

# CONSTRUCCIÓN DE MOSAICOS DE IMÁGENES

**TESIS**

Que para obtener el grado de  
**MAESTRÍA EN CIENCIAS DE LA INGENIERÍA ELÉCTRICA**

presenta

**José Luis Medina Murillo**

**Félix Calderón Solorio**

**Director de Tesis**

Universidad Michoacana de San Nicolás de Hidalgo

Junio 2007



# Resumen

Cuando se requiere cubrir ángulos amplios en la captura de un escenario o se necesitan digitalizar grandes panoramas, la limitación en el campo visual de una cámara no proporciona todo el campo de visión necesario para algunas aplicaciones sobre imágenes. En estos casos una opción es la creación de mosaicos de imágenes. Un mosaico de imágenes consiste en una composición de varias imágenes relacionadas por una misma escena, para formar una imagen panorámica.

En este trabajo se realiza la construcción de mosaicos de imágenes, utilizando registro en dos etapas, como un primer paso realizamos el registro de pares de imágenes utilizando puntos correspondientes, obteniendo así una transformación inicial que se utilizará con la segunda técnica de registro basada en tono de gris. De este modo se estima la transformación final para proceder con la construcción del mosaico panorámico.

En el presente trabajo de tesis planteamos un desarrollo pocas veces utilizado para la construcción de los mosaicos de imágenes, en primer lugar se calcula la homografía que relaciona la escena actual con la escena referencia haciendo uso de técnicas de correspondencia entre puntos. Para esta primera estimación utilizamos el método RANSAC que es el estado del arte entre los métodos robustos como muestra Zisserman en [Hartley04], como esta primera estimación puede estar influenciada negativamente por pequeños errores, se plantea el uso de un nuevo algoritmo de registro con el fin de mejorar la precisión de la estimación. Este planteamiento hace uso del registro en tono de gris minimizando una función de costo (SSD-ARC), usando el método de optimización Gauss Newton Levenberg Marquardt para minimizar la función de error SSD-ARC, esta función se desenvuelve muy bien ante la presencia de datos atípicos, realizando este proceso se mejora considerablemente la estimación realizada en el primer paso de nuestro algoritmo de construcción.



# Abstract

When it is needed to cover wide angles in the apprehension of a scene or it is necessary to digitize big panoramas, the limitation in the visual field of a camera and its lens do not provide the whole field of vision necessary for any applications on images. In these cases an option is the creation of mosaics of images. A mosaic of images consists of a composition of several images related by the same scene, to form a panoramic image.

In this work there is realized the construction of mosaics of images, using two technologies. I register as the first step we realize the record of couples of images using corresponding points, obtaining this way an initial transformation that was in use in the second step of the construction, using technologies of based record tone of grey, thus the final transformation is estimated for the construction of the mosaic. The mosaics are associated by a counterfoil of transformation that joins the image of entry with the later images that are indexed to the first one. We introduce a representation that associates the transformation proyectiva, which is, the mas completes of the transformations.

We present a slightly common exposition for the construction of the mosaics, first there is calculated the homografia that relates the current scene to the scene indexes using technologies of correspondence between points, this first estimation can be influenced negatively by small mistakes, there appears the use of a new algorithm of register with of improving the precision of the estimation, this exposition uses the record based on tone of grey minimizing a function of cost (SSD-ARC), which is unrolled very well before the presence of atypical information, using a combination of the method of optimization Gauss Newton Levenberg Marquardt to minimize the function of mistake SSD-ARC and consequently to obtain a good estimation of the final mosaic.



# Contenido

Resumen . . . . .	III
Abstract . . . . .	V
Contenido . . . . .	VI
Lista de Figuras . . . . .	XI
Lista de Tablas . . . . .	XIII
Lista de Algoritmos . . . . .	XV
Lista de Símbolos . . . . .	XVII
1. Introducción . . . . .	1
1.1. Antecedentes . . . . .	2
1.2. Planteamiento del Problema . . . . .	6
1.3. Objetivo . . . . .	7
1.4. Contribuciones . . . . .	7
1.5. Descripción de la Tesis . . . . .	10
2. Registro de Imágenes Basado en Puntos . . . . .	13
2.1. Selección de los Puntos Característicos . . . . .	14
2.1.1. Método de Kanade Lucas Tomasi . . . . .	16
2.2. El Problema de la Estimación y Emparejamiento de Puntos . . . . .	21
2.2.1. Métodos Basados en Correlación . . . . .	21
2.2.2. Estimación del Movimiento por Correlación . . . . .	23
2.3. Determinación de la Transformación entre Imágenes . . . . .	24
2.3.1. Métodos Robustos . . . . .	25
2.3.2. Métodos por Mínimos Cuadrados . . . . .	31
3. Registro de Imágenes Basado en Tono de Gris . . . . .	33
3.1. Introducción . . . . .	33
3.2. Conceptos Geométricos Básicos . . . . .	34
3.2.1. Interpolación Bilineal . . . . .	35
3.2.2. Coordenadas Homogéneas (CH) . . . . .	36
3.2.3. Transformación Proyectiva . . . . .	37
3.3. El Problema de Registro Paramétrico . . . . .	39
3.4. Definición de las Funciones de error . . . . .	39
3.4.1. Mínimos Cuadrados . . . . .	40

3.4.2.	Estimador Lorenziano (LE)	41
3.4.3.	Suma de Diferencias al Cuadrado con Condición de Reposo Adaptable	42
3.5.	Buscando el Conjunto de Parámetros para el Registro	45
3.5.1.	Modelo de la Matriz de Coordenadas	46
3.5.2.	Calculo de la Derivada de la Imagen	47
3.5.3.	Derivada de la Imagen Transformada	48
3.6.	Método de Optimización para la función de error	50
3.6.1.	Procedimiento de Minimización	51
3.7.	Método de Levenberg Marquardt	52
3.8.	Resultados Obtenidos con Registro de Imágenes	54
3.8.1.	Imagen Lena	54
3.8.2.	Imagen Lena con Datos Atípicos	56
3.8.3.	Imagen Sintética de la Tierra	56
3.8.4.	Imagen Sintética de la Tierra con Datos Atípicos	58
3.9.	Conclusiones	60
4.	Aplicación del Registro Automático: Mosaicos de Imágenes en 2-D	63
4.0.1.	Planteamiento General	63
4.1.	Construcción de mosaicos	64
4.1.1.	Resumiendo el Registro de Imágenes	66
4.1.2.	Tipos de Mosaicos	67
4.1.3.	Aplicaciones de los Mosaicos	68
4.2.	Desarrollo de la Aplicación para la Construcción de Mosaicos	69
4.2.1.	Obedeciendo Algunas Reglas para la Construcción de Mosaicos	70
4.3.	Relación entre Dos Imágenes	73
4.4.	Desarrollo Práctico del Trabajo	75
4.4.1.	Correspondencia entre Puntos	76
4.4.2.	Estimar la Matriz H	76
4.4.3.	Transformar y Empatar las Imágenes	76
5.	Resultados	83
5.1.	Registro Basado en Puntos Característicos	83
5.1.1.	Selección de Puntos Característicos	83
5.1.2.	Búsqueda de Correspondencias	83
5.1.3.	Determinación de la Transformación de la Imagen	85
5.2.	Construcción del Mosaico	86
5.3.	Mosaicos de Mas de Dos Imágenes	87
5.3.1.	Mosaico Posgrado FIE	87
5.3.2.	Mosaico de las Tarascas de Morelia	90
5.3.3.	Mosaico con Imágenes Extraídas de Google Earth ©	91
5.3.4.	Mosaico de la Plaza de San Pedro	94
5.4.	Otros Ejemplos	95
6.	Conclusiones	97
6.1.	Trabajo futuro	100





# Lista de Figuras

1.1. Imagen panorámica elaborada por nuestra aplicación con imágenes de satélite extraídas de google earth © . . . . .	2
1.2. Etapas de la construcción de los mosaicos de imágenes . . . . .	9
2.1. Resultado de calcular los valores propios de una imagen. Arriba derecha: valor propio mayor (aristas), debajo izquierda: valor propio menor (esquinas), debajo derecha: puntos esquina detectados aplicando una umbralización y una búsqueda de máximo locales. . . . .	17
2.2. Clasificación del cambio de los valores propios de la matriz C . . . . .	19
2.3. Ventana de correlación y búsqueda. . . . .	22
2.4. Estimación del desplazamiento hacia adelante y hacia atrás. . . . .	23
2.5. RANSAC: Datos atípicos y típicos . . . . .	27
2.6. Datos típicos estimados por RANSAC . . . . .	29
3.1. Ante cualquier transformación geométrica, los píxeles de la rejilla transformada (líneas discontinuas) no tienen por que coincidir con los de la rejilla destino . . . . .	34
3.2. Interpolación bilineal, la interpolación se hace primero en el sentido de $x$ obteniendo $T_1$ y $T_2$ y después se hace en el sentido de $y$ para obtener el color final $T$ . . . . .	36
3.3. Transformación proyectiva . . . . .	38
3.4. Ejemplo de las funciones de error . . . . .	44
3.5. Experimento con la imagen lena usando SSD-ARC-GNLM . . . . .	55
3.6. Experimento con la imagen lena usando SSD-ARC-GNLM con datos atípicos . . . . .	57
3.7. Experimento con la imagen tierra usando SSD-ARC-GNLM . . . . .	57
3.8. Experimento con la imagen tierra usando SSD-ARC-GNLM con datos atípicos . . . . .	59
4.1. Construcción de un mosaico . . . . .	65
4.2. Mosaico esférico de la luna, obtenida a partir de imágenes de Viking Orbiter, en 1980 . . . . .	71
4.3. Mosaico de la superficie de Marte, obtenido a partir de fotografías de Mars Pathfinder, de Octubre de 1998 . . . . .	71
4.4. Imagen panorámica de un edificio (secuencia tomada de [Paixao03]) reconstruido con nuestra aplicación . . . . .	71

4.5. Mapeo de un punto de un plano en dos imágenes diferentes . . . . .	74
4.6. Relación de las imágenes con una homografía . . . . .	75
4.7. Dimensión del mosaico final $605 \times 799$ usando imágenes de $640 \times 416$ , con la imagen inicial en ella . . . . .	77
4.8. Definición del área del mosaico, en relación a la imagen referencia (azul) y la imagen transformada (roja) . . . . .	78
4.9. (a) Mosaico con la Imagen Inicial, (b) Mosaico con la secuencia de imágenes 1 y 2, (c) Mosaico final . . . . .	81
5.1. Puntos característicos marcados con el método KLT . . . . .	84
5.2. Puntos correspondientes seleccionados por el coeficiente de correlación . . . . .	84
5.3. Puntos Correspondientes después de eliminar los datos atípicos por RANSAC . . . . .	85
5.4. Secuencia de imágenes del Posgrado de la FIE . . . . .	88
5.5. Imagen Panorámica del Edificio del Posgrado de la FIE estimado con GNLM-SSD-ARC . . . . .	89
5.6. Imagen Panorámica del Edificio del Posgrado de la FIE estimado con RANSAC y mínimos cuadrados . . . . .	89
5.7. Par de Imágenes de Dimensiones $640 \times 480$ para Construir el Mosaico Tarascas de Morelia . . . . .	90
5.8. Comparando el Método RANSAC (a) con la Función de Optimización SSD-ARC (b) con Imágenes Reales (Tarascas de Morelia) . . . . .	91
5.9. Imágenes Extraídas de Google Earth © . . . . .	92
5.10. Imagen Panorámica de las Imágenes Extraída de Google Earth © . . . . .	93
5.11. Secuencia de imágenes de la Fachada del Vaticano . . . . .	93
5.12. Imagen Panorámica de la Fachada del Vaticano . . . . .	94
5.13. Imagen Panorámica de un Edificio Elaborada de Derecha a Izquierda . . . . .	95
5.14. Imagen Panorámica de un Edificio Elaborada de Izquierda a Derecha . . . . .	96

# Lista de Tablas

3.1. Resultados para imagen Lena, usando LS, LE, SSD-ARC y RANSAC . . . . .	55
3.2. Resultados para imagen Lena, usando LS, LE, SSD-ARC y RANSAC con datos atípicos en la imagen . . . . .	58
3.3. Resultados para imagen Tierra, usando LS, LE, SSD-ARC, RANSAC con el método de optimización GNLM . . . . .	59
3.4. Resultados para imagen Tierra, usando las funciones LS, LE, SSD-ARC, RANSAC con el método de optimización GNLM, con datos atípicos . . . . .	60
5.1. Valores de los diferentes parámetros durante los experimentos. . . . .	87



# Lista de Algoritmos

1.	Algoritmo KLT . . . . .	20
2.	Algoritmo RANSAC: para el ajuste de una recta . . . . .	28
3.	Algoritmo RANSAC para estimar la Homografía para el registro de imágenes . . . . .	30
4.	Algoritmo Levenberg Marquardt . . . . .	53
5.	Algoritmo General para la Construcción de Mosaicos . . . . .	72
6.	Algoritmo General para Estimar la Matriz $H$ . . . . .	77
7.	Algoritmo para Conocer las Dimensiones del Mosaico Final y Colocar la Imagen Origen en Ella . . . . .	78
8.	Algoritmo para Conocer las Dimensiones del Mosaico Final y Colocar la Imagen Origen en Ella . . . . .	80



# Lista de Símbolos

---

$C(p)$	Matriz de valores propios
$D_x$	Derivada en x
$D_{xy}$	Derivada en y
$D_y$	Derivada en x, y
$\lambda$	Valor Propio
$Xs$	Conjunto de datos
$k$	índice de imágenes
$I_k(x, y)$	Imagen
CH	Coordenadas Homogéneas
$F_e$	Factor de Escala
SDC	Suma de Diferencias al Cuadrado
SDA	Suma de Diferencias Absolutas
TP	Transformación Proyectiva
$H_p$	Matriz de la Transformación Proyectiva
S	Conjunto
s	Subconjunto
$\tau$	umbral para KLT
$A_i$	Matriz del sistema de mínimos cuadrados
$b_i$	Vector del sistema de mínimos cuadrados
LS	Mínimos Cuadrados
LE	Estimador Lorenziano
SSD-ARC	Suma de Diferencias al Cuadrado con Condición de Reposo Adaptable
GNLM	Gauss Newton Levenberg Marquardt
MMC	Modelo de la Matriz de Coordenadas
MCD	Matriz de Corrección de las Derivadas
GNLM	Gauss Newton Levenberg Marquardt

---



## Capítulo 1

# Introducción

La realización de este trabajo tiene por base matemática la teoría de Geometría Proyectiva Plana. Estas transformaciones están relacionadas con la distorsión geométrica introducida cuando una cámara construye una imagen 2D a partir de una imagen 3D. Algunas propiedades geométricas son alteradas, por ejemplo, las líneas que eran paralelas no necesariamente seguirán siendo paralelas, otras se mantienen sin alterar por ejemplo, una línea recta continua siendo una línea recta como lo muestra Zisserman en [Hartley04].

Un mosaico de imágenes consiste en una composición de varias imágenes relacionadas por una misma escena, para formar una imagen panorámica. El ejemplo más sencillo es cuando se toma una serie de fotografías y se unen para reconstruir todo el panorama. Las imágenes pueden ser adquiridas por una cámara de vídeo e ir generando el mosaico al mismo tiempo. Las imágenes que se utilizan para generar un mosaico pueden ser adquiridas desde diferentes perspectivas y con diferente distancia a la cámara, siendo éste, el problema más general. Uno de los puntos a evaluar que se encuentra al realizar la reconstrucción por mosaicos, es precisamente los diferentes tamaños de las imágenes y la perspectiva desde las que son tomadas, lo que implica que las imágenes deben ser procesadas de manera conveniente para su unión como lo muestra arias en [Arias04]. En este capítulo se describen los antecedentes, el objetivo y los alcances de esta tesis. Se presentan diferentes temas relacionados con el registro de imágenes, y el procesamiento de imágenes que servirán de base para una mejor comprensión de los capítulos posteriores.

## 1.1. Antecedentes

Actualmente nos maravillamos con escenas tomadas por satélite en alta resolución como la figura 1.1, las cuales ejemplifican una gran superficie terrestre o diversidad de paisajes, aunque la tecnología en cámaras convencionales y cámaras digitales han evolucionado mucho, nos encantaría poder tomar una imagen de un paisaje sin que se pierdan detalles. En estos casos una opción para la solución de este conflicto es la creación de mosaicos de imágenes los cuales ayudan a formar vistas panorámicas, mapas, aplicaciones como super-resolución, realidad virtual, sistemas de información geográfica, monitores ambientales, entre otros como muestra Arias en [Arias04].



Figura 1.1: Imagen panorámica elaborada por nuestra aplicación con imágenes de satélite extraídas de google earth ©

La construcción de mosaicos de imágenes generan algunos problemas a solucionar. El primero de ellos es realizar el registro de un par de imágenes utilizando la correspondencia entre imágenes de la misma escena, para el cual se han desarrollado diferentes estrategias de solución. Una técnica muy utilizada para establecer la correspondencia entre dos imágenes es la de búsqueda de regiones similares como lo muestran en [Lane94],[Marapane92],[Domingo04]. En esta técnica se segmentan en cada una de las dos imágenes las regiones que sean de interés. Luego, se extraen las características de estas regiones y se comparan las características de las regiones de la imagen 1 con las características de las regiones de

la imagen 2. Aquellas regiones que tengan características similares serán entonces regiones correspondientes.

En este trabajo de tesis hay dos estrategias la primera de ellas consiste en utilizar la correspondencia entre puntos, esto puede realizarse mediante la selección de una serie de elementos característicos, que resulten fácilmente identificables y distinguibles en la imagen, caracterizándolos mediante un patrón. En unos casos se seleccionan puntos de la imagen que correspondan físicamente a elementos conocidos de la escena como lo muestra Atienza en [Atienza04]. Un método de selección de puntos característicos interesante es el operador de Moravec como lo muestra Moravec en [Moravec80]. El método selecciona un conjunto de puntos de interés que presenten una varianza local significativa. El procedimiento se basa en la definición de una medida de interés  $m(f, x, y, p)$  que evalúa el interés de un punto  $(x, y)$  para una imagen  $f$  obteniendo una medida de la varianza direccional en una pequeña ventana cuadrada de tamaño  $p \times p$  centrada en la posición  $(x, y)$ . Para obtener la medida de interés del punto  $(x, y)$  se comienza calculando las sumas de los cuadrados de las diferencias entre píxeles adyacentes, en cuatro direcciones (horizontal, vertical y las dos diagonales). Como otros procedimientos de selección, el número final de puntos seleccionados no puede ser definido de antemano, sino que depende de las características de la imagen y el número de máximos locales obtenidos tras la aplicación de la medida de interés.

Otro operador similar, desarrollado específicamente para la detección de esquinas y adecuado por tanto para la selección de puntos de interés es mostrado por Kitchen y Rosenfeld en [Kitchen82]. Este operador está basado en derivadas parciales de la imagen, de primer y segundo orden. Las esquinas se localizan entonces como los máximos locales de este operador.

Para la selección de puntos de control este trabajo de tesis se basa en los trabajos de Tomasi y Censi en [Tomasi98], [Censi99], en los cuales proponen un planteamiento más analítico al escoger los puntos característicos adecuados, considerando de forma más global el problema del seguimiento. En general, todos ellos se basan en los trabajos de Lucas y Kanade como lo muestran en [Lucas81] para movimiento traslacional, o los trabajos de Shi y Tomasi como lo muestran en [Shi94] para movimiento afín. En estos trabajos se propone un método de selección de puntos característicos denominado KLT (Kanade Lucas Tomasi)

que en lugar de escoger los puntos por un criterio arbitrario de textura, selecciona aquellos para los que se demuestra analíticamente que son capaces de optimizar la precisión del seguimiento, es decir, aquellos cuyas correspondencias pueden ser establecidas con mayor fiabilidad.

