Reconstrucción tridimensional a partir de una secuencia de imágenes

TESIS

Alonso A. Patrón Pérez

Maestría en Ciencias Matemáticas
Facultad de Matemáticas
Universidad Autónoma de Yucatán
Mérida, Yucatán, 2006

Resumen

Esta tesis presenta una manera para obtener reconstrucciones tridimensionales de escenas a partir de una secuencia de imágenes. El proceso que se describe a lo largo de este documento tiene un enfoque modular, tomando como guía la metodología descrita en [Pollefeys et al., 2004]. Gracias al enfoque modular se tuvo una gran flexibilidad al momento de experimentar con diversos métodos. Podemos resumir las fases del proceso de reconstrucción desarrollado en este trabajo en tres etapas principales: relacionar las imágenes, recuperar la estructura proyectiva de la escena y por último la obtención de la reconstrucción métrica. El objetivo principal de esta tesis es obtener un modelo de la escena formado por puntos tridimensionales. Para realizar esto se probaron varios métodos en cada una de las etapas ya mencionadas. Así mismo se impusieron algunas restricciones al movimiento de la cámara para simplificar algunos de los métodos. En esta implementación no se requiere conocer la calibración de la cámara de antemano, lo cual la hace práctica y amplía su campo de aplicación. Cada una de las fases se describe de manera independiente pero siempre manteniéndolas dentro del contexto del objetivo central. Los modelos tridimensionales resultantes tuvieron una muy buena calidad visual.

Agradecimientos

Primero quisiera agradecer a mi asesor, el Dr. Arturo Espinosa Romero, por su amistad y por su invaluable ayuda en el desarrollo de esta tesis. Al Dr. Ricardo Legarda Saenz por sus comentarios y recomendaciones que siempre considero valiosos. Un muy especial agradecimiento al Dr. Alberto Muñoz Ubando por animarme a entrar a este extraordinario mundo de la ciencia. A la Facultad de Matemáticas de la Universidad Autónoma de Yucatán por su apoyo y ayuda cuando la he necesitado.

Muchas gracias a mi familia por su comprensión y paciencia, lo cual ha sido muy importante para mí y me anima a seguir adelante. Gracias también a mis compañeros que han hecho de la maestría una muy grata experiencia durante los últimos dos años. Por último gracias a mis amigos y a todos aquellos que de alguna manera ayudaron con sus comentarios o recomendaciones a la elaboración de esta tesis.

Declaración

Esta tesis esta dirigida a la Facultad de Matemáticas de la Universidad Autónoma de Yucatán como cumplimiento a uno de los requerimientos para obtener el grado de Maestro en Ciencias Matemáticas. Por este medio declaro que esta tesis fue realizada enteramente por mi y describe mi propio trabajo de investigación con excepción de las partes que así se indique.

Alonso A. Patrón Pérez
Mérida, Yucatán
México
26 de Mayo de 2006

Índice general

Índice de figuras

Índice de cuadros

Capítulo 1

Introducción

El ser humano lleva ya mucho tiempo construyendo máquinas que sirvan para ayudarlo en sus tareas diarias o en trabajos que se consideran peligrosos (en ocasiones imposibles de realizar para el hombre). Dichas máquinas carecen de una característica importante con la que contamos los humanos: el sentido de la vista. Durante la última mitad del siglo XX, una gran cantidad de investigaciones se han orientado en esta dirección, es decir, tratar de proporcionarles esta capacidad a las máquinas, lo cual ayudaría a expandir de gran manera su ámbito de aplicación. En la actualidad existen algunas máquinas que pueden analizar cierta información del medio que los rodea mediante el uso de cámaras, como por ejemplo, los robots que verifican piezas en una línea de ensamblaje.

El proceso de la visión de máquina o visión por computadora, no fue tomado en serio por mucho tiempo y se creía, ingenuamente, un problema trivial que podía ser realizado como un proyecto de clase para estudiantes. Esto siguió así hasta que David Marr [Marr, 1982] realizó sus primeros experimentos en este campo y descubrió las dificultades que conlleva adaptar o sintetizar en un conjunto de procedimientos o algoritmos (que luego pudieran ser interpretados por una computadora) un proceso tan complicado como lo es el de la visión humana, el cual aún en la actualidad sigue sin ser comprendido en su totalidad.

El campo de la visión computacional ha tenido un gran auge en los últimos años e involucra varias ramas de la ciencia como son las matemáticas, física, computación y teoría de aprendizaje entre otras. Como se puede ver, la visión computacional cada vez se ve más relacionada con otras áreas de la ciencia y ha alcanzando un nivel de desarrollo aceptable en algunas de las partes que la componen. La visión computacional ha tenido grandes avances y logros tanto en la teoría como en la práctica, en especial podemos nombrar un campo llamado visión geométrica, que nos ayuda a describir como un objeto cambia cuando es observado desde diferentes puntos de vista. Aún después de todo este progreso quedan muchas cosas que mejorar.

1.1. Objetivo del Trabajo

La reconstrucción tridimensional es uno de los temas más importantes dentro de la visión computacional ya que su campo de aplicación es muy amplio. Consiste en la obtención de modelos tridimensionales de objetos o escenas del mundo real las cuales pueden ser utilizadas para navegación de robots, inspección visual y medición de objetos entre otras aplicaciones. Actualmente existen varios métodos para obtener modelos tridimensionales de los cuales muchos se han desarrollado en los últimos años. Entre estos métodos podemos contar a los sistemas de luz estructurada (Figura 1.1a), codificación de patrones, escáneres (Figura 1.1b), y reconstrucción a través de secuencias de vídeo o de imágenes, todos ellos tienen sus ventajas y desventajas dependiendo de la aplicación para la cual se requieran.

(a) (b)

Figura 1.1: Métodos de reconstrucción. (a) Sistemas de luz estructurada. (b) Escáneres láser.

El objetivo principal de este trabajo es lograr recuperar la información de profundidad de una escena determinada a partir de la información que se puede extraer de una secuencia de imágenes tomadas de dicha escena. El resultado final será un conjunto de puntos tridimensionales (también llamado *nube de puntos*) que diferirán de la escena real sólo por un factor de escala y un movimiento rígido (traslación y rotación). Como ejemplo de esto en la Figura 1.2a se muestra una imagen de un objeto (perteneciente a una secuencia) y en la Figura 1.2b la nube de puntos obtenida.

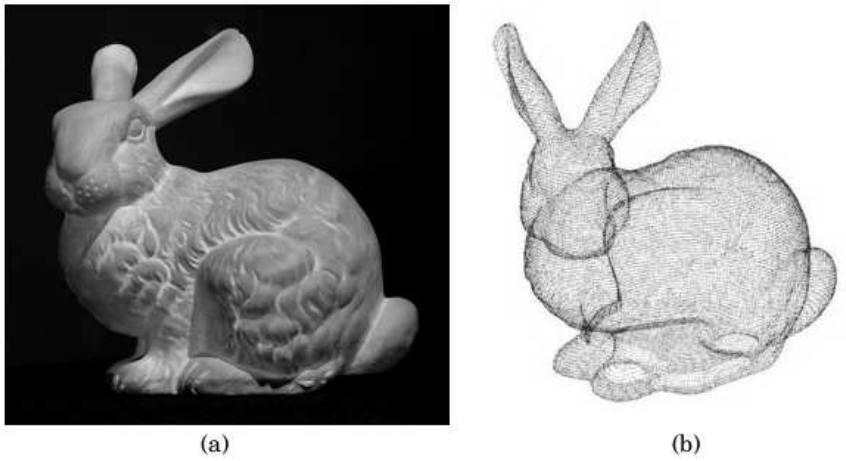


Figura 1.2: Nube de puntos. (a) Imagen de un conejo. (b) Nube de puntos tridimensionales extraídos. (Imágenes obtenidas de la página del laboratorio de gráficas por computadora de la Universidad de Stanford)

El enfoque que se seguirá en este documento esta basado en el artículo de Marc Pollefeys [Pollefeys et al., 2004]; es un enfoque modular en el cual la reconstrucción se llevará a cabo en varias fases. El resultado de cada fase será usado como información de entrada de la fase siguiente. Cada fase presenta un caso de estudio particular por sí mismo por lo que un objetivo intermedio es el de seleccionar un método para realizar cada fase así como implementarlo y valorar sus resultados basándose en métodos estadísticos. Así se evaluarán tanto los resultados obtenidos en cada una de las fases como el resultado obtenido al finalizar el proceso completo. Las fases que se seguirán se describen a continuación, cada una de estas fases es tratada a fondo en capítulos posteriores.

Relacionando las imágenes. En esta primera fase se busca una manera de encontrar correspondencias entre las imágenes, así como establecer una geometría entre ellas. Empezamos analizando los detectores de puntos característicos, luego se utilizan métodos de auto-correlación para obtener un conjunto de correspondencias tentativas. Después se procede a estimar la matriz fundamental; esta estimación se realiza de manera robusta utilizando un método para eliminar correspondencias erróneas y un algoritmo lineal para calcular de manera rápida la matriz fundamental. Una vez obtenida la matriz fundamental, esta se utiliza para hacer una búsqueda guiada de correspondencias.

Obteniendo la estructura proyectiva. La siguiente fase de la reconstrucción comienza con el establecimiento de un marco de referencia, esto se realiza seleccionando dos imágenes de la secuencia y calculando las matrices de proyección de ambas. Luego se triangulan las correspondencias obtenidas en la fase anterior para conseguir un conjunto inicial de puntos tridimensionales. A continuación se añaden nuevas vistas (imágenes) una por una. Cada vez que se añade una vista se calcula su matriz de proyección, se refinan los puntos tridimensionales ya calculados y se triangulan nuevos. Al finalizar esta fase se cuenta con las matrices de proyección de todas las imágenes así como un conjunto inicial de puntos tridimensionales que difieren de su posición real por una transformación proyectiva[1].

Reconstrucción métrica. La última fase consiste en encontrar la transformación que convierte la reconstrucción proyectiva a una métrica. Para esto es necesario encontrar los parámetros internos de la cámara (foco, punto principal, entre otros). Para este fin usamos un método de auto-calibración basado en la cónica absoluta. Una vez encontrados los parámetros internos se prosigue a la rectificación de las imágenes, proceso que se lleva a cabo para simplificar la búsqueda de correspondencias. Por último se realiza la estimación densa de la superficie en donde se encuentran correspondencias entre casi todos los pixeles de las imágenes utilizando un método para sistemas

[1] El concepto de transformación proyectiva se tratará en el siguiente capítulo

calibrados[2], obteniendo la nube de puntos tridimensionales final.

1.2. Conceptos Generales

Antes de empezar con la descripción detallada de la reconstrucción tridimensional y los métodos que se han propuesto para obtenerla, existen algunos conceptos básicos los cuales deben de ser comprendidos de manera clara ya que son necesarios para entender los conceptos más avanzados descritos en capítulos posteriores de esta tesis. Dichos conceptos básicos serán tratados a continuación.

Podemos describir una imagen como *'Una función bidimensional de la intensidad de la luz $I(x, y)$ donde x, y son coordenadas espaciales y cualquier punto (x, y) es proporcional a la brillantez de la imagen en ese punto'* [Gonzalez and Woods, 1993] (Figura 1.3). Siguiendo esta definición lo que conocemos como imagen digital corresponde a una imagen que ha sido discretizada tanto en las coordenadas espaciales como en los niveles de brillantez. Podemos imaginar una imagen como una matriz cuyos índices de fila y columna identifican a un punto en la imagen y el valor del elemento de la matriz identifica el nivel de gris (en caso de imágenes en escala de grises) en ese punto. Los elementos que forman esta matriz son llamados comúnmente *pixeles*.

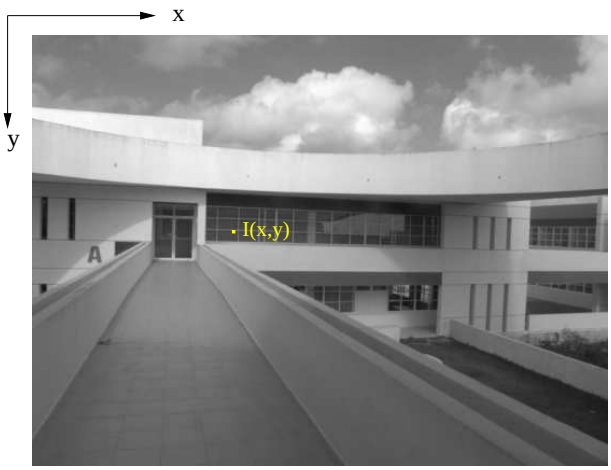


Figura 1.3: Imagen Digital

[2] Un sistema calibrado es aquel en el que se conocen los parámetros internos de la cámara

El proceso necesario para reconstruir una escena tridimensional, partiendo de información bidimensional contenida en imágenes, lleva ya varios años de investigación y ha pasado por muchas etapas. Actualmente existen varias ramas de investigación en las que este problema ha desembocado. Para entender bien cómo la reconstrucción se lleva a cabo, y cómo han surgido las diferentes aproximaciones a su solución, debemos tener cierta comprensión de la manera en que el mundo tridimensional se proyecta en nuestros ojos y el proceso de cómo nuestro cerebro es capaz de extraer información de profundidad de dichas proyecciones bidimensionales.

El problema de cómo recuperamos la profundidad de los objetos, ha sido tratado por varios autores desde hace mucho tiempo. En particular fue abordado de manera notable por Berkeley (1709). Las ideas de Berkeley y otros empiristas británicos han dominado nuestra forma de pensar en muchos aspectos de la percepción. En general ellos pensaban que toda la información es recibida por los sentidos, por lo que la percepción de profundidad se lograba mediante una mezcla de pistas dadas tanto por el sentido de la vista como por el tacto. Pistas fisiológicas como la convergencia y acomodamiento de los ojos y otras como la difuminación de la imagen han sido estudiadas, aunque se ha descubierto que no proveen suficiente información. Además de esto, existe otra fuente importante de información, esta fuente se llama: *estereopía*.

La estereopía es una característica que tienen los animales con campos visuales que se sobreponen, generados por la separación horizontal de los ojos, por lo que cada ojo tiene una vista diferente del mundo, esta diferencia ocasiona una disparidad entre las vistas, la cual es una propiedad muy importante en el calculo de profundidades (Figura 1.4). Debido al extendido uso de esta propiedad en métodos de reconstrucción tridimensional, la estereopía será discutida con más detalle posteriormente.

Existen otras pistas que nos permiten obtener información de la profundidad, muchas de ellas están relacionadas con la perspectiva, las cuales surgen por la forma en que el mundo tridimensional se proyecta en la retina 'bidimensional'. Una de ellas es la perspectiva lineal; un ejemplo de este efecto se observa en la separación horizontal de

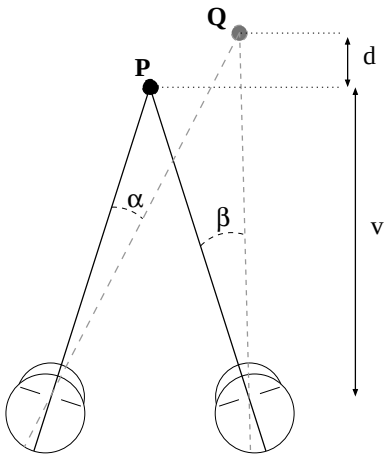
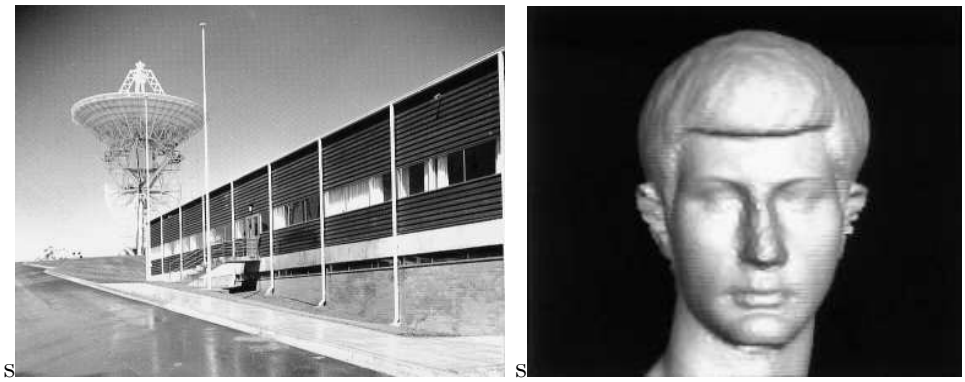


Figura 1.4: Estereopía. Los puntos P y Q se encuentran a diferentes profundidades.

lineas paralelas (Figura 1.5a), ésta es más grande a una profundidad menor y decrece a mayores profundidades. Otra característica importante es el sombreado (Figura 1.5b), que proporciona la impresión de solidez o profundidad de los objetos que vemos en la imagen. La oclusión es un aspecto presente en muchas de las escenas del mundo en donde se ven objetos a diferentes distancias. Cuando un objeto se sobrepone u ocluye a otro podemos decir que éste se encuentra más cerca; en general esto sólo nos proporciona información acerca del orden de objetos en cuanto a su profundidad, pero no nos da alguna medida de ésta.



(a) (b)

Figura 1.5: Pistas de profundidad.(a) Perspectiva lineal. Las líneas horizontales de la fachada del edificio son paralelas. Se observa que la separación entre ellas disminuye a mayor profundidad. (b) Sombreado. Las sombras proporcionan una impresión de solidez y profundidad en los objetos.

1.3. Descripción del documento

Este documento esta conformado de la siguiente manera: en el capítulo 2 se da un repaso general de algunos métodos utilizados para hacer reconstrucción a partir de una secuencia de imágenes. El capítulo 3 presenta algunos métodos para relacionar las imágenes comenzando desde la búsqueda de puntos característicos hasta la obtención de la geometría epipolar. En el capítulo 4 se obtiene una primera reconstrucción (proyectiva) basándose en la geometría encontrada en la fase anterior y se da un algoritmo para la triangulación de puntos tridimensionales. El capítulo 5 se obtiene la matriz de calibración la cual se usa para transformar la estructura obtenida de proyectiva a métrica. Para este fin, se introduce el concepto de auto-calibración así como la estimación densa de la superficie. El capítulo 6 muestra la realización de algunos experimentos utilizando todos los métodos descritos anteriormente. En el último capítulo se presentan las conclusiones y recomendaciones derivadas de lo observado y analizado durante las implementaciones y las pruebas.

Capítulo 2

Antecedentes

2.1. Introducción

En la Figura 2.1 se muestra un esquema de las ramas en que se han divido los métodos para realizar la reconstrucción tridimensional. El esquema no pretende ser exhaustivo sino dar un panorama general de los métodos más utilizados y los que llevan ya un largo tiempo de ser investigados.

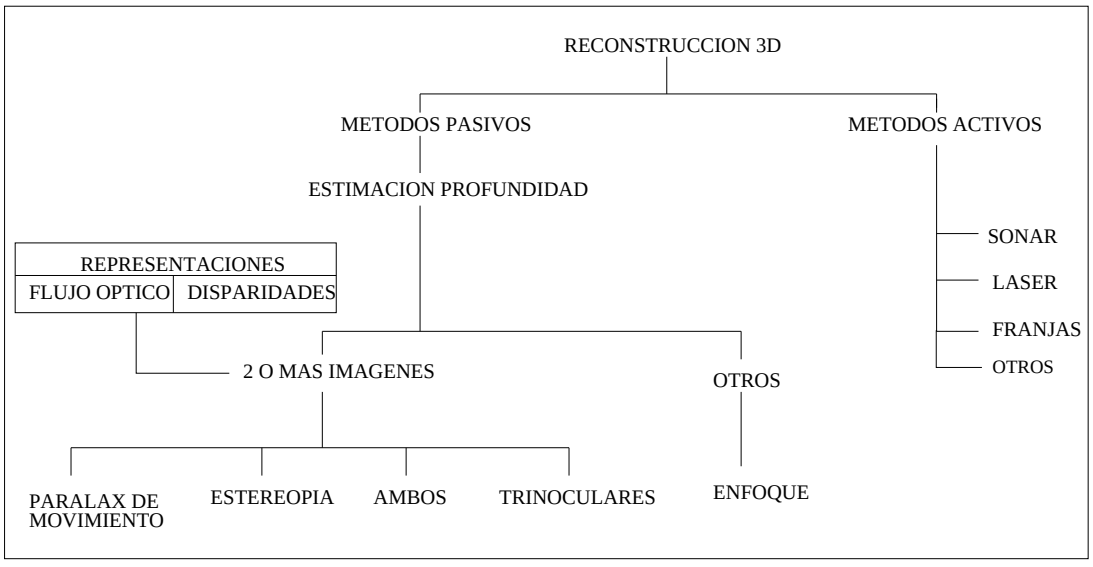


Figura 2.1: Panorama general de los métodos utilizados para realizar reconstrucción tridimensional

Existen varios métodos para lograr la reconstrucción tridimensional, los cuales pode-

mos clasificar en dos grandes campos como lo son los métodos activos y los pasivos. La gran diferencia entre estos radica en que los métodos activos emiten energía de forma controlada en el medio ambiente, recuperando así la información que requieren para realizar la reconstrucción; debido a esto, el problema de la reconstrucción se simplifica aunque por otro lado también se reduce su campo de aplicación. Ejemplos de los sistemas emisores de energía utilizados por los métodos activos son: el sonar, el láser y la proyección de franjas, entre otros, como se puede observar en la Figura 2.1. No entraremos en la descripción de estos métodos ya que no forman parte del objetivo principal de este documento. Los métodos pasivos, que a diferencia de los activos no emiten energía en el proceso de obtención de la información de una escena, son más flexibles pero computacionalmente más demandantes. A lo largo de este capítulo se discutirán algunos de ellos y como han ido evolucionando al paso del tiempo.

Como ya se ha mencionado, el objetivo de estos métodos es recuperar la estructura tridimensional de una escena. También se ha hablado en el capítulo anterior de la existencia de pistas o claves en las imágenes que nos proporcionan la información necesaria, o parte de ella, para realizar la reconstrucción. David Marr (1982) propuso una serie de esquemas o niveles en que dichas pistas se van combinando y procesando hasta obtener una representación tridimensional de la escena. El primer nivel es el esbozo primario (*primal sketch*) en donde se recaba la información de discontinuidades, lineas, bordes, grupos de objetos, textura, sombreado, etc.; toda esta información contribuye a crear el siguiente nivel de representación llamado el esquema $2\frac{1}{2}$-D, que captura ya varios datos acerca de profundidades (distancia desde el observador) y orientaciones de superficies entre otros, pero no hace explícita la estructura tridimensional de los objetos. El esquema $2\frac{1}{2}$-D es una de las principales aportaciones de Marr a la teoría de representación visual, aunque aun sigue bajo investigación si la visión humana funciona de esta manera.

Dentro de los métodos pasivos se puede encontrar una sub-clasificación en dos ramas muy marcadas: los que utilizan dos o mas imágenes (ya sea adquiridas por una cámara fotográfica o de una secuencia de vídeo), y aquellos que usan una sola imagen

u otras técnicas. Las primeras son por mucho las mas estudiadas y en las cuales se han logrado muchos avances y buenos resultados. Cabe hacer énfasis en este momento que la elección de alguno de los métodos o tipos de representación, descritos posteriormente, para cierta aplicación particular depende en buena medida en los requerimientos y resultados esperados por dicha aplicación y no tanto en el desempeño que tiene ese método en general.

A continuación se dará un repaso a los métodos más utilizados para reconstrucción tridimensional basados en el uso de dos o más imágenes, así como sus representaciones y los enfoques principales desde los cuales se han abordado.

2.2. Representaciones

Ubicándonos dentro del marco de los métodos que utilizan dos o más imágenes, existen dos principales maneras de representar la información adquirida de estas: el flujo óptico y las disparidades. En los apartados siguientes se describe cada una de éstas.

2.2.1. Flujo Óptico

Desde el punto de vista de visión computacional podemos definir el flujo óptico como *'el movimiento aparente de patrones de brillo en la imagen'* ocasionado por el movimiento relativo entre el observador y los objetos en la escena. En general el flujo óptico corresponde al campo de movimiento, aunque esto no siempre es cierto en la realidad, por ejemplo, si tenemos una esfera sin marcas que se encuentra rotando con una iluminación constante, no podemos detectar su movimiento en una imagen ya que no hay cambio de la intensidad en el tiempo.

Existen varios modelos computacionales para recuperar el movimiento del observador a través del flujo óptico. La mayoría de los modelos comienzan intentando encontrar el campo vectorial que describe como la imagen cambia en el tiempo y asumen que las velocidades bidimensionales de la imagen pueden obtenerse de las intensidades

cambiantes, resultantes del movimiento del observador o de objetos en la escena. Un ejemplo del flujo óptico se puede observar en la Figura 2.2.

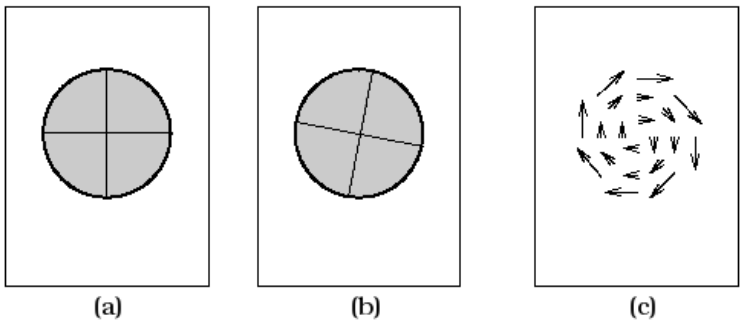


Figura 2.2: Flujo Óptico. (a) Tiempo 1. (b) Tiempo 2. (c) Flujo Óptico

Los modelos computacionales para recuperar el flujo óptico, se pueden dividir en las siguientes categorías (las cuales se describen con mas detalle en [Hildreth and Royden, 1998] y [Barron et al., 1992]):

1. Modelos Discretos

   En este modelo se escogen algunos puntos o características de la imagen, los cuales se siguen en el tiempo. Las posiciones de estos puntos en la secuencia de imágenes nos sirven para calcular los parámetros de la estructura tridimensional y el movimiento. Este tipo de algoritmos son vulnerables al error, aunque éste puede ser minimizado si consideramos un mayor número de imágenes.

2. Modelos Diferenciales

   Los modelos diferenciales se basan el derivadas espaciales del campo de flujo de la imagen para recuperar los parámetros deseados, por ejemplo, se utilizan propiedades invariantes a la rotación del observador como son la divergencia y deformación del campo de flujo para recuperar la traslación. La mayoría de estos modelos requieren que el campo de flujo óptico sea continuo y suave.

3. Modelos de Paralax de movimiento

   Estos modelos utilizan la característica de los componentes de translación de

las velocidades de la imagen, la cual nos dice que la velocidad depende de la profundidad de los puntos en la escena. De esta manera podemos eliminar los componentes rotacionales restando la velocidad de dos puntos localizados en una discontinuidad de profundidad. Estos modelos funcionan bastante bien en presencia de ruido en las velocidades y nos proporcionan información extra como ubicación de bordes de objetos. Algunos de estos modelos se han mostrado consistentes con lo observado en estudios perceptuales anteriores.

4. Modelos de minimización de error

   Un acercamiento clásico son los modelos de minimización de error, los cuales se basan en la estimación de los parámetros que mejor se ajusten al flujo óptico medido. Esto se realiza escogiendo un criterio de error. Este tipo de modelos toleran un error sustancial en la medición del movimiento de la imagen.

5. Modelos de Plantillas

   Este acercamiento esta basado en una familia de plantillas, cada una diseñada para responder a un campo de flujo generado por un movimiento específico del observador. Después se escoge la plantilla que mejor responda al flujo óptico medido. Estos modelos manejan el ruido de las mediciones de velocidades en la imagen de una manera efectiva y requieren pocas operaciones complejas, aunque por otro lado está la necesidad de contar con un número grande de plantillas la todas las combinaciones posibles de traslación y rotación.

6. Modelos dinámicos

   La teoría de sistemas dinámicos es utilizada para obtener un modelo de la evolución del campo de flujo óptico en el tiempo. El cambio de la estructura del campo de flujo alrededor de singularidades como el foco de expansión se utiliza para recuperar los parámetros de movimiento del observador.

7. Modelos directos

   A diferencia de los modelos anteriores, estos no realizan el cálculo del campo de flujo de la imagen, sino que obtienen los parámetros del movimiento y la estructura directamente de las derivadas espaciales y temporales de las intensidades de la imagen.

2.2.2. Disparidades

La idea de utilizar disparidades en el proceso de reconstrucción tridimensional proviene de la separación que existe entre los ojos de los humanos y algunos otros animales, lo cual genera diferencias entre las imágenes captadas por cada ojo. Son estas diferencias las que nos proporcionan una muy buena pista para obtener información de profundidad. Esta superposición de los campos visuales se conoce como estereopía.

Experimentos realizados desde el siglo XIX han demostrado cómo la información proporcionada por las disparidades contribuye de manera notable a la percepción de profundidad, el estereoscopio de Wheatstone (1838) así como los estereogramas de puntos aleatorios desarrollados por Bela Julesz en la década de los sesenta son ejemplos de éstos y nos presentan evidencia clara de cómo, a partir de las disparidades, podemos apreciar profundidad sin contar con otra información disponible de antemano. Un algoritmo sencillo para generar estereogramas de puntos aleatorios se puede encontrar en [Thimbleby et al., 1994].

Algunos de los métodos descritos en la siguiente sección se han seleccionado para realizar las reconstrucciones en este trabajo. Explicaciones más detalladas de dichos métodos así como las razones para su selección se proporcionan en el siguiente capítulo.

2.3. Métodos basados en disparidades

Existen muchas métodos utilizados para obtener un modelo tridimensional a partir de una secuencia de imágenes. En general se sigue una serie de pasos, que se resumen de la siguiente manera:

1. Captura de las imágenes.
2. Encontrar puntos característicos en las imágenes
3. Cálculo de las correspondencias entre dichos puntos característicos.
4. Encontrar una geometría que relacione las imágenes.

5. Obtención de una reconstrucción proyectiva.

6. Auto-calibración.

7. Rectificación y estimación densa de la superficie.

En la historia de los métodos de reconstrucción tridimensional destaca el creado por Longuet-Higgins [Longuet-Higgins, 1981] en 1981 conocido como el algoritmo de los ocho puntos (*eight-point algorithm*), en él se describe cómo se puede generar una reconstrucción a partir de dos imágenes si se cuenta con ocho puntos cuyas posiciones se conocen en ambas imágenes. La importancia de este algoritmo radica en la primera aparición de lo que posteriormente se conocerá como la matriz esencial, un concepto sumamente importante en el desarrollo de futuros métodos, la cual nos proporciona información que facilita la búsqueda de correspondencias entre puntos de dos imágenes. Otra característica de este algoritmo es su linealidad que lo hace rápido y fácil de implementar. A pesar de todo esto, el algoritmo de los ocho puntos ha sido criticado por ser excesivamente sensible al ruido en el momento de calcular las correspondencias de puntos, por lo que otros métodos han sido propuestos para lograr un mejor cálculo de la matriz fundamental, aunque la mayoría de estos de una mayor complejidad.

Durante mucho tiempo ésta siguió siendo la opinión general, hasta que Hartley [Hartley, 1997] en su artículo llamado, de manera muy descriptiva, 'En defensa del algoritmo de ocho puntos' (*In Defense of the Eight-Point Algorithm*), demostró que tomando en cuenta algunas consideraciones numéricas el algoritmo de los ocho puntos proporcionaba resultados cercanos a los obtenidos cuando se utilizan los mejores y computacionalmente más demandantes algoritmos iterativos y en ocasiones los superaba. Los cambios que propuso Hartley al algoritmo consisten principalmente en una normalización de las coordenadas antes de aplicar el algoritmo. Como un primer paso las coordenadas en cada imagen se trasladan para llevar el centroide del conjunto de puntos al origen, luego los puntos son escalados de tal manera que la distancia promedio al origen es igual a $\sqrt{2}$, esta transformación se aplica a las dos imágenes de manera independiente. Esta simple normalización de coordenadas mejora de una manera notable el condicionamiento del problema así como los resultados obtenidos,

sin la normalización el comportamiento del algoritmo es bastante pobre obteniendo en ocasiones errores de hasta 10 pixeles (tomando en cuenta que muchas veces se trabaja con exactitudes de subpixeles, los resultados son inservibles).

