [43], para asegurarse de que la información entre fotografías sea suficiente.

1.6.4 Reconstrucción de un modelo tridimensional burdo en línea.

Como resultado del procesamiento de las imágenes durante el vuelo se obtiene una nube de puntos en el espacio tridimensional, permitiendo apreciar visualmente un modelo de la fachada a la

par de la adquisición de fotografías y desplazamiento del **VANT**. Los marcadores permiten obtener una escala absoluta, cuando una nueva imagen es procesada la escala relativa a las primeras dos imágenes es determinada por medio del modelo tridimensional.

1.6.5 Infraestructura

Se cuenta con un vehículo aéreo no tripulado tipo cuadricóptero *AR.Drone* de la marca *Parrot* mostrado en la Figura 1.7, este **VANT** cuenta con las siguientes características:

- Procesador ARM9 a 468MHz
- DDR RAM de 128MB a 200MHz
- Autonomía de vuelo de 13 minutos
- Comunicación vía Wi-Fi con rango de 50 metros
- Altímetro ultrasónico para estabilidad vertical con rango de 6 metros
- Cámara frontal 640×480px VGA con campo visual aproximado de $75^\circ \times 60^\circ$
- Cámara vertical 176×144px con campo visual aproximado de $45^\circ \times 35^\circ$
- Unidad Inercial de 6 grados de libertad
- Peso: 420g
- Deriva de guiñada de 12° por minuto durante vuelo y 4° por minuto en modo *espera*
- Costo promedio 300.00 USD
- Computadora de escritorio Intel® Core™ i5-$2400S$@2.5GHz,
  4GB de memoria DDR3@1333MHz, tarjeta gráfica integrada AMD Radeon™HD 6750M, con sistema operativo Ubuntu 13.04 de 64bits.

Figura 1.7: Cuadricóptero *AR.Drone*, **VANT** tipo **VTOL**.

1.7 Estructura del documento

En el segundo capítulo se presentan los fundamentos teóricos básicos para la comprensión de los problemas, así como las soluciones, que integran esta investigación. El capítulo tercero aborda los trabajos más representativos relacionados con el modelado tridimensional de fachadas y edificios empleando **VANT**. El capítulo cuarto detalla la metodología propuesta para la solución al problema de investigación, donde se describen los enfoques considerados. En el quinto capítulo se describen los experimentos realizados para la validación del sistema, exponiendo sus resultados. El sexto y último capítulo plantea las conclusiones obtenidas para la propuesta así como las posibles mejoras y el trabajo futuro.

2

Estado del Arte

La reconstrucción tridimensional de fachadas de edificios empleando imágenes monoculares obtenidas por vehículos aéreos no tripulados (**VANT**) se sitúa en la convergencia de dos áreas fundamentales: la robótica móvil y la visión por computadora.

Para la propuesta de solución se analizaron inicialmente los trabajos del estado de arte referentes a la visión por computadora aplicada a la reconstrucción tridimensional de escenas, y particularmente aquellos que emplean una sola cámara, de manera que la recuperación de características tridimensionales depende completamente del análisis de las imágenes. Estos trabajos han comenzado a emplearse en la reconstrucción de medios ambientes de robots móviles con ruedas y, muy recientemente, en **VANT**.

El problema abordado se aleja un poco del problema clásico de robótica móvil denominado localización y mapeo (**SLAM**), donde el robot debe estimar su posición y generar un mapa en un

medio ambiente desconocido. En el caso de la reconstrucción de fachadas, el **VANT** debe realizar el mapa tridimensional y desplazarse en frente de un área de dimensiones conocidas *a priori* de manera aproximada, por lo que el medio ambiente no es totalmente desconocido. De hecho, el problema se acerca más a la estructura a partir de movimiento (**SfM**), que busca resolver el problema de recuperar la pose relativa de la cámara y generar una estructura tridimensional partiendo de un conjunto de imágenes obtenidas por dicha cámara. Un caso particular de **SfM** es la odometría visual (**VO**), que estima el movimiento tridimensional de la cámara de manera secuencial en tiempo real. Debido a que la propuesta aquí presentada emplea los sensores inerciales embarcados en el **VANT**, la textbfVO está incorporada como una herramienta auxiliar, siendo una mejora sustancial al sistema final de localización.

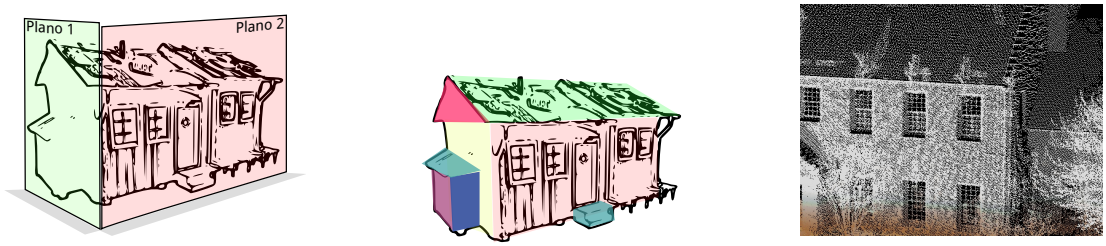
A continuación se presentan las publicaciones más recientes y representativas, de acuerdo a su importancia para el trabajo actual, comenzando por las técnicas visuales y finalizando con las metodologías que consideran el uso de un**VANT** y las restricciones de tiempo para el procesamiento.

2.1 Reconstrucción 3D a partir de imágenes

La fotogrametría de corto alcance busca obtener medidas tridimensionales precisas a partir de imágenes. Remondino y El-Hakim realizan en su artículo [46] una descripción y análisis de los métodos necesarios para producir un modelo tridimensional a partir de fotografías. En este clasifican el modelado de objetos y escenas en las siguientes categorías:

- Renderización basada en imágenes (**IBR**)
- Modelado basado en imágenes (**IBM**)
- Modelado basado en rango
- Combinación de imagen y rango

La **IBR** no genera un modelo geométrico de la escena ni de un objeto sino que sólo logra apariencia tridimensional. En contraste, el **IBM** emplea correspondencias entre fotografías 2D para recuperar las características 3D de la escena o del objeto. El modelado basado en rango emplea sensores activos los cuales emiten algún tipo de onda o luz y miden la respuesta o reflejo del objeto o la escena, como por ejemplo la tecnología **LIDAR**, luz estructurada, etc. La representación de los modelos tridimensionales obtenidos por **IBR**, **IBM** y modelado basado en rango se muestran en la Figura 2.1.



(a) Modelo **IBR**, planos para simular 3D

(b) Modelo **IBM**, cuerpos 3D

(c) Modelo por escáner láser, nube de puntos densa

Figura 2.1: Representación de como son construidos los modelos tridimensionales por diferentes técnicas de modelado

Para el caso de **IBM** y las correspondencias entre imágenes, Remondino y El-Hakim destacan que la exactitud mejora conforme una misma característica aparece en varias imágenes, aunque después de cuatro imágenes no hay mejoría perceptible. De igual forma hay que considerar que la distribución de los puntos en una imagen es más importante que la cantidad.

Barazzetti *et al.* en su artículo [2] proponen una metodología para el procesamiento de fotografías con el objetivo de obtener de forma automática un modelo tridimensional. La metodología propuesta por los autores es distinta si las imágenes forman parte de una secuencia ordenada o se trata de imágenes individuales dispersas. El modelo se genera partiendo de un par de imágenes ordenadas ampliándolo de forma incremental. En este artículo se realiza la comparativa de tiempo computacional

necesario entre los detectores de puntos característicos **SIFT** y **SURF**, aunque se concluye que un detector como **FAST** o similar mejora los tiempos de procesamiento para aplicaciones de tiempo real.

Para la etapa de emparejamiento entre imágenes a partir de la correspondencia entre puntos característicos, Barazzetti *et al.* realizan el estudio comparativo entre una búsqueda exhaustiva de correspondencias (cuadrática) y una búsqueda de tipo vecinos más cercanos[4], prefiriendo ésta última. Para obtener robustez en los resultados del emparejamiento, recomiendan obtener dos correspondencias candidatas $(d_{mn})^1$ y $(d_{mn})^2$ para cada punto, aceptando la primera correspondencia como válida solo si se cumple la Ecuación 2.1, donde el umbral t generalmente varía entre 0.5 y 0.8.

$$(d_{mn})^1 < t(d_{mn})^2 \tag{2.1}$$

Para la obtención de una nube de puntos densa en 3D de un objeto, a partir de una secuencia de n imágenes ordenadas I_i, Barazzetti *et al.* proponen agrupar la secuencia en $n-2$ tripletas $T_i = \{I_i, I_{i+1}, I_{i+2}\}$ y emparejar las imágenes $\{I_i, I_{i+1}\}$ y $\{I_{i+1}, I_{i+2}\}$ mientras que el par $\{I_i, I_{i+2}\}$ se relacionan con las tripletas previamente analizadas. Posteriormente, para integrar el modelo se compara con la tripleta siguiente $T_{i+1} = \{I_{i+1}, I_{i+2}, I_{i+3}\}$ que comparte puntos comunes logrando acoplar las coordenadas tridimensionales. Esta aproximación se representa en la Figura 2.2.

Esta estrategia ha comenzado a emplearse para la reconstrucción de edificios empleando vehículos terrestres, bajo condiciones de trabajo controladas.

2.2 Análisis de fachadas

Xiao *et al.* en su artículo [65] organizan la literatura relativa a la reconstrucción de fachadas en:

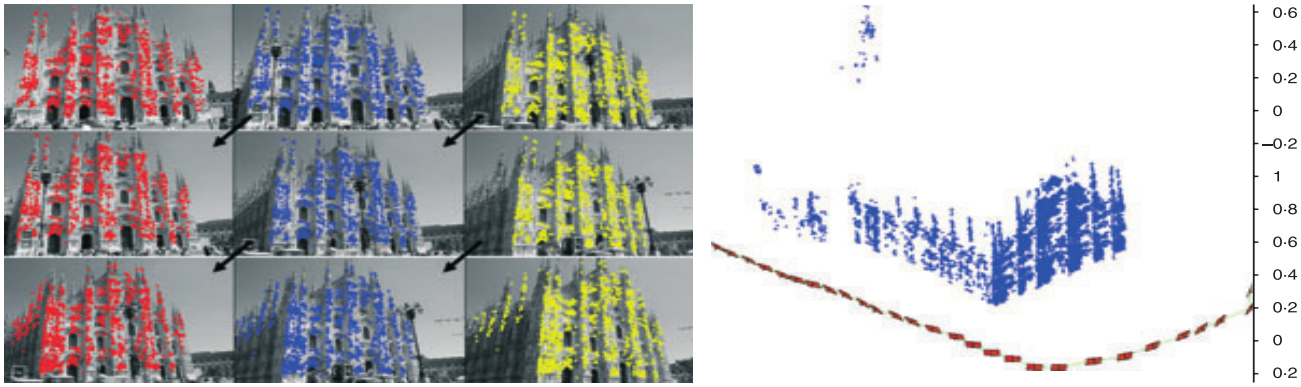


Figura 2.2: Tripletas propuestas por Barazzetti *et al.* en [2] para la construcción de un modelo tridimensional

- Basada en reglas
- Basada en imágenes
- Basada en visión

Para el modelado basado en reglas se requiere de un conjunto de patrones específicos para la descripción de la geometría del edificio. Ejemplos de este modelado pueden ser encontrados en los trabajos de Müller *et al.* en [34] y Yan *et al.* en [66]. Un modelado semiautomático es el basado en imágenes junto con interacción de un usuario, empleando fotografías para generar modelos de arquitecturas de forma interactiva. En este tipo de modelado Debevec *et al.* [10] proponen un modelo híbrido donde la geometría básica tipo caja se recupera por fotogrametría y posteriormente se refina el modelo proyectando pares de imágenes sobre éste.

Se considera el modelado basado en visión aquel en el que de forma automática se reconstruyen escenas a partir de imágenes. En las propuestas de Müller *et al.* en [35] y Xiao *et al.* [65] se divide la fachada en segmentos rectangulares, los cuales son empleados para identificar patrones repetitivos, como por ejemplo ventanas. La búsqueda de patrones repetidos en fachadas considera tres fases principales: división, agrupación y rectificación. Shen *et al.* en [53] resalta que en las propuestas de Müller *et al.* y Xiao *et al.* los segmentos rectangulares integran rejillas fijas; sin embargo, existen

fachadas para las cuales estas rejillas no se ajustan. Para solucionar esto, Shen *et al.* proponen ajustar los segmentos rectangulares en los que se divide la fachada de forma flexible, permitiendo encontrar patrones de forma adaptativa. Ejemplos de la segmentación de fachadas propuesto por Shen *et al.* se muestran en la Figura 2.3.

Figura 2.3: Ejemplos de segmentación de fachadas propuesta por Shen *et al.* en [53]

2.3 Reconstrucción 3D de fachadas con VANT

Para el modelado basado en imágenes, se considera utilizar un **VANT** para la adquisición de fotografías que permitan construir un modelo tridimensional. Entre las ventajas de utilizar un **VANT** para esta tarea destacan la rápida adquisición y transmisión de datos así como su movilidad. Estas propiedades son mencionadas por Remondino *et al.* en su artículo [45] el cual trata sobre fotogrametría empleando **VANT**.

Irschara *et al.* en su artículo [22] dan mayor importancia en acelerar el proceso de obtención de puntos característicos de las imágenes y su emparejamiento empleando **GPU**s para agilizar el tiempo de cómputo. En su propuesta parten de un conjunto de fotografías no organizadas de un edificio obtenidas por un **VANT**, considerando a éste simplemente como una cámara móvil sin restricciones

para obtener un conjunto extenso de imágenes. El procesamiento en su propuesta se realiza fuera de línea una vez obtenidas las imágenes, por lo que no existe una restricción de tiempo de procesamiento.

Kung *et al.* [25] obtienen un modelo tridimensional de un edificio de forma semiautomática fuera de línea. La trayectoria de vuelo del **VANT** es inicialmente calculada considerando el edificio del que se desea obtener el modelo, para lograr que el **VANT** vuele a su alrededor obteniendo tantas fotografías como le sea posible. De la nube de puntos obtenida se proyectan en un modelo tipo caja para obtener la forma tridimensional del edificio. Las etapas de nube de puntos, proyección de caja y modelo final se muestran en la Figura 2.4.

Figura 2.4: Modelado de edificios semiautomático de Kung *et al.* en [25]

Diskin y Asarien en su artículo [12] presentan un método para recuperar información tridimensional de una fachada en línea, partiendo de un flujo de video del que se obtiene un mapa de planos de profundidad. Diskin y Asarien emplearon una cámara con resolución 720p (1280×720 pixeles), con orientación y escala fijas desplazándose a lo largo de una fachada (Figura 2.5a), de manera como lo haría un vehículo o robot terrestre. Emplean la técnica **SURF** [3] para la obtención de los puntos característicos a partir del flujo óptico [21]. Para garantizar la validez de los puntos característicos detectados se considera que éstos deben estar presentes en cinco cuadros de video consecutivos. Cabe resaltar que en la propuesta de Diskin y Asarien no realizaron pruebas con un **VANT**, por lo que las imágenes correspondientes al desplazamiento a lo largo de la fachada son muy estables, sin

cambios de escala ni de orientación.

(a) Fachada (b) Planos de profundidad

Figura 2.5: Fachada y modelo obtenido por Diskin y Asarien en [12]

A partir de la hipótesis de movimiento paralelo de la cámara, es posible generar un mapa de disparidad de los puntos característicos detectados, en el cual los puntos cercanos a la cámara tienen alta disparidad mientras que los lejanos presentan poco desplazamiento. A partir de la disparidad se calcula la profundidad de los puntos, considerando la posición y la longitud focal de la cámara. Los valores de profundidad son discretos por lo que se obtienen planos de diferentes profundidades, como se puede ver en la Figura 2.5b. Finalmente para suavizar el efecto de planos discretos se propone emplear técnicas de súper-resolución para obtener un mayor número de planos.

Wefelscheid *et al.* [63] aplican con un **VANT** un método semejante al de Barazzetti *et al.* en el artículo [2] para obtener un modelo tridimensional del exterior de una casa. La adquisición de imágenes es automática conforme al trayecto de vuelo previamente trazado por lugares específicos o *landmarks*. La reconstrucción final y completa del modelo se realiza fuera de línea.

Como detector de puntos característicos, en este caso esquinas, emplearon los operadores Förtstner y la técnica **SIFT** para construir los descriptores. Dado un punto característico, éste se busca en las imágenes previas y de no encontrarse en tres imágenes consecutivas es descartado.

La nube de puntos y el modelo fuera de línea obtenido por esta propuesta se puede observar en la Figura 2.6.

(a) Nube de puntos dispersa con posiciones de la cámara

(b) Nube de puntos densa

Figura 2.6: Nubes de puntos obtenidas por Wefelscheid *et al.* en [63]

De los trabajos presentados se resumen las características de mayor interés para la presente propuesta en la Tabla 2.1. Al tratar con imágenes en secuencia ordenada se conoce *a priori* que algunas características presentes en una imagen deben aparecer en la imagen próxima según la adquisición, contrario a la secuencia no ordenada donde se debe realizar la búsqueda de características de una imagen en todo el conjunto de imágenes adquiridas. El modelo tridimensional obtenido a partir de las imágenes puede ser de forma automática, sin intervención de un usuario o de forma asistida, en una metodología semiautomática. Las trayectorias para los **VANT** suelen ser precalculadas por un usuario conforme el edificio a capturar y por *landmarks* que pueden ser determinados por coordenadas **GPS** o visuales.

2.4 Trabajos complementarios

Los **VANT** tipo **VTOL** tienen relativamente pocas restricciones físicas de posición y orientación, a diferencia de los robots terrestres o incluso los propios **VANT** de ala fija. Esta ventaja puede aprovecharse para la adquisición de imágenes con el objetivo de recuperar un modelo tridimensional

Autores	En secuencia	Automático	Procesamiento	Características
Irschara *et al.* [22]	No ordenada	Si	Fuera de línea	**GPU**: *SIFT* y emparejamiento
Kung *et al.* [25]	Ordenada	Semi	Fuera de línea	Trayectorias precalculadas
Diskin y Asarien [12]	Ordenada	Si	En línea	Mapa de profundidades. Súper-resolución (Fuera de línea)
Wefelscheid *et al.* [63]	Ordenada	Si	Fuera de línea	Navegación por *landmarks*

Tabla 2.1: Trabajos relacionados de **VANT** para modelado tridimensional de fachadas

considerando las técnicas establecidas de estructura a partir de movimiento (**SfM**). La **SfM** consiste en el problema de recuperar la pose relativa de la cámara así como una estructura tridimensional partiendo de un conjunto de imágenes obtenidas por dicha cámara. La propuesta busca aportar a los trabajos como el de Wnuk en su artículo [64] donde se busca acoplar la visión por computadora con la robótica mediante el uso de técnicas **SfM** con visión monocular y odometría.

Rachmielowski *et al.* en su artículo [43] resalta la necesidad de generar los modelos tridimensionales a partir de fotografías en tiempo real, almacenando únicamente aquellas imágenes que permitan generar un modelo denso en un procesamiento posterior y fuera de línea. El sistema está conformado por dos procesos paralelos, **SLAM** para la estructura tridimensional y renderizado para mostrar el modelo burdo en tiempo real. El sistema inicia con una posición conocida y posteriormente emplea segmentos de la imagen inicial para hacer una búsqueda por relación cruzada normalizada (**NCC**) donde se esperan las características del segmento.

Las imágenes se evalúan para determinar si aportan características originales ya que se busca almacenar el menor número de imágenes posibles. La medida de la distancia se obtiene de la ecuación 2.2 considerando la posición del centro de la cámara en el nuevo cuadro cc_{new} y cada uno de los

centros de cámara anteriores cc_i ponderando por la profundidad media inversa a los puntos de la escena X_j (donde $j \in [\ 1, m\ ]$) relativo a la nueva escena:

$$d_i = \frac{dist(cc_{new}, cc_i)}{\Sigma_j(depth_{cc_new}(X_j))/m} \tag{2.2}$$

La imagen se considera original si el mínimo de la distancia evaluado para todas las cámaras almacenadas $min_i(d_i)$ es mayor que el umbral α, donde usualmente $\alpha = 0.1$. En caso de que el mínimo se encuentre por debajo del umbral, para cada cc_i con $d_i < \alpha$ se calcula la distancia de su rayo de visión principal contra el del nuevo cuadro. Si la distancia angular está por arriba del segundo umbral β entonces es original, usualmente $\beta = \pi/4$.

Para la generación del modelo burdo se proyectan los puntos tridimensionales obtenidos a la vista virtual, sobre los puntos 2D resultantes se calcula una triangulación de Delaunay que se reproyecta en el espacio tridimensional. A dicha representación se añade la textura de las imágenes de forma que si la cámara toma una escena previamente registrada se forma un modelo renderizado tridimensional, como se aprecia en la Figura 2.7.

Figura 2.7: Renderizado incremental y vista virtual de la propuesta de Rachmielowski *et al.* en [43]

La navegación de robots móviles por medio de marcadores permite obtener una posición absoluta del robot, eliminando la acumulación de errores de odometría o estimación de la posición por medio

de sensores. Lamberti *et al.* en el artículo [26] combina la odometría visual con marcadores para ajustar el referencial local de un **VANT** al sistema de coordenadas global de la escena. Se conoce como odometría visual (*VO*) al proceso de estimar el movimiento de un agente (robot, vehículo o humano) solamente mediante información visual obtenida por una o más cámaras abordo de éste. En este artículo se ubican marcadores en posiciones absolutas conocidas, el **VANT** empleando una cámara orientada hacia el piso estima su posición por **VO** y en caso de detectar un marcador actualiza la posición absoluta sobre la estimada para eliminar la incertidumbre de odometría.

2.5 Conclusiones

En el presente estado del arte se menciona las distintas metodologías empleadas para la construcción de modelos tridimensionales de escenas, en particular las propuestas presentadas que parten de un conjunto de fotografías de una fachada.

Las técnicas semánticas de adquisición de fachadas presentadas requieren una base de conocimientos de las formas y objetos presentes. Dichas metodologías están principalmente enfocadas a una sola imagen de la fachada, simplificando a ésta y pasando por alto los elementos artísticos de interés. En contraste, las técnicas de obtención de nube de puntos y modelos tridimensionales a partir de pares o secuencias de imágenes son más flexibles a diversas formas y geometrías presentes. Debido a que se propone un método en línea de recuperación de características tridimensionales, se dará prioridad a encontrar un mayor número de detalles descartando por lo tanto las técnicas semánticas.

En el planteamiento del problema de la presente investigación, debemos considerar que es necesario la recuperación de características tridimensionales a partir de imáneges tomadas por un

VANT que es necesario localizar adecuadamente en el espacio 3D. Al emplear robots móviles se debe resaltar la capacidad de éstos para orientarse y desplazarse libremente en una escena así como los sensores con los que cuenta abordo.

En la literatura presentada la recuperación del modelo tridimensional de las fachadas se procesa fuera de línea. Si bien en algunos trabajos con **VANT** se menciona la capacidad de hacerlo en línea e incluso tiempo real, se ha pasado por alto realizar la experimentación correspondiente o las demostraciones correspondientes.

El problema de la construcción de un modelo tridimensional de la fachada de un edificio empleando un **VANT** es un tema de gran interés y en este trabajo se propone una solución a dicho problema. La propuesta trata con las restricciones de tiempo de vuelo del aeronave para generar un modelo tridimensional el cual permita visualizar al momento las características recuperadas de la fachada. El realizar este procesamiento en línea permite evaluar la calidad del modelo al momento de la captura de imágenes.

3

Fundamentos teóricos

El presente capítulo contempla los fundamentos teóricos necesarios para el desarrollo de este trabajo de investigación. Los temas se agrupan en las siguientes áreas: conceptos generales, geometría proyectiva, visión por computadora y robótica. Los conceptos generales permiten definir brevemente fundamentos teóricos necesarios para las demás áreas. La sección de geometría proyectiva permite modelar la adquisición de las fotografías, la cámara y su posición, así como el recuperar las características tridimensionales presentes. La visión por computadora permite detectar correspondencias entre las imágenes y así aplicar los conceptos de geometría proyectiva. Finalmente, en la sección de robótica se definen los conceptos básicos para comprender el funcionamiento del **VANT** cuadricóptero y su integración con la fotogrametría.

3.1 Conceptos generales

3.1.1 Sistema de coordenadas

Un sistema coordenado cartesiano permite representar en un espacio euclidiano tridimensional $\mathbb{R}^3$ la ubicación de un punto. Éste se conforma por un conjunto de tres ejes ortonormales (x, y, z) que coinciden en el origen del sistema en la coordenada $(0, 0, 0)$. La localización de un punto se especifica mediante la tripleta $(x, y, z) \in \mathbb{R}^3$, donde cada coordenada representa la proyección ortogonal de la localización del punto sobre el eje correspondiente. Los sistemas coordenados tridimensionales pueden ser de tipo derecho o izquierdo dependiendo de la dirección que tenga el eje z. Un sistema derecho y los planos que se forman entre los ejes se muestran en la Figura 3.1.

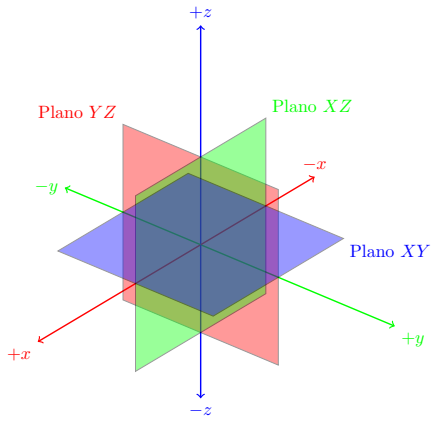


Figura 3.1: Sistema coordenado derecho en $\mathbb{R}^3$

Para describir la posición de un punto, partícula u objeto respecto a su entorno se emplea un sistema coordenado o *referencial* centrado en el mundo (**wcs**, *world coordinate system*) cuyo origen tiene que ser previamente ubicado de forma arbitraria. De forma semejante, al fijar un sistema coordenado local a un objeto, un punto de dicho objeto podrá ser descrito mediante coordenadas mundiales (**wcs**) o locales, dependiendo del referencial que se tome como base para describir su posición. Un ejemplo de lo anterior es el sistema coordenado local de la Figura 3.2, considerado

para un vehículo aéreo no tripulado tipo cuadricóptero.

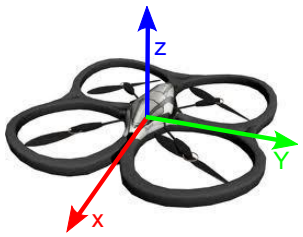


Figura 3.2: Sistema de coordenadas locales fijado al **VANT** cuadricóptero de la propuesta

Esto permite enunciar que la orientación de un cuerpo rígido se describe mediante la orientación relativa entre un sistema coordenado fijado a dicho cuerpo y un sistema coordenado fijo o inercial. Cada sistema coordenado está representado por un conjunto de tres vectores ortonormales que representan distintas bases para un mismo espacio vectorial $\mathbb{R}^3$.

3.1.2 Transformaciones

Las transformaciones permiten alterar la geometría de un objeto o figura modificando la posición de sus vértices respecto a algún sistema de coordenadas en el que se encuentre descrito. Las transformaciones más comunes son: traslación, rotación y escalamiento. Otras transformaciones son: reflexión, oblicuidad y proyección. Una transformación final o total está compuesta por una combinación de transformaciones, por ejemplo, una rotación seguida de una traslación; en este caso cabe mencionar que el orden de las transformaciones afecta la posición del objeto o figura final.

La geometría euclidiana describe al entorno por medio de ángulos, paralelismos y ortogonalidad, las cuales se conservan cuando se realiza una transformación, es decir, un cambio de coordenadas o desplazamiento. Una de las diferencias principales con la geometría proyectiva, la cual extiende a la euclidiana, consiste en que en la proyectiva las líneas paralelas convergen en el horizonte, en el llamado punto de desvanecimiento (*vanishing point*), mientras que en la geometría euclidiana éstas nunca se encuentran.

3.1.2.1. Rotación, Ángulos de Euler

El desplazamiento más general de un sólido con un punto fijo es un giro alrededor de algún eje, de tal forma que una matriz de rotación R puede representarse como una rotación simple alrededor de un eje en el espacio como se muestra en 3.1:

$$R = R(k, \theta) \tag{3.1}$$

donde $k \in \mathbb{R}^3$ es un vector unitario que define el eje de rotación y $\theta \in \mathbb{R}$ es el ángulo de rotación (usualmente en radianes) alrededor de dicho eje.

Las rotaciones elementales para los ejes x, y, z se expresan en las Ecuaciones 3.2, 3.3 y 3.4 respectivamente:

$$R(x, \phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\phi) & -sen(\phi) \\ 0 & sen(\phi) & cos(\phi) \end{bmatrix} \tag{3.2}$$

$$R(y, \theta) = \begin{bmatrix} cos(\theta) & 0 & sen(\theta) \\ 0 & 1 & 0 \\ -sen(\theta) & 0 & cos(\theta) \end{bmatrix} \tag{3.3}$$

$$R(z, \psi) = \begin{bmatrix} cos(\psi) & -sen(\psi) & 0 \\ sen(\psi) & cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.4}$$

Para cada una de las rotaciones anteriores se fija el eje de rotación, quedando cada una de ellas en función de su ángulo respectivo. Supóngase un cuerpo rígido en el que claramente se identifica una vista frontal y una vista superior; se fija un referencial o sistema coordenado a dicho cuerpo tal que un eje se encuentra dirigido hacia el frente y otro hacia arriba, el tercer eje se coloca de acuerdo a un referencial derecho, de tal manera que apunta hacia uno de los laterales del cuerpo. A los movimientos de rotación alrededor de los ejes que apuntan hacia el frente, hacia el lateral y hacia arriba se les denomina alabeo (*roll*), cabeceo (*pitch*) y guiñada (*yaw*), respectivamente. Para el caso del **VANT**, estos se aprecian con mayor claridad en la Figura 3.3.

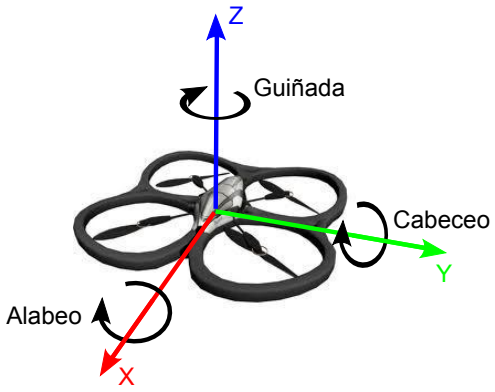


Figura 3.3: Ángulos de rotación (alabeo, cabeceo y guiñada) en un cuadricóptero.

3.1.2.2. Traslación

La traslación permite cambiar la posición de un objeto desplazándolo hacia un punto en el espacio. Si $\mathbf{M}_0$ es un punto definido en un cierto referencial y éste se traslada en una cierta dirección definida por el vector unitario k, una cierta distancia d, entonces su posición final $\mathbf{M}_1$ estará dada por la Ecuación 3.5.

$$\mathsf{M}_1 = \mathsf{M}_0 + d\mathsf{k}$$

$$\begin{bmatrix} x + t_x \\ y + t_y \\ z + t_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{3.5}$$

3.1.3 Pose de un objeto

La pose de un objeto refiere a la ubicación y orientación que tiene éste respecto a un sistema de coordenadas o referencial. La pose se define por las coordenadas (x, y, z) y los ángulos de rotación (ϕ, θ, ψ) respecto a los tres ejes, por lo que un objeto en el espacio tiene seis grados de libertad. Este concepto es de gran importancia en el presente trabajo, ya que para la recuperación de características tridimensionales a partir de imágenes se requiere conocer la pose de la cámara.

3.2 Geometría en visión por computadora

En esta sección se introducen los conceptos que permiten recuperar las coordenadas tridimensionales de una escena o entorno empleando fotografías obtenidas con sólo una cámara monocular.

3.2.1 Modelo de cámara oscura

El proceso de formación de una imagen es modelado como una proyección perspectiva de la escena en un plano retinal, proyectivo o plano de la imagen. La escena se define como un conjunto de puntos, líneas y superficies en el espacio Euclidiano $\mathbb{R}^3$. El modelo de cámara oscura (*pin-hole*) describe el proceso de formación de las imágenes. Se considera un sistema de coordenadas de la cámara al sistema ortonormal de coordenadas centrado en el centro del lente C con dos ejes paralelos a las

coordenadas el plano de la imagen y el tercer eje paralelo al eje óptico. En la Figura 3.4 se muestra el sistema coordenado tridimensional, él cual se simplifica en la Figura 3.5 para dos ejes.

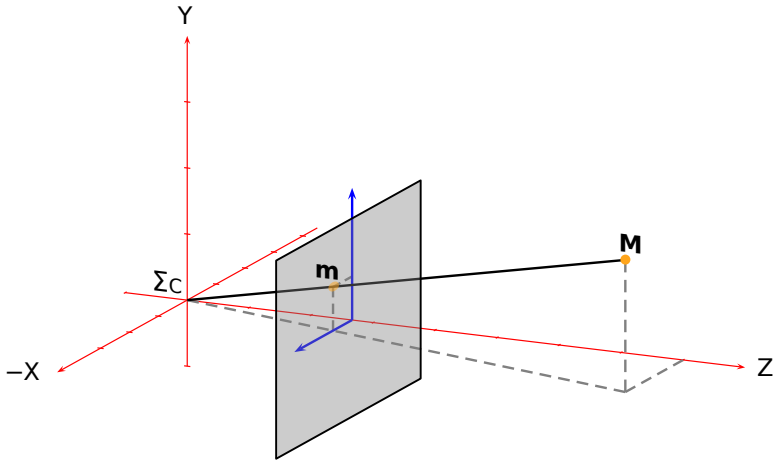


Figura 3.4: Sistema coordenado tridimensional para el modelo de cámara oscura

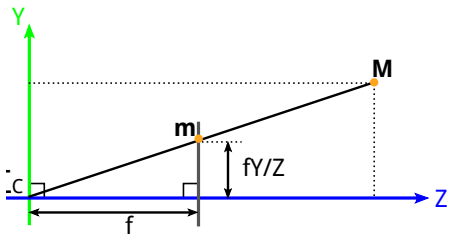


Figura 3.5: Plano YZ del modelo de cámara oscura

Si bien el plano de la imagen se encuentra en las cámaras por detrás el centro del lente C, dejando a éste entre la escena y la imagen, es posible ubicar el plano frente a éste como se muestra en la Figura 3.4 sin alterar el modelo proyectivo.