Una vez que tenemos el conjuntos de puntos de control, se procede a determinar los puntos correspondientes entre este conjunto de puntos de control, para este problema se plantean estrategias como una búsqueda exhaustiva como muestra Stiefelhagen en [Stiefelhagen96] que es un método habitual para localizar el desplazamiento que hace mínimo el criterio de similitud (o disparidad) seleccionado, consiste en realizar una búsqueda exhaustiva evaluando la medida de similitud para todos los posibles valores de desplazamiento  $d$  dentro de un rango. Este intervalo viene definido por el establecimiento de una ventana de búsqueda  $A$  en la imagen  $f_k$ . El método de estimación de desplazamiento por búsqueda exhaustiva resulta sencillo de implementar y asegura que el desplazamiento obtenido minimiza la función de disimilaridad para el área seleccionada, que puede tener un tamaño arbitrario. El inconveniente es el elevado costo temporal, proporcional al tamaño de la ventana de búsqueda, debido a la necesidad de evaluar el criterio de similitud para todos y cada uno de los posibles vectores de desplazamiento en el intervalo considerado.

Otra estrategia para la estimación de correspondencias es la estimación de movimiento por correlación [Gennery80]. Este método considera que todos los puntos de la vecindad de píxeles alrededor del punto característico sufren el mismo desplazamiento, por lo que la apariencia de la región no cambia de una imagen a otra. Se considera por tanto un modelo de movimiento puramente traslacional. El método de estimación de los dos parámetros de desplazamiento consiste en determinar la posición correspondiente del punto característico para la cual se maximiza algún criterio de similitud entre regiones de píxeles. Son varias las posibilidades que pueden encontrarse en trabajos de seguimiento a la hora de establecer el criterio de similitud. La medida de similitud adoptada por este trabajo de tesis es el coeficiente de correlación mostrado por González en [Gonzalez95].

Una vez determinados los desplazamientos sufridos por los puntos característicos en la nueva imagen, se plantea a continuación la estimación de los parámetros de movimiento. Esta etapa tiene por objetivo determinar los parámetros de traslación, rotación, escalamiento

que se deducen de las nuevas observaciones de las posiciones de los puntos en la imagen y la consideración del modelo geométrico y el modelo de formación de la imagen (ortogonal, perspectivo, etc.). En este caso el número de parámetros a determinar dependerá del modelo de movimiento que se considere (2 para un modelo traslacional, 4 para un modelo traslación-rotación, 6 en el caso de un modelo afín y 8 para un modelo perspectivo u homográfico).

En general se trata de solucionar un problema de regresión, donde se pretende determinar el valor óptimo de una serie de parámetros de forma que se maximice algún índice de bondad en el ajuste. La estimación de la transformación proyectiva sufrida por una imagen es a menudo un problema sobredeterminado, porque habitualmente hay más medidas disponibles que las estrictamente necesarias. Es muy frecuente la introducción de técnicas de estimación por mínimos cuadrados, así como el empleo de métodos de estadística robusta.

Un gran número de trabajos utilizan todas las medidas disponibles mediante algún tipo de criterio de ajuste por mínimos cuadrados como muestran Gennery, Lowe y Colombo en [Gennery92], [Lowe92], [Colombo99] respectivamente. Los métodos de regresión por mínimos cuadrados ofrecen buenos resultados cuando el ruido en las medidas es pequeño o se ajusta correctamente al modelo gaussiano esperado. Sin embargo muchos autores señalan que el modelo gaussiano para modelar los errores que se presentan típicamente en los resultados de los buscadores de puntos característicos resulta poco adecuado como lo muestra Gee en [Gee96].

Para evitar estos inconvenientes otros métodos alternativos usan conceptos de estadística robusta durante la estimación los parámetros de movimiento como lo muestran Fischler y Gee en [Fischler81], [Gee96], de forma que no se vean influenciados por la aparición de este tipo de datos atípicos o medidas discrepantes (medidas que violan el modelo de distribución esperado). Una clase de métodos robustos frecuentemente utilizados para estimación en algoritmos de seguimiento son los denominados métodos de información mínima. Se trata de utilizar un subconjunto mínimo de las medidas disponibles para realizar la estimación de la pose. Se basan en la hipótesis de que si se encuentra un subconjunto suficientemente preciso, el resto de medidas pueden ser ignoradas y de esta forma se evita que medidas erróneas puedan afectar negativamente a la estimación de la posición.

Al tener una primera aproximación de la transformación proyectiva basada en selección de puntos característicos abordaremos el segundo problema que consiste en realizar el registro de un par de imágenes en escala de gris, este planteamiento se maneja en algunos trabajos como muestran Atienza y Calderón en [Atienza04], [Calderón03], [Calderón06]. Contando con la primera aproximación de nuestros parámetros de transformación podremos utilizar el Registro Paramétrico en el cual por medio de unos pocos parámetros se logra hacer una deformación de toda la imagen. El modelo Proyectivo que se definirá en el siguiente capítulo también se le puede aplicar un modelo no Paramétrico en el cual tenemos un campo de deformaciones que aplicamos en forma independiente a cada punto de la imagen, al utilizar esta técnicas de registro estaremos minimizando una función de error por medio de algún método de optimización como Newton o Levenberg Marquardt, este método se utiliza en conjunto con la primera estimación de la transformación proyectiva realizada ya que esto asegura que no nos quedaremos estancados en un mínimo local como lo muestra Calderón en [Calderón03].

## 1.2. Planteamiento del Problema

La mayoría de los esquemas de construcción de mosaicos de imágenes reportan el uso de técnicas de registro basado en correspondencias de puntos o el registro basado en escala de gris, una de las dos técnicas a la vez como muestran Miravet y Paixao en [Miravet98], [Paixao03]. Las debilidades identificadas en las estrategias propuestas para resolver el problema de la construcción de mosaicos, hacen de este un campo abierto para nuevas contribuciones.

Concretamente se requiere de una estrategia que no requiera la ayuda humana como muestran Miravet y Arias en [Miravet98], [Arias04], se pretende que sea completamente autónoma para resolver la construcción de mosaicos de imágenes. Además es necesario que la estrategia propuesta resuelva el movimiento mas cercano al presentado por una cámara (movimiento proyectivo).

### 1.3. **Objetivo**

El objetivo general de la tesis es proponer una estrategia para la construcción de mosaicos de imágenes de operación totalmente autónoma, basada en dos técnicas de registro, la primera basada en la selección de puntos característicos, estimando sus puntos correspondientes en las imágenes y la segunda basada en el registro de tono de gris.

Dicho objetivo se concreta en una serie de objetivos particulares que implican a su vez el desarrollo de una serie de tareas, que se detallan a continuación:

- Estudio de los diferentes planteamientos establecidos en la literatura con relación al problema general del seguimiento basado en puntos característicos y sus aplicaciones [Tomasi98].
- Evaluación de las diferentes técnicas estudiadas y selección de aquellas que se consideren más convenientes atendiendo a las características de los problemas específicos que se pretenden resolver.
- Desarrollo de técnicas adecuadas para la resolución de los subproblemas planteados, tales como la selección de los puntos característicos, la determinación de la posición 2-D con un modelo proyectivo, determinación de la posición mediante técnicas de registro de imágenes.
- Justificación y evaluación experimental de las soluciones propuestas.
- Desarrollo y evaluación de una aplicación de construcción de mosaicos de imágenes basada en la información de alineación obtenida con métodos de seguimiento basado en puntos característicos y las técnicas de registro de imágenes paramétrico. Se pretende que este planteamiento aporte ventajas en cuanto a reducción del tiempo de proceso respecto a otros métodos alternativos que no hacen uso de técnicas de seguimiento.

### 1.4. **Contribuciones**

Las contribuciones más significativas que se aportan en el presente trabajo tienen relación en la elaboración y aplicación de técnicas específicas que optimizan problemas con-

cretos, las contribuciones más destacables se detallan a continuación.

- Se establece un planteamiento general de construcción de mosaicos de imágenes de control totalmente autónomo, que integra dos técnicas de registro: el realizado a partir de las posiciones de los puntos característicos y el registro de las imágenes basado en tono de gris ejecutado por un método de optimización como lo es Levenberg-Marquardt como muestra Chapra en [Chapra99] el cual optimiza un función de error SSD-ARC es robusta a datos atípicos la cual se estudiará en la sección 3.4.3 como muestra Calderón en [Calderón06]. Este planteamiento demuestra que se obtienen soluciones ventajosas con respecto a soluciones basadas exclusivamente en una de las dos técnicas.
- Se realiza una evaluación experimental del procedimiento de determinación de la transformación proyectiva de la imagen. Se aporta un estudio experimental de este procedimiento, en sus dos diferentes variantes. Se comparan sus prestaciones con las ofrecidas por otros de uso habitual en tareas de esta índole: un método por mínimos cuadrados y el método RANSAC que se estudiara en la sección 2.3.1, que cabe clasificar en el grupo de los métodos robustos al igual que la función a optimizar SSD-ARC que se estudiara en la sección 3.4.3.
- Se propone la utilización de la función de error SSD-ARC que se estudiará en la sección 3.4.3 como muestra Calderón en [Calderón06], la cual es robusta para realizar el registro de las imágenes, con lo que se garantiza una buena estimación final de la transformación proyectiva para utilizarla en la construcción de un mosaico de imágenes.
- Se establece un modelo proyectivo con técnicas de registro de imágenes. El modelo consiste en encontrar un conjunto de parámetros (en este caso 8 de la transformación proyectiva), que fusione dos imágenes las cuales describen el mismo escenario, aplicando a una de ellas una serie de transformaciones espaciales.

En la Figura 1.2 se muestra, de forma esquemática, las etapas que conforman la construcción de un mosaico de imágenes para este trabajo de tesis.

En resumen se puede considerar el siguiente procedimiento para la construcción de mosaicos de imágenes. Como muestra la figura 1.2 el paso básico es adquirir las imágenes ya

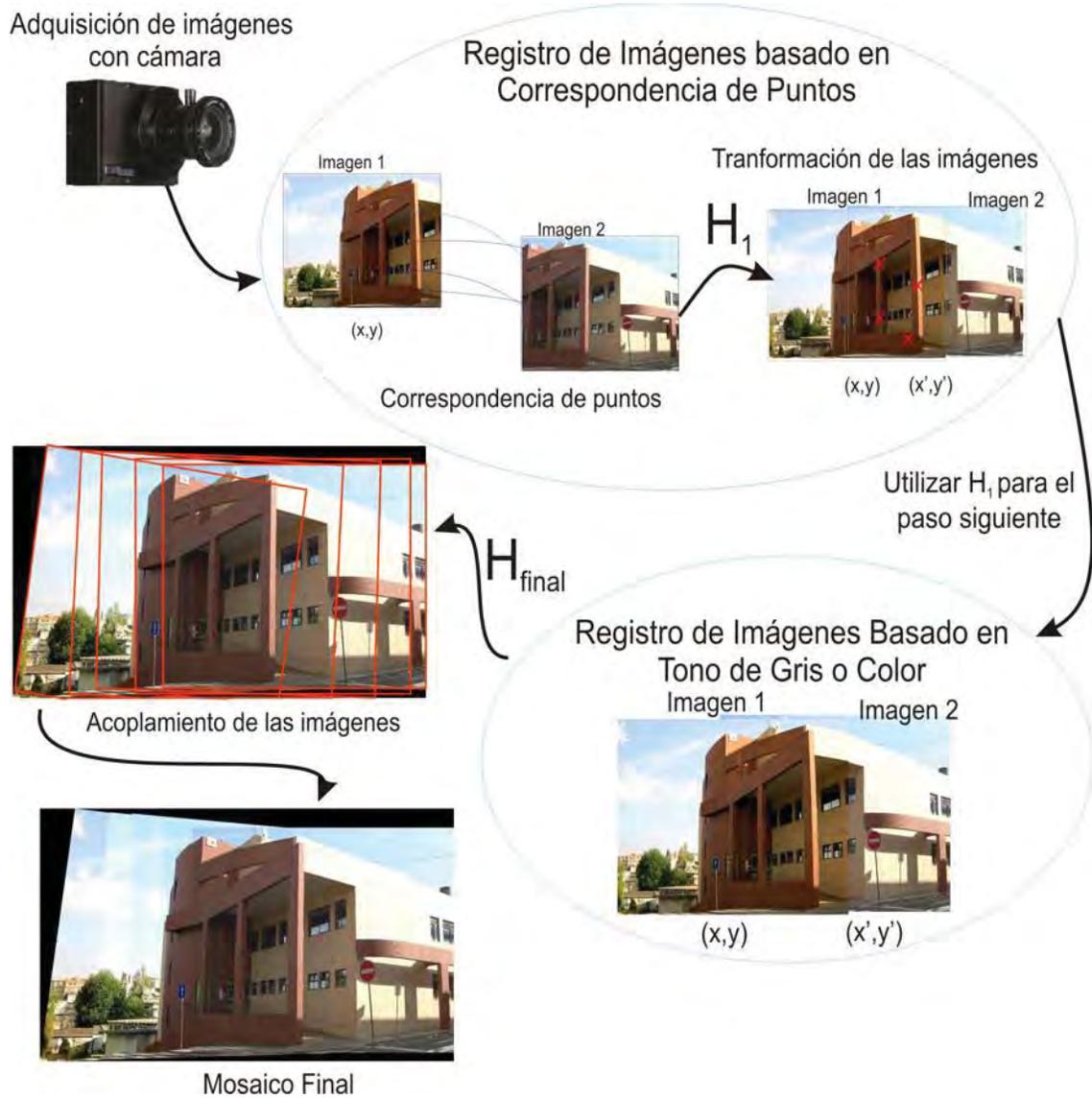


Figura 1.2: Etapas de la construcción de los mosaicos de imágenes

sea con una cámara digital o una vídeo cámara y después obtener los frames o cuadros de las imágenes (fotografías), lo siguiente que hay que hacer es realizar el registro de imágenes en dos pasos el primer paso usando registro basado en puntos correspondientes, para la selección de puntos característicos se utiliza un método automático de selección como lo es KLT (Kanade Lucas Tomasi) el cual se describirá en la sección 2.1.1, después de tener la selección de puntos, se realiza una selección putativa de puntos correspondientes por medio del coeficiente de correlación que se estudiara en la sección 2.2.2, después de tener la correspondencia putativa se utiliza un método robusto como lo es RANSAC estudiado en la sección 2.3.1 para eliminar datos atípicos que puedan provocar errores al estimar la transformación proyectiva de la imagen, después de eliminar los datos atípicos se utiliza el método de mínimos cuadrados para estimar una primera aproximación  $H_1$ .

El segundo paso se realiza usando registro basado en tono de gris para mejorar la primer aproximación de la homografía y calcular  $H_{final}$ , para el cual utilizaremos un método de optimización como Levenberg Marquardt en conjunto con una función de error robusta a datos atípicos como lo es SSD-ARC. Esta función tiene la característica de presentar un muy buen comportamiento ante la presencia de datos atípicos, al utilizar este procedimiento poco común en la literatura se mejora la estimación de la transformación proyectiva final y por consiguiente lograr una muy buena construcción del mosaico final [Calderón06].

## 1.5. Descripción de la Tesis

El presente trabajo esta organizado de la siguiente manera:

El Capítulo 1 presenta una primera vista del tema de técnicas de registro de imágenes para construcción de mosaicos, ente se aborda una introducción al tema de registro y un planteamiento del problema a desarrollar en este trabajo de tesis.

El Capítulo 2 da un seguimiento a las técnicas de registro basadas en puntos correspondientes que se utilizaran en este trabajo, se analizan los métodos de selección de puntos característicos, correspondencia entre puntos, estimación de la transformación de una imagen, analizando la técnica conocida como RANSAC, por medio de las técnicas analizadas en este capitulo se estimara una Homografía inicial para nuestro trabajo ya que en la literatura

como muestra Zisserman en [Hartley04] se reconoce a RANSAC como el estado del arte en este tipo de trabajos, todo esto buscando técnicas que nos permitan darle un enfoque automático a la construcción de mosaicos de imágenes para nuestro trabajo final que será la construcción de mosaicos de imágenes automáticos .

El Capítulo 3 presenta el registro basado en tono de gris, este tipo de registro presentado hace uso de una primera aproximación de la homografía que será calculada por las técnicas analizadas en el capítulo 2, ya que de lo contrario es propenso a quedarse en mínimos locales al tener que minimizar una función de error robusta a datos atípicos, la cual se optimiza por medio de un método de minimización como lo es Levenberg Marquardt, también se analiza el modelo matemático planteado para este trabajo.

El Capítulo 4 detalla la aplicación final: Mosaicos de Imágenes en 2-D. Al tener estimada la Homografía final para la construcción de lo mosaicos, en el capítulo 4 se detalla el proceso de empatar las imágenes y la construir del mosaico final.

El Capítulo 5 muestra algunos de los resultados obtenidos en este trabajo de tesis, por ejemplo que ocurre cuando las imágenes presentan una gran cantidad de datos atípicos, así como la comparación entre nuestro trabajo y el método RANSAC el cual es el estado del arte en este tipo de trabajo sobre imágenes, mejorando nuestro trabajo por mucho la solución arrojada por el método RANSAC.

El Capítulo 6 presenta las conclusiones, las aportaciones del presente trabajo y algunos de los trabajos a futuro que se pueden desarrollar con las técnicas de registro analizada.



## Capítulo 2

# Registro de Imágenes Basado en Puntos

El registro de imágenes es el proceso de calcular las correspondencias entre dos o más imágenes de una misma escena tomadas generalmente en tiempos diferentes, desde distintas orientaciones y/o captadas por varios tipos de sensores.

Aunque existen una multitud de métodos para registrar imágenes, bien generales o sólo para tipos de imágenes específicos como muestra Brown en [Brown92], es posible agruparlos en dos o tres clases, dependiendo del grado de intervención humana que precisen para su funcionamiento. Las dos clases principales serían las de registro manual y automático. Puede hablarse, en algunos casos, de registro semiautomático. Esta clasificación corresponde, además, con el orden histórico en el desarrollo de los diferentes procedimientos de registro existentes.

En la mayoría de las aplicaciones actuales el registro se realiza de forma manual; sólo en algunos casos especiales se dispone de métodos automáticos. En el proceso de registro manual, es un operario humano el que identifica parejas de puntos coincidentes en las imágenes a registrar, a partir de las cuales se determinarán las funciones de deformación de una de las imágenes para que quede registrada con la otra como muestra López en [López99].

En el registro automático todo el proceso de registro se realiza sin intervención humana como muestra López en [López99]. Hasta ahora sólo puede hablarse de registro

automático en algunas aplicaciones concretas, en las que existe un modelo o entorno de laboratorio controlado. Las aplicaciones en el que interviene el registro automático es, por ejemplo, el caso de la estereoscopia; la estereoscopia o fotografía estereoscópica es la producción de imágenes fotográficas capaces de dar una impresión de profundidad parecida a la que percibe la visión normal o binocular en la que el contenido de las imágenes a registrar es muy similar.

Cuando las imágenes a registrar son muy diferentes, el registro automático resulta complicado, y suele ser necesario recurrir al registro manual. También puede ocurrir que, por su complejidad, sea difícil realizar el registro de forma completamente autónoma, y casi siempre existe algún tipo de umbral que debe ser fijado de antemano por un operador humano. En aquellos casos en los que la supervisión humana sea considerable, puede hablarse de registro semiautomático.

El problema de registro de imágenes consiste en encontrar, la transformación geométrica en este caso transformación proyectiva que empate dos imágenes que describen el mismo escenario, aplicando a una de ellas una serie de transformaciones espaciales. En otras palabras el registro de imágenes trata de encontrar un conjunto de parámetros que empate la imagen  $I_1$  en la imagen  $I_2$ .

## 2.1. Selección de los Puntos Característicos

En el capítulo anterior se dijo que un sistema basado en puntos característicos debe resolver una serie de etapas comunes. En la bibliografía revisada como en [Tomasi98], [Lucas81], [Amil03], [Fischler81], se encontró que cada unas de las tareas generales como lo son la selección de puntos característicos, la correspondencia putativa de los puntos, y la estimación de la posición de la imagen o transformación de la imagen puede hacerse mediante el uso de diferentes técnicas. La selección de unas u otras, o la necesidad de desarrollar procedimientos específicos estarán estrechamente relacionados con la aplicación a la que se destine el sistema. Diferentes aplicaciones plantean necesidades distintas, por ejemplo en cuanto a la necesidad de la automatización de procedimientos como el de selección de los puntos característicos.

Para la selección de puntos de control es indispensable la detección automática de pares de puntos de control en las imágenes, es esencial para que el sistema tenga éxito puesto que son los que establecen las relaciones geométricas entre las imágenes. Estos puntos, en un número suficiente para hacer frente a los inevitables problemas de correspondencias erróneas, han de ser lo más precisos posible y estar uniformemente distribuidos por toda la imagen.