Como se puede ver, en el caso de Longuet-Higgins y Hartley descrito anteriormente, una simple modificación puede arrojar resultados muy superiores. Esto se debe tener en cuenta al momento de evaluar cualquier algoritmo. En la siguiente sección se dará una breve descripción de algunos conceptos básicos para luego continuar con un repaso de los métodos más utilizados para realizar el proceso de reconstrucción.

2.3.1. Conceptos Generales

Un concepto importante dentro de la reconstrucción corresponde al de transformación proyectiva. Aunque no nos demos cuenta, en nuestra vida diaria vemos transformaciones proyectivas a cada momento; por ejemplo, vemos círculos que parecen elipses, líneas que sabemos que son paralelas pero que parecieran intersectarse si se extendieran, etc. La transformación que mapea estos objetos a una imagen es también un ejemplo de transformación proyectiva. Una pregunta que debemos hacernos es cuáles propiedades geométricas son preservadas por estas transformaciones, de los ejemplos podemos notar que la forma y el paralelismo no son propiedades que se preserven, tampoco lo son los ángulos ni las distancias, de hecho son pocas las propiedades invariantes entre ellas se encuentran la colinealidad y la razón cruzada de distancias. Un ejemplo de una reconstrucción proyectiva se ilustra en la Figura 2.3.

Ya que en primera instancia lo que se conseguirá es una reconstrucción proyectiva, trabajaremos dentro de un espacio proyectivo. Un espacio proyectivo es una extensión del espacio Euclidiano en el cual los puntos se representan como vectores homogéneos. En el espacio Euclidiano bidimensional un punto es representado por dos números reales (x, y), un vector homogéneo que representa al mismo punto se forma añadiendo una coordenada extra $(x, y, 1)$. Hacemos una definición ahora de clases de equivalencia entre vectores homogéneos, así el vector $(x, y, 1)$ representa el mismo punto que el vector (kx, ky, k) (pertenecen a la misma clase de equivalencia), para k diferente de

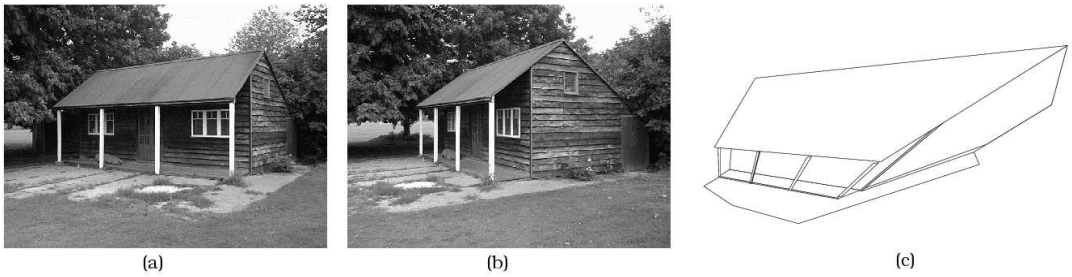


Figura 2.3: Ejemplo de una reconstrucción proyectiva. (a) y (b) Imágenes de una escena. (c) Una reconstrucción proyectiva de la escena. (Imágenes tomadas del libro *Multiple View Geometry in Computer Vision*, Hartley and Zisserman, 2004.)

cero. A lo largo de este documento se trabajará con vectores homogéneos (a menos que se indique lo contrario). Una revisión más profunda de geometría proyectiva se puede encontrar en [Hartley and Zisserman, 2004]. Si se prefiere una aproximación puramente algebraica ésta se puede encontrar en [Semple and Kneebone, 1952].

2.3.2. Relacionando las Imágenes

Detectores de Esquinas. El primer paso que se debe realizar, luego de haber adquirido las imágenes, es el cálculo de correspondencias, que consiste en encontrar una transformación que relacione una imagen con otra, es decir, dado un punto en una imagen encontrar el punto correspondiente en otra. Como ya se ha mencionado, este no es un problema sencillo ya que muchas características de la imagen dificultan el cálculo de las correspondencias, produciendo resultados incorrectos, por ejemplo, áreas homogéneas en la imagen (donde la variación de intensidad es baja o nula) tiende a producir falsas correspondencias así como también la oclusión (aparece cuando un área visible en una imagen no lo es en otra). Para establecer una relación geométrica entre las imágenes no es necesario determinar la relación entre todos los puntos de la imagen, sólo es necesario un subconjunto de éstos; por lo tanto, el primer paso para realizar el cálculo de correspondencias es la selección de un conjunto de puntos característicos los cuales se puedan diferenciar de sus vecinos y no sufran el efecto de oclusión. A estos puntos se les ha dado el nombre común de esquinas[1] (aunque no siempre representan

[1] En el campo de visión computacional la palabra esquina es utilizada para indicar pixeles en la imagen que tienen una gran variación de intensidad en todas direcciones con respecto a los pixeles vecinos, esta es la definición que usaremos aquí (aunque no existe una definición matemática aceptada de

'esquinas' en el sentido común de la palabra).

Un buen detector de esquinas debe satisfacer varios criterios como: detectar todas las esquinas verdaderas, no detectar esquinas falsas[2], buena localización, robusto respecto al ruido, estable y ser computacionalmente eficiente. Se han propuesto una gran cantidad de detectores de esquinas desde la aparición del primero de ellos a finales de los setentas. Una línea del tiempo de algunos de estos se puede observar en la Figura 2.4.

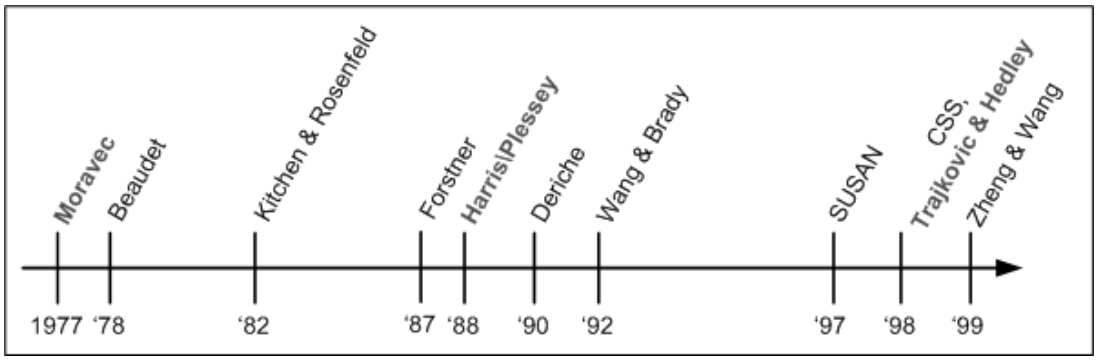


Figura 2.4: Línea del tiempo de algunos detectores de esquinas

Hacer una clasificación de estos métodos es bastante difícil pues muchas veces mezclan diferentes técnicas. Kitchen y Rosenfeld propusieron una medida de esquinidad basadas en el cambio de dirección del gradiente a lo largo de un borde multiplicado por la magnitud del gradiente local [Kitchen and Rosenfeld, 1982]. Wang y Brady observaron que la curvatura total de la intensidad de gris de la imagen es proporcional a la derivada direccional de segundo orden en la dirección de la normal del borde e inversamente proporcional a la fuerza del borde [Wang and Brady, 1995]. Trajkovich y Hedley implementaron la propiedad directa de las esquinas de que el cambio de intensidad de la imagen debe ser alto en todas direcciones [Trajkovic and Hedley, 1998]. Algunos autores han clasificado estos métodos en tres principales categorías: los métodos relacionados con los bordes. los topológicos y los de auto-correlación. Entre los métodos de auto-correlación tenemos el de Moravec [Moravec, 1977, 1979] y el de Harris [Harris

manera general).

[2] Un ejemplo de esquina 'falsa' es un punto aislado con un fondo contrastante ya que este tipo de puntos pueden ser ocasionados por ruido en la imagen

and Stephens, 1988] siendo este último uno de los más utilizados en la literatura y que será implementado en este documento, una descripción más detallada del mismo se da en el siguiente capítulo. Uno de los métodos más recientes y de gran aceptación, debido a su velocidad y robustez frente al ruido, es el de SUSAN (*Smallest Univalue Segment Assimilating Nucleus*) presentado por primera vez en [Smith and Brady, 1997]. También tenemos el método de CSS (*Curvature Scale Space*) descrito en [Mokhtarian and Suomela, 1998], éste método es bueno para recuperar características geométricas de una curva a varias escalas. Existen métodos que utilizan otras características de la imagen como líneas o curvas en lugar de puntos para establecer las correspondencias, una breve descripción de estos se puede encontrar en [Hartley and Zisserman, 2004].

Búsqueda de Correspondencias. Una vez que se han seleccionado puntos característicos en las imágenes el siguiente paso es determinar una estrategia de búsqueda y algún tipo de medida para determinar con un grado de confiabilidad dado que tan buena es la correspondencia encontrada. Las técnicas más comunes para encontrar correspondencias se pueden dividir de manera general en dos ramas: las basadas en correlación y las basadas en rasgos. Las primeras consisten en calcular la correlación entre la distribución de disparidad de una ventana centrada en un punto de una imagen y una ventana del mismo tamaño centrada en el punto a analizar de la otra imagen. Por otro lado las técnicas basadas en rasgos comparan primitivas de alto nivel (bordes, segmentos, curvas, regiones) que poseen un conjunto de características invariantes a la proyección en mayor o menor medida. En este trabajo sólo se utilizarán técnicas basadas en correlación. Un ejemplo de técnicas basadas en rasgos puede encontrarse en [Mount et al., 1997].

Algunas medidas típicas de correlación son: correlación directa, correlación de media normalizada, correlación de varianza normalizada, sumas de cuadrados de las diferencias entre pixeles correspondientes (SSD), suma de las magnitudes de las diferencias, correlación cruzada de media cero normalizada (ZNCC), entre otras. Un estudio comparativo de algunas de estas medidas se puede encontrar en [Burt et al., 1982]. Se ha sugerido también [Burt et al., 1981] el uso de imágenes filtradas con un Laplaciano com-

binado con alguna de las medidas de correlación citadas anteriormente. Este método es utilizado por Anandan [Anandan, 1984] en conjunto con el SSD obteniendo buenos resultados en casos con oclusión, también presenta una medida de confiabilidad y un algoritmo jerárquico para la búsqueda de correspondencias.

Estimando la matriz fundamental. Teniendo un conjunto de correspondencias iniciales se trata ahora de determinar la geometría existente entre cada una de las imágenes; para esto nos valemos de la geometría epipolar. La geometría epipolar es la geometría proyectiva intrínseca entre dos vistas. Existe una matriz que encapsula esta geometría llamada la matriz fundamental. Esta matriz es una generalización de la matriz esencial descrita por Longuet-Higgins, pero, a diferencia de la matriz esencial, no considera que la cámara está calibrada (el decir que una cámara esta calibrada consiste en conocer sus parámetros intrínsecos como son distancia focal, punto principal, razón de aspecto de los pixeles entre otras). Como ya se ha descrito anteriormente, el algoritmo de los 8 puntos normalizado puede ser utilizado de igual manera para encontrar la matriz fundamental y es esta aproximación la que se usará aquí. Existen otros métodos para el cálculo de la matriz fundamental, la mayoría de ellos enfocados a la minimización de alguna función de distancia, ejemplos de estos métodos pueden encontrarse en [Chojnacki et al., 2001] y [Zhang and Loop, 2001] donde se describe una nueva manera de parametrizar la matriz fundamental, otra técnicas se describen en [Hartley and Zisserman, 2004].

Un aspecto que se debe tener en consideración antes del cálculo de la matriz fundamental es la presencia de *outliers* (correspondencias erróneas) en el conjunto de correspondencias hallado en la fase anterior, ya que la implementación del algoritmo de los 8 puntos es una aproximación de mínimos cuadrados (en el caso de usar más de 8 puntos), y un sólo outlier podría generar resultados totalmente erróneos. Por esta razón se debe encontrar una manera de clasificar nuestras correspondencias en *outliers* e *inliers* (correspondencias correctas). Entre los métodos que se han desarrollado para tratar con este problema destacan el RANSAC propuesto por [Fischler and Bolles, 1981] y el LMS descrito en [Zhang et al., 1995].

2.3.3. Recuperación de la estructura proyectiva

Después de dar un repaso en la sección anterior a algunos de los métodos existentes para relacionar imágenes, pasamos a la siguiente fase de la reconstrucción. Ésta consiste en recuperar la estructura de la escena y el movimiento de la cámara a partir de la relación obtenida entre las imágenes en la fase anterior. Diversos métodos se han desarrollado para obtener la estructura de una escena cuando se trabaja con múltiples imágenes, entre ellos tenemos el uso de tensores descrito en Hartley and Zisserman [2004], y los descritos en [Beardsley et al., 1997; Koch et al., 1999].

Un paso esencial en este proceso corresponde al cálculo de las matrices de proyección correspondientes a cada vista (imagen). El procedimiento que se seguirá en este documento está basado en el descrito en [Pollefeys et al., 2004], y de manera general consiste en lo siguiente: primero se seleccionan dos imágenes y se determina un marco inicial de reconstrucción entre ellas, luego la pose de la cámara para las otras vistas se calcula con referencia a este marco, cada vez que se agrega una nueva vista la reconstrucción se refina. En nuestro caso se asume que la cámara sigue un movimiento suave y continuo por lo que una imagen solo se relaciona con su predecesora y con su sucesora, cuando esta condición no se cumple existen métodos robustos para relacionar una vista con varias de las imágenes que componen la secuencia, algunos de estos métodos se describen en [Pollefeys, 2000].

Una vez que se han definido las matrices de proyección (cámaras), se puede obtener una reconstrucción proyectiva inicial; para este fin necesitamos de algún método de triangulación. Existen varias formas de realizar la triangulación tanto con métodos lineales, los cuales necesitan solo de 6 correspondencias entre dos imágenes así como sus respectivas matrices de proyección, como métodos no lineales que buscan la minimización de alguna función de distancia. Un solución óptima, aunque computacionalmente más costosa, que mezcla las técnicas anteriores se encuentra en Hartley and Zisserman [2004] y es la que se utilizará en este trabajo.

2.3.4. Reconstrucción métrica

Hasta este momento contamos con un conjunto de puntos tridimensionales que representan la escena proyectivamente. Para lograr una reconstrucción métrica necesitamos conocer la calibración de la cámara, es decir, conocer los parámetros intrínsecos de ésta. Al proceso de obtener estos parámetros basándonos únicamente en la información obtenida de la secuencia de imágenes se le denomina auto-calibración.

Los algoritmos de auto-calibración presentan muchas variantes, muchos consideran que ningún parámetro intrínseco se conoce pero que todos permanecen constantes a lo largo de la secuencia de imágenes. Por otro lado también se han propuesto algoritmos para parámetros intrínsecos variantes. Algo que se debe de tener presente es que existen algunos movimientos de la cámara que no son lo suficientemente generales para permitir la auto-calibración. Varios métodos para llevar acabo la auto-calibración se han desarrollado, encontrándose entre los principales el uso de las ecuaciones de Kruppa, originalmente introducidas en el campo de visión por computadora por Faugeras, Luong y Maybank [Faugeras et al., 1992]. En este documento el método lineal desarrollado en [Hartley and Zisserman, 2004] será descrito.

Para simplificar la búsqueda densa de correspondencias se utiliza la rectificación epipolar de las imágenes. Esta rectificación consiste en encontrar una transformación de las imágenes que haga corresponder las filas de las imágenes con las líneas epipolares. Al igual que en las fases anteriores del proceso de reconstrucción, se cuenta con varios acercamientos a este problema. El enfoque tradicional consiste en transformar los planos de la imagen de manera que los planos correspondientes en el espacio coincidan. Este enfoque falla cuando los epipolos se encuentran dentro de la imagen. Estas transformaciones pueden generar grandes distorsiones en las imágenes rectificadas, Gluckman and Nayar [2001] proponen un método para minimizar estos efectos parametrizando la familia de transformaciones y seleccionando la que minimice el cambio en área local integrado sobre el área de las imágenes. El método presentado por Roy et al. [1997] rectifica sobre un cilindro en lugar de un plano y funciona aunque el epipolo se encuentre

dentro de la imagen, por otro lado es algo complejo y no muy eficiente. En [Pollefeys et al., 1999] se utiliza una parametrización polar de la imagen alrededor del epipolo; al igual que el método de Roy, funciona cuando el epipolo está dentro de la imagen, es más sencillo y se obtienen imágenes rectificadas pequeñas. Hartley [Hartley, 1999] propone un método basado en la matriz fundamental el cual es sencillo de implementar aunque no funciona cuando el epipolo se encuentra dentro o cerca del área de las imágenes.

Por último se realiza la estimación densa de la superficie. Uno de los métodos que han dado mejores resultados se describe en [Cox et al., 1996], en donde se utilizan conceptos probabilísticos. A partir de éste método han surgido otros como los desarrollados en [Falkenhagen, 1994] y [Falkenhagen, 1997].

Capítulo 3

Relacionando las imágenes.

3.1. Introducción

El objetivo principal de los métodos descritos a lo largo de este capítulo es el de lograr una estimación robusta de la matriz fundamental la cual encapsula la información de la geometría entre dos imágenes. Esta estimación se obtendrá a partir de puntos correspondientes entre las imágenes. Para este fin se describen los conceptos elementales de la geometría epipolar, también se describen con más profundidad algunos métodos utilizados para lograr la estimación deseada. La matriz fundamental estimada es utilizada para guiar la búsqueda de nuevas correspondencias. En los capítulos posteriores esta matriz es usada para inicializar la estructura de la escena.

La sección 3.2 nos presenta métodos para encontrar puntos característicos en las imágenes, en la sección 3.3 métodos de auto-correlación son utilizados para encontrar un conjunto de correspondencias tentativas. La sección 3.4 describe los conceptos generales de la geometría epipolar, éstos conceptos son básicos para la comprensión de las siguientes secciones. En la sección 3.5 se presenta el algoritmos para la estimación de la matriz fundamental asumiendo que varias de las correspondencias proporcionadas por los métodos de correlación son falsas y se muestra como la matriz fundamental puede ser utilizada para la búsqueda de más correspondencias. Cada una de las secciones cuenta con un apartado de experimentos donde se muestran algunos resultados obtenidos por las implementaciones realizadas.

3.2. Selección de puntos característicos

A continuación se describen con un poco más de detalle dos de los métodos más utilizados en la detección de esquinas.

3.2.1. Detector de Esquinas de Moravec

Desarrollado por Hans Moravec en 1977, específicamente para su investigación sobre la navegación de un robot [Moravec, 1977, 1979], como un método que le permitiera identificar la existencia y ubicación de objetos en el camino de éste. Se considera un detector de esquinas ya que encuentra puntos característicos en la imagen donde existe una variación grande de intensidad en todas direcciones.

Moravec propuso que la variación de intensidad en un pixel p de la imagen puede ser calculada colocando una ventana cuadrada (3x3, 5x5 o 7x7 pixeles usualmente) centrada en p y moviéndola un pixel en cada una de las ocho direcciones principales[1]. La variación de intensidad para cada uno de estos movimientos se calcula mediante la suma cuadrada de las intensidades correspondientes a los pixeles en las dos ventanas. Tener una gran variación de intensidad en todas direcciones es equivalente a que la menor variación calculada sea grande. Para dar una idea más clara en la Figura 3.1 se muestra la ventana en diferentes ubicaciones y su relación con la variación de intensidad. El algoritmo de Moravec se describe en el Cuadro 3.1.

Como se puede observar en la Figura 3.1, un solo pixel es detectado como esquina por lo que este método es muy sensible al ruido. Con respecto a características que deben tener los algoritmos de detección de esquinas podemos decir que este algoritmo es muy fácil de implementar pero muy sensible al ruido y no es invariante rotacionalmente, además tiende a encontrar falsas esquinas a lo largo de los bordes y en pixeles aislados. Una característica buena es su eficiencia computacional la cual puede ser un punto muy importante para aplicaciones en tiempo real o con poca capacidad de cálculo.

[1] Arriba, abajo, izquierda , derecha y las cuatro direcciones diagonales

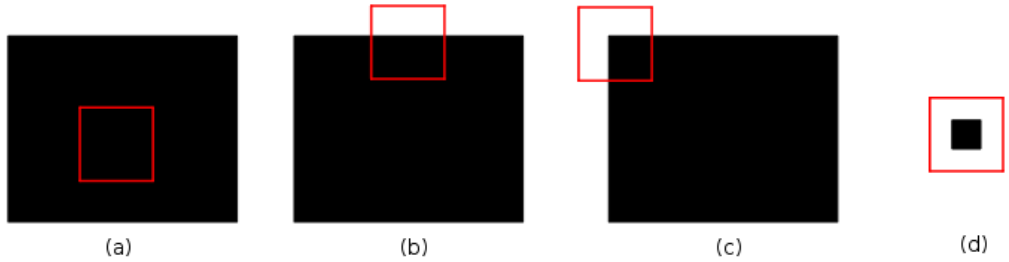


Figura 3.1: (a) Región interior. poca variación de intensidad en cualquier dirección. (b) Borde. poca variación de intensidad a lo largo del borde, variación grande en la dirección perpendicular al borde. (c) Esquina. Gran variación de intensidad en todas direcciones (d) Pixel solo. Gran variación de intensidad en todas direcciones.

3.2.2. Detector de Esquinas de Harris-Stephens

El método de detección de esquinas de Harris [Harris and Stephens, 1988], también conocido en la literatura de visión computacional como el algoritmo de Plessey, es un método de auto-correlación que intenta corregir los problemas que sufre el operador de Moravec y que fueron descritos en la sección anterior. Las correcciones que realiza son las siguientes:

- **Respuesta no isotrópica.** Ésta se debe a que solo se consideran desplazamientos en las ocho direcciones principales, es decir, cada 45 grados, esto es posible corregirlo realizando una expansión analítica alrededor del origen del desplazamiento

$$\mathbf{V}_{u,v}(x,y) = \sum_{\forall \text{i en la ventana centrada en (x,y)}} \left(u\frac{\partial I_i}{\partial x} + v\frac{\partial I_i}{\partial y} \right)^2$$

- **La respuesta es ruidosa.** Esto sucede puesto que la ventana es binaria y rectangular. Para resolver esto se puede utilizar una ventana suave circular como una Gaussiana:

$$w_{u,v} = exp^{-\frac{u^2+v^2}{2\sigma^2}}$$

Así la variación de intensidad puede escribirse como:

$$\mathbf{V}_{u,v}(x,y) = \sum_{\forall \text{i en la ventana centrada en (x,y)}} w_i \left(u\frac{\partial I_i}{\partial x} + v\frac{\partial I_i}{\partial y} \right)^2$$

Entrada:
Imagen en escala de gris $\mathbf{I}(x, y)$, el tamaño de la ventana y un umbral T.
Salida:
Matriz $\mathbf{C}$ del mismo tamaño de la imagen indicando las esquinas encontradas.

Algoritmo:

1. Para cada pixel en la imagen calcula la variación de intensidad para un desplazamiento (u, v) como sigue:

$$\mathbf{V}_{u,v}(x, y) = \sum_{\forall \text{a,b en la ventana}} (\mathbf{I}(x + u + a, y + v + b) - \mathbf{I}(x + a, y + b))^2$$

   donde los desplazamientos considerados son:

$$(1, 0), (1, 1), (0, 1), (-1, 1), (-1, 0), (-1, -1), (0, -1), (1, -1)$$

2. Construye el mapa de *esquinidad* (Una matriz del mismo tamaño de la imagen que contiene los valores de *esquinidad*), calculando la medida de *esquinidad* $\mathbf{C}(x, y)$ para cada pixel en la imagen:

$$\mathbf{C}(x, y) = \text{mín}(\mathbf{V}_{u,v}(x, y))$$

3. Umbraliza el mapa convirtiendo a cero todos los valores de $\mathbf{C}(x, y)$ menores al umbral T.

4. Realiza una supresión de valores no máximos* (*Non-maximal supression*) al mapa para encontrar los máximos locales.

Todos los valores no cero que quedan en el mapa de *esquinidad* $\mathbf{C}$ son las esquinas encontradas.

* La supresión de valores no máximos consiste en verificar que cada punto sea un máximo dentro de una vecindad determinada. Los puntos que no cumplan con este requisito son eliminados.

Cuadro 3.1: Algoritmo de Moravec

donde w_i es el peso de la ventana Gaussiana en la posición i.

- **Respuesta grande en los bordes.** Para corregir este problema se creo una nueva forma de medir la *esquinidad*[2] tomando en cuenta la variación de intensidad en todas las direcciones de desplazamiento consideradas:

[2] El concepto *esquinidad* se refiere a al valor asignado a un pixel que nos indica que tanto puede o no ser considerado como una esquina.

$$\begin{aligned}\mathbf{V}_{u,v}(x,y) &= \sum_{\forall \text{i en la ventana centrada en (x,y)}} w_i \left(u\frac{\partial I_i}{\partial x} + v\frac{\partial I_i}{\partial y}\right)^2 \\ &= \sum_{\forall \text{i en la ventana centrada en (x,y)}} w_i \left(u^2\frac{\partial I_i^2}{\partial x} + 2uv\frac{\partial I_i}{\partial x}\frac{\partial I_i}{\partial y} + v^2\frac{\partial I_i^2}{\partial y}\right)^2 \\ &= \mathbf{A}u^2 + 2\mathbf{C}uv + \mathbf{B}v^2\end{aligned}$$

donde $\mathbf{A} = (\frac{\partial I}{\partial x})^2 \otimes w, \mathbf{B} = (\frac{\partial I}{\partial y})^2 \otimes w, \mathbf{C} = (\frac{\partial I}{\partial x}\frac{\partial I}{\partial y}) \otimes w$

esto expresado de manera matricial es:

$$\mathbf{V}_{u,v}(x,y) = \mathbf{A}u^2 + 2\mathbf{C}uv + \mathbf{B}v^2 = \begin{bmatrix} u & v \end{bmatrix} \mathbf{M} \begin{bmatrix} u \\ v \end{bmatrix} \quad , donde \quad \mathbf{M} = \begin{bmatrix} \mathbf{A} & \mathbf{C} \\ \mathbf{C} & \mathbf{B} \end{bmatrix}$$

La matriz **M** contiene los operadores diferenciales que describen la geometría de la superficie de la imagen (la imagen puede considerarse como la gráfica de una superficie dada por $z = f(x, y)$ donde (x, y) denotan la posición de un pixel en la imagen y $f(x, y)$ el valor de intensidad en dicha posición). Los eigenvalores de M son proporcionales a las curvaturas principales de la superficie de la imagen. Sin embargo como los componentes de M son estimados solo utilizando los gradientes verticales y horizontales, no son rotacionalmente invariantes. Tomemos como referencia la Figura 3.1 para explicar como los eigenvalores nos proporcionan información acerca de la posible ubicación de una esquina. En la Figura 3.1a se tiene poca curvatura en todas direcciones por lo que el valor de ambos eigenvalores será pequeño, en la Figura 3.1b tenemos que a lo largo del borde la curvatura es pequeña pero en la dirección perpendicular al borde es grande por lo que uno de los eigenvalores sera pequeño y el otro grande, cuando se tiene una esquina como en la Figura 3.1c ambos eigenvalores serán grandes. Gracias a esta descripción podemos diferenciar un borde de una esquina de una mejor manera que en el método de Moravec evitando así encontrar falsas esquinas a lo largo de un borde.

Harris y Stephens propusieron la siguiente medida de *esquinidad*:

$$\mathbf{E}(x,y) = \det(\mathbf{M}) - k(traza(\mathbf{M}))^2$$

donde $\det(\mathbf{M}) = \lambda_1\lambda_2 = \mathbf{AB} - \mathbf{C}^2$, $traza(\mathbf{M}) = \lambda_1 + \lambda_2 = \mathbf{A} + \mathbf{B}$ y λ_1, λ_2 representan los eigenvalores de M.

Este algoritmo consume muchos más recursos computacionales que el de Moravec, y, aunque mejora el numero de verdaderas esquinas encontradas sigue siendo sensible al ruido ya que depende del uso de gradientes, esta sensibilidad puede ser reducida aumentando el tamaño de la ventana Gaussiana pero esto trae en consecuencia un aumento en la demanda computacional, otra de las características es la mala localización de las esquinas en algunas uniones. A pesar de estas desventajas este método es más robusto que el de Moravec y ha sido objeto de mejoras recientes las cuales lo han convertido en un operador isotrópico, una de estas modificaciones puede encontrarse en [Zheng et al., 1999]. El algoritmo se describe en el Cuadro 3.2.

Entrada:
Imagen en escala de gris $\mathbf{I}(x, y)$, la varianza de la Gaussiana (la ventana normalmente tiene un radio de 3 veces la desviación estándar), el valor k y el umbral T.
Salida:
Matriz $\mathbf{E}$ del mismo tamaño de la imagen indicando las esquinas encontradas.

Algoritmo:

1. Para cada pixel en la imagen calcula la matriz de auto-correlación $\mathbf{M}$.
2. Construye el mapa de *esquinidad* calculando la medida de *esquinidad* para cada pixel en la imagen:
$$\mathbf{E}(x, y) = \det(\mathbf{M}) - k(traza(\mathbf{M}))^2$$
3. Umbraliza el mapa convirtiendo a cero todos los valores de $\mathbf{E}(x, y)$ menores al umbral T.
4. Realiza una supresión de valores no máximos al mapa para encontrar los máximos locales.

Todos los valores no cero que quedan en el mapa de *esquinidad* $\mathbf{E}$ son las esquinas encontradas.

Cuadro 3.2: Algoritmo de Harris-Stephens

3.2.3. Experimentos

Ambos métodos fueron implementados y se realizaron pruebas con diferentes imágenes tanto artificiales como naturales. En el caso del detector de esquinas de Harris el calculo discreto del gradiente se realizó utilizando el operador de Prewitt, el cual consiste en la convolución de la imagen con las siguientes máscaras:

$$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

Algunas de las características distintivas de ambos detectores de esquinas se pueden observar en la Figura 3.2, pero la observación no es suficiente para determinar la eficacia de algún método. Para esto se necesita de alguna manera numérica de medir y evaluar los resultados obtenidos. En el apéndice A se mencionan algunos métodos de evaluación y se implementa uno de los criterios ahí descritos.

En la Figura 3.2 se presentan ejemplos de las esquinas encontradas con los dos métodos. Para cada imagen se probó con varios valores para los parámetros de entrada. En general para el caso de Harris se utilizo una ventana de 7x7 para la convolución con la Gaussiana de varianza 1, la constante de Harris (k) varió entre 0.04 y 0.125, y el radio de supresión de valores no máximos (*non-maximal supression*) fue de 5 pixeles.

Algunos puntos sobresalen de la implementación realizada, por ejemplo, usando el valor recomendado por Harris para su constante k =0.04 se obtuvieron en general los mejores resultados. El valor del umbral que define cuales pixeles son considerados esquinas fue seleccionado de manera empírico y que los valores de *esquinidad* tienen una gran varianza de una imagen a otra, una manera mejor de controlar este valor sería normalizando los valores de *esquinidad* de manera que éstos se encuentren siempre dentro de un rango definido.