La luz que refleja un punto de la escena se considera en forma de rayo único, el cual es proyectado en una superficie que es el plano de la imagen o plano proyectivo. El eje óptico es la línea que pasa por C y que es perpendicular al plano de la imagen. El punto p_c conocido como **punto principal**, éste se ubica donde el eje óptico cruza el plano de la imagen.

La distancia focal describe la distancia f entre el punto C y el plano de la imagen. El tamaño relativo de un objeto distante en la imagen depende de la distancia focal. La proyección **m** en el plano de la imagen de un punto tridimensional **M** es la intersección del rayo óptico $(C, \mathbf{M})$ con el plano de la imagen. La relación entre los ejes de coordenadas $\mathbf{M} = [X, Y, Z]'$ y de la proyección $\mathbf{m} = (u, v)$ está dada por la Ecuación 3.6, que empleando coordenadas proyectivas se reescribe como se indica en la Ecuación 3.7, considerando $\lambda = Z$ como el factor de escalamiento homogéneo.

$$u = \frac{Xf}{Z}, v = \frac{Yf}{Z} \tag{3.6}$$

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_u & 0 & 0 & 0 \\ 0 & f_v & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{3.7}$$

Para que el sistema coordenado de la imagen tenga como origen al punto principal $p_c = (o_x, o_y)$ es necesario transformar el sistema de coordenadas. Esta transformación al ser integrada en la Ecuación 3.7 produce la Ecuación 3.8.

$$\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & o_x & 0 \\ 0 & f_y & o_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{3.8}$$

La Ecuación 3.8 puede ser reescrita como se muestra en la Ecuación 3.9:

$$\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & \gamma & o_x & 0 \\ 0 & f & o_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} \mathsf{K}|0_3 \end{bmatrix} \mathsf{M} \tag{3.9}$$

Los parámetros intrínsecos de la cámara son descritos por K, f_u y f_v representa la distancia focal expresada en píxeles. Las coordenadas del punto principal son representadas por (o_x, o_y). La oblicuidad (*skew*) γ usualmente es cero y describe la simetría del paralelogramo en los píxeles. El punto tridimensional $\mathsf{M} = [X, Y, Z, 1]'$ está definido en coordenadas homogéneas.

Para obtener la posición de un pixel $\mathsf{m} = (x, y, 1)$ de un punto tridimensional homogéneo M la cámara debe ser trasladada al origen de Σ_W y posteriormente rotada. Esto se expresa mediante la Ecuación 3.10:

$$\lambda \mathsf{m} = \begin{bmatrix} \mathsf{K}|0_3 \end{bmatrix} \begin{bmatrix} R & 0 \\ 0_3 & 1 \end{bmatrix} \begin{bmatrix} I_3 & -t \\ 0 & 1 \end{bmatrix} \mathsf{M} \tag{3.10}$$

Simplificando la Ecuación 3.10 se obtiene la Ecuación 3.11:

$$\lambda \mathsf{m} = \begin{bmatrix} \mathsf{K}|0_3 \end{bmatrix} \begin{bmatrix} R & -Rt \\ 0_3^T & 1 \end{bmatrix} \mathsf{M} = \mathsf{K}R \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} - \mathsf{K}Rt \tag{3.11}$$

Las matrices R y t forman la matriz de parámetros extrínsecos $\mathsf{P} = [R|t]$, que describe la orientación y posición de la cámara con respecto al marco de referencia global. Está definida por una matriz de 4×4 que describe un cambio en el sistema de coordenadas global, dado por rotación

$R_{3\times3}$ y traslación $t_{3\times1}$, llamados parámetros extrínsecos.

3.2.1.1. Calibración de la cámara oscura

La calibración de la cámara refiere a estimar los valores intrínsecos (matriz **K**) y la distorsión de la lente. Estos valores intrínsecos son fijos e invariantes a la escena y únicos para cada cámara. En cuanto a la distorsión, existen dos tipos de distorsión que se pueden producir en las fotografías: la distorsión radial, producida por la forma esférica del lente de la cámara, y la tangencial, producto de que el lente no se encuentre perfectamente paralela al plano de la imagen. Es posible atenuar los efectos de la distorsión mediante procesamiento de imagen.

Conocer los valores intrínsecos **K** que son: distancia focal f y el punto principal p_c así como compensar la distorsión es indispensable para la recuperación de características tridimensionales por medio de imágenes.

Más adelante repasaremos los principios que permiten realizar la calibración automática de una cámara.

3.2.1.2. Pose de la cámara

La pose para una cámara con parámetros intrínsecos conocidos se puede determinar mediante cuatro puntos coplanares no colineales. Un marcador artificial cuadrado permite por medio de sus vértices aprovechar lo anterior. Considerando las esquinas del marcador como $\mathbf{m}_i, i = 1, 2, 3, 4$ y el modelo de cámara oscura explicado anteriormente, podemos enunciar la relación de coordenadas de la forma como se muestra en la Ecuación 3.12:

$$\mathbf{m}_i = \mathbf{KPM}_i \tag{3.12}$$

Sustituyendo $\mathbf{KP} = G$, entonces la relación queda expresada como en la Ecuación 3.13:

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \begin{bmatrix} g_1 & g_2 & g_3 & g_4 \\ g_5 & g_6 & g_7 & g_8 \\ g_9 & g_{10} & g_{11} & g_{12} \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \tag{3.13}$$

ya que se conoce las coordenadas del marcador en el espacio tridimensional, se tienen ocho ecuaciones, una para cada una de las dos coordenadas de cada esquina del marcador y seis grados de libertad. Una solución para G se puede estimar por métodos no iterativos como la transformación lineal directa (**DLT**). Una vez estimada una matriz $\hat{G}$ se reproyectan los puntos **M** en el plano de la imagen obteniendo los puntos $\hat{\mathbf{m}}$, esto es 3.14:

$$\hat{\mathbf{m}} = \hat{G}\mathbf{M} \tag{3.14}$$

y minimizando el error de reproyección $\|\mathbf{m} - \hat{\mathbf{m}}\|$ se obtiene la mejor estimación de G.

3.2.2 Transformaciones entre dos imágenes

3.2.2.1. Geometría epipolar

Al obtener imágenes de una misma escena desde posiciones diferentes, las cuales contengan elementos o puntos comunes, es posible relacionarlas por medio de geometría epipolar. Un ejemplo de posición de las cámaras y los planos de las imágenes respecto a un volumen se muestran en la Figura 3.6.

De un punto **M** en el espacio, la proyección de éste en los planos de dos imágenes distintas son **m** en la primera imagen y **m'** en la segunda. Se dice que son **correspondencias** al ser proyecciones del

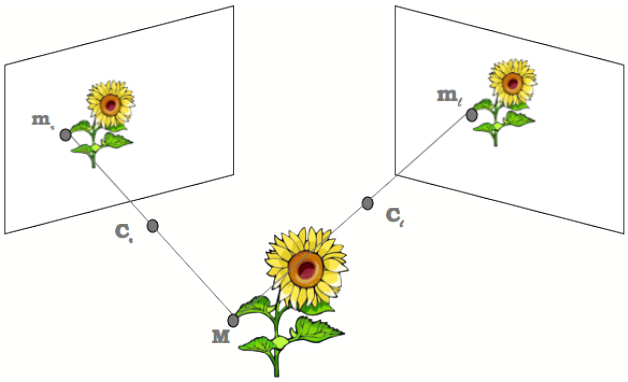


Figura 3.6: Perspectivas de la misma escena, donde la proyección de M tiene un punto m para cada imagen

mismo punto del espacio tridimensional **M**. Los centros de las cámaras que generaron las imágenes se denotan como C y C'. Al unir geométricamente mediante una línea recta los centros de ambas cámaras, intersecan los planos de ambas imágenes, a dichos puntos representados como e y e' se les denomina epipolos. Cuando un conjunto de puntos **M** al proyectarse en una imagen son representados por un solo punto y en la segunda imagen éstos forman una línea, a ésta línea se le denomina epilínea. Considerando como vértices los centros de las cámaras y el punto **M** se puede definir un plano, conocido como epiplano. En la Figura 3.7 lo anterior se aprecia con mayor claridad.

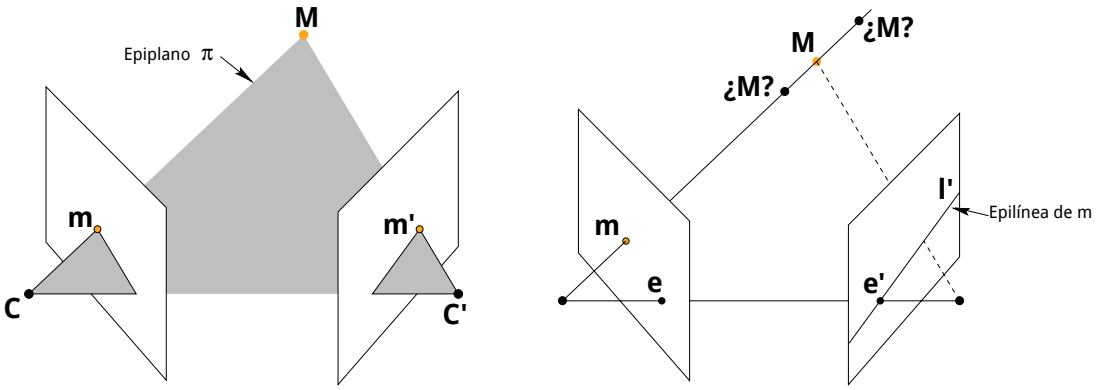


Figura 3.7: Geometría de correspondencia de epipolos

3.2.2.2. Matriz esencial

Definamos los siguientes vectores coplanares: de C a **M** como M_1, de $C\prime$ a **M** como M_2 y la trasformación entre C y $C\prime$ como T. Al encontrarse sobre el mismo plano se describe a M_1 con la Ecuación 3.15:

$$(M_1 - T)^T T \times M_1 = 0 \tag{3.15}$$

Definiendo M_2 en términos de M_1 mediante una traslación y rotación se obtiene la Ecuación 3.16:

$$M_2 = R(M_1 - T) \tag{3.16}$$

$$M_2^T RT \times M_1 = 0 \tag{3.17}$$

$$M_2^T RSM_1 = 0 \tag{3.18}$$

donde S representa el vector de traslación en forma matricial. Finalmente, se agrupa R y S en la Ecuación 3.19.

$$M_2^T EM_1 = 0 \tag{3.19}$$

La matriz esencial $E_{3\times3}$ representa la transformación entre las cámaras por medio de una correspondencia tridimensional. Mediante descomposición de valores singulares de E es posible calcular las posiciones de la cámara para las imágenes a las cuales pertenezcan las correspondencias. Sin embargo, el cálculo de poses arroja algunas soluciones que corresponden a la cámara invertida respecto del plano de la imagen, como se ilustra en la Figura 3.8. En la Subfigura 3.8a las posiciones son las correctas ya que el punto se proyecta en los planos de las imágenes y éstos se ubican frente al centro de su cámara respectiva. Al calcular las poses de las cámaras a partir de la matriz esencial es necesario determinar cuáles son válidas respecto a los puntos proyectados en el espacio tridimensional.

Figura 3.8: Cuatro posibles poses estimadas de la matriz esencial E

3.2.2.3. Matriz fundamental

La matriz fundamental $\mathbf{F}_{3\times3}$ describe las restricciones epipolares disponibles de las correspondencias proyectivas. Para un punto $\mathbf{M}$ del espacio tridimensional y sus proyecciones $\mathbf{m}_1$ y $\mathbf{m}_2$ en los planos de las imágenes para las cámaras C_1 y C_2 se conocen las Ecuaciones de proyección 3.20, recordando que $\mathbf{KP} = G$:

$$\begin{aligned} \mathbf{m}_1 &= G_1\mathbf{M}_1 \\ \mathbf{m}_2 &= G_2\mathbf{M}_2 \end{aligned} \tag{3.20}$$

donde G corresponde al modelo de las cámaras con parámetros intrínsecos K y extrínsecos P. La matriz fundamental F relaciona a los puntos $\mathbf{m}_1$, $\mathbf{m}_2$, las matrices de los modelos de cámara (G_1

y G_2) y la matriz esencial E de la forma como se muestra en la Ecuación 3.21:

$$\begin{aligned} \mathbf{m}_2^T(G_2^{-T}EG_1^{-1})\mathbf{m}_1 = 0 \\ \mathbf{m}_2^T\mathbf{F}\mathbf{m}_1 = 0 \end{aligned} \tag{3.21}$$

Con suficientes correspondencias de puntos en $\mathbb{R}^3$ entre las imágenes se puede determinar **F**, aún sin conocimiento de la matriz esencial. Desarrollando 3.21, que describe la relación de puntos y la matriz fundamental, se conforma la Ecuación 3.22:

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}^T \begin{bmatrix} \mathbf{f}_{11} & \mathbf{f}_{12} & \mathbf{f}_{13} \\ \mathbf{f}_{21} & \mathbf{f}_{21} & \mathbf{f}_{23} \\ \mathbf{f}_{31} & \mathbf{f}_{32} & \mathbf{f}_{33} \end{bmatrix} \begin{bmatrix} x'_i \\ y'_i \\ 1 \end{bmatrix} = 0 \tag{3.22}$$

del cual se obtiene la Ecuación 3.23 para un par de puntos:

$$x_i x'_i \mathbf{f}_{11} + x_i y'_i \mathbf{f}_{12} + x_i \mathbf{f}_{13} + y_i x'_i \mathbf{f}_{21} + x'_i y'_i \mathbf{f}_{22} + y_i \mathbf{f}_{23} + x'_i \mathbf{f}_{31} + y'_i \mathbf{f}_{32} + \mathbf{f}_{33} = 0 \tag{3.23}$$

Al considerar un conjunto de correspondencias para un par de imágenes los valores para $\mathbf{f}_{m\times n}$ son comunes, por lo que se pueden organizar las ecuaciones 3.23 para cada correspondencia en un sistema de ecuaciones y así determinar los valores para **f**.

Con ocho correspondencias hay una solución única la cual se obtiene linealmente, como fue presentado en [20]. Para un cálculo de **F** más robusto, el método presentado en [69] divide la imagen en una retícula de 8×8 y aplica el método **RANSAC** [15] seleccionando ocho correspondencias, una correspondencia por casilla, y calculando **F** por el método de [20] para los

puntos seleccionado considerando el error de **F** con respecto al conjunto total de correspondencias. Empleando este método iterativo es posible determinar una matriz **F** robusta que además descarte las correspondencias incorrectas marcándolas como atípicas u *outliers*.

La matriz esencial E puede ser obtenida de la matriz fundamental **F** mediante la Ecuación 3.24.

$$E = \mathbf{K}^T \times \mathbf{F} \times \mathbf{K} \tag{3.24}$$

3.2.3 Homografía

La transformación proyectiva u homografía, denotada H, relaciona las coordenadas de pixeles de dos imágenes. Considerando un punto $\mathbf{m} = (u, v, 1)$ en la imagen I y otro punto $\mathbf{m'} = (u', v', 1)$ en la imagen I' la homografías es una matriz que cumple con las Ecuaciones 3.25 y 3.26.

$$\mathbf{m'} = H\mathbf{m}$$
$$\begin{bmatrix} u\prime \\ v\prime \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{21} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \tag{3.25}$$

$$\begin{aligned} u\prime &= \frac{h_{11}u + h_{12}v + h_{13}}{h_{31}u + h_{32}v + h_{33}} \\ v\prime &= \frac{h_{21}u + h_{22}v + h_{23}}{h_{31}u + h_{32}v + h_{33}} \end{aligned} \tag{3.26}$$

Para eliminar un grado de libertad se considera que $h_{33} = 1$. Estarán relacionadas por una homografía H un par de imágenes que observen el mismo plano de una escena con diferente ángulo de cámara.

Con cuatro puntos y sus correspondientes en las imágenes, se pueden calcular las ocho incógnitas de H.

3.2.4 Triangulación

Existen diversas formas de recuperar información 3D de una escena: Recuperar la profundidad de un punto mediante su distancia respecto al plano de la imagen, recuperar sus coordenadas tridimensionales o recuperar las profundidades relativas entre dos puntos. A partir de una imagen sólo podemos inferir la posición del rayo óptico M, no la posición de un punto tridimensional $\mathbf{M}$. Con dos imágenes y la correspondencia de $(\mathbf{m}, \mathbf{m\prime})$ podemos intersecar sus respectivos rayos ópticos en un sistema de coordenadas común y determinar $\mathbf{M}$. En ausencia de ruido la triangulación es trivial, pero en general existe ruido y los rayos ópticos no se intersecan, por lo que es necesario estimar el punto de intersección. Un método de triangulación lineal es descrito en [19] basado en mínimos cuadrados y suponiendo ruido gaussiano.

El punto $\mathbf{M}$ es visible en dos imágenes, por lo que $\mathbf{m}_1$ y $\mathbf{m}_2$ son las proyecciones correspondientes para cada imagen. Se deben conocer las matrices de posición P_1 y P_2 para cada imagen así como la matriz fundamental F del par de imágenes. Ya que $\mathbf{m}_1 = \mathbf{K}P\mathbf{M}$, al expresar en coordenadas homogéneas a $\mathbf{m}_1 = s[u, v, 1]^T$ donde $[u, v]$ son las coordenadas de $\mathbf{m}_1$ en el plano y s es un factor de escalamiento, obtenemos las Ecuaciones 3.27:

$$\begin{aligned} su &= \mathbf{p}_1^T\mathbf{M} \\ sv &= \mathbf{p}_2^T\mathbf{M} \\ s &= \mathbf{p}_3^T\mathbf{M} \end{aligned} \tag{3.27}$$

donde $\mathbf{p}_i$ corresponden a la fila i de la matriz $\mathbf{P}$. Eliminando s podemos simplificar las Ecuaciones 3.27 como se muestra en las Ecuaciones 3.28:

$$\begin{aligned} u\mathbf{p}_3^T\mathbf{M} &= \mathbf{p}_1^T\mathbf{M} \\ v\mathbf{p}_3^T\mathbf{M} &= \mathbf{p}_2^T\mathbf{M} \end{aligned} \tag{3.28}$$

obteniendo cuatro ecuaciones lineales para las coordenadas de $\mathbf{M}$, denotándolas como $A\mathbf{M} = 0$ para una matriz A de 4×4. Esta ecuación define $\mathbf{M}$ para un factor de escala indeterminado y se desea una solución diferente de cero para $\mathbf{M}$. Para resolver por el método lineal de mínimos cuadrados se asume que el punto tridimensional es $\mathbf{M} = (x, y, z, 1)^T$ en coordenads homogéneas, por lo que el sistema $A\mathbf{M} = 0$ se reduce a un conjunto de cuatro ecuaciones no homogéneas de tres incógnitas, quedando $A\mathbf{M} = B$ donde $A_{4\times3}$, $\mathbf{M}_{3\times1}$, y $B_{4\times1}$. En la Ecuación 3.29 se desarrolla $A\mathbf{M} = B$, con las proyecciones de $\mathbf{M}$ como ($\mathbf{m}_1$,$\mathbf{m}_2$) y las poses de las cámaras ($\mathbf{P}1$,$\mathbf{P}2$).

$$A\mathbf{M} = B$$

$$\begin{bmatrix} \mathbf{m}_{1x}\mathbf{P}1_{2,0} - \mathbf{P}1_{0,0} & \mathbf{m}_{1x}\mathbf{P}1_{2,1} - \mathbf{P}1_{0,1} & \mathbf{m}_{1x}\mathbf{P}1_{2,2} - \mathbf{P}1_{0,2} \\ \mathbf{m}_{1y}\mathbf{P}1_{2,0} - \mathbf{P}1_{1,0} & \mathbf{m}_{1y}\mathbf{P}1_{2,1} - \mathbf{P}1_{1,1} & \mathbf{m}_{1y}\mathbf{P}1_{2,2} - \mathbf{P}1_{1,2} \\ \mathbf{m}_{2x}\mathbf{P}2_{2,0} - \mathbf{P}2_{0,0} & \mathbf{m}_{2x}\mathbf{P}2_{2,1} - \mathbf{P}1_{0,1} & \mathbf{m}_{2x}\mathbf{P}2_{2,2} - \mathbf{P}2_{0,2} \\ \mathbf{m}_{2y}\mathbf{P}2_{2,0} - \mathbf{P}2_{1,0} & \mathbf{m}_{2y}\mathbf{P}2_{2,1} - \mathbf{P}1_{1,1} & \mathbf{m}_{2y}\mathbf{P}2_{2,2} - \mathbf{P}2_{1,2} \end{bmatrix} \mathbf{M} = \begin{bmatrix} \mathbf{m}_{1x}\mathbf{P}1_{2,3} - \mathbf{P}1_{0,3} \\ \mathbf{m}_{1y}\mathbf{P}1_{2,3} - \mathbf{P}1_{1,3} \\ \mathbf{m}_{2x}\mathbf{P}2_{2,3} - \mathbf{P}2_{0,3} \\ \mathbf{m}_{2y}\mathbf{P}2_{2,3} - \mathbf{P}2_{1,3} \end{bmatrix} \tag{3.29}$$

Este método puede ser mejorado mediante iteraciones caracterizadas por añadir pesos a las ecuaciones lineales y ajustando hasta obtener un error de ajuste previamente definido, también propuesto en [19]. Los pesos $(w1, w2)$ son incorporados de la forma como se muestra en la Ecuación

3.30.

$$\begin{bmatrix} (\mathbf{m}_{1x}\mathbf{P}1_{2,0} - \mathbf{P}1_{0,0})/w1 & (\mathbf{m}_{1x}\mathbf{P}1_{2,1} - \mathbf{P}1_{0,1})/w1 & (\mathbf{m}_{1x}\mathbf{P}1_{2,2} - \mathbf{P}1_{0,2})/w1 \\ (\mathbf{m}_{1y}\mathbf{P}1_{2,0} - \mathbf{P}1_{1,0})/w1 & (\mathbf{m}_{1y}\mathbf{P}1_{2,1} - \mathbf{P}1_{1,1})/w1 & (\mathbf{m}_{1y}\mathbf{P}1_{2,2} - \mathbf{P}1_{1,2})/w1 \\ (\mathbf{m}_{2x}\mathbf{P}2_{2,0} - \mathbf{P}2_{0,0})/w2 & (\mathbf{m}_{2x}\mathbf{P}2_{2,1} - \mathbf{P}1_{0,1})/w2 & (\mathbf{m}_{2x}\mathbf{P}2_{2,2} - \mathbf{P}2_{0,2})/w2 \\ (\mathbf{m}_{2y}\mathbf{P}2_{2,0} - \mathbf{P}2_{1,0})/w2 & (\mathbf{m}_{2y}\mathbf{P}2_{2,1} - \mathbf{P}1_{1,1})/w2 & (\mathbf{m}_{2y}\mathbf{P}2_{2,2} - \mathbf{P}2_{1,2})/w2 \end{bmatrix} \mathbf{M} = \\ = \begin{bmatrix} (\mathbf{m}_{1x}\mathbf{P}1_{2,3} - \mathbf{P}1_{0,3})/w1 \\ (\mathbf{m}_{1y}\mathbf{P}1_{2,3} - \mathbf{P}1_{1,3})/w1 \\ (\mathbf{m}_{2x}\mathbf{P}2_{2,3} - \mathbf{P}2_{0,3})/w2 \\ (\mathbf{m}_{2y}\mathbf{P}2_{2,3} - \mathbf{P}2_{1,3})/w2 \end{bmatrix} \tag{3.30}$$

3.2.5 Error de reproyección

Dado un punto en una imagen y su punto correspondiente en la escena tridimensional es posible determinar la diferencia que existe entre el punto original de la imagen y la proyección del punto tridimensional en dicho plano. Esto se conoce como error de reproyección, para lo cual se aplica el modelo de cámara oscura, los parámetros intrínsecos y extrínsecos, en la Ecuación 3.6 con el punto tridimensional.

Un punto $\mathbf{m}$ visto en la imagen I_n se denota como $\mathbf{m}_n$ y su reproyección está dada por $\hat{\mathbf{m}}_n = MP_n$ de forma que el error de reproyección se obtiene mediante $\|\mathbf{m}_n - \hat{\mathbf{m}}_n\|$.

3.3 Emparejamiento de imágenes con visión por computadora

En esta sección se incluyen los fundamentos que permiten relacionar dos imágenes para aplicar los métodos y técnicas planteadas en las secciones anteriores. Estos son: detección de puntos característicos, descripción de dichos puntos y emparejamiento.

3.3.1 Calibración de la cámara de forma automática

Una cámara cuyos parámetros intrínsecos son conocidos se dice que está calibrada. Existen diversos algoritmos para la calibración de la cámara. Un ejemplo de estos algoritmos es el de Zhang *et al.* presentado en [70], donde se describe un método eficiente basado en el análisis de dos o más vistas de un patrón de calibración (un tablero de ajedrez) de dimensiones conocidas. Estos algoritmos están compuestos de: cálculo de la homografía, establecer las restricciones de homografía y resolver el sistema lineal homogéneo para encontrar la matriz $\mathbf{K}$ y cálculo de la matriz $\mathbf{R}$ y el vector $\mathbf{t}$.

El modelo clásico para la calibración consiste en determinar la matriz de proyección para el modelo de cámara oscura, mediante puntos de control tridimensionales conocidos. Para los puntos de control conocidos se emplea una malla con patrón de tablero de ajedrez asimétrico, el cual se muestra en la Figura 3.9.

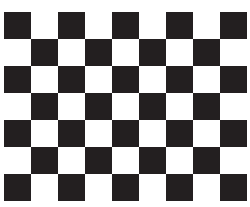


Figura 3.9: Malla con patrón de tablero de ajedrez

Los puntos de referencia **M** son conocidos en algún sistema de referencia tridimensional y sus proyecciones **m** detectadas. Ya que **P** tiene 11 variables independientes, con por lo menos seis correspondencias de 2D a 3D es posible determinar la matriz de proyección. Una vez que **P** es conocida se descompone en parámetros intrínsecos y extrínsecos.

El procedimiento de Zhang *et al.* [70] consiste en obtener un conjunto de fotografías del patrón. Ya que se detectan los vértices se calculan las homografías, con este cálculo se realiza una estimación de valores para los parámetros intrínsecos y extrínsecos de la cámara. Finalmente con los valores estimados se realiza un ajuste no lineal por Levenbeg-Marquardt para obtener la solución final junto con los coeficientes de la dispersión de la lente.

3.3.2 Detección puntos característicos

Los puntos característicos son pixeles (u, v) en una imágen que contienen información que ayuda a describir la imagen. La detección de estos puntos pueden ser por esquinas, cambios de iluminación o color con los píxeles vecinos, por mencionar algunos. Los algoritmos de detección de puntos deben garantizar repetibilidad, que se detecten los mismos puntos si aparecen en distintas imágenes, distintivos, que se pueda tener la certeza de la correspondencia, e invariantes a cambios geométricos e iluminación. Los puntos de interés deben ser robustos a traslaciones, rotaciones, escalamientos y proyecciones. Además de estas propiedades es de gran importancia la velocidad de detección y descripción para las tareas con tiempo limitado, siendo esta característica particularmente resaltada en la literatura.

La detección y descripción de puntos característicos en imágenes son problemas ampliamente abordados en la literatura. Los primeros trabajos de identificación de puntos característicos son los detectores de esquinas y bordes propuestos por Harris y Stephens en el artículo [18]. Los algoritmos

más reconocidos en el área son **SIFT** presentado por David G. Lowe en el artículo [29] y **SURF** por Herbert Bay *et. al* en [3]. **SIFT** y **SURF** están basados en histogramas de gradientes locales de la imagen. Estos algoritmos, además de detectar puntos, utilizan las mismas características que permitieron detectarlos para describir dichos puntos. **SIFT** presenta alta calidad en rasgos distintivos e invarianza a costa de un alto costo computacional, mientras que **SURF** compensa esto último, siendo más veloz que **SIFT**.

Existen sin embargo otros algoritmos de detección diseñados para problemas con restricciones de tiempo real, tal es el caso de **FAST** [48], **BRISK** [27] y **FREAK** [1]. Éstos se basan en un análisis inmediato de los píxeles vecinos, en cuanto a cambios de luminancia.

El algoritmo **FAST** considera un círculo Bresenham de 16 píxeles alrededor del píxel candidato, evaluando como candidato a todos los pixeles de la imágen. Cada píxel vecino del círculo puede ser más oscuro, de igual luminancia o con mayor brillo respecto al píxel evaluado. El pixel candidato se clasifica como una esquina si existe un conjunto de 12 píxeles en el círculo con valores más claros (u oscuros). El píxel candidato y el círculo Bresenham considerados por el algoritmo de **FAST** se muestran en la Figura 3.10.

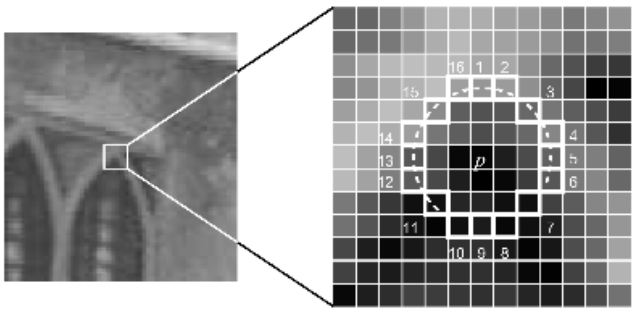


Figura 3.10: Píxel candidato y vecindario considerado por **FAST**

Rosten y Drummond, en el artículo [48] donde presentan **FAST**, determinan que se obtiene mejor

resultado con un círculo Bresenham de nueve píxeles de radio. La desventaja de este algoritmo es que no es robusto a altos niveles de ruido y que depende mucho del umbral con el que se determinan si las diferencias entre píxeles son significativas.

De forma semejante, el detector **BRISK** [27] busca dar invarianza a la escala para lo cual emplea una pirámide de la imagen. Las pirámides consisten de n octavas c_i y n interoctavas d_i, siendo recomendado por los autores $n = 4$. Las octavas se obtienen submuestreado la resolución de la imagen original c_0 a la mitad y así progresivamente. Las interoctavas se obtienen submuestreando c_0 a 1.5 veces la resolución original, siendo la escala t entonces $t(c_i) = 2^i$ y $t(d_i) = 2^i \times 1.5$. Las octavas e interoctavas se muestran en la Figura 3.11.

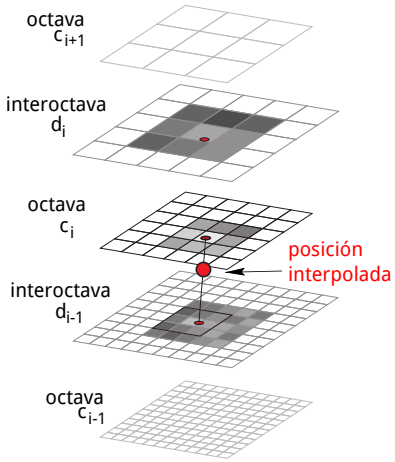


Figura 3.11: Espacio construido para detección de puntos característicos por **BRISK**

La detección de puntos se realiza por medio de *FAST* con la plantilla de círculo Bresenham de 16 píxeles y considerando la condición de 9 píxeles consecutivos diferentes al píxel candidato. Esto se realiza en todas los niveles de la pirámide, dando seguimiento para asegurar que la detección sea continua, la interpolación entre niveles se realiza por medio de un ajuste cuadrático.

3.3.3 Descripción de puntos característicos

La descripción de puntos permite seleccionar un conjunto de características de los píxeles vecinos y de la imagen que permitan representar dicho punto. En un contexto práctico, donde la velocidad y recursos computacionales son limitados, los descriptores deben tener altas tasas de reconocimiento y ser tan computacionalmente ligeros como sea posible. Para tal fin, se consideran detectores binarios contra aquellos que emplean punto flotante. A estos últimos pertenecen **SURF**, que emplea un vector de 64 valores de punto flotante, es decir 256 *bytes*, y **SIFT** con 128 valores por punto. Los descriptores binarios aprovechan el cálculo de la distancia de Hamming para realizar los emparejamientos, opuesto a los descriptores flotantes para los que se prefieren distancias Euclidianas.

Calonder *et al.* en su artículo [8] fueron los primeros en presentar los descriptores binarios con el algoritmo de *BRIEF*. Éste define una prueba τ para un parche o plantilla p de tamaño $S \times S$ considerando la intensidad suavizada del píxel como $p(x)$ donde $x = (u, v)^T$, como se indica en la Ecuación 3.31:

$$\tau(p; , x, y) := \begin{cases} 1 & si\ p(x) < p(y) \\ 0 & \text{cualquier otro} \end{cases} \tag{3.31}$$

La cadena de descripción binaria de longitud n_d depende de la comparación entre pares seleccionados, quedando expresa según la Ecuación 3.32:

$$f_{n_d}(p) := \sum_{1 \leq i \leq n_d} 2^{i-1} \tau(p; x_i, y_i) \tag{3.32}$$

El vecindario del píxel se suaviza con un *kernel* gaussiano de 9×9 para quitar la sensibilidad al ruido, aumentando la estabilidad y repetibilidad. Eligiendo un conjunto de n_d pares (x, y)-ubicados

se definen las pruebas binarias. La plantilla para los posibles n_d pares de píxeles vecinos (x_i, y_i) que generan la cadena se muestra en la Figura 3.12.