Para construir un mosaico de imágenes se obtienen puntos de correspondencia entre las imágenes sucesivas. Con base en estos puntos se encuentra la transformación que une las imágenes en la secuencia. Gran parte del éxito para realizar una buena construcción de mosaicos está en la correcta selección de los puntos o la relación entre las imágenes. Usualmente se generan las transformaciones en una vecindad de la respuesta y se calcula un índice de error para seleccionar la que tenga el menor error. Las imágenes se alinean de acuerdo a los resultados obtenidos por la homografía como muestra Szeliski en [Szeliski97]. Esto puede provocar que el resultado tenga un error acumulado que hace que la primera y última imagen sea registrada pobremente y el mosaico resultante no tenga la imagen adecuada.

La primera tarea que debe realizarse consiste en la selección de un número  $n$  de puntos característicos en la imagen. El criterio de selección de estos puntos suele ser muy dependiente de la aplicación. Siempre se exigirá que los puntos seleccionados posean características que favorezcan su localización, como son una buena distinción, y algunas propiedades que muestra Weng en [Weng93], en este artículo se resumen las características que deben reunir estos puntos de interés en cuatro propiedades: distinción, unicidad, invarianza y estabilidad.

La **distinción** significa que un punto debe ser diferente de sus vecinos inmediatos. Esto excluye la selección de puntos pertenecientes a áreas uniformes de la imagen o bordes rectilíneos, ya que las distintas partes de un borde rectilíneo no pueden diferenciarse entre sí. Es lo que se conoce como “problema de la apertura” (aperture problem) como muestra Hartley en [Hartley04].

La **unicidad** significa que un punto debería ser distinguible globalmente, es decir, idealmente no debería parecerse a ningún otro punto de la imagen. Para asegurar que los puntos cumplan esta propiedad debería evitarse la selección de elementos que, aunque re-

sulten localmente distinguibles, aparezcan repetidamente en la imagen. Este tipo de puntos repetidos afecta negativamente a los procesos de búsqueda de correspondencias, al provocar situaciones de confusión.

La *invarianza* de un punto se refiere a que la apariencia de éste no debería variar a consecuencia de las distorsiones geométricas o radiométricas que se prevé que puedan ocurrir, debido a las características del objeto y su movimiento o en relación con la iluminación de la escena o el proceso de formación de la imagen.

La *estabilidad* se refiere, por último, a que la apariencia del punto debería ser invariante respecto al punto de vista. Puntos interesantes de la imagen deben corresponder a puntos de interés del objeto. Deben excluirse puntos que resultan del cruce de bordes de distintos objetos o de un objeto y el fondo, por ejemplo.

Existen procedimientos automáticos para seleccionar puntos con buenas propiedades. En general, los procedimientos de selección automática de puntos característicos buscan puntos con buena distinción, basándose en alguna medida de varianza local de la imagen o en alguna característica como la calidad de esquina o borde.

### 2.1.1. Método de Kanade Lucas Tomasi

El método de selección de puntos característicos Kanade Lucas Tomasi (KLT) utilizado en este trabajo se basa en la selección de puntos con buenas propiedades que lo distinguan, considerando la estimación de desplazamiento o búsqueda de correspondencia que hará uso de puntos seleccionados. Si bien existen diferentes métodos para efectuar esta búsqueda, por su robustez y precisión se ha optado por el algoritmo KLT. Tomasi y Lucas proponen un planteamiento analítico a la hora de escoger los puntos característicos adecuados en [Tomasi98] [Lucas81]. En general, este método está basado en el trabajo realizado por Lucas y Kanade mostrado en [Lucas81], para movimiento traslacional o de Shi y Tomasi mostrado en [Tomasi98], para movimiento afín.

Analizaremos la forma en que trabaja el método KLT para la selección de puntos característicos. Aparentemente este método de selección de puntos supone un planteamiento totalmente diferente a los que realizan la selección basándose en criterios de textura o selección de esquinas. Sin embargo este método se encuentra estrechamente relacionado con

los que hacen selección por textura o selección de esquinas, ya que está basado en la estimación de gradientes de la imagen y una matriz  $C$ , que nos da una ventana cuadrada de la imagen de tamaño  $(2N + 1) \times (2N + 1)$  como muestra Zisserman en [Hartley04]. Para una imagen uniforme la matriz  $C = 0$  y sus dos valores propios son cero  $\lambda_1 = 0$  y  $\lambda_2 = 0$ , lo cual correspondería a una región de la imagen con una intensidad prácticamente constante y no sería tomada en cuenta. La imagen de un borde produce  $\lambda_1 > 0, \lambda_2 = 0$ , donde el valor propio asociado con el valor propio positivo es la normal al borde, por lo cual tampoco serían seleccionadas ventanas que dieran un alto valor para uno de los valores propios pero un valor pequeño para el otro, esta combinación corresponde a ventanas con un patrón de intensidad orientado en una dirección única. Una esquina produce dos valores propios positivos. Cuanto mayores sean mayor será el contraste en la imagen producido por la esquina  $\lambda_1 > 0, \lambda_2 > 0$ , en cambio, dos valores propios altos suelen corresponder a esquinas o regiones de textura alta con distribución no orientada y darían lugar a que la ventana fuera seleccionada.

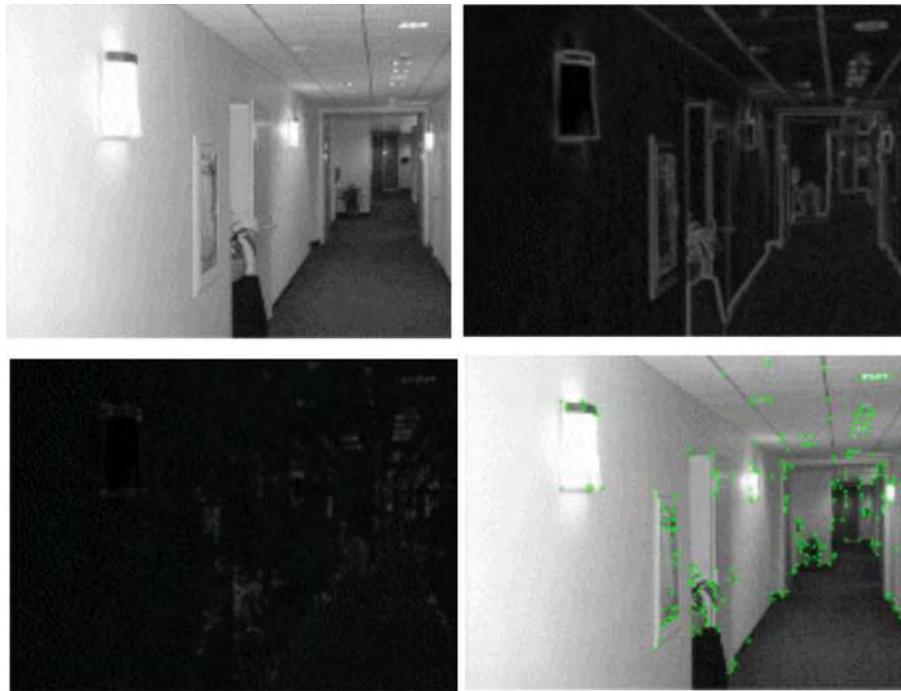


Figura 2.1: Resultado de calcular los valores propios de una imagen. Arriba derecha: valor propio mayor (aristas), debajo izquierda: valor propio menor (esquinas), debajo derecha: puntos esquina detectados aplicando una umbralización y una búsqueda de máximos locales.

Consideremos una imagen  $I$  y su gradiente  $(D_x, D_y)$ , el cálculo de las derivadas de la imagen es calculado como un par de convoluciones en  $1-D$  con filtros obtenidos por muestras de la función gaussiana, el cálculo de las derivadas se muestra en 3.5.2. Para un píxel genérico  $p$  y sus vecinos  $Q_p$ , definimos:

$$C(p) = \begin{bmatrix} \sum D_x^2 & \sum D_{xy} \\ \sum D_{xy} & \sum D_y^2 \end{bmatrix} \quad (2.1)$$

Al ser  $C(p)$  simétrica y semidefinida positiva es diagonalizable a una matriz de la forma:

$$C(p) = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \quad (2.2)$$

Cálculo del menor de los valores propios de la matriz  $C(p)$  diagonalizada mostrada en 2.2.

$$\begin{aligned} g_{xx} &= \sum D_x^2 \\ g_{xy} &= \sum D_{xy} \\ g_{yy} &= \sum D_y^2 \end{aligned} \quad (2.3)$$

$$\lambda_1^* = \frac{g_{xx} + g_{yy} - \sqrt{(g_{xx} - g_{yy})^2 + 4g_{xy}^2}}{2}$$

La Figura 2.2 muestra la clasificación del cambio que ocurre con los valores propios de la matriz  $C$ . Al analizar esta figura, se observa como en la parte superior derecha de la imagen se marca con un recuadro rojo y una cruz una área donde los gradientes de  $I_x, I_y$  son altos este es el típico problema de la apertura, en este caso se encuentra que en un borde un valor propio es grande  $\lambda_1 > 0$  y el otro es pequeño  $\lambda_2 = 0$  lo cual no es una buena característica para seleccionar un punto, en la parte inferior derecha de la imagen se muestra como en una esquina los dos valores propios tienen intensidades grandes  $\lambda_2 > 0, \lambda_1 > 0$ , estas intensidades son buenas características para seleccionar buenos puntos de control, para esta combinación seleccionamos el menor valor propio ya que este nos da las esquinas como lo muestra la figura 2.1 y por ultimo en la parte izquierda de la imagen se muestra como en una región constante los dos valores propios son pequeños  $\lambda_2 = 0, \lambda_1 = 0$

ya que sus gradientes tienen pequeñas magnitudes, como muestra Kanade, Lucas y Tomasi en [Lucas81], [Tomasi98]. Se puede considerar una esquina a aquella matriz de estructura local que tiene el menor valor propio suficientemente grande.



Figura 2.2: Clasificación del cambio de los valores propios de la matriz  $C$

El algoritmo 1 nos sirve para resumir el proceso de selección de puntos característicos en una imagen, para este proceso se utiliza la imagen de entrada, de la cual calculamos el gradiente y para cada punto de esta imagen calculamos la matriz  $C(p)$  ya definida y buscamos el menor de los valores propios ya que en este quedan las esquinas de la imagen como muestra Kanade, Lucas y Tomasi en [Lucas81], [Tomasi98], después se guardan las coordenadas del punto si cumplen con un umbral determinado, el umbral para el valor propio mas pequeño debe ser elegido con cuidado en función del histograma de los valores propios y guardamos las coordenadas de  $p$  en un conjunto o vector denominado  $Xs$ , ordenamos los elementos de  $Xs$  en forma decreciente y borramos los elementos que se repiten en  $Xs$  y ordenamos de nuevo, con esto garantizamos que nos quedamos con los mas altos valores propios (esquinas), los cuales nos dan las mejores características de la imagen, como lo muestran Kanade, Lucas y Tomasi en [Lucas81], [Tomasi98].

---

**Algoritmo 1** Algoritmo KLT

---

La entrada del algoritmo esta formada por la Imagen  $I$ , el umbral para el menor valor propio de la matriz  $C(p), \tau$  y una ventana cuadrada de lado  $2N + 1$ .

1. Calcular el gradiente de la imagen  $I$  ( $D_x, D_y$ ).
  2. Para cada punto  $p$ :
    - Calcular la matriz  $C(p)$  definida en (2.1) para la ventana de tamaño  $(2N + 1) \times (2N + 1)$ .
    - Calcular  $\lambda_i^*$ , el menor valor propio de  $C(p)$  con (2.3)
    - Si  $\lambda_i^* > \tau$  guardar las coordenadas de  $p$  en un conjunto que notaremos  $X_s$
  3. Ordenar los elementos de  $X_s$  en orden decreciente de  $\lambda_i^*$ .
  4. Para cada píxel en  $X_s$  borrar todos sus vecinos que aparezcan en  $X_s$ .
  5. Ordenar los elementos de  $X_s$  en orden decreciente de  $\lambda_i^*$ .
  6. Regresamos  $X_s$ .
-

## 2.2. El Problema de la Estimación y Emparejamiento de Puntos

Hipótesis básicas:

1. Muchos puntos de la escena son visibles en ambas imágenes.
2. Las regiones que se corresponden son similares

Estas hipótesis se verifican en los sistemas estéreo y entre parejas de imágenes en las cuales la distancia del punto fijo a las cámaras es mucho mayor que la distancia entre los centros de las cámaras. En general, sin embargo, ambas hipótesis pueden ser falsas y el problema de la correspondencia puede ser muy difícil. Por ahora consideraremos que se verifican siempre las dos hipótesis y miraremos al problema de la correspondencia como un problema de búsqueda: dado un elemento en la imagen izquierda, buscamos el elemento correspondiente en la imagen derecha. Esto conlleva dos decisiones.

- ¿Que elementos de las imágenes emparejar?
- ¿Que medida de similitud se debe adoptar?

Por conveniencia, clasificaremos los algoritmos de correspondencias en dos clases, basados en correlación y basados en características como muestra Zisserman en [Hartley04]. Aunque conceptualmente son muy similares, son en cambio muy diferentes desde el punto de vista de la implementación a la que conducen. Por ejemplo, los métodos de correlación se aplican a la totalidad de los puntos de la imagen, mientras que los métodos basados en características intenta establecer correspondencias entre conjuntos dispersos de características de la imagen.

### 2.2.1. Métodos Basados en Correlación

En los métodos basados en correlación, los elementos a emparejar son ventanas de la imagen de tamaño fijo, y el criterio de similitud es una medida de la correlación entre las ventanas en las dos imágenes. El elemento en correspondencia esta dado por la ventana que maximiza el criterio de similitud dentro de una región de búsqueda. En la primera figura

de las siguientes se muestra como se definen las ventanas de correlación y búsqueda, en la segunda imagen se muestra para una ventana dada las distintas posiciones de correlación que se calculan.

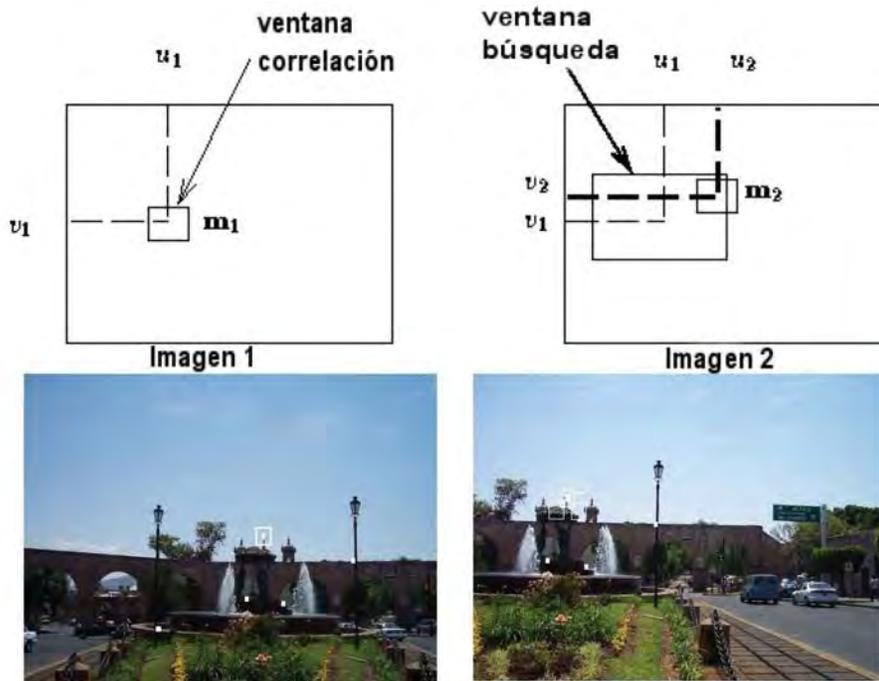


Figura 2.3: Ventana de correlación y búsqueda.

Los problemas de movimiento en las imágenes, habitualmente se afrontan de tal manera que se determinan los parámetros de desplazamiento sufridos en cada cuadro de la secuencia de imágenes. Esto se lleva a cabo haciendo uso de técnicas de correlación como el coeficiente de correlación lineal entre ambas imágenes. Esta medida, que puede absorber variaciones lineales en contraste entre ambas vistas, viene dada, para ventanas locales de dimensión  $N \times N$ . Algunas veces se puede utilizar Suma de Diferencias al Cuadrado (SDC), Suma de Diferencias Absolutas (SDA) como lo muestran Atienza en [Atienza04], una medida eficiente muchas veces utilizada es el coeficiente de correlación para los cuales se considera que estos puntos característicos no cambian de uno a otro cuadro, o bien contemplando modelos de movimiento mas complejos como el modelo afín o proyectivo.

Para el análisis de las escenas representadas en las imágenes, a menudo, queremos

investigar mas específicamente un área dentro de dicha imagen, llamada región de interés. El objetivo de una operación geométrica es la transformación de los valores de una imagen, tal y como podría observarse desde otro punto de vista. Así las operaciones de aumentar o disminuir una imagen no son otra cosa que acercar o alejar el punto de vista; rotarla equivale a girar el punto de observación y trasladarla es hacer lo mismo con el punto de observación.

Dada una secuencia de imágenes digitales  $I_k(x, y)$ , donde  $(x, y)$  son las coordenadas de la imagen y  $k$  el índice temporal de la imagen, la estimación de desplazamiento 2-D se puede definir como la estimación de los vectores de correspondencia  $d(x, y, k)$  del tiempo  $k$  al tiempo  $k - 1$  (estimación hacia atrás) o del tiempo  $k$  al tiempo  $k + 1$  (estimación hacia adelante). La Figura 2.4 ilustra las dos posibilidades como muestra Atienza en [Atienza04].

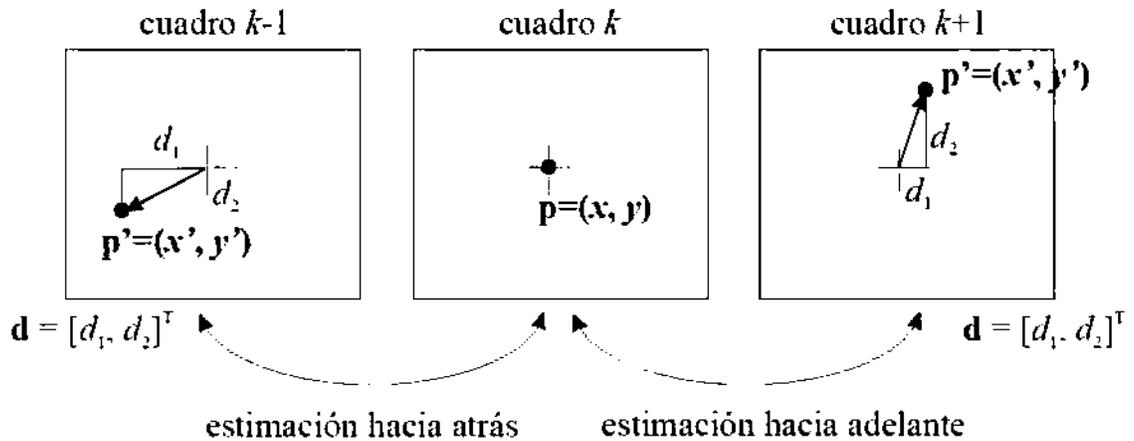


Figura 2.4: Estimación del desplazamiento hacia adelante y hacia atrás.

Los vectores de correspondencia varían normalmente de un píxel a otro (por ejemplo debido a rotación en la escena) y a lo largo del tiempo (debido a aceleraciones del objeto).

### 2.2.2. Estimación del Movimiento por Correlación

Este método considera que todos los puntos de la vecindad de los píxeles alrededor de un punto característico sufren el mismo desplazamiento, por lo que la apariencia de la región no cambia de una imagen a otra.