3.3. Cálculo de correspondencias

Una vez obtenidos puntos característicos (esquinas) en las imágenes debemos encontrar una manera de relacionarlos. Esto puede ser muy difícil debido a distintos

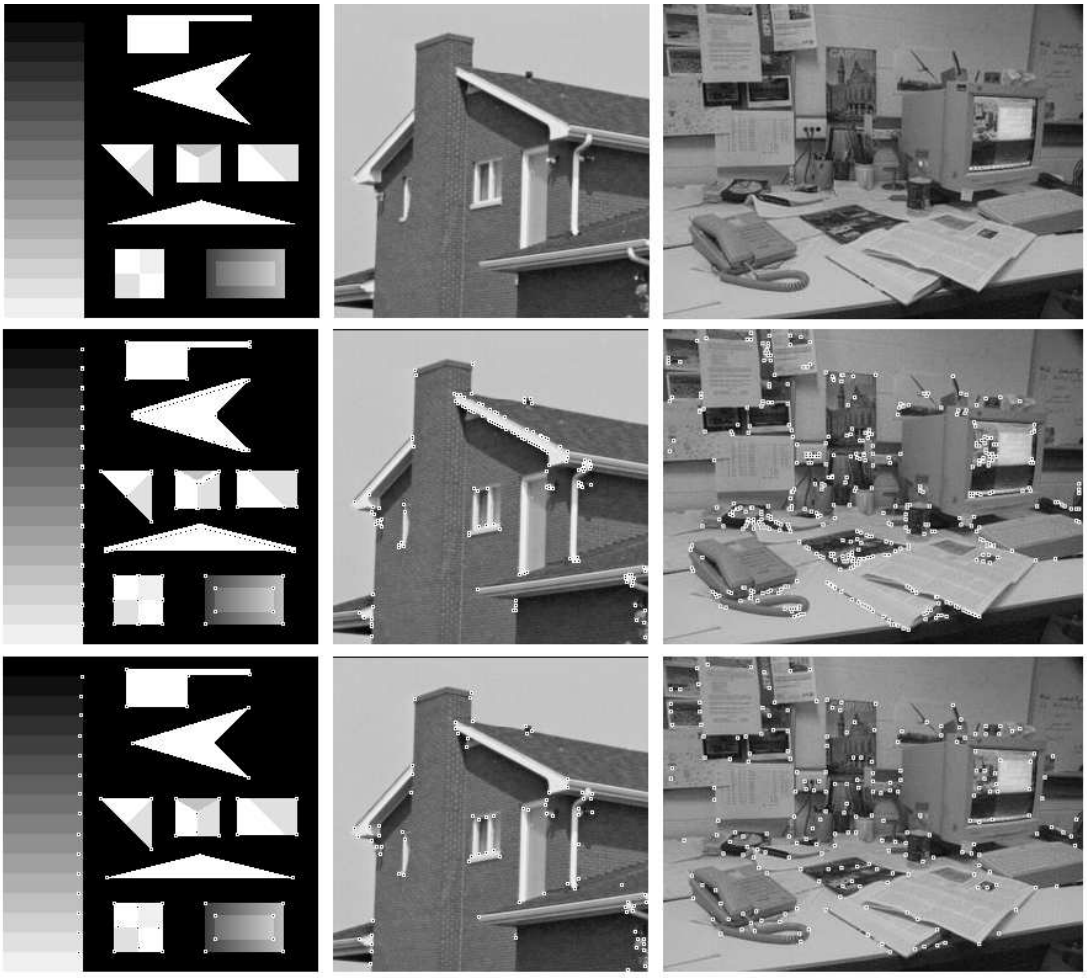


Figura 3.2: Comparación de Detectores de Esquinas. Fila superior, imágenes originales. Centro, esquinas encontradas por el detector de esquinas de Moravec. Inferior, esquinas encontradas por el detector de esquinas de Harris.

problemas que se pueden presentar como: oclusiones, es decir, algunos puntos tal vez no tengan una correspondencia debido a la presencia de algún objeto que obstruye la visibilidad del punto correspondiente, otro problema lo constituyen las falsas correspondencias, esto se da debido a la presencia de puntos similares en la vecindad de la correspondencia verdadera, también pueden haber cambios en la intensidad en puntos correspondientes de una imagen a otra debido al cambio de dirección de la luz, ruido o cambio de tamaño entre otros.

Un primer paso para encontrar correspondencias consiste en definir alguna medida que nos permita establecer cuando dos puntos característicos en diferentes imágenes son similares. Se han propuesto varias medidas en la literatura de visión computacional [Pollefeys, 2000; Pollefeys et al., 2004]. A continuación se describen dos de ellas las cuales se analizarán posteriormente.

3.3.1. Correlación cruzada de media cero normalizada (ZNCC)

Este método se basa en comparar vecindades de las esquinas mediante la correlación cruzada de sus intensidades. Considerando la vecindad como una ventana de dimensiones $(2N+1) \times (2N+1)$ pixeles centrada en la esquina entonces la medida de similitud entre los dos puntos (esquinas) (x, y) y (x', y') se obtiene de la siguiente manera:

$$C = \sum_{i=-N}^{N} \sum_{j=-N}^{N} (I(x-i, y-j) - \bar{I})(I'(x'-i, y'-j) - \bar{I}')$$

donde I e I' representan los valores de intensidad de cierto punto y los valores de $\bar{I}$ e $\bar{I}'$ representan la intensidad media (promedio de la intensidad) en la vecindad considerada. Si se considera que el desplazamiento entre las imágenes es pequeño entonces se puede reducir la complejidad combinatoria del algoritmo, es decir, en lugar de comparar cada esquina encontrada en una imagen con todas las esquinas en la otra imagen, sólo calculamos la similitud con las esquinas que se encuentren dentro de una vecindad cercana. Esto es, de manera más precisa, dada una esquina localizada en la posición (x, y) en una imagen, ésta sólo se compara con esquinas en la otra imagen localizadas en el intervalo $[x - w_i, x + w_i] \times [y - h_i, y + h_i]$, donde usualmente el tamaño de w_i y h_i corresponden con el 10 % o el 20 % del tamaño de la imagen.

Otra de las características de este método es que en ocasiones para una esquina dada existen varias esquinas en la otra imagen cuyas medidas calculadas pueden ser muy similares o incluso iguales, debido a esto no se puede confiar que todas las correspondencias encontradas serán correctas por lo que otros métodos deben de ser usados para tratar con estos *outliers* (correspondencias erróneas), los cuales en ocasiones representan una parte considerable del total de correspondencias encontradas. El método de

correlación cruzada es invariante a traslaciones y cambios en la intensidad de la imagen.

Existe una forma normalizada de esta medida conocida como ZNCC (*Zero mean Normalized Cross Correlation*) dada por:

$$S = \frac{\int\int_W (J(T(x,y)) - \bar{J}) \cdot (I(x,y) - \bar{I}) w(x,y) dxdy}{\sqrt{\int\int_W (J(T(x,y)) - \bar{J})^2 w(x,y) dxdy \cdot \int\int_W (I(x,y) - \bar{I})^2 w(x,y) dxdy}}$$

con $\bar{J} = \int\int_W J(T(x,y))dxdy$ y $\bar{I} = \int\int_W I(x,y)dxdy$ el promedio de la intensidad de la imagen en el área considerada.

Su forma discreta es:

$$C = \frac{\sum_{i=-N}^{N} \sum_{j=-N}^{N} (I(x-i, y-j) - \bar{I})(I'(x'-i, y'-j) - \bar{I'})}{\sqrt{\sum_{i=-N}^{N} \sum_{j=-N}^{N} (I(x-i, y-j) - \bar{I})^2 (I'(x'-i, y'-j) - \bar{I'})^2}}$$

Se han creado nuevas técnicas basadas en esta medida, una de ellas muy interesante, que utiliza redes neuronales para mejorar el proceso puede encontrarse en [Gallo et al., 2005].

3.3.2. Suma de diferencias cuadradas (SSD)

El método de suma de diferencias cuadradas (*sum-of-squared-differences, SSD*) es más sencillo de implementar y mucho más rápido que el de correlación cruzada ya que no requiere del cálculo de la intensidad media de la vecindad. Si consideramos una ventana W en una imagen I y su correspondiente región $T(W)$ en la imagen J, entonces la diferencia o disimilitud entre las imágenes basada en SSD esta dada por:

$$D = \int\int_W [J(T(x,y)) - I(x,y)]^2 w(x,y) dxdy$$

donde w es una función de peso, normalmente igual a uno o a una Gaussiana.

Siguiendo el mismo proceso de comparación que el método anterior, esta medida de similitud se puede expresar de manera discreta como sigue:

$$C = \sum_{i=-N}^{N} \sum_{j=-N}^{N} (I'(x'-i, y'-j) - I(x-i, y-j))^2$$

como se aprecia simplemente se calcula la diferencia de intensidades de las vecindades de los pixeles que se quieren comparar. Por otro lado al no usar información de la intensidad media este método es sensible a los cambios de intensidad que puedan sufrir las imágenes.

3.3.3. Experimentos

Antes de realizar un experimento teniendo un número grande de esquinas, se realizó una pequeña prueba para ilustrar de manera general el manejo del umbral para las medidas de similitud antes descritas. Primero se seleccionaron dos imágenes y se obtuvieron las esquinas mediante el método de Harris-Stephens descrito en la sección anterior. Se calculó la medida de similitud (ZNCC y SSD) entre una de las esquinas encontradas en la imagen izquierda con cinco de las esquinas encontradas en la imagen derecha. Las esquinas seleccionadas así como un acercamiento a las respectivas ventanas de comparación se pueden ver en las Figuras 3.3 y 3.4. Se utilizaron ventanas de comparación de 7x7 pixeles y los resultados obtenidos se pueden ver en el Cuadro 3.3.

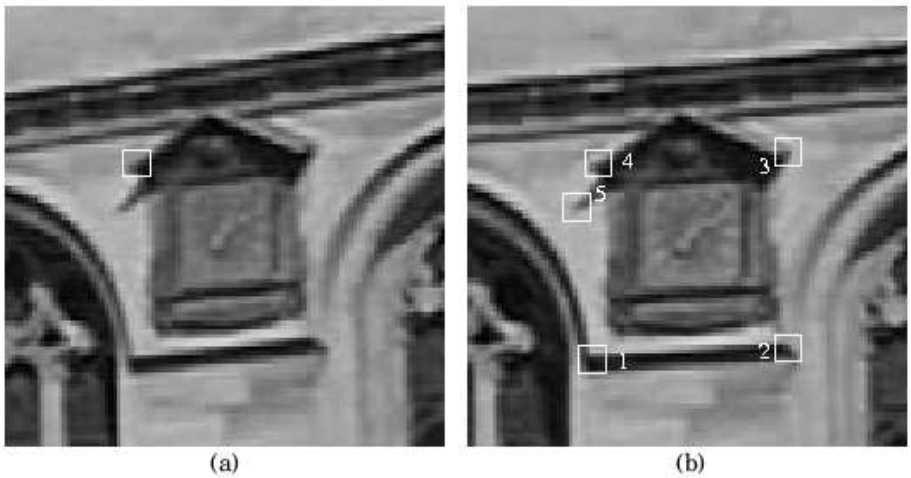
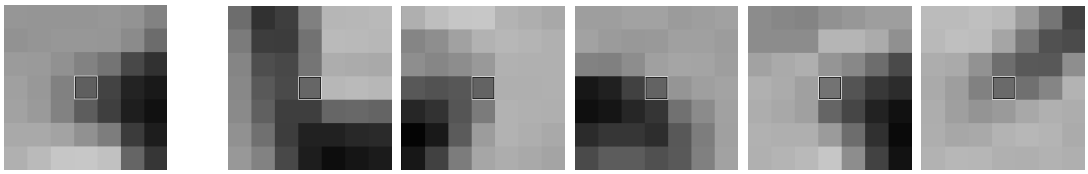


Figura 3.3: Comparación entre puntos característicos. (a) Ventana de comparación de una esquina encontrada. (b) Cinco posibles correspondencias.



(a) (b)

Figura 3.4: Ventanas de comparación de 7x7 pixeles. (a) Detalle de la ventana mostrada en la Figura 3.3a. (b) Detalle de las ventanas mostradas en la Figura 3.3b, 1-5.

Esquina	ZNCC	SSD
1	-0.1279	307,176
2	-0.4164	334,304
3	-0.4144	355,575
4	**0.9234**	**20,813**
5	0.1816	189,525

Cuadro 3.3: Resultados obtenidos por las medidas de similitud al comparar la esquina mostrada en la Figura 3.3a con las cinco esquinas que se muestran en la Figura 3.3b, utilizando ventanas de 7x7 (Figura 3.4). Los valores en negrita corresponden a los mejores valores de similitud.

Se realizaron también otros experimentos comparando todas las esquinas detectadas en las imágenes, uno de estos experimentos puede observarse en la Figura 3.5. Las imágenes tienen una resolución de 400x300 pixeles. En este experimento se encontraron poco mas de 600 esquinas en cada imagen.El resultado de nuestra implementación del método de correlación utilizando ZNCC arrojó un conjunto de 245 correspondencias. El tamaño de la ventana de búsqueda fue de 100x100 pixeles y el tamaño de la ventana de comparación de 7x7 pixeles. El umbral utilizado fue de 0.8. Como se ve en la Figura 3.5e existen varias correspondencias erróneas por lo que es necesario en las siguientes fases el uso de métodos robustos ante esta clase de problemas.

Para los experimentos que se realizarán en capítulos posteriores se ha elegido utilizar la medida ZNCC ya que se comportó de mejor manera en las diversas pruebas y también porque el valor dado por esta medida siempre se encuentra dentro de un rango ([-1, 1]), lo cual facilita su uso. Aunque un poco más compleja de implementar y más lenta que el SSD, el ZNCC es invariante ante cambios de intensidad de la imagen lo cual lo hace útil para imágenes tomadas en el exterior.

(a) (b)

(c) (d)

(e)

Figura 3.5: Cálculo Automático de correspondencias utilizando ZNCC (a) (b) Imágenes izquierda y derecha del Wadham College de la Universidad de Oxford. (c) (d) Esquinas detectadas. Hay poco mas de 600 esquinas detectadas en cada imagen. (e) 245 correspondencias encontradas utilizando ZNCC.

3.4. Geometría Epipolar

La geometría epipolar es la geometría proyectiva intrínseca entre dos vistas [Hartley and Zisserman, 2004]. Esta geometría es muy importante en el proceso de reconstrucción tridimensional ya que es independiente de la estructura de la escena y sólo depende de pose de la cámara y sus parámetros internos. Es muy importante entender el concepto de geometría epipolar ya que es una parte esencial en la metodología utilizada en este documento para realizar la reconstrucción tridimensional. Supongamos que un punto tridimensional $\mathbf{X}$ se proyecta en dos imágenes en las posiciones $\mathbf{x}$ y $\mathbf{x}'$, como se puede observar en la Figura 3.6a, estos puntos en la imagen junto con el punto tridimensional forman un plano (π). Si sólo conocemos $\mathbf{x}$, el plano π esta determinado por el rayo que pasa por el centro de la primera cámara $\mathbf{C}$ y el punto $\mathbf{x}$ y por la línea que une los centros de las dos cámaras (llamada la línea base). El punto $\mathbf{x}'$ se encuentra sobre la línea $\mathbf{l}'$ que es la intersección del plano π con el plano de la segunda imagen (Figura 3.6b).

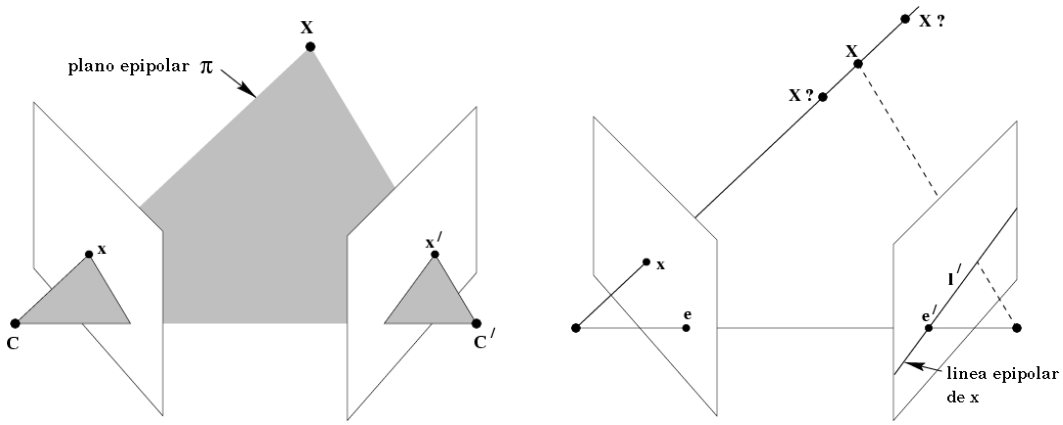
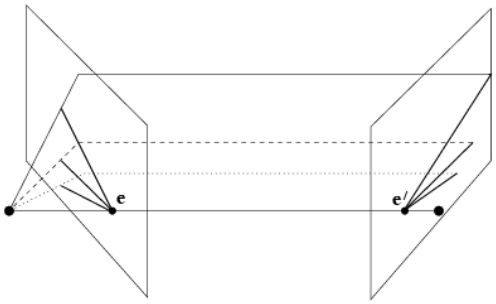


Figura 3.6: Geometría de puntos correspondientes. (a) Los centros de las cámaras ($\mathbf{C}$ y $\mathbf{C}'$) junto con un punto tridimensional $\mathbf{X}$ forman el plano π en donde también se encuentran los puntos correspondientes $\mathbf{x}$ y $\mathbf{x}'$. (b) Se muestran los epipolos $\mathbf{e}$ y $\mathbf{e}'$ localizados en la intersección de la línea que une los centros de las cámaras y los planos de la imagen, también se observa la línea epipolar $\mathbf{l}'$ correspondiente al punto $\mathbf{x}$. Imágenes obtenidas de [Hartley and Zisserman, 2004]

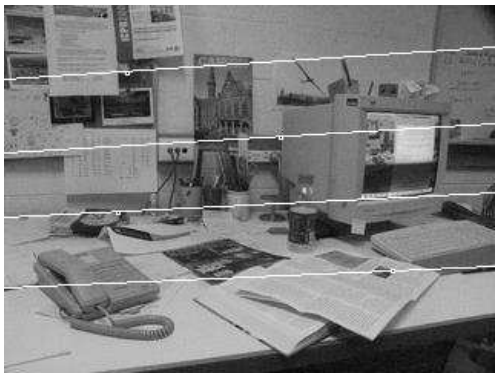
Algunos de los términos utilizados se definirán a continuación [Hartley and Zisserman, 2004]:

- **Epipolo.** Punto de intersección de la línea base con el plano de la imagen.
- **Plano epipolar.** Plano que contiene la línea base.
- **Línea epipolar.** Intersección del plano epipolar con el plano de la imagen.

En la Figura 3.7 vemos la geometría epipolar entre un par de imágenes, como se observa los epipolos se encuentran fuera del área visible de las imágenes.



(a)



(b)

(c)

Figura 3.7: Lápiz de líneas epipolares. (a) Geometría epipolar de dos vistas. (b) (c) Dos vistas de una escena con algunas líneas epipolares sobrepuestas

Como se describió anteriormente existe una matriz que encapsula la geometría epipolar: la matriz fundamental, la cual denotaremos **F**. La matriz fundamental es una matriz de 3x3 y de rango 2. Dado un par de imágenes la matriz fundamental relaciona un punto en una imagen con su correspondiente línea epipolar en la otra imagen de la siguiente manera:

$$\mathbf{l}' = \mathbf{F}\mathbf{x}$$

Los epipolos corresponden con el espacio nulo izquierdo y derecho de la matriz fundamental es decir:

$$\begin{aligned} \mathbf{F}\mathbf{e} &= \mathbf{0} \\ \mathbf{F}^T\mathbf{e}' &= \mathbf{0} \end{aligned}$$

3.5. Estimación de la matriz Fundamental

A continuación se presenta uno de los algoritmos más usados y sencillos para calcular la matriz fundamental. La matriz fundamental tiene la propiedad de relacionar puntos correspondientes en 2 imágenes, es decir, dado un punto $\mathbf{x} = (x, y, 1)$ en una imagen y su correspondiente $\mathbf{x}' = (x', y', 1)$ en la otra imagen, estos puntos satisfacen la relación ${\mathbf{x}'}^T\mathbf{F}\mathbf{x} = 0$. Esta matriz puede encontrarse dada una serie de correspondencias entre dos imágenes.

3.5.1. El algoritmo de los ocho puntos normalizado

Este método, bastante simple, requiere la construcción y solución de un conjunto de ecuaciones lineales. El algoritmo original fue propuesto por Longuet-Higgins [Longuet-Higgins, 1981] y aunque durante mucho tiempo sufrió de críticas debido a su sensibilidad al ruido sólo fue necesaria una pequeña modificación [Hartley, 1997] para convertirlo en un algoritmo confiable y que se desempeña muy bien, incluso mejor que varios algoritmos iterativos.

Como la matriz fundamental es una matriz de 3x3 determinada hasta un factor de escala, sólo son necesarias 8 ecuaciones para obtener una solución única. La forma más simple de calcularla es utilizando la ecuación ${\mathbf{x}'}^T\mathbf{F}\mathbf{x} = 0$. que puede reescribirse de la siguiente manera:

$$\begin{bmatrix} xx' & yx' & x' & xy' & yy' & y' & x & y & 1 \end{bmatrix} \mathbf{f} = 0$$

donde $\mathbf{f}$ es un vector que contiene las 9 entradas de la matriz $\mathbf{F}$.

Agrupando ocho de estas ecuaciones (una por cada correspondencia) en una matriz $\mathbf{A}$ se obtiene la ecuación

$$\mathbf{Af} = 0$$

Este sistema se resuelve de manera sencilla utilizando SVD (Descomposición de valores singulares). Aplicando esta descomposición a la matriz $\mathbf{A}$ obtenemos $\mathbf{A} = \mathbf{USV^T}$ con $\mathbf{U}$ y $\mathbf{V}$ matrices ortogonales y $\mathbf{S}$ una matriz diagonal que contiene los valores singulares en orden decreciente. La solución de nuestro sistema es el vector singular correspondiente al menor valor singular el cual esta dado por la última columna de $\mathbf{V}$.

Una de las propiedades de la matriz fundamental es su singularidad, de hecho como ya se ha mencionado tiene rango 2. Muchas aplicaciones de la matriz fundamental dependen de esta propiedad por lo cual debemos asegurarnos de que se cumpla. La matriz $\mathbf{F}$, obtenida como se describió anteriormente, no tiene rango 2 en general Una manera de forzar esta restricción es sustituyendo la matriz $\mathbf{F}$ por la matriz $\mathbf{F'}$ que minimice la norma de Frobenius $||\mathbf{F} - \mathbf{F'}||$ sujeta a la condición $\det \mathbf{F'} = 0$. Para realizar esto Tsai y Huang [Tsai and Huang, 1984] propusieron un método que minimiza la norma de Frobenius de $\mathbf{F} - \mathbf{F'}$ y consiste en lo siguiente: dada $\mathbf{F} = \mathbf{USV^T}$ donde $\mathbf{USV^T}$ es la descomposición SVD de $\mathbf{F}$ y $\mathbf{S} = \text{diag}(r, s, t)$ con $r \geq s \geq t$, entonces $\mathbf{F'}$ esta dada por $\mathbf{F'} = \mathbf{U}\text{diag}(r, s, 0)\mathbf{V^T}$.

La normalización fue propuesta por primera vez por Hartley (para una justificación de este procedimiento refiérase a [Hartley, 1997]). Esta normalización consiste en calcular transformaciones de similitud $\mathbf{T}$ y $\mathbf{T'}$ correspondientes a los puntos de cada imagen de tal manera que su centroide sea el origen coordenado y cuya distancia promedio del origen sea $\sqrt{2}$. Dichas transformaciones están dadas por:

$$\mathbf{T} = \begin{bmatrix} k & 0 & -k\bar{x} \\ 0 & k & -k\bar{y} \\ 0 & 0 & 1 \end{bmatrix} \qquad \mathbf{T'} = \begin{bmatrix} k' & 0 & -k'\bar{x}' \\ 0 & k' & -k'\bar{y}' \\ 0 & 0 & 1 \end{bmatrix}$$

donde

$$\bar{\mathbf{x}} = \begin{bmatrix} \bar{x} \\ \bar{y} \\ \bar{z} \end{bmatrix} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{x}_i, \qquad k = \frac{\sqrt{2}}{\frac{1}{N}\sum_{i=1}^{N} ||\mathrm{x}_i - \bar{\mathrm{x}}||}$$

$$\bar{\mathbf{x}}' = \begin{bmatrix} \bar{x}' \\ \bar{y}' \\ \bar{z}' \end{bmatrix} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{x}'_i, \qquad k' = \frac{\sqrt{2}}{\frac{1}{N}\sum_{i=1}^{N} ||\mathrm{x}'_i - \bar{\mathrm{x}}'||}$$

Se aplican las transformaciones $\mathbf{T}$ y $\mathbf{T}'$ para obtener los puntos normalizados de la siguiente manera: $\tilde{\mathbf{x}}_i = \mathbf{T}\mathbf{x}_i$ y $\tilde{\mathbf{x}}'_i = \mathbf{T}'\mathbf{x}'_i$.

Realizar la normalización antes de formular las ecuaciones lineales lleva a una mejora sustancial en el condicionamiento del problema y por lo tanto a la estabilidad del resultado. Una vez realizada la normalización de las correspondencias se sigue al cálculo de la matriz fundamental como se describió anteriormente. Luego para obtener la matriz $\mathbf{F}$ buscada tenemos que realizar la denormalización de la matriz $\hat{\mathbf{F}}$ encontrada de la siguiente manera: $\mathbf{F} = \mathbf{T}'^{T}\hat{\mathbf{F}}\mathbf{T}$.

3.5.2. RANSAC

En las sección anterior se propuso un método para calcular la matriz fundamental, este método asume que las correspondencias proporcionadas son correctas excepto por el error en las mediciones, el cual sigue una distribución Gaussiana. Si el conjunto de correspondencias utilizado esta contaminado incluso con un pequeño conjunto de *outliers* (correspondencias erróneas) el resultado de nuestro algoritmo probablemente sería inservible. El problema que tenemos ahora es el de clasificar nuestras correspondencias en *inliers* (correspondencias correctas) y *outliers*, el cual no es un problema fácil de resolver. Una solución general al problema de los *outliers* fue propuesto por [Fischler and Bolles, 1981]. Su algoritmo es llamado RANSAC (*Random Sample Consensus*) y puede ser aplicado a varios tipos de problemas.

Objetivo:
Dadas $n \geq 8$ correspondencias $\{\mathbf{x} \leftrightarrow \mathbf{x}'\}$, determinar la matriz $\mathbf{F}$ tal que ${\mathbf{x}'_i}^T \mathbf{F} \mathbf{x}_i = 0$.

Algoritmo:

1. **Normalización:** Transforma las coordenadas de acuerdo a $\hat{\mathbf{x}}_i = \mathbf{T}\mathbf{x}$ y $\hat{\mathbf{x}}'_i = \mathbf{T}'\mathbf{x}'$, donde $\mathbf{T}$ y $\mathbf{T}'$ son transformaciones de normalización que consisten en una traslación y un escalamiento.
2. Encuentra la matriz fundamental $\hat{\mathbf{F}}'$ con las correspondencias $\hat{\mathbf{x}} \leftrightarrow \hat{\mathbf{x}}'$:
   - **Solución lineal:** Determina $\hat{\mathbf{F}}$ del vector singular correspondiente al menor valor singular de la matriz $\hat{\mathbf{A}}$, la cual esta compuesta por las correspondencias $\hat{\mathbf{x}} \leftrightarrow \hat{\mathbf{x}}'$ de la manera descrita anteriormente.
   - **Restricción:** Sustituye $\hat{\mathbf{F}}$ por $\hat{\mathbf{F}}'$ tal que $\det \hat{\mathbf{F}}' = 0$ usando SVD.
3. **Denormalización:** $\mathbf{F} = {\mathbf{T}'}^T \hat{\mathbf{F}}' \mathbf{T}$. La matriz $\mathbf{F}$ es la matriz fundamental correspondiente a los datos originales $\{\mathbf{x} \leftrightarrow \mathbf{x}'\}$.

Cuadro 3.4: Algoritmo de los ocho puntos normalizado. (tomado de [Hartley and Zisserman, 2004])

El principio del algoritmo se basa en tomar aleatoriamente subconjuntos del total de datos (el tamaño del subconjunto depende del problema a resolver) y calcular soluciones con estos subconjuntos, cada una de estas soluciones parciales nos permite dividir el total de datos en *inliers* y *outliers*, este proceso se repite un N número de veces y se escoge la solución que proporcione más *inliers*.

A primera vista hay muchas preguntas sobre este algoritmo, tales como el número de iteraciones que se deben de realizar para asegurar que se haya seleccionado una buena solución, también cómo calcular la distancia que decide cuando un dato es considerado un *inlier* y cuando no, y cuándo se ha alcanzado un número suficiente de *inliers* que permita acortar el número de iteraciones del algoritmo.

Empecemos a responder estas preguntas. Es computacionalmente innecesario seleccionar todos los posibles subconjuntos, en lugar de eso el número N de muestras que se deben tomar se obtiene de manera que podamos asegurar con una probabilidad p

que al menos una de las muestras seleccionadas este libre de *outliers*. Suponiendo que w es la probabilidad de que un punto sea un *inlier*, definimos $\epsilon = 1 - w$ la probabilidad de que sea un *outlier*. Entonces el número necesario de pruebas esta dado por:

$$N = \log(1-p)/\log(1-(1-\epsilon)^s)$$

donde s representa el número de datos que se toman en cada muestra.

Queremos escoger la distancia t tal que con una probabilidad α el punto es un *inlier*, esta distancia en la práctica se escoge empíricamente, aunque, si asumimos que el error en las mediciones de los puntos sigue una distribución Gaussiana con desviación estándar σ, entonces es posible calcular un valor para t. Para una revisión de como el valor es obtenido consultar [Hartley and Zisserman, 2004]. En el caso del cálculo de la matriz fundamental esta distancia esta dada, para una probabilidad de 95 % de que el punto es un *inlier*, es decir, $\alpha = 0.95$, por $t = 1.96\sigma$ pixeles. En la práctica σ es difícil de calcular y se le puede asignar el valor de 0.5 o 1 pixel.

Una cosa que sucede en la mayoría de los casos es que desconocemos la fracción de *outliers* presentes en nuestro conjunto de datos (ϵ), por lo tanto no podemos determinar N. Para resolver este problema podemos empezar suponiendo el peor caso posible de ϵ y con ese valor calcular una N inicial, luego, si se encuentra una muestra cuya solución presenta una proporción de *outliers* menor que la estimada actualizamos N con ese nuevo valor. Este procedimiento de comparación de la ϵ se realiza en cada iteración y de esta manera el cálculo de N se realiza de forma dinámica. Si el nuevo valor de N es menor al número de iteraciones ya realizadas el algoritmo se detiene. El pseudocódigo de este procedimiento se encuentra descrito en el cuadro 3.5.

El último paso del RANSAC es volver a estimar el modelo usando todos los *inliers*. Esta nueva estimación debe ser óptima e involucra la minimización de una función de costo de máxima similitud (*MLE, Maximum Likelihood Estimation*), los métodos para realizar esta minimización son por lo general iterativos y la solución encontrada en el

- $N = \infty$, contadorMuestras = 0.
- Mientras $N >$ contadorMuestras, Repite
  - Escoge una muestra y cuenta el número de *inliers*.
  - Asigna $\epsilon = 1 -$ (numero *inliers*)/(total de puntos).
  - Actualiza $N = \log(1-p)/\log(1-(1-\epsilon)^s)$, con $p = 0.99$.
  - Incrementa el contadorMuestras en 1.
- Termina.