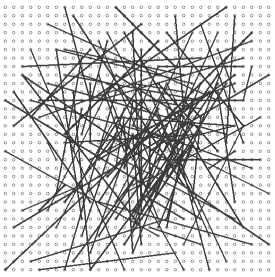


Figura 3.12: Plantilla gaussiana con $\sigma^2 = \frac{1}{25}S^2$ para cálculo de descriptor **BRIEF**

La desventaja de *BRIEF* es ser variante a rotaciones, aunque tolera rotaciones de escena entre 10 a 15 grados. Para una cadena de longitud de 256 bits, que requiere 32 *bytes* para almacenar, el emparejamiento entre imágenes con desplazamientos cortos da resultados casi óptimos, mientras que para el resto de las trasformaciones de la escena los autores recomiendan la longitud de 512 bits en 64 *bytes*.

El detector de puntos **BRISK** mencionado anteriormente también cuenta con una etapa de descripción. El descriptor *BRISK* presentado por Leutenegger *et al.* en el artículo [27] es una alternativa a *BRIEF* con tolerancia a transformaciones y distorsiones de la imagen. A diferencia de *BRIEF*, este descriptor usa muestreo determinista y emplea menor número de píxeles vecinos para la comparación de intensidades. Para este descriptor se obtiene una cadena de 512 bits.

Al conocer la dirección de un punto característico, *BRISK* calcula los descriptores de manera normalizada a orientación. La plantilla de muestreo para BRISK se muestra en la Figura 3.13, los círculos azules denotan los puntos de muestreo mientras que los rojos corresponden a la desviación estándar del *kernel* gaussiano empleado para suavizar los valores de intensidad. El suavizado gaussiano es proporcional a la distancia de los puntos con respecto al punto característico central.

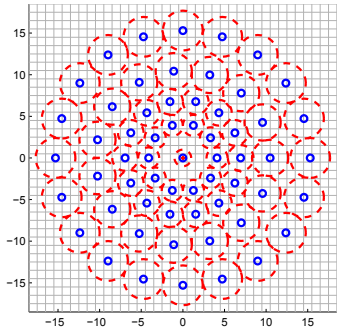


Figura 3.13: Plantilla de muestreo en **BRISK** de escala $t = 1$ con $N = 60$ centros de muestreo

Para cada uno de los puntos de la plantilla de muestreo sobre el vecindario del punto, se procesan los gradientes de intensidad locales para determinar la dirección del punto característico. Según la dirección obtenida por la plantilla, se rota alrededor del píxel para finalmente realizar las comparaciones de brillo con orientación normalizada, las cuales conforman al descriptor. Con esto, los puntos lejanos estiman la dirección del punto característico mientras que los que se encuentran próximos son empleados para formar el descriptor después de rotar la plantilla.

Del emparejamiento y descripción en el contexto de la navegación con robots móviles, Schmidt *et al.* en el artículo [52] realizan una comparativa considerando que para esta aplicación los detectores deben ser robustos a los cambios de escala y transformaciones afines. La combinación del detección de puntos por medio de *FAST* y descripción por medio de *BRIEF* son hasta el momento los mejores algoritmos para aplicaciones con restricción de tiempo real.

3.4 Pose y posición del VANT

3.4.1 Odometría

La odometría consiste en estimar la posición en la que se encuentra un robot móvil en un momento determinado, empleando los sensores de éste. En el caso de un **VANT** tipo **VTOL** cuadricóptero, partiendo de los ángulos de navegación obtenidos con los sensores inerciales es posible determinar su posición respecto a un marco de referencia o sistema de coordenadas con origen en su posición inicial al despegue. La distancia recorrida por el **VANT** se obtiene a partir de la lectura de velocidades (vx, vy, y vz) y el intervalo de tiempo Δt transcurrido entre mediciones, descrito de la forma mostrada en la Ecuación 3.33.

$$D = V(\Delta t) = \begin{bmatrix} vx(\Delta t) \\ vy(\Delta t) \\ vz(\Delta t) \end{bmatrix} \tag{3.33}$$

La distancia D permite actualizar la posición $\mathbf{P}_{actual}$ según la Ecuación 3.34, donde (Rz, Ry, Rx) corresponden a las matrices de rotación presentadas en las Ecuaciones 3.2, 3.3 y 3.4.

$$\mathbf{P}_{actual} = \mathbf{P}_{anterior} + Rz \times Ry \times Rx \times D \tag{3.34}$$

Dependiendo de la frecuencia con que se realizan las mediciones, es la precisión de la odometría. La principal desventaja de la odometría consiste en la acumulación de errores, provocando inexactitud al estimar la posición del robot.

Al calcular la odometría, se deben diferenciar entre los errores sistemáticos, causados por la

cinemática del robot, y los no sistemáticos producidos por el medio donde se desplaza el robot. En el caso del vuelo de un **VANT**, entre los primeros se consideran la inercia sufrida al cambiar la dirección de desplazamiento, así como la deriva que existe al mantenerse estático en una posición fija; los no sistemáticos son provocados por los factores externos al robot, como las condiciones climatológicas, principalmente viento.

La posición del robot, por lo tanto se puede describir con una *elipse de error*, la cual indica la región de incertidumbre en la posición actual. La elipse crece con respecto a la distancia recorrida hasta encontrar una posición conocida o absoluta, con la cual se reinicia el tamaño del elipse. Una gráfica de elipses de incertidumbre para el desplazamiento del **VANT** se muestra en la Figura 3.14.

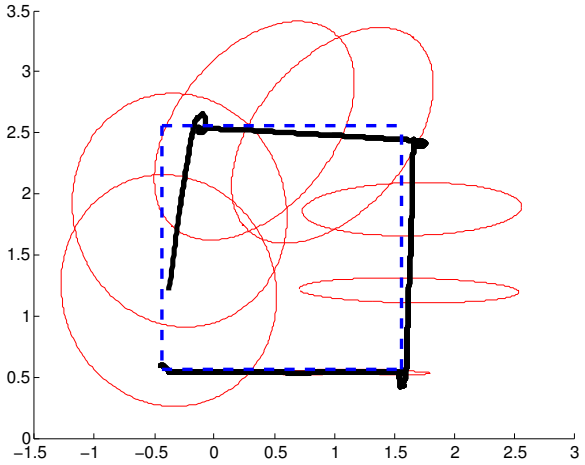


Figura 3.14: Incertidumbre de odometría representada por elipses para un trayecto cuadrado realizado por el **VANT**

El desplazamiento del **VANT** se produce al modificar respecto a la posición horizontal de estabilidad el ángulo de cabeceo (*pitch*) para desplazarse sobre su eje x y del ángulo de alabeo (*roll*) respecto al eje y. Para modificar la guiñada, ángulo de giro en el eje z, el **VANT** debe

encontrarse estático para reducir los errores de odometría.

3.4.2 Localización del robot

Para la localización absoluta del robot en interiores es necesario emplear marcadores de fácil distinción asociados con una posición en Σ_W conocida. Lim *et al.* en su artículo [28] proponen emplear marcadores artificiales similares a los empleados en realidad aumentada. Cada marcador es usado para actualizar su posición y reducir el error entre su posición real y la estimada.

3.5 Conclusiones

En este capítulo se presentaron los fundamentos teóricos necesarios para la recuperación de características tridimensionales por medio de visión monocular. Para todo procesamiento de fotografías es indispensable la calibración de la cámara para conocer sus valores intrínsecos y los coeficientes de distorsión de la lente.

Por medio de la homografía del marcador visual artificial, el **VANT** puede estimar su posición real con respecto a la escena. La posición de la aeronave es determinada por medio de odometría, a la par que adquiere imágenes. Estas imágenes son emparejadas por medio de la detección y descripción de puntos característicos.

La relación entre las imágenes permite calcular las matrices esencial y fundamental, con las cuales posteriormente se lleva a cabo la recuperación de características tridimensionales. La triangulación iterativa, que procede del método lineal, permite aproximar los puntos tridimensionales calculados a su posición real, reduciendo el error de reproyección.

4

Propuesta de solución

En este capítulo se presenta el detalle de la implementación de la solución propuesta para la reconstrucción de la fachada de un edificio a partir de un conjunto de fotografías obtenidas por un vehículo aéreo no tripulado (**VANT**) tipo **VTOL** equipado con una cámara monocular.

4.1 Modelo general de la propuesta

La propuesta de solución para la construcción del modelo tridimensional requiere de una correcta comunicación inalámbrica entre el **VANT** que toma las fotografías con su cámara embarcada y una computadora en tierra que procesará la información para obtener dicho modelo. El proceso completo de captura de imágenes y reconstrucción se describe a continuación:

1. El usuario coloca un marcador visual artificial de dimensiones conocidas en la fachada de una edificación, que servirá como punto de referencia para el proceso de reconstrucción.

2. Se ubica al **VANT** frente al marcador, de forma que en el momento del despegue el marcador

esté en campo de visión de la cámara del **VANT**, sin oclusiones.

3. En la computadora se inicia el visualizador de modelo tridimensional en el cual se va mostrando el modelo tridimensional conforme se construye.

4. De igual forma, en la computadora se inicia la aplicación que controla al **VANT** y procesa las imágenes para la construcción del modelo tridimensional.

5. Al iniciar la aplicación el **VANT** despega y se ubica de forma automática frente al marcador, a la misma altura.

6. Una vez ubicado, el **VANT** realiza una trayectoria predeterminada en la cual va adquiriendo fotografías que envía a la computadora.

7. Se procesan las imágenes para construir un modelo tridimensional, el cual se puede ir apreciando en el visualizador.

8. Una vez terminada la trayectoria, el **VANT** aterriza dando por terminada la construcción del modelo tridimensional de la fachada.

Como resultado del proceso de captura y reconstrucción, se obtiene un modelo tridimensional en el visualizador en forma de nube de puntos, el cual puede ser explorado por el usuario.

Es importante hacer hincapié que la aportación de la propuesta consiste en **la generación del modelo del edificio en línea**, mientras se encuentra en vuelo el **VANT**. Los algoritmos y métodos considerados principalmente han sido propuestos para robots terrestres o aplicaciones en visión por computadora, así como para la reconstrucción tridimensional fuera de línea.

La recuperación de características tridimensionales a partir de fotografías requiere de la solución de dos subproblemas distintos: conocer la posición de la cámara al momento de adquirir la imagen

y el reconocimiento de puntos que aparecen en diferentes imágenes para triangular su posición tridimensional. El **VANT** cuenta con sensores que le permiten estimar su posición, la cual se considera es la misma para la cámara monocular. Ésta relación permite emplear la posición estimada por la plataforma aérea para resolver el problema de recuperar características tridimensionales. La propuesta para el subproblema de localización y navegación en trayectorias se resuelve por medio de la colocación de marcadores artificiales en la fachada, de forma que el **VANT** detecta, identifica y estima la posición respecto a éste. Para dar solución al subproblema de generar un modelo tridimensional en línea se consideran algoritmos de visión por computadora que fueron desarrollados para reconstrucción en tiempo real.

Los componentes identificados para solucionar éste problema se dividieron en los componentes que operan el movimiento y posición del **VANT** y los componentes del sistema de visión por computadora propiamente dicho. En los primeros se incluyen el control de vuelo de la plataforma aérea, la estimación de posición y el seguimiento de trayectorias. El sistema de visión por computadora considera el análisis de las imágenes obtenidas tanto como para detección de marcadores así como extracción de información relevante, emparejamiento y recuperación de las características tridimensionales de la fachada y reconstrucción de un modelo tridimensional. La propuesta de solución es relevante al considerar éstos componentes como estrechamente relacionados para lograr un procesamiento en línea, de tal forma que se demuestra la viabilidad y madurez alcanzada por las técnicas de visión por computadora aplicadas al campo de la robótica móvil.

En el diagrama de la Figura 4.1 se muestran los módulos que conforman al sistema, los cuales se desarrollan a lo largo de este capítulo.

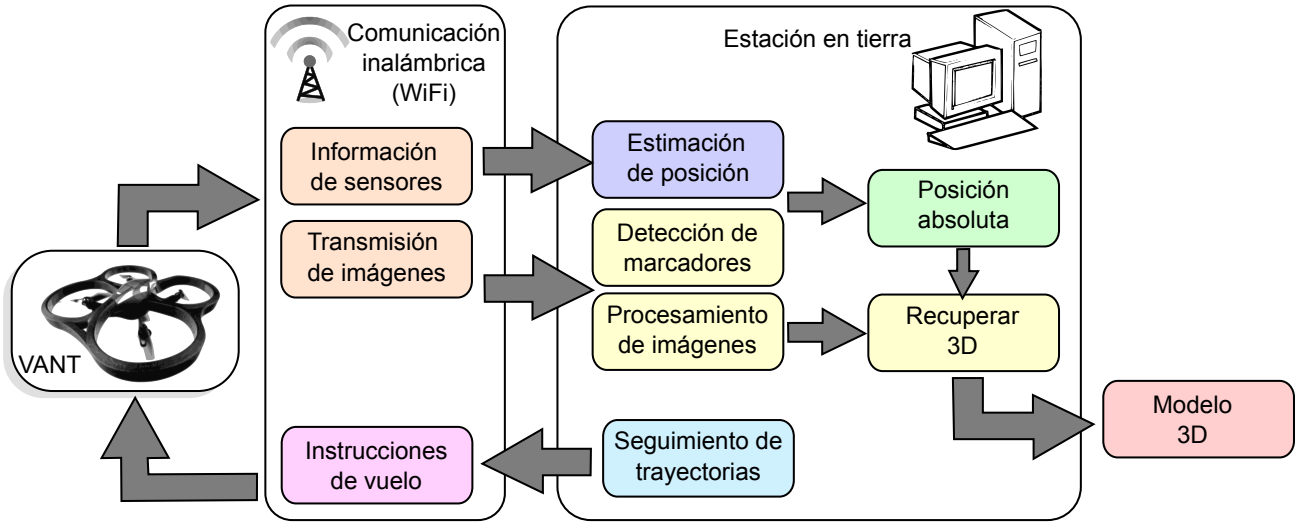


Figura 4.1: Diagrama de la propuesta para la generación de un modelo 3D con imágenes obtenidas por un **VANT**

4.2 Comunicación inalámbrica

Para la solución propuesta, el **VANT** debe establecer comunicación inalámbrica via *WiFi* con la computadora o estación en tierra. La información que transmite la aeronave es la información de los sensores de desplazamiento así como las imágenes que adquiere por medio de la cámara monocular. La información de los sensores permite estimar la posición del **VANT** mediante odometría. Las imágenes son enviadas a la computadora para ser procesadas por el sistema de visión, el cual se describe más adelante. Es también por medio de esta vía de comunicación que se envían instrucciones de vuelo al **VANT**, las cuales consisten en velocidades de desplazamiento sobre alguno de los ejes principales del aeronave.

4.3 Odometría

Se emplea la información de los sensores inerciales del **VANT**, así como la información visual, para determinar su posición respecto a la escena, la fachada y el referencial absoluto asociado a ésta.

Durante esta etapa, se restringe el desplazamiento del **VANT** para que se efectúe únicamente sobre los ejes (x, y) de su referencial local, como se mostró en la Figura 3.2. Esta consideración permite que durante el desplazamiento no se altere el ángulo de guiñada (*yaw*) sobre el eje z, dado que resulta ineficiente pues induce una enorme inercia durante el desplazamiento que empuja al **VANT** lejos del camino deseado. Se asume también que el **VANT** cuenta con control de estabilidad, el cual se encarga de reducir la deriva al estar en vuelo sobre un punto fijo (vuelo estacionario).

Para obtener la estimación de posición por odometría, a partir de los datos de los sensores inerciales, se utiliza el Algoritmo 1, que obtiene la posición con respecto a un referencial centrado en la ubicación del **VANT** al momento del despegue, integrando numéricamente la velocidad del vehículo.

Algoritmo 1 Odometría, posición por estimación con sensores inerciales

Entrada: Velocidades (vx, vy, vx), ángulos (ϕ, θ, ψ), altura h
Salida: $P = (x, y, z)$ Posición estimada

1: $(t_{anterior}, t_{actual}) \leftarrow$ Iniciar tiempo
2: $P = (0, 0, 0) \leftarrow$ Iniciar posición actual en origen
3: **mientras VANT** esté en vuelo **hacer**
4: $Rx = R(x, \phi) \leftarrow$ Rotación en x, de la ecuación 3.2
5: $Ry = R(y, \theta) \leftarrow$ Rotación en y, de la ecuación 3.3
6: $Rz = R(z, \psi) \leftarrow$ Rotación en z, de la ecuación 3.4
7: $t \leftarrow$ Obtener tiempo inmediato
8: $dt = t_{actual} - t_{anterior}$
9: $M = [vx * dt, vy * dt, vz/dt]$ (Ecuación 3.33)
10: $P = P + (Rz * Ry * Rx * M) \leftarrow$ Actualizar posición
11: $t_{anterior} \leftarrow t_{actual}$
12: **fin mientras**

La estimación de la posición del **VANT** con los sensores inerciales no está exenta de errores, la incertidumbre entre la posición real y la estimada depende del número de observaciones. Para evitar que las incertidumbres crezcan indefinidamente es necesario determinar posiciones absolutas en la escena de las cuales se conozca su posición. La aproximación es semejante a la odometría visual la cual estima el desplazamiento del robot mediante el análisis de imágenes y, al encontrar

una característica previamente asignada cuya posición es conocida (un marcador artificial), ajusta la posición estimada para eliminar los errores. Con tal aproximación es posible conocer la posición real del **VANT** y corregir la odometría, siendo la detección de marcadores de gran importancia.

4.4 Control de vuelo y seguimiento de trayectoria

Definimos como control de vuelo a las instrucciones que se transmiten al **VANT** para su desplazamiento. Ya que las instrucciones de movimiento corresponden a velocidades para que la aeronave se desplace a la distancia deseada, es necesario llevar registro del tiempo de movimiento. Con el tiempo y la velocidad se calcula la distancia recorrida por el **VANT**, sin embargo es importante considerar la inercia del sistema, que produce un desplazamiento adicional en el frenado. Por ello, la posición real del **VANT** no puede ser determinada únicamente con los sensores de velocidad, pues se obtiene solamente una aproximación de posición.

Considerando que son fijas las distancias propuestas a las que se desplaza el **VANT**, es posible realizar una calibración para dichas distancias logrando que la posición final considere el efecto de inercia, logrando el desplazamiento deseado. Una calibración satisfactoria que permita reducir el error entre el desplazamiento deseado y el real es indispensable para el seguimiento de trayectorias del **VANT**.

La trayectoria que sigue el **VANT** para la obtención de fotografías de la fachada es fija y centrada en el marcador colocado por el usuario previo al despegue del aeronave. Se debe ubicar la aeronave preferentemente frente del marcador inicial de forma que al despegue pueda ser detectado sin que existan oclusiones y exista una distancia menor a tres metros de distancia entre éste y el **VANT**. La navegación entre marcadores permite reducir los errores e incertidumbre de la odometría por lo que en cada detección se actualiza la posición estimada del **VANT** con la absoluta respecto de la

fachada.

En la Figura 4.2 se muestra una trayectoria propuesta para el **VANT**, en ésta se indica el sistema de coordenadas local del vehículo con el eje z paralelo a la altura de la fachada. Dado que el usuario requiere colocar los marcadores en la fachada, se asume que se conoce la altura aproximada del primer marcador respecto del suelo o de la superficie donde despega el **VANT**. Una vez realizada la detección de marcador inicial se desplaza a la posición de inicio de trayectoria la cual se ubica a una distancia calculada con la altura máxima, también establecida previamente.

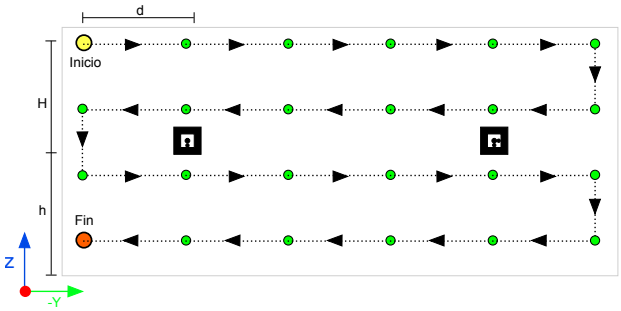


Figura 4.2: Trayectoria propuesta *a priori* para la adquisición de fotografías por medio del **VANT**

La detección de un segundo marcador durante la trayectoria, ubicado a una distancia conocida del primero, permite al **VANT** saber la posición real a la que se encuentra con respecto a la fachada y corregir la odometría. Una vez ubicado en la posición de inicio, el **VANT** se desplaza y adquiere fotografías en posiciones fijas, marcadas una circunferencia verde en la Figura 4.2. En estas posiciones se adquieren imágenes con ángulos de guiñada $(-\frac{\pi}{4}, 0, \frac{\pi}{4})$ en el eje z del referencial fijado en el **VANT**. Estos ángulos, mostrados en la Figura 4.3, son coherentes con lo reportado por Rachmielowski *et al.* en su artículo [43] para obtener imágenes con elementos originales útiles para generar modelos tridimensionales en tiempo real.

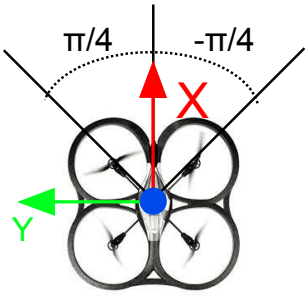


Figura 4.3: Giro del **VANT** sobre eje z para adquisición de imágenes

4.5 Sistema de visión

El sistema de visión tiene como entrada las imágenes adquiridas por una cámara monocular, la cual está montada en el **VANT**. Este sistema realiza,por un lado, la detección de marcadores artificiales, los cuales permiten estimar la posición de la cámara con respecto a la fachada, y por otro, la extracción de marcadores naturales, los cuales se utilizan para determinar las características tridimensionales de la edificación. Para el emparejamiento de imágenes se emplean puntos de interés, considerando detectores y descriptores que fueron desarrollados para operar en tiempo real. Las etapas que conforman el sistema de visión son: Detección de marcadores, procesamiento de imágenes y obtención de características tridimensionales.

4.5.1 Detección de marcadores artificiales

Los marcadores artificiales permiten estimar la localización del **VANT** con respecto a la fachada con respecto a una posición absoluta en el marco de coordenadas global Σ_W. Un marcador visual ideal es fácil de detectar bajo todas las circunstancias, siendo las diferencias de luminancia (brillo) más simples de identificar que las de crominancia (color). El Algoritmo 2 describe de manera general el proceso de detección de marcadores.

Algoritmo 2 Procedimiento general para el detector de marcadores

1: Entrenamiento de marcadores para crear base de marcadores
2: **para** Cada imagen a analizar **hacer**
3: Procesar la imagen para detectar marcador
4: **si** Se encuentra el marcador **entonces**
5: Identificar marcador para relacionarlo a posición absoluta en W
6: Estimar pose del **VANT**
7: **fin si**
8: **fin para**

4.5.2 Estimación de posición

Al considerar un marcador plano es posible estimar la matriz de homografía H entre una imagen de la base de datos de marcadores con la imagen de la escena. Mediante la matriz H se encuentra la posición y orientación del **VANT** respecto al sistema coordenado del marcador. Un marcador propuesto y la representación de transformación homográfica se representan en la Figura 4.4.

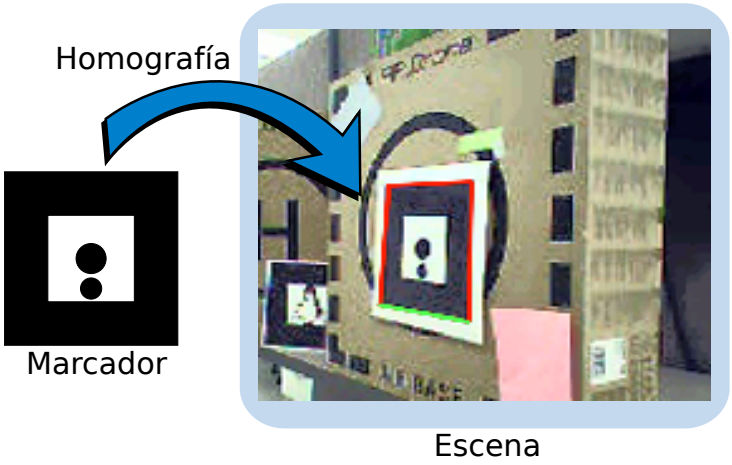


Figura 4.4: Representación de transformación homográfica del las imágenes del marcador a la escena

La estimación de los parámetros extrínsecos de la cámara (su postura en el espacio 3D) consiste en relacionar puntos tridimensionales $M = \left[X; Y; Z\right]'$ conocidos presentes en la escena con su correspondiente proyección en la imagen $m = (u, v)$. Los parámetros de posición $\left[R|t\right]$ son estimados

por medio de la matriz H; este problema se conoce como la *Perspectiva n-Puntos* (**PnP**) y es resuelto mediante el algoritmo *nDLT* (Transformación Lineal Directa Normalizada, *Normalized Direct Linear Transformation*).

Los componentes $\left[R|t\right]$ dependen de la matriz de calibración de la cámara K y mediante H es posible calcular la posición y orientación del **VANT**. Partiendo de la medida real del marcador y la imagen de la escena es posible determinar los valores $\left[R|t\right]$.

4.5.2.1. Entrenamiento de marcadores

Para lograr la detección de un marcador conocido en la escena, es deseable conocer de antemano las posibles orientaciones para un marcador. En un proceso único y previo a la detección de marcadores se genera una base de marcadores siguiendo el Algoritmo 3, el cual considera marcadores cuadrados.

Algoritmo 3 Procesamiento de imagen para la detección de marcadores

Entrada: Imágenes de los marcadores P_n
Salida: Base de conocimiento para marcadores esperados

```
1: para Cada marcador i hacer
2:    Validar que las dimensiones de P_i sean coherentes con una figura cuadrada
3:    para Rotar P_i 90 grados para cuatro orientaciones hacer
4:       Almacenar orientación y P_i rotado
5:    fin para
6: fin para
```

Para la detección del marcador y en particular para el caso de los vértices se requieren procesar las imágenes de la escena adquiridas por el **VANT**. El Algoritmo 4 describe el proceso para la detección de marcadores. Ésta técnica no es robusta a oclusiones, siendo una desventaja importante si se consideran ambientes con tránsito de personas. Para la propuesta de solución se considera que no presentan oclusiones e inicialmente algún marcador es visible para la cámara del **VANT**.

Inicialmente el **VANT** obtiene una fotografía de tamaño 320×240, previo al procesamiento de la imagen a ésta se le realiza una ecualización. Posteriormente se realiza una transformación para

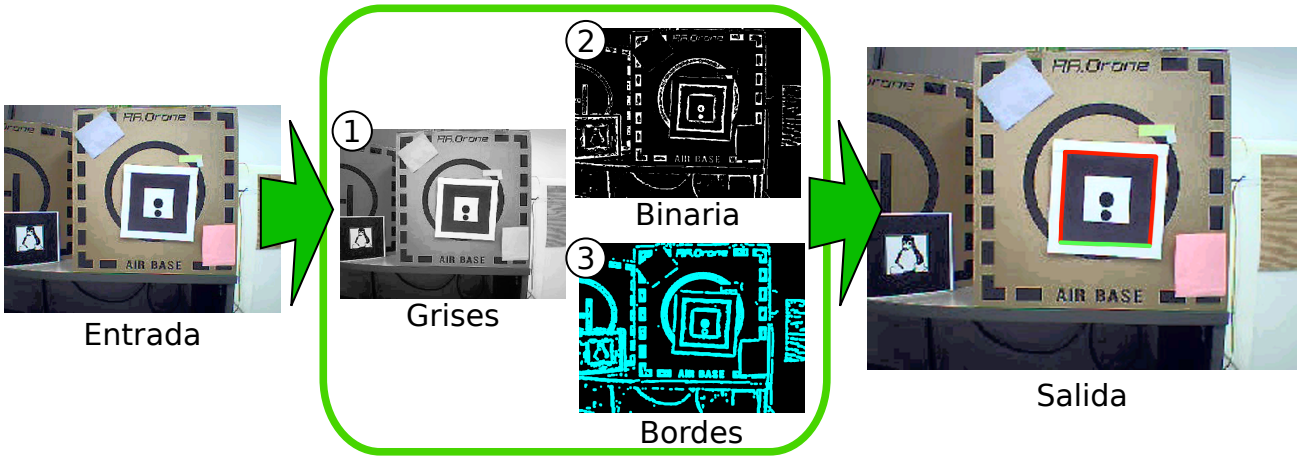


Figura 4.5: Imágenes derivadas del proceso de detección de marcadores

Algoritmo 4 Procesamiento de imagen para la detección de marcadores

Entrada: Imagen en escala de grises I, parámetros intrínsecos de cámara K, marcador P
Salida: Coordenadas en la imagen de vértices del marcador, posición de la cámara $\left[R|t\right]$

1: Binarizar mediante umbralado $I \rightarrow I'$
2: Detección de bordes $I' \rightarrow (C'_n, C' = \{x_i, y_i\}^m)$
3: Reducción de puntos $(C'_n, C' = \{x_i, y_i\}^m) \rightarrow (C''_n, C'' = \{x_i, y_i\}^4)$
4: Calculo de homografías para los candidatos $\{H\}^n$
5: Transformación proyectiva de candidatos $\{R, t\}^n$
6: Comparación P con $\{R, t\}^n$
7: Selección del mejor candidato
8: Determinar $\left[R|t\right]$ para el marcador

compensar la distorsión de la imagen producto de la lente de la cámara. Finalmente se amplía la imagen a 960×720 mediante interpolación bicúbica, ésta tiene un efecto de suavizado, reduce los artefactos de interpolación y permite preservar la separación entre bordes y esquinas. La interpolación bicúbica considera una ventana de 4×4 para estimar el valor del píxel conforme a la ecuación 4.1 considerando f_x, f_y y f_{xy}; los coeficientes a_{ij} se determinan de 16 ecuaciones de igual número de incógnitas descritas mediante los vecinos más cercanos del punto (x, y). Una forma de realizar la interpolación bicúbica mediante convolución fue presentada por Keys en el artículo [23].

$$v(x, y) = \sum_{i=0}^{3} \sum_{j=0}^{3} a_{ij} x^i y^j \tag{4.1}$$

La imagen ecualizada I de tamaño 960×720 es simplificada a un canal, es decir a escala de grises. Posteriormente se realiza un umbralado para obtener una imagen binaria, considerando una correlación cruzada con ventana Gaussiana de 45×45. La imagen binaria obtenida permite resaltar el marcador del resto de la escena para el procesamiento de detección de bordes. A partir de la imagen binaria se extraen los bordes mediante seguimiento, conforme lo presentado por Suzuki *et al.* en el artículo [58]. En la Figura 4.5 se muestran las imágenes derivadas en el proceso, resaltando que los bordes obtenidos son descritos mediante vectores de píxeles y que éstos se grafican con fines ilustrativos.

A continuación, se efectúa una simplificación al conjunto de bordes obtenidos aproximando cada vector de píxeles conectados a polígonos, mediante el algoritmo presentado por Douglas y Peucker en [13]. Dicho algoritmo realiza, dado una curva compuesta de segmentos de líneas, la simplificación de ésta mediante un menor número de puntos pertenecientes al conjunto original de puntos. De los polígonos obtenidos se consideran candidatos o posibles marcadores aquellos que describan cuadriláteros convexos, debido a que los marcadores empleados son cuadrados. Para cada uno de los candidatos que cumplen con las condiciones anteriores se calcula su postura dentro de la imagen a través de la transformación homográfica $H_{3\times3}$, la cual se estima con los vértices del candidato que corresponden a las esquinas del marcador propuesto.

Con la matriz H se realiza una transformación de perspectiva del candidato a un cuadrilátero d de dimensiones semejantes al marcador. Para determinar la correspondencia del candidato con algún marcador y orientación de la base de marcadores entrenados, se calcula el coeficiente de correlación de Pearson r para n muestras de (x, y) mediante la ecuación 4.2:

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n\sum x^2 - (\sum x)^2]\,[n\sum y^2 - (\sum y)^2]}} \tag{4.2}$$

Se consideran dos regiones para validar el marcador, la región exterior cuya función es diferenciar el marcador en la escena y la región central o interior que permite identificar al marcador así como determinar la orientación de éste. De la ecuación 4.2, se asignan las variables (x, y) para que corresponden a las regiones centrales del candidato y del marcador evaluado, respectivamente. Se calculan las normas euclidianas cuadradas l, así como el promedio v de valores de píxeles para x, y. La ecuación 4.2 queda de la forma 4.3, donde N indica el tamaño del marcador:

$$r = \frac{(x \cdot y) - (N v_x v_y)}{\sqrt{\left[l_x - N v_x^2\right]\left[l_y - N v_y^2\right]}} \tag{4.3}$$

La correlación del candidato para con cada marcador y para cada orientación determina su grado de semejanza o certidumbre que se trate de dicho marcador. De esta forma es posible identificar al candidato en la base de datos de marcadores y su orientación.

El método de detección de marcadores descrito permite identificar diferentes marcadores en una escena, siempre y cuando los marcadores sean visibles, es decir, no existan oclusiones y se encuentren a una distancia de la cámara que permita identificarlo con certidumbre suficiente. Para la navegación del **VANT**, el uso de marcadores permite ubicarlo en el entorno y corregir la estimación que se obtenga por odometría.

En cuanto al proceso de modelado tridimensional de la escena, la detección de marcadores en el primer par de imágenes permiten tener una escala métrica conocida al modelo obtenido.