Un método frecuentemente utilizado para determinar que tan parecida es una vecindad  $\mathbf{W}$  de  $N1 \times N2$  píxeles, a otra desplazada  $d$  píxeles ( $d = [d_1, d_2]^T$ ), consiste en calcular

una medida de similitud como muestra Tekalp en [Tekalp95] basada en sumas de diferencias al cuadrado:

$$SDC(d_1, d_2) = \frac{1}{N_1 N_2} \sum_{\langle x, y \rangle \in W} [I_k(x, y) - I_{k-1}(x + d_1, y + d_2)]^2 \quad (2.4)$$

Este es el método empleado en propuestas de seguimiento como muestra Tomasi o Rehg en [Shi94], [Rehg91] respectivamente. La estimación de desplazamiento se realiza entonces tomando el valor de  $(d_1, d_2)$  que minimiza la medida SDC, es decir

$$[\widehat{d}_1, \widehat{d}_2]^T = \arg \min_{(d_1, d_2)} SDC(d_1, d_2) \quad (2.5)$$

Otra posibilidad es calcular una suma de diferencias absolutas como muestra Tekalp en [Tekalp95]:

$$SDA(d_1, d_2) = \frac{1}{N_1 N_2} \sum_{\langle x, y \rangle \in W} |I_k(x, y) - I_{k-1}(x + d_1, y + d_2)| \quad (2.6)$$

La medida SDA se calcula rápidamente al requerir únicamente sumas y restas, resulta sencilla de implementar en sistemas hardware de procesamiento de imágenes, por lo que es muy utilizada. Otra medida de uso frecuente es el coeficiente de correlación como muestra González en [?]. Ese índice es insensible a cambios de brillo de las imágenes y se define como:

$$\gamma(d_1, d_2) = \frac{\sum_{\langle x, y \rangle \in W} [I_k(x, y) - \bar{I}_k(x, y)][I_{k-1}(x - d_1, y - d_2) - \bar{I}_{k-1}(x, y)]}{\sqrt{\sum_{\langle x, y \rangle \in W} [I_k(x, y) - \bar{I}_k(x, y)]^2 \sum_{\langle x, y \rangle \in W} [I_{k-1}(x - d_1, y - d_2) - \bar{I}_{k-1}(x, y)]^2}} \quad (2.7)$$

donde  $\bar{I}_k(x, y)$  es el valor medio del nivel de gris en las coordenadas  $(x, y)$  en la región  $W$ .

### 2.3. Determinación de la Transformación entre Imágenes

Analizando los problemas anteriores, se tienen resueltos dos problemas planteados al inicio de nuestro trabajo para la construcción de mosaicos de imágenes, una vez determinados los parámetros de desplazamiento, se plantea a continuación la estimación de los

parámetros de movimiento conjunto, lo que habitualmente se conoce como pose del objeto. La siguiente etapa de nuestro proyecto se centra en el determinar los parámetros de el modelo de movimiento utilizado en este caso, se utiliza un modelo geométrico y el modelo de formación de la imagen (ortogonal, perspectiva, etc.) en este caso un modelo de formación de la imagen en perspectiva.

En este planteamiento se consideran restricciones en el tipo de movimiento del objeto, de la cámara o en las condiciones de formación de la imagen. De esta forma se hace posible el uso de modelos 2D que expliquen las transformaciones observadas en la imagen a consecuencia del movimiento. Estas restricciones pueden consistir en considerar que el objeto sólo puede rotar en un eje perpendicular al plano de la imagen o que se dan situaciones de perspectiva débil, en que las distancias entre los puntos del objeto en el espacio 3D son mucho menores que la distancia del objeto a la cámara, lo que minimiza los efectos de perspectiva.

En este caso el número de parámetros a determinar dependerá del modelo de movimiento que se considere (2 para un modelo traslacional, 4 para un modelo traslación-rotación, 6 en el caso de un modelo afín y 8 para un modelo perspectivo u homografías). El interés no radica ahora tanto en determinar el movimiento real 2D ó 3D sufrido por el objeto respecto a la cámara, sino en caracterizar las transformaciones observadas en la imagen a consecuencia del movimiento proyectado. Este planteamiento tiene especial justificación en aplicaciones de estabilización de imágenes, construcción de mosaicos panorámicos, compresión de vídeo y otras.

En general se trata de solucionar un problema de regresión, donde se pretende determinar el valor óptimo de una serie de parámetros de forma que se maximice algún índice de bondad en el ajuste. La estimación de la pose es a menudo un problema sobredeterminado, porque habitualmente hay más medidas disponibles que las estrictamente necesarias. Es muy frecuente la introducción de técnicas de estimación por mínimos cuadrados, así como el empleo de métodos de estadística robusta.

### **2.3.1. Métodos Robustos**

Para la estimación sobredeterminada de los parámetros buscados que se definirá en 2.3.2, se listan varios de los métodos entre los cuales están mínimos cuadrados y mínimos

cuadrados ponderados los cuales ofrecen buenos resultados cuando el ruido en las medidas es pequeño o se ajusta correctamente al modelo gaussiano esperado. Sin embargo muchos autores señalan que el modelo gaussiano para modelar los errores que se presentan típicamente en los resultados de los buscadores de puntos característicos resulta poco adecuado. La aparición de este tipo de errores en las medidas afecta sensiblemente a la estimación final por mínimos cuadrados, pudiendo suceder que una mala estimación en un cuadro provoque un error de seguimiento irrecuperable en el siguiente. Para evitar estos inconvenientes otros métodos alternativos usan conceptos de estadística robusta durante la estimación de los parámetros de movimiento, de forma que no se vean influenciados por la aparición de este tipo de datos atípicos.

Los datos atípicos se presentan cuando algunas correspondencia entre puntos, pueden ser mas precisas que otras, y es posible también tener errores en correspondencia, es por lo tanto aconsejable contar con un algoritmo que considere el manejo datos atípicos.

Una clase de métodos robustos frecuentemente utilizados para estimación en algoritmos de seguimiento son los denominados métodos de información mínima. Se trata de utilizar un subconjunto mínimo de las medidas disponibles para realizar la estimación de la transformación proyectiva para este trabajo de tesis como muestran Fischler y Bolles en [Fischler81]. Se basan en la hipótesis de que si se encuentra un subconjunto suficientemente preciso, el resto de medidas pueden ser ignoradas y de esta forma se evita que medidas erróneas puedan afectar negativamente a la estimación de la posición.

El método RANSAC (Random Sample Consensus) [Fischler81] es un estimador robusto de Fischler y Bolles. Considérese un conjunto de puntos correspondientes  $p_1, p_2$  cuya única fuente de error posible es la medida de sus posiciones en las imágenes, la cual sigue una distribución gaussiana. Con muchos casos esta suposición no es correcta por que en realidad hubo un error de correspondencia. A estos se les llama valores atípicos, como se muestra en la figura 2.5, y pueden afectar severamente al resultado de la estimación de los parámetros. A los valores que cumplen con la distribución gaussiana se les llama datos típicos.

En la figura 2.5 se muestra un ajuste de una recta por medio del método RANSAC, para este ejemplo se espera estimar una recta que deje fuera los datos atípicos, en este

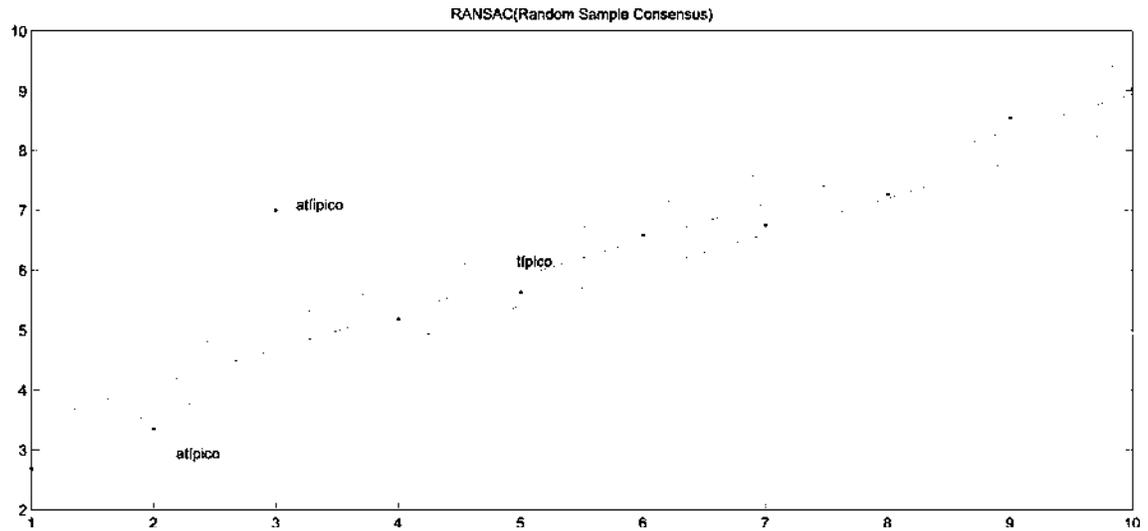


Figura 2.5: RANSAC: Datos atípicos y típicos

experimento contamos con 8 datos, de los cuales 2 de ellos son datos atípicos y el resto son datos típicos, los datos atípicos se encuentran marcados fuera de las líneas punteadas en la figura, al estimar la línea recta entre los datos, los datos atípicos no deben considerarse, RANSAC elimina estos datos atípicos para hacer una buena estimación de la recta final solo con los datos que mejor ajusten la recta.

A continuación se muestra el algoritmo para la estimación del ajuste de una recta por medio del método RANSAC, utilizado para la elaboración de las figuras 2.5, 2.6.

El objetivo entonces es determinar los valores uniformes a partir de un subconjunto del conjunto completo de correspondencia. Este subconjunto se escoge al azar del conjunto original. Para seleccionar el mejor subconjunto se utiliza un criterio de consenso: si la pose sugerida por un subconjunto también explica una parte significativa del resto de medidas, entonces es muy probable que la pose sea correcta, y puede ignorarse el resto de medidas.

En la figura 2.6 se puede observar como la línea azul no pasa por todos los datos mostrados en la figura, esta es una estimación de una recta realizada por RASANC, y la línea roja de la figura es la estimación final realizada con los datos que cumplen con la distribución gaussiana, (distribución gaussiana: Su propio nombre indica su extendida utilización, justificada por la frecuencia o normalidad con la que ciertos fenómenos tienden

---

**Algoritmo 2** Algoritmo RANSAC: para el ajuste de una recta

---

1. Seleccionar aleatoriamente una muestra  $s$  de puntos del conjunto total  $S$  y calcular el modelo con este subconjunto.
  2. Determinar el subconjunto de puntos  $S_i$  que están dentro de un umbral de distancia al modelo. El subconjunto  $S_i$  es el conjunto consenso y define los datos típicos de  $S$ .
  3. Si el conjunto  $S_i$  es mayor que un umbral  $T$ , volver a estimar el modelo utilizando todos los puntos de  $S_i$  y terminar.
  4. Si el tamaño de  $S_i$  es menor que  $T$ , seleccionar un nuevo subconjunto y repetir el paso anterior.
  5. Después de  $N$  intentos seleccionar el subconjunto de  $S_i$  con mayor consenso, volviendo a estimar el modelo utilizando todos los puntos del subconjunto  $S_i$ .
-

a parecerse en su comportamiento a esta distribución). Se puede observar que la línea roja pasa por la mayoría de los puntos mostrados en esta gráfica.

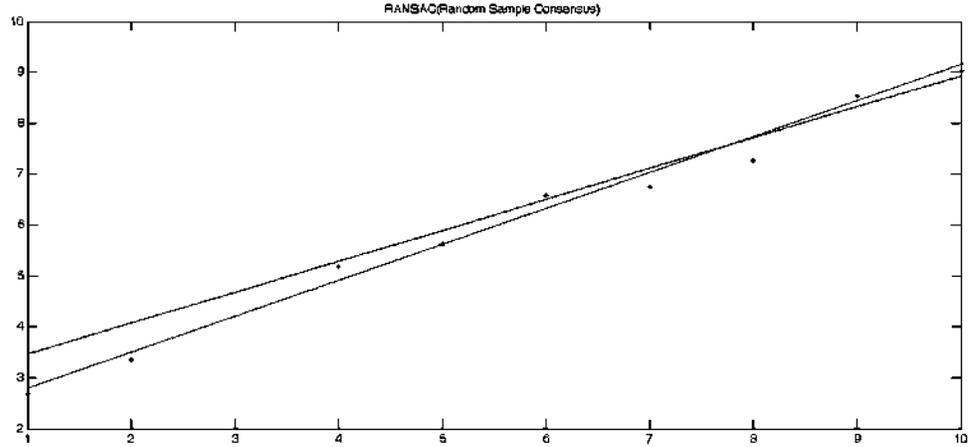


Figura 2.6: Datos típicos estimados por RANSAC

El algoritmo basado en RANSAC para el registro de imágenes calcula una Homografía dada por los últimos cuatro puntos correspondientes  $x_i \longleftrightarrow x_i^*$  los cuales son relacionados por  $x_i^* = Hx_i$  como muestra el sistema de la ecuación (2.8). Aquí  $x_i$  y  $x_i^*$  están dados en coordenadas homogéneas y  $H_i$  es una matriz de  $3 \times 3$ . Esta ecuación puede ser expresada en términos de producto cruz de vectores como  $x_i^* \times Hx_i = 0$ , de una manera corta  $A_i H = 0$ , con  $A_i$  matriz de  $2 \times 9$ . Si los datos no son exactos o tenemos más correspondencias de las que necesitamos, se busca una solución por mínimos cuadrados. Si se tienen 4 o más puntos (no colineales) correspondientes, se arman la solución  $AH = 0$ , con  $2n$  filas para  $n$  puntos correspondientes. Si se tienen exactamente 4 puntos, en este caso la solución de este sistema es directa:  $H = A^{-1}b$ . Obtenemos la DVS (Descomposición en Factores Singulares) de  $A$ . La solución para  $H$  es el vector singular que se corresponde con el valor singular de  $A$  más pequeño, esto es, la última columna de  $V$ , siendo  $A = UDV^T$  y  $D$  la matriz diagonal de valores positivos descendientes, para mayores detalles ver [Hartley04].

Los pasos para Estimar una Homografía Automáticamente para registro de imágenes utilizando el método RANSAC eliminando datos atípicos, dadas dos imágenes son:

---

**Algoritmo 3** Algoritmo RANSAC para estimar la Homografía para el registro de imágenes

---

1. Puntos de Control: Calcular los puntos de control (determinados por el usuario) en cada imagen usando el algoritmo KLT mostrado por Lucas en [Lucas81].
  2. Correspondencias Putativas: Calcular el conjunto de puntos de control que empatan las intensidades del vecindario (usando correlación) con la ecuación (2.7).
  3. RANSAC Estimador robusto: Repetir para M ejemplos
    - Seleccionar aleatoriamente un conjunto de cuatro puntos correspondientes y calcular la homografía  $H^{(k)}$ , resolviendo la ecuación (2.8).
    - Dado  $H^{(k)}$ , calcular la distancia para cada correspondencia putativa.
    - Calcular el número de datos típicos consistentes con  $H^{(k)}$ .
    - Elegir la  $H^{(k)}$  con el mayor número de datos típicos.
  4. Optimizar Estimación: Reestimamos  $H^{(*)}$  para todos los puntos correspondientes con el conjunto de consenso.
-

### 2.3.2. Métodos por Mínimos Cuadrados

El término mínimos cuadrados describe el problema muy frecuente de resolver sistemas de ecuaciones lineales sobre-determinados, esto es, sistemas lineales con más ecuaciones que incógnitas. En tal caso, en lugar de resolver las ecuaciones de manera exacta, habitualmente no existe tal solución, se busca sólo minimizar la suma de los cuadrados de los residuos.

Un gran número de trabajos sobre imágenes basado en puntos característicos utilizan todas las medidas disponibles mediante algún tipo de criterio de ajuste por mínimos cuadrados como muestran Gennery, Lowe, Odone y Colombo en [Gennery92], [Lowe92], [Odone04], [Colombo99] respectivamente. En términos de álgebra lineal se trata de resolver sistemas lineales  $Ax = b$  con más ecuaciones que incógnitas, donde el término resolver se entiende en el sentido de minimizar la norma euclídea de vector residuo  $r = Ax - b$ . Un ejemplo sencillo trataría de determinar los parámetros de una transformación afín que explicaran de forma óptima (en el sentido de mínimos cuadrados) el desplazamiento sufrido por un conjunto de puntos característicos [ $\langle x_1, y_1 \rangle, \langle x_2, y_2 \rangle, \dots, \langle x_N, y_N \rangle$ ] en un cuadro  $k$  a las posiciones [ $\langle i_1, j_1 \rangle, \langle i_2, j_2 \rangle, \dots, \langle i_N, j_N \rangle$ ] en un cuadro  $k + 1$ . Dada la ecuación (3.2) para un solo par de puntos  $\langle x_1, y_1 \rangle, \langle i_1, j_1 \rangle$ , entonces con  $\theta_8 = 1$  las dos últimas ecuaciones pueden ser escritas de manera matricial como:

$$\begin{bmatrix} i & j & 1 & 0 & 0 & 0 & -xi & -xj \\ 0 & 0 & 0 & i & j & 1 & -yi & -yj \end{bmatrix} \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \\ \theta_7 \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} \quad (2.8)$$

o bien  $Ah = b$ . Se observa que para cada correspondencia de puntos se obtienen dos ecuaciones. Suponiendo  $n$  pares de puntos correspondientes se puede establecer el siguiente sistema de  $2n$  ecuaciones y 8 incógnitas.

$$\begin{bmatrix} A_1 \\ A_1 \\ \vdots \\ A_n \end{bmatrix} H = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \quad (2.9)$$

donde  $A_i$  es la matriz  $A$  y  $b_i$  el vector  $b$  obtenidas en (2.8) para el punto  $i$ . El sistema de ecuaciones anteriores puede ser expresado como  $Ah = b$ . Para  $n = 4$  existe una solución directa dada por  $h = A^{-1}$ . Si  $n < 4$  sin embargo el sistema queda sobredeterminado, el sistema se dice sobredeterminado por que  $m > n$ , hay mas ecuaciones que incógnitas. En este caso se utiliza el método de los mínimos cuadrados en el que se encuentra un vector  $h$  tal que minimice  $\|\tilde{A}h - b\|$ . Para 5 o más puntos se debe utilizar un método de mínimos cuadrados cuya solución estaría dada por  $h = [\tilde{A}^T \tilde{A}]^{-1} \tilde{A}^T b$  para mayores detalles consultar Zisserman en [Hartley04].

## Capítulo 3

# Registro de Imágenes Basado en Tono de Gris

### 3.1. Introducción

Registro Paramétrico en el cual por medio de unos pocos parámetros se logra hacer una deformación de toda la imagen. El modelo Afín o el modelo de transformación Rígida de una imagen serán ejemplos típicos de estos modelos. Por otro lado tenemos un modelo no-Paramétrico en el cual tenemos un campo de deformaciones que aplicamos en forma independiente a cada punto de la imagen. En ambos casos tenemos que resolver problemas no lineales que tienen mínimos locales y obviamente el utilizar el segundo modelo eleva la complejidad numérica del problema

Antes de abordar este capítulo, se describirá la transformación en perspectiva utilizada para este trabajo, así como las técnicas de registro que nos permitirán encontrar el conjunto de parámetros de la transformación, también se analizarán las funciones de error analizadas para medir las diferencias entre dos imágenes. Mas adelante se utiliza el método de Gauss-Newton-Levenberg-Marquardt para resolver el problema de la minimización de la función de error.

### 3.2. Conceptos Geométricos Básicos

El paso previo a toda operación geométrica comienza por observar la distribución espacial de los píxeles en la imagen origen y en la transformada. Puesto que la imagen digital es discreta, no existen valores de intensidad entre los valores discretos de las coordenadas, que coinciden con las intersecciones de los valores discretos tanto horizontales como verticales. Al transformar esta rejilla según un desplazamiento, un giro, un acercamiento o cualquier otra transformación geométrica, los nuevos píxeles ya no tienen por que quedar situados sobre tales intersecciones y caerán en general, sobre puntos intermedios de ellos. Al tener que proyectar estos píxeles sobre los de la imagen final, que deben asumir una estructura similar a la de la imagen original, es necesario pasar de las coordenadas pseudodiscretas a las discretas definitivas y, en concreto, es preciso calcular los valores de los píxeles finales en función de los transformados [Pajares03].