Cuadro 3.5: Actualización dinámica del número de muestras. (tomado de [Hartley and Zisserman, 2004])

paso anterior sirve como punto de partida de estos métodos. Con la solución obtenida por el método de minimización se reclasifican los datos en *inliers* y *outliers*, y se repite el proceso hasta que el número de *inliers* converja. El algoritmo para calcular la matriz fundamental de manera automática usando RANSAC se describe en el Cuadro 3.6, este algoritmo es el utilizado en este trabajo.

3.5.3. Medidas de Distancia

Distancia de Sampson

La distancia de Sampson se considera una aproximación de primer orden a la distancia geométrica. Fue usada originalmente por Sampson [Sampson, 1982] para ajustar cónicas. En el caso particular de estimar la matriz fundamental la función de costo esta dada por:

$$C = \sum_i \frac{(\mathbf{x}_i'^T \mathbf{F}\mathbf{x}_i)^2}{(\mathbf{F}\mathbf{x}_i)_1^2 + (\mathbf{F}\mathbf{x}_i)_2^2 + (\mathbf{F}^T\mathbf{x}_i')_1^2 + (\mathbf{F}^T\mathbf{x}_i')_2^2}$$

donde $(\mathbf{F}\mathbf{x}_i)_j^2$ representa el cuadrado de la j-ésima entrada del vector $\mathbf{F}\mathbf{x}_i$.

Distancia epipolar simétrica

La distancia epipolar simétrica se utiliza para minimizar la distancia que existe entre un punto y su línea epipolar calculada en las dos imágenes, esta distancia esta dada

por:

$$C = \sum_i d((x)'_i, \mathbf{F}\mathbf{x}_i)^2 + d((x)_i, \mathbf{F}^T\mathbf{x}'_i)^2$$

$$= \sum_i (\mathbf{x}_i'^T \mathbf{F}\mathbf{x}_i)^2 \left(\frac{1}{(\mathbf{F}\mathbf{x}_i)_1^2 + (\mathbf{F}\mathbf{x}_i)_2^2} + \frac{1}{(\mathbf{F}^T\mathbf{x}'_i)_1^2 + (\mathbf{F}^T\mathbf{x}'_i)_2^2} \right)$$

Objetivo:
Dado un par de imágenes en escala de grises, calcular la matriz fundamental **F** que define la geometría epipolar entre ellas de manera robusta.

Algoritmo:

1. **Puntos de interés:** Calcula los puntos de interés (esquinas) en cada imagen. (Harris).
2. **Correspondencias tentativas:** Calcula un conjunto de correspondencias entre los puntos de interés basadas en una medida de similitud (ZNCC, SSD).
3. **Estimación robusta, RANSAC:** Repite para N muestras, donde N se calcula dinámicamente.
   - Selecciona una muestra aleatoria de 8 correspondencias y calcula la matriz fundamental **F** mediante el algoritmo de los 8 puntos normalizado.
   - Calcula la distancia $d_\perp$ para cada correspondencia (la función de distancia utilizada es la distancia de Sampson).
   - Calcula el número de *inliers* consistentes con **F** (número de correspondencias que cumplen $d_\perp < t$, con $t = 1.96\sigma$).

   Escoge la **F** con el mayor número de *inliers*. En el caso de empate escoge la solución que tiene la menor desviación estándar de *inliers*.
4. **Minimización no lineal:** Vuelve a estimar **F** utilizando todas las correspondencias clasificadas como *inliers* minimizando una función de costo (distancia de Sampson), mediante el algoritmo de Levenberg-Marquardt ([Levenberg, 1944; Marquardt, 1963]), una implementación de este algoritmo se presenta en [Press et al., 1988].Este último punto se repite hasta que el número de correspondencias sea estable.

Cuadro 3.6: RANSAC para el cálculo de la matriz fundamental. (método descrito en [Hartley and Zisserman, 2004])

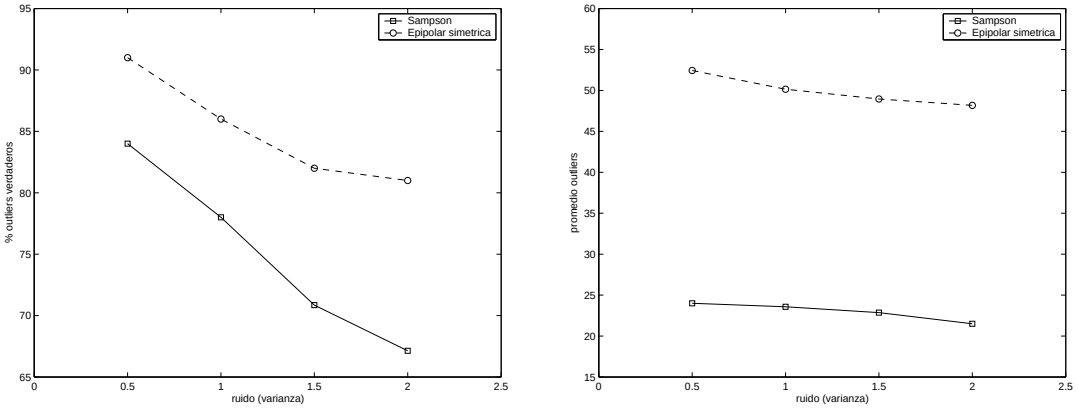
3.5.4. Búsqueda dirigida

Un paso más que se añade al final del algoritmo descrito en el Cuadro 3.6 (después de la minimización no lineal) es la búsqueda dirigida. La búsqueda dirigida consiste en usar la matriz fundamental **F** estimada para refinar nuestro conjunto de correspondencias. Como ya se ha visto en la sección 3.4, la matriz fundamental relaciona puntos en una imagen con sus respectivas lineas epipolares, gracias a esto el área en que se busca una correspondencia se reduce. Así para cada esquina **x** en una imagen se busca su correspondencia en la otra imagen dentro de una banda definida por su linea epipolar **Fx**. Debido a que el área de búsqueda es más precisa y menor podemos disminuir el umbral utilizado por la medida de similitud que se utiliza para comparar esquinas. El nuevo conjunto de correspondencias se reclasifica en *inliers* y *outliers* utilizando la misma función de distancia usada en el RANSAC.

3.5.5. Experimentos

Se realizaron varios experimentos para probar el funcionamiento del RANSAC para el cálculo de la matriz fundamental obteniendo en general buenos resultados. Para el primer experimento, cuyo objetivo es comparar las funciones de distancia en cuanto a su capacidad para clasificar *inliers* y *outliers* al aumentar el ruido en la posición de las coordenadas, se generaron 100 puntos tridimensionales aleatorios los cuales fueron proyectados a dos planos obteniendo así un conjunto de 100 correspondencias exactas, luego a estas correspondencias se les aumento ruido Gaussiano aleatorio con una varianza determinada ($\sigma^2 \in$ 0.5,1,1.5,2), se seleccionaron 20 de estas correspondencias de manera aleatoria y se les aumento ruido Gaussiano con $\sigma = 50$ (convirtiéndolos en *outliers*), con este conjunto de correspondencias como entrada se utilizó el RANSAC para obtener un promedio de *outliers* detectados y el porcentaje de estos que eran verdaderos. Este experimento se realizó 100 veces para cada varianza del ruido y para cada función de distancia, los parámetros del RANSAC fueron: $\epsilon = 0{,}40$, $t = 3{,}84\sigma^2$ y $p = 0{,}99$. Los resultados se observan en la Figura 3.8.

Otro experimento se realizó utilizando dos pares de imágenes correspondientes (uno de ellos sintético) para calcular el error RMS (*Root mean squared*) antes y después de



(a) (b)

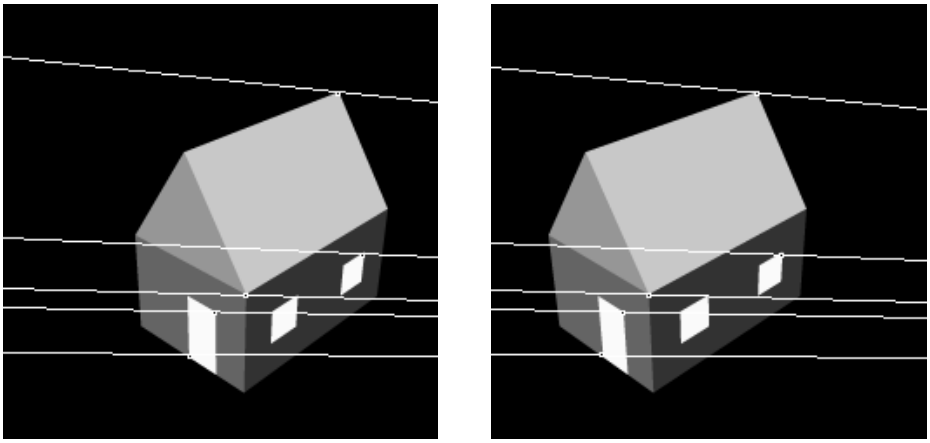
Figura 3.8: Comparación de medidas de distancia (Sampson y Epipolar simétrica). Se utilizó un conjunto de 100 correspondencias sintéticas de las cuales 20 eran *outliers*. El algoritmo del RANSAC se corrió 100 veces para cada varianza del ruido. (a) Porcentaje de *outliers* verdaderos detectados. (b) Promedio de *outliers* detectados.

la minimización no lineal y la búsqueda dirigida. Para el par de imágenes sintéticas (Figura 3.9a) se utilizó un umbral de $t = 1.25$ para el RANSAC y para la búsqueda dirigida se utilizó una ventana de comparación de 7x7 y una distancia a la línea epipolar de 1 pixel, el umbral para la medida de similitud fue de 0.5. En el caso de las imágenes naturales (Figura 3.9b) el único parámetro diferente fue el umbral de la medida de similitud para la búsqueda dirigida el cual fue de 0.8. En el Cuadro 3.7 se muestran los errores obtenidos para ambos pares.

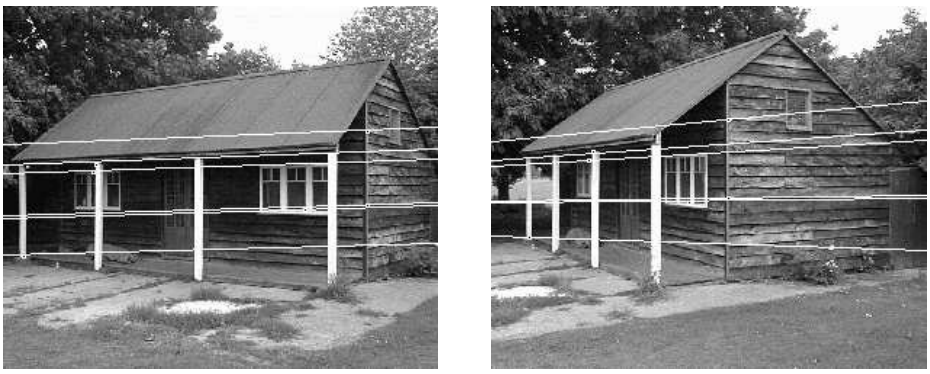
Imagen	Antes MLE	Después MLE
Artificial	0.1830	0.1014
Natural	0.2218	0.2061

Cuadro 3.7: Comparación de los errores RMS antes y después de la minimización no lineal y la búsqueda dirigida.

Por último la Figura 3.10 muestra los resultados obtenidos del cálculo automático de la matriz fundamental para el par de imágenes mostradas en la Figura 3.5. Se contaba con 245 correspondencias tentativas producto del método de correlación ZNCC. El umbral de los *inliers* fue $t = 1$ pixel. Se requirió un total de 192 muestras para durante el RANSAC y 3 iteraciones de la minimización no lineal y búsqueda dirigida. Para la minimización no lineal se utilizó el algoritmo de Levenberg-Marquardt. El error RMS después del RANSAC fue de 0.2309 (para 162 correspondencias) y después de la mini-



(a)

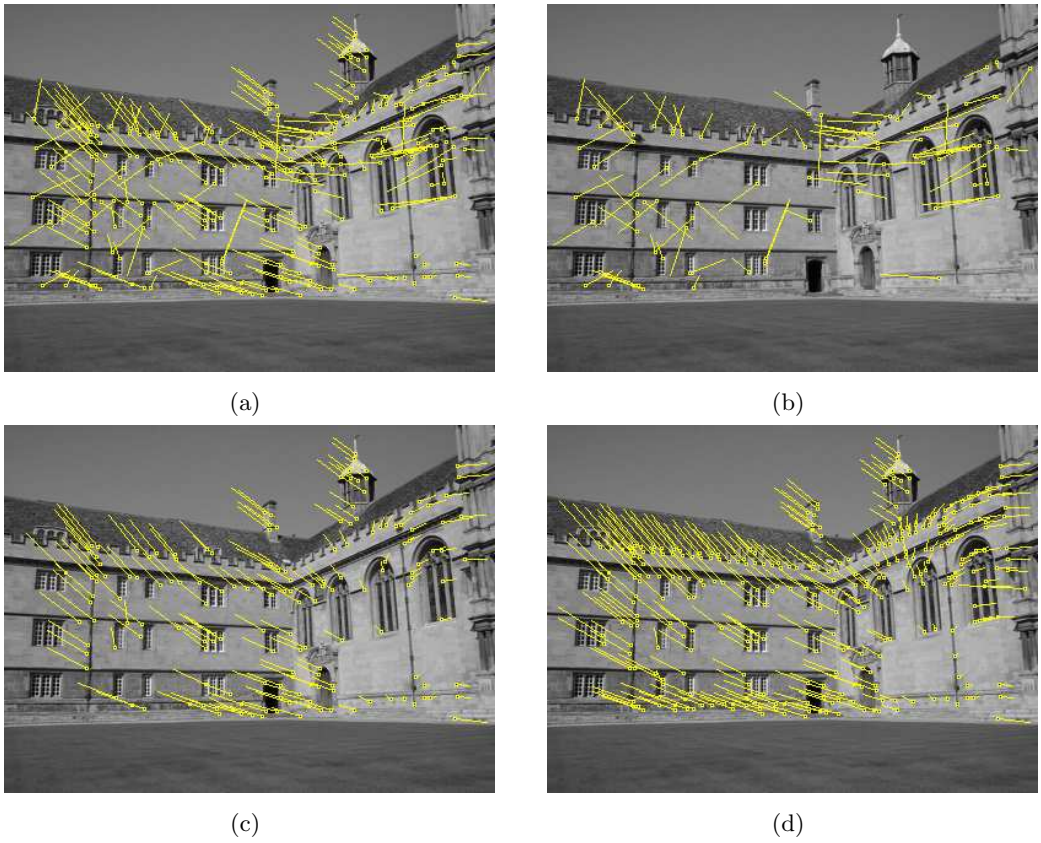


(b)

Figura 3.9: Pares de imágenes utilizadas para la comparación del uso de minimización no lineal. Las imágenes se muestran con algunas líneas epipolares superpuestas. (a) Imágenes Sintéticas (b) Imágenes Naturales

mización no lineal y la búsqueda dirigida fue de 0.1503 (para 303 correspondencias).

Se probaron minimizaciones con dos tipos de distancias: Sampson y epipolar simétrica. La distancia epipolar simétrica obtuvo buenos resultados con datos aleatorios, pero esto fue debido a que clasificaba en promedio la mitad de los puntos como *outliers*, pero al utilizarla con imágenes los resultados fueron inferiores a los obtenidos con la distancia de Sampson la cual se mostró más estable.



(a) (b)

(c) (d)

Figura 3.10: Cálculo automático de la matriz Fundamental usando RANSAC. Las imágenes son de 400x300 pixeles, correspondientes a las de la Figura 3.5 (a) 245 correspondencias tentativas (ZNCC). (b) 83 *outliers* detectados. (c) 162 *inliers* consistentes con la matriz F encontrada. (d) 303 correspondencias encontradas luego de la minimización no lineal y la búsqueda guiada.

3.6. Discusión

El objetivo de este capítulo fue encontrar la matriz fundamental entre dos imágenes, a partir de correspondencias entre ellas. Primero se implementó el método de Harris-Stephens para encontrar puntos característicos en las imágenes. No existe una regla para escoger el mejor umbral a utilizar, por lo que la elección del mismo se realiza de manera independiente para cada imagen. Luego las correspondencias tentativas se obtuvieron usando técnicas de auto-correlación, experimentando con dos medidas de similitud ZNCC y SSD. La medida que mejores resultados dio es el ZNCC ya que no sólo se comportó mejor en los experimentos sino que es más sencilla de interpretar. Por último se implementó el RANSAC para estimar la matriz fundamental.

Capítulo 4

Recuperación proyectiva de la estructura

4.1. Introducción

En el capítulo anterior se estimaron las matrices fundamentales entre las imágenes de la secuencia, estas matrices se usaron para realizar una búsqueda dirigida de correspondencias entre los puntos característicos de las imágenes. El objetivo de este capítulo es obtener la estructura proyectiva de la escena; esto consiste en estimar las matrices de proyección para cada una de las vistas, así como obtener un conjunto de puntos tridimensionales iniciales. Las matrices de proyección a su vez nos permitirán obtener nuevas correspondencias entre las imágenes, reduciendo aún más el espacio de búsqueda. La reconstrucción proyectiva obtenida en este capítulo difiere de la métrica por una homografía. El cálculo de dicha homografía es uno de los temas principales del siguiente capítulo.

En la sección 4.2 se establece un marco inicial para la estructura. El sistema coordenado de la reconstrucción estará en referencia a este marco. También se describe un método óptimo de triangulación. La sección 4.3 explica el proceso iterativo de añadir nuevas imágenes a la reconstrucción, esto incluye la estimación de la matriz de proyección para la nueva vista, y la triangulación de nuevos puntos tridimensionales. Al final de la sección se explica un método de refinamiento global de la estructura obtenida, usando toda la información estimada. Los experimentos realizados se muestran

en la sección 4.4 y una discusión sobre los resultados de las implementaciones y de los resultados obtenidos se encuentra en la sección 4.5.

4.2. Marco Inicial

El primer paso a seguir consiste en seleccionar dos imágenes de la secuencia que nos sirvan para inicializar nuestra estructura, ésta estructura inicial nos servirá de marco de referencia; las demás imágenes se irán añadiendo una a una y sus posiciones calculadas con respecto al marco de referencia inicial; este proceso se discutirá en la siguiente sección.

Para secuencias de vídeo (en las cuales no hay mucha variación entre las imágenes) se debe considerar una distancia mínima entre ellas para que la estructura inicial este bien condicionada. En nuestro caso no se tiene este problema ya que las imágenes obtenidas se han seleccionado de manera que no se encuentren muy cerca una de la otra y considerando también que una buena cantidad de puntos correspondientes puedan hallarse.

4.2.1. Obtención de las matrices de proyección

Luego de seleccionar las imágenes necesitamos obtener las matrices de proyección[1] de las vistas seleccionadas. Una matriz de proyección, es la expresión algebraica del mapeo entre puntos tridimensionales y puntos en el plano de la imagen. La proyección central será la empleada en este documento y uno de sus modelos básicos (*Pinhole*) se ilustra en la Figura 4.1.

La ecuación de proyección es:

$$\mathbf{x} \sim \mathbf{PX}$$

donde $\sim$ representa igualdad excepto por un factor de escala diferente de cero, $\mathbf{X}$ es un vector (4x1) que representa un punto tridimensional en coordenadas homogéneas,

[1] A la matriz de proyección también se le llama el modelo de la cámara ya que codifica los parámetros intrínsecos (foco, punto principal, etc) y extrínsecos (pose) de la misma.

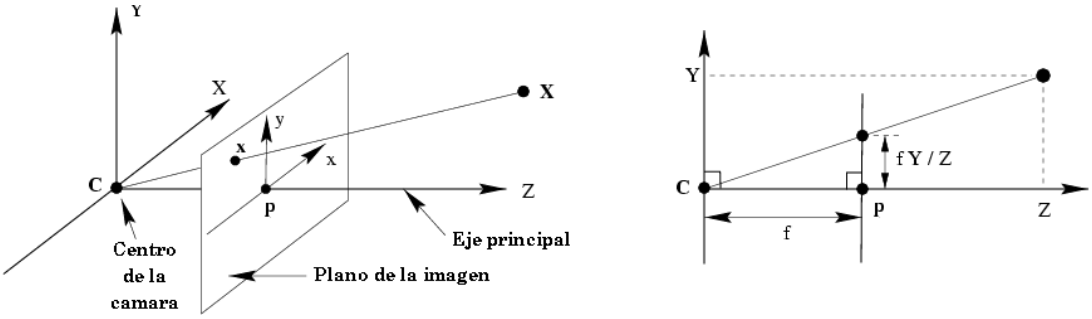


Figura 4.1: Geometría de la cámara *pinhole*. **C** es el centro de la cámara y **p** es el punto principal (intersección del eje principal con el plano de la imagen). Imagen tomada de [Hartley and Zisserman, 2004].

x es un vector (3x1) que representa el punto 2D correspondiente en la imagen y **P** es una matriz de proyección de 3x4.

La matriz de proyección para la primera imagen esta alineada con el sistema de coordenadas del mundo real, es decir, el origen del sistema coordenado esta ubicado en el centro de la cámara. Para calcular la matriz de proyección correspondiente a la segunda imagen utilizamos la matriz fundamental entre las dos imágenes (una descripción de los métodos para obtener la matriz fundamental fue dada en el Capítulo 2). Las matrices de proyección así obtenidas se conocen como *cámaras canónicas* y tienen la siguiente forma:

$$\begin{aligned} \mathbf{P}_1 &= [\mathbf{I}_{3x3}|0_3] \\ \mathbf{P}_2 &= [[\mathbf{e}_{12}]_x\mathbf{F}_{12} + \mathbf{e}_{12}\mathbf{a}^T|\sigma\mathbf{e}_{12}] \end{aligned}$$

donde $\mathbf{I}_{3x3}$ es una matriz identidad de 3x3, $\mathbf{F}_{12}$ es la matriz fundamental entre la imagen 1 y la 2, y $\mathbf{e}_{12}$ es el epipolo correspondiente al espacio nulo derecho de la matriz $\mathbf{F}_{12}$. Siguiendo las recomendaciones hechas en [Pollefeys et al., 2004], asignamos $\mathbf{a} = [0\,0\,0]^T$ y $\sigma = 1$.

La notación $[\mathbf{m}]_x$ para un vector $\mathbf{m} = (m_1, m_2, m_3)$ representa la matriz antisimétrica siguiente:

$$\begin{bmatrix} 0 & -m_3 & m_2 \\ m_3 & 0 & -m_1 \\ -m_2 & m_1 & 0 \end{bmatrix}$$

La multiplicación entre $[\mathbf{m}]_x$ y otro vector $\mathbf{n}$ es equivalente al producto cruz de ambos vectores.

4.2.2. Métodos de Triangulación

Una vez obtenidas las matrices de proyección el siguiente paso para la obtención de la estructura inicial es calcular la posición tridimensional de las correspondencias anteriormente calculadas utilizando estas matrices. Para realizar esto es necesario utilizar un método de triangulación. El concepto de triangulación puede observarse en la Figura 4.2.

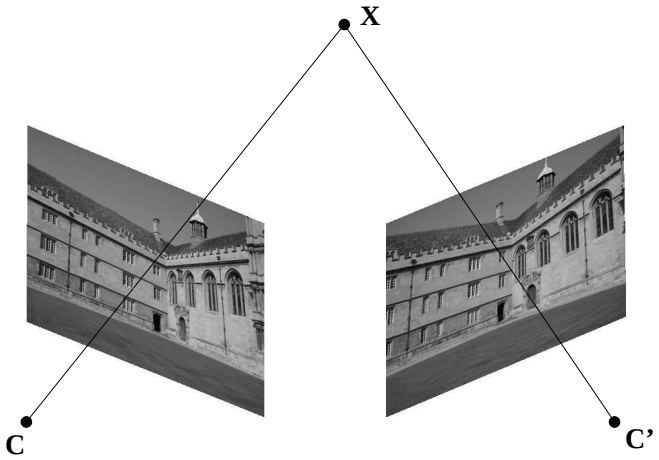


Figura 4.2: Triangulación de puntos correspondientes. $\mathbf{C}$ y $\mathbf{C}'$ son los centros de las cámaras y $\mathbf{X}$ es un punto tridimensional.

Se asume que existe cierto error en las correspondencias, este error se produce por varias razones, entre ellas, la mala localización del detector de esquinas. Debido a este error, la triangulación realizada simplemente proyectando hacia atrás los rayos que pasan por el centro de la cámara y un punto de la imagen fallaría ya que dichos rayos entre puntos correspondientes en general no se intersectarían en el espacio tridimensional. Por lo tanto debemos utilizar un mejor método para estimar la mejor solución posible para el punto tridimensional.

Triangulación Lineal

El método de triangulación lineal es relativamente simple y su derivación algebraica muy similar a la del algoritmo de los 8 puntos. Este método esta basado en la combinación de las ecuaciones de proyección $\mathbf{x} = \mathbf{PX}$ y $\mathbf{x}' = \mathbf{P}'\mathbf{X}$ para formar un sistema de ecuaciones de la forma $\mathbf{AX} = \mathbf{0}$. La matriz $\mathbf{A}$ esta dada por:

$$\mathbf{A} = \begin{bmatrix} x\mathbf{p}^{3T} - \mathbf{p}^{1T} \\ y\mathbf{p}^{3T} - \mathbf{p}^{2T} \\ x'\mathbf{p}'^{3T} - \mathbf{p}'^{1T} \\ y'\mathbf{p}'^{3T} - \mathbf{p}'^{2T} \end{bmatrix}$$

donde $\mathbf{p}^{iT}$ representa la $i-$ésima fila de la matriz de proyección $\mathbf{P}$, y $\mathbf{x} = (x, y, 1)$, $\mathbf{x}' = (x', y', 1)$ son las coordenadas homogéneas de puntos correspondientes.

Este sistema se puede resolver mediante SVD, encontrando el punto tridimensional $\mathbf{X}$ como el vector singular correspondiente al menor valor singular de $\mathbf{A}$.

Triangulación Óptima

Como se mencionó al principio de esta sección las correspondencias con que se cuenta son ruidosas, es decir, por lo general no cumplen de manera exacta con la restricción epipolar $\mathbf{x}'^T\mathbf{F}\mathbf{x} = 0$. Los correspondencias correctas son puntos cercanos a los calculados ($\mathbf{x}$,$\mathbf{x}'$) y que satisfacen la condición epipolar de manera exacta. Una manera de obtener una mejor estimación del punto tridimensional consiste en encontrar dichas correspondencias correctas y realizar la triangulación con ellas (Figura 4.3). Expresado de manera formal, lo que buscamos son puntos $\hat{\mathbf{x}}$ y $\hat{\mathbf{x}}'$ que minimicen la siguiente función:

$$C(\mathbf{x}, \mathbf{x}') = d(\mathbf{x}, \hat{\mathbf{x}})^2 + d(\mathbf{x}', \hat{\mathbf{x}}')^2, \quad \text{sujeto a} \quad \hat{\mathbf{x}}'^T\mathbf{F}\hat{\mathbf{x}} = 0$$

donde $d(m, n)$ representa la distancia Euclidiana entre dos puntos.

Una manera de realizar la triangulación minimizando el error geométrico $C(\mathbf{x}, \mathbf{x}')$ se describe en [Hartley and Zisserman, 2004]. En general este método consiste de las siguientes partes (una descripción detallada se muestra en el Cuadro 4.1):

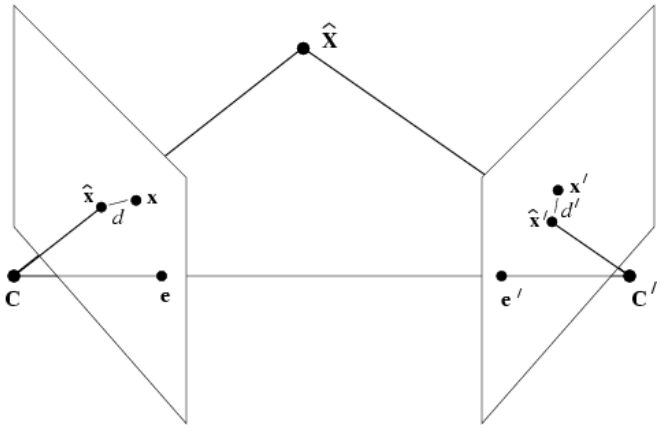


Figura 4.3: Minimización del error geométrico. Los puntos $\mathbf{x}$ y $\mathbf{x}'$ representan las correspondencias calculadas, $\hat{\mathbf{x}}$ y $\hat{\mathbf{x}}'$ son las correspondencias estimadas al minimizar el error geométrico y que satisfacen la restricción epipolar, por último $\hat{\mathbf{X}}$ representa el punto tridimensional calculado a partir de las correspondencias corregidas.

1. Parametrizar el conjunto de líneas epipolares en la primera imagen por un parámetro t. De esta manera una línea en la primera imagen puede escribirse como $\mathbf{l}(t)$.

2. Usando la matriz fundamental $\mathbf{F}$, calcular la correspondiente línea epipolar $\mathbf{l}'(t)$ en la segunda imagen.

3. Expresar la función de distancia $d(\mathbf{x}, \mathbf{l}(t))^2 + d(\mathbf{x}', \mathbf{l}'(t))^2$ explícitamente como una función de t. Encontrar el valor de t que minimiza la función.

4.3. Añadiendo una nueva vista

En esta sección se explicará cómo añadir una nueva vista a la reconstrucción obtenida en el paso anterior. Para cada nueva vista que se desee aumentar se sigue una serie de pasos que se describen a continuación.

4.3.1. Estimación de la nueva pose

Para añadir una nueva vista, el primer paso a seguir consiste en calcular la matriz de proyección correspondiente. Esta matriz se debe estimar con respecto al marco inicial definido anteriormente, es decir, la pose de la nueva vista se da, como se ha mencionado, respecto al sistema de coordenadas cuyo origen se encuentra en el centro de la primera

cámara (correspondiente a la primera vista en el marco inicial). Este proceso se ilustra en la Figura 4.4.

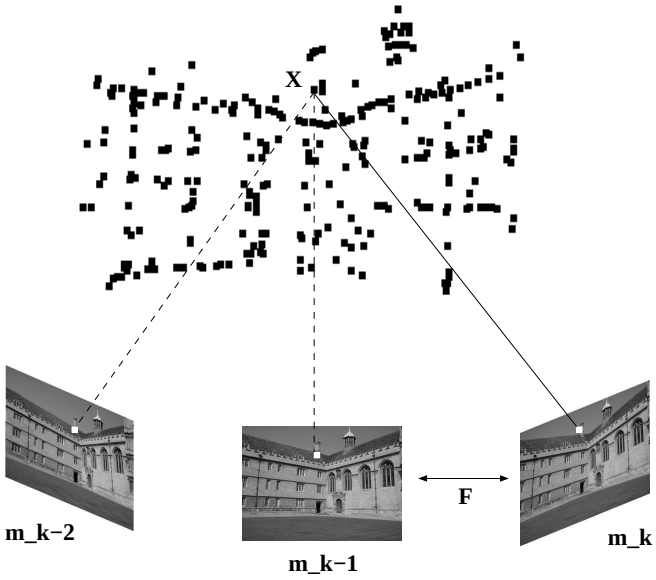


Figura 4.4: Estimación de la nueva pose. La pose de la nueva vista, m_k, se calcula a partir de las correspondencias con la vista anterior (m_{k-1}) y los puntos ya reconstruidos ($\mathbf{X}_i$)

El cálculo de la matriz de proyección, como se describe en [Pollefeys et al., 2004], se realiza de la siguiente manera:

1. Encontrar la geometría epipolar (matriz fundamental) entre la nueva vista m_k y la vista anterior m_{k-1}. Con esto se cuenta con un conjunto de correspondencias robusto entre ambas vistas (imágenes).

2. Entre las correspondencias calculadas existen puntos en la vista que m_{k-1} que ya han sido reconstruidos, estos puntos se utilizan para generar un conjunto de correspondencias entre puntos tridimensionales y puntos 2D en la nueva vista.