4.5.3 Procesamiento de imágenes

Además de la detección de marcadores y la estimación de posición, el sistema de visión tiene como función recuperar característica tridimensionales de las fotografías. Posteriormente es necesario relacionar las imágenes, emparejarlas, para encontrar características o elementos comunes de la escena

que aparezcan en ambas, este emparejamiento se realiza por medio de detectores y descriptores de puntos característicos. Una vez relacionadas las imágenes y conociendo la posición de la cámara al momento que fueron adquiridas es posible estimar la posición tridimensional de dichas características.

4.5.3.1. Calibración de cámara

Como se mencionó en la sección 3.3.1, la calibración parte de la proyección de puntos de referencia conocidos, para lo cual se emplea un patrón plano. Los puntos detectados en el patrón de calibración corresponden a las esquinas de patrón de tablero de ajedrez, los cuales se muestran en la Figura 4.6 con los ejes (Xc, Yc) del sistema de coordenadas de la cámara.

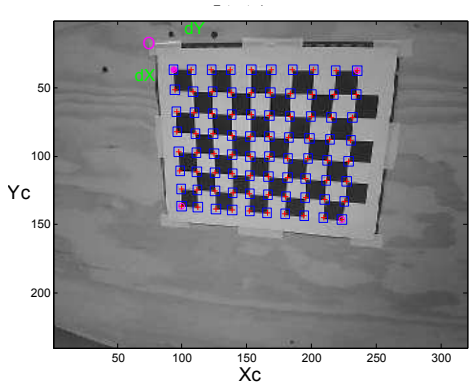


Figura 4.6: Esquinas detectadas en patrón para una imagen 320×240

Al conocer las dimensiones de la cuadrícula del patrón de ajedrez, para cada imagen o fotografía capturada se calcula su homografía. Al conjunto de imágenes donde aparece el patrón de calibración y sus homografías, se aplica el algoritmo presentado por Zhang *et al.* en [70]. Idealmente un mayor número de imágenes para la calibración permite una mejor estimación de parámetros intrínsecos y coeficientes de distorsión, para lo cual el procesamiento de imágenes se realiza de forma automática. A manera de ejemplo, un subconjunto de imágenes empleado para la calibración se muestra en la Figura 4.7.

Figura 4.7: Subconjunto de imágenes obtenidas por el **VANT** para realizar la calibración de cámara mediante el algoritmo de Zhang *et al.* en [70]

El Algoritmo 5 describe el proceso para la calibración de una cámara considerando un conjunto de imágenes I_n previamente obtenidas, donde aparece el patrón de calibración en diferentes perspectivas y posiciones.

Algoritmo 5 Procedimiento general para el detector de marcadores

Entrada: Imágenes del patrón de calibración I_n, medidas reales métricas del patrón y tamaño del tablero.

Salida: Matriz de parámetros intrínsecos K y coeficientes de distorsión d

1: **para** Cada n imagen en I_n **hacer**
2: Detectar esquinas en patrón de ajedrez
3: Realizar un estimado en subpíxeles de esquinas m_n
4: Estimar la pose de la cámara para M y m mediante *nDLT*
5: Minimizar el error de reproyección
6: **fin para**

4.5.3.2. Emparejamiento de imágenes

Se considera que la cámara del **VANT** adquiere una imagen para un instante dado y realiza un desplazamiento o cambio de pose antes de adquirir la siguiente imagen a dicho instante. Mediante la detección y emparejamiento de la proyección de características de la escena en ambas imágenes

es posible determinar la relación entre éstas. Finalmente, empleando dicha información para estimar el cambio de pose de la cámara así como recuperar la posición en el espacio de éstas características y generar un modelo tridimensional de la escena.

4.5.3.3. Detección y descripción de puntos característicos

Los puntos de interés $(x, y, d^{\rightarrow})$ se construyen a partir de regiones de la imagen que presentan cambios de contraste, las cuales son caracterizadas como una diferencia local de intensidades $d^{\rightarrow}$ y una ubicación precisa en la imagen (x, y).

Ya que el procesamiento de características se debe realizar mientras que el **VANT** está en vuelo, se consideran los algoritmos **FAST** y **BRISK** para la detección de puntos, y los descriptores binarios **BRIEF** y **BRISK**. En el artículo [2] presentado por Barazzetti *et al.*, en el cual se basa esta propuesta de solución para la generación de un modelo tridimensional partiendo de fotografías, los autores sugieren emplear el detector de puntos **FAST**. Así mismo, Schmidt *et al.* en el artículo [52] determinan que en la navegación de robots es recomendable emplear el detector **FAST** junto con el descriptor de puntos **BRIEF**.

La estación en tierra, mostrada en la Figura 4.1, al recibir una nueva imagen del **VANT** la procesa como se indica en el Algoritmo 6. Se asume que la imagen adquirida ya fue corregida con los valores obtenidos por la etapa de calibración de la cámara, para compensar la distorsión. Para el manejo de los puntos característicos, se propone una estructura de datos *m_Imagen* compuesta por la matriz de imagen I_n, el índice n, el vector de puntos característicos, la matriz de los descriptores correspondientes y la matriz de posición del **VANT** determinada por odometría.

De tal forma que existe una estructura *m_Imagen* de índice n para cada imagen adquirida, permitiendo que en las siguiente etapa del sistema de visión se cuente con la información necesaria para la construcción del modelo tridimensional.

Algoritmo 6 Procedimiento general para el detector de marcadores

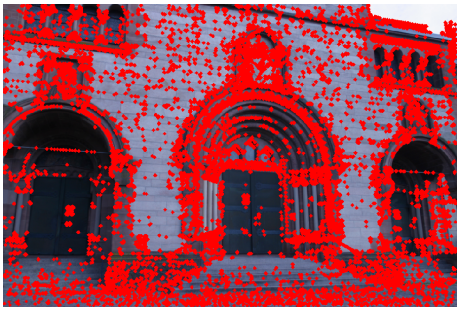
Entrada: Imagen I_n adquirida por el **VANT**
Salida: *m_Imagen*

1: **para** Imagen I_n **hacer**
2: Detección de puntos por **FAST**
3: Descripción de puntos característicos
4: Matriz de posición del **VANT** P_n
5: **fin para**

En la Figura 4.8a se aprecia una imagen del set de datos *Herz-Jesu-P8* proporcionado por Strecha *et al.* en el artículo [57]. En ésta se aprecian los puntos de interés detectados por medio de **FAST** en la Subfigura 4.8b y **BRISK** en la Subfigura 4.8c.

(a) Imagen original del set de datos

(b) 1572 puntos obtenidos por **BRISK** con umbral 10, 3 octavas



(c) 25708 puntos obtenidos por **FAST** con umbral 10, sin supresión de no máximos

Figura 4.8: Ejemplo de puntos característicos para imagen del set de datos *Herz-Jesu-P8*[57]

4.5.3.4. Problema de emparejamiento

Considerando un par estructuras *m_Imagen* de índice n y $n+1$ compuestas cada una por las imágenes (I_n, I_{n+1}) con vectores puntos característicos previamente localizados (p_n, p_{n+1}) y la respectiva matriz de descriptores para dichos puntos (d_n, d_{n+1}). El problema del emparejamiento consiste en determinar cuáles puntos de característicos de la imagen I_n corresponden efectivamente a aquellos de la imagen I_{n+1}. La relación entre estos puntos se determina mediante los descriptores (d_n, d_{n+1}), calculando la diferencia o distancia entre éstos. En el caso de descriptores binarios, utilizados en esta propuesta, se emplean distancias Hamming para comparar a los descriptores, a diferencia de los descriptores no binarios en los cuales se prefieren distancias Euclidianas.

El enfoque exhaustivo para determinar las correspondencias con descriptores binarios y distancia Hamming se muestra en el Algoritmo 7. Emparejar descriptores binarios consiste en el cálculo de la distancia Hamming, el número total de bits diferentes en ambos descriptores es la medida de disimilitud. Esta operación es simplemente un **XOR** seguido de un conteo de bits.

Algoritmo 7 Emparejamiento de descriptores binarios mediante distancia Hamming

Entrada: Descriptores (d_n, d_{n+1}) correspondientes a los puntos (p_n, p_{n+1}) encontrados en las imágenes (I_n, I_{n+1})
Salida: Conjunto de correspondencias C entre (d_n, d_{n+1})

```
1: para Elemento i de todos los elementos de d_n hacer
2:    para Elemento j de todos los elementos de d_{n+1} hacer
3:       Comparar (d(n)_i, d(n+1)_j) mediante operación XOR
4:       Disimilitud: Conteo de bits
5:    fin para
6:    C_{i,j} ← menor disimilitud
7: fin para
```

Para eliminar falsas correspondencias encontradas por el Algoritmo 7 se realiza una prueba de simetría. Ésta prueba se presenta en el Algoritmo 8, y consiste en relacionar I_n con I_{n+1}, posteriormente I_{n+1} con I_n. y finalmente eliminar aquellas correspondencias que no aparecen en

ambas relaciones.

Algoritmo 8 Prueba de simetría para emparejamiento de descriptores

Entrada: Descriptores (d_n, d_{n+1})
Salida: Correspondencias C

```
A ← Emparejar (d_n, d_{n+1}) mediante el Algoritmo 7
B ← Emparejar (d_{n+1}, d_n)
para Correspondencia i de A hacer
    si B_i == A_i entonces
        Agregar A_i a C_i
    si no
        Descartar A_i
    fin si
fin para
```

El emparejamiento por similitud proporcional determina por lo menos un par de correspondencias en d_{n+1} para cada elemento de d_n. Esta prueba es una búsqueda por el algoritmo de vecinos más cercanos empleando *hashing* sensible a localidad con múltiple sondeo, presentado por Lv *et al.* en el artículo [30]. Previo al emparejamiento se realiza el mapeo de los elementos de d_n en tablas *hash*. La búsqueda de correspondencias para un elemento d_{n+1} es similar al mapeo anterior considerando un factor de perturbación, mediante el cual se determina que los vecinos semejantes a dicho elemento pertenecen a un rango de tablas simplificando la búsqueda de elementos semejantes.

Con por lo menos dos elementos de d_{n+1} relacionados a d_n se descartan aquellas correspondencias en las cuales la distancia del elemento más próximo debe ser mayor que la distancia del segundo elemento multiplicado por una constante t. Barazzetti *et al.* en su artículo [2] sugieren un valor de t entre 0.5 y 0.8. El emparejamiento por similitud proporcional se describe en el Algoritmo 9:

Los métodos previamente descritos son utilizados para encontrar correspondencias y eliminar aquellas que no sean consistentes. Conociendo las correspondencias para un par de imágenes se puede obtener un modelo matemático el cual describa el cambio de pose de la cámara. Para la obtención

Algoritmo 9 Emparejamiento de descriptores binarios mediante vecinos más cercanos

Entrada: Descriptores (d_n, d_{n+1}), relación t
Salida: Correspondencias C

para Todos los elemento i de d_n **hacer**
  $a, b \leftarrow$ Dos correspondencias de $d_{n_{i+1}}$ para d_{n_i}
  $(a_d, b_d) \leftarrow$ Similitud de correspondencias
  si $a_d < t * b_d$ **entonces**
    $C_i \leftarrow$ Correspondencia $d_{n_{i+1}}$ con $a \in d_{n_i}$
  si no
    Descartar correspondencias para d_{n_i}
  fin si
fin para

de este modelo se prefiere emplear el algoritmo **RANSAC** [15], el cual remueve las correspondencias no consistentes, es decir, que no satisfagan a dicho modelo.

4.5.4 Reconstrucción 3D

Para la recuperación de características tridimensionales de la escena, primero se debe establecer la relación de escala que tendrá el modelo tridimensional, conocida como *baseline*. Ésta se establece a partir del primer par de imágenes que se procesan y los puntos tridimensionales que resulten. Una vez que se han recuperado las características tridimensionales para el par de imágenes inicial añadir más puntos al modelo no es trivial, debido a que la reconstrucción es a escala y cada par de imágenes se encuentra a diferente escala. Para añadir más puntos al modelo una solución se necesita estimar la posición de la cámara con respecto a los puntos que ya han sido obtenidos, conocido como el problema de los n puntos (**PnP**). Otra solución consiste en generar una segunda nube de puntos y evaluar como se acopla con la previamente obtenida, con lo cual se obtiene la posición de la nueva cámara, conocido como *Iterative Closest Point* (**ICP**). En esta propuesta de solución, se realiza la reconstrucción incremental del modelo por medio del método **PnP**, ya que se desea que el modelo de puntos tridimensionales se actualice en línea. A continuación se indican los algoritmos necesarios para la recuperación de características tridimensionales y la pose de la cámara y posteriormente la

ampliación de éste con **PnP**. Lo anterior se expresa mediante el Algoritmo 10.

Algoritmo 10 Construcción del modelo tridimensional por medio de un conjunto ordenado de fotografías

Entrada: Conjunto de fotografías ordenadas de la fachada de una edificación
Salida: Modelo tridimensional de la fachada en forma de nube de puntos

```
1: para Todas las imágenes i de la fachada hacer
2:    si Primer par de imágenes n y n + 1 del conjunto entonces
3:       Emparejar ambas imágenes
4:       Baseline ← Generar primera nube de puntos mediante triangulación
5:    si no
6:       Emparejar ambas imágenes n + (i − 1), n + i
7:       Relacionar correspondencias C_i de la imagen n + i con puntos de nube 3D previamente
         obtenidos de la imagen n + (i − 1)
8:       Estimar posición de la cámara resolviendo PnP para C_i
9:       Generar mediante triangulación nube de puntos coherentes con baseline
10:   fin si
11: fin para
```

Para esta etapa, se asume que ambas imágenes fueron adquiridas con la misma cámara, que los conjuntos de puntos para realizar la búsqueda de correspondencias tienen igual número de elementos y se encuentran expresados en coordenadas homogéneas.

4.5.5 Triangulación

Por medio de la triangulación es posible recuperar las coordenadas tridimensionales de un punto de la escena si es visible en un par de imágenes las cuales se conozca la posición de la cámara que las adquirió. Este proceso fue descrito ampliamente por Hartley y Sturm en el artículo *"Triangulation"* [19]. Para la presente propuesta se emplea el método de mínimos cuadrados iterativos, el cual recomiendan los autores por su velocidad contra métodos como el polinomial y el de valores propios iterativos, evaluados en dicho artículo.

El algoritmo para la triangulación lineal por mínimos cuadrados iterativa se presenta en el Algoritmo 11.

Algoritmo 11 Triangulación por mínimos cuadrados

Entrada: Puntos $\mathbf{m}$ y $\mathbf{m\prime}$ en las imágenes y poses de cámara $(P, P\prime)$ respectivas
Salida: Punto tridimensional de la escena $\mathbf{M}_{3\times1}$

1: $A_{4\times3} \leftarrow$ De la ecuación 3.29
2: $B_{4\times1} \leftarrow$ Posición en el espacio, de ecuación 3.29
3: $X \leftarrow$ Resolver $A\mathbf{M} = B$ por mínimos cuadrados

Sin embargo, esta forma de obtener un punto en el espacio considera que las coordenadas para $(\mathbf{m}, \mathbf{m\prime})$ están libres de ruido y que las poses $(P, P\prime)$ no contienen errores. Debido a que estas circunstancias son imposibles de obtener en una aplicación real y que la proyección de los puntos en el espacio difícilmente van a coincidir con su correspondencia, es que Hartley y Sturm proponen un método iterativo [19]. Este método propone pesos $(w, w\prime)$ los cuales se ajustan de forma iterativa, ponderando la ecuación 3.30, minimizando el error de reproyección del punto en el espacio obtenido. En el Algoritmo 12 se indica como se realiza el ajuste de los pesos para el cálculo de $\mathbf{M}$. El número de iteraciones recomendado por los autores es 10.

Algoritmo 12 Triangulación iterativa

Entrada: Puntos $\mathbf{m}$ y $\mathbf{m\prime}$, poses $(P, P\prime)$, ε error reproyección
Salida: Punto $\mathbf{M}_{3\times1}$

1: $w \leftarrow$ Peso para ajustar $\mathbf{m}$
2: $w\prime \leftarrow$ Peso para ajustar $\mathbf{m\prime}$
3: **mientras** $Iteraciones < 10$ **hacer**
4: $\hat{\mathbf{M}} \leftarrow$ Punto estimado para $\mathbf{M}$
5: $\hat{\mathbf{M}} \leftarrow$ Triangular $(\mathbf{m}, \mathbf{m\prime})$ con Algoritmo 11
6: $\mathbf{M} \leftarrow \hat{\mathbf{M}}$
7: $\hat{\mathbf{m}} \leftarrow$ Última fila de $P \times \mathbf{M}$
8: $\hat{\mathbf{m}}\prime \leftarrow$ Última fila de $P\prime \times \mathbf{M}$
9: **si** $w - \hat{\mathbf{m}} \leq \varepsilon$ **and** $w\prime - \hat{\mathbf{m}}\prime \leq \varepsilon$ **entonces**
10: Salir de iteraciones, $\mathbf{M}$ ideal
11: **fin si**
12: $w \leftarrow \hat{\mathbf{m}}$
13: $w\prime \leftarrow \hat{\mathbf{m}}\prime$
14: $A_{4\times3} \leftarrow$ De la ecuación 3.29
15: $B_{4\times1} \leftarrow$ Posición en el espacio, de ecuación 3.29
16: $\mathbf{M} \leftarrow$ Resolver $A\mathbf{M} = B$ por mínimos cuadrados
17: Aumentar $Iteraciones$
18: **fin mientras**

La triangulación descrita en el Algoritmo 11, así como la aproximación iterativa explicada en el Algoritmo 12, son aplicables para un único par de puntos $(\mathbf{m}, \mathbf{m\prime})$. Para todo el conjunto de puntos, se emplea el Algoritmo 14, que considera todas las correspondencias entre un par de imágenes además de calcular el error de reproyección promedio para todo el conjunto.

Algoritmo 13 Triangulación para todos los puntos entre dos imágenes

Entrada: Conjunto de puntos $\mathbf{m}_n$ y $\mathbf{m\prime}_n$, poses $(P, P\prime)$, parámetros intrínsecos K de la cámara y la inversa K^{-1}
Salida: Conjuntos de puntos $\mathbf{M}_{3\times1}$, error de reproyección e

1: $P^{-1} \leftarrow$ Calcular matriz inversa de P
2: **para** Para todos los elementos i de $\mathbf{m}_n$ **hacer**
3: $K^{-1} * \mathbf{m}_i \leftarrow$ Cambio de referencial de imagen a escena
4: $K^{-1} * \mathbf{m\prime}_i$
5: $\mathbf{M}_i \leftarrow$ Triangulación iterativa (Algoritmo 12)
6: $n\mathbf{m}_i = K * \mathbf{M}_i \leftarrow$ Cambio de referencial de imagen a escena
7: Dividir $n\mathbf{m}_i$ entre componente $n\mathbf{m}_i.z$ para hacer punto 2D
8: $e_i \leftarrow$ Error de reproyección $n\mathbf{m}_i - \mathbf{m}_i$
9: Almacenar $\mathbf{M}_i$ en nube tridimensional
10: **fin para**
11: Calcular error promedio de reproyección e

4.5.6 Construcción del modelo tridimensional con el VANT

Considerando el Algoritmo 10 para la generación del modelo tridimensional, la presente propuesta busca realizar dicha construcción del modelo a la par que el **VANT** se encuentra en vuelo. Del primer par de imágenes (I_n, I_{n+1}) obtenidas y procesadas se obtiene la base del modelo tridimensional. La siguiente imagen I_{n+2} se empareja con la inmediata anterior, I_{n+1} y así sucesivamente. La posición y postura iniciales del **VANT** se estiman por medio del marcador visual.

Previamente en la Subsección 4.5.2 se planteó como el problema de **PnP** es resuelto mediante el algoritmo *nDLT*, y permite determinar la transformación espacial de un conjunto de puntos 2D a un espacio 3D. Dado que se desea añadir más puntos a un modelo de la escena ya existente con escala

fija, se requiere calcular la pose de la cámara mediante el algoritmo *nDLT*. Este algoritmo requiere un conjunto de puntos tridimensionales y sus respectivos puntos 2D para calcular la proyección. Se conocen el conjunto de puntos $\mathbf{m}_{n+1}$ en la imagen I_{n+1} así como las proyecciones en el espacio $\mathbf{M}_{n+1}$ obtenidas mediante triangulación. Al procesar la imagen I_{n+2}, se realiza el emparejamiento de sus puntos característicos $\mathbf{m}_{n+2}$ sólo con aquellos puntos de $\mathbf{m}_{n+1}$ para los que existe un valor de $\mathbf{M}_{n+1}$. Una vez obtenido este subconjunto de $\mathbf{m}_{n+2}$ y sus correspondencias de $\mathbf{M}_{n+1}$, es sobre estos dos subconjuntos que se emplea el algoritmo *nDLT*.

Una vez determinada la transformación T para I_{n+2}, se realiza el emparejamiento con todos los puntos de I_{n+1} aunque no tengan una correspondencia de $\mathbf{M}_{n+1}$. La pose de I_{n+1} es conocida, lo que permite determinar la de I_{n+2} mediante *nDLT*, con el Algoritmo 14. La pose de la cámara es conocida para el primer par de imágenes gracias a la detección del marcador, para las siguientes imágenes se determina la posición de la cámara mediante el algoritmo *nDLT* y la odometría del **VANT**.

4.6 Conclusiones

Se presentaron los distintos algoritmos que conforman las soluciones a cada uno de los elementos de los que se compone el problema a resolver. El diagrama de la Figura 4.1 muestra cada uno de los módulos descritos y ordenados en el algoritmo 14. Respecto al **VANT** se requiere estimar su posición en todo momento, para lo cual se emplean los sensores inerciales y en caso de estar presente, el marcador visual para obtener una posición absoluta. La detección de marcadores permite ubicar de manera precisa la posición del **VANT** con respecto a la edificación, corregir la navegación así como dar una escala real al modelo tridimensional obtenido.

Algoritmo 14 Generación de un modelo tridimensional en línea por medio de imágenes transmitidas por un **VANT**

Entrada: Parámetros intrínsecos K de la cámara, imágenes transmitidas por el **VANT** I
Salida: Nube de puntos $\mathbf{M}_{3\times1}$

1: $(I_0, I_1) \leftarrow$ Primer par de imágenes
2: $(P_0, P_1) \leftarrow$ Detectar marcador y calcular pose (Algoritmo 2)
3: $(\mathbf{m}_0, \mathbf{m}_1) \leftarrow$ Emparejamiento (Algoritmo 7)
4: $\mathbf{M}_0 \leftarrow$ Triangulación para obtener primera nube de puntos (Algoritmo 14)
5: $c\mathbf{m}_1, X0) \leftarrow$ Correspondencia del subconjunto de puntos para imagen I_1 con nube de puntos
6: **mientras VANT** transmite imagen n **hacer**
7: $\quad\mathbf{m}_n \leftarrow$ Detectar puntos característicos de I_n
8: $\quad$Emparejamiento $(c\mathbf{m}_{n-1}, \mathbf{m}_n)$
9: $\quad P_n \leftarrow$ Calcular pose con *nDLT*
10: $\quad$Emparejamiento $(\mathbf{m}_{n-1}, \mathbf{m}_n)$
11: $\quad\mathbf{M}_n \leftarrow$ Triangulación iterativa de $(\mathbf{m}_{n-1}, \mathbf{m}_n)$ con (P_{n-1}, P_n)
12: $\quad(c\mathbf{m}_n, \mathbf{M}_n) \leftarrow$ Correspondencia del subconjunto de puntos $\mathbf{m}_n$ con $\mathbf{M}_n$
13: $\quad n \leftarrow$ Aumentar índice de imagen
14: **fin mientras**

Los algoritmos de visión necesarios para recuperar las características tridimensionales parten de la extracción y descripción de puntos de interés, así como del emparejamiento de éstos en la secuencia de imágenes. En éste último se requiere que las correspondencias no tengan errores para calcular el modelo que permita la triangulación de estas características.

El modelo tridimensional consta de una nube de puntos conformada por las características obtenidas de las imágenes y reproyectadas al espacio tridimensional. Estos puntos cuentan con una posición (x, y, z), así como un valor de color.

Es posible realizar un análisis más profundo de las imágenes con el objetivo de obtener una nube de puntos densa del edificio que se está reconstruyendo. Sin embargo, este procesamiento posterior de las imágenes no es considerado para el presente trabajo, ya que el objetivo principal de nuestro desarrollo se enfoca en obtener un modelo en línea, para determinar zonas que no han sido suficientemente capturadas.

5

Validación experimental

En el presente capítulo se describen los resultados obtenidos para cada una de las etapas que componen la propuesta de solución para la construcción de un modelo tridimensional de la fachada de un edificio, a partir de fotografías monoculares obtenidas por un vehículo aéreo no tripulado **VANT**. El objetivo principal de esta etapa es la validación del funcionamiento de cada uno de los componentes de la solución final, que al ser integrados permiten evaluar la propuesta de solución completa.

Para cada uno de los componentes que integran la solución propuesta se ha obtenido el mejor compromiso entre los distintos parámetros que afectan su desempeño, para que, al ser integrados, resulten en una solución general apropiada al problema a resolver. Así, para la odometría construida únicamente con información de sensores inerciales, se busca reducir la incertidumbre en la medida de lo posible, ya que los errores sistemáticos son propios de cada **VANT** y no es posible obtener un modelo válido para todas las plataformas. Con la detección de marcadores se corrige la estimación

de posición de la plataforma aérea respecto a la fachada, para lo cual se evaluó la distancia y la precisión con la que se determina la pose de la cámara respecto al marcador.

Durante la calibración de los algoritmos de visión, se evaluaron los diferentes componentes teniendo siempre como objetivo final la recuperación del modelo tridimensional. Para la detección de puntos característicos se evaluaron los tiempos y cantidad de puntos adquiridos, de lo cual depende la densidad del modelo construido. Se consideraron los algoritmos **FAST** y **BRISK**, así como la descripción por valores binarios por los algoritmos **BRIEF** y **BRISK**, para procesar las imágenes en el menor tiempo posible, y lograr tiempos compatibles con la velocidad de adquisición de las imágenes. Para los algoritmos de emparejamiento de imágenes, se buscó obtener la mayor cantidad de coincidencias correctas, ya que de esto depende el cálculo de la posición del **VANT** para cada imagen. Se eligió la mediana como valor representativo durante la experimentación debido a que la distribución no es simétrica, y de esta forma dicho valor permite generalizar sobre los resultados. Esto se observó sobre todo en la distribución de puntos característicos detectados, existiendo un mayor número en las últimas fotografías del conjunto de imágenes.

Finalmente se presentan los resultados obtenidos por la solución completa, el tiempo necesario para el procesamiento de las imágenes y la nube de puntos obtenida.

5.1 Plataforma de prueba

Las pruebas se realizaron con el **VANT** Parrot AR. Drone 1, que satisface los requisitos de estabilización autónoma, cámara frontal y comunicación inalámbrica con una estación en tierra. Para el desarrollo de la propuesta de solución es muy importante que este **VANT** ya cuente con un control de estabilidad y de desplazamiento sobre sus ejes (x, y), así como el giro sobre el eje z. La

comunicación con la aeronave incluye la lectura de sus sensores inerciales así como recibir imágenes en el momento que se soliciten. Las características técnicas de esta plataforma se muestran en la Tabla 5.1.

Procesador ARM9 RISC de 32bits@468MHz
Módulo WiFi b/g integrado
Cámara frontal de 320×240
Autonomía de vuelo de 10 a 15 minutos
Velocidad de vuelo 5 m/s
Altímetro por ultrasonido @ 40KHz
IMU: acelerómetro y tres giroscopios

Tabla 5.1: Parrot AR.Drone v1.8

La computadora empleada como estación en tierra para el procesamiento de las imágenes y reconstrucción del modelo es una computadora de escritorio con procesador Intel® Core™ i5-$2400S$@2.5GHz, 4GB de memoria DDR3@1333MHz con tarjeta gráfica integrada AMD Radeon™HD 6750M, bajo el sistema operativo Ubuntu 13.04 de 64bits. Se utilizaron las bibliotecas *OpenCV* para los algoritmos de visión y *OpenGL* para la visualización de la nube de puntos.

5.1.1 Comunicación y control del VANT

El **VANT** seleccionado crea una red WiFi *ad-hoc* con tres puertos **UDP**, por los cuales transmite la información de los sensores (puerto 5554), las imágenes (puerto 5555) y recibe instrucciones de control de movimiento (puerto 5556). Las instrucciones de control de movimiento se limitan a desplazamiento sobre los ejes de su marco de referencia así como giro del ángulo ψ en la guiñada (sobre el eje z). El sistema local de coordenadas ubicado sobre el **VANT** se mostró en la Figura 3.2. Las instrucciones de vuelo están expresadas en términos de velocidad, y para estimar su posición en un marco de referencia global se emplea la información de los sensores y la odometría.

5.2 Calibración de la cámara

Los parámetros intrínsecos, expresados dentro de la matriz $K_{3\times3}$, de la cámara frontal del **VANT** se estimaron con la herramienta *Camera Calibration Toolbox for Matlab*[7], que es una implementación del Algoritmo presentado por Zhang *et al.* en [70]. Esta herramienta emplea una malla rectangular con textura de patrón de tablero de ajedrez, el cual se mostró en la Figura 3.9. El patrón consiste en una rejilla de (10×7) cuadros, de $25mm$ de lado. Se tomaron 30 fotografías de una resolución de 640×480 pixeles, desde distintas posiciones y orientaciones del **VANT**. Los valores estimados de la matriz K y las distorsiones $dist$, obtenidos por la herramienta [7], se muestran en la Ecuación 5.1.

$$K = \begin{bmatrix} 417.60092 & 0 & 323.73329 \\ 0 & 417.37459 & 226.62748 \\ 0 & 0 & 1 \end{bmatrix} \tag{5.1}$$

$$dist = \begin{bmatrix} 118.62 \times 10^{-3} & -209.26 \times 10^{-3} & 0.31 \times 10^{-3} & -1.84 \times 10^{-3} & 0.0 \end{bmatrix}$$

Como ejemplo, en la Figura 5.1 se muestran un par de imágenes tomadas por la cámara del **VANT** y el resultado de la corrección de las distorsiones $dist$ estimadas. Las subfiguras 5.1a y 5.1c muestran las imágenes originales, mientras que en las subfiguras 5.1b y 5.1d se muestran las imágenes rectificadas correspondientes. Las distorsiones que presentan las imágenes originales son de tipo radial, en el cual los pixeles se alejan, conocida como de tipo *barril*, o se acercan, tipo *alfiletero*, con respecto al centro de la imagen p_c. La distorsión es evidente en las esquinas de las imágenes, donde las líneas rectas aparecen curvas.

(a) Original

(b) Rectificado

(c) Original

(d) Rectificado

Figura 5.1: Rectificación de fotografías obtenidas con la cámara frontal del **VANT** de prueba

El tamaño de las imágenes utilizadas para la calibración (640×480 pixeles) determina la matriz K de parámetros intrínsecos de la cámara para esa escala. Al cambiar de escala las imágenes, ampliarlas o reducirlas, es necesario escalar los valores de distancia focal f y centro óptico $p_c = (o_x, o_y)$ de K en la misma proporción. Los valores para los factores de distorsión son independientes de la escala de la imagen, por lo que permanecen constantes. Para las siguientes etapas de la experimentación, podemos asumir que todas las imágenes obtenidas por el **VANT** están rectificadas, con una matriz K debidamente escalada para el tamaño de las imágenes de video (960×720 pixeles) que se reciben del **VANT**.

5.3 Detección del marcador

Las características de mayor relevancia en el módulo de detección de marcadores artificiales de la solución propuesta son la distancia máxima a la cual el marcador es detectado correctamente por el sistema, así como la exactitud de la estimación de la posición del **VANT** con respecto a éste. Por lo tanto, al evaluar la detección de marcadores, se determinan los errores en el cálculo de la posición entre el **VANT** y el marcador, así como la exactitud y repetibilidad de las detecciones.

El experimento consistió en ubicar manualmente al **VANT** frente al marcador a distancias conocidas, procesar las fotografías y, mediante información de la imagen, estimar la distancia respecto al marcador. De esta forma se determinada la distancia frontal máxima d_f en la cual el error entre la distancia estimada y la real se vuelve considerable. Posteriormente se ubica al **VANT** en un ángulo θ respecto a la perpendicular del marcador artificial, a una distancia d_f, para determinar el ángulo θ máximo en el cual se deja de detectar el marcador, como se muestra en la Figura 5.2. El eje principal de la cámara se alineó al centro del marcador, para que éste aparezca en el centro de la imagen, con el fin de reducir los posibles errores generados por la propia lente.

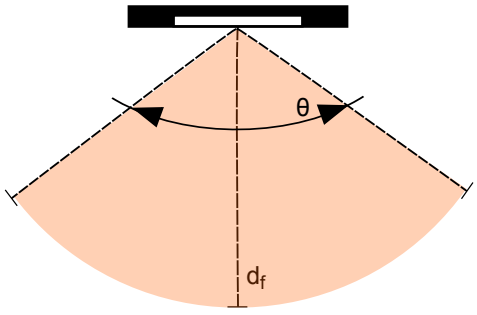


Figura 5.2: Determinando el rango de detección de marcador

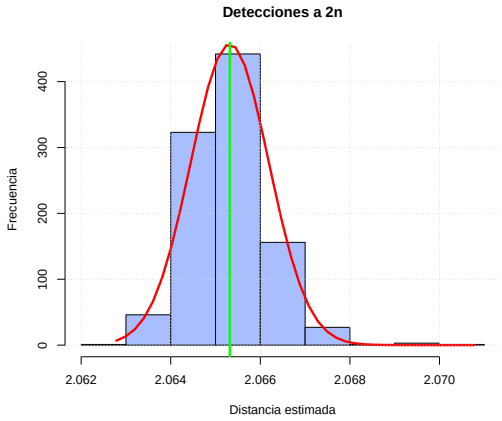
Las imágenes obtenidas por el **VANT** AR.Drone con la cámara frontal tienen una resolución *VGA* de (320×240) y se amplían a (960×720) mediante interpolación bicúbica. El marcador es un

cuadrado de dimensiones de $n = 20$ cm por lado. La detección del marcador depende del tamaño que éste tiene dentro de la imagen, por lo que se ve afectado tanto por la dimensión física del marcador como por la resolución de la cámara. Dado que la cámara tiene una resolución fija, el único parámetro variable es la dimensión del marcador, por lo que se decidió utilizar el valor n como unidad de referencia.