Para ilustrar lo anterior mostramos en la figura 3.1 una transformación geométrica, que en este caso es una rotación. La rejilla con líneas continuas muestra la disposición de los píxeles de la imagen original, mientras que la rejilla discontinua ilustra como queda la rejilla original después de girarla un ángulo dado alrededor de un punto determinado, es decir, después de rotarla.

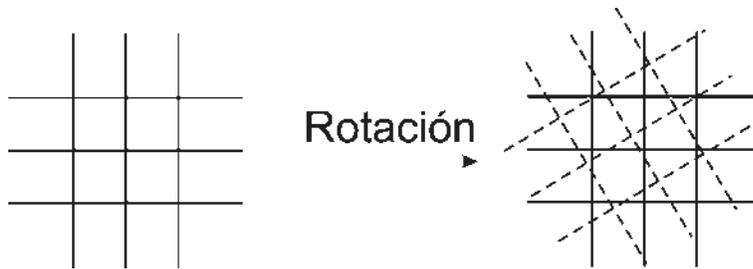


Figura 3.1: Ante cualquier transformación geométrica, los píxeles de la rejilla transformada (líneas discontinuas) no tienen por que coincidir con los de la rejilla destino

Para llegar al resultado final se enumeran los dos pasos a seguir:

1. Determinar las coordenadas de cada píxel  $(i, j)$  mediante la transformación proyectiva perteneciente a la rejilla original en la rejilla transformada (en líneas discontinuas). En general, estos nuevos píxeles  $(i', j')$  obtenidos tras la transformación no serán valores

enteros y, por lo tanto, no coinciden con píxeles de la rejilla destino.

2. Calcular los valores de los píxeles  $(x, y)$  finales en la rejilla destino a partir de los valores conocidos de los píxeles  $(i', j')$ .

El primer paso depende de la transformación a realizar, es decir, de la función de transformación que se aplique en cada momento, mientras que el segundo se corresponde con una operación de interpolación. Tras la transformación geométrica, es preciso obtener los valores de intensidad asociados con dicha transformación de forma que la imagen original aparezca transformada geométricamente pero con los valores de intensidad obtenidos de la imagen original [Pajares03].

### 3.2.1. Interpolación Bilineal

Una interpolación bilineal, aplicada a una imagen, consiste en asignarle un valor de tono de gris a un píxel en base al tono de gris de sus cuatro vecinos. Para rellenar los huecos de las imágenes corregidas, en este trabajo utilizamos un método de interpolación bilineal en donde, en lugar de utilizar la información de los vecinos en la imagen corregida, utilizamos información de los vecinos en la imagen original. El problema de querer utilizar la interpolación bilineal en base a la imagen original, consistirá en mapear la imagen expandida a la imagen distorsionada. Como en general los píxeles de la imagen corregida no tienen una correspondencia con los píxeles de la imagen original distorsionada, será necesario ubicarlos a nivel subpíxel en la imagen distorsionada y a partir de esta utilizar a sus cuatro vecinos para obtener el tono de gris que asignaremos al píxel en la imagen corregida.

Sea  $x_e$  y  $x_f$  la parte entera y la parte fraccional respectivamente de  $x$ . Sea  $y_e$  y  $y_f$  la parte entera y la parte fraccional de  $y$ . En la figura 3.2 se ilustra el método de la interpolación bilineal para encontrar el valor de intensidad del píxel  $I(x_e + x_f, y_e + y_f)$ , basándose en la intensidad de los 4 píxeles adyacentes [William92].

En la figura 3.2 primero se interpola  $I(x_e, y_e + y_f)$  en base a  $I(x_e, y_e)$  y  $I(x_e, y_e + 1)$ , asumiendo una relación lineal entre ellos, simbolizada por la línea recta que conecta las partes superiores de las barras verticales (representando el nivel de gris) de los píxeles involucrados. Posteriormente se interpolan  $I(x_e + 1, y_e + y_f)$  en base a los valores de intensidad de los

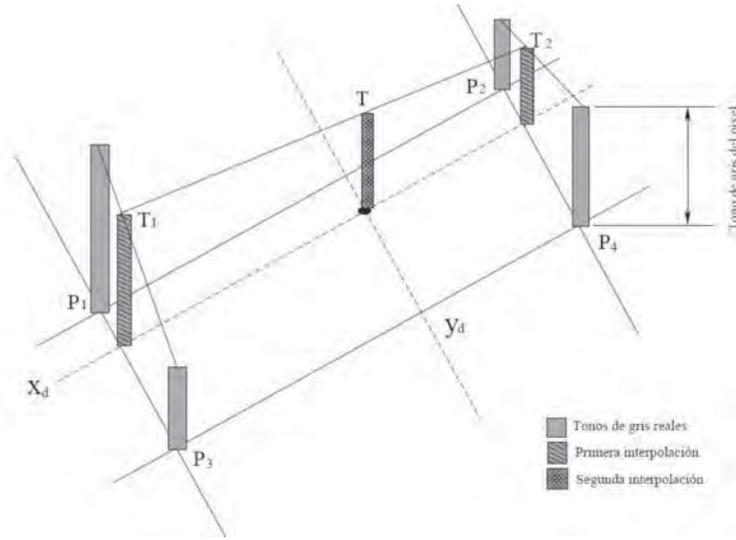


Figura 3.2: Interpolación bilineal, la interpolación se hace primero en el sentido de  $x$  obteniendo  $T_1$  y  $T_2$  y después se hace en el sentido de  $y$  para obtener el color final  $T$ .

píxeles  $I(x_e+1, y_e)$  y  $I(x_e+1, y_e+1)$ . Por último, de los dos valores calculados previamente se obtiene finalmente el valor deseado, en este caso  $I(x_e+x_f, y_e+y_f)$ . La notación matemática de la interpolación bilineal quedará entonces de la siguiente manera:

$$\begin{aligned}
 I(x_e, y_e + y_f) &= (1 - y_f)I(x_e, y_e) + y_f I(x_e, y_e + 1) \\
 I(x_e + 1, y_e + y_f) &= (1 - y_f)I(x_e + 1, y_e) + y_f I(x_e + 1, y_e + 1) \\
 I(x_e + x_f, y_e + y_f) &= (1 - x_f)I(x_e, y_e + y_f) + x_f I(x_e + 1, y_e + y_f)
 \end{aligned}
 \tag{3.1}$$

### 3.2.2. Coordenadas Homogéneas (CH)

Antes de continuar, introducimos el concepto de coordenadas homogéneas que permiten la realización de transformaciones de imágenes mediante el uso de las matrices correspondientes. Para representar un punto en coordenadas homogéneas, se considera el punto como un vector de la forma  $[F_e x, F_e y, F_e]^T$ , donde  $F_e$  es un factor de escala. Por ejemplo, el punto  $[12, 6]^T$  en un sistema de coordenadas cartesiano, podría representarse como  $[24, 12, 2]^T$ , o bien, como  $[6, 3, 0.5]^T$ . En ambos casos, los dos vectores representan el mismo punto, si bien en el primer caso el valor de  $F_e$  es 2 y en el segundo caso es 0.5.

Generalmente y por conveniencia, las componentes de un vector se normalizan

dividiéndolas por el factor de escala  $F_e$ ; esto hace que el nuevo factor de escala sea 1 y las demás componentes sean las mismas que las originales, esto facilita el manejo de las coordenadas como muestra Zisserman en [Hartley04]. Así, la representación normalizada en coordenadas homogéneas del punto  $[12, 6]^T$  resulta ser  $[12, 6, 1]^T$  como muestra Pájaras en [Pajares03].

### 3.2.3. Transformación Projectiva

Al establecer una relación entre un plano del mundo real y una imagen, también será posible establecer una relación entre dos imágenes diferentes del mismo plano real, esto se conoce como homografía, o transformación proyectiva, esta es la más general de las transformaciones que mapea líneas a líneas, y esta generaliza la transformación afín. La matriz  $H_p$  para esta transformación tiene 9 elementos (solo 8 parámetros se usan, por las CH proporcionan vectores que representan el mismo vector, no se conserva el paralelismo). Una transformación proyectiva puede ser calculada a partir de la correspondencia entre cuatro puntos como se muestra en la sección 2.3.2. La transformación perspectiva explica el movimiento observado en la imagen cuando se tiene una proyección perspectiva de la escena en el plano de imagen.

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} = \begin{bmatrix} \theta_0 & \theta_1 & \theta_2 \\ \theta_3 & \theta_4 & \theta_5 \\ \theta_6 & \theta_7 & \theta_8 \end{bmatrix} \begin{bmatrix} i \\ j \\ 1 \end{bmatrix} \quad (3.2)$$

Una transformación proyectiva transforma un plano  $R_1$  en otro plano  $R_2$  equivalente y en el cual se conservan algunas de las características del primero, aunque otras se pierden. Una de las características que se conservan es la de la colinealidad, es decir, las líneas que son rectas en  $P$  seguirán siendo rectas en  $P'$ .

En la figura 3.3 se muestran dos planos  $R_2$  y  $R_1$  que comparten el mismo eje de proyección  $Z$  y en donde idealmente los dos planos son perpendiculares a dicho eje. De la misma forma se muestra como  $R_1$  es proyectado en  $R_2$ , utilizando como referencia un punto llamado punto de proyección. Este tipo de proyección es conocido como proyección en perspectiva, en el cual, las líneas paralelas en una escena que no son paralelas al plano de

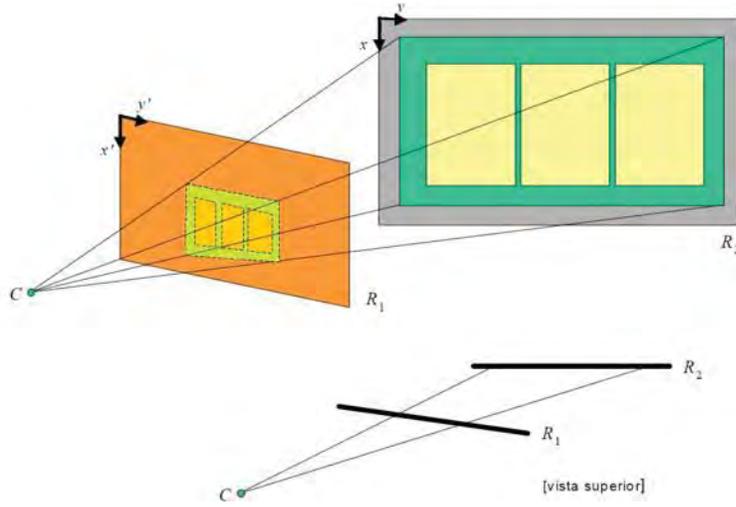


Figura 3.3: Transformación proyectiva

la imagen, se proyectan en líneas convergentes. Este tipo de proyección hace parecer a las imágenes más reales ya que esta es la manera en que nuestros ojos y la lente de una cámara forman las imágenes como muestra Donald en [Donald95].

Considerando que el objetivo principal es emparejar dos imágenes, en donde una de ellas es una proyección de la otra, será necesario encontrar un conjunto de parámetros que nos permita encontrar la serie de transformaciones espaciales.

Para el caso de la transformación proyectiva se utilizarán 9 parámetros. La ecuación de la transformación proyectiva expresada en coordenadas homogéneas se presenta en la ecuación (3.3).

En donde, por facilidad matemática, dicha matriz es normalizada dividiendo todo entre  $\theta_8$ , por lo que  $\theta_8 = 1$ . Además, la presencia de  $\theta_6$  y  $\theta_7$  nos obligan a dividir todo el sistema entre un mismo factor  $w(\theta, i, j) = \theta_6x + \theta_7y + 1$  para que  $w$  siga siendo la unidad y  $(x, y)$  correspondan a las coordenadas cartesianas.

$$\begin{aligned}
 x(\theta, i, j) &= \frac{\theta_0i + \theta_1j + \theta_2}{w(\theta, i, j)} \\
 y(\theta, i, j) &= \frac{\theta_3i + \theta_4j + \theta_5}{w(\theta, i, j)} \\
 w(\theta, i, j) &= \theta_6x + \theta_7y + 1
 \end{aligned} \tag{3.3}$$

### 3.3. El Problema de Registro Paramétrico

En el registro paramétrico tenemos dos imágenes  $I_1$  e  $I_2$  de las cuales  $I(i, j)$  denota un valor de gris (típicamente un valor entre 0 y 255), para coordenadas enteras  $\langle i, j \rangle$ ,  $I(i, j)$  nos brinda el valor de intensidad del píxel asociado en la posición  $(i, j)$  y  $I_r(i, j)$  será nuestra imagen referencia.

Al buscar un conjunto de parámetros el cual representaremos como  $\Theta$ , el problema de registro paramétrico es buscar un conjunto  $\Theta$  que minimice la suma de las diferencias al cuadrado o alguna de las funciones de error  $E$  analizadas en secciones anteriores, donde  $E$  puede ser expresado de la siguiente manera.

$$\begin{aligned} E(I_1(\Theta), I_r) &= \sum_{\forall \langle i, j \rangle \in I_r} \rho(e_{ij})^2 \\ e_{ij} &= I_1(x(\theta, i, j), y(\theta, i, j)) - I_r(i, j) \end{aligned} \quad (3.4)$$

Por ejemplo, dada una posición  $x = i + 1$  y  $y = j$ , un píxel  $I_r(i, j)$  este será comparado con el píxel  $I_1(i + 1, j)$ . Esta situación es equivalente a tener una imagen transformada,  $I_t$ , donde todos los píxeles de la imagen de entrada, se han movido a la siguiente posición hacia arriba. El error  $E$ , entre  $I_t$  y  $I_r$ , compara cada píxel de ambas imágenes en la misma posición  $(i, j)$ . Con la correcta  $\Theta$ , la imagen  $I_t$  e  $I_r$  deben ser muy similares y  $E$  debería tener un valor mínimo, donde  $I_t$  es calculada como:

$$I_t(i, j) = I(x(\theta, i, j), y(\theta, i, j)) \quad (3.5)$$

Para el calculo de esta imagen se utiliza el método de interpolación bilineal analizado en el capítulo anterior.

### 3.4. Definición de las Funciones de error

Muchos algoritmos de registro tratan de minimizar la suma de las diferencias al cuadrado (SDC) entre la imagen origen y la imagen referencia, algunas otras se basan estimadores robustos como el Estimador Lorenziano (LE) o el método RANSAC analizado en el capítulo anterior, algunos autores utilizan el Estimador Lorenziano en imágenes con ruido como muestra Calderón en [Calderón06], obteniendo buenos resultados.

Otro método robusto es llamado, Suma de las diferencias al cuadrado con condición de reposo adaptable (SSD-ARC), este es un algoritmo robusto al ruido, que resuelve grandes desplazamientos, utiliza información de un par de imágenes y esta basado en la minimización de una función de energía. Calderón muestra en [Calderón03] algunas aplicaciones para el registro de imágenes con SSD-ARC y aplicaciones para la calibración de cámaras se muestran en [Calderón04].

Para el registro nosotros contamos con dos imágenes,  $I_1$  e  $I_r$ , a las cuales llamaremos imagen origen e imagen referencia respectivamente, estas imágenes representan la misma escena pero con algunas diferencias geométricas que pudieran ser originadas, por ejemplo, por una transformación proyectiva (rotación, traslación, escalado, etc).

Podemos plantear al empate de las dos imágenes en base a una función de error, usando  $\rho$ -función, definida por Hampel en [Hampel86], se puede definir la función de error por mínimos cuadrados como  $\rho_{LS}(e_i) = e_i^2$  la cual se encargará de medir las diferencias, píxel a píxel, entre ambas imágenes.

$$E_{LS}(\theta) = \sum_{i=1}^N \rho_{LS}(e_i(\theta)) \quad (3.6)$$

donde  $N$  es el número de píxeles, y  $\rho$  es la función de error dada por:

$$\rho_{LS}(e_i) = e_i^2 \quad (3.7)$$

### 3.4.1. Mínimos Cuadrados

Asumiendo que  $e_i(\theta)$  tiene una distribución con media cero y varianza constante, un método para calcular  $\Theta^*$  es mínimos cuadrados (LS) [McLachlan88]. En este caso, la probabilidad de  $p(\theta)$  está dada por la ecuación (3.8) y  $\Theta^*$  es llamado estimador de máxima verosimilitud.

$$p(\Theta) = \prod_{i=1}^N \frac{1}{2\pi} \exp^{-[e_i(\theta)]^2} \quad (3.8)$$

El objetivo de la ecuación (3.8) es buscar un conjunto  $\Theta$  que maximice la probabilidad  $p(\theta)$ . Sacando el logaritmo de la ecuación (3.8), uno puede ver que esto es equivalente a

minimizar la ecuación (3.4) para el mismo conjunto de datos. La función de influencia esta definida como la derivada de la función de error y esto nos ayuda a ver la contribución de el error para la correcta solución. En el caso de LS, la función esta dada por la ecuación (3.9).

$$\Psi_{LS}(e_i) = \frac{\partial \rho_{LS}(e_i)}{\partial(e_i)} = \frac{\partial e_i^2}{\partial e_i} = 2e_i \quad (3.9)$$

Esto significa que errores grandes tienen una gran contribución a el vector gradiente global. Entonces cuando usamos un algoritmo basado en la búsqueda del gradiente, este factor contribuye a alcanzar una solución parcial con datos atípicos. El problema con LS es muy sensible a datos atípicos, por que estos contribuyen mucho a el error general  $E_{LS}$ , como describe Hampel en [Hampel86].

### 3.4.2. Estimador Lorenziano (LE)

Hampel muestra en [Hampel86] la estadística robusta la cual presenta las siguientes ventajas: proporciona un mejor conjunto de parámetros describiendo mejor los datos de entrada e identificando los datos atípicos. como muestra M. J, Black en [Black96]. Black usa una distribución T-student con dos grados de libertad, en lugar de la distribución normal para el estimador de máxima probabilidad, por que la distribución T-student es usada en poblaciones con una varianza desconocida. La distribución T-student es definida por la ecuación (3.10).

$$h(t) = \frac{\Gamma[(v+1)/2]}{\Gamma(v/2)\sqrt{\pi v}} \left(1 + \frac{t^2}{v}\right)^{-(v+1)/2} \quad (3.10)$$

donde  $\Gamma$  es la función gama y  $v$  son los grados de libertad, la distribución T-student con dos grados de libertad y el logaritmo de esta, es dada por las ecuaciones (reg:tstudentb) y (reg:tstudentlog).

$$h(t) = \frac{1}{2\sqrt{2}} \left(1 + \frac{t^2}{2}\right)^{-1.5} \quad (3.11)$$

$$\log(h(t)) = \log\left(\frac{1}{2\sqrt{2}}\right) - 1.5 \log\left(1 + \frac{t^2}{2}\right) \quad (3.12)$$

Black en [Black96] sustituye  $t = e_i/\sigma$  en el segundo termino de la ecuación (3.12). Eliminando el factor de asociación, el termino resultante mostrado en la ecuación (3.13), es conocido como Estimador Lorenziano (LE), el cual tiene una función de influencia dada por la ecuación (3.14).

$$\rho_{LE}(e) = \log \left( 1 + \frac{e^2}{2\sigma^2} \right) \quad (3.13)$$

$$\psi_{LE}(e) = \frac{2e}{2\sigma^2 + e^2} \quad (3.14)$$

La principal ventaja de usar LE en lugar de LS es la función de influencia, en la cual el factor  $1/(2\sigma^2 + e^2)$  reduce la contribución de los datos atípicos, produciendo en el caso de errores grandes, derivadas cercanas a cero. El parámetro  $\sigma$  controla la frontera entre los datos típicos y los datos atípicos para LE.