3. Se utiliza un proceso similar al RANSAC para el cálculo de la matriz fundamental para estimar la matriz de proyección $\mathbf{P}_k$. Un método lineal para estimar la matriz de proyección a partir 6 correspondencias se describe en el Apéndice B.

4. La matriz $\mathbf{P}_k$ estimada puede ser utilizada para proyectar puntos ya reconstrui-

dos, obteniendo así correspondencias adicionales que serán utilizadas para refinar $\mathbf{P}_k$ (esto es el equivalente a la búsqueda guiada en el caso de la estimación de la matriz fundamental).

4.3.2. Refinando y añadiendo puntos tridimensionales

El paso a seguir es *refinar la estructura*, es decir, estimar nuevamente la posición de los puntos tridimensionales ya reconstruidos, a partir de su posición actual y la información proporcionada por la nueva vista. Entre las formas de realizar la estimación se encuentra el uso del filtro de Kalman [Pollefeys, 2000] (para una introducción a la teoría del filtro de Kalman consultar [Welch and Bishop, 1995]), el filtro de Kalman es un método iterativo que se basa en el uso de matrices de covarianza para estimar parámetros a partir de información no confiable; este filtro ha sido utilizado en ingeniería, control, y navegación, entre otros campos así como también en visión computacional. En este documento se utilizará un método lineal más sencillo descrito en [Pollefeys et al., 2004], el cual se describe a continuación.

Basándonos en la ecuación $\mathbf{x} = \mathbf{PX}$ podemos derivar el sistema de ecuaciones siguiente:

$$\begin{aligned} \mathbf{P}_3\mathbf{X}x - \mathbf{P}_1\mathbf{X} &= 0 \\ \mathbf{P}_3\mathbf{X}y - \mathbf{P}_2\mathbf{X} &= 0 \end{aligned}$$

donde $\mathbf{P}_i$ representa la $i-$ésima fila de la matriz $\mathbf{P}$ y (x, y) son las coordenadas del punto en la imagen. Podemos encontrar un estimado del punto tridimensional resolviendo el sistema de ecuaciones formado de todas las vistas donde las coordenadas en la imagen del punto tridimensional están disponibles. Una recomendación de Pollefeys para obtener un mejor resultado consiste en minimizar la siguiente función de distancia $\sum d(\mathbf{PX}, \mathbf{x})^2$, esto se puede aproximar resolviendo de manera iterativa (mediante SVD) el siguiente sistema de ecuaciones:

$$\frac{1}{\mathbf{P}_3\tilde{\mathbf{X}}} \begin{bmatrix} \mathbf{P}_3x - \mathbf{P}_1 \\ \mathbf{P}_3y - \mathbf{P}_2 \end{bmatrix} \mathbf{X} = 0$$

donde $\tilde{\mathbf{X}}$ representa la solución previa de $\mathbf{X}$.

Es muy posible que en cada nueva imagen aparezcan puntos que no se encontraban en las anteriores o que se cuenta con la posición del punto correspondiente en la imagen anterior pero que no han sido reconstruidos. Si dichos puntos existen, se utilizan para inicializar nuevos puntos tridimensionales (esto se realiza como el método de triangulación descrito en la sección 4.2.2). Para que un punto reconstruido sea tomado en cuenta, es necesario que sea observado desde un número mínimo de vistas; debido a que en los experimentos realizados en este trabajo se utilizaran secuencias formadas de pocas imágenes, por lo general entre 5 y 7, éste mínimo será de tres imágenes. Esta restricción nos permite eliminar algunas correspondencias erróneas y hace más robusto nuestro modelo. Después de seguir estos pasos para todas las imágenes, contamos ahora con las matrices de proyección para todas las vistas y un conjunto de puntos tridimensionales reconstruidos.

4.3.3. Refinamiento global de la estructura

Una forma de lograr una mejor estimación, tanto de las matrices de proyección como de los puntos tridimensionales, es realizar una estimación de máxima similitud global. Esta estimación global se obtiene mediante un método denominado ajuste de haz (*bundle adjustment*). El objetivo es encontrar los parámetros de las matrices de proyección $\mathbf{P}_i$ y los puntos tridimensionales $\mathbf{X}_j$ que minimicen el error entre los puntos observados en las imágenes $\mathbf{x}_{ij}$ y los puntos reproyectados $\mathbf{P}_i(\mathbf{X}_j)$. Información detallada sobre este método puede encontrarse en [Triggs et al., 2000; Pollefeys, 2000]. Para m vistas y n puntos la función a minimizar esta dada por:

$$\min_{\mathbf{P}_i,\mathbf{X}_j} = \sum_{i=1}^{m}\sum_{j=1}^{n} d(\mathbf{m}_{ij}, \mathbf{P}_i(\mathbf{X}_j))^2$$

4.4. Experimentos

Se realizaron experimentos con diferentes secuencias de imágenes. A manera de seguimiento de los experimentos realizados en el capítulo anterior a continuación se presenta uno de dichos experimentos para el cual se utilizó la secuencia de cinco imágenes mos-

trada en la Figura 4.5. Las imágenes tienen una resolución de 1024 x 768 pixeles. Se encontraron aproximadamente 1500 esquinas en cada imagen. Al final del RANSAC para la estimación de las matrices fundamentales se contaba con un promedio de 800 correspondencias entre cada una de las imágenes. Las primeras dos imágenes de la secuencia (Figura 4.5) se utilizaron como marco de referencia. Luego se fueron añadiendo una a una las imágenes restantes. El método de triangulación óptimo descrito en el Cuadro 4.1 se seleccionó para este experimento, aunque también se realizaron experimentos utilizando sólo el método lineal.

Figura 4.5: Secuencia de imágenes utilizada para los experimentos.

El umbral usado dentro del RANSAC en el cálculo de las matrices de proyección fue de 5.99. Para la implementación del ajuste de haz (*bundle adjustment*) se utilizaron las bibliotecas de funciones proporcionadas por [Lourakis and Argyros, 2004]. Se seleccionaron los puntos tridimensionales cuya proyección se había localizado por lo menos en 3 de las 5 imágenes. Al final del proceso se obtuvieron 1017 puntos tridimensionales.

Algunas vistas de la reconstrucción proyectiva de los puntos característicos pueden observarse en la Figura 4.6. Los efectos de la distorsión perspectiva son claramente visibles (como la longitud de la pared izquierda del edificio), otra característica notoria es el ángulo entre las paredes del edificio que se aprecia en la vista superior (Figura

4.6d), estas paredes, en el edificio real, forman un ángulo recto.

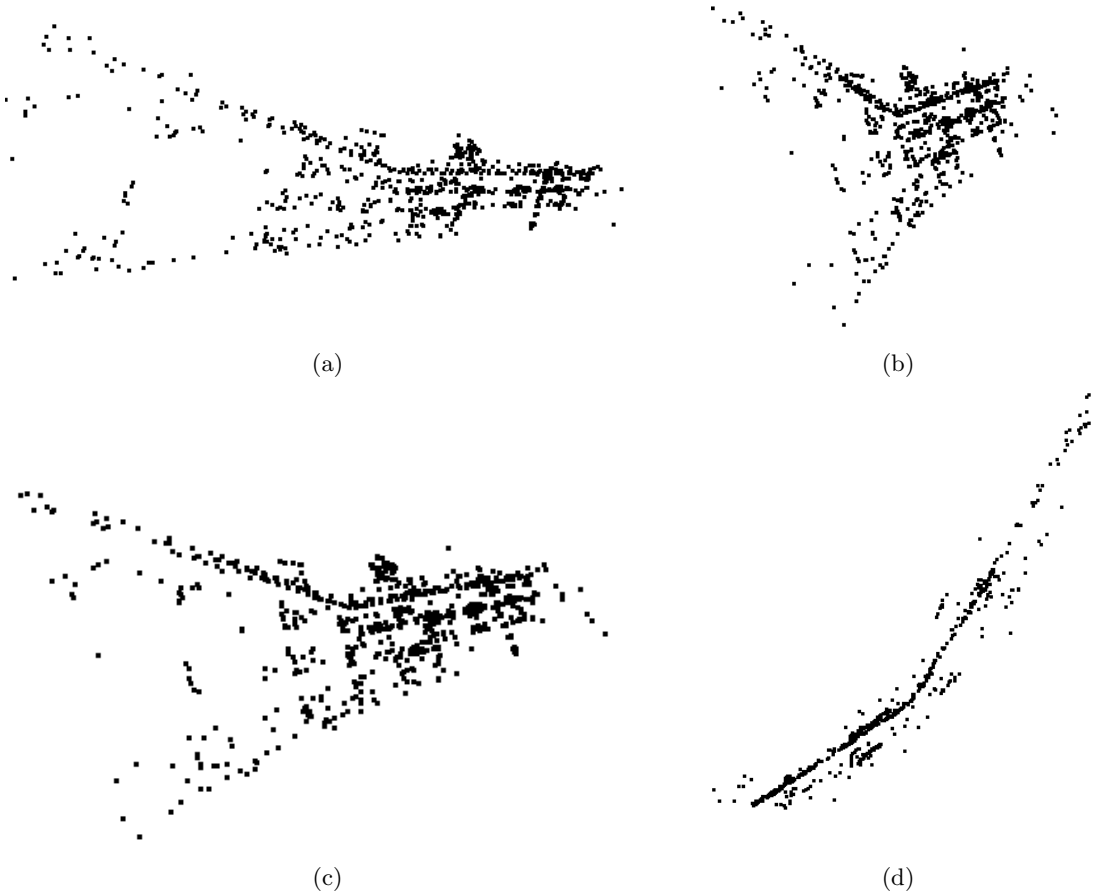


Figura 4.6: Algunas vistas de la reconstrucción proyectiva de los puntos característicos. (a),(b) y (c) Vistas Frontales. (d) Vista superior.

Aunque se experimentó con ambos métodos de triangulación, el método óptimo dio mejores resultados para la reconstrucción proyectiva descrita en este capítulo. Se debe tener cuidado siempre que se realiza una búsqueda guiada con el objetivo de encontrar correspondencias adicionales, tanto en el caso de la matriz fundamental como con las matrices de proyección, que las correspondencias entre imágenes siempre se den uno a uno, es decir, que un punto característico en una imagen sólo se relacione con uno en otra vista. Igualmente el umbral de la medida de similitud para la búsqueda guiada de correspondencias con las matrices de proyección debe ser menor que en las fases anteriores ya que el área de búsqueda se reduce aun más por lo que el número de posibles correspondencias disminuye. (en el caso de nuestros experimentos se redujo a una ventana de 5x5).

4.5. Discusión

A lo largo de este capítulo se proporcionaron métodos para realizar la reconstrucción proyectiva de una escena. Primero se seleccionaron dos imágenes para inicializar el marco de referencia. La selección de este par inicial es importante ya que no cualquier par de imágenes de la secuencia se puede elegir, debe existir una distancia adecuada entre las vistas, así como suficientes correspondencias entre ellas, de manera que el marco inicial este bien condicionado. Una vez determinado esto, se definen las matrices de proyección para las dos vistas y se procede a la triangulación de los puntos tridimensionales.

Después de establecer el marco de referencia inicial, las demás vistas se fueron añadiendo una por una y se estimó para cada una de ellas su respectiva matriz de proyección. Dichas matrices se utilizaron para encontrar más correspondencias entre las imágenes. En el experimento mostrado en la sección anterior, el método de ajuste de haz, utilizado como paso final para realizar un refinamiento global de todos los datos estimados, no produjo grandes variaciones a los resultados obtenidos antes de éste. Aunque se recomienda siempre su uso, y más cuando se tiene una secuencia de imágenes irregular.

Objetivo:
Dado un conjunto de correspondencias $\mathbf{x} \leftrightarrow \mathbf{x}'$, y una matriz fundamental $\mathbf{F}$ calcular las correspondencias corregidas $\hat{\mathbf{x}} \leftrightarrow \hat{\mathbf{x}}'\}$ que minimicen el error geométrico $C(\mathbf{x}, \mathbf{x}') = d(\mathbf{x}, \hat{\mathbf{x}})^2 + d(\mathbf{x}', \hat{\mathbf{x}}')^2$ sujetas a la restricción epipolar $\hat{\mathbf{x}}'^T \mathbf{F} \hat{\mathbf{x}}$.

Algoritmo:

1. Define matrices de transformación

$$\mathbf{T} = \begin{bmatrix} 1 & & -x \\ & 1 & -y \\ & & 1 \end{bmatrix}, \qquad \mathbf{T}' = \begin{bmatrix} 1 & & -x' \\ & 1 & -y' \\ & & 1 \end{bmatrix}$$

   Estas son las traslaciones que llevan $\mathbf{x} = (x, y, 1)$ y $\mathbf{x}' = (x', y', 1)$ al origen.

2. Sustituye $\mathbf{F}$ por $\mathbf{T}'^{-T}\mathbf{F}\mathbf{T}^{-1}$. La nueva $\mathbf{F}$ corresponde a las coordenadas trasladadas.

3. Calcula los epipolos $\mathbf{e} = (e_1, e_2, e_3)^T$ y $\mathbf{e}' = (e'_1, e'_2, e'_3)^T$, tal que $\mathbf{e}'\mathbf{F} = 0$ y $\mathbf{F}\mathbf{e} = 0$. Normaliza $\mathbf{e}$ tal que $e_1^2 + e_2^2 = 1$ y haz lo mismo para $\mathbf{e}'$.

4. Forma las matrices

$$\mathbf{R} = \begin{bmatrix} e_1 & e_2 & \\ -e_2 & e_1 & \\ & & 1 \end{bmatrix}, \qquad \mathbf{R}' = \begin{bmatrix} e'_1 & e'_2 & \\ -e'_2 & e'_1 & \\ & & 1 \end{bmatrix}$$

   y observa que $\mathbf{R}$ y $\mathbf{R}'$ son matrices de rotación, y que $\mathbf{R}\mathbf{e} = (1, 0, e_3)^T$ y $\mathbf{R}'\mathbf{e}' = (1, 0, e'_3)^T$.

5. Sustituye $\mathbf{F}$ por $\mathbf{R}'\mathbf{F}\mathbf{R}^T$.

6. Asigna $f = e_3$, $f' = e'_3$, $a = F_{22}$, $b = F_{23}$,$c = F_{32}$,$d = F_{33}$.

7. Forma el polinomio $g(t)$ y resuélvelo para obtener 6 raíces.

$$g(t) = t((at + b)^2 + f'^2(ct + d)^2)^2 - (ad - bc)(1 + f^2t^2)^2(at + b)(ct + d) = 0$$

8. Evalúa la función de costo $s(t)$ con la parte real de las raíces de $g(t)$, también calcula el valor para $t = \infty$ dado por $1/f^2 + c^2/(a^2 + f'^2c^2)$. Selecciona el valor t_{min} de t que de el valor más pequeño de la función de costo.

$$s(t) = \frac{t^2}{1 + f^2t^2} + \frac{(ct + d)^2}{(at + b)^2 + f'^2(ct + d)^2}$$

9. Evalúa las dos líneas $\mathbf{l} = (tf, 1, -t)$ y $\mathbf{l}' = (-f'(ct+d), at+b, ct+d)$ en t_{min} y encuentra $\hat{\mathbf{x}}$ y $\hat{\mathbf{x}}'$ como los puntos más cercanos al origen sobre estas líneas. Para una línea en general (λ, μ, ν) el punto más cercano de la línea al origen esta dado por $(-\lambda\nu, -\mu\nu, \lambda^2 + \mu^2)$.

10. Transfiere las coordenadas estimadas a las originales sustituyendo $\hat{\mathbf{x}}$ por $\mathbf{T}^{-1}\mathbf{R}^T\hat{\mathbf{x}}$ y $\hat{\mathbf{x}}'$ por $\mathbf{T}'^{-1}\mathbf{R}'^T\hat{\mathbf{x}}'$

11. El punto tridimensional $\hat{\mathbf{X}}$ puede obtenerse con el método homogéneo descrito anteriormente.

Cuadro 4.1: Método óptimo de triangulación. (tomado de [Hartley and Zisserman, 2004])

Capítulo 5

Reconstrucción Métrica

5.1. Introducción

La reconstrucción obtenida hasta este punto es una reconstrucción proyectiva. Dicha reconstrucción no preserva líneas paralelas ni ángulos rectos, entre otras características, de los objetos presentes en la escena. Uno de los objetivos de este capítulo es encontrar la transformación que convierta una reconstrucción proyectiva a una métrica. Por otro lado se llega al resultado final de esta tesis, el cual consiste en obtener una reconstrucción densa (nube de puntos tridimensionales) que sólo difiere de la escena real por un movimiento rígido y un factor de escala.

En la sección 5.2 se describe un método de auto-calibración, lo que nos permite alcanzar una reconstrucción métrica. Este método esta basado en la cónica absoluta. Una breve introducción a las cónicas se presenta al principio de esta sección. La sección 5.3 muestra un método sencillo de rectificación epipolar para pares de imágenes, el cual permite simplificar los algoritmos de búsqueda densa de correspondencias. Finalmente en la sección 5.4 se utiliza un método de correlación junto con una estrategia de programación dinámica para realizar la estimación densa de la superficie. Cada sección cuenta con su propio apartado de experimentos.

5.2. Auto-Calibración

Existen varios métodos para obtener la calibración de la cámara (estimar sus parámetros intrínsecos). En muchas ocasiones la cámara es calibrada antes de tomar las fotografías o la secuencia de vídeo, utilizando algún objeto de calibración conocido (como el tablero que se muestra en la Figura 5.1). En nuestro caso no utilizaremos este tipo de métodos sino un método de auto-calibración. Se conoce como auto-calibración al proceso mediante el cual obtenemos los parámetros internos de la cámara basándonos únicamente en un conjunto de imágenes no calibradas.

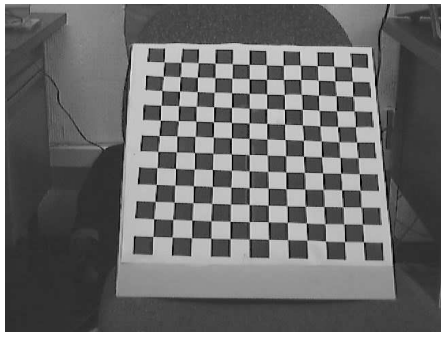


Figura 5.1: Objeto de calibración.

Aprovecharemos el hecho de que contamos con una reconstrucción proyectiva de la escena (capítulo anterior), es decir, tenemos un conjunto de puntos tridimensionales $\mathbf{X}_i$ y un conjunto de matrices de proyección $\mathbf{P}_k$ que difieren del modelo Euclidiano por una transformación proyectiva. En un marco Euclidiano una matriz de proyección se descompone $\mathbf{P} = \mathbf{KR}[\,\mathbf{I}\,|\,\mathbf{t}\,]$, donde $\mathbf{K}$ representa la matriz de calibración que contiene los parámetros intrínsecos de la cámara, $\mathbf{R}$ es una matriz de rotación y $\mathbf{t}$ es un vector de traslación. Como las matrices de proyección se obtuvieron en un marco proyectivo no se descompondrán de esa manera. Asumimos que los parámetros internos de la cámara son constantes para todas las vistas, por lo tanto existe una homografía $\mathbf{H}$ que transforma las matrices de proyección, hasta ahora obtenidas, de manera tal que:

$$\mathbf{P}_i\mathbf{H} = \mathbf{KR}_i[\,\mathbf{I}\,|\,\mathbf{t}_i\,].$$

$$\mathbf{K} = \begin{bmatrix} \alpha_x & s & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix}.$$

donde α_x y α_y representan la distancia focal de la cámara en términos de las dimensiones de los pixeles en las direcciones x y y respectivamente; s es la oblicuidad (*skew*) de los pixeles y (x_0, y_0) son las coordenadas del punto principal que también se encuentran en términos de las dimensiones de los pixeles. Esta misma homografía se puede aplicar a los puntos tridimensionales ($\mathbf{H}^{-1}\mathbf{X}_j$) para obtener una reconstrucción métrica. El método elegido esta basado en la cónica absoluta (AC, *Absolute conic*) y su proyección en cada una de las imágenes, ya que, bajo trasformaciones de similitud[1], la cónica absoluta permanece fija. A continuación se presenta una descripción de la cónica absoluta y su relación con los parámetros intrínsecos de la cámara.

5.2.1. Cónicas y Cuádricas

Una cónica, como se conoce generalmente, es una curva descrita por una ecuación de segundo grado en el plano (parábolas, círculos, elipses, etc.). En un espacio proyectivo una cónica se representa con una matriz simétrica homogénea $\mathbf{C}$ de 3x3 , que cumple $\mathbf{x}^T\mathbf{C}\mathbf{x} = 0$, para todos los puntos homogéneos $\mathbf{x} = (x_1, x_2, x_3)$ que pertenecen a dicha cónica. La razón de la expresión anterior se comprende de manera más clara si definimos la cónica como:

$$\mathbf{C} = \begin{bmatrix} a & b/2 & d/2 \\ b/2 & c & e/2 \\ d/2 & e/2 & f \end{bmatrix}$$

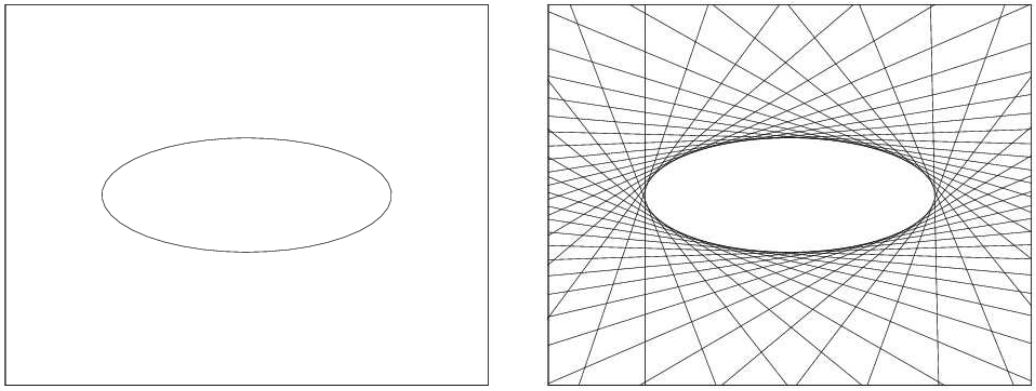
y utilizamos coordenadas inhomogéneas para los puntos $\mathbf{x} = (x, y, 1)$. De esta manera, desarrollando $\mathbf{x}^T\mathbf{C}\mathbf{x} = 0$ obtenemos:

$$ax^2 + bxy + cy^2 + dx + ey + f = 0$$

que es la ecuación general de una cónica en geometría analítica.

[1] Rotación, traslación y escalamiento

La cónica definida anteriormente es una cónica de puntos. Existe una dual a esta cónica denominada envoltorio cónico (*conic envelope*), $\mathbf{C}^*$, definida por las líneas tangentes a la cónica en lugar que por los puntos pertenecientes a ella. Por lo tanto una línea $\mathbf{l}$ tangente a la cónica $\mathbf{C}$ cumple $\mathbf{l}^T\mathbf{C}^*\mathbf{l} = 0$. Ejemplos de dichas cónicas se pueden ver en la Figura 5.2.



(a) (b)

Figura 5.2: Cónicas. (a) Cónica de puntos. (b) Envoltorio cónico.

Una cuádrica es una superficie en un espacio proyectivo tridimensional (esferas, hiperboloides, elipsoides, etc.). Está definida por la ecuación $\mathbf{X}^T\mathbf{Q}\mathbf{X} = 0$, donde $\mathbf{Q}$ es una matriz simétrica homogénea de 4x4 y $\mathbf{X}$ representa un punto tridimensional en coordenadas homogéneas. Al igual que con las cónicas las cuádricas tienen duales. Una cuádrica dual ($\mathbf{Q}^*$) esta definida por los planos π tangentes a la cuádrica $\mathbf{Q}$ y cumple $\pi^T\mathbf{Q}^*\pi = 0$.

Bajo una trasformación de puntos $\mathbf{X}' = \mathbf{H}\mathbf{X}$, una cuádrica dual se transforma de la siguiente manera:

$$\mathbf{Q}^{*'} = \mathbf{H}\mathbf{Q}^*\mathbf{H}^T.$$

Para un tratamiento más extensivo de este tema se recomienda consultar [Hartley and Zisserman, 2004], [Forsyth and Ponce, 2003] y [Semple and Kneebone, 1952].

5.2.2. La cónica Absoluta

La cónica absoluta es uno de los conceptos más importantes dentro de los métodos de auto-calibración, debido a que su posición relativa con respecto a una cámara en movimiento no cambia ya que es invariante ante transformaciones de similitud. Se puede considerar a la cónica absoluta como un objeto de calibración presente en todas las escenas. Localizándola podemos transformar nuestra reconstrucción de proyectiva a métrica.

En términos formales, la cónica absoluta es una cónica de puntos denominada por $\mathbf{\Omega}_\infty$ que se encuentra en el el plano al infinito (π_∞) y está formada solamente por puntos imaginarios. En un marco métrico las coordenadas del plano al infinito son $\pi_\infty = (0,0,0,1)^T$. Una forma de representar esta cónica es mediante la cuádrica absoluta dual (DAQ, *Dual absolute quadric*), denotada por $\mathbf{\Omega}^*_\infty$, formada por los planos tangentes a la cónica absoluta. La DAQ encapsula información tanto de la cónica absoluta como del plano al infinito, de hecho, el plano al infinito corresponde al espacio nulo de $\mathbf{\Omega}^*_\infty$. En un espacio Euclidiano la DAQ esta dada por $\mathbf{\Omega}^*_\infty = \text{diag}(1,1,1,0)$ y, al igual que la cónica absoluta, es invariante ante transformaciones de similitud.

La cuádrica absoluta dual es una matriz simétrica homogénea de 4x4 y de rango 3. La proyección de la DAQ en plano de la imagen esta dada por

$$\omega^* \sim \mathbf{P}\mathbf{\Omega}^*_\infty\mathbf{P}^T. \tag{5.1}$$

Como se ha mencionado, en un marco Euclidiano una matriz de proyección se descompone $\mathbf{P} = \mathbf{K}\mathbf{R}[\,\mathbf{I}\,|\,\mathbf{t}\,]$ y $\mathbf{\Omega}^*_\infty = \text{diag}(1,1,1,0)$, sustituyendo esto en la ecuación 5.1 obtenemos

$$\omega^* \sim \mathbf{K}\mathbf{K}^T. \tag{5.2}$$

Donde $\mathbf{K}$ es la matriz de calibración de la cámara. De esta manera establecemos una relación entre los parámetros intrínsecos de la cámara y la cuádrica absoluta dual.

Cuando nos encontramos en un marco proyectivo la representación de la cuádrica absoluta dual no será la misma que en el marco Euclidiano, sino que tendrá la forma general $\mathbf{\Omega}^*_\infty = \mathbf{T}\mathbf{\Omega}^*_M\mathbf{T}^T$, donde $\mathbf{\Omega}^*_M$ es la representación Euclidiana de la DAQ. Como las imágenes son independientes de la base proyectiva de la reconstrucción la ecuación 5.2 siempre es válida.

5.2.3. Descripción del método

Existen varios métodos que utilizan la cónica absoluta como base para la autocalibración, entre los más conocidos se encuentra el propuesto por [Trigss, 1997]. En [Hartley and Zisserman, 2004] se presentan métodos tanto lineales como no lineales. Una primera aproximación al método presentado en esta sección se dio en [Pollefeys et al., 1998]. Mejoramientos a este método, algunos de los cuales serán utilizados, fueron hechos en [Pollefeys, 2000; Pollefeys et al., 2004]. El método que se presentará a continuación, consiste en dos fases: primero se resuelve un sistema de ecuaciones lineales; el resultado de esta primera fase sirve como solución inicial para un refinamiento global de mínimos cuadrados no lineales.

La primera etapa se basa en realizar restricciones lineales a los parámetros internos de la cámara. Dichas restricciones se transfieren a la cuádrica absoluta dual mediante la ecuación 5.1. Las restricciones se establecen en la parte izquierda de dicha ecuación cuya forma desarrollada es:

$$\omega^* \sim \mathbf{K}\mathbf{K}^T = \begin{bmatrix} \alpha_x^2 + s^2 + x_0^2 & s\alpha_y + x_0 y_0 & x_0 \\ s\alpha_y + x_0 y_0 & \alpha_y^2 + y_0^2 & y_0 \\ x_0 & y_0 & 1 \end{bmatrix}. \tag{5.3}$$

Por ejemplo, si conocemos las coordenadas del punto principal tenemos 2 restricciones por cada imagen, ya que el punto principal puede ser trasladado al origen y por lo tanto $x_0 = y_0 = 0$, generando las ecuaciones $\mathbf{P}^{(1)}\mathbf{\Omega}^*_\infty\mathbf{P}^{(3)^T} = 0$ y $\mathbf{P}^{(2)}\mathbf{\Omega}^*_\infty\mathbf{P}^{(3)^T} = 0$, donde $\mathbf{P}^{(i)}$ representa la $i-$ésima fila de la matriz de proyección $\mathbf{P}$.

En este trabajo se supone que el punto principal se encuentra en el centro de la

imagen, la oblicuidad (*skew*) es cero y la razón entre las dimensiones de los pixeles es uno (por lo tanto $\alpha_x = \alpha_y$), esto genera un total de 4 restricciones por imagen. Como la cuádrica absoluta dual tiene 9 grados de libertad[2], un total de 3 imágenes es suficiente para conseguir el mínimo de nueve ecuaciones. En nuestra implementación se obtendrán 4 ecuaciones para cada imagen de la secuencia, generando de esta manera un sistema sobre-determinado de ecuaciones.

Antes de generar las ecuaciones, las matrices de proyección se normalizan de manera que cumplan con las restricciones propuestas. Esta normalización se propone en [Pollefeys et al., 2004] y consiste en lo siguiente:

$$\mathbf{P}_N = \mathbf{K}_N^{-1}\mathbf{P}$$

$$\mathbf{K}_N = \begin{bmatrix} w+h & 0 & w/2 \\ & w+h & h/2 \\ & & 1 \end{bmatrix}$$

donde w y h representan las dimensiones de la imagen (ancho y alto respectivamente). Con esta normalización el punto principal es trasladado cerca del origen y la distancia focal es escalada (una distancia focal de 60mm es escalada a 1). Como consideramos que la oblicuidad es cero y la razón de aspecto de los pixeles es uno, obtenemos las siguientes 4 ecuaciones por cada imagen:

$$\begin{array}{rcl} \mathbf{P}_i^{(1)}\mathbf{\Omega}_\infty^*\mathbf{P}_i^{(1)^T} & = & \mathbf{P}_i^{(2)}\mathbf{\Omega}_\infty^*\mathbf{P}_i^{(2)^T} \\ 2\mathbf{P}_i^{(1)}\mathbf{\Omega}_\infty^*\mathbf{P}_i^{(2)^T} & = & 0 \\ 2\mathbf{P}_i^{(1)}\mathbf{\Omega}_\infty^*\mathbf{P}_i^{(3)^T} & = & 0 \\ 2\mathbf{P}_i^{(2)}\mathbf{\Omega}_\infty^*\mathbf{P}_i^{(3)^T} & = & 0 \end{array} \tag{5.4}$$

Debemos recordar que el rango de la DAQ es 3 y por lo tanto la restricción $\det \mathbf{\Omega}_\infty^* = 0$ debe de forzarse. Una manera de hacer esto, se mostró en el caso de la matriz fundamental, y consiste en forzar que el valor singular más pequeño de la matriz de ecuaciones sea cero (usando SVD). Una vez obtenida la cuádrica absoluta dual, podemos obtener la transformación $\mathbf{H}$, que nos lleva de un marco proyectivo a uno métrico, de la

[2] Los nueve grados de libertad provienen de los 10 parámetros independientes de la DAQ por ser una matriz simétrica menos uno debido a que es una matriz homogénea.

ecuación diag(1,1,1,0) = $\mathbf{H}\mathbf{\Omega}^*_\infty\mathbf{H}^T$. La trasformación se obtiene mediante la descomposición de $\mathbf{\Omega}^*_\infty$ en $\mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$ donde $\mathbf{\Lambda}$ es una matriz diagonal cuyos elementos son los eigenvalores de $\mathbf{\Omega}^*_\infty$ y $\mathbf{U}$ es la matriz de los eigenvectores correspondientes. Mediante el uso de permutaciones en $\mathbf{U}$ y $\mathbf{\Lambda}$, podemos transformar $\mathbf{\Lambda}$ de forma que

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \lambda_3 & \\ & & & \lambda_4 \end{bmatrix}$$

con λ_i los eigenvalores de $\mathbf{\Omega}^*_\infty$ y donde $|\lambda_i| > |\lambda_j|$ para $i < j$ y $\lambda_4 = 0$. La trasformación $\mathbf{H}$ deseada esta dada por:

$$\mathbf{H} = \mathbf{L}\mathbf{U}^{-1}$$

donde

$$\mathbf{L} = \begin{bmatrix} \sqrt{\frac{1}{|\lambda_1|}} & & & \\ & \sqrt{\frac{1}{|\lambda_2|}} & & \\ & & \sqrt{\frac{1}{|\lambda_3|}} & \\ & & & 1 \end{bmatrix} \tag{5.5}$$

La estructura métrica se obtiene $\mathbf{P}_M = \mathbf{P}\mathbf{H}^{-1}$ y $\mathbf{X}_M = \mathbf{H}\mathbf{X}$. La matriz de calibración de la cámara puede obtenerse de la ecuación 5.2 mediante la descomposición de Cholesky.