El **VANT** se colocó a una distancia inicial de $2n = 40$ cm con incrementos de $2n$ entre cada medición. Se realizaron 1000 fotografías para cada distancia en las cuales se busca al marcador, de esta forma se determina el porcentaje de aciertos y la repetibilidad de las mediciones de distancia.

En todos los casos, la detección del marcador en la imagen es satisfactoria; sin embargo, existe un error entre la distancia estimada y la real, que el conjunto de experimentos siguientes permitió establecer como $\pm 0.25n$ a una distancia de $6n$ del marcador (± 5 cm a 1.20 m en este caso) como rango de estimación. A continuación se muestran los resultados obtenidos.

(a)

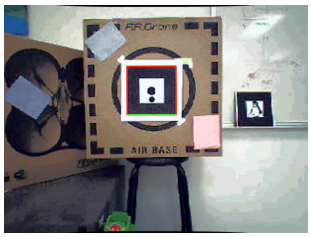


(b)

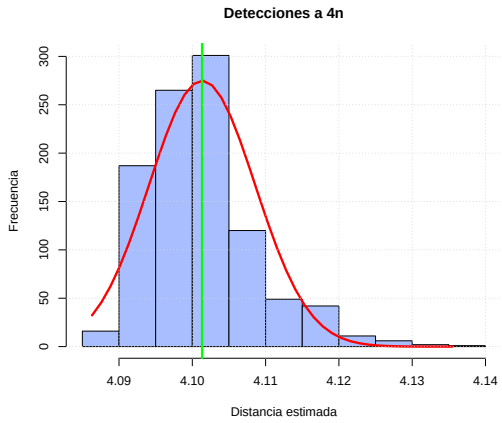
Figura 5.3: Detecciones a $2n$, media estimada de $2.07n$

A una distancia de $2n$ del marcador se obtuvo una detección del $100\,\%$, con una confianza media

de 0.93. Como se muestra en la de la Figura 5.3b, la relación distancia/marcador media estimada fue de $2.07n$, con una desviación estándar de $8.72 \times 10^{-4}n$.



(a)



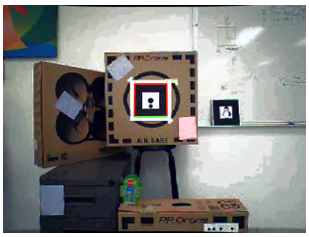
(b)

Figura 5.4: Detecciones a $4n$, media estimada de $4.10n$

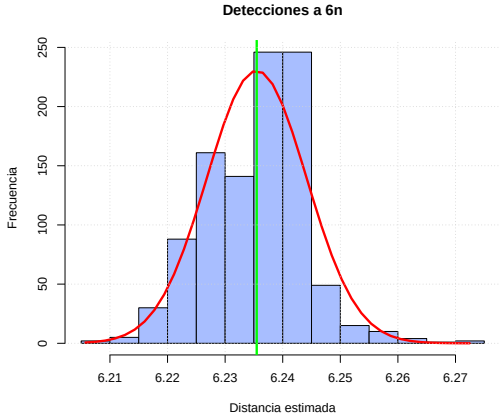
A una distancia $4n$, el marcador fue detectado en el $100\,\%$ de las imágenes, con una confianza media de 0.90. En la gráfica de la Figura 5.4b se muestra que la relación distancia/marcador media estimada fue de $4.10n$. con desviación estándar de $72.55 \times 10^{-4}n$.

Ubicando al **VANT** a una distancia de $6n$, el marcador se detectó en el $100\,\%$ de las fotografías, con una confianza media de 0.88. En la Figura 5.5b, la relación distancia/marcador media estimada fue de $6.23n$, con una desviación estándar de $86.61 \times 10^{-4}n$.

A una distancia $8n$ del marcador se obtuvo una detección del $100\,\%$, con una confianza media de 0.85. En la gráfica 5.6b de la Figura 5.6 se muestra que la relación distancia/marcador media estimada fue de $8.48n$. con desviación estándar de $15.76 \times 10^{-3}n$.

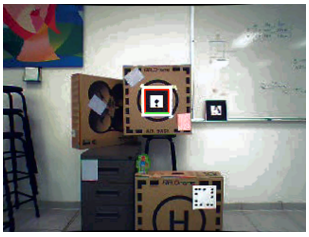


(a)

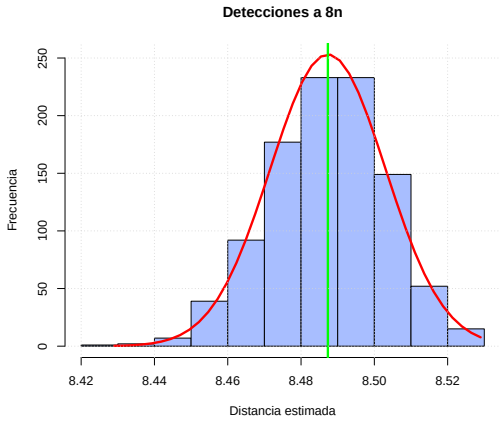


(b)

Figura 5.5: Detecciones a $6n$, media estimada fue de $6.23n$

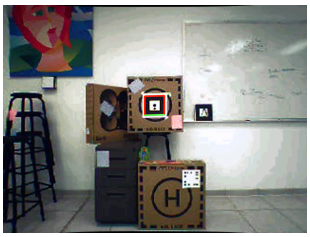


(a)

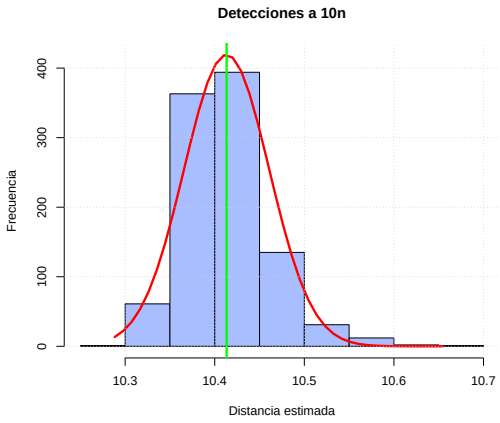


(b)

Figura 5.6: Detecciones a $8n$, media estimada de $8.48n$



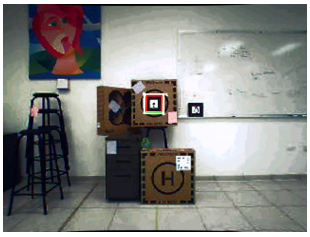
(a)



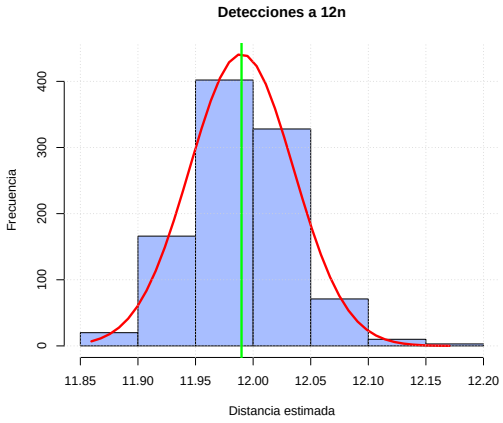
(b)

Figura 5.7: Detecciones a $10n$, media estimada de $10.41n$

A $10n$ del marcador se obtuvo una detección del $100\,\%$, con una confianza media de 0.81. En la Figura 5.7b se muestra que la relación distancia/marcador media estimada fue de $10.41n$. con desviación estándar de $47.56 \times 10^{-3}n$.



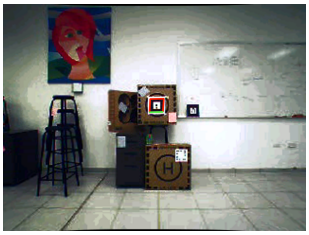
(a)



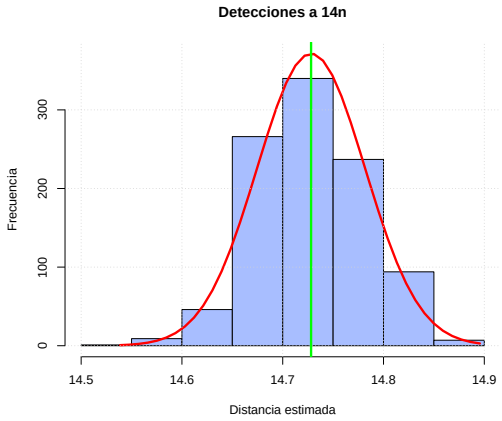
(b)

Figura 5.8: Detecciones a $12n$, media estimada de $11.98n$

Ubicando al **VANT** a una distancia de $12n$ del marcador se detectó en el $100\,\%$ de las fotografías, con una confianza media de 0.79. En la gráfica 5.8a de la Figura 5.8, la relación distancia/marcador media estimada fue de $11.98n$, con una desviación estándar de $45.20 \times 10^{-3}n$.



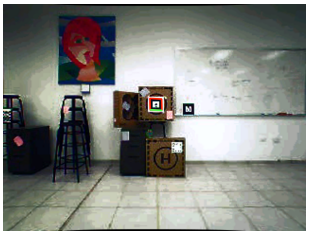
(a)



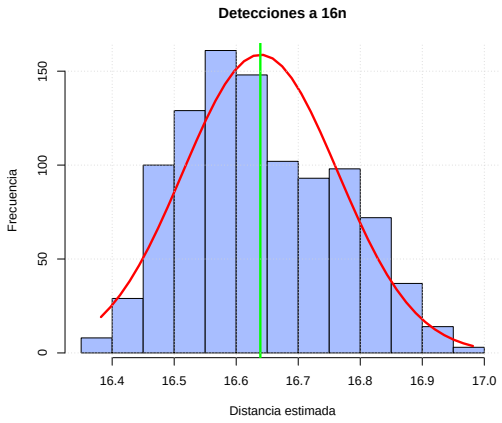
(b)

Figura 5.9: Detecciones a $14n$, media estimada de $14.72n$

A una distancia $14n$ del marcador se obtuvo una detección del $100\,\%$, con una confianza media de 0.78. En la gráfica de la Figura 5.9b se muestra que la relación distancia/marcador media estimada fue de $14.72n$. con desviación estándar de $53.67 \times 10^{-3}n$



(a)

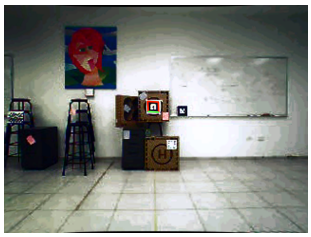


(b)

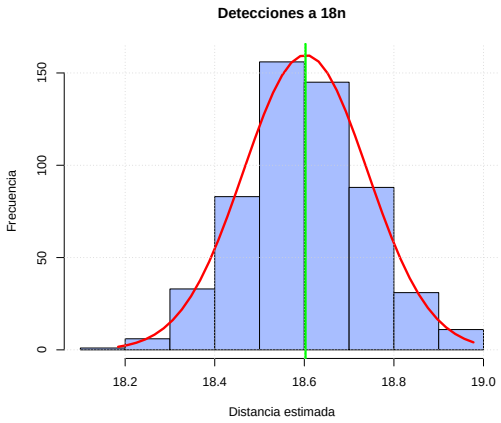
Figura 5.10: Detecciones a $16n$, media estimada de $16.63n$

A $16n$ del marcador se obtuvo una detección del $99.4\,\%$, con una confianza media de 0.73. La Figura 5.10b muestra que la relación distancia/marcador media estimada fue de $16.63n$. con

desviación estándar de $124.87 \times 10^{-3}n$.



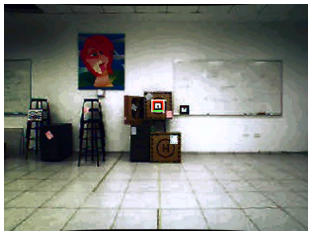
(a)



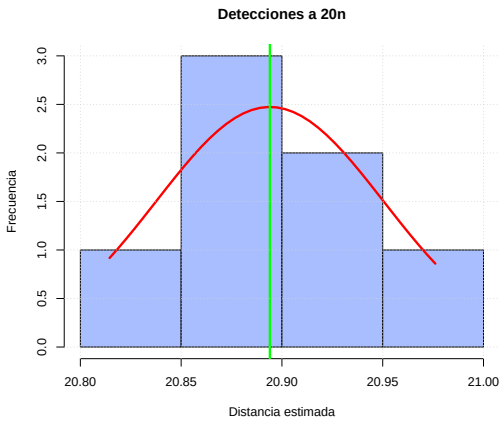
(b)

Figura 5.11: Detecciones a $18n$, media estimada de $18.60n$

Ubicando al **VANT** a una distancia de $18n$ del marcador se detectó en el $55.4\,\%$ de las fotografías, con una confianza media de 0.62. En la Figura 5.11b, la relación distancia/marcador media estimada fue de $18.60n$, con una desviación estándar de $138.37 \times 10^{-3}n$.



(a)



(b)

Figura 5.12: Detecciones a $20n$, media estimada de $20.89n$

A una distancia $20n$ del marcador se obtuvo una detección del $0.7\,\%$, con una confianza media de 0.62. En la gráfica 5.12a de la Figura 5.12 se muestra que la relación distancia/marcador media

estimada fue de $20.89n.$ con desviación estándar de $56.45 \times 10^{-3}n.$

Los datos obtenidos de los experimentos de detección de marcador a una distancia conocida se presentan en la Tabla 5.2.

d_r	d_e	Error	Detecciones($\%$)	Confianza	$sigma$
2	2.07	0.07	100	0.93	8.72×10^{-4}
4	4.10	0.10	100	0.90	72.55×10^{-4}
6	6.23	0.23	100	0.88	86.61×10^{-4}
8	8.48	0.48	100	0.85	15.76×10^{-3}
10	10.41	0.41	100	0.81	47.56×10^{-3}
12	11.98	-0.02	100	0.79	45.20×10^{-3}
14	14.72	0.72	100	0.78	53.67×10^{-3}
16	16.63	0.63	99.4	0.73	124.87×10^{-3}
18	18.60	0.60	55.4	0.62	138.37×10^{-3}
20	20.89	0.89	0.7	0.62	56.45×10^{-3}

Tabla 5.2: Valores para el módulo de detección de marcadores

Con los datos anteriores podemos construir la Figura 5.13, que muestra los errores entre la distancia estimada y la distancia real. De esta gráfica es posible señalar, considerando que un error tolerable en la estimación de posición para lograr una buena triangulación es de $0.23n$, que la distancia desde la cual se deben realizar las fotografías es de $6n$. Sin embargo, es importante resaltar que este valor depende de la resolución original de la cámara, por lo que empleando una cámara de mayor resolución es posible tomar fotografías fiables desde una distancia mayor, sin incrementar el tamaño del marcador.

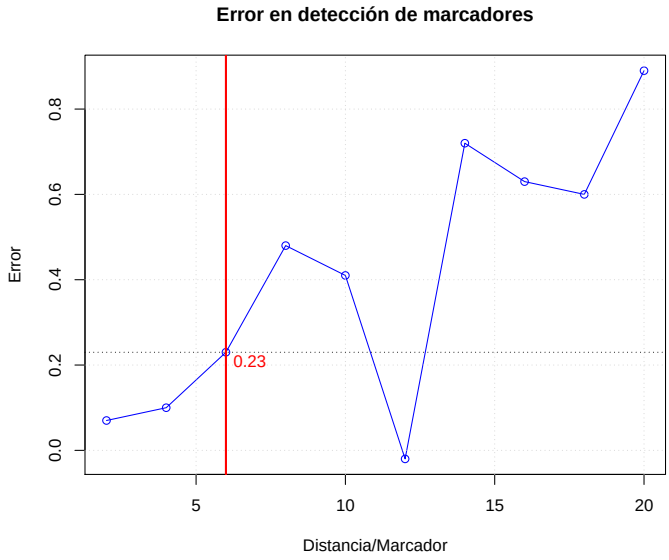
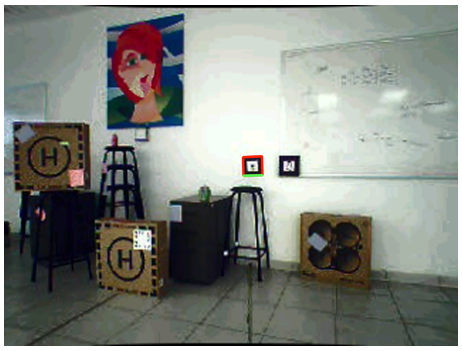
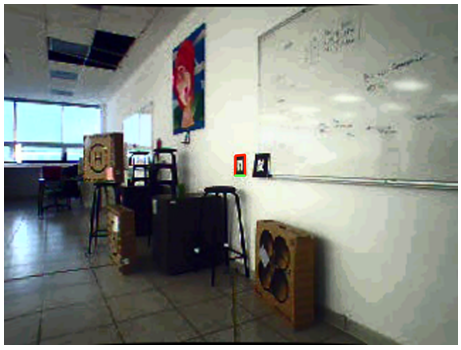


Figura 5.13: Relación entre distancia al marcador y error de detección



(a)

(b)



(c)

Figura 5.14: Ejemplo de escena del marcador desde un ángulo lateral

Ángulo	Detecciones($\%$)	Confianza
$0°$	100	78.43×10^{-2}
$20°$	87.1	71.74×10^{-2}
$40°$	78.1	64.08×10^{-2}
$60°$	22.5	62.16×10^{-2}

Tabla 5.3: Detección de marcador desde un ángulo lateral

Para determinar el ángulo máximo con el cual se realiza una detección del marcador, se consideró una distancia al marcador de $d_f = 14n$. La estimación de la posición de detección efectiva se realizó variando el ángulo con el que el **VANT** observa al marcador. Se inició desde una posición frontal al marcador con incrementos de $20°$. Se obtuvieron 1000 fotografías con perspectiva de $20°$, $40°$ y $60°$ respecto al marcador, como se muestra en la Figura 5.14. Los datos obtenidos de este experimento se resumen en la Tabla 5.3.

En la Figura 5.15 se muestra la gráfica para los ángulos evaluados y el porcentaje de detecciones exitosas. El número de detecciones es proporcional a la confianza de detección del marcador, como se muestra en la Figura 5.16. Con base en estos datos se selecciona el ángulo máximo para la detección del marcador en $40°$, considerando la confianza y el número de detecciones.

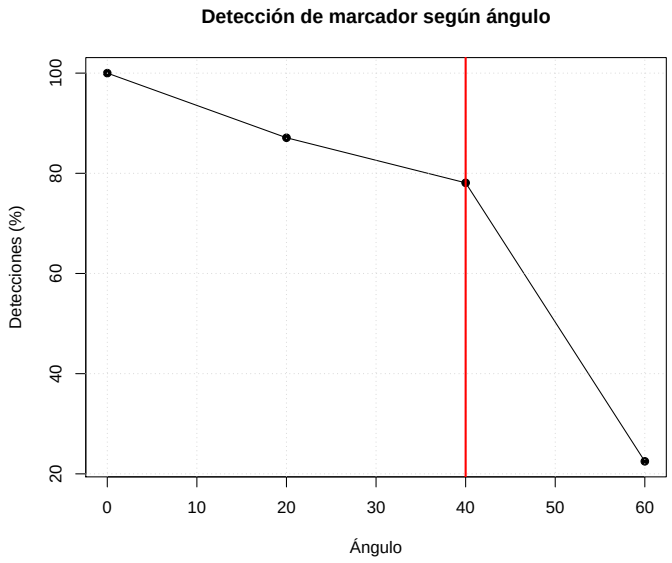


Figura 5.15: Detecciones exitosas con diferentes ángulos al marcador

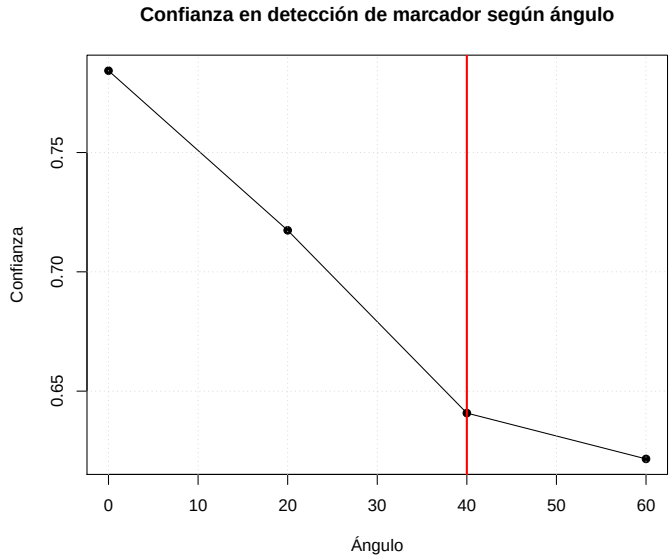


Figura 5.16: Relación entre ángulo al marcador y confianza de detección

Los experimentos realizados permiten determinar los valores con los cuales el módulo de detección de marcadores es fiable. Estos se presentan en la Tabla 5.5.

Distancia frontal máxima	d_f	$6n$
Apertura máxima	θ	$80°$

Tabla 5.4: Valores para el módulo de detección de marcadores

El éxito en la detección del marcador depende de la iluminación de la escena y de la resolución de las imágenes. Para las fotografías obtenidas con la cámara frontal del **VANT** empleado en la experimentación, es notorio que el efecto de viñeteado y la baja calidad de las imágenes influyen considerablemente al módulo de detección de marcadores.

5.4 Odometría

El desplazamiento y control del **VANT** con el que se realizan los experimentos requiere se indique la velocidad a la que se desea se desplace en alguno de sus tres ejes y el ángulo sobre el eje z. Sin embargo, mantener la velocidad por algún intervalo de tiempo no garantizan que se desplace la distancia deseada, debido a los efectos de inercia, fuerza centrífuga y errores no sistemáticos o del medio.

Para reducir el error de la distancia que se desea desplazar el **VANT** y la distancia recorrida realmente, se modificaron las instrucciones de desplazamiento para que, en función de la velocidad deseada, incluyeran un intervalo de tiempo a dicha velocidad y un periodo final con una velocidad opuesta de frenado que minimiza la inercia. Estas instrucciones se realizaron mediante ajuste manual de las velocidades, considerando un error de ± 15cm para cada eje y $10°$ para las rotaciones en el eje z. Los desplazamientos considerados fueron 40 cm, 1 m, 1.20 m, 2 m y los ángulos de guiñada (*yaw*) de $15°$, $30°$ y $90°$.

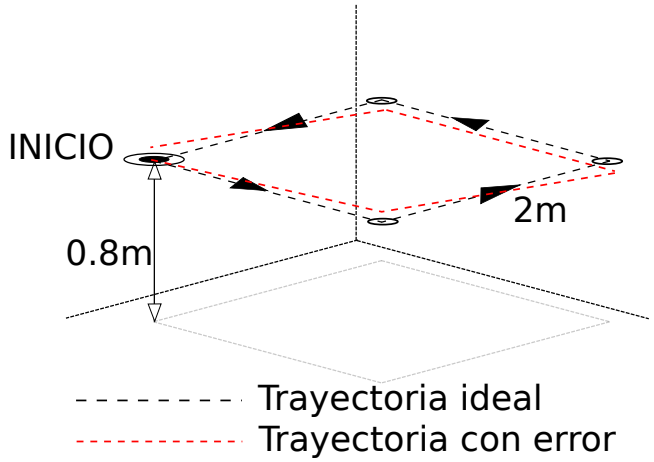


Figura 5.17: Experimento de odometría

Para evaluar el desempeño de las nuevas instrucciones de desplazamiento junto con la odometría,

se siguió el experimento presentado por Borenstein *et al.* en [6], el cual permite medir el error de odometría para un robot móvil. Este experimento considera el recorrido de una trayectoria cuadrada en ambos sentidos, como se muestra en la Figura 5.17. El **VANT** se coloca en una ubicación considerada el origen (x_0, y_0) elevándose a una altura de $80cm$ y realiza una trayectoria cuadrada, en el plano xy, hasta regresar la posición inicial. La diferencia real entre la posición final absoluta medida de forma manual con respecto al origen y la posición final estimada por odometría determinan el error sistemático de odometría. Se obtienen un conjunto de posiciones de error por odometría, denotado como ϵ_x, ϵ_y con las Ecuaciones 5.2.

$$\begin{aligned} \epsilon_x &= x_{real} - x_{calc} \\ \epsilon_y &= y_{real} - y_{calc} \\ \epsilon_\theta &= \theta_{real} - \theta_{calc} \end{aligned} \tag{5.2}$$

donde $\epsilon_x, \epsilon_y, \epsilon_\theta$ corresponden a los errores de posición y orientación final producto de la odometría, $x_{real},\ y_{real},\ \theta_{real}$ la posición real del robot y $x_{calc},\ y_{calc},\ \theta_{calc}$ la posición y orientación del robot calculada por la odometría. El recorrido a lo largo del cuadrado es en ambos sentidos, horario y antihorario, para aislar los errores sistemáticos.

Se realizaron las trayectorias cinco veces para cada sentido. Con las Ecuaciones 5.3 se calcula el desplazamiento para ambos sentidos, horario r_h y antihorario r_{ah}, para un cuadrado con lados de dos metros.

$$CG(X_{h/ah}) = \frac{1}{n}\sum_{i=1}^{n} x_{i,h/ah}$$
$$CG(Y_{h/ah}) = \frac{1}{n}\sum_{i=1}^{n} y_{i,h/ah} \tag{5.3}$$

$$r_{h/ah} = \sqrt{x_{h/ah}^2 + y_{h/ah}^2} \tag{5.4}$$

El valor mayor de (r_h,r_{ah}) de las Ecuaciones 5.4 determina la *medida de la exactitud de odometría para errores sistemáticos* $E_{max,syst}$. Las gráficas para las trayectorias realizadas en sentido horario se muestra en la Figura 5.19, así como en la Figura 5.20 se muestran las cinco trayectorias eralizadas en sentido antihorario. La línea azul corresponde a la trayectoria ideal, las marcas verde y roja indican respectivamente el origen y final de la trayectoria. A manera de ejemplo se muestra en la Figura 5.18 una gráfica tridimensional correspondiente a la primera trayectoria en sentido horario presentada, en la Subfigura 5.19a.

Los resultados para las pruebas de odometría, empleando solamente información de los sensores inerciales del **VANT**, se resumen en la Tabla 5.5. El error sistemático es de 0.41 metros, siendo más notorio en los recorridos de sentido antihorario. En la práctica, las instrucciones de desplazamiento requieren recalibración para compensar estos errores. Sin embargo, en la presente propuesta se compensa dicho error con el cálculo de la posición mediante información visual.

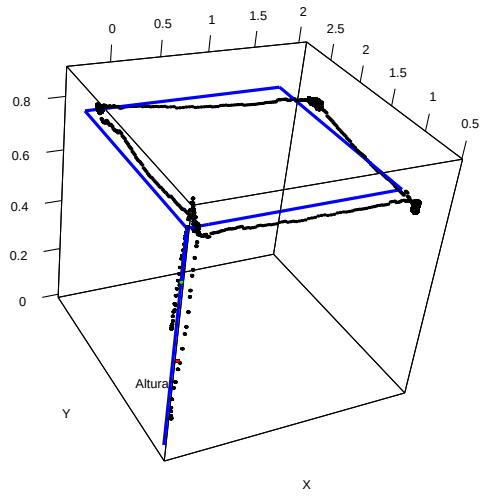


Figura 5.18: Gráfica tridimensional de odometría para la trayectoria de la Subfigura 5.19a

Horario		Antihorario	
$CG(X_h)$	$CG(Y_{h/ah})$	$CG(X_{ah})$	$CG(Y_{ah})$
0.38 m	-0.14 m	0.12 m	0.29 m
r_h	0.41 m	r_{ah}	0.31 m
$E_{max,syst}$	0.41 m		

Tabla 5.5: Medida de la exactitud de odometría para errores sistemáticos

Considerando una posición conocida para el origen (punto de despegue) del **VANT**, es posible estimar la posición final en donde se localizará el **VANT** después de un desplazamiento. Dado que cada nueva estimación de la posición depende de la anterior, el error e incertidumbre se acumula con el tiempo. La incertidumbre entre la posición real P del robot (desconocida) y la estimación de la posición $\hat{P}$ obtenida por odometría, se expresa en términos estadísticos a través de la desviación estándar asociada con $\hat{P}$. Gráficamente, la incertidumbre queda representada por medio de elipses, que reflejan la correlación entre las variables reales y estimadas, y el tamaño de las elipses es

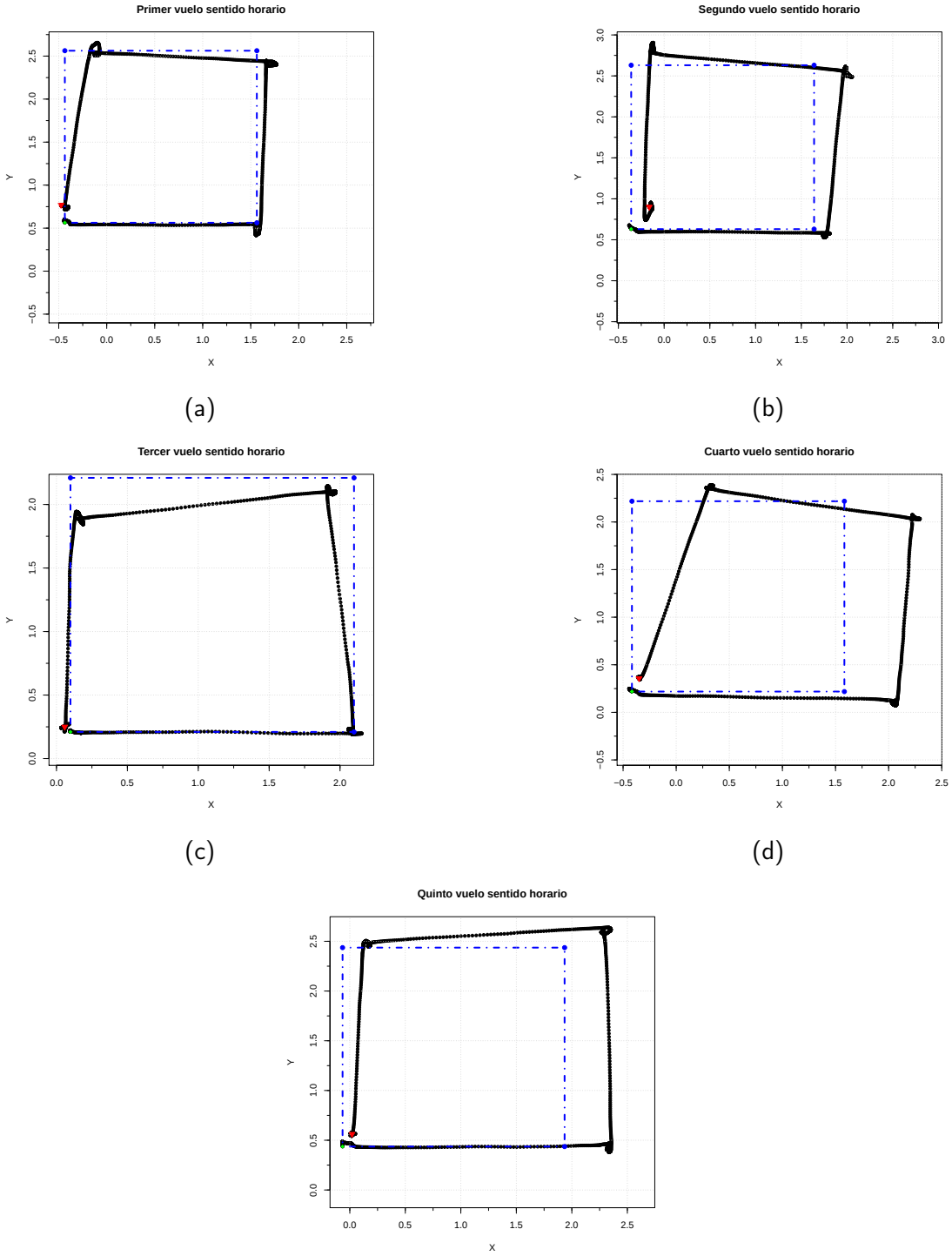
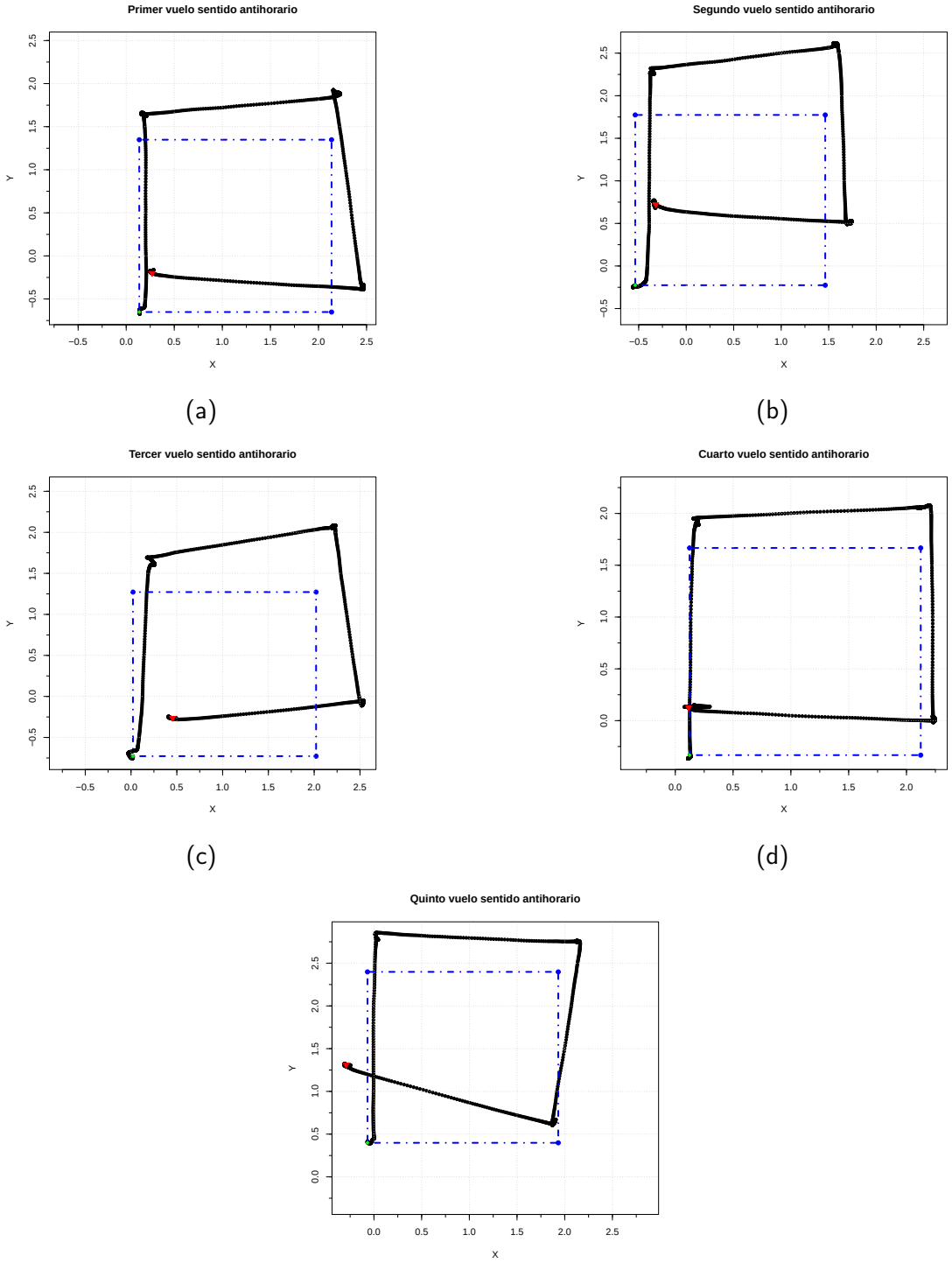


Figura 5.19: Trayectorias para validar odometría en sentido horario



(a) (b) (c) (d) (e)

Figura 5.20: Trayectorias para validar odometría en sentido antihorario

proporcional a la incertidumbre, como se presentan Smith y Cheeseman en [54] y Krajnik *et al.* en [24]. Los ejes de la elipse corresponden a los valores propios de las matrices de covarianza, en función a la distancia recorrida.