### 3.4.3. Suma de Diferencias al Cuadrado con Condición de Reposo Adaptable

La Suma de las Diferencias al Cuadrado con Condición de reposo adaptable (SSD-ARC) para el registro no paramétrico de imágenes fue presentado, por Calderón en [Calderón03], como la minimización de una función cuadrática de energía  $\widehat{E}_{SSD-ARC}$  con el termino  $l_i h_i$  se reduce enormemente la contribución del error dado por (3.15).

$$\widehat{E}_{SSD-ARC}(\Theta, l) = \sum_{i=1}^N (e_i(\Theta) - l_i h_i)^2 + \mu \sum_{i=1}^N l_i^2 \quad (3.15)$$

Donde  $h_i$  es un error deseable dependiendo del peso del error de la función,  $l_i \in [0, 1]$  es un indicador de la función, bajo el control de parámetro  $\mu$ ,  $\Theta$  es el vector de desplazamiento por cada píxel de la imagen. Al asumir el modelo proyectivo, con  $h_i = e_i(\theta)$ , y anulando el ultimo parámetro de (3.15), finalmente una función paramétrica para SSD-ARC es obtenida como:

$$\widehat{E}_{SSD-ARC}(\theta, l) = \sum_{i=1}^N e_i^2(\theta)(1 + l_i)^2 + \mu \sum_{i=1}^N l_i^2 \quad (3.16)$$

Para la ecuación (3.16), el termino  $(1 - l_i)^2$  nos permite descartar datos atípicos; uno esperaría valores entre cero y uno para la variable  $l_i$ , cero para datos típicos, y uno para datos atípicos. El segundo termino en (3.16) restringe el numero de datos atípicos por medio de  $\mu$ , para  $\mu = 0$  significa que todos los datos de la imagen son datos atípicos en contraste con  $\mu = \infty$ , significaría que los píxeles de la imagen son buenos o son datos típicos y la solución es cercana a LS. La minimización de  $l_i^*$  para (3.16) se puede calcular resolviendo  $\frac{\partial \widehat{E}_{SSD-ARC}(\theta, l)}{\partial l_i} = 0$ , y esta dada por la ecuación (3.17).

$$l_i^* = \frac{e_i^2(\theta)}{\mu + e_i^2(\theta)} \quad (3.17)$$

La variable  $l_i$  es limitada pos los intervalos  $[0, 1]$  una condición indispensable para estimar los datos atípicos. Reemplazando el valor de  $l_i^*$  en 3.16, tenemos una nueva función paramétrica SSD-ARC que denominaremos  $E_{SSD-ARC}(\theta)$

$$E_{SSD-ARC}(\theta) = \sum_{i=1}^N \frac{\mu e_i^2(\theta)}{\mu + e_i^2(\theta)} \quad (3.18)$$

La  $\rho$ -función es en este caso:

$$\rho_{SSD-ARC}(e_i) = \frac{\mu e_i^2}{\mu + e_i^2} \quad (3.19)$$

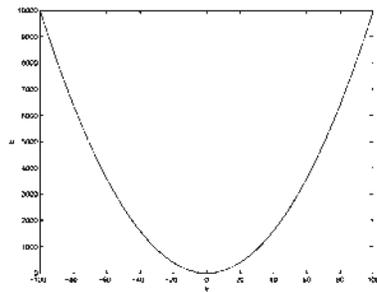
la cual es unimodal respecto a  $e$ . La función de influencia (primera derivada) y segunda derivada están dadas por  $\psi_{SSD-ARC}(e_i)$  y  $\varphi_{SSD-ARC}(e_i)$  respectivamente.

$$\psi_{SSD-ARC}(e_i) = \frac{2\mu^2 e_i}{(\mu + e_i^2)^2} \quad (3.20)$$

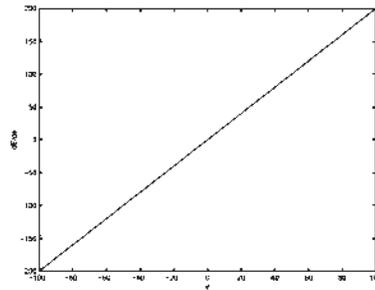
$$\varphi_{SSD-ARC}(e_i) = \frac{2\mu^2(\mu - 3e_i^2)}{(\mu + e_i^2)^3} \quad (3.21)$$

Las funciones de error para  $\rho_{LS}$ ,  $\rho_{LE}$  y  $\rho_{SSD-ARC}$ , son unimodales con respecto a el termino de error, como se muestra en las figuras 3.4(a), 3.4(c) y 3.4(e). LS no tiene la posibilidad de rechazar datos atípicos por que su derivada es proporcional al termino de error (ver figura 3.4(b)). Note que la función de influencia de SSD-ARC, exhibe un comportamiento similar e el Estimador Lorenziano. En ambas funciones, un error grande

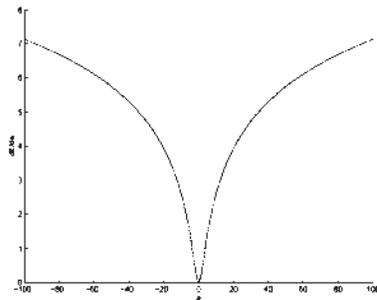
proporciona valores en sus derivadas cercanos a cero, como se puede observar en las figuras 3.4(d) y 3.4(f). Esto significa que no hay problema con errores grandes proporcionados por datos atípicos.



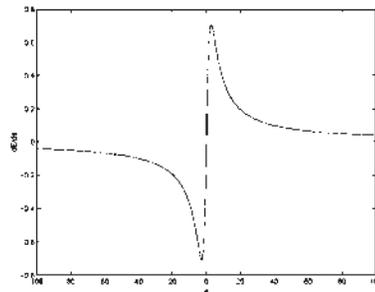
(a) Función de error LS



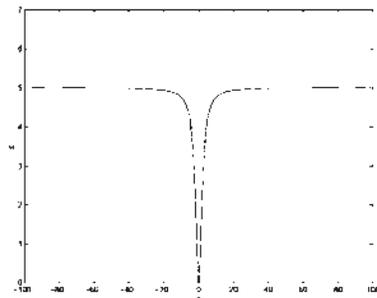
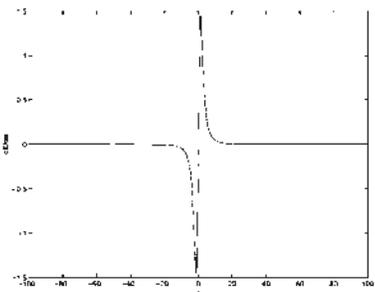
(b) Función de Influencia LS



(c) LE: Función de error con sigma = 2



(d) LE: Función de influencia

(e) SSD-ARC: Función de error con  $\mu=10$ 

(f) Función de Influencia para SSD-ARC

Figura 3.4: Ejemplo de las funciones de error

### 3.5. Buscando el Conjunto de Parámetros para el Registro

Haciendo referencia a la ecuación (3.4) en la cual, la función  $\rho(e_{ij})$  puede ser cambiada por una función cuadrática o algún M-estimador (ver [Huber03]). En el caso de la función cuadrática la solución es conocida como Suma de las diferencias al cuadrado. Dado un conjunto  $\Theta = \{\theta_0, \dots, \theta_k, \dots, \theta_K\}$  de  $K$  parámetros. Como ya se menciono en la sección del problema de registro, se busca un conjunto de parámetros  $\Theta$ , donde  $E$  será mínimo cuando su derivada sea igual a 0,  $\partial E / \partial \theta_k$  para toda  $k$ . El vector gradiente  $G_k$  puede ser calculado como en la ecuación (3.22).

$$\begin{aligned} G_k(\Theta) &= \frac{\partial E}{\partial \theta_k} = \sum_{\forall \langle i,j \rangle \in Ir} \frac{\partial \rho(e_{ij})}{\partial e_{ij}} \frac{\partial e_{ij}}{\partial \theta_k} \\ G_k(\Theta) &= \sum_{\forall \langle i,j \rangle \in Ir} \varphi_{ij}(\Theta) J_{ij}(\Theta) \\ k &= 0, 1, \dots, K \end{aligned} \quad (3.22)$$

En la literatura, la derivada  $\frac{\partial \rho(e_{ij})}{\partial e_{ij}}$  es llamada función de influencia como muestra Huber en [Huber03] y es representada por  $\varphi(e_{ij})$ . En el caso de una función de error cuadrático  $\varphi(e_{ij}) = 2e_{ij}$ . Si ponemos el vector gradiente como  $G(\Theta) = [G_0(\Theta), G_1(\Theta), \dots, G_k(\Theta)]^T$  y el vector Jacobiano como  $J_{ij}(\Theta)$  como la ecuación (3.25).

$$J_{ij}(\Theta) = \left[ \frac{\partial e_{ij}}{\partial \theta_0} \quad \frac{\partial e_{ij}}{\partial \theta_1} \quad , \dots, \quad \frac{\partial e_{ij}}{\partial \theta_K} \right]^T \quad (3.23)$$

El elemento  $k - th$  del gradiente en la ecuación (reg:gradiente) puede ser escrito como:

$$G(\Theta) = \sum_{\forall \langle i,j \rangle \in Ir} \varphi_{ij}(\Theta) J_{ij}(\Theta) \quad (3.24)$$

Vamos a desarrollar una expresión para cada termino del vector  $J_{ij}(\Theta) = [J_{ij0}(\Theta), G_{ij1}(\Theta), \dots, G$  para la ecuación (3.4) de modo que la expresión para  $J_{ijk}(\Theta)$  puede ser derivada como la ecuación 3.25 así que tomando como referencia la ecuación (3.4) y la ecuación (3.22) y simplificando la notación, podremos expresar el vector gradiente como el producto de la Matriz Jacobiana  $J(\Theta)$  y la derivada de la función de error  $\varphi_{ij}(\Theta)$ . En esta sección detallamos el desarrollo del Jacobiano.

$$J_{ij}(\Theta) = \frac{\partial e_{ij}}{\partial \theta_k} = \frac{\partial I_1(x(\theta, i, j), y(\theta, i, j))}{\partial \theta_k} \quad (3.25)$$

Usando la regla de la cadena del calculo diferencial, una columna del Jacobiano puede ser calculada como:

$$J_{ij}(\Theta) = \frac{\partial I_1(x(\theta, i, j), y(\theta, i, j))}{\partial x(\theta, i, j)} \frac{\partial x(\theta, i, j)}{\partial \theta_k} + \frac{\partial I_1(x(\theta, i, j), y(\theta, i, j))}{\partial y(\theta, i, j)} \frac{\partial y(\theta, i, j)}{\partial \theta_k} \quad (3.26)$$

Usando la notación matricial, la columna  $ij$  -  $th$  del Jacobiano esta dada por  $J_{ij}(\Theta) = M(\Theta, i, j) \nabla I_1(x, y)$  con

$$M(\Theta, i, j) = \begin{bmatrix} \frac{\partial x(\theta, i, j)}{\partial \theta_0} & \frac{\partial y(\theta, i, j)}{\partial \theta_0} \\ \vdots & \vdots \\ \frac{\partial x(\theta, i, j)}{\partial \theta_k} & \frac{\partial y(\theta, i, j)}{\partial \theta_k} \end{bmatrix} \quad (3.27)$$

$$\nabla I_1(x, y) = \begin{bmatrix} \frac{\partial I_1(x(\theta, i, j), y(\theta, i, j))}{\partial x(\theta, i, j)} \\ \frac{\partial I_1(x(\theta, i, j), y(\theta, i, j))}{\partial y(\theta, i, j)} \end{bmatrix}$$

Donde  $M$  es el Modelo de la Matriz de Coordenada (MMC) el cual depende de las características del modelo y  $\nabla I_1(x, y)$  es el vector gradiente de la imagen respecto a las coordenadas  $x$  y  $y$  en la subsección 3.28 se presenta el modelo de la matriz de coordenadas considerado para la transformación proyectiva y en la subsección 3.5.2 se presenta el método para el calculo del vector de derivadas  $\nabla I_1(x, y)$ .

### 3.5.1. Modelo de la Matriz de Coordenadas

Derivando la ecuación (3.26) para cada elemento de la ecuación (3.3) respecto de  $\langle x, y \rangle$  obtenemos los elementos de la matriz de coordenadas mostrada en la ecuación (3.27) colocando cada elemento en la matriz formulamos el modelo de la matriz de coordenadas. Considerando nuestro modelo de transformación proyectiva dado por la ecuación (3.3) y la definición para MMC de la ecuación (3.27), el modelo de coordenadas quedara como en (3.28).

$$M(\Theta, i, j) = \begin{bmatrix} \frac{\partial x}{\partial \theta_0} & \frac{\partial y}{\partial \theta_0} \\ \frac{\partial x}{\partial \theta_1} & \frac{\partial y}{\partial \theta_1} \\ \frac{\partial x}{\partial \theta_2} & \frac{\partial y}{\partial \theta_2} \\ \frac{\partial x}{\partial \theta_3} & \frac{\partial y}{\partial \theta_3} \\ \frac{\partial x}{\partial \theta_4} & \frac{\partial y}{\partial \theta_4} \\ \frac{\partial x}{\partial \theta_5} & \frac{\partial y}{\partial \theta_5} \\ \frac{\partial x}{\partial \theta_6} & \frac{\partial y}{\partial \theta_6} \\ \frac{\partial x}{\partial \theta_7} & \frac{\partial y}{\partial \theta_7} \end{bmatrix} = \frac{1}{w} \begin{bmatrix} i & 0 \\ j & 0 \\ 1 & 0 \\ 0 & i \\ 0 & j \\ 0 & 1 \\ -ix & -iy \\ jx & -jy \end{bmatrix} \quad (3.28)$$

### 3.5.2. Cálculo de la Derivada de la Imagen

Para el registro de imágenes utilizado en este trabajo es necesario calcular las derivadas de la imagen las cuales se pueden aproximar por una diferencia central como muestra Trucco en [Trucco98], por la ecuación (3.29), o un método más preciso es el que considera más píxeles en el vecindario también presentado en [Trucco98] con la ecuación 3.30;

$$\begin{aligned} \frac{\partial I(i,j)}{\partial i} &= \frac{I(i+1,j) - I(i-1,j)}{2} \\ \frac{\partial I(i,j)}{\partial j} &= \frac{I(i,j+1) - I(i,j-1)}{2} \end{aligned} \quad (3.29)$$

$$\begin{aligned} \frac{\partial I(i,j)}{\partial i} &= \frac{-I(i+2,j) + 8I(i+1,j) - 8I(i-1,j) + I(i-2,j)}{12} \\ \frac{\partial I(i,j)}{\partial j} &= \frac{-I(i,j+2) + 8I(i,j+1) - 8I(i,j-1) + I(i,j-2)}{12} \end{aligned} \quad (3.30)$$

Un método mejor para el cálculo de estas derivadas es el llamado derivadas de gaussianas como muestra Soatto y Romeny en [Soatto95],[Romeny94] este método responde mucho mejor que los anteriores en imágenes con ruido, la función Gaussiana se presenta continuación así como las derivadas de esta función.

$$\begin{aligned} g(t) &= \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{t^2}{2\sigma^2}} \\ g'(t) &= \frac{dg(t)}{dt} = \frac{-t}{\sigma^2\sqrt{2\pi}\sigma} e^{-\frac{t^2}{2\sigma^2}} \end{aligned} \quad (3.31)$$

El cálculo de las derivadas de la imagen es realizado como un par de convoluciones en  $1 - D$  con filtros obtenidos por muestras de la función gaussiana y las derivadas de esta,

$$\begin{aligned}\frac{\partial I(i,j)}{\partial i} &= I(i,j) * g'(i) * g(j) = \sum_{k=-w/2}^{w/2} \sum_{l=-w/2}^{w/2} [I(i,j)g'(i-k)g(j-l)] \\ \frac{\partial I(i,j)}{\partial j} &= I(i,j) * g(i) * g'(j) = \sum_{k=-w/2}^{w/2} \sum_{l=-w/2}^{w/2} [I(i,j)g(i-k)g'(j-l)]\end{aligned}\quad (3.32)$$

### 3.5.3. Derivada de la Imagen Transformada

Considerando que  $\langle i, j \rangle \in N^2$  y  $\langle i, j \rangle \in R^2$ , si  $\partial I(i, j)/\partial i$  y  $\partial I(i, j)/\partial j$  con calculadas usando uno de los métodos de la subsección 3.5.2, un método simple y rápido es el calcular  $\partial I(x, y)/\partial x$  usando interpolación bilineal para obtener un valor aproximado de  $\partial I(i, j)/\partial i$ . De forma similar, podemos estimar  $\partial I(i, j)/\partial j$  con la aproximación de  $\partial I(x, y)/\partial y$ .

Considerando la imagen transformada  $I_t$  y la ecuación (3.5), otra aproximación es dada por la ecuación (3.33).

$$\begin{aligned}\frac{\partial I_1(x(\Theta, i, j), y(\Theta, i, j))}{\partial x(\Theta, i, j)} &= \frac{\partial I_t(i, j)}{\partial i} \\ \frac{\partial I_1(x(\Theta, i, j), y(\Theta, i, j))}{\partial y(\Theta, i, j)} &= \frac{\partial I_t(i, j)}{\partial j}\end{aligned}\quad (3.33)$$

Considerando que  $I_t(i, j) = I_1(x(\Theta, i, j), y(\Theta, i, j))$ , y solo considerando las derivadas dadas por la ecuación (3.33). Se concluye que la primera consideración es correcta, pero no la segunda, por que los incrementos en  $x$  no necesariamente corresponden a los mismos incrementos en  $i$  (manejando el mismo argumento para  $y$  y  $j$ ). Se tiene que las derivadas correctas del método se pueden calcular aplicando la regla de la cadena, como sigue:

$$\begin{aligned}\frac{\partial I_1(x, y)}{\partial x} &= \frac{\partial I_t(i, j)}{\partial i} \frac{\partial i}{\partial x} + \frac{\partial I_t(i, j)}{\partial j} \frac{\partial j}{\partial x} \\ \frac{\partial I_1(x, y)}{\partial y} &= \frac{\partial I_t(i, j)}{\partial i} \frac{\partial i}{\partial y} + \frac{\partial I_t(i, j)}{\partial j} \frac{\partial j}{\partial y}\end{aligned}\quad (3.34)$$

Usando la notación matricial la ecuación 3.34 puede ser escrita como:

$$\begin{aligned}\begin{bmatrix} \frac{\partial I_1(x, y)}{\partial x} \\ \frac{\partial I_1(x, y)}{\partial y} \end{bmatrix} &= \begin{bmatrix} \frac{\partial i}{\partial x} & \frac{\partial j}{\partial x} \\ \frac{\partial i}{\partial y} & \frac{\partial j}{\partial y} \end{bmatrix} \begin{bmatrix} \frac{\partial I_t(i, j)}{\partial i} \\ \frac{\partial I_t(i, j)}{\partial j} \end{bmatrix} \\ \nabla I_1(x, y) &= N(\Theta, x, y) \nabla I(i, j)\end{aligned}\quad (3.35)$$

Donde definiremos  $N(\Theta, x, y)$  como la Matriz de Corrección de Derivadas (MCD),  $\nabla I_t(i, j)$  es el vector gradiente de la imagen transformada  $I_t$  y puede ser calculado usando, las derivadas del filtro gaussiano, mencionada en la subsección 3.5.2 y  $\nabla I_1(i, j)$  es el vector gradiente de la imagen en coordenadas enteras. Para calcular la MCD, en el caso de la transformación proyectiva, el modelo dado por 3.3 lo podemos reescribir como un sistema de ecuaciones (3.38) y el procedimiento para llegar a este sistema es el siguiente, primero despejamos el términos  $w$  de 3.3 quedando la ecuación 3.36.

$$(\theta_6 i + \theta_7 j + 1)x = \theta_0 i + \theta_1 j + \theta_2 \quad (3.36)$$

A continuación desarrollamos el producto de la izquierda de la ecuación 3.36 y por ultimo agrupamos los términos en  $i$  y  $j$  quedando la ecuación 3.37, se aplica el mismo procedimiento para la  $y$ .

$$(x\theta_6 - \theta_0)i + (x\theta_7 - \theta_1)j = \theta_2 - x \quad (3.37)$$

La solución a este sistema de ecuaciones, respecto a  $i$  y  $j$ , esta dado por la ecuación (3.40), por definición de la ecuación (3.34), la MCD esta dada por la ecuación 3.41.

$$\begin{aligned} (x\theta_6 - \theta_0)i + (x\theta_7 - \theta_1)j &= (\theta_2 - x) \\ (y\theta_6 - \theta_3)i + (y\theta_7 - \theta_4)j &= (\theta_5 - y) \end{aligned} \quad (3.38)$$