La solución obtenida hasta este momento puede ser optimizada si se utiliza como la solución inicial de un método no lineal de mínimos cuadrados. Pollefeys et al. [1998] propone el siguiente criterio de minimización:

$$\text{mín} \sum_{i=1}^{n} \left|\left| \frac{\mathbf{K}\mathbf{K}^T}{\| \mathbf{K}\mathbf{K}^T \|_F} - \frac{\mathbf{P}_i\mathbf{\Omega}^*_\infty\mathbf{P}_i^T}{\| \mathbf{P}_i\mathbf{\Omega}^*_\infty\mathbf{P}_i^T \|_F} \right|\right|^2_F \tag{5.6}$$

donde $||\quad||_F$ representa la norma de Frobenius. En el Cuadro 5.1 se presenta un resumen del algoritmo utilizado en este documento para encontrar la calibración de la cámara.

Objetivo:
Dado un conjunto de correspondencias entre varias imágenes y restricciones sobre la matriz de calibración $\mathbf{K}$, calcular la reconstrucción métrica de los puntos y las cámaras.

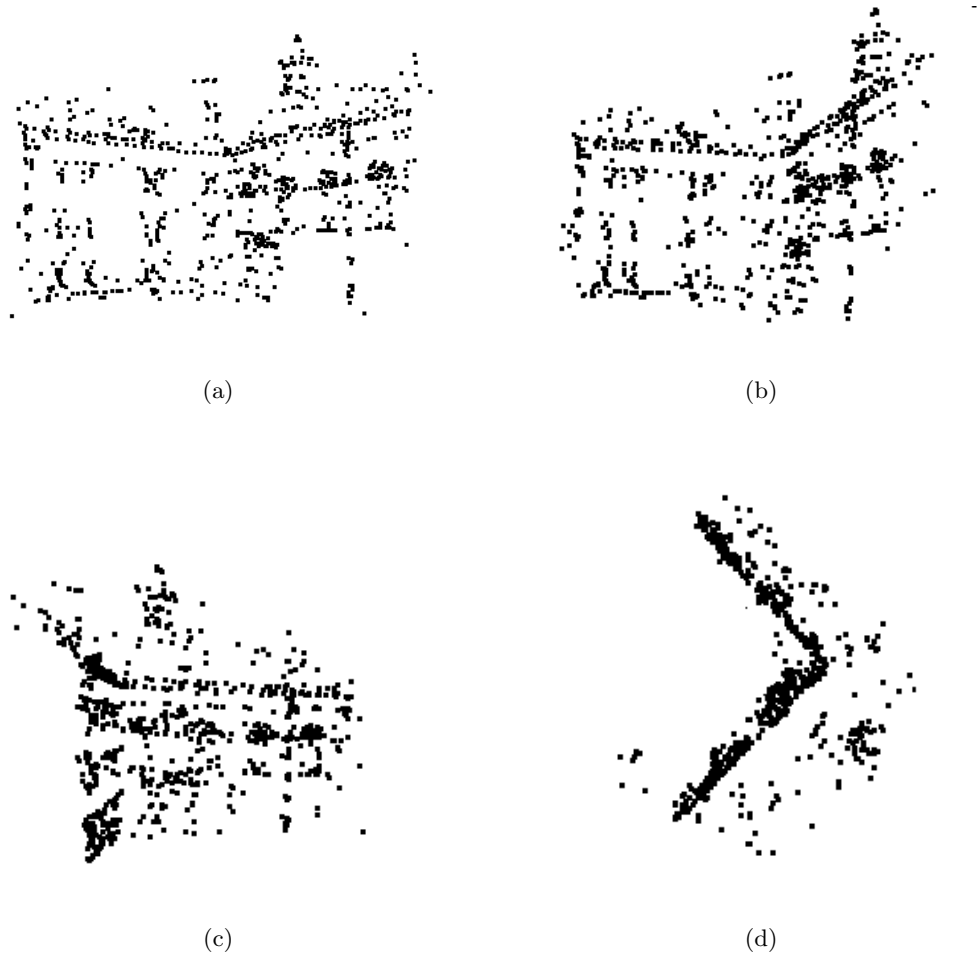
Algoritmo:

1. **Reconstrucción Proyectiva.** Calcula la reconstrucción proyectiva a partir de la secuencia de imágenes proporcionada, obteniendo matrices de proyección para cada vista $\mathbf{P}_i$ y un conjunto de puntos tridimensionales $\mathbf{X}_j$.
2. **Estimación de la DAQ.** Utiliza las restricciones dadas sobre la matriz de calibración y la ecuación 5.1, para formar un sistema de ecuaciones como se muestra en 5.4 para estimar $\mathbf{\Omega}^*_\infty$. Forzar la restricción $\det \mathbf{\Omega}^*_\infty = 0$.
3. **Descomposición de la DAQ.** Descomponer $\mathbf{\Omega}^*_\infty$ como $\mathbf{U\Lambda U}^T$. Donde $\mathbf{U}$ y $\mathbf{\Lambda}$ son las matrices de eigenvectores y eigenvalores respectivamente.
4. **Matriz de transformación.** La matriz de transformación $\mathbf{H}$ se obtiene de la siguiente manera: $\mathbf{H} = \mathbf{LU}^{-1}$, donde $\mathbf{L}$ esta definida como se muestra en la ecuación 5.5.
5. Aplica la transformación a los puntos ($\mathbf{HX}_j$) y a las matrices de proyección ($\mathbf{P}_i\mathbf{H}^{-1}$) para obtener una reconstrucción métrica.
6. Utiliza mínimos cuadrados no lineales para mejorar la solución. Un criterio de minimización puede ser el mostrado en la ecuación5.6.
7. La matriz de calibración $\mathbf{K}$ puede hallarse mediante la descomposición Choleski de ω^*.

Cuadro 5.1: Algoritmo de auto-calibración basado en la cuádrica absoluta dual $\mathbf{\Omega}^*_\infty$.

5.2.4. Experimentos

Se aplicaron los conceptos desarrollados en esta sección a los resultados obtenidos en el capítulo anterior para la secuencia mostrada en la Figura 4.5. Utilizando las matrices de proyección halladas, se encontró la transformación que traslada los 1017 puntos tridimensionales de un marco proyectivo a uno Euclidiano. Varias vistas de la reconstrucción métrica de dichos puntos se observan en la Figura 5.3. A pesar de la presencia de algunos *outliers*, la trasformación es claramente perceptible al comparar los resultados con los obtenidos en la reconstrucción proyectiva (Figura 4.6), en particular el ángulo recto que forman las paredes del edificio se puede apreciar de manera notoria en la Figura 5.3c.



(a) (b)

(c) (d)

Figura 5.3: Algunas vistas de la reconstrucción métrica de los puntos característicos, correspondientes a la secuencia de imágenes mostrada en la Figura 4.5. Un total de 1017 puntos característicos fueron reconstruidos. (a),(b) y (c) Vistas Frontales. (d) Vista superior.

Otros experimentos se realizaron utilizando diferentes secuencias de imágenes. Una de estas secuencias se muestra en la Figura 5.4. Esta secuencia esta formada por cuatro imágenes de resolución 1024 x 768 pixeles. Se reconstruyó un total de 914 puntos tridimensionales. Los resultados pueden observarse en la Figura 5.5.

El método de auto-calibración descrito en esta sección es altamente dependiente de los resultados obtenidos anteriormente, sobretodo de la correcta estimación de las matrices de proyección. Como se ha comentado previamente, un aspecto muy importante en la estimación de las matrices de proyección es la selección del marco inicial, si éste se selecciona incorrectamente (por ejemplo, si el movimiento de la cámara entre las dos

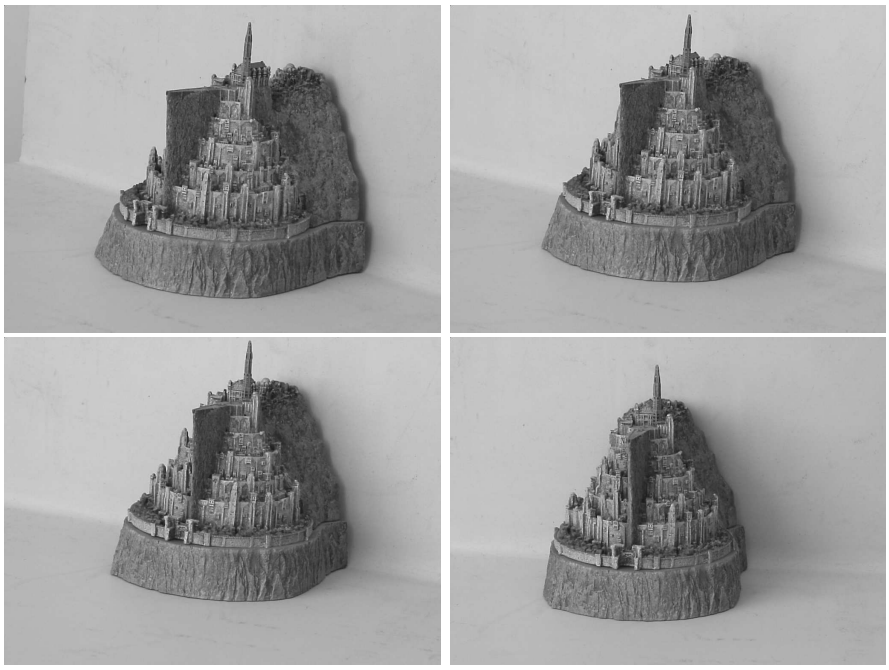


Figura 5.4: Secuencia de imágenes de un modelo a escala. Las imágenes tiene una resolución de 1024 x 768 pixels

vistas no es lo suficientemente general[3]), los resultados del método de auto-calibración no serán los esperados.

5.3. Rectificación epipolar

Una vez obtenidos los parámetros internos de la cámara, el siguiente paso consiste en realizar una búsqueda densa de la superficie. Este proceso se simplifica si las imágenes son rectificadas. La rectificación se basa en aplicar una trasformación a las imágenes de manera que las líneas epipolares estén alineadas horizontalmente. Esta transformación hace que los algoritmos de búsqueda de correspondencias sean más eficientes y sencillos, debido a que la búsqueda se realiza a lo largo de líneas horizontales correspondientes. El método que se describirá en esta sección fue propuesto por Hartley [1999], está basado en la matriz fundamental y consiste en encontrar una transformación que traslade el epipolo a un punto al infinito.

[3] Un movimiento general se refiere a una traslación en varias direcciones acompañada de una rotación.

Figura 5.5: Dos vistas de la reconstrucción métrica de los puntos característicos correspondientes a la secuencia mostrada en la Figura 5.4. Un total de 914 puntos característicos fueron reconstruidos.

5.3.1. Llevando el epipolo al infinito

En esta sección se discute cómo encontrar una transformación $\mathbf{H}$ que lleve el epipolo a un punto al infinito. En geometría proyectiva los puntos al infinito son aquellos cuya tercera coordenada (en coordenadas homogéneas) es cero, es decir, los puntos de la forma $(x, y, 0)^T$. Para que las líneas epipolares se transformen en líneas paralelas al eje x, el epipolo debe ser llevado al punto particular $(f, 0, 0)^T$. Esto le deja a $\mathbf{H}$ cuatro grados de libertad. Para evitar que la transformación seleccionada produzca severas distorsiones proyectivas en la imagen, se debe insistir en que la transformación $\mathbf{H}$ actúe como una transformación rígida en la vecindad de un punto $\mathbf{x}_0$ de la imagen.

Si suponemos que el epipolo $e = (f, 0, 1)^T$ se encuentra sobre el eje x, entonces la siguiente transformación $\mathbf{G}$ lleva al epipolo al punto al infinito deseado.

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ -1/f & 0 & 1 \end{bmatrix}$$

Para una situación general la transformación buscada estará dada por:

$$\mathbf{H} = \mathbf{GRT}$$

donde $\mathbf{T}$ es una traslación que lleva el punto $\mathbf{x}_0$ al origen $(0,0,1)^T$, $\mathbf{R}$ es una rotación que lleva al epipolo[4] a un punto $(f,0,1)^T$ sobre el eje x y $\mathbf{G}$ es la transformación anteriormente descrita que lleva $(f,0,1)^T$ al infinito.

5.3.2. Transformaciones correspondientes

El objetivo de la rectificación es el de simplificar la búsqueda de correspondencias en un par de imágenes, transformando las líneas epipolares de manera que sean paralelas al eje x. En la sección anterior se mostró cómo se puede realizar esto para una imagen. El objetivo de esta sección es, dado un par de imágenes J y J', encontrar un par de trasformaciones correspondientes $\mathbf{H}$ y $\mathbf{H}'$ de manera que, después de aplicar dichas transformaciones a las imágenes, las líneas epipolares correspondientes sigan siéndolo[5].

Como primer paso para encontrar este par de transformaciones, se estimará $\mathbf{H}'$ de la forma descrita en la sección anterior. Para encontrar la transformación $\mathbf{H}$ correspondiente nos basaremos en el siguiente resultado [Hartley, 1999]:

Resultado 5.1. *Sean J y J' imágenes cuya matriz fundamental es $\mathbf{F} = [\mathbf{e}']_{\mathrm{x}}\mathbf{M}$ y sea $\mathbf{H}'$ una transformación proyectiva de J'. Una transformación proyectiva $\mathbf{H}$ de J corresponde a $\mathbf{H}'$ sí y solo sí $\mathbf{H}$ es de la forma*

$$\mathbf{H} = (\mathbf{I} + \mathbf{H}'\mathbf{e}'\mathbf{a}^T)\mathbf{H}'\mathbf{M} \tag{5.7}$$

para algún vector $\mathbf{a}$.

La prueba a este resultado se puede encontrar en [Hartley, 1999; Hartley and Zisserman, 2004]. Basándonos en el resultado anterior y conociendo que en nuestro caso $\mathbf{H}'$ lleva al epipolo a un punto al infinito, entonces, $\mathbf{I} + \mathbf{H}'\mathbf{e}'\mathbf{a}^T = \mathbf{I} + (f,0,0)^T\mathbf{a}^T$ toma la forma

[4] Nos referimos al epipolo ya trasladado $\mathbf{e}_T = \mathbf{T}\mathbf{e}$

[5] Si $\mathbf{l}$ y $\mathbf{l}$ son líneas epipolares correspondientes entonces, $\mathbf{H}^{-T}\mathbf{l} = \mathbf{H}'^{-T}\mathbf{l}'$

$$\mathbf{H}_{\mathrm{A}} = \begin{bmatrix} a & b & c \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5.8}$$

Tomando $\mathbf{H}_0 = \mathbf{H}'\mathbf{M}$ y combinando 5.7 y 5.8 tenemos que la transformación buscada esta dada por $\mathbf{H} = \mathbf{H}_{\mathrm{A}}\mathbf{H}_0$. Podemos calcular $\mathbf{H}_0$ ya que contamos con $\mathbf{H}'$ y $\mathbf{M} = [\mathbf{e}']_{\mathrm{x}}\mathbf{F} + \mathbf{e}'\mathbf{v}^T$, donde $\mathbf{v}$ es un vector aleatorio utilizado para asegurarnos que $\mathbf{M}$ no sea singular. De esta manera el problema de encontrar la transformación $\mathbf{H}$ se reduce a estimar $\mathbf{H}_{\mathrm{A}}$. Si se cuenta con un conjunto de correspondencias entre las imágenes, $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$, escribiendo $\hat{\mathbf{x}}'_i = \mathbf{H}'\mathbf{x}'_i$ y $\hat{\mathbf{x}}_i = \mathbf{H}_0\mathbf{x}_i$, convertimos la estimación de $\mathbf{H}_{\mathrm{A}}$ en un problema de minimización, cuya función de costo es:

$$\sum_i d(\mathbf{H}_{\mathrm{A}}\hat{\mathbf{x}}_i, \hat{\mathbf{x}}'_i)^2 \tag{5.9}$$

Si $\hat{\mathbf{x}}_i = (\hat{x}_i, \hat{y}_i, 1)$ y $\hat{\mathbf{x}}'_i = (\hat{x}'_i, \hat{y}'_i, 1)$ y considerando que $(\hat{y}_i - \hat{y}'_i)^2$ es constante, la ecuación 5.9 se puede escribir de la siguiente manera:

$$\sum_i (a\hat{x}_i + b\hat{y}_i + c - \hat{x}'_i)^2 \tag{5.10}$$

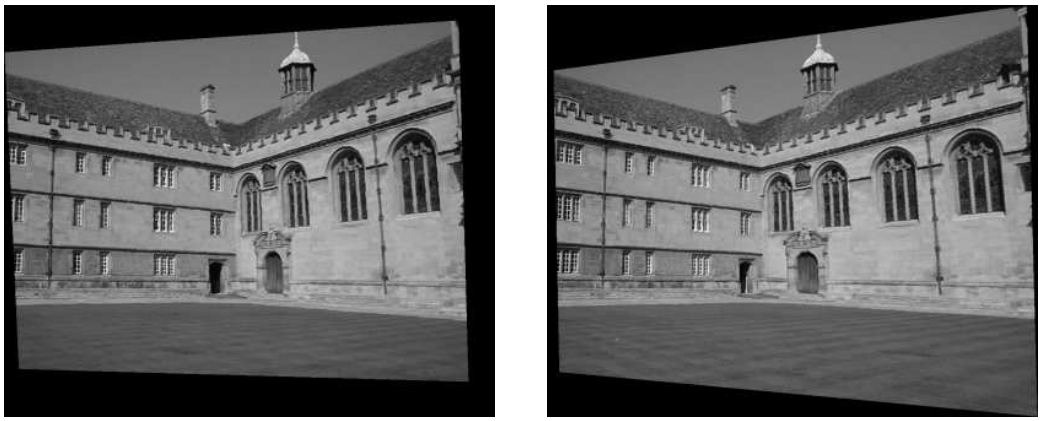
Ya que $\mathbf{H}_{\mathrm{A}}$ es una transformación afín, este problema de minimización es lineal. Algoritmos para resolver este tipo de sistemas se pueden encontrar en [Press et al., 1988] y en los apéndices de [Hartley and Zisserman, 2004].

5.3.3. Experimentos

La Figura 5.6a muestra un par de imágenes tomadas de la secuencia que se muestra en la Figura 4.5. Utilizando los correspondencias y la matriz fundamental previamente estimadas, las dos imágenes se transformaron utilizando el método descrito anteriormente. Los resultados de esta transformación se observan en la Figura 5.6b.

Una buena estimación de la matriz fundamental nos da en general buenas rectificaciones (siempre y cuando los epipolos no se encuentren dentro del área de la imagen).

(a)



(b)

Figura 5.6: Rectificación de imágenes. (a) Dos imágenes tomadas de la secuencia mostrada en la Figura 4.5. (b) Imágenes rectificadas utilizando el método descrito en esta sección.

Para generar las imágenes rectificadas se utilizó interpolación bilineal. Mejores resultados (en cuanto a la calidad de las imágenes rectificadas) podrían obtenerse utilizando otro tipo de interpolación, por ejemplo la cúbica.

5.4. Estimación densa de la superficie

Hasta este momento contamos con las matrices de proyección correspondientes a cada vista de nuestra secuencia de imágenes. De igual manera contamos con un modelo tridimensional basado en los puntos característicos obtenidos por el detector de esquinas. La fase final del proceso de reconstrucción presentado en este trabajo, consiste en

obtener una nube de puntos densa de la escena[6]. El método presentado en esta sección asume que las imágenes están rectificadas (sección 5.3).

El problema de encontrar correspondencias se revisó en el Capítulo 3. Es un problema muy complejo, y como se ha visto anteriormente varios métodos se han desarrollado buscando la mejor manera de solucionarlo. Los métodos basados en correlación son de los más utilizados, y entre estos destacan los propuestos por Cox et al. [1996] y Falkenhagen [1994, 1997]. Este último método será descrito a continuación.

5.4.1. Restricciones en la escena

Cuando el objetivo es encontrar correspondencias para la mayoría de los pixeles de la escena y no sólo para algunos puntos característicos, el uso de restricciones en la búsqueda facilita el proceso y mejora los resultados. La principal restricción que se utilizará es la restricción epipolar. Dicha restricción, limita el área de búsqueda de la correspondencia de un punto $\mathbf{x}_i$ en una imagen a una línea en la otra imagen (la línea epipolar correspondiente). En nuestro caso las imágenes se encuentran rectificadas, por lo que las líneas epipolares coinciden con los renglones de las imágenes, lo cual simplifica el algoritmo de búsqueda.

Otras restricciones se que se pueden hacer son [Pollefeys, 2000]:

1. Restricción de orden. El orden de los pixels en líneas epipolares correspondientes se preserva. Esta característica es una de las bases del esquema de programación dinámico que se describirá más adelante.

2. Restricción de unicidad. La correspondencia establecida entre dos pixeles es bidireccional y uno a uno, mientras no exista una oclusión en alguna de las imágenes.

3. Límite de disparidad. La banda de búsqueda a lo largo de la línea epipolar está restringida debido a que una escena tiene un rango de profundidad limitado.

[6] Con esto nos referimos a encontrar correspondencias para la mayoría de los pixeles de las imágenes, con las cuales se triangularán puntos tridimensionales

4. Restricción de continuidad. La disparidad de las correspondencias varía de manera continua excepto en los bordes de los objetos.

5.4.2. Mapa de Correlación

Varias de las restricciones descritas anteriormente no se pueden aplicar si se calcula la disparidad para cada pixel de manera independiente. Por lo tanto el calculo de la disparidad se realiza considerando a todos los pixeles sobre líneas epipolares correspondientes. La primera parte del método descrito en [Falkenhagen, 1994], comienza recorriendo las imágenes renglón por renglón (renglones correspondientes equivalen a líneas epipolares correspondientes en las imágenes rectificadas). Para cada par de renglones correspondientes se crea un mapa de correlación. Este mapa se genera calculando una medida de similitud entre los pixeles de un renglón y los pixeles candidatos en el renglón correspondiente. La medida de similitud utilizada fue ZNCC, la cual se describió en el Capítulo 3. Los pixeles candidatos son pixeles que se encuentran dentro de un rango fijo de disparidad. En la Figura 5.7 se observan dos maneras de almacenar un mapa de correlación calculado para dos renglones correspondientes. La intensidad de los pixeles muestra una mayor o menor medida de similitud (a mayor similitud mayor intensidad).

5.4.3. Mapa de Costo

Basándonos en el mapa de correlación se construye un mapa de costo aplicando una función de costo a cada par de pixeles candidatos. Un esquema de programación dinámico se sigue para obtener la mejor estimación de disparidad posible. Para cada pixel la función de costo esta formada por dos sumandos, un costo local y el costo del predecesor más probable. Se evalúan tres posibles predecesores para cada par de candidatos (Figura 5.8).

Dependiendo de la diferencia de disparidad el costo local puede ser un costo fijo C_{cambio} para un cambio de disparidad o un costo de correspondencia C_{match} para disparidad constante. El predecesor que nos lleva al costo mínimo se elige y su posición es almacenada. La función de costo esta dada por:

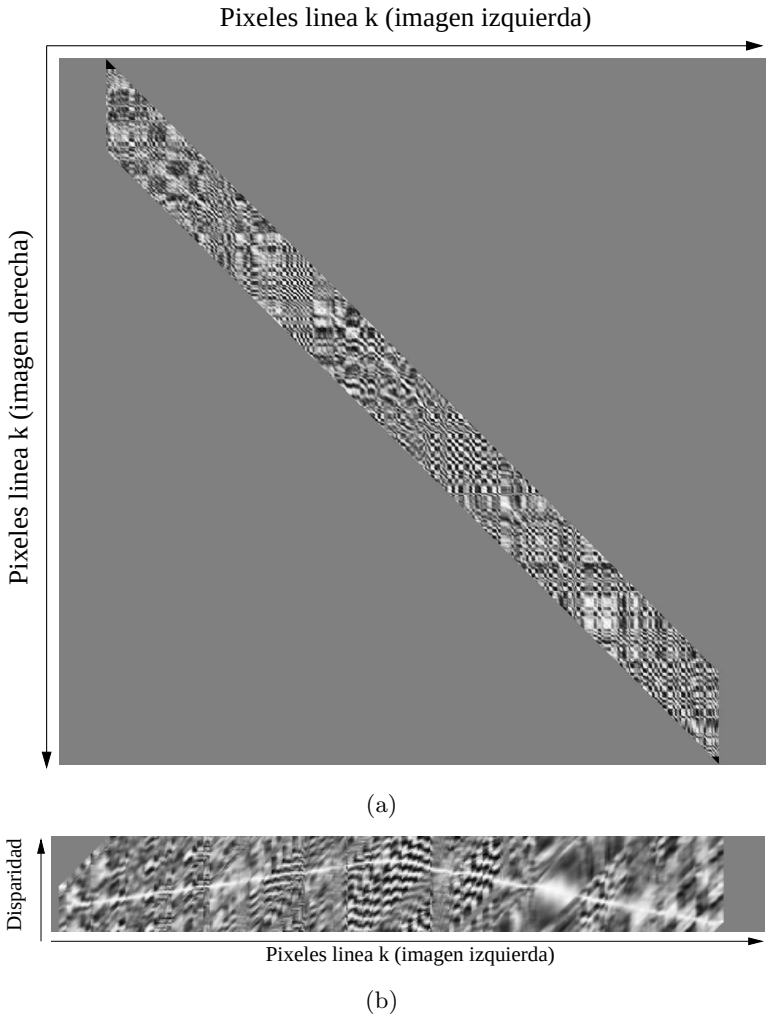


Figura 5.7: Dos maneras de almacenar mapas de correlación entre renglones correspondientes de imágenes rectificadas. A mayor intensidad se tiene una mayor similitud

$$C(i,j) = \text{mín} \begin{cases} C_1 & + & C_{cambio} \\ C_0 & + & C_{match} \\ C_2 & + & C_{cambio} \end{cases}$$

Una vez que se cuenta con el mapa de costo, se busca al candidato con el menor costo, correspondiente al último pixel de la línea. Comenzando con este candidato los valores de disparidad se van calculando hacia atrás, utilizando la información del predecesor almacenada para cada elemento del mapa de costo. En la Figura 5.9b se muestra el camino obtenido utilizando este método.

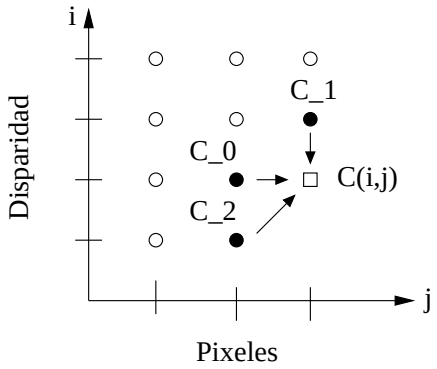
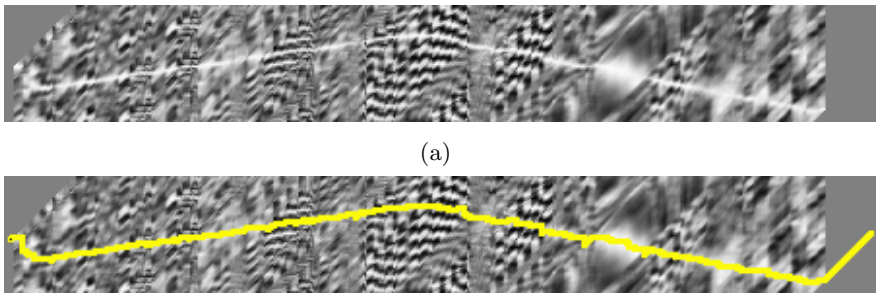


Figura 5.8: Cálculo de la función de costo. Se muestra la ubicación de los predecesores en el mapa de costo cuando se utiliza un esquema de almacenamiento como el mostrado en la Figura 5.7b.



(a)

(b)

Figura 5.9: Camino óptimo de disparidad.(a) Mapa de correlación. (b) Camino óptimo calculado.

El valor del cambio de disparidad C_{cambio} depende de la probabilidad de que una punto sea visto en ambas imágenes P_D (el cual a su vez depende del número de oclusiones) y de la varianza del ruido en la imagen σ^2. El valor de P_D normalmente se encuentra entre 0.95 y 0.98. Por otro lado el costo de disparidad constante depende del valor de similitud calculado. Los valores de estos costos están dados por [Falkenhagen, 1994]:

$$C_{cambio} = \ln\left(\frac{1-P_D}{P_D} \cdot \frac{1}{\sqrt{2\pi\sigma^2}}\right)$$

$$C_{match} = 255 \cdot \frac{1 - ZNCC^2}{4\sigma^2}$$

5.4.4. Mapa de disparidad

Luego de calcular el mapa de costo y el camino óptimo contamos con un valor de disparidad[7] para la mayoría de los pixeles de las imágenes rectificadas. En la Figura 5.10 se muestran dos imágenes rectificadas y sus correspondientes mapas de disparidad, donde mayor intensidad equivale a una mayor disparidad. Con los mapas de disparidad podemos calcular correspondencias entre las imágenes, ya que contamos con un desplazamiento estimado de los pixeles de una imagen con respecto a otra. En nuestro caso, las coordenadas de dos pixeles correspondientes en las imágenes rectificadas se deben transformar a su ubicación en las imágenes originales, utilizando para esto la inversa de la transformación obtenida por el método de rectificación. Una vez transformadas, estas correspondencias se utilizan para triangular puntos tridimensionales valiéndonos de las matrices de proyección obtenidas del proceso de auto-calibración.