En la Figura 5.21 se muestra la incertidumbre de la estimación de posición para la trayectoria de la Subfigura 5.19a. La línea azul corresponde a la trayectoria ideal y la línea negra al desplazamiento calculado por odometría. A medida que se desplaza el **VANT**, los errores acumulados en su posición aumentan, por lo que el tamaño y dimensiones de las elipses reflejan dicha incertidumbre. A cada instante, la posición real del **VANT** se encuentra en algún punto al interior de la elipse de incertidumbre alrededor de la posición estimada en dicho momento.

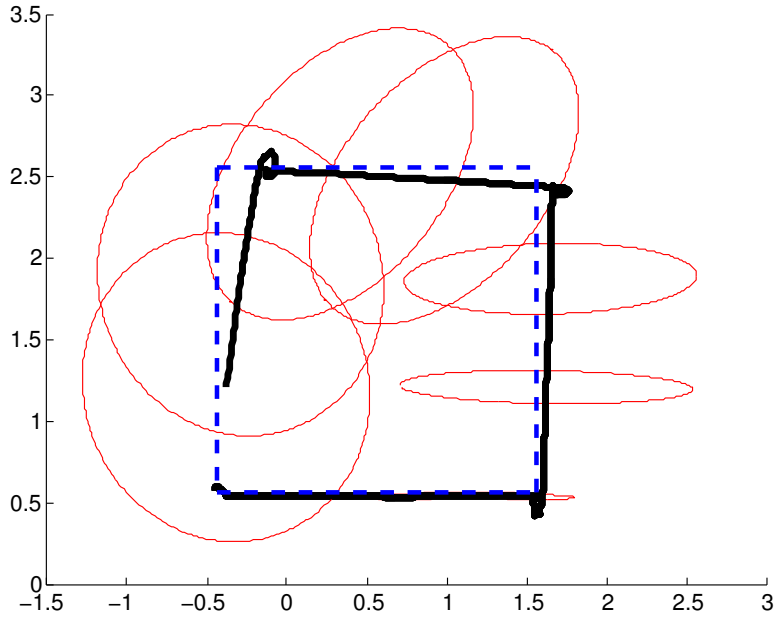


Figura 5.21: Incertidumbre de la odometría representada por elipses, para trayectoria de la Subfigura 5.19a

Al ubicar una posición real conocida en la escena, como por ejemplo un marcador visual conocido, la incertidumbre se vuelve cero y por lo tanto también el tamaño de los elipses. De esta forma es posible reducir el error de incertidumbre, tomando como base posiciones conocidas que pueden ser

identificadas visualmente.

5.5 Reconstrucción tridimensional

5.5.1 Emparejamiento de imágenes

Pare evaluar el emparejamiento de imágenes mediante detección automática de puntos de interés, se evalúan los algoritmos de detección de puntos **FAST** y **BRISK**, y para el cálculo de los descriptores los algoritmos **BRIEF** y **BRISK**. El propósito de este experimento es determinar la combinación eficaz de detección de puntos en conjunto con descriptores que permitan realizar emparejamiento correcto, con tiempos compatibles a la aplicación de reconstrucción de una fachada.

Se consideraron las combinaciones de los métodos para detección y descripción de la siguiente forma, según el algoritmo detector de puntos y el de cálculo de descriptores:

- FAST/BRIEF
- FAST/BRISK
- BRISK/BRISK
- BRISK/BRIEF

Las imágenes con las que se probaron estas combinaciones pertenecen al conjunto de datos *Fountain-R25*, proporcionado por Strecha *et al.* en [57]. Este conjunto está integrado por 25 fotografías, con sus respectivas matrices de valores intrínsecos y extrínsecos de la cámara. Esta secuencia de imágenes se emplea comúnmente como *benchmark*, en la literatura de reconstrucción tridimensional a partir de imágenes.

Debido a que el **VANT** envía imágenes de 960×720 pixeles y la resolución original de las imágenes de *Fountain-R25* es de 3072×2048, estas últimas fueron reducidas a un tamaño de 768×512 (submuestreo por cuatro). Así mismo, los valores de la matriz K de parámetros de la cámara, proporcionada con el conjunto de imágenes, fueron transformados en la misma proporción. Una muestra de este conjunto de imágenes se presenta en la Figura 5.22.

(a) (b) (c) (d)

Figura 5.22: Subconjunto de imágenes de *Fountain-R25* [57]

Para la evaluación del desempeño se considera de gran importancia el tiempo de procesamiento y el número de puntos característicos emparejados satisfactoriamente. Se asume la efectividad de los detectores y descriptores de puntos para lograr invarianza a transformaciones geométricas, enfocando las pruebas experimentales en los tiempos de procesamiento para lograr la reconstrucción en línea. El emparejamiento se realiza mediante los métodos cuadrático exhaustivo y por similitud proporcional, descrito en el Algoritmo 9. Para evaluar si el emparejamiento de los puntos entre dos imágenes es coherente con el conjunto de correspondencias, se calcula la matriz fundamental que relaciona ambas imágenes mediante el algoritmo de **RANSAC**.

Para cada una de las combinaciones detector/descriptor se realizaron 40 pruebas, donde se emparejaron las 25 imágenes del conjunto. Se evalúan solamente las etapas de detección, descripción y emparejamiento, sin considerar la recuperación de características tridimensionales. Las imágenes

son procesadas de forma ordenada, una imagen con su inmediata siguiente, registrando los tiempos de cada proceso.

Para la detección con **FAST** se considera un umbral de 5 sin supresión de no máximos, en el caso del detector **BRISK** el umbral también es de 5 y se consideran las cuatro octavas recomendadas por sus autores. Es importante recordar que la diferencia en la detección de puntos característicos entre **BRISK** y **FAST** consiste en que esta última realiza la detección en pirámides de la imagen original. Para la descripción de puntos por **BRIEF** y **BRISK** se considera un vector de 512 *bits*, es decir, 64*bytes*. Finalmente, para los algoritmos de emparejamiento no se efectúa la prueba de simetría.

A continuación, se presentan los resultados para las etapas de detección y descripción, para posteriormente relacionar los valores de tiempo y eficacia con los resultados de la etapa de emparejamiento, y finalmente seleccionar la combinación de algoritmos que obtenga un mayor número de puntos coherentes con la homografía general en el menor tiempo.

5.5.1.1. Detección de puntos característicos

Para las 25 imágenes de la fuente se realizó la detección de puntos característicos por medio de los algoritmos **BRISK** y **FAST**. La mediana de puntos obtenidos en las 40 pruebas, así como la mediana de tiempo para cada imagen se muestra en la Tabla 5.6.

En la gráfica de la Figura 5.23 se presenta la cantidad de puntos característicos detectados por imagen al utilizar el algoritmo **BRISK**, mientras que en la gráfica de la Figura 5.24 presenta los resultados del algoritmo **FAST**.

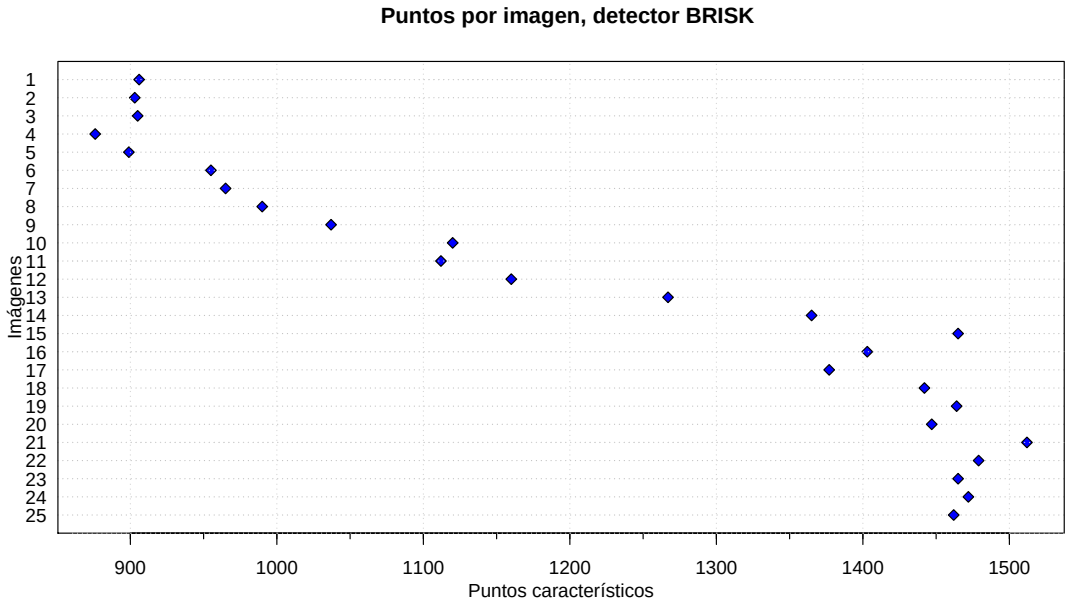


Figura 5.23: Puntos característicos detectados por el algoritmo **BRISK** para el conjunto *Fountain-R25*

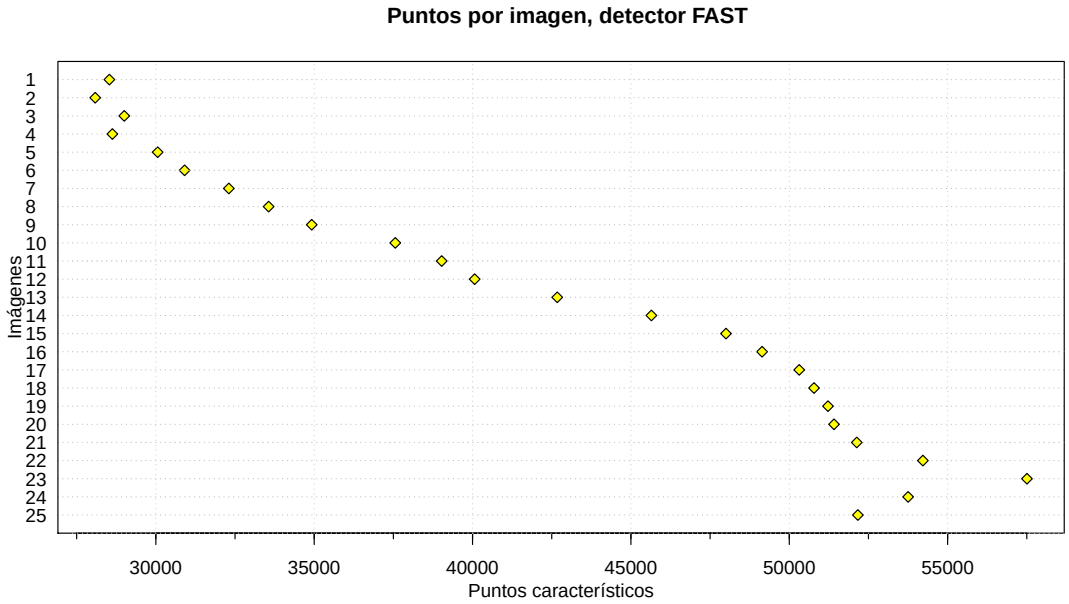
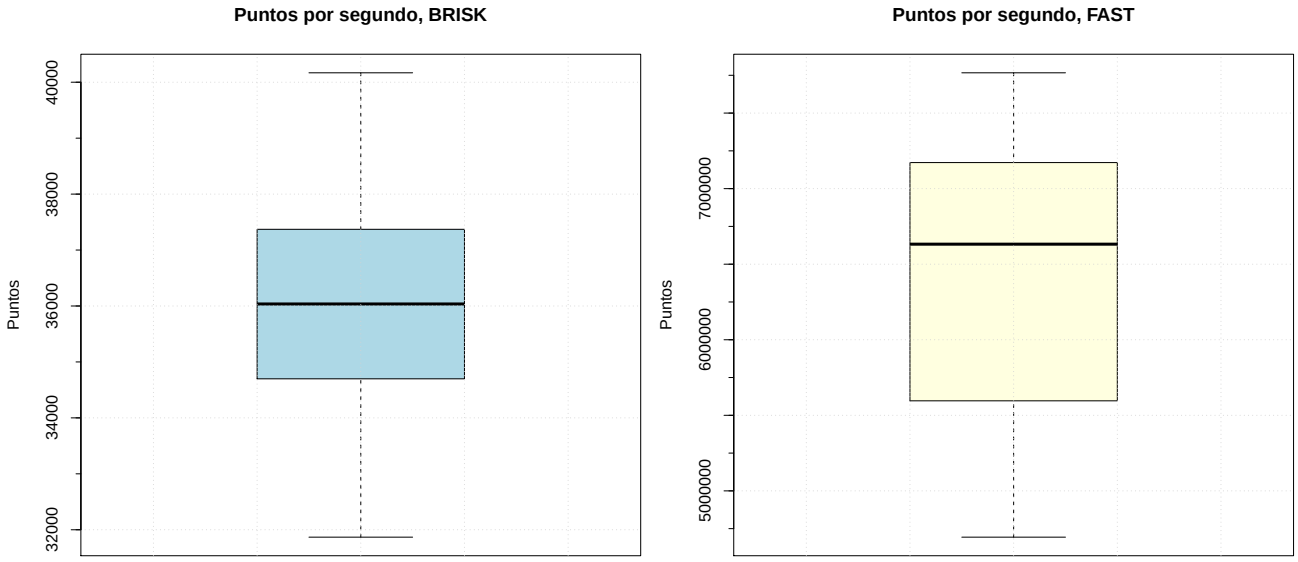


Figura 5.24: Puntos característicos detectados por el algoritmo **FAST** para el conjunto *Fountain-R25*

En la Figura 5.25 se presenta una comparativa mediante gráfica de caja entre los algoritmos

BRISK y **FAST** con los datos presentados en la Tabla 5.6. En esta gráfica se muestra la relación entre el total de puntos obtenidos y el tiempo de procesamiento para obtenerlos. Los algoritmos se presentan por separado por claridad y diferencia de escalas.



(a) (b)

Figura 5.25: Puntos por segundo con **BRISK** y **FAST** para el conjunto *Fountain-R25*

Para el algoritmo **BRISK** se obtiene una mediana de 36035.9 puntos por segundo, en el caso de **FAST** este valor es de 6632494.46 puntos por segundo, lo que muestra la mayor capacidad de detección del algoritmo **FAST**.

Imagen	BRISK			FAST		
	Puntos	Tiempo(seg)	Puntos/seg	Puntos	Tiempo(seg)	Puntos/seg
1	906	2.763×10^{-2}	32785.46	28533	0.6079×10^{-2}	4693668.74
2	903	2.619×10^{-2}	34479.99	28085	0.5444×10^{-2}	5159345.42
3	905	2.723×10^{-2}	33240.65	29003	0.5713×10^{-2}	5076471.76
4	876	2.749×10^{-2}	31867.76	28628	0.5655×10^{-2}	5062109.33
5	899	2.739×10^{-2}	32820.16	30057	0.5603×10^{-2}	5364763.59
6	955	2.731×10^{-2}	34974.51	30908	0.5635×10^{-2}	5484673.51
7	965	2.817×10^{-2}	34260.92	32305	0.5773×10^{-2}	5595848.28
8	990	2.853×10^{-2}	34698.13	33561	0.5781×10^{-2}	5804925.05
9	1037	2.989×10^{-2}	34698.29	34922	0.5770×10^{-2}	6052601.93
10	1120	3.093×10^{-2}	36215.37	37560	0.5989×10^{-2}	6271623.41
11	1112	3.152×10^{-2}	35281.65	39026	0.6091×10^{-2}	6406768.92
12	1160	3.220×10^{-2}	36028.43	40067	0.6215×10^{-2}	6446334.71
13	1267	3.365×10^{-2}	37650.96	42674	0.6434×10^{-2}	6632494.47
14	1365	3.529×10^{-2}	38684.56	45647	0.6647×10^{-2}	6867215.58
15	1465	3.647×10^{-2}	40168.24	48002	0.6757×10^{-2}	7104029.74
16	1403	3.622×10^{-2}	38735.08	49142	0.6907×10^{-2}	7114635.95
17	1377	3.684×10^{-2}	37375.31	50314	0.7030×10^{-2}	7157183.78
18	1442	3.859×10^{-2}	37370.68	50782	0.7070×10^{-2}	7182297
19	1464	3.861×10^{-2}	37918.72	51222	0.7103×10^{-2}	7211532.37
20	1447	3.930×10^{-2}	36820.37	51413	0.7282×10^{-2}	7060033.56
21	1512	4.066×10^{-2}	37188.53	52131	0.7268×10^{-2}	7172428.03
22	1479	4.104×10^{-2}	36035.90	54219	0.7409×10^{-2}	7317675.58
23	1465	4.150×10^{-2}	35301.20	57505	0.7404×10^{-2}	7766474.98
24	1472	4.072×10^{-2}	36151.89	53753	0.7306×10^{-2}	7357357.36
25	1462	3.966×10^{-2}	36866.78	52166	0.7112×10^{-2}	7335339.45

Tabla 5.6: Puntos característicos por imagen empleando **BRISK** y **FAST**

5.5.1.2. Descripción de puntos característicos

A partir de los puntos característicos obtenidos en la prueba anterior, se realizó el cálculo de descriptores con los algoritmos **BRIEF** y **BRISK**. Se midió la capacidad del algoritmo para crear descriptores, calculando el número de puntos descritos por unidad de tiempo.

En la Figura 5.26 se muestra la gráfica correspondiente al cálculo de descriptores por medio de los algoritmos **BRIEF** y **BRISK**, a partir de los puntos detectados utilizando el algoritmo **BRISK**.

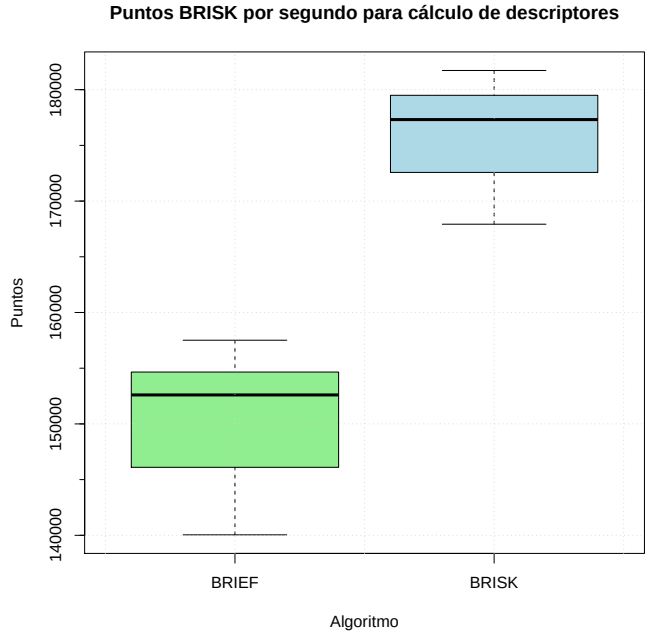


Figura 5.26: Cálculo de descriptores con los algoritmos **BRIEF** y **BRISK** para los puntos detectados por **BRISK**

La mediana de descripción de puntos por segundo, para los puntos característicos detectados por **BRISK**, por medio del algoritmo **BRIEF** es de 152600.751 puntos por segundo, y la descripción con el algoritmo **BRISK** es de 177315.909 puntos por segundo.

La gráfica correspondiente al cálculo de descriptores de los puntos característicos obtenidos por medio de **FAST** se muestra en la Figura 5.27. La descripción se realizó, de la misma forma, con los algoritmos **BRIEF** y **BRISK**.

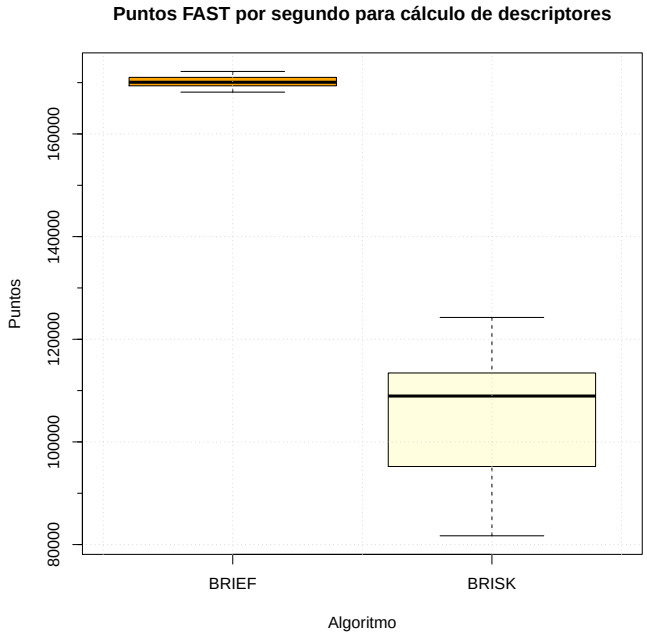


Figura 5.27: Cálculo de descriptores con los algoritmos **BRIEF** y **BRISK** para los puntos detectados por **FAST**

La mediana de descripción de puntos por segundo, para los puntos característicos detectados por **FAST**, por medio del algoritmo **BRIEF** es de 170073.295 y la descripción con el algoritmo **BRISK** es de 108940.180. En la Tabla 5.7 se resume la descripción de los puntos por segundo para las características detectadas con **BRISK** y **FAST**, descritas con los algoritmos **BRIEF** y **BRISK**.

BRISK		FAST	
BRIEF	BRISK	BRIEF	BRISK
152600.751	177315.909	170073.2949	108940.1804

Tabla 5.7: Puntos por segundo para cálculo de descriptores

5.5.1.3. ***Emparejamiento de imágenes***

Los puntos por segundo procesados, junto con los emparejamientos exitosos entre las imágenes permitirán seleccionar la mejor combinación de algoritmos para la construcción de un modelo tridimensional en línea. Del éxito en la etapa de emparejamiento depende plenamente el cálculo de la construcción de dicho modelo. Los algoritmos de emparejamiento evaluados son el exhaustivo o cuadrático y el de vecinos más cercanos. Se consideran emparejamientos correctos aquellos que resultan coherentes con el modelo de homografía proyectiva, descrita por una matriz fundamental que describa un movimiento de la cámara.

Para fines ilustrativos se presentan el total de parejas obtenidas así como el subconjunto de éstas que se considera correcto en la Tabla 5.8 y en las Figuras 5.28 y 5.29. De estas gráficas se puede observar que, en general, la detección de puntos empleando el algoritmo **FAST** resulta en una mayor proporción de puntos detectados y emparejados correctamente, comparado con el algoritmo de detección **BRISK**.

	BRISK			
	BRIEF/Cuad.	**BRIEF**/vecinos	**BRISK**/Cuad.	**BRISK**/vecinos
Correctas	360	316	208	84.5
Incorrectas	344	828	515	1074.5
Totales	704	1144	723	1159
	FAST			
	BRIEF/Cuad.	**BRIEF**/vecinos	**BRISK**/Cuad.	**BRISK**/vecinos
Correctas	24651.5	35840.5	27887.5	34152.5
Incorrectas	2536	5530	2965.5	8008.5
Totales	27187.5	41370.5	30853	42161

Tabla 5.8: Correspondencias para las distintas combinaciones de algoritmos evaluados

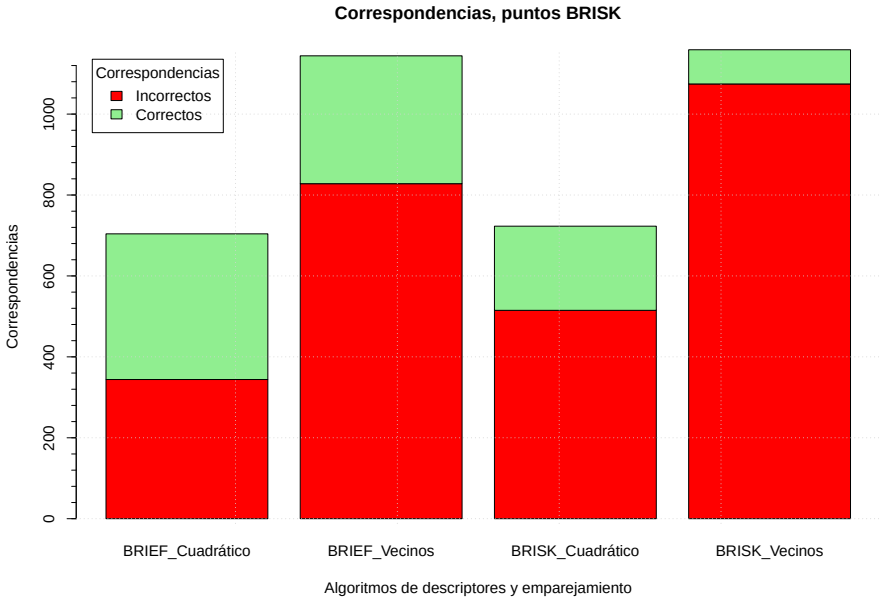


Figura 5.28: Correspondencias para puntos detectados con **BRISK**

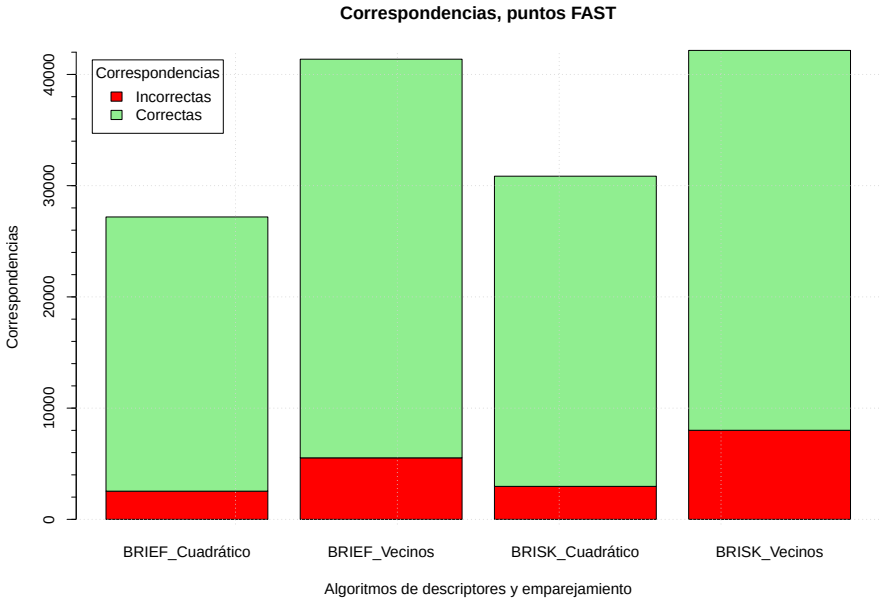


Figura 5.29: Correspondencias para puntos detectados con **FAST**

5.5.1.4. *Selección de algoritmos para la propuesta de solución*

En la Figuras 5.30 y 5.31 se muestran las gráficas de caja de los tiempos de procesamiento para los datos obtenidos en los experimentos previamente en las Figuras 5.28 y 5.29.

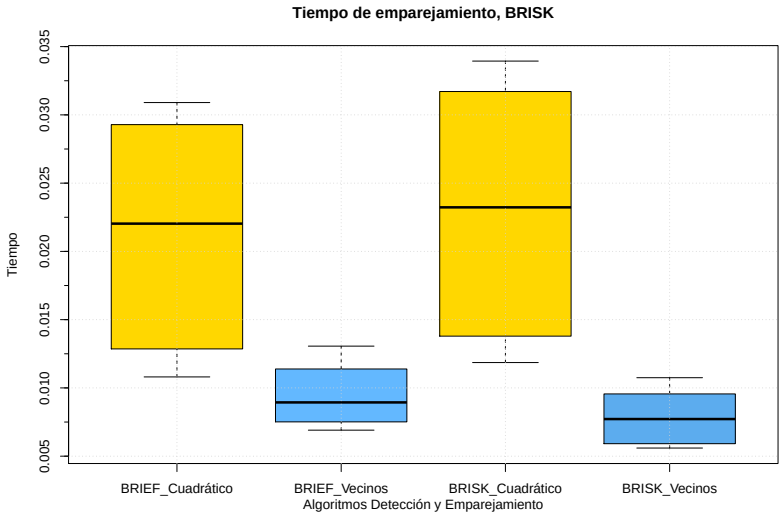


Figura 5.30: Tiempo para estimar correspondencias de puntos detectados con **BRISK**

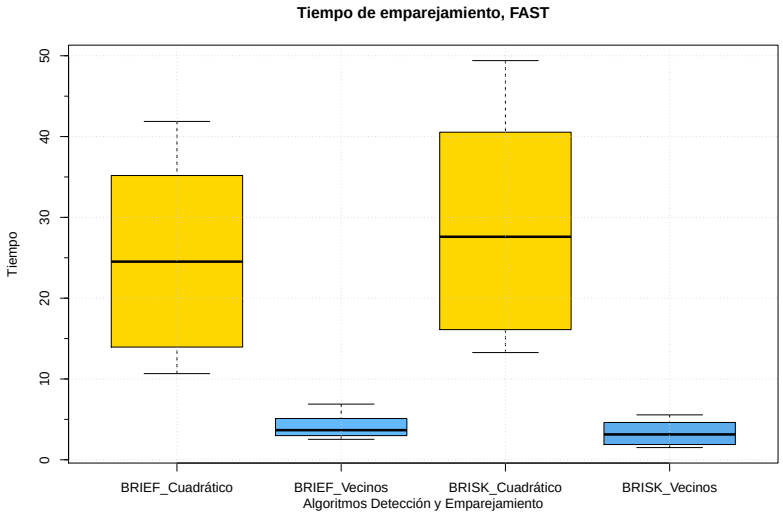


Figura 5.31: Tiempo para estimar correspondencias de puntos detectados con **FAST**

Estas gráficas están en función de los puntos detectados por cada algoritmo. La métrica empleada para la selección de los algoritmos depende del número de correspondencias correctas y el tiempo requerido para obtenerlas, de acuerdo a la Ecuación 5.5, puesto que finalmente, para nuestra aplicación, lo más importante es obtener el mayor número de correspondencias correctas en un tiempo dado, ya que cada correspondencia representará un punto del modelo tridimensional de la edificación.

$$tiempo\ por\ correspondencia = \frac{correspondencias\ correctas}{tiempo\ de\ emparejamiento} \tag{5.5}$$

En la Tabla 5.9 se resumen los resultados para todas las etapas por algoritmo, de las cuales se obtiene el tiempo total de cada combinación evaluada. Así mismo se aplica la Ecuación 5.5 a los valores obtenidos para medir el desempeño de la combinación detector/descriptor. En la Figura 5.32 se grafican las correspondencias correctas por segundo para cada una de las combinaciones.