El sistema de ecuaciones dado por 3.38, puede ser resuelto usando la regla de Cramer's del álgebra lineal, desarrollando la regla de Cramer's tenemos:

$$\begin{aligned} i &= \frac{\begin{vmatrix} \theta_2 - x & x\theta_7 - \theta_1 \\ \theta_5 - y & y\theta_7 - \theta_4 \end{vmatrix}}{\begin{vmatrix} x\theta_6 - \theta_0 & x\theta_7 - \theta_1 \\ y\theta_6 - \theta_3 & y\theta_7 - \theta_4 \end{vmatrix}} = \frac{(\theta_2 - x)(y\theta_7 - \theta_4) - (\theta_5 - y)(x\theta_7 - \theta_1)}{(x\theta_6 - \theta_0)(y\theta_7 - \theta_4) - (y\theta_6 - \theta_3)(x\theta_7 - \theta_1)} \\ j &= \frac{\begin{vmatrix} x\theta_6 - \theta_0 & \theta_2 - x \\ y\theta_6 - \theta_3 & \theta_5 - y \end{vmatrix}}{\begin{vmatrix} x\theta_6 - \theta_0 & x\theta_7 - \theta_1 \\ y\theta_6 - \theta_3 & y\theta_7 - \theta_4 \end{vmatrix}} = \frac{(x\theta_6 - \theta_0)(\theta_5 - y) - (y\theta_6 - \theta_3)(\theta_2 - x)}{(x\theta_6 - \theta_0)(y\theta_7 - \theta_4) - (y\theta_6 - \theta_3)(x\theta_7 - \theta_1)} \end{aligned} \quad (3.39)$$

Desarrollando algebraicamente la ecuación 3.39 tenemos que la solución para  $i$  y  $j$  estará dada por:

$$\begin{aligned} i &= \frac{(\theta_4 - \theta_5 \theta_7)x + (\theta_2 \theta_7 - \theta_1)y + (\theta_1 \theta_5 - \theta_2 \theta_4)}{(\theta_3 \theta_7 - \theta_4 \theta_6)x + (\theta_1 \theta_6 - \theta_0 \theta_7)y + (\theta_0 \theta_4 - \theta_1 \theta_3)} \\ j &= \frac{(\theta_5 \theta_6 - \theta_3)x + (\theta_0 - \theta_2 \theta_6)y + (\theta_2 \theta_3 - \theta_0 \theta_5)}{(\theta_3 \theta_7 - \theta_4 \theta_6)x + (\theta_1 \theta_6 - \theta_0 \theta_7)y + (\theta_0 \theta_4 - \theta_1 \theta_3)} \end{aligned} \quad (3.40)$$

Para obtener los elementos de la matriz  $N$  derivamos la ecuación 3.40 para cada elemento de la matriz en 3.35 y después de algunos paso sencillos de álgebra obtenemos la matriz MCD como:

$$\begin{aligned} N(\Theta, x, y) &= F \begin{bmatrix} -(\theta_7 y - \theta_4) & (\theta_6 y - \theta_3) \\ (\theta_7 x - \theta_1) & -(\theta_6 x - \theta_0) \end{bmatrix} \\ F &= \frac{\theta_0(\theta_4 - \theta_5 \theta_7) - \theta_1(\theta_3 - \theta_5 \theta_6) + \theta_2(\theta_3 \theta_7 - \theta_4 \theta_6)}{[(\theta_3 \theta_7 - \theta_4 \theta_6)x + (\theta_1 \theta_6 - \theta_0 \theta_7)y + (\theta_0 \theta_4 - \theta_1 \theta_3)]^2} \end{aligned} \quad (3.41)$$

Una forma de comprobar nuestro procedimiento es por ejemplo, cuando una transformación proyectiva esta dada solo por una transformación afín, entonces el vector de parámetros estará dado por  $\Theta = [\theta_0, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5, 0, 0]$  y la MCD estará definida como:

$$N(\Theta, x, y) = \frac{1}{\theta_0 \theta_4 - \theta_1 \theta_3} \begin{bmatrix} \theta_4 & -\theta_3 \\ -\theta_1 & \theta_0 \end{bmatrix} \quad (3.42)$$

En el caso de que la transformación solo sea una traslación, el vector de parámetros es escrito como  $\Theta = [1, 0, 0, 0, 1, 0, 0, 0]$  y para este caso la MCD es

$$N(\Theta, x, y) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (3.43)$$

### 3.6. Método de Optimización para la función de error

En esta sección estableceremos primeramente el método que optimizara nuestra primera aproximación de la matriz  $H$ , se describirá el Método de Gauss-Newton-levenberg-Marquardt para resolver el problema de minimizar la función de error. Minimizar la función de error nos permitirá aproximarnos a solucionar el problema de registro de imágenes.

### 3.6.1. Procedimiento de Minimización

Ahora bien, dado un conjunto  $\Theta$ ,  $E$  será mínimo cuando su derivada sea igual a 0. El procedimiento de minimización de la función con el que se trabajo en esta tesis es simplemente decir que tenemos  $G_k(\Theta^n) = \partial E / \partial \Theta_k$ , ( $k = 0, \dots, k$ ), con  $\Theta^n$  que es el conjunto inicial de valores para los parámetros,  $\Theta^n + 1$  un conjunto de parámetros mejorado, donde  $\Theta_k^{n+1} = \Theta_k^n + \Delta \Theta_k^n$ . Ahora el problema se convierte en buscar un conjunto de incrementos  $\Delta \Theta_k^n$  tal que  $G_k(\Theta^n) = 0$ . El calculo de los incrementos puede ser aproximado usando la serie de Taylor truncando a su primer derivada:

$$G_k(\Theta^{n+1}) = G_k(\Theta^n) + \frac{\partial G_k(\Theta^n)}{\partial \theta_0} \Delta \theta_0^n + \frac{\partial G_k(\Theta^n)}{\partial \theta_1} \Delta \theta_1^n + \dots + \frac{\partial G_k(\Theta^n)}{\partial \theta_k} \Delta \theta_k^n \quad (3.44)$$

Igualando a cero el termino  $G_k(\Theta^{n+1}) = 0$ , y resolviendo el sistema tenemos:

$$\frac{\partial G_k(\Theta^n)}{\partial \theta_0} \Delta \theta_0^n + \frac{\partial G_k(\Theta^n)}{\partial \theta_1} \Delta \theta_1^n + \dots + \frac{\partial G_k(\Theta^n)}{\partial \theta_k} \Delta \theta_k^n = -G_k(\Theta^n) \quad (3.45)$$

Esta estrategia para minimizar la función objetivo en este caso, la dirección de búsqueda se determina utilizando la segunda derivada de la función objetivo. El método aproxima la función objetivo  $E(\Theta)$  en la vecindad de un mínimo con una serie de Taylor truncada hasta el termino de segundo orden.

El mínimo de la función  $G_k(\Theta^{n+1})$  se determina haciendo  $G_k(\Theta^{n+1}) = 0$  y calculando el valor de  $x_i$  que satisface la ecuación. Se obtiene entonces el sistema:

$$H \Delta \Theta = G \quad (3.46)$$

Donde despejando tenemos:

$$H = \begin{bmatrix} \frac{\partial G_0(\Theta^n)}{\partial \theta_0} & \frac{\partial G_0(\Theta^n)}{\partial \theta_1} & \dots & \frac{\partial G_0(\Theta^n)}{\partial \theta_k} \\ \frac{\partial G_1(\Theta^n)}{\partial \theta_0} & \frac{\partial G_1(\Theta^n)}{\partial \theta_1} & \dots & \frac{\partial G_1(\Theta^n)}{\partial \theta_k} \\ \frac{\partial G_k(\Theta^n)}{\partial \theta_0} & \frac{\partial G_k(\Theta^n)}{\partial \theta_1} & \dots & \frac{\partial G_k(\Theta^n)}{\partial \theta_k} \end{bmatrix} \quad (3.47)$$

$$\Delta \Theta = [\Delta \theta_0^n, \Delta \theta_1^n, \dots, \Delta \theta_k^n]^T$$

$$G = [-G_0(\Theta^n), -G_1(\Theta^n), \dots, -G_k(\Theta^n)]^T$$

Una vez que este sistema es resuelto, un nuevo conjunto de parámetros  $\Theta^{n+1}$  puede ser calculados y usando el mismo procedimiento otro conjunto de parámetros  $\Theta^{n+2}$  es estimado. El proceso iterativo termina cuando todos los incrementos son muy pequeños.

Si todas las derivadas de  $\partial G_r(\Theta^n)/\partial \theta_c$  son calculadas para (3.22), con este calculo se tiene el método de Newton como muestra Nocedal en [Nocedal99]. Pero si descartamos la segunda derivada, el método se llama Gauss-Newton este es comúnmente usado debido a que es simple y por que garantiza tener una matriz Hessiana definida semipositiva:

$$\begin{aligned} H(\Theta^n) &= \frac{\partial^2 E(\Theta^n)}{\partial \theta_r \partial \theta_c} = \sum_{\forall \langle i,j \rangle \in I_r} \psi(e_{ij}) \frac{\partial e_{ij}}{\partial \theta_r} \frac{\partial e_{ij}}{\partial \theta_c} \\ H(\Theta^n) &= \sum_{\forall \langle i,j \rangle \in I_r} \psi(e_{ij}) [J_{ij}(\Theta^n)]^T [J_{ij}(\Theta^n)] \end{aligned} \quad (3.48)$$

donde  $\psi(e_{ij}) = \frac{\partial^2 \rho(e_{ij})}{\partial^2 e_{ij}}$ . Este método mostrado en la sección no siempre encuentran el valor mínimo de la función ya que se pueden quedar en mínimos locales, por que solo usamos la primer derivada en la serie de Taylor. Un método mas robusto para el calculo es el denominado Levenberg-Marquardt que se presenta en la siguiente sección.

### 3.7. Método de Levenberg Marquardt

Esta es una técnica iterativa para problemas no lineales especialmente designado para minimizar funciones las cuales tiene las forma de suma de diferencias como muestra Zisserman en [Hartley04], esta técnica a cada iteración calcula  $E(\Theta)$ , el incremento de los parámetros  $\Delta(\Theta)$ , y se resuelve el siguiente sistema lineal de ecuaciones:

$$(H + \Lambda)\Delta\Theta = G \quad (3.49)$$

Donde la matriz  $H$  y  $G$  son definidas por las ecuaciones (3.47),  $\Lambda$  es una matriz diagonal  $\Lambda = \text{diag}(\lambda, \lambda, \dots, \lambda)$ , y  $\lambda$  es un parámetro el cual cambia a cada iteración.

El proceso inicia con la imagen de entrada,  $I$ , y la imagen referencia,  $I_r$ , y valores iniciales para los parámetros de  $(\Theta^0)$ .

Note que cuando  $\lambda = 0$ , el método es igual al método de Newton, y cuando  $\lambda$  tiende a infinito,  $\Delta\Theta$  es como el denominado descenso de gradiente y el incremento en  $\Delta\Theta$  tiende a cero.

---

**Algoritmo 4** Algoritmo Levenberg Marquardt
 

---

1. Iniciamos con un valor pequeño para  $\lambda$  (digamos  $\lambda = 0.001$ ), un valor inicial para  $\Theta^0$ , la función de error ( $\rho_{SSD-ARC}(e_i) = \frac{\mu e_i^2}{\mu + e_i^2}$ ) e iniciamos  $n = 0$
  2. Para un  $\Theta^n$ , calculamos la imagen transformada  $I_t^n$  (3.5), aplicando interpolación bilineal usando la ecuación (3.1) para mejorar la calidad de la imagen.
  3. Calculamos el error total,  $E(\Theta)$  usando la ecuación (3.4).
  4. Calculamos un nuevo conjunto de parámetros usando los siguientes pasos.
    - Para un modelo dado, calculamos  $M(\Theta, i, j)$  y  $N(\Theta, x, y)$  usando las ecuaciones (3.28) y (3.41) respectivamente.
    - Calculamos el vector Gradiente de la imagen  $\nabla I(i, j)$  aplicando un filtro gaussiano de derivadas (3.32)
    - Calcula la matriz Jacobiano como  $J_{ij}(\Theta^n) = M(\Theta, i, j)N(\Theta, x, y)\nabla I_r(i, j)$
    - Calculamos el vector Gradiente  $G(\Theta^n)$  y la matriz Hessiana  $H(\Theta^n)$  por las ecuaciones (3.22) y (3.48) respectivamente.
    - Resolvemos el sistema lineal de ecuaciones dado por (3.49) para  $\Delta\Theta^n$ , y entonces calculamos  $E(\Theta^n + \Delta\Theta^n)$
  5. Si  $E(\Theta^n + \Delta\Theta^n) > E(\Theta^n)$ , incrementamos  $\lambda$  por un factor de 10, y regresamos al paso previo. Si  $\lambda$  crece muy grande, esto significa que no hay forma de mejorar la solución  $\Theta^n$  y el algoritmo termina con esta solución como la óptima  $\Theta^* = \Theta^n$ .
  6. Si  $E(\Theta^n + \Delta\Theta^n) \leq E(\Theta^n)$ , decrementar  $\lambda$  por un factor de 10. Actualizamos  $\Theta^{n+1} = \Theta^n + \Delta\Theta^n$ , e incrementamos  $n = n + 1$  y regresamos al paso 2.
-

### 3.8. Resultados Obtenidos con Registro de Imágenes

En el resto del capítulo, por convención se usará SSD-ARC seguido del método de optimización, por ejemplo si se usa el GNLM (Newton-Levenberg-Marquardt) solo se pondrá GNLM; si se utiliza alguna función de error se antepondrá al método de optimización SSD-ARC-GNLM.

En esta sección, nosotros compararemos los resultados del registro de imágenes usando las funciones de error vistas en la sección 3.4 y el método RANSAC.

En los experimentos con imágenes sintéticas nosotros conoceremos los parámetros correctos de  $\hat{\Theta}$  para la imagen transformada tomada como referencia. Conociendo estos parámetros se propone usar la Distancia Euclídea  $|\Delta\Theta|$  entre el vector de parámetros  $\hat{\Theta}$  conocido y el vector de parámetros  $\Theta$  estimado como una medida de proximidad. Los algoritmos con mejor comportamiento tendrán un gran parecido al vector de parámetros real.

Para las imágenes sintéticas de Lena, se usara como aproximación inicial el vector  $\Theta^{(0)} = [1, 0, 0, 0, 1, 0, 0, 0]^T$ , y un criterio de parada  $\frac{E^{(k+1)} - E^{(k)}}{E^{(k+1)}} < 1e - 5$  o 1000 iteraciones.

Los valores de  $\mu$  para SSD-ARC y  $\sigma$  para LE se buscaron para cada función asegurando su mejor resultado y estos valores están entre 20 y 25 respectivamente.

Para las imágenes sintéticas de la Tierra, se usara como aproximación inicial el vector estimado por RANSAC y mínimos cuadrados  $\Theta = [0.8, -0.3, 20.0, 0.3, 0.8, -20.0, 0.0, 0.0]^T$ , con 500 iteraciones para el método de registro.

#### 3.8.1. Imagen Lena

El principio global de este experimento es dar un ejemplo de como la función de error SSD-ARC se desempeña, el primer experimento fue hecho usando GNLM para función de error LS, LE, y SSD-ARC. En este caso el experimento tiene una rotación de  $18^\circ$ , dado por  $\hat{\Theta} = [0.9511, -0.3090, 0.0000, 0.3090, 0.9511, 0.0000, 0.0000, 0.0000]$  la cual es aplicada a la imagen Lena.

Los resultados para las diferentes funciones de error LS, LE, SSD-ARC minimizadas con el método de optimización GNLM, son presentados en la tabla 3.1; la misma tabla presenta el resultado del método RANSAC usando las mismas imágenes aplicando la  $\Theta^{(0)}$



Figura 3.5: Experimento con la imagen lena usando SSD-ARC-GNLM

Tabla 3.1: Resultados para imagen Lena, usando LS, LE, SSD-ARC y RANSAC

algoritmos	parámetros	$\Theta$	$ \Delta\Theta $
LS-GNLM		0.95109 -0.30899 0.0000 0.30899 0.95109 0.0000 0.0000 0.0000	0.0000
SSD-ARC-GNLM	$\mu = 20$	0.95109 -0.30899 0.0000 0.30899 0.95109 0.0000 0.0000 0.0000	0.00137
SSD-ARC-GNLM	$\mu = 5000$	0.95109 -0.30899 0.0000 0.30899 0.95109 0.0000 0.0000 0.0000	0.0000
LE-GNLM	$\sigma = 25$	1.124 -0.057 -11.015 0.067 0.969 -4.005 0.0000 0.0000	11.72742
RANSAC		0.9449,-0.3078,0.7086 0.3038,0.9473,1.2068 0.0000,0.0000	1.3994

establecida arriba. De acuerdo con los resultados presentado en la tabla 3.1, LS tiene mejor desempeño que RANSAC, esto se debe a que las imágenes no presentan datos atípicos, LE presenta los peores resultados.

En la tabla 3.1 se demuestra como **SSD-ARC-GNLM** presenta los mejores resultados también presenta diferentes resultados usando diferentes valores de  $\mu$ , con  $\mu = 5000$ , **SSD-ARC-GNLM** presenta mejores resultados que con  $\mu = 20$ . Esto muestra que para este experimento, SSD-ARC-GNLM con valores altos de  $\mu$  presenta similar comportamiento que LS-GNLM. Esta condición puede ser usada en este caso por que las imágenes no presentan datos atípicos. Nosotros sugerimos usar valores pequeños para  $\mu$  para permitir el uso de datos atípicos para la función SSD-ARC.

### 3.8.2. Imagen Lena con Datos Atípicos

En este experimento, en contraste con el anterior, pondremos en negro (valor cero) el 10% de los píxeles de la imagen, para simular datos atípicos. Usando la misma transformación del experimento previo, la imagen referencia se muestra en la figura 3.6. De manera similar al experimento anterior, la tarea de registro fue realizada usando LS, LE, SSD-ARC y RANSAC en combinación con el algoritmo de minimización de la función de error GNLM. La función SSD-ARC presenta un comportamiento similar a LS; Note que el vector de parámetros estimado por SSD-ARC-GNLM es el mas cercano a el vector real por lo que se concluye que este es el mejor método al estimar la transformación final.

### 3.8.3. Imagen Sintética de la Tierra

Al igual que en la sección anterior este experimento es elaborado para el análisis de las funciones de error LS, LE, SSD-ARC en conjunto del método de optimización GNLM, el primer experimento fue hecho usando GNLM para función de error LS, LE, y SSD-ARC. En este caso el experimento tiene el siguiente vector de parámetros  $\hat{\Theta} = [0.8, -0.3, 20.0, 0.3, 0.8, -20.0, 0.0, 0.0]$  la cual es aplicada a la imagen origen, en este caso utilizaremos la estimación hecha por RANSAC y mínimos cuadrados como primera aproximación de la homografía, el vector de parámetros de esta primera aproximación es  $\Theta^{(0)} = [0.8057, -0.3069, 20.2699, 0.3079, 0.7963, -20.2699, 0.0, 0.0, 1.0]$ , el cual se aplica a la



Figura 3.6: Experimento con la imagen lena usando SSD-ARC-GNLM con datos atípicos

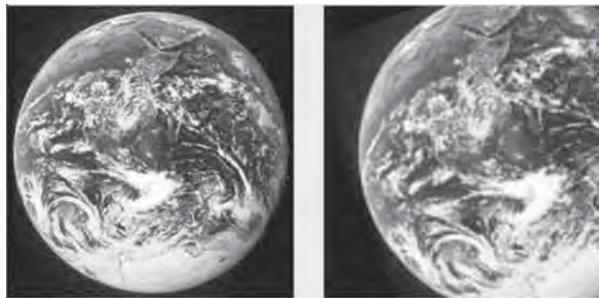


Figura 3.7: Experimento con la imagen tierra usando SSD-ARC-GNLM

Tabla 3.2: Resultados para imagen Lena, usando LS, LE, SSD-ARC y RANSAC con datos atípicos en la imagen

algoritmos	parámetros	$\Theta$	$ \Delta\Theta $
algoritmos LS-GNLM	parámetros	$\Theta$	$ \Delta\Theta $
		0.934 -0.304 -0.648	16.23385
		0.0050 0.663 16.215	
		0.0000 0.0000	
SSD-ARC-GNLM	$\mu = 20$	0.998 0.0070 -0.749	0.91321
		-0.0030 1.004 -0.264	
		0.0000 0.0000	
SSD-ARC-GNLM	$\mu = 1.5$	0.95108 -0.30899 0.0022	0.00355
		0.30898 0.95108 0.0027	
		0.0000 0.0000	
LE-GNLM	$\sigma = 25$	0.954 -0.304 -0.0648	10.4385
		0.0050 0.063 10.215	
		0.0000 0.0000	
RANSAC		0.9506 -0.3093 0.2366	0.31274
		0.3092 0.9496 0.2045	
		0.0000 0.0000	

imagen a transformar, la figura 3.7 muestra en la parte izquierda la imagen origen y a su derecha la imagen referencia a la cual se aplico el vector  $\hat{\Theta}$ .