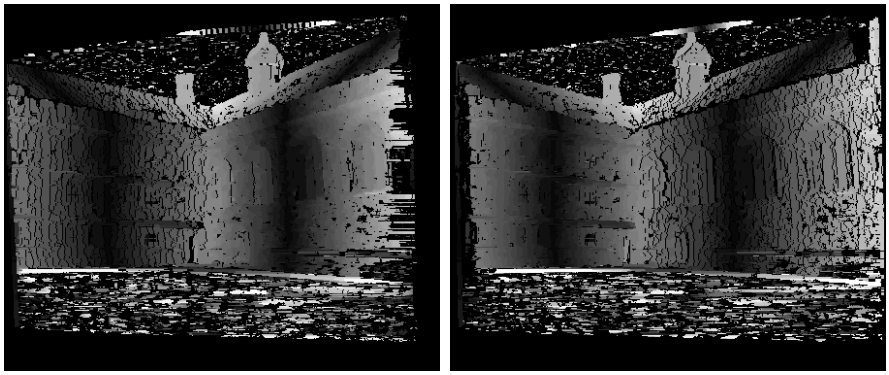
5.4.5. Experimentos

Utilizando las imágenes de la Figura 4.5, se aplicó el método descrito en esta sección para encontrar una nube densa de puntos. El valor para P_D utilizado fue de 0.98 y la varianza del ruido en las imágenes fue $\sigma^2 = 4$. En la Figura 5.11 se muestran varias vistas de la nube de puntos obtenida. Un total de 431,066 puntos fueron triangulados.

Como se puede ver en los resultados obtenidos se reconstruyeron muchos puntos correspondientes al cielo de las imágenes. Estos puntos no debieron ser reconstruidos ya que dichos puntos se encuentran en el infinito. Este tipo de problemas es ocasionado por la rectificación realizada y la calidad de las imágenes. En las Figuras 5.11a, 5.11b y 5.11d se observa un espacio entre el techo y las paredes del edificio, este espacio es ocasionado por una oclusión en las imágenes. Aunque existen varios *outliers*, sobre todo en el piso (Figura 5.11c), se obtuvo un aceptable nivel de detalle en la reconstrucción, éste se puede apreciar en las ventanas de una de las paredes del edificio claramente visibles en las Figuras 5.11g y 5.11h.

[7] El término disparidad se refiere al la diferencia de posición entre puntos correspondientes. En el caso particular de imágenes rectificadas, la disparidad es la diferencia entre las coordenadas x (ya que las correspondencias se encuentran en la misma línea).

(a)



(b)

Figura 5.10: Mapas de disparidad. (a) Imágenes rectificadas. (b) Mapas de disparidad correspondientes.

5.5. Discusión

En este capítulo se cumplieron varios objetivos. Primero se describió un método lineal para realizar la auto-calibración de la cámara. Este método aunque sencillo dio buenos resultados en la mayoría de los experimentos realizados. Luego se presentó un algoritmo para realizar la rectificación epipolar de un par de imágenes. Dicha rectificación no funciona de manera correcta cuando los epipolos se encuentran dentro de la imagen o incluso cerca de ella, produciéndose en ese caso severas distorsiones. Por ultimo se realizó la estimación densa de la superficie, obteniendo con esto un modelo de la escena consistente en una nube de puntos tridimensionales. En el siguiente capítulo se muestran otros experimentos realizados tanto con imágenes naturales como con sintéticas.

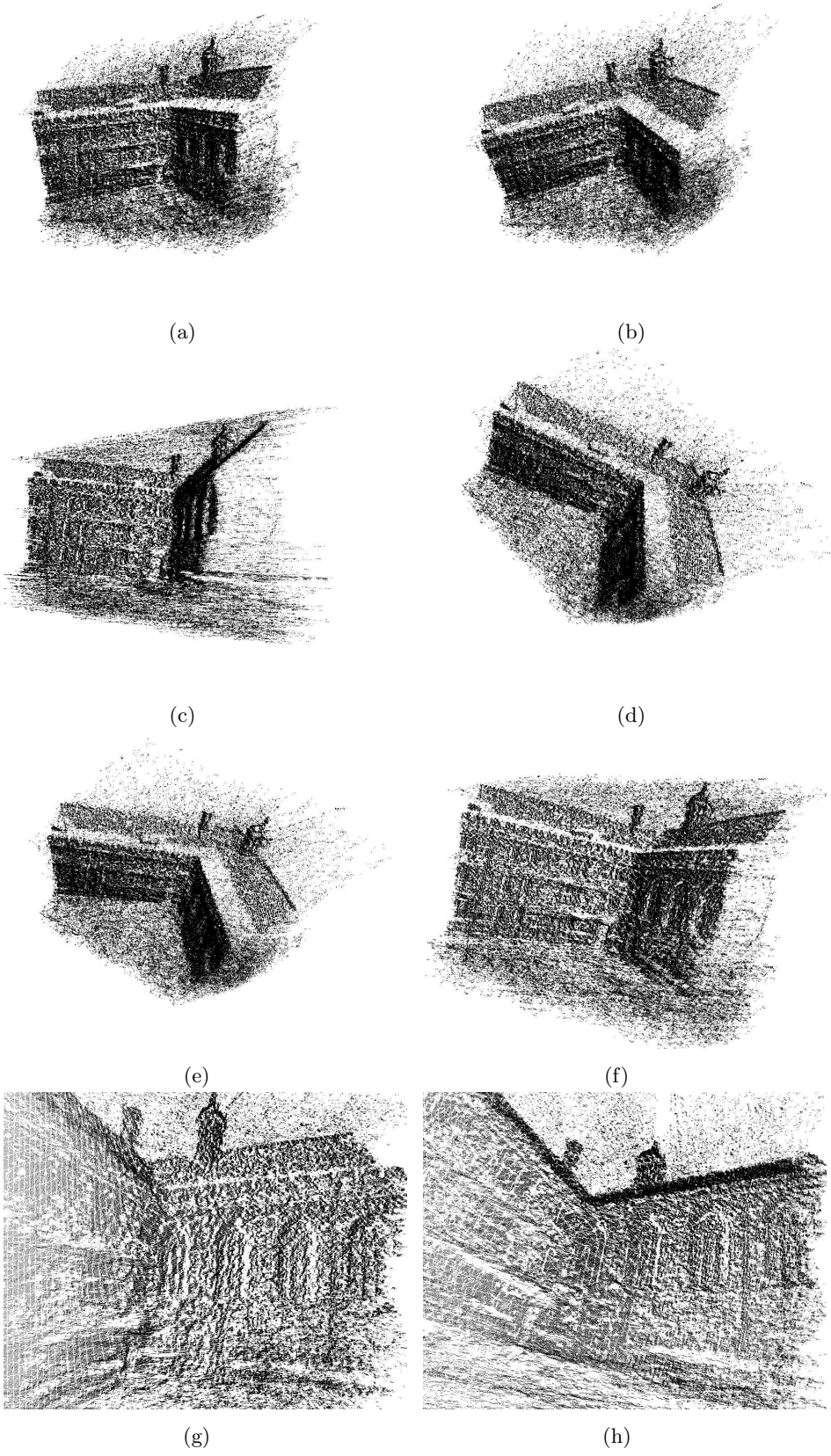


Figura 5.11: Reconstrucción densa. Diferentes vistas de la reconstrucción densa obtenida. Se reconstruyeron un total de 431,066 puntos.

Capítulo 6

Experimentos

A continuación se presentan dos experimentos realizados, acompañados de los resultados obtenidos en cada una de las fases del proceso de reconstrucción. Ambos experimentos fueron realizados en una computadora PowerBook de 1.33 Ghz, con 256 MB de memoria, corriendo bajo Linux. Las imágenes se almacenaron en dos formatos: PGM y PPM.

6.1. Edificio Maya (Uxmal)

En el primer experimento llevado a cabo se usó la secuencia de imágenes mostrada en la Figura 6.1. Esta secuencia fue tomada en el sitio arqueológico de Uxmal. Las imágenes tienen una resolución de 1024 x 768. Se utilizó el método de Harris-Stephens para detectar esquinas, con los siguientes parámetros: una ventana Gaussiana de 7x7 y varianza 1, $k = 0.04$ (constante de Harris), y la ventana de supresión de valores no máximos fue de 7x7. El operador de Prewitt se seleccionó para aproximar el gradiente de la imagen. El número de esquinas obtenido para cada imagen se muestra en el Cuadro 6.1.

Para el siguiente paso, correspondiente a la búsqueda de correspondencias tentativas, se utilizó el método del ZNCC (descrito en la sección 3.3.1). Usando una ventana de comparación de 10x10 pixeles y un área de búsqueda, en promedio, de una cuarta parte de la imagen se obtuvieron los resultados mostrados en el Cuadro 6.2. En la Figura 6.2

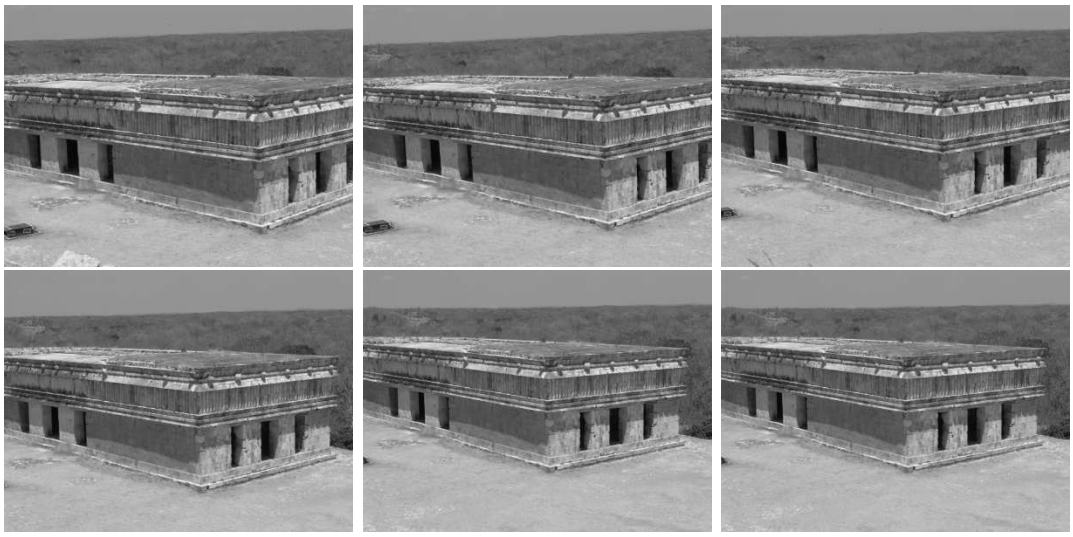


Figura 6.1: Secuencia de imágenes de un edificio Maya en Uxmal.

Imagen	# Esquinas
1	1792
2	1742
3	1673
4	1638
5	1552
6	1489

Cuadro 6.1: Número de esquinas obtenidas en cada imagen de la secuencia mostrada en la Figura 6.1.

se muestran las 761 correspondencias obtenidas para las imágenes 3 y 4 de la secuencia. Se nota la presencia de *outliers* en la parte central de la imagen. El valor del umbral fue de 0.85, es decir, sólo se consideraron como posibles candidatos, a los parejas de pixeles con valores de ZNCC superiores al umbral.

Imágenes	# Corresp (ZNCC)
1 - 2	845
2 - 3	862
3 - 4	761
4 - 5	716
5 - 6	732

Cuadro 6.2: Número de correspondencias obtenidas entre las imágenes de la secuencia de Uxmal. Se utilizó ZNCC como medida de similitud.

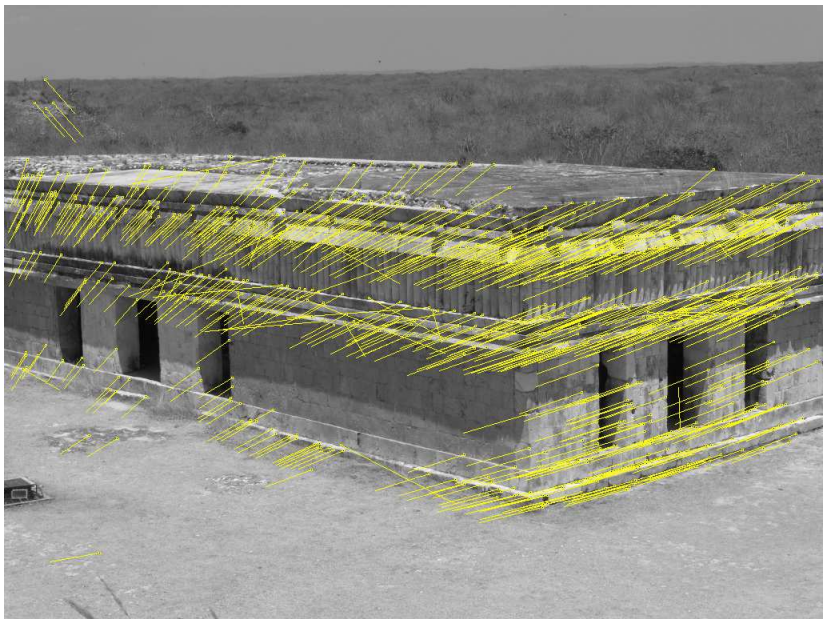


Figura 6.2: Correspondencias entre las imágenes 3 y 4 de la secuencia de Uxmal. Se encontró un total de 761 correspondencias utilizando la medida de similitud ZNCC. Algunos *outliers* son claramente visibles, sobre todo en la parte central de la imagen.

Las correspondencias obtenidas se utilizaron como entrada al algoritmo de RANSAC para la estimación de las matrices fundamentales. El valor de los parámetros utilizados para el RANSAC fue: un umbral $t = 1.25$, ϵ =0.50 y $p = 0.99$. Para la búsqueda guiada se utilizó una ventana de comparación de 10x10 pixeles, y el radio del área de búsqueda alrededor de las líneas epipolares fue de 2 pixeles. El umbral para la medida de similitud (ZNCC) se redujo a 0.65. En la Figura 6.3 se observan los 1049 *inliers* generados después de la búsqueda guiada correspondientes a las imágenes 3 y 4 de la secuencia. La mayoría de los *outliers* que se observaban en la Figura 6.2 han sido eliminados. Un resumen de los resultados obtenidos en esta fase se da en el Cuadro 6.3, donde, para cada par de imágenes se proporciona el número de *inliers* y el error RMS (*Root mean squared*) antes y después de efectuar la estimación de máxima similitud (MLE) y la búsqueda guiada.

Las matrices de proyección se inicializaron tomando como marco de referencia las primeras dos imágenes de la secuencia. Se probó con varias combinaciones de imágenes para elegir el marco de referencia que nos diera mejores resultados en la reconstrucción.

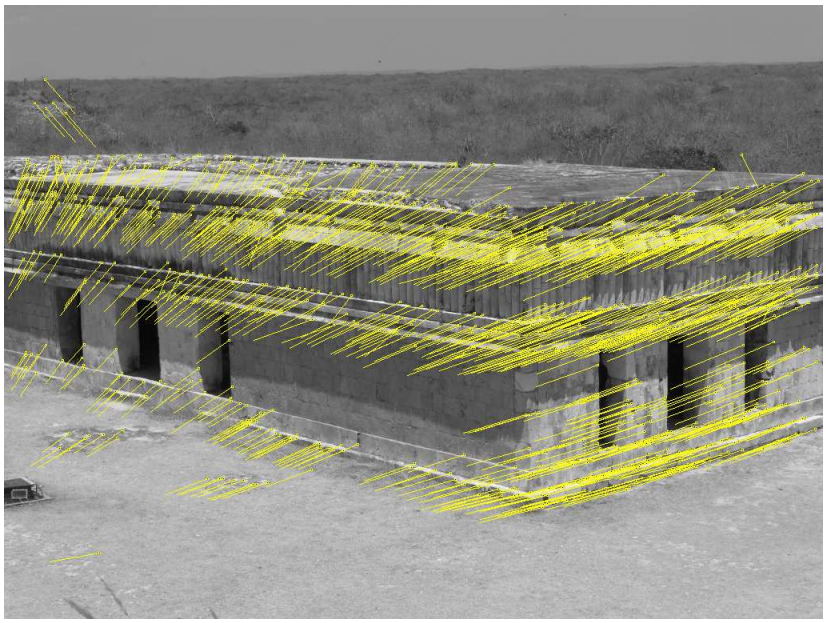


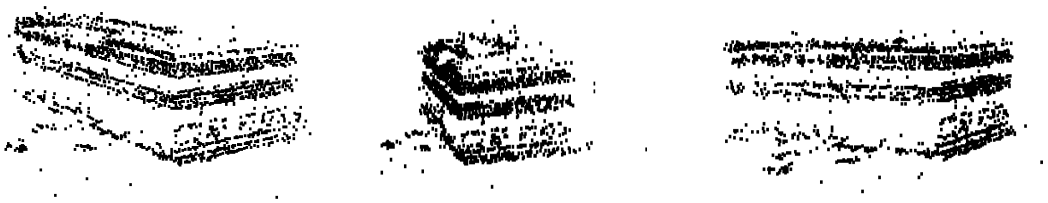
Figura 6.3: *Inliers* entre las imágenes 3 y 4 de la secuencia de Uxmal obtenidos después del RANSAC y la búsqueda guiada. Un total de 1049 *inliers*.

Imgs	In. (RANSAC)	RMS antes MLE	In. (Busq. G.)	RMS después MLE
1 - 2	802	0.211852	1137	0.201933
2 - 3	783	0.199379	1112	0.177737
3 - 4	706	0.199558	1049	0.159484
4 - 5	669	0.170066	1045	0.16566
5 - 6	719	0.160768	976	0.15112

Cuadro 6.3: Resultados del RANSAC para la secuencia de Uxmal. Para cada par de imágenes se muestra el número de *inliers* y el error RMS (*Root mean squared*) antes y después de efectuar la estimación de máxima similitud (MLE) y la búsqueda guiada.

Con las matrices de proyección obtenidas se buscaron correspondencias adicionales (de la misma manera que se hizo con las líneas epipolares en la búsqueda guiada en la fase anterior), bajando a 0.5 el umbral para la medida de similitud (ZNCC). Con el conjunto final de correspondencias, se triangularon puntos tridimensionales con el método de triangulación óptimo descrito en el capítulo 4. Un total de 2035 puntos tridimensionales fueron reconstruidos. Se aplicó un ajuste de haz (*bundle adjustment*) a las matrices de proyección y los puntos tridimensionales para refinar los resultados. Para este fin se utilizó la biblioteca de funciones proporcionadas por [Lourakis and Argyros, 2004].

La fase de auto-calibración se implementó de la manera descrita en el capítulo 5, obteniendo de esta manera las trasformaciones (proyectiva a métrica) correspondientes a las matrices de proyección y a los puntos tridimensionales. Se asume que los parámetros internos de la cámara son invariantes y que la oblicuidad es cero. Tres vistas de la reconstrucción métrica de los 2035 puntos característicos se observan en la Figura 6.4.



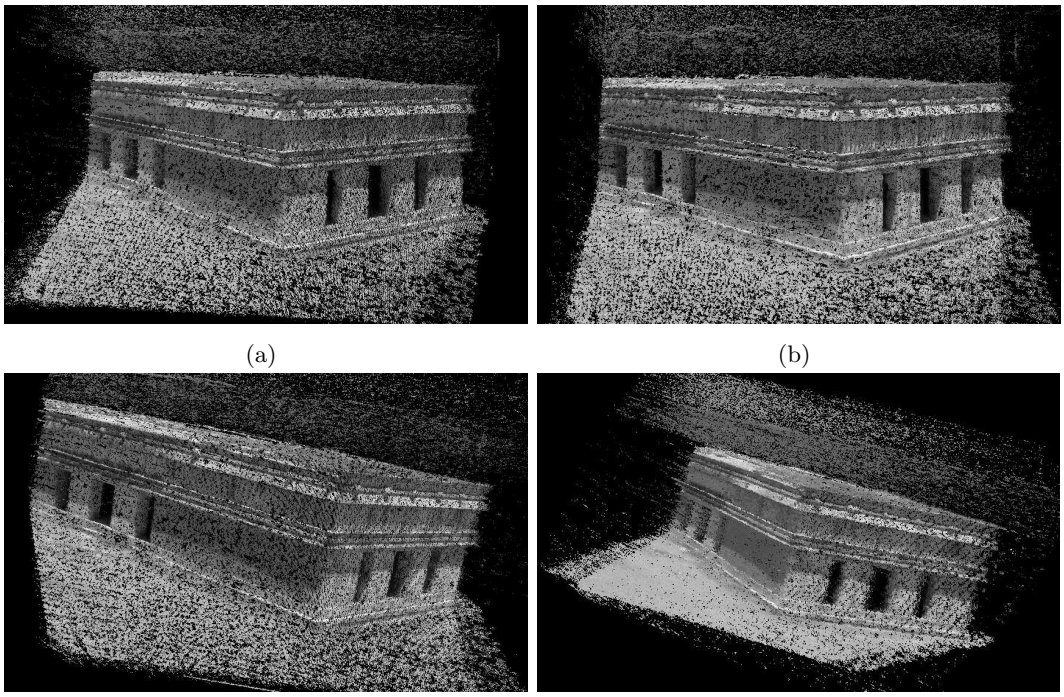
(a) (b) (c)

Figura 6.4: Tres vistas de la reconstrucción métrica de puntos característicos, correspondientes al edificio Maya de la secuencia de Uxmal. Un total de 2035 puntos tridimensionales fueron reconstruidos.

Contamos ahora con los elementos necesarios para realizar una estimación densa de la superficie y obtener la reconstrucción final. Las imágenes se rectificaron (sección 5.3) usando las matrices fundamentales anteriormente estimadas. Utilizando las imágenes resultantes de dicha rectificación, se calcularon mapas de disparidad para cada par de imágenes de la secuencia. Estos mapas permiten encontrar un gran número de correspondencias entre las imágenes, las cuales son trianguladas utilizando las matrices de proyección obtenidas luego de la auto-calibración. El proceso de reconstrucción de las correspondencias obtenidas en toda la secuencia de imágenes, es análogo al descrito en el capítulo 4 para la reconstrucción proyectiva, con la diferencia que se utilizó directamente el método de triangulación lineal.

En la Figura 6.5 se observan algunas vistas del modelo tridimensional obtenido. Dicho modelo consta de 560,548 puntos tridimensionales, lo que contrasta con los 2035 puntos triangulados anteriormente. Cada punto tridimensional se coloreó con el todo de gris del pixel correspondiente a su proyección en la imagen. Detalles de la reconstrucción se muestran en la Figura 6.6, donde se pueden apreciar de manera clara los ángulos y relieves del edificio. Aún después de todo el proceso, existen muchos *outliers* en la

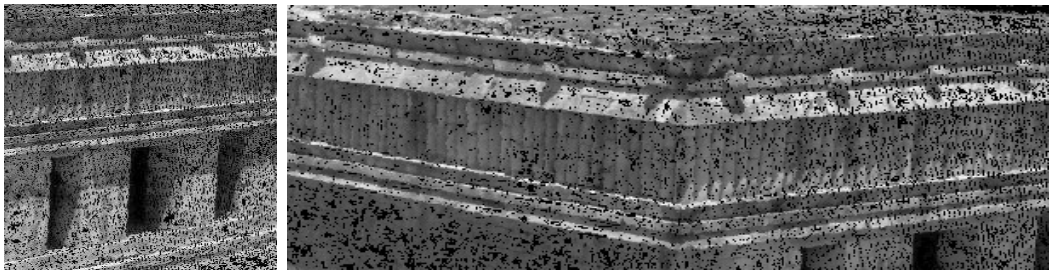
reconstrucción final, sobre todo en la parte correspondiente al cielo (Figura 6.5d).



(a) (b)

(c) (d)

Figura 6.5: Cuatro vistas de la reconstrucción obtenida del edificio Maya. Los puntos tridimensionales han sido coloreados de acuerdo al color del pixel correspondiente a su proyección en la imagen. El modelo esta formado por un total de 560,548 puntos tridimensionales.



(a) (b)

Figura 6.6: Detalle de la reconstrucción mostrada en la Figura 6.4. Se pueden apreciar de manera clara los ángulos y el relieve de la fachada del edificio.

6.2. Modelo a Escala

El segundo experimento que se presentará en este capítulo, consiste en una secuencia de imágenes sacada de un modelo a escala. Dicha secuencia ya ha sido utilizada anteriormente en este trabajo (Capítulo 5). Las cuatro imágenes que conforman la secuencia (Figura 6.7), tienen una resolución de 1024 x 768. Siguiendo los mismos pasos que en el experiemento anterior, se detectaron las esquinas (puntos característicos) utilizando el método de Harris-Stephens descrito en la sección 3.2.2. El tamaño de la ventana Gaussiana fue de 9x9 pixeles con una varianza de 1, la constante de Harris fue k =0.04 y el operador de Prewitt se utilizó para aproximar el gradiente de la imagen. La ventana de supresión de valores no máximos fue de 7x7 pixeles. El número de esquinas obtenido para cada imagen de la secuencia se enlistan en el Cuadro 6.4.

Figura 6.7: Secuencia de imágenes de un modelo a escala.

Imagen	# Esquinas
1	1570
2	1510
3	1457
4	1424

Cuadro 6.4: Número de esquinas obtenidas en cada imagen de la secuencia mostrada en la Figura 6.7.

En la fase de búsqueda de correspondencias entre los puntos característicos, se utilizó, al igual que en el experimento de Uxmal, un método de auto-correlación basado en la medida de similitud ZNCC. La ventana de comparación fue de 10x10 pixeles y el área de búsqueda se restringió a una ventana de 250x250. El valor del umbral usado para filtrar a los posible candidatos fue de 0.85. En la Figura 6.8 se muestran las 375 correspondencias obtenidas para las imágenes 3 y 4 de la secuencia. Una gran

cantidad de *outliers* es claramente visible. Los resultados completos de la búsqueda de correspondencias se listan en el Cuadro 6.5.

Imágenes	# Corres (ZNCC)
1 - 2	554
2 - 3	536
3 - 4	375

Cuadro 6.5: Número de correspondencias obtenidas (modelo a escala). Se utilizó ZNCC como medida de similitud.

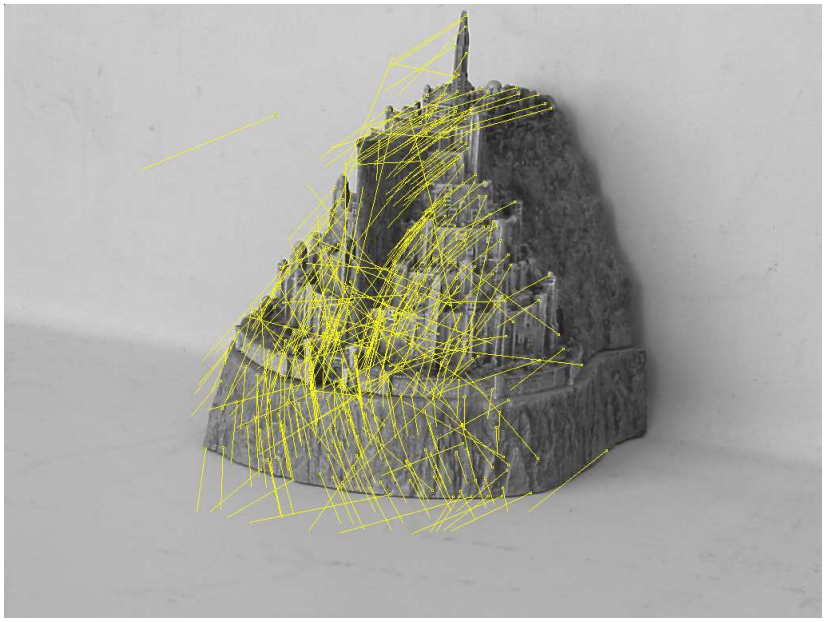


Figura 6.8: Correspondencias entre las imágenes 3 y 4 (Figura 6.7). Se encontraron un total de 375 correspondencias utilizando la medida de similitud ZNCC.

La implementación del método RANSAC, para la estimación de las matrices fundamentales y la clasificación de las correspondencias en *outliers* e *inliers*, se corrió con los siguientes parámetros: un umbral $t = 1.25$, ϵ =0.40 y $p = 0.99$. Para la búsqueda guiada se utilizó una ventana de comparación de 10x10 pixeles, y el radio del área de búsqueda alrededor de las líneas epipolares fue de 1.5 pixeles. El umbral para la medida de similitud (ZNCC) se redujo a 0.6. En la Figura 6.9 se observan los 398 *inliers* generados después de la búsqueda guiada correspondientes a las imágenes 3 y 4 de la secuencia. La mayoría de los *outliers* han sido eliminados. Un resumen de los resultados obtenidos en esta fase se lista en el Cuadro 6.6, donde, para cada par de imágenes se

proporciona el número de *inliers* y el error RMS (*Root mean squared*) antes y después de efectuar la estimación de máxima similitud (MLE) y la búsqueda guiada.

Imgs	In. (RANSAC)	RMS antes MLE	In. (Busq. G.)	RMS después MLE
1 - 2	516	0.212092	520	0.159040
2 - 3	395	0.297037	458	0.209020
3 - 4	206	0.253081	398	0.179275

Cuadro 6.6: Resultados del RANSAC para la secuencia del modelo a escala. Para cada par de imágenes se muestra el número de *inliers* y el error RMS (*Root mean squared*) antes y después de efectuar la estimación de máxima similitud (MLE) y la búsqueda guiada.

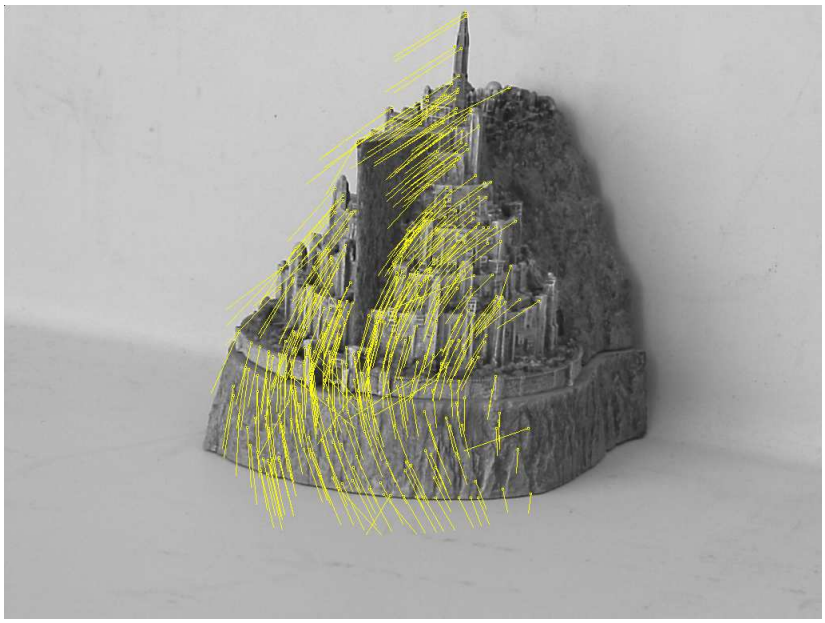
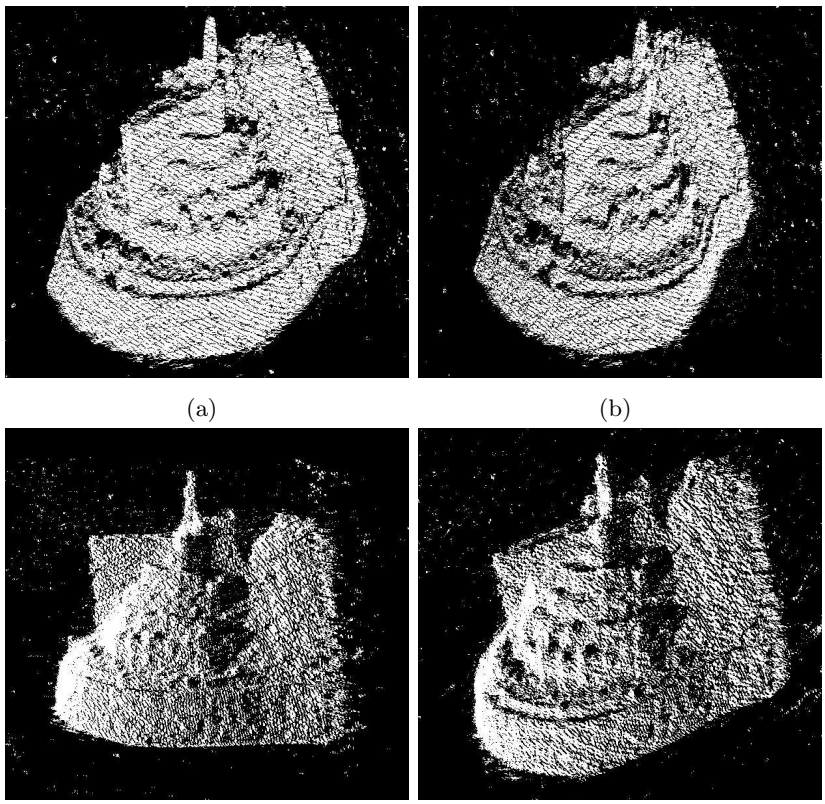


Figura 6.9: *Inliers* entre las imágenes 3 y 4 ((Figura 6.7) obtenidos después del RANSAC y la búsqueda guiada. Un total de 398 *inliers*.