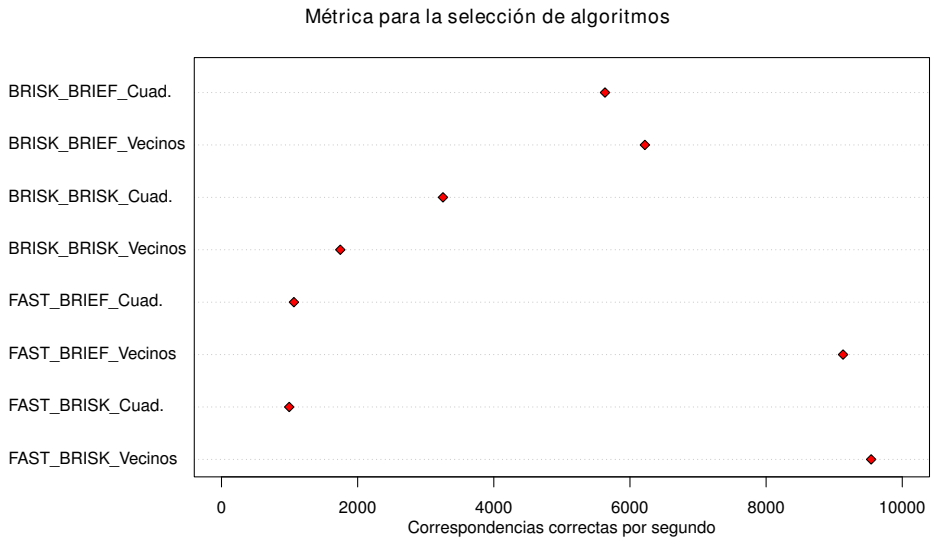


Figura 5.32: Correspondencias correctas por segundo, para las combinaciones de algoritmos evaluados

De los resultados anteriores, podemos concluir que los algoritmos que arrojan el mayor número de correspondencias correctas por unidad de tiempo son la combinación: **FAST** como algoritmo de detección, **BRISK** para la descripción y Vecinos más cercanos para el emparejamiento.

	BRISK			
Etapa	**BRIEF**/Cuad.	**BRIEF**/vecinos	**BRISK**/Cuad.	**BRISK**/vecinos
Detección(seg)	3.37×10^{-2}	3.37×10^{-2}	3.37×10^{-2}	3.37×10^{-2}
Descripción(seg)	0.822×10^{-2}	0.822×10^{-2}	0.706×10^{-2}	0.706×10^{-2}
Emparejamiento(seg)	2.20×10^{-2}	0.894×10^{-2}	2.32×10^{-2}	0.772×10^{-2}
Tiempo total(seg)	6.39×10^{-2}	5.08×10^{-2}	6.39×10^{-2}	4.84×10^{-2}
Corresp. correctas	360	316	208	84.5
Métrica(corresp/seg)	5633.28	6219.19	3253.22	1744.848
	FAST			
Etapa	**BRIEF**/Cuad.	**BRIEF**/vecinos	**BRISK**/Cuad.	**BRISK**/vecinos
Detección	6.43×10^{-3}	6.43×10^{-3}	6.43×10^{-3}	6.43×10^{-3}
Descripción	2.52×10^{-1}	2.52×10^{-1}	4.33×10^{-1}	4.33×10^{-1}
Emparejamiento	2.29×10^{1}	3.67	2.76×10^{1}	3.14
Tiempo total	2.32×10^{1}	3.93	2.80×10^{1}	3.58
Corresp. correctas	24651.5	35840.5	27887.5	34152.5
Métrica	1063.60	9130.20	994.63	9542.87

Tabla 5.9: Resumen de resultados de algoritmos evaluados para el emparejamiento de imágenes

5.5.2 Construcción del modelo tridimensional mediante imágenes

Con los algoritmos de detección, descripción y emparejamiento, se evaluó la recuperación de características tridimensionales. Además del tiempo requerido para el emparejamiento de dos

imágenes, es necesario considerar el tiempo para el cálculo de la posición de la cámara y la construcción del modelo tridimensional.

Para esta prueba se empleó nuevamente el conjunto de imágenes *Fountain-R25*. Se realizaron 40 pruebas con los algoritmos previamente seleccionados. Para la etapa de cálculo de pose se realiza un emparejamiento adicional, en el cual se seleccionó el algoritmo de vecinos más cercanos con los mismos parámetros que en la experimentación previa. En la Figura 5.33 se muestra el tiempo de procesamiento para cada una de las etapas. La mediana de tiempos para cada una de las etapas se muestran en la Tabla 5.10.

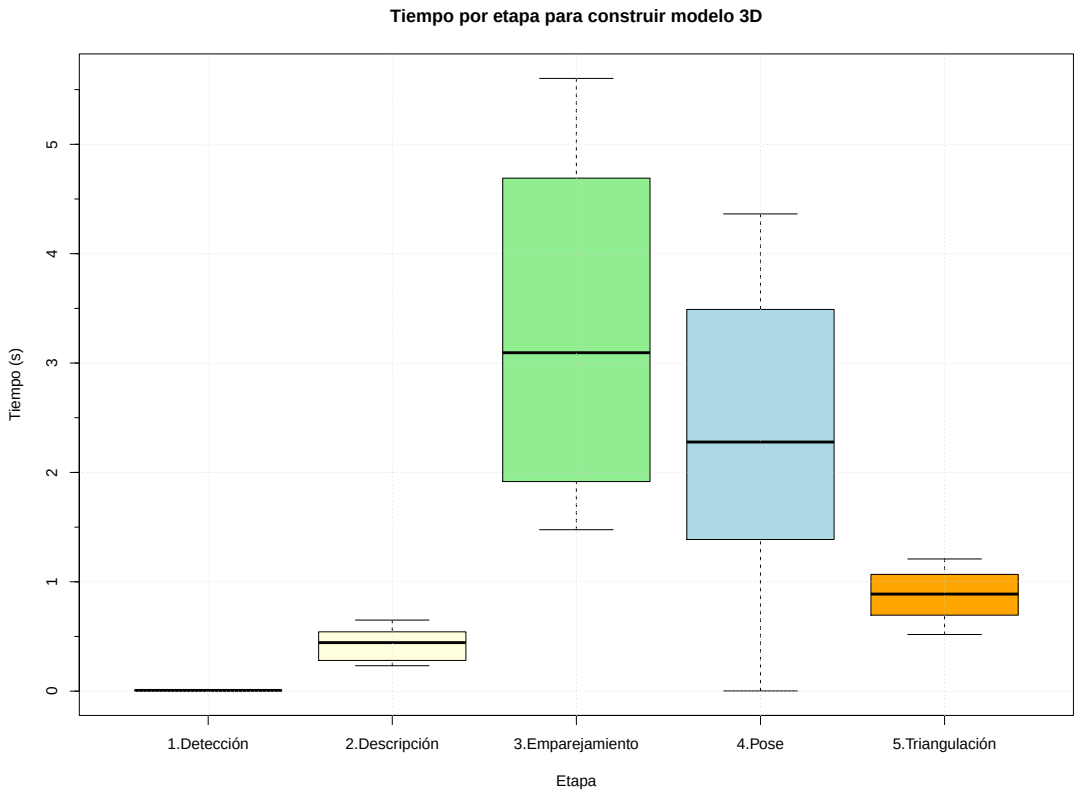


Figura 5.33: Tiempo por etapas para construir modelo tridimensional

Detección	Descripción	Emparejamiento	Pose	Triangulación
6.8178×10^{-3}	4.44×10^{-1}	3.094	2.278	8.88×10^{-1}
Tiempo total estimado			6.71segundos	

Tabla 5.10: Tiempo para cada una de las etapas durante la construcción del modelo tridimensional

Un modelo generado durante las pruebas se muestra en la Figura 5.34. De las 25 imágenes de *Fountain-R25*, se extrae una nube de puntos que, en promedio, incluye 33915.75 puntos tridimensionales para cada par de imágenes.

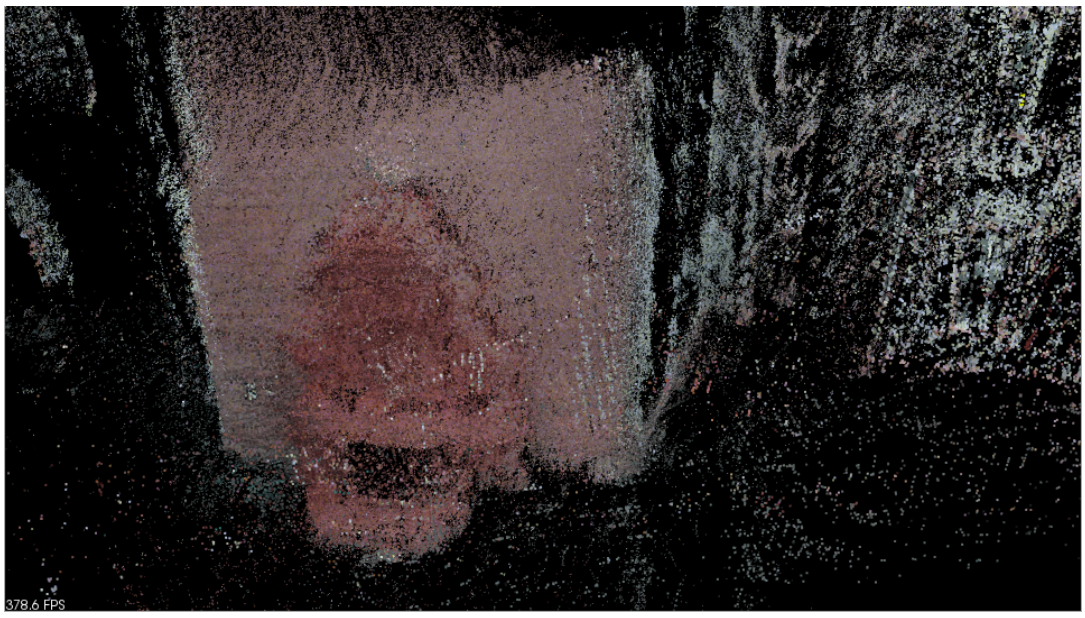


Figura 5.34: Nube de puntos para el conjunto de datos *Fountain-R25*

Para el modelo de la Figura 5.34, es importante mencionar que las imágenes fueron procesadas según el orden de numeración del conjunto. Sin embargo, durante las pruebas de construcción del modelo tridimensional se hizo evidente que la calidad del modelo o nube de puntos obtenido depende en gran medida del par inicial de imágenes, aquellas con las que se estima el *baseline* del modelo tridimensional. Inclusive, el mayor error de reproyección obtenido, con una mediana de 4.56,

corresponde a este par de imágenes inicial, ya que los pares subsecuentes de imágenes tienen un error de reproyección menor a 0.5.

Se observó entonces que los requisitos, para obtener un *baseline* apropiado, son que en el par inicial de imágenes exista poco movimiento de la cámara y sin cambios de escala, de forma que la mayor cantidad de características entre ambas imágenes puedan ser emparejadas. De preferencia, debe apreciarse la totalidad de la fachada, de manera que la escala resulte más estable para los cálculos de pose incrementales. En el caso de el conjunto de datos *Fountain-R25*, estas características las cumple el par de imágenes $(16, 17)$, mostrado en la Figura 5.35. La diferencia entre las imágenes es más notorio del lado derecho.

(a) (b)

Figura 5.35: Imágenes $(16, 17)$, ejemplo de *baseline* sugerido para el conjunto *Fountain-R25*

Con este par de imágenes como *baseline*, se obtuvo el modelo tridimensional mostrado en la Figura 5.36. Puede apreciarse aquí algunos "huecos.[en] el modelo: la pared que une la fuente con la pared del fondo no está suficientemente representada en el modelo tridimensional, así como los detalles laterales de la fuente.

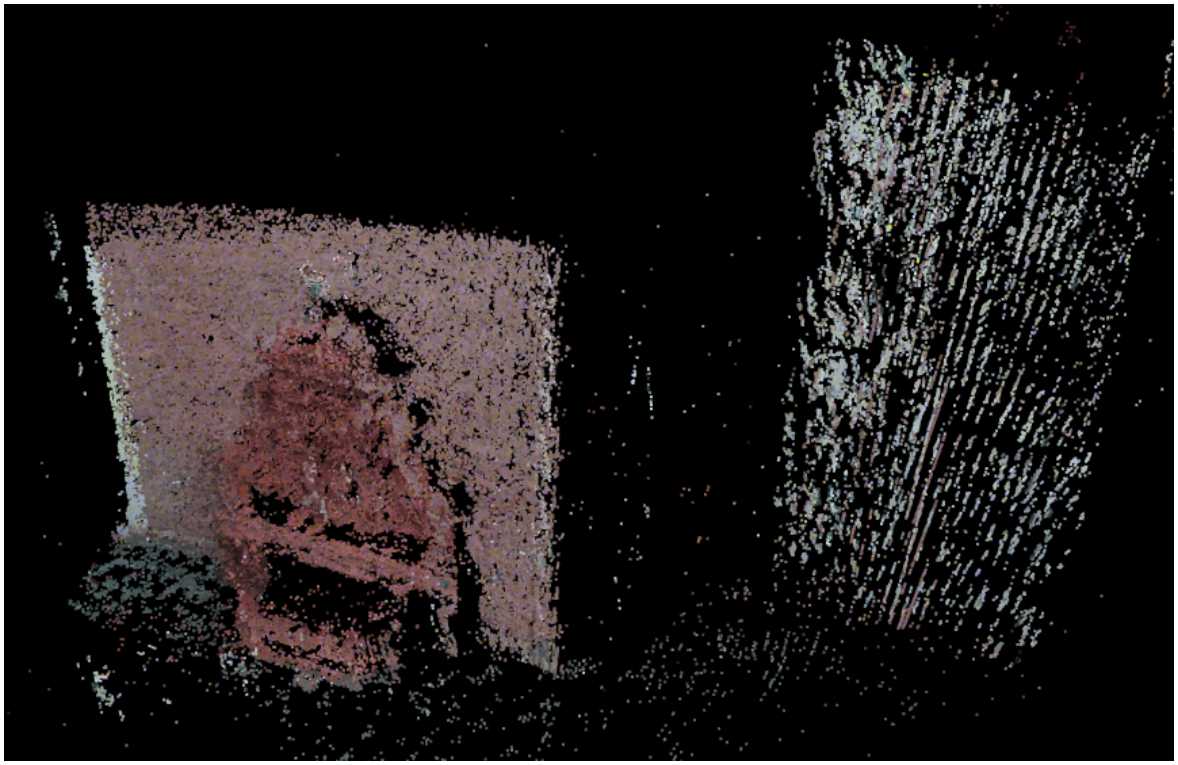


Figura 5.36: Modelo tridimensional considerando como *baseline* las imágenes $(16, 17)$ del cojunto *Fountain-R25*

5.5.3 Propuesta de solución completa

Con las pruebas anteriores se estimó que el tiempo de procesamiento para construir una nube de puntos tridimensionales de aproximadamente 33915.75 puntos a partir de un par de imágenes es de 6.71 segundos. Este tiempo, que puede parecer muy grande, debe situarse en el contexto de que corresponde a dos imágenes tomadas desde puntos distintos, por lo que se requiere mover al **VANT** a la nueva posición y estabilizar su vuelo, antes de tomar la siguiente fotografía. Con esto en mente, el tiempo de 6.71 segundos resulta totalmente compatible con el tipo de aplicación en tiempo real que estamos proponiendo. Así mismo, se mencionó la importancia del *baseline* para la calidad del modelo final.

Para validar la solución propuesta directamente sobre el **VANT**, se realizaron pruebas en el interior de las instalaciones del Cinvestav-Tamaulipas, para reducir la inestabilidad del **VANT** que pueda producir el viento. Se construyó una estructura de forma que simule una fachada con elementos visuales característicos, la cual se muestra en la Figura 5.37. Puede apreciarse igualmente el marcador artificial colocado al centro de la "fachada".

(a) (b)

(c)

Figura 5.37: "Fachada"para probar la propuesta de solución

Se siguió el proceso indicado en la Sección 4.1. El **VANT** se ubica frente a la fachada de forma que el marcador sea visible, a una distancia no mayor a $14n$ ó 2.80m, como se determinó en la Sección 5.3. Se limitó a 5 segundos el tiempo máximo que tiene el **VANT** para ubicarse frente al marcador y la fachada.

Para tener un *baseline* amplio y obtener un modelo útil, las primeras dos imágenes del **VANT** son adquiridas desde la misma posición, y evitar así los cambios de escala. Se considera una trayectoria para el **VANT** similar a la propuesta en la Figura 4.2, considerando un solo marcador, mostrada en la Figura 5.38. En esta trayectoria se consideran tres planos de captura (x, y) a distintas altitudes de vuelo: el *Plano 0* se encuentra la altura z del suelo y al nivel del marcador, el *Plano 1* se encuentra a una altura $H = 40\ cm$ por encima del *Plano 0* y finalmente el *Plano -1* se encuentra a una altura $h = \frac{z}{2}$, entre el *Plano 0* y el piso.

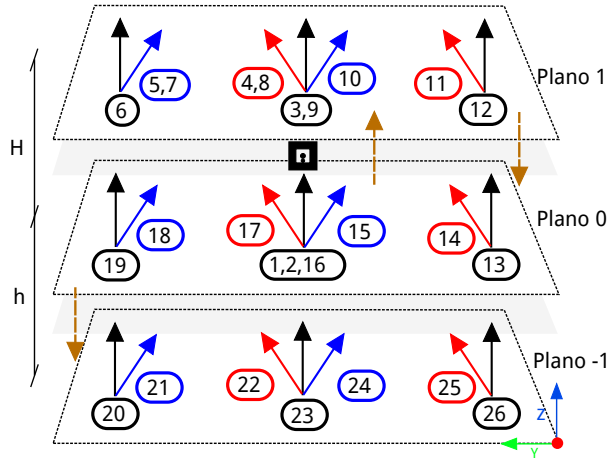
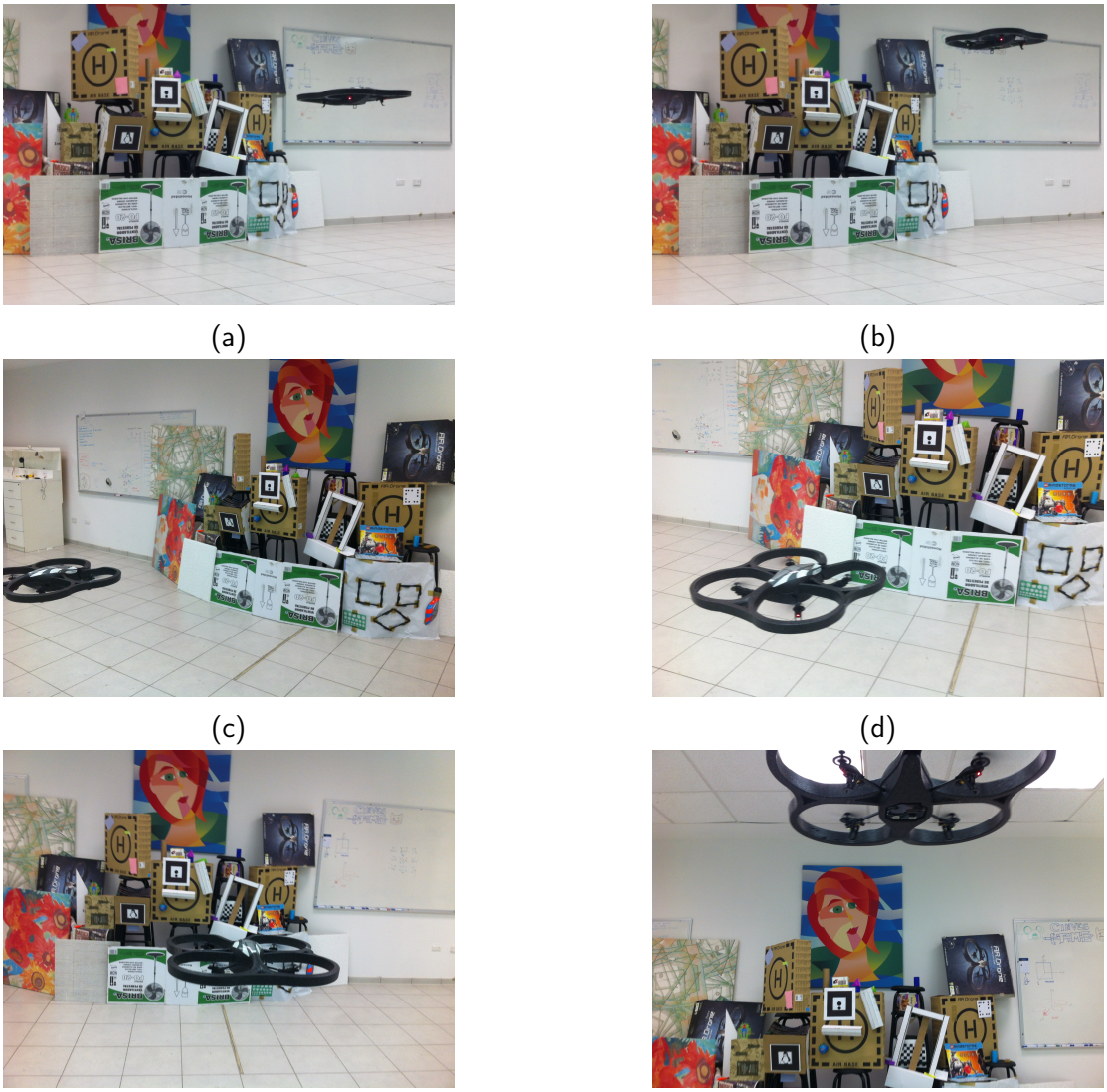


Figura 5.38: Trayectoria *a priori* usada para la adquisición de fotografías por medio del **VANT**

Se consideraron 26 poses distintas para la adquisición de fotografías, desde nueve puntos diferentes en los cuales se varían los ángulos de guiñada, como fue mostrado en la Figura 4.3. Estas posiciones $(-\frac{\pi}{4}, 0, \frac{\pi}{4})$ son las indicadas en la Figura 5.38. Se numeraron según la trayectoria seguida por el **VANT**, de forma que las imágenes comparten gran cantidad de características comunes que pueden ser emparejadas. La adquisición de fotografías repetidas para algunas poses y posiciones permite que se encuentren nuevas características y se obtenga una nube más densa.

En la Figura 5.39 se muestra al **VANT** durante el seguimiento de trayectoria y adquisición

de fotografías, frente a la fachada de prueba. El seguimiento de esta trayectoria es totalmente automático, sin intervención del usuario, por medio de las instrucciones modificadas para el control de vuelo y la estimación de posición por odometría inercial y visual, descritas en el capítulo anterior.



(a) (b) (c) (d) (e) (f)

Figura 5.39: **VANT** en vuelo realizando la trayectoria para adquisición de imágenes

Un subconjuto procedente de las 26 fotografías obtenidas por el **VANT** se muestra en la Figura 5.40. A partir de estas imágenes, se obtiene el modelo tridimensional representado por la nube de

puntos que se muestra en las Figuras 5.41 y 5.42.

(a) Imagen 2 (b) Imagen 3 (c) Imagen 5

(d) Imagen 8 (e) Imagen 12 (f) Imagen 16

(g) Imagen 19 (h) Imagen 20 (i) Imagen 24

Figura 5.40: Subconjunto de imágenes obtenidas por el **VANT** de la fachada de prueba

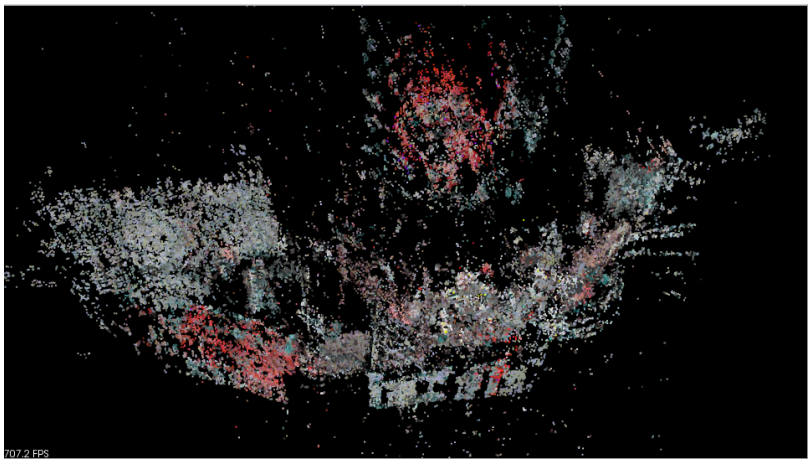


Figura 5.41: Nube de puntos construida con el subconjuto de imágenes de la Figura 5.40

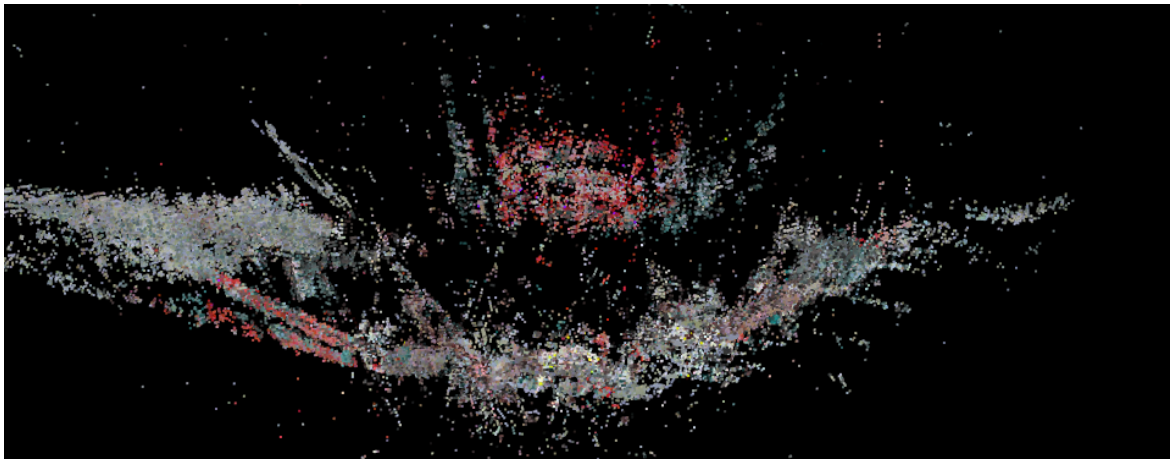


Figura 5.42: Vista aérea de la nube de puntos mostrada en la Figura 5.41

Las medianas de los tiempos obtenidos para las etapas que permiten la generación del modelo se presentan en la Tabla 5.11, estos corresponden al procesamiento de las nueve imágenes en 40 pruebas. En la Figura 5.43 se muestra la gráfica de caja correspondiente a los tiempos, donde se aprecia la variación entre las imágenes.

Detección	Descripción	Emparejamiento	Pose	Triangulación
3.06×10^{-3}	1.18×10^{-1}	7.43×10^{-1}	3.97×10^{-1}	1.54×10^{-1}
Tiempo total			1.414segundos	

Tabla 5.11: Tiempo por etapas para la construcción del modelo tridimensional de la fachada de prueba

El valor de la mediana para el error de reproyección fue de 4.456 cm considerando 6087.25 puntos tridimensionales por cada par de imágenes y un tiempo de 1.414 segundos.

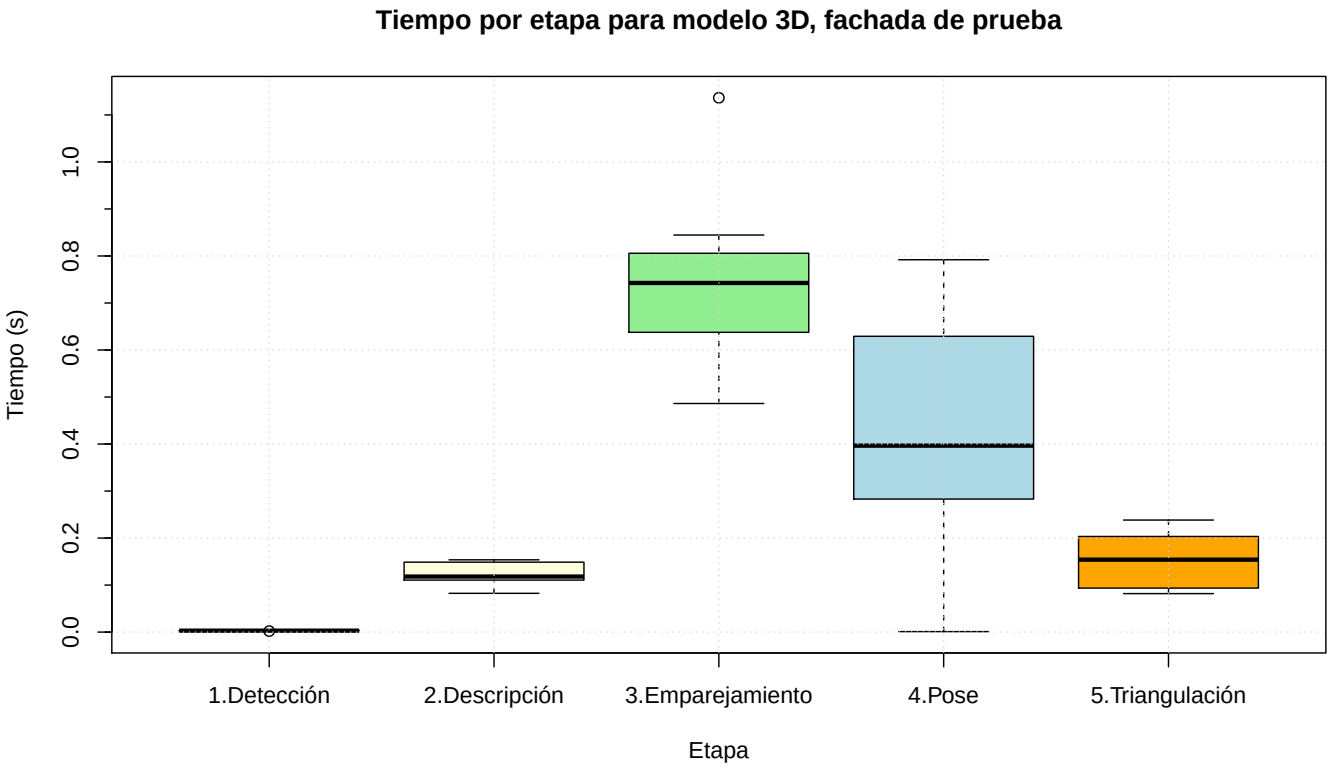


Figura 5.43: Tiempo por etapas para la fachada de prueba

Considerando las características de las imágenes seleccionadas con las cuales se generó el modelo

presentado en las Figuras 5.41 y 5.42 se adquirieron con el *VANT* solamente nueve imágenes de la forma como se presenta en la Figura 5.44. Las posiciones consideran giros de $15°$ en el ángulo de guiñada para las fotografías de los extremos, la altura de cada plano es semejante a la trayectoria anterior. El conjunto de fotografías se muestra en la Figura 5.48. Se aprecian capas en el modelo tridimensional, correspondientes a diferencias en la triangulación de los distintos pares de imágenes.

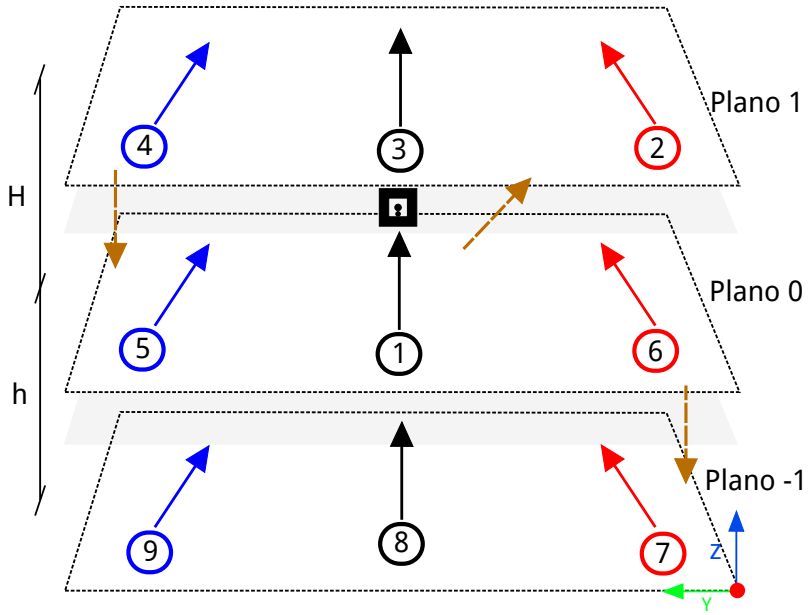


Figura 5.44: Trayectoria de solo nueve imágenes para la fachada de prueba

El modelo generado con este conjunto de nueve fotografías se muestra en las Figuras 5.45, 5.46 y 5.47.

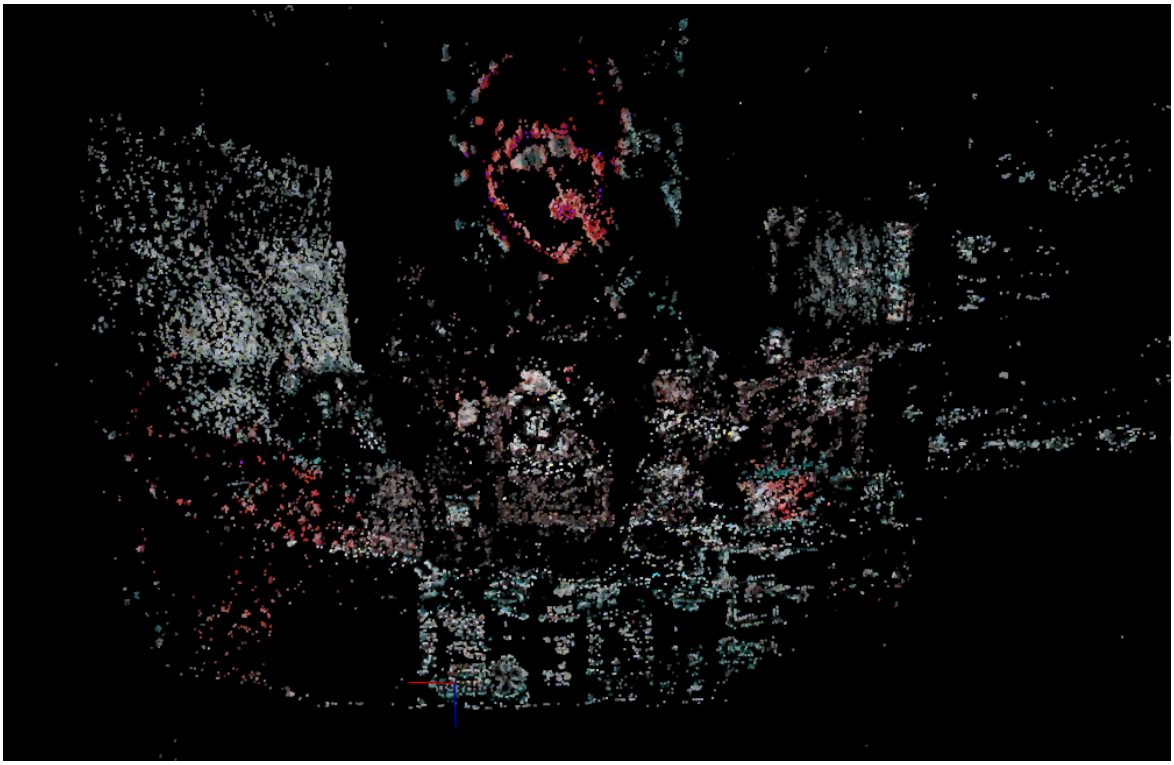


Figura 5.45: Nube de puntos para la trayectoria de nueve imágenes de la fachada de prueba

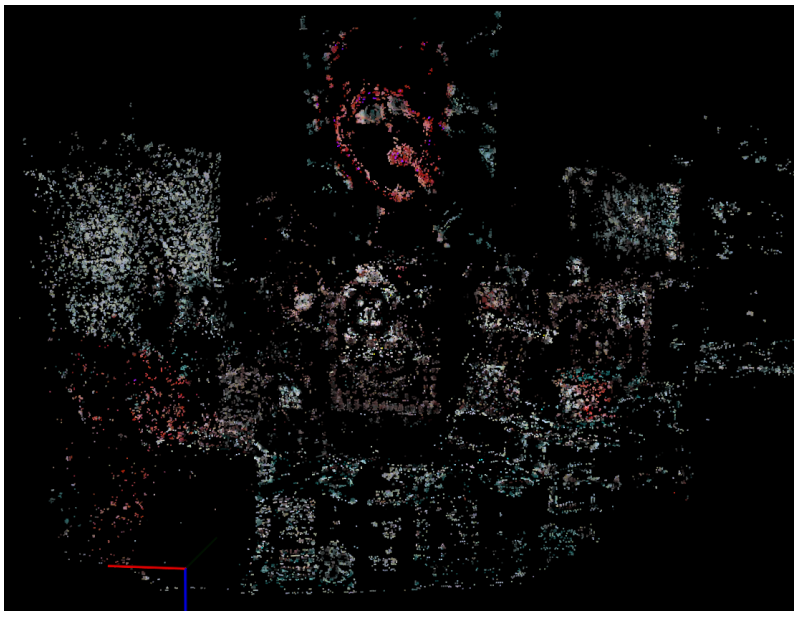


Figura 5.46: Vista del modelo de la Figura 5.45

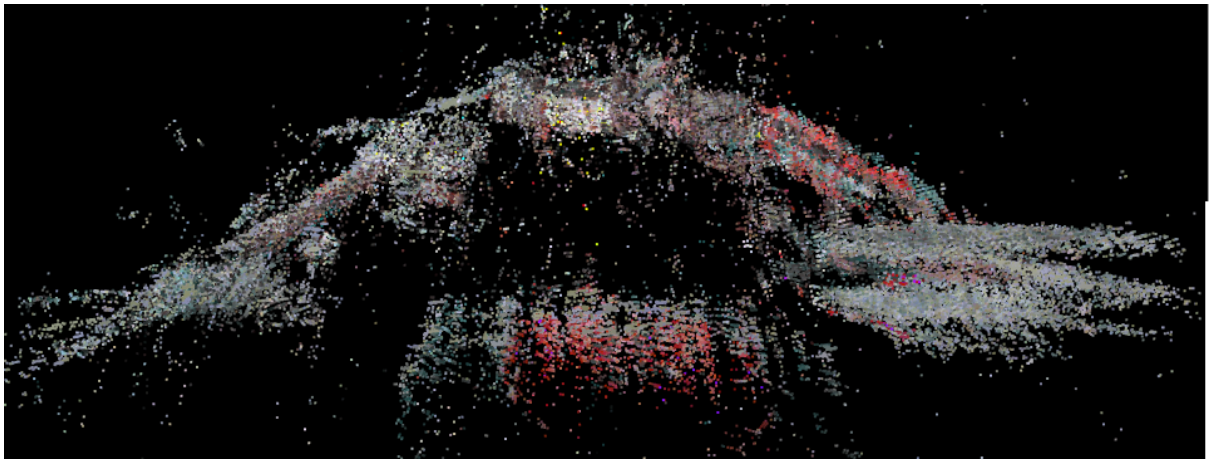
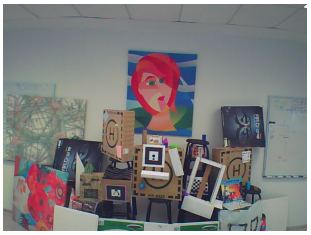
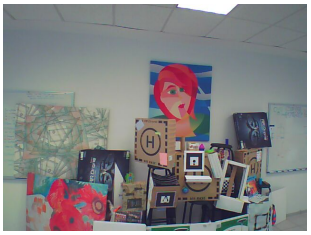


Figura 5.47: Vista aérea de la nube de puntos mostrada en la Figura 5.45

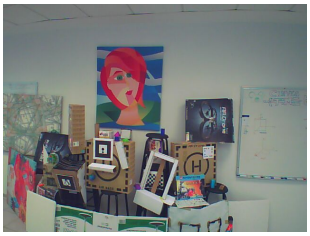
(a) Imagen 1



(b) Imagen 2



(c) Imagen 3



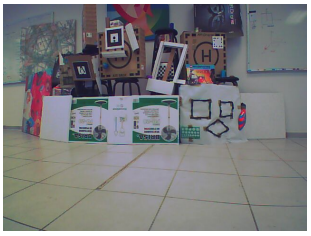
(d) Imagen 4

(e) Imagen 5

(f) Imagen 6

(g) Imagen 7

(h) Imagen 8



(i) Imagen 9

Figura 5.48: Conjunto de imágenes obtenidas por el **VANT** para la trayectoria mostrada en la Figura 5.44

5.6 Conclusiones

En este capítulo se describieron las pruebas realizadas y los resultados obtenidos para la validación de cada una de las etapas que conforman la propuesta de solución. Se presentó también el resultado para el modelo tridimensional obtenido. Los experimentos se enfocaron principalmente en medir el desempeño de los algoritmos, empleando la métrica de correspondencias correctas por unidad de tiempo obtenidas por las distintas combinaciones de algoritmos de detección-descripción-emparejamiento-triangulación de puntos. Los resultados obtenidos mostraron la factibilidad de las soluciones propuestas para los diferentes problemas que se trataron. De cada una de las etapas podemos concluir:

- Control autónomo del **VANT**. Las instrucciones de control que fueron probadas con odometría inercial requieren de una mejor calibración que reduzca el error entre el desplazamiento real, el deseado y el estimado por odometría mediante los sensores del **VANT**. Estas instrucciones fueron suficientes para desplazamientos laterales y giros de poca precisión. Considerando desplazamientos de baja velocidad y un menor error de odometría, se puede implementar algún sistema de control como un sistema *PID*[1] o lógica difusa, por mencionar algunos, para desarrollar un control automático preciso de vuelo.

- Modulo de odometría. El sistema de estimación de posición empleando solamente los sensores del **VANT** requiere calibración de los intervalos de muestreo para obtener una mejor resolución.

- Detección del marcador: La detección de marcadores depende de la resolución de la cámara así como de la calidad de las imágenes y la iluminación de la escena. Al aumentar la escala de las imágenes obtenidas por el **VANT** (320×240 pixeles), se inducen errores en cuanto a repetibilidad para estimar la pose de la cámara por medio de marcadores; sin embargo, la calidad de la postura estimada mostró ser muy próxima a la obtenida a partir de los datos

[1] *Proporcional-Integral-Derivativo*

reales medidos manualmente durante las pruebas.

- Emparejamiento de imágenes: Se determinó la mejor combinación de algoritmos disponibles en el estado del arte para procesamiento de imágenes, con el objetivo de obtener un algoritmo integral compatible con el procesamiento de imágenes en línea que queríamos desarrollar. Se concluye entonces que para detección de puntos de interés en la imagen debe utilizarse el algoritmo **FAST**, para descripción de los puntos detectados el algoritmo **BRISK** y como algoritmo de emparejamiento, vecinos más cercanos por tablas *hash*.

- Generación del modelo tridimensional: El cálculo de pose, la calibración de la cámara y reducir la distorsión de las imágenes son fundamentales la correcta generación de un modelo tridimensional. Para el cálculo de la pose en línea se propone utilizar la información visual para corregir la incertidumbre del sistema de odometría. Con la estimación adecuada de la posición desde la que se tomó cada fotografía, es posible realizar una correcta triangulación de todos los puntos detectados, de manera que el error de reproyección sea lo más pequeño posible.

- Propuesta completa: Los tiempos de procesamiento para la construcción del modelo tridimensional son apropiados para su funcionamiento en línea, es decir, durante el vuelo del **VANT**. Las pruebas se realizaron considerando un solo marcador y una fachada de prueba al interior de un edificio; sin embargo, con una fachada real, como en el conjunto de prueba *Fountain-R25*, donde la diferencia de profundidades es mayor entre los distintos elementos de la fachada, se puede apreciar que el modelo construido es visualmente más atractivo y refleja de manera fiel la estructura arquitectónica del edificio fotografiado.

6

Conclusiones y trabajo futuro

En este trabajo de tesis se propuso una solución para la construcción de un modelo tridimensional de la fachada de un edificio en línea, mediante imágenes monoculares obtenidas por un **VANT** de forma autónoma. Se basó la propuesta de solución en la metodología de extracción de características tridimensionales para cámaras monoculares y robots móviles terrestres. Si bien ya se han empleado con anterioridad los **VANT** para la adquisición de fotografías de una edificación, el proceso de reconstrucción tridimensional se efectúa en una etapa posterior, fuera de línea, analizar dichas imágenes para construir un modelo tridimensional.

La propuesta de solución permite la construcción de un modelo tridimensional incremental de la fachada de la edificación mientras que el **VANT** se encuentra en vuelo, validando la hipótesis de trabajo: *Considerando un **VANT** tipo **VTOL** con tiempo de vuelo limitado que adquiere fotografías monoculares de una fachada, es posible desarrollar un procesamiento de imágenes rápido y ligero que permita la construcción de un modelo tridimensional burdo del edificio, en línea e incremental,*

para detectar las zonas de las cuales no existe suficiente información para construir un modelo con un mayor nivel de detalle, es decir, zonas que requieren ser fotografiadas nuevamente.

El éxito de la construcción del modelo tridimensional incremental en línea consiste en su capacidad para procesar las imágenes, determinar su relación, y triangular la posición de puntos de interés en tiempos de cómputo compatibles con el tiempo de vuelo del **VANT**.

La detección de marcadores visuales permiten ubicar al **VANT** dentro de la escena, estableciendo el origen del modelo y determinando la posición del vehículo con respecto a la fachada. Los marcadores, además, complementan la odometría inercial de la plataforma aérea, eliminando la incertidumbre y los errores propios del cálculo de posición por odometría.

Se propuso también un sistema de control para el **VANT**, basado en el control de fábrica, para realizar los desplazamientos en las trayectorias deseadas y eefctuar la adquisición de fotografías de la fachada de forma autónoma.

6.1 Componentes de la solución propuesta

Sobre los componentes de la solución propuesta

- **Control de desplazamiento del *VANT* y Estimación de posición del *VANT*.**

  Este componente envía instrucciones vía inalámbrica al **VANT** para que realice algún desplazamiento o movimiento, además de recibir la información de los sensores de velocidad. Se realizaron instrucciones específicas para desplazamientos fijos de 40cm, 1m, 1.20m, 2m sobre los ejes (x, y) y giro en la guiñada (*yaw*) de 15, 30 y 90 grados. Las instrucciones de control para desplazamientos específicos fueron suficientes para ubicar al **VANT** frente a la

fachada, así como realizar trayectorias conocidas *a priori*.

- **Seguimiento de trayectorias del VANT**. El desplazamiento siguiendo una trayectoria *a priori* se realizó enviando en forma ordenada al **VANT** las instrucciones de control, por lo que el éxito de este componente depende de las instrucciones de movimientos. Se comprobó mediante experimentación los desplazamientos junto con la odometría para estimar la posición del **VANT**. Para una trayectoria de validación mostrada en la Figura 5.17 presentada por Borenstein *et al.* en [6] se obtuvo un error sistemático de 0.41 metros, tras un recorrido de 8 m.

- **Detección visual de un marcador artificial.** Este componente ubica marcadores visuales artificiales, como el mostrado en la Figura 1.6, en la escena. Con una detección correcta del marcador es posible determinar la pose del **VANT** respecto a la fachada, además que es posible eliminar la incertidumbre que se obtiene al estimar la posición por odometría. El análisis de la imagen depende de la iluminación de la escena, que el marcador sea visible y libre de obstrucciones, así como de la calidad de las imágenes. Con la cámara frontal del **VANT** utilizado para la experimentación, con imágenes de 960×720 pixeles, se determinó que la distancia máxima para considerar confiable una detección para un marcador de $n \times n$ ($n = 20$ cm) es de $6n$, y un ángulo de visibilidad máximo de $\pm 40°$, medidos desde la perpendicular al marcador. Con la pose estimada mediante la detección del marcador fue posible determinar los movimientos necesarios para que el **VANT** se ubique de forma automática frente al marcador.

- **Reconstrucción de un modelo tridimensional burdo en línea.** Este componente recibe las imágenes obtenidas por el **VANT** de forma secuencial, encuentra características en común y, después de estimar la pose de la cámara, calcula la posición tridimensional de dichas características. El resultado de este componente es una nube de puntos que corresponde a la fachada de la edificación. En un proceso independiente fue necesario la calibración de la

cámara, siendo parte indispensable para la correcta construcción del modelo. Al considerar que el modelo se construye de forma incremental en línea, mediante experimentación se evaluaron las siguientes características:

- Para la detección de puntos característicos se evaluaron los algoritmos **FAST** y **BRISK**, ambos con un umbral de 5 sin supresión de no máximos. En el caso de **BRISK** se consideraron cuatro octavas, como recomiendan sus autores Leutenegger *et. al* en el artículo [27].
- La descripción de puntos característicos mediante los algoritmos **BRIEF** y **BRISK**, ambos con una longitud de 64 *bytes* para el vector de descripción.
- Emparejamiento mediante el método exhaustivo o cuadrático y el método de vecinos más cercanos con tablas *hash* considerando cuatro tablas, ruido de búsqueda y elementos de longitud 20.
- Con el conjunto de datos *Fountain-R25* se evaluaron dichos algoritmos, finalmente comparando las correspondencias correctas contra el tiempo de procesamiento. El grupo de algoritmo que, en conjunto, mejor cumple esta métrica fueron **FAST** para detección, **BRISK** como descriptor y emparejamiento de vecinos más cercanos, con 9542.87 parejas correctas por segundo.
- Para el conjunto de 25 imágenes *Fountain-R25* se procesa una nube de 33915.75 puntos tridimensionales, en 6.71 segundos, por cada par de imágenes. Del conjunto de nueve fotografías realizadas con el **VANT** de la fachada de prueba se obtuvieron 6087.25 puntos tridimensionales, en 1.414 segundos, para cada par de imágenes.
- Aumentando el umbral del detector de puntos es posible reducir la cantidad de puntos característicos y por lo tanto la densidad de la nube de puntos, siendo estas propiedades inversamente proporcionales al tiempo de procesamiento.

- **Visualización del modelo burdo.** El visualizador del modelo de nube de puntos tridimensional es un proceso independiente a la construcción del modelo, el procesamiento de imágenes y control autónomo del **VANT**. Este componente se actualiza para cada sub-nube de puntos obtenida de un par de imágenes, mostrando tanto las características cuantitativas, los puntos estimados en el espacio tridimensional, como características cualitativas, correspondiente al color. Este modelo es explorable, interactivo, pudiendo ajustar la cámara e incluso el tamaño de los puntos tridimensionales mostrados.

La propuesta de solución es aplicable a cualquier **VANT** con comunicación inalámbrica para control a distancia, cámara frontal y transmisión de imágenes en línea.

6.2 Trabajo futuro

El trabajo desarrollado es muy prometedor. Ofrece diversos puntos que pueden ser mejorados:

- El **VANT** *AR.Drone* de la marca *Parrot* utilizado para la experimentación es de bajo costo, las características de estabilidad en vuelo así como la baja calidad de imágenes son aceptables. Empleando un **VANT** con mejores prestaciones de estabilidad y calidad de fotografía los resultados obtenidos mejorarían.

- Las instrucciones de control del **VANT** fueron calibradas manualmente, siendo una gran área de mejora el implementar algún control como el Proporcional-Integral-Derivativo o lógica difusa, por mencionar algunos, que se apoyen en la odometría para realizar los movimientos de forma precisa.

- Los datos de odometría son utilizados .^{en} crudo", como se adquieren. Para mejorar la estimación de posición se recomendaría el uso de algún filtro predictivo, como el filtro de Kalman. Con

la estimación de pose obtenida con el marcador al marco visual artificial permitirían acoplar el marco de referencia de odometría al del modelo tridimensional.

- El componente de detección de marcadores puede ser entrenado para detectar e identificar más de un marcador en la escena. Emplear varios marcadores permiten la navegación de forma autónoma del **VANT** en la escena. Incluso el uso de diversos marcadores podrían ser aprovechados para organizar a más de un **VANT**, volando simultáneamente, de modo que se pueda abarcar una mayor área de la fachada para un mismo modelo, de forma que la construcción de un modelo tridimensional más grande tenga un enfoque colaborativo.

- Considerando las funciones de procesamiento de imagen optimizadas para *GPU*s, sería posible reducir los tiempos de procesamiento y obtener una nube de puntos densa.

- El modelo tridimensional construido corresponde a los puntos característicos proyectados en el espacio tridimensional. Sin embargo, es posible agregar un nuevo módulo de procesamiento que mejore la calidad del modelo construido, como por ejemplo el algoritmo de crecimiento de parches presentado por Furukawa *et. al* en el artículo [16] o la estimación de planos tridimensionales con textura como el empleado por Wnuk *et. al* en el artículo [64].

- La biblioteca empleada para el componente de visualización del modelo tridimensional permite la transmisión de la nube de puntos a un dispositivo móvil. Esto abre la posibilidad a aplicar la propuesta de solución en una computadora sin contar con un monitor, que realice la comunicación con el **VANT** y el procesamiento de imágenes, y transmita el modelo generado a dispositivos móviles. En el escenario anterior se mejora la movilidad del Sistema Aéreo No Tripulado (**SANT**) para la implementación de la propuesta de solución.

Bibliografía

[1] Alahi, A., Ortiz, R., and Vandergheynst, P. (2012). FREAK: Fast Retina Keypoint. In *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE.

[2] Barazzetti, L., Scaioni, M., and Remondino, F. (2010). Orientation and 3d modelling from markerless terrestrial images: combining accuracy with automation. *The Photogrammetric Record*, 25(132):356–381.

[3] Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. (2006). SURF: Speeded Up Robust Features. In *European Conference on Computer Vision*, pages 404–417.

[4] Beis, J. S. and Lowe, D. G. (1997). Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, CVPR '97, pages 1000–, Washington, DC, USA. IEEE Computer Society.

[5] Blaer, P. and Allen, P. (2007). Data acquisition and view planning for 3-D modeling tasks. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 417–422.

[6] Borenstein, J. and Feng, L. (1995). UMBmark: A Benchmark Test for Measuring Odometry Errors in Mobile Robots. In *SPIE Conference on Mobile Robots*.

[7] Bouguet, J.-Y. (2013). Camera Calibration Toolbox for Matlab. `http://www.vision.caltech.edu/bouguetj/calib_doc/`. [Online; accessed 21-XI-2013].

[8] Calonder, M., Lepetit, V., Strecha, C., and Fua, P. (2010). BRIEF: Binary Robust Independent Elementary Features. In *European Conference on Computer Vision*.

[9] Colomina, I., Blázquez, M., Molina, P., Parés, M., and Wis, M. (2008). Towards a new paradigm for high-resolution low-cost photogrammetry and remote sensing. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXVII Part B1:1201–1206.

[10] Debevec, P. E., Taylor, C. J., and Malik, J. (1996). Modeling and rendering architecture from photographs: a hybrid geometry- and image-based approach. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, SIGGRAPH '96, pages 11–20, New York, NY, USA. ACM.

[11] Dempsey, M. E. (2010). Eyes of the Army U.S. Army Roadmap for Unmanned Aircraft Systems 2010-2035. United States Army.

[12] Diskin, Y. and Asari, V. K. (2012). Dense point-cloud creation using superresolution for a monocular 3D reconstruction system. *Proc. SPIE 8399, Visual Information Processing XXI, 83990N (May 1, 2012)*, pages 83990N–83990N–9.

[13] Douglas, D. H. and Peucker, T. K. (2011). *Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or its Caricature*, pages 15–28. John Wiley & Sons, Ltd.

[14] Everaerts, J. (2008). The Use of Unmanned Aerial Vehicles (UAVS) for Remote Sensing and Mapping. *IAPRS&SIS*, 37(B1), Beijing, China:1187–1192.

[15] Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395.

[16] Furukawa, Y. and Ponce, J. (2008). Accurate camera calibration from multi-view stereo and bundle adjustment. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8.

[17] Gleason, J., Nefian, A., Bouyssounousse, X., Fong, T., and Bebis, G. (2011). Vehicle detection from aerial imagery. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 2065 –2070.

[18] Harris, C. and Stephens, M. (1988). A combined corner and edge detector. In *Proceedings of the 4th Alvey Vision Conference*, pages 147–151.

[19] Hartley, R. and Sturm, P. (1995). Triangulation. In *Computer Analysis of Images and Patterns*, volume 970 of *Lecture Notes in Computer Science*, pages 190–197. Springer Berlin Heidelberg.

[20] Hartley, R. I. (1997). In defense of the eight-point algorithm. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(6):580–593.

[21] Horn, B. K. P. and Schunck, B. G. (1981). Determining Optical Flow. *Artificial Intelligence*, 17:185–203.

[22] Irschara, A., Kaufmann, V., Klopschitz, M., Bischof, H., and Leberl, F. (2010). Towards fully automatic photogrammetry reconstruction using digital images taken from UAVs. In *ISPRS TC VII Symposium - 100 Years ISPRS*, volume Vol. XXXVIII Part 7A.

[23] Keys, R. (1981). Cubic convolution interpolation for digital image processing. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 29(6):1153–1160.

[24] Krajnik, T., Nitsche, M., Pedre, S., Preucil, L., and Mejail, M. (2012). A simple visual navigation system for an UAV. In *Systems, Signals and Devices (SSD), 2012 9th International Multi-Conference on*, pages 1–6.

[25] Küng, O., Strecha, C., Fua, P., Gurdan, D., Achtelik, M., Doth, K.-M., and Jan, S. (2011). Simplified building models extraction from ultra-light UAV imagery. *International Conference on Unmanned Aerial Vehicle in Geomatics (UAV-g)*, XXXVIII/C22:6.

[26] Lamberti, F., Sanna, A., Paravati, G., Montuschi, P., Gatteschi, V., and Demartini, C. (2013). Mixed Marker-Based/Marker-Less Visual Odometry System for Mobile Robots. *International Journal of Advance Robotic System*, 10:260.

[27] Leutenegger, S., Chli, M., and Siegwart, R. (2011). Brisk: Binary robust invariant scalable keypoints. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2548–2555.

[28] Lim, H. and Lee, Y.-S. (2009). Real-time single camera SLAM using fiducial markers. In *ICCAS-SICE, 2009*, pages 177–182.

[29] Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2*, ICCV '99, pages 1150–, Washington, DC, USA. IEEE Computer Society.

[30] Lv, Q., Josephson, W., Wang, Z., Charikar, M., and Li, K. (2007). Multi-probe lsh: efficient indexing for high-dimensional similarity search. In *Proceedings of the 33rd international conference on Very large data bases*, VLDB '07, pages 950–961. VLDB Endowment.

[31] Makynen, J., Saari, H., Holmlund, C., Mannila, R., and Antila, T. (2012). Multi and hyperspectral UAV imaging system for forest and agriculture applications. In *Proceedings of the SPIE - The International Society for Optical Engineering*, volume 8374, page 837409 (9 pp.). Next-Generation Spectroscopic Technologies V, 23-24 April 2012, Baltimore, MD, USA.

[32] MATTERNET (2012). Transportation using a network of UAVs. `http://matternet.us`. [Online; accessed 30-X-2012].

[33] McCurdy, P., editor (1944). *Manual of Photogrammetry*. Pitman Publishing Co.

[34] Müller, P., Wonka, P., Haegler, S., Ulmer, A., and Van Gool, L. (2006). Procedural modeling of buildings. *ACM Trans. Graph.*, 25(3):614–623.

[35] Müller, P., Zeng, G., Wonka, P., and Van Gool, L. (2007). Image-based procedural modeling of facades. *ACM Trans. Graph.*, 26(3).

[36] National Oceanic and Atmospheric Administration (2008). NOAA invests $3 million for Unmanned Aircraft System Testing. `http://www.noaanews.noaa.gov/stories2008/20080122_aircraft.html`. [Online; accessed 30-X-2012].

[37] Niranjan, S., Gupta, G., Sharma, N., Mangal, M., and Singh, V. (2007). Initial efforts toward mission-specific imaging surveys from aerial exploring platforms: UAV. In *Map World Forum, Hyderabad, India*.

[38] Nüchter, A., Lingemann, K., and Hertzberg, J. (2006). Extracting drivable surfaces in outdoor 6D SLAM. In *In the 37nd Symp. on Robotics (ISR06), Munich*.

[39] Pollefeys, M., Van Gool, L., Vergauwen, M., Verbiest, F., Cornelis, K., Tops, J., and Koch, R. (2004). Visual modeling with a hand-held camera. *Int. J. Comput. Vision*, 59(3):207–232.

[40] Polski, P. (2004). DHS View of Unmanned Aerial Vehicle Needs. *Proceedings of AIAA 3rd Unmanned Unlimited Technical Conference, Chical, IL, USA*, September.

[41] Pop, G., Bucksch, A., and Gorte, B. (2007). 3D buildings modelling based on a combination of techniques and methodologies. In *XXI CIPA Symposium - Athens, GREECE - 1 October - 6 October 2007 Proceedings*.

[42] Püschel, H., Sauerbier, M., and Eisenbeiss, H. (2008). A 3D model of Castle Landemberg (CH) from combined photogrammetric processing of terrestrial and UAV-based images. *IAPRS&SIS*, 37(B6), Beijing, China:96–98.

[43] Rachmielowski, A., Birkbeck, N., Jagersand, M., and Cobzas, D. (2008). Realtime visualization of monocular data for 3D reconstruction. In *Computer and Robot Vision, 2008. CRV08. Canadian Conference on*, pages 196–202.

[44] Rathinam, S., Kim, Z., Soghikian, A., and Sengupta, R. (2005). Vision based following of locally linear structures using an unmanned aerial vehicle. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05. 44th IEEE Conference on*, pages 6085 – 6090.

[45] Remondino, F., Barazzetti, L., Nex, F., Scaioni, M., and Sarazzi, D. (2011). UAV Photogrammetry for mapping and 3D modelling -current status and future perspectives-. In *Int. Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences V. XXXVIII*.

[46] Remondino, F. and El-Hakim, S. (2006). Image-based 3D modeling: A review. *The Photogrammetric Record Journal*, 21(115):269–291.

[47] Rizwan, Y., Waslander, S., and Nielsen, C. (2011). Nonlinear aircraft modeling and controller design for target tracking. In *American Control Conference (ACC), 2011*, pages 3191 –3196.

[48] Rosten, E. and Drummond, T. (2006). Machine learning for high-speed corner detection. In *Proceedings of the 9th European conference on Computer Vision - Volume Part I*, ECCV'06, pages 430–443, Berlin, Heidelberg. Springer-Verlag.

[49] Ryan, A. and Hedrick, J. (2005). A mode-switching path planner for UAV-assisted search and rescue. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05. 44th IEEE Conference on*, pages 1471 – 1476.

[50] Sauerbier, M. and Eisenbeiss, H. (2010). UAVS for the documentation of archaeological excavations. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVIII Part 5.*

[51] Scaioni, M., Barazzetti, L., R., B., Cuca, B., Fassi, F., and Prandi, F. (2009). RC-Heli and structure & motion techniques for the 3-D reconstruction of a Milan Dome spire. *IAP*, 38(5/W1), Trento, Italy.

[52] Schmidt, A., Kraft, M., Fularz, M., and Domagala, Z. (2012). The comparison of point feature detectors and descriptors in the context of robot navigation. In *Workshop on Perception for Mobile Robots Autonomy*.

[53] Shen, C.-H., Huang, S.-S., Fu, H., and Hu, S.-M. (2011). Adaptive partitioning of urban facades. *ACM Trans. Graph.*, 30(6):184:1–184:10.

[54] Smith, R. C. and Cheeseman, P. (1986). On the Representation and Estimation of Spatial Uncertainly. *International Journal of Robotics Research*, 5(4):56–68.

[55] Steffen, R. and Foerstner, W. (2008). On visual real time mapping for unmanned aerial vehicle. *IAPR*, 37(B3a):57–62.

[56] Strategic Defence Intelligence (2013). The Global UAV Market $2013 - 2023$. `http://www.prweb.com/releases/2013/11/prweb11307433.htm`. [Online; accessed 21-XI-2013].

[57] Strecha, C., Von Hansen, W., Van Gool, L., Fua, P., and Thoennessen, U. (2008). On benchmarking camera calibration and multi-view stereo for high resolution imagery. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8.

[58] Suzuki, K., Horiba, I., and Sugie, N. (2003). Linear-time connected-component labeling based on sequential local operations. *Comput. Vis. Image Underst.*, 89(1):1–23.

[59] Teal Group (2013). Teal Group Predicts Worldwide UAV Market Will Total $89 Billion in Its 2013 UAV Market. `http://tealgroup.com/index.php/about-teal-group-corporation/press-releases/94-2013-uav-press-release`. [Online; accessed 21-XI-2013].

[60] Thrun, S., Hahnel, D., Ferguson, D., Montemerlo, D., Triebel, R., Burgard, W., Baker, C., Omohundro, Z., Thayer, S., and Whittaker, W. (2003). A system for volumetric robotic mapping of abandoned mines. In *Robotics and Automation, 2003. Proceedings. ICRA03. IEEE International Conference on*, volume 3, pages 4270–4275 vol.3.

[61] Wang, J. and Li, C. (2007). Acquisition of UAV images and the application in 3D city modeling. *Proc. SPIE 6623, International Symposium on Photoelectronic Detection and Imaging 2007: Image Processing*, pages 66230Z–66230Z–11.

[62] Watts, A. C., Ambrosia, V. G., and Hinkley, E. A. (2012). Unmanned aircraft systems in remote sensing and scientific research: Classification and considerations of use. *Remote Sensing*, 4(6):1671–1692.

[63] Wefelscheid, C., Hänsch, R., and Hellwich, O. (2011). Three-Dimensional building reconstruction using images obtained by unmanned aerial vehicles. In *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVIII-1/C22 UAV-g 2011, Conference on Unmanned Aerial Vehicle in Geomatics, Zurich, Switzerland*.

[64] Wnuk, K., Dang, F., and Dodds, Z. (2004). Dense 3d mapping with monocular vision. Proceedings of the International Conference on Autonomous Robots and Agents.

[65] Xiao, J., Fang, T., Tan, P., Zhao, P., Ofek, E., and Quan, L. (2008). Image-based façade modeling. In *ACM SIGGRAPH Asia 2008 papers*, SIGGRAPH Asia '08, pages 161:1–161:10, New York, NY, USA. ACM.

[66] Yan, H.-P., Liu, C., Gagalowicz, A., and Guiard, C. (2009). Facade structure parameterization based on similarity detection from single image. In Gagalowicz, A. and Philips, W., editors, *Computer Vision/Computer Graphics CollaborationTechniques*, volume 5496 of *Lecture Notes in Computer Science*, pages 389–400. Springer Berlin / Heidelberg.

[67] Yan, L., Zhe, L., and Ying, S. (2012). The particularity of aerial photogrammetry for architectural heritages by UAV. In *Remote Sensing, Environment and Transportation Engineering (RSETE), 2012 2nd International Conference on*, pages 1 –4.

[68] Yanushevsky, R. (2011). *Guidance of Unmanned Aerial Vehicles*. Taylor & Francis.

[69] Zhang, Z. (1998). Determining the epipolar geometry and its uncertainty: A review. *Int. J. Comput. Vision*, 27(2):161–195.

[70] Zhang, Z. (2000). A flexible new technique for camera calibration. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(11):1330–1334.