Las primeras dos imágenes de la secuencia se tomaron como marco de referencia para inicializar las matrices de proyección. Se siguió el mismo proceso del primer experimento, buscando correspondencias adicionales y bajando a 0.5 el umbral para la medida de similitud (ZNCC). Un total de 914 puntos tridimensionales fueron reconstruidos. Se obtuvieron menos puntos que en el experimento anterior, esto se debe al menor número de imágenes en la secuencia y área de la imagen ocupada por el modelo a escala. Esta secuencia de imágenes fue utilizada para realizar uno de los experimentos de

auto-calibración en el capítulo 5. Los resultados obtenidos luego de la auto-calibración, se muestran en la Figura 5.5 (dos vistas de la reconstrucción métrica de los puntos característicos), y no serán reproducidos aquí.

La rectificación y la estimación densa de la superficie se realizaron como se explica en el capítulo 5. Varias vistas de la reconstrucción final se pueden observar en las Figura 6.10 y 6.11, un total de 226,334 puntos fueron reconstruidos. Las imágenes presentadas en la Figura 6.10, nos muestran vistas del modelo sin color. En la Figura 6.11 se aplicó color (nivel de gris) a los puntos tridimensionales, de la manera descrita en el experimento anterior.

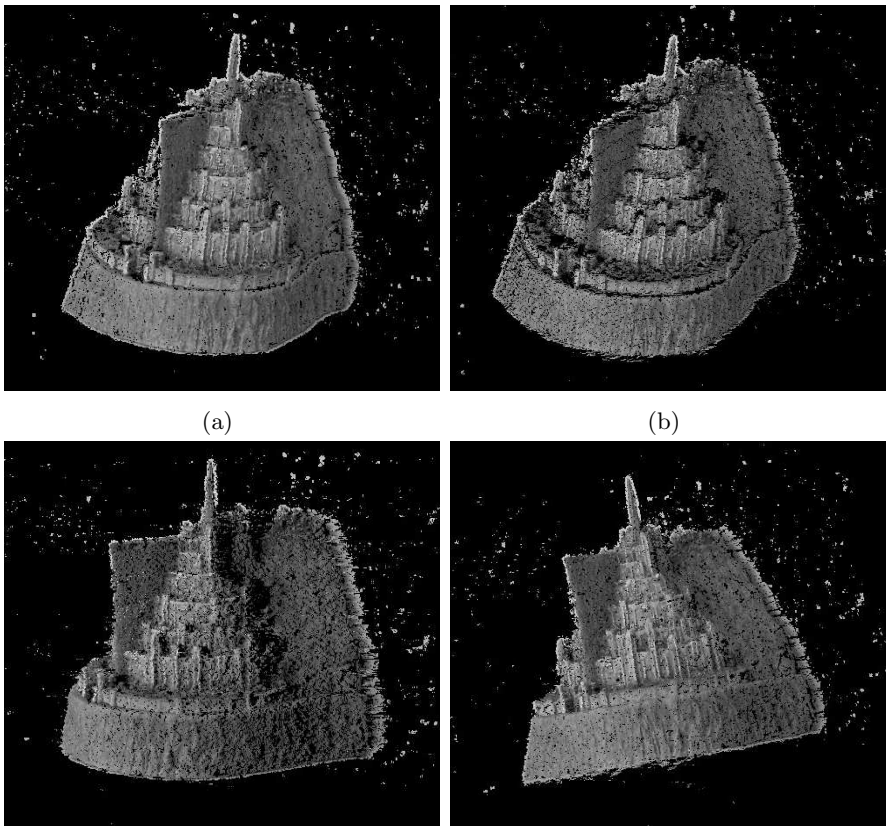


(a) (b)

(c) (d)

Figura 6.10: Varias vistas de la reconstrucción obtenida del modelo a escala. (a)-(d) Puntos tridimensionales sin colorear. El modelo esta formado por un total de 226,334 puntos tridimensionales.

Algunos *outliers* se observan alrededor del modelo, producto de manchas presentes en el fondo de las imágenes. Los efectos de la oclusión se aprecian en la Figura6.10c, donde se observa un área en el centro del modelo en donde no se triangularon puntos, debido a que esta sección no era visible en las imágenes.



(a) (b)

(c) (d)

Figura 6.11: Varias vistas de la reconstrucción obtenida del modelo a escala. (a)-(d) Los puntos tridimensionales han sido coloreados de acuerdo al color del pixel correspondiente a su proyección en la imagen. El modelo esta formado por un total de 226,334 puntos tridimensionales.

Capítulo 7

Conclusiones

7.1. Sumario

El trabajo presentado en esta tesis trata de la recuperación de un modelo tridimensional obtenido a partir de una secuencia de imágenes. Tomando esto como el objetivo central, un enfoque modular a este problema fue propuesto basado principalmente en el artículo escrito por [Pollefeys et al., 2004]. Guiándonos en dicho artículo, y revisando otros métodos disponibles en la literatura de visión computacional, se seleccionó un conjunto de métodos para llevar a cabo la reconstrucción en sus diversas fases. Una característica importante de esta aproximación modular, es su flexibilidad, ya que los métodos elegidos en este trabajo pueden ser sustituidos sin alterar el proceso (pero obteniendo diferentes resultados).

Todo el proceso de reconstrucción se realiza de manera semi-automática, esto implica que para cada secuencia de imágenes (y en ocasiones para cada imagen de la secuencia), la elección de los valores asignados a los parámetros utilizados por cada método se realizó no sólo de manera independiente sino que en la mayoría de los casos de manera empírica. La reconstrucción se llevó a cabo siguiendo las fases: primero puntos característicos (esquinas) fueron obtenidos de las imágenes y se extrajo un conjunto de correspondencias tentativas entre ellos. Luego utilizando estas correspondencias se estimaron las matrices fundamentales entre las imágenes, las cuales encapsulan la información de la geometría epipolar; dichas matrices se emplearon para refinar y encontrar correspondencias adicionales. Se estableció un marco de referencia y se cal-

cularon las matrices de proyección de cada vista. Una reconstrucción proyectiva de los puntos característicos fue posible gracias a estas matrices. Como siguiente paso se buscó auto-calibrar las cámaras imponiendo restricciones en los parámetros internos de las mismas; gracias a la auto-calibración se consiguió estimar transformaciones que nos llevan la reconstrucción de proyectiva a métrica. Para simplificar la búsqueda de correspondencias un método sencillo de auto-calibración se implementó. Por último usando las imágenes rectificadas se obtuvo el modelo final mediante un método de búsqueda densa de correspondencias. Cabe recalcar que todos los métodos descritos en este trabajo fueron implementados desde cero, y sólo se utilizaron implementaciones ya existentes en el caso del Levenberg-Marquardt y el ajuste de haz (*bundle adjustment*), lo cual se indicó en su momento.

Existen varios puntos a recalcar que surgieron producto de los experimentos y que se deben tener muy en cuenta a la hora de implementar los métodos descritos y realizar experimentos propios para obtener buenos resultados. Se debe elegir un método para encontrar puntos característicos que nos asegure un buen grado de *repitabilidad*[1], ya que esto no sólo facilita la detección de correspondencias sino que propicia que un número mayor de correspondencias correctas sea encontrado. Algo que se ha repetido en varias secciones de este documento, es la importancia de una buena selección en las imágenes usadas para estimar el marco inicial de la reconstrucción. Una mala selección para el marco inicial nos lleva a una pésima reconstrucción. Otro punto a considerar al momento de adquirir las imágenes, es el movimiento de la cámara; si el movimiento no es lo suficientemente general el método de auto-calibración descrito no dará buenos resultados y por consiguiente la reconstrucción métrica no será posible (los movimientos para los cuales el método de auto-calibración falla se describen en [Pollefeys et al., 2004]). Como un último comentario el método implementado para la estimación densa de la superficie, aunque nos proporcionó buenos resultados, consume mucho espacio de memoria, lo cual lo hace muy lento cuando se trabaja con imágenes con alta resolución, tardando de 30 hasta 60 minutos en algunos experimentos (imágenes de 1024 x 768).

[1] Una descripción sobre el criterio de *repitabilidad* se da en el apéndice A.

7.2. Trabajo Futuro

Concluimos indicando algunas de las áreas más importantes de este trabajo donde creemos se pueden realizar futuras modificaciones que ayudarían a mejorar los resultados y la eficiencia del proceso.

- Primero, una investigación a fondo en cuanto a la selección de los valores óptimos asignados a los parámetros utilizados por los métodos descritos en este trabajo, sería de gran importancia para obtener mejores resultados. Hasta ahora no existen muchos trabajos que discutan este tema con claridad y en general dichos parámetros son elegidos de manera empírica. Dicha investigación debería realizarse, no sólo de manera particular para cada método (existen varios estudios de este tipo), sino de manera global, es decir, estudiar las correlaciones entre los parámetros de los métodos usados en cada fase.

- En el caso de la rectificación, el uso de métodos que generen menores distorsiones en las imágenes rectificadas, nos ayuda a obtener mapas de disparidad más exactos y por lo tanto una mejor calidad en las reconstrucciones.

- El modelo 3D que se obtuvo en esta tesis esta formado por un conjunto de puntos tridimensionales. Reconstrucciones de mucho mayor calidad visual se obtienen aplicando algún método de triangulación para formar superficies.

- Por último, un área muy interesante y que ha tenido bastante auge recientemente, es la computación paralela. Debido a la implementación modular y a la estructura de varios de los métodos presentados en este trabajo, se considera que el proceso de reconstrucción es altamente paralelizable, lo cual conllevaría un gran aumento en la eficiencia del todo proceso, reduciendo drásticamente el tiempo de procesamiento necesario.

Bibliografía

Anandan, P. (1984). Computing dense displacement fields with confidence measures in scenes containing occlusion. *SPIE Intelligent Robots and Computer Vision*, 521:184–194.

Barron, J., Fleet, D., and Beauchemin, S. (1992). Performance of optical flow techniques. *Computer Vision and Pattern Recognition*, 92:236–242.

Beardsley, P., Zisserman, A., and Murray, D. (1997). Sequential Updating of Projective and Affine Structure from Motion. *International Journal of Computer Vision*, 23(3):235–259.

Brand, P. and Mohr, R. (1994). Accuracy in image measure. In El-Hakim, S., editor, *the SPIE Conference on Videometrics III*, volume 2350, pages 218–228, Boston, Massachusetts, USA.

Burt, P., Hong, T., and Rosenfeld, A. (1981). Image segmentation and region property computation by cooperative hierarchical computation. *IEEE Transactions on Systems, Man and Cybernetics*, 11:802–809.

Burt, P., Yen, C., and Xu, X. (1982). Local correlation measures for motion analysis: A comparative study. *IEEE Proceedings of Pattern Recognition and Information Processing*, pages 269–274.

Chojnacki, W., Brooks, M., van den Hengel, A., and Gawley, D. (2001). A fast MLE-Based Method for Estimating the Fundamental Matrix. *Image Processing*, 2:189–192. Department of Computer Science University of Adelaide Adelaide, SA 5005, Australia.

Cox, I. J., Hingorani, S. L., Rao, S. B., and Maggs, B. M. (1996). A Maximum Likelihood Stereo Algorithm. *Computer Vision and Image Understanding*, 63(3):542–567.

Deriche, R. and Giraudon, G. (1993). A computational approach for corner and vertex detection. *International Journal of Computer Vision*, 10(2):101–124.

Falkenhagen, L. (1994). Depth estimation from stereoscopic image pairs assuming picewise continous surfaces. In *Proc. of European Workshop on combined Real and Synthetic Image Processing for Broadcast and Video Production*, Hamburg.

Falkenhagen, L. (1997). Hierarchical Block-Based Disparity Estimation Considering Neighbourhood Constraints. International workshop on SNHC and 3D Imaging, September 5-9, 1997, Rhodes, Greece.

Faugeras, O., Luong, Q., and Maybank, S. (1992). Camera self-calibration: Theory and experiments. In *Proceedings of the European Conference on Computer Vision*, volume LNCS 558, pages 321–334. Springer-Verlag.

Fischler, M. and Bolles, R. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395.

Forsyth, D. and Ponce, J. (2003). *Computer Vision: A Modern Approach.* Prentice Hall.

Gallo, I., Binagli, E., and Raspanti, M. (2005). Neural adaptive stereo matching. *Pattern Analysis, Statistical Modelling and Computational Learning*, 25(15):1743–1758.

Gluckman, J. and Nayar, S. K. (2001). Rectifying transformations that minimize resampling effects. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Kauai, Hawaii.

Gonzalez, R. and Woods, R. (1993). *Digital Image Processing.* Addison-Wesley, 2nd edition.

Harris, C. and Stephens, M. (1988). A combined corner and edge detector. In *Proceedings of the Alvey Vision Conference, University of Manchester*, pages 147–151. The Plessey Company.

Hartley, R. (1999). Theory and Practice of Projective Rectification. *International Journal of Computer Vision*, 35(2):115–127.

Hartley, R. and Zisserman, A. (2004). *Multiple View Geometry in Computer Vision.* Cambridge University Press, 2nd edition.

Hartley, R. I. (1997). In defense of the eight-point algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(6):580–593.

Heath, M., Sarkar, S., Sanocki, T., and Bowyer, K. (1997). A robust visual method for assessing the relative performance of edge-detection algorithms. *IEEE Transactions in Pattern Analysis and Machine Intelligence*, 19(12):1338–1359.

Hildreth, E. and Royden, C. (1998). *Computational, Neurobiological and Psychophysical perspectives.* Mit Press. Camb Massachusetts. London, England.

Kitchen, L. and Rosenfeld, A. (1982). Gray level corner detector. *Pattern Recognition Letters*, 1:95–102.

Koch, R., Pollefeys, B., Heigl, B., Van Gool, L., and Niemann, H. (1999). Calibration of Hand-held Camera Sequences for Plenoptic Modeling. In *Proceedings of the International Conference on Computer Vision*, pages 585–591, Corfu (Greece).

Levenberg, K. (1944). A Method for the Solution of Certain Problems in Least Squares. *Quarterly of Applied Mathematics*, 2:164–168.

Longuet-Higgins, H. C. (1981). A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135.

Lourakis, M. and Argyros, A. (2004). The Design and Implementation of a Generic Sparse Bundle Adjustment Software Package Based on the Levenberg-Marquardt Algorithm. Technical Report 340, Institute of Computer Science - FORTH, Heraklion, Crete, Greece.

Marquardt, D. (1963). An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *SIAM Journal on Applied Mathematics*, 11:431–441.

Marr, D. (1982). *Vision: a computational investigation into the human representation and processing of visual information*. W. H. Freeman, San Francisco.

Mokhtarian, F. and Suomela, R. (1998). Curvature Scale Space Based Image Corner Detection. In *Proceedings of the European Signal Processing Conference*, pages 2549–2552, Island of Rhodes, Greece.

Moravec, H. (1977). Towards Automatic Visual Obstacle Avoidance. In *Proceedings of the 5th International Joint Conference on Artificial Intelligence*, page 584.

Moravec, H. (1979). Visual Mapping by a Robot Rover. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 598–600.

Mount, D. M., Netanyahu, N. S., and Le Moigne, J. (1997). Efficient Algorithms for Robust Feature Matching. In *Proceedings of the CESDIS Image Registration Workshop*, pages 247–256, NASA Goddard Space Flight Center (GSFC), Greenbelt, MD. NASA.

Pollefeys, M. (2000). Tutorial on 3D Modeling from Images. Lecture Notes.

Pollefeys, M., Koch, R., and Van Gool, L. (1998). Self-Calibration and Metric Reconstruction in spite of Varying and Unknown Intrinsic Camera Parameters. In *Proceedings of the International Conference on Computer Vision*, pages 90–95.

Pollefeys, M., Koch, R., and Van Gool, L. (1999). A simple and efficient rectificacion method for general motion,. In *Proceedings of the International Conference on Computer Vision*, pages 496–501.

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

Press, W., Flannery, B., Teukolsky, S., and Vetterling, W. (1988). *Numerical Recepies in C*. Cambridge University Press.

Roy, S., Meunier, J., and Cox, I. (1997). Cylindrical Rectification to Minimize Epipolar Distortion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 393–399.

Sampson, P. (1982). Fitting conic sections to 'very scattered' data: An interactive refinement of the Bookstein algorithm. *Computer Vision, Graphics, and Image Processing*, 18:97–108.

Schmid, C., Moht, R., and Bauckhage, C. (2000). Evaluation of Interest Point Detectors. *International Journal of Computer Vision*, 37(2):151–172.

Semple, J. and Kneebone, G. (1952). *Algebraic Projective Geometry*. Oxford Classic Texts in the Physical Sciences. Oxford University Press.

Smith, S. M. and Brady, J. M. (1997). SUSAN - A New Approach to Low Level Image Processing. *International Journal of Computer Vision*, 23(1):45–78.

Thimbleby, H., Inglis, S., and Witten, I. (1994). Displaying 3d images: Algorithms for single-image random-dot stereograms. *Computer*, 27(10):38–48.

Trajkovic, M. and Hedley, M. (1998). Fast corner detector. *Image and Vision Computing*, 16:75–87.

Triggs, B., McLauchlan, P., Hartley, R., and Fitzgibbon, A. (2000). Bundle Adjustmet – A Modern Synthesis. In Triggs, W., Zisserman, A., and Szeliski, R., editors, *Vision Algorithms: theory and Practice*, LNCS, pages 298–375. Springer Verlag.

Trigss, B. (1997). The Absolute Quadric. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition*, pages 609–614. IEEE Computer Soc. Press.

Tsai, R. and Huang, T. (1984). Uniqueness and Estimation of Three Dimensional Motion Parameters of Rigid Objects With Curved Surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:13–27.

Wang, H. and Brady, M. (1995). Real-time corner detection algorithm for motion estimation. *Image and Vision Computing*, 13(9):695–703.

Welch, G. and Bishop, G. (1995). An Introduction to the Kalman Filter. Technical Report TR95-041, University of North Carolina at Chapel Hill, Chapel Hill, NC, USA.

Zhang, Z., Deriche, R., Faugueras, O., and Luong, Q. (1995). A Robust Technique for Matching Two Uncalibrate Images Through the Recovery of the Unknown Epipolar Geometry. *Artificial Intelligence*, (78):87–119.

Zhang, Z. and Loop, C. (2001). Estimating the Fundamental Matrix by Transforming Image Points in Projective Space. *Computer Vision and Image Understanding*, 82(2):174–180.

Zheng, Z., Wang, H., and Teoh, E. (1999). Analysis of gray level corner detection. *Pattern Recognition Letters*, 20:149–162.

Apéndice A

Comparación de detectores de esquinas

Existen varios criterios para evaluar detectores de puntos característicos como lo son: inspección visual [Heath et al., 1997], precisión de localización [Brand and Mohr, 1994] y análisis teórico [Deriche and Giraudon, 1993] entre otros. Para ejemplificar el uso de estos criterios se implementó uno de los dos criterios descritos en [Schmid et al., 2000]: la Repitabilidad, la cual evalúa la estabilidad geométrica bajo diferentes transformaciones. El otro criterio descrito en [Schmid et al., 2000] es el Contenido de información, que mide que tan distintos son los puntos característicos entre si.

La razón de repitabilidad se define como el número de puntos repetidos entre dos imágenes con respecto al número total de puntos detectados. El valor de esta medida varía entre 0 y 1, resultados cercanos al uno tienen una mejor medida de repitabilidad. Para más detalles de la implementación consultar [Schmid et al., 2000]. Se probará el criterio de repitabilidad con dos diferentes tipos de transformaciones: rotación y escalamiento. Otras transformaciones pueden probarse tales como el cambio de intensidad y de punto de vista. En la Figura A.1 se muestra una pintura de Van Gogh y dos rotaciones de ella, así como también las esquinas encontradas por el método de Harris-Stephens.

La repitabilidad de esta secuencia se muestra en la Figura A.3a. Los ángulos de rotación varían de 20 a 180 y el error de localización es $\epsilon = 2$. Como se mencionó anterior mente también se probó la repitabilidad bajo cambios de escala. En la Figura A.2 se puede ver una secuencia que ejemplifica este procedimiento. Se utilizaron los mismos parámetros para el detector de esquinas de Harris-Stephens que en la secuencia de rotación. Una gráfica comparativa de los resultados obtenidos bajo los dos métodos se muestra en la Figura A.3b. Se observa claramente que el detector de esquinas de Moravec tiene una repitabilidad nula en esta secuencia para cambios relativamente grandes de escala.

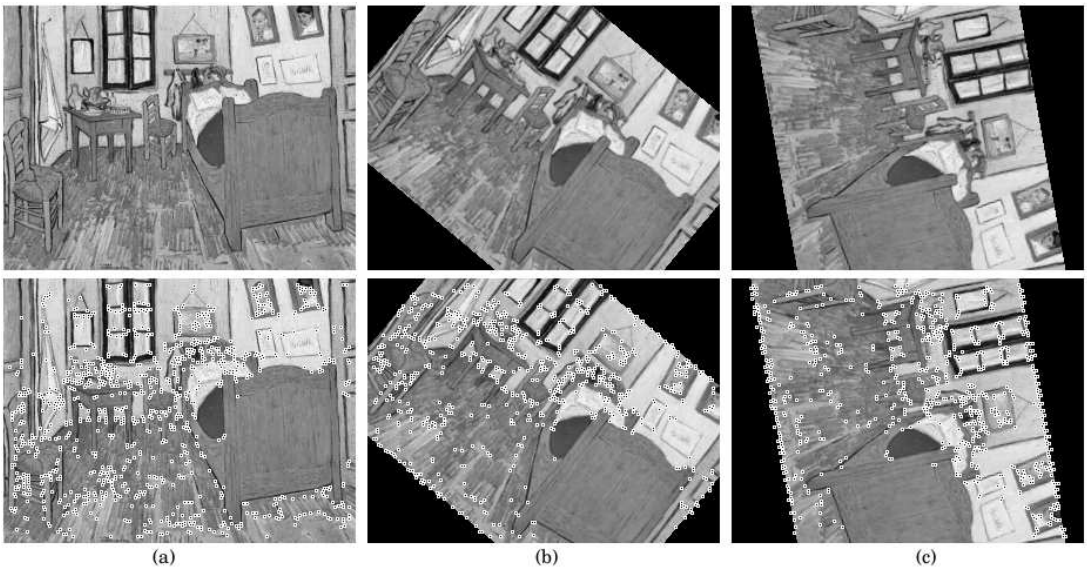


Figura A.1: Fila superior: Secuencia de rotación de una imagen (a) Imagen Original. (b) Rotación de 40 (c) Rotación de 80. Fila inferior: esquinas encontradas por el método de Harris-Stephens ($\sigma = 1$, $k = 0.04$, nms = 3x3)

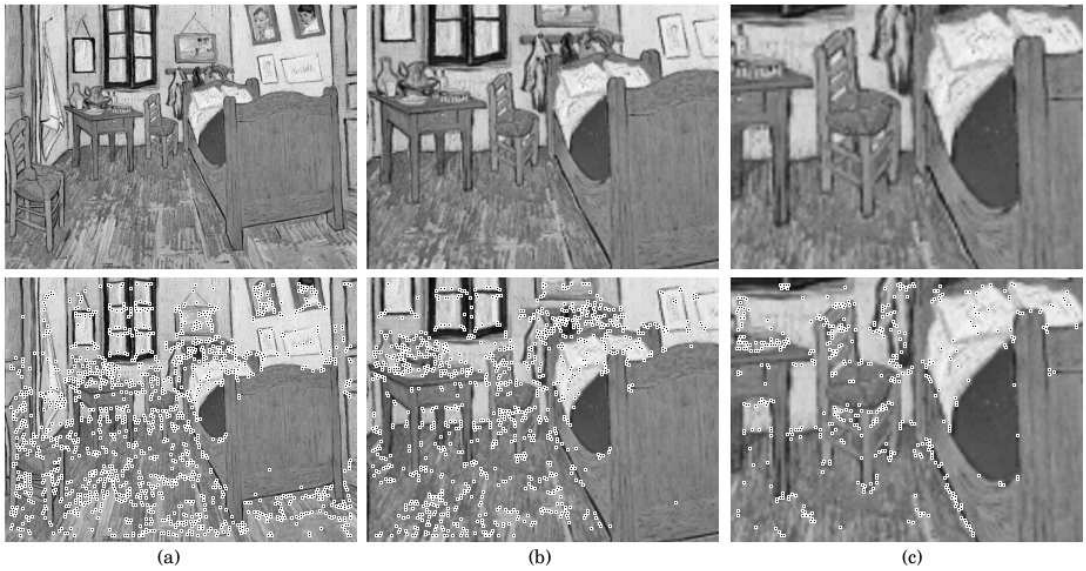


Figura A.2: Fila superior: Secuencia de escalamiento de una imagen (a) Imagen Original. (b) Escala 1.5 (c) Escala 2.5. Fila inferior: esquinas encontradas por el método de Harris-Stephens ($\sigma = 1$, $k = 0.04$, nms = 3x3)

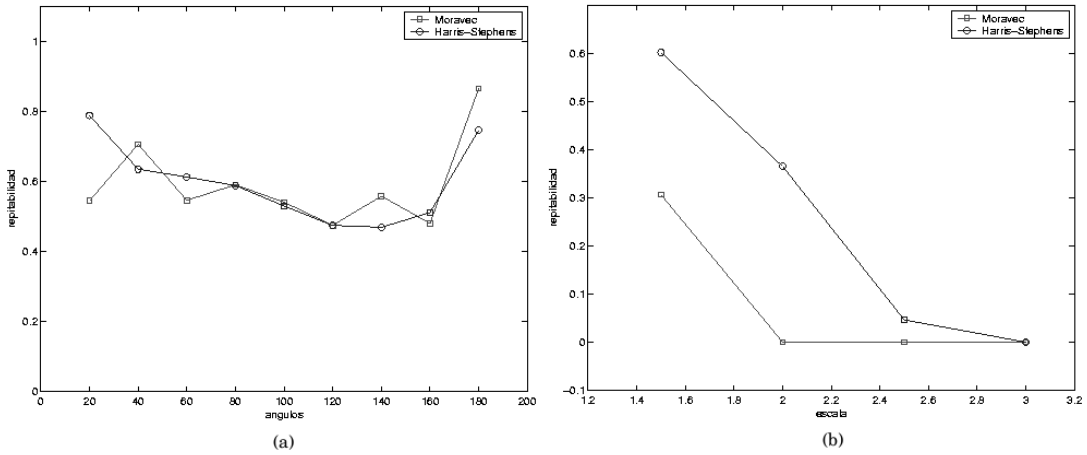


Figura A.3: Gráfica comparativa de repitabilidad para los métodos de Moravec y Harris. (a) Valores de repitabilidad obtenidos por ambos métodos en la secuencia de rotación. (b) Valores de repitabilidad obtenidos por ambos métodos en la secuencia de escala.

Apéndice B

Estimación de la matriz de Proyección

En este apéndice se describe un método para estimar una matriz de proyección asumiendo que se cuenta con un conjunto de correspondencias entre puntos tridimensionales y puntos en la imagen ($\mathbf{X}_i \leftrightarrow \mathbf{x}_i$). El algoritmo se describe con más profundidad en [Hartley and Zisserman, 2004]. La matriz de proyección $\mathbf{P}$ es una matriz de 3x4 y relaciona las correspondencias de la siguiente manera:

$$\mathbf{x}_i = \mathbf{P}\mathbf{X}_i$$

El método usado es análogo al cálculo de la matriz fundamental o al de una homografía entre dos imágenes. Con cada correspondencia se pueden construir un sistema de dos ecuaciones linealmente independientes, que expresadas de manera matricial toman la siguiente forma:

$$\begin{bmatrix} \mathbf{0}^T & -w_i\mathbf{X}_i^T & y_i\mathbf{X}_i^T \\ w_i\mathbf{X}_i^T & \mathbf{0}^T & -x_i\mathbf{X}_i^T \end{bmatrix} \begin{pmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{pmatrix} = \mathbf{0}$$

donde $\mathbf{x}_i = (x_i, y_i, w_i)$ y $\mathbf{p}_i$ representa la $i-$ésima fila de la matriz $\mathbf{P}$.

Para n correspondencias contamos con una matriz $\mathbf{A}$ de $2n$ x 12. La matriz de proyección tiene 12 entradas, pero al ser una matriz homogénea (definida hasta una escala), cuenta sólo con 11 grados de libertad. De esto se sigue que son necesarias 11 ecuaciones para determinarla y por lo tanto un conjunto mínimo de 6 correspondencias.

Un aspecto importante es la normalización de las correspondencias antes de resolver el sistema de ecuaciones. La normalización de las coordenadas de los puntos bidimensionales (puntos de la imagen) se realiza de la misma manera que en el caso del método para encontrar la matriz fundamental, trasladando los puntos de manera que su centroide sea el origen y escalándolos de manera que su distancia al origen sea $\sqrt{2}$. En el

caso de los puntos tridimensionales un tipo similar de normalización se puede llevar a cabo si no existe una gran variación entre las profundidades[1] de los puntos, de existir esta variación otro tipo de normalización debe ser utilizado. De esta manera trasladamos el centroide de los puntos tridimensionales al origen y los escalamos tal que su distancia al origen sea $\sqrt{3}$. Una vez normalizados los puntos y creada la matriz $\mathbf{A}$, la solución del sistema se encuentra como el vector singular correspondiente al valor singular más pequeño de la matriz $\mathbf{A}$, el cual se encuentra de manera sencilla mediante SVD (*Descomposición de valores singulares*). Un resumen del algoritmo para estimar la matriz de proyección se observa en el Cuadro B.1.

Objetivo:
Dadas $n \geq 6$ correspondencias $\mathbf{X}_i \leftrightarrow \mathbf{x}_i$), estimar la matriz de proyección $\mathbf{P}$.

Algoritmo:

1. **Normalización.** Usa una transformación de similitud $\mathbf{T}$ para normalizar los puntos en la imagen y otra transformación $\mathbf{U}$ para normalizar los puntos tridimensionales. Los puntos normalizados son: $\tilde{\mathbf{x}}_i = \mathbf{T}\mathbf{x}_i$ y $\tilde{\mathbf{X}}_i = \mathbf{U}\mathbf{X}_i$

2. **Solución Lineal.** Forma la matriz $\mathbf{A}$ de $2n$ x 12 (caso mínimo $11x12$) a partir de las correspondencias $\tilde{\mathbf{X}}_i \leftrightarrow \tilde{\mathbf{x}}_i$. Encuentra la matriz de proyección $\tilde{\mathbf{P}}$ como la solución al sistema $\mathbf{A}\mathbf{p} = \mathbf{0}$ obtenida como el vector singular correspondiente al valor singular más pequeño de $\mathbf{A}$ (mediante SVD).

3. **Denormalización.** La matriz de proyección para las coordenadas originales esta dada por $\mathbf{P} = \mathbf{T}^{-1}\tilde{\mathbf{P}}\mathbf{U}$.

Cuadro B.1: Método lineal para la estimación de la matriz de Proyección.

[1] Nos referimos a la profundidad relativa con respecto a la cámara