La tabla 3.3 presenta el resultado del método RANSAC usando las mismas imágenes, como también los resultados para las funciones LS, LE, SSD-ARC. De acuerdo con los resultados presentado en la tabla 3.3, SSD-ARC-GNLM tiene mejor desempeño que todos los algoritmos presentados en esta tabla, la ventaja de este método es la función SSD-ARC la cual tiene un buen comportamiento ante la presencia de datos atípicos y aun sin ellos, también se observa que LS hace una buena estimación ya que la imagen no tiene datos atípicos, LE tiene un bajo desempeño al igual que RANSAC.

#### 3.8.4. Imagen Sintética de la Tierra con Datos Atípicos

En este experimento, en contraste con el anterior, pondremos en negro (valor cero) el 60% de los píxeles de la imagen, para simular datos atípicos. Usando la misma transformación del experimento 3.8.3, la imagen referencia se muestra en la figura 3.8. De manera similar al experimento anterior, la tarea de registro fue realizada usando LS, LE, SSD-ARC y RANSAC en combinación con el método de optimización GNLM para minimizar la funcione

Tabla 3.3: Resultados para imagen Tierra, usando LS, LE, SSD-ARC, RANSAC con el método de optimización GNLM

algoritmos	Iteraciones	parámetros	$\Theta$	$ \Delta\Theta $
LS-GNLM	60		0.8 -0.3 20.0 0.3 0.95109 -20.0 0.0000 0.0000	2.23968E-4
SSD-ARC-GNLM	500	$\mu = 20$	0.8 -0.3 20.0 0.299 0.799 -19.999 0.0000 0.0000	8.3941E-5
SSD-ARC-GNLM	67	$\mu = 5000$	0.8 -0.3 20.0 0.299 0.799 -19.999 0.0000 0.0000	1.59107E-4
LE-GNLM	118	$\sigma = 25$	0.8 -0.3 20.002 0.3 0.8 -20.008 0.0000 0.0000	0.00843
RANSAC			0.9449 -0.3078 0.7086 0.3038 0.9473 1.2068 0.0000 0.0000	1.3994

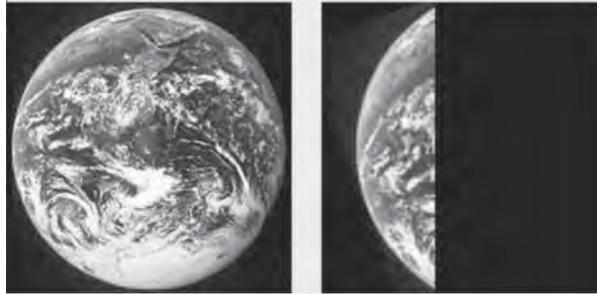


Figura 3.8: Experimento con la imagen tierra usando SSD-ARC-GNLM con datos atípicos

Tabla 3.4: Resultados para imagen Tierra, usando las funciones LS, LE, SSD-ARC, RANSAC con el método de optimización GNLM, con datos atípicos

algoritmos	Iteraciones	parámetros	$\Theta$	$ \Delta\Theta $
LS-GNLM	200		0.318 -0.108 7.216 -0.636 0.287 36.759 -0.0090 0.0000	58.1932
SSD-ARC-GNLM	11	$\mu = 20$	0.27 -0.149 8.452 -0.206 0.0358 8.684 0.0000 -0.012	30.94001
SSD-ARC-GNLM	175	$\mu = 1.5$	0.8 -0.299 19.999 0.299 0.8 -19.999 0.0000 0.0000	1.39392E-4
LE-GNLM	11	$\sigma = 25$	0.805 -0.306 20.269 0.307 0.796 -20.269 0.0000 0.0000	0.3819
RANSAC			0.9506 -0.3093 0.2366 0.3092 0.9496 0.2045 0.0000 0.0000	0.31274

de error.

En la tabla 3.4 se aprecia como el método de minimización GNLM con la función de error SSD-ARC tiene un buen desempeño, el vector encontrado por este algoritmo es muy parecido al vector real, se observa que modificando el valor de  $\mu$  para SSD-ARC se logra mejorar la solución, así como la suma de las diferencias al cuadrado entre las imágenes, se observa como este es el mejor método al presentarse datos atípicos, si bien el método RANSAC llega a la mejor solución después del método GNLM con la función SSD-ARC esta no es tan buena para crear un mosaico de imágenes.

### 3.9. Conclusiones

Hemos presentado un técnica para el registro paramétrico de imágenes basado en SSD con una condición de reposo adaptable. De las pruebas numéricas realizadas con imágenes sintéticas e imágenes reales se puede ver lo robusto del método ya que se utilizaron los mismos parámetros  $\sigma$  de derivadas y  $\mu$  en presencia de ruido. SSD-ARC esta basado totalmente en la minimización de una función de energía, lo cual permite en principio, utilizar algoritmos de minimización más eficientes computacionalmente.

En este capítulo, presentamos el método de optimización SSD-ARC-GNLM, un algoritmo para el registro paramétrico de imágenes, basado en el algoritmo de registro no paramétrico SSD-ARC. El objetivo de esta función es minimizar la función de error SSD-ARC refinando la solución dada por el método RANSAC y mínimos cuadrados con el método de optimización GNLM. El algoritmo final se muestra robusto aun cuando hay datos atípicos en las imágenes. La comparación realizada con otros métodos de registro como lo es RANSAC y LE, fue presentada demostrando lo robustos del método SSD-ARC-GNLM. Probamos SSD-ARC-GNLM usando diferentes medidas de datos atípicos y en todos los casos, SSD-ARC-GNLM mejora el resultado de los métodos RANSAC y LE.

El método SSD-ARC-GNLM prueba una nueva forma de registrar imágenes con datos atípicos y no necesitar un proceso extra para procesar los datos atípicos de la imagen. En el caso de las imágenes presentadas en este capítulo, la medida de comparación usada fue el vector de parámetros original para los métodos SSD-ARC-GNLM y RANSAC. El método SSD-ARC-GNLM tiene la ventaja de calcular los datos atípicos dentro de la función a minimizar. En contraste con el método RANSAC, SSD-ARC-GNLM calcula los datos atípicos para cualquier imagen usando una simple ecuación. Esta ecuación para datos atípicos esta implícita en la función paramétrica SSD-ARC.



## Capítulo 4

# Aplicación del Registro Automático: Mosaicos de Imágenes en 2-D

En este capítulo se propone una aplicación de construcción de mosaicos panorámicos basada en el registro de correspondencias de puntos y registro basado en tono de gris que se ha analizado en capítulos anteriores. Se aplicarán las técnicas adecuadas para la óptima selección del conjunto de puntos característicos, así como un método de registro de imágenes que explota de forma eficiente la información suministrada por la selección de los puntos característicos, para obtener una buena estimación de la homografía final.

### 4.0.1. Planteamiento General

El proceso de construcción de un mosaico conlleva generalmente a la realización de dos pasos: el alineamiento de las imágenes de la secuencia, la integración de estas imágenes en la imagen mosaico.

El proceso de *alinear las imágenes* es el establecimiento del sistema de referencia y alineamiento de la secuencia. Para realizar el proceso de alineación de las imágenes respecto a un sistema de referencia común es preciso definir: cuál va a ser este sistema de referencia y qué tipo de transformación o modelo de movimiento se va a usar durante el proceso de registro que permita realizar el alineamiento. En función de cómo se establezca el sistema de referencia y entre qué pares de imágenes se realice el proceso de registro es posible plantear

tres aproximaciones diferentes:

- Alinear el cuadro al mosaico (método *frame to mosaic*): Durante la construcción de mosaicos estáticos, habitualmente el sistema de referencia sobre el cual se alinean las imágenes de la secuencia lo establece una de estas imágenes, que actúa como cuadro de referencia (frecuentemente, el primero de los cuadros de la secuencia). Inicialmente el mosaico está formado únicamente por este cuadro de referencia. Después se va registrando cada uno del resto de cuadros respecto a la imagen mosaico obtenida hasta ese momento y se añade a él, hasta completar el mosaico final, que integra la totalidad de los cuadros de la secuencia.
- Alinear cuadro a cuadro (método *frame to frame*): Este método consiste en realizar el registro entre cada cuadro de la secuencia y el cuadro inmediatamente anterior. Los parámetros de alineación pueden ser entonces compuestos para obtener la alineación de los cuadros respecto a cualquier cuadro de la secuencia que actúe como cuadro de referencia. Este método tiene la ventaja de que al realizar el registro sobre imágenes temporalmente adyacentes, el desplazamiento producido habitualmente será pequeño y por tanto fácil de estimar. El inconveniente de este método es que los errores ocurridos durante el registro de cualquier par de cuadros adyacentes pueden ir acumulándose al componer los parámetros de alineación. Una solución a este problema consiste en utilizar estos parámetros como estimación inicial de alineación respecto al mosaico, para luego ser refinada mediante registro directo del cuadro con el mosaico.
- Alinear el mosaico al cuadro (método *mosaic to frame*): Esta solución es adecuada para la construcción de mosaicos dinámicos, donde interesa que el mosaico se encuentre permanentemente alineado respecto al cuadro actual de la secuencia.

## 4.1. Construcción de mosaicos

Recientemente ha existido gran interés en el uso de imágenes de mosaicos para representar la información contenida en un vídeo como muestra Shum y Tomasi en [Shum98], [Tomasi98] respectivamente. Los mosaicos se construyen alineando e integrando todos los

cuadros de una secuencia de vídeo dada, ofreciendo una vista panorámica de la escena. Como las sucesivas imágenes de una secuencia habitualmente presentan una importante superposición, las imágenes de mosaico proporcionan una considerable reducción en la cantidad de información requerida para representar la información visual contenida en la secuencia.



Figura 4.1: Construcción de un mosaico

En la figura 4.1 se muestra el proceso de construcción de un mosaico, generalmente algunos trabajos realizan esto en dos pasos: el primero es el alinear las imágenes de la secuencia y el segundo es el que integra las imágenes en el mosaico. El alineamiento de las imágenes debe realizarse con algún método de registro. Los métodos utilizados para este propósito pueden clasificarse en dos grupos: métodos basados en correspondencias entre puntos característicos y métodos de registro paramétrico o basados en tono de gris. Para este trabajo de tesis el primer paso es seleccionar en una de las imágenes una serie de puntos característicos que para este trabajo la tarea se realiza por medio del método KLT y se buscan en la segunda imagen los puntos con que se deben alinear para encontrar las correspondencias, esto se realiza buscando la correlación putativa entre los puntos localizados en ambas imágenes para esta tarea se utilizó el coeficiente de correlación. A partir del conjunto

de pares de posiciones correspondientes el método de registro basado en correspondencias estima los parámetros de la transformación geométrica que alinea ambas imágenes de forma óptima, eliminando los datos atípicos con el método RANSAC, este método le pasa los puntos correspondientes sin datos atípicos al método de mínimos cuadrados para solucionar el sistema que nos dará la primera aproximación de la Homografía. En los métodos de registro basados en tono de gris se hace uso de las funciones de la imagen de ambos cuadros para plantear un registro que utiliza la información completa de imagen, esto se realiza aplicando la técnica de registro paramétrico de imágenes, minimizando una función de error denominada SSD-ARC que es robusta a datos atípicos. Esta función será minimizada con la ayuda del método de optimización de Levenberg Marquardt. Shum y Szeliski muestran en [Szelinski94] y [Shum98] ejemplos significativos de trabajos que hacen uso de estas técnicas para la construcción de mosaicos.

#### 4.1.1. Resumiendo el Registro de Imágenes

Los métodos utilizados habitualmente en aplicaciones de construcción de mosaicos pueden clasificarse en dos grandes grupos: métodos basados en establecimiento de correspondencias y métodos de registro basados en tono de gris.

En el primer caso, el cálculo de la homografía se realiza a partir de un conjunto de correspondencias entre puntos pertenecientes a las dos imágenes a registrar. El método de registro basado en tono de gris hace uso directo de las funciones de la imagen para obtener los parámetros de la homografía, formulando la minimización de un índice de error, frecuentemente un error cuadrático de intensidades de la imagen. En la aplicación de generación de mosaicos descrita en este capítulo se implementa una versión del registro de imágenes en tono de gris, con una nueva función de error robusta a datos atípicos, denominada SSD-ARC ya descrita en el capítulo 3. Se trata de minimizar esta función de error por medio de un método de optimización como lo es el denominado Gauss-Newton-Levenberg-Marquardt.

El algoritmo obtiene a cada iteración el vector incremental de parámetros de movimiento  $\Theta$  formado por los elementos de la homografía. El índice a minimizar toma la forma de la ecuación (3.4); de esta ecuación se puede ver que  $I(x(\theta, i, j), y(\theta, i, j))$  es la transformación de la imagen  $I_r(i, j)$ . A cada paso se calcula la imagen transformada, su gradi-

ente  $\nabla I(i, j)$ , su jacobiano  $J_{ij}(\Theta)$ , el error de intensidad  $e_{ij} = I(x(\theta, i, j), y(\theta, i, j)) - I_r(i, j)$  y la resolución de un sistema de tamaño  $8 \times 8$  para hallar los parámetros del vector  $\Theta$ .

Los métodos de registro basados en correspondencias, aunque ofrecen resultados menos precisos, resultan mucho menos costosos computacionalmente ya que no necesitan procesar la imagen completa. Los métodos de registro que minimizan un índice de error, aunque ofrecen una estimación más densa y precisa, tienen un mayor costo computacional y son propensos a devolver resultados no óptimos al converger a mínimos locales, en este trabajo de tesis para evitar este problema se utiliza la primera estimación de la homografía realizada por el registro basado en correspondencia de puntos. Además, para aplicaciones de construcción de mosaicos, los métodos basados en correspondencias permiten la introducción de modelos dinámicos de movimiento y establecer planteamientos de seguimiento de imágenes como muestra Tomasi y Censi en [Tomasi98], [Censi99] respectivamente, si se logra un buen acoplamiento entre estos dos métodos se cuenta con una herramienta poderosa para la construcción de mosaicos de imágenes.

#### 4.1.2. Tipos de Mosaicos

Los mosaicos generalmente se clasifican en 2 tipos fundamentales:

- **Mosaicos estáticos.** Los mosaicos estáticos habitualmente se construyen alineando a posterior todos los cuadros de una escena respecto a un sistema de coordenadas de referencia, que puede ser definido por el usuario o automáticamente, integrando todas las imágenes de una secuencia en una única imagen mosaico. Tras esta integración, los objetos en movimiento o bien desaparecen o aparecen como trazos poco definidos en la imagen panorámica del mosaico.
- **Mosaicos dinámicos.** La limitación fundamental de los mosaicos estáticos es que deben ser construidos a posterior, es decir, cuando se dispone de todos los cuadros de la secuencia. Por esta razón no pueden seguir completamente el aspecto dinámico de la secuencia. Por el contrario, el contenido de los mosaicos dinámicos es variable y está siendo constantemente actualizado con la información del cuadro más reciente. Un mosaico dinámico puede entenderse entonces como una secuencia de imágenes de

mosaico que evoluciona, donde el contenido de cada nuevo mosaico es actualizado con la información del último cuadro disponible.

### 4.1.3. Aplicaciones de los Mosaicos

Se han estudiado muy diversas aplicaciones de los mosaicos. Entre las más significativas cabe destacar las siguientes:

- Edición y manipulación de vídeo. Odone muestra en [Odone04] un ejemplo de inclusión de un elemento sintético (un rótulo en una carretera) en una secuencia de vídeo con fuerte movimiento de la cámara. El elemento es incluido fácilmente editando la imagen mosaico generada para la secuencia.
- Realidad virtual. Una aplicación típica es la generación de recorridos virtuales de escenarios 3-D a partir de secuencias de vídeo, para el desarrollo de juegos o aplicaciones educativas o comerciales (recorridos de museos, edificios históricos, edificios en venta o alquiler, lugares turísticos, etc.) como muestra Szeliski en [Szelinski94].
- Obtención de imágenes de alta resolución mediante cámaras de resolución estándar. Zappala muestra en [Zappala99] una aplicación de este tipo encaminada a la exploración de documentos.
- Generación de imágenes de fondos marinos. García muestra en [García01] el registro de fondos marinos la cual es una técnica ampliamente utilizada en oceanografía y operaciones forenses submarinas. Las imágenes que se obtienen de un fondo marino profundo mediante una cámara instalada en un vehículo submarino no pueden ser de superficies muy extensas. Ello es debido a las limitaciones de potencia de la fuente de luz en este tipo de vehículos semiautomáticos y a la rápida absorción de esta. Por ello, superficies de varios metros de tamaño no pueden ser registradas con una única imagen, lo que convierte la técnica de generación de mosaicos en una herramienta imprescindible [Tomasi98].

## 4.2. Desarrollo de la Aplicación para la Construcción de Mosaicos

En este trabajo tratamos de desarrollar una aplicación autónoma que, integra las técnicas analizadas en los capítulos 2 y 3, en los cuales se tratan los problemas de seguimiento basado en puntos característicos, la estimación de la transformación, correspondencias entre imágenes y la optimización de los resultados obtenidos con las técnicas analizadas en el capítulo 3.

Durante el proceso de **Correspondencias**, se debe seleccionar un conjunto de puntos característicos o puntos de control, los cuales serán posteriormente seguidos de cuadro a cuadro. El proceso de selección de puntos característicos, en principio, podrá seleccionar cualquier punto de la imagen que presente buenas propiedades que aseguren su correcta localización a lo largo de la secuencia de imágenes. Sin embargo, se introducirán en este proceso de selección algunas consideraciones específicas, de forma que se favorezca la precisión del posterior proceso de registro basado en las posiciones de los puntos característicos seleccionados. La forma como se lleva a cabo esta etapa de selección se describe con detalle en el capítulo 2, que trata de los aspectos generales del algoritmo de seguimiento.

Una vez completada la etapa de correspondencias, el *Módulo de Transformación de la Imagen* suministra para cada nuevo cuadro de la secuencia una estimación de la posición en la imagen de cada uno de los puntos característicos, así como una valoración de la calidad de esta estimación. El algoritmo de seguimiento utilizado se ajusta, en esencia, al funcionamiento descrito en el Capítulo 2.

A partir de la información suministrada por el módulo de correspondencias de puntos, el módulo de generación del mosaico tiene como tarea la construcción progresiva de una imagen de mosaico que integre las diferentes vistas de la escena ofrecidas por los cuadros de la secuencia, con un adecuado alineamiento. En líneas generales, el procedimiento de alineamiento seleccionado para la construcción del mosaico corresponde al método denominado "*frame to mosaic*" (alinear el cuadro al mosaico).

El algoritmo de correspondencias ofrece estimaciones del desplazamiento de los puntos seguidos respecto al cuadro del que fueron extraídos. Es por ello que resulta conve-

niente esta definición de cuadro de referencia: de esta forma la estimación de los parámetros de alineamiento de un cuadro dado podrá hacerse adecuadamente a partir de las posiciones estimadas de los puntos en él y las posiciones iniciales en el cuadro donde fueron extraídos.

Como ya se ha comentado, uno de los inconvenientes de los métodos de registro basados en correspondencias como el que se plantea aquí es su reducida precisión esto se debe a que estos algoritmos no analizan todos los casos posibles de solución, esta reducida precisión puede dar lugar a deficiencias de registro que lleguen a resultar apreciables en el mosaico construido. Para solventar este problema, la solución que se ha establecido consiste en integrar el método de registro basado en correspondencias con un segundo método de registro que minimiza un índice de error. Esta segunda etapa de registro hará uso de la estimación inicial ofrecida por el primer método y de la función de imagen correspondiente al cuadro actual y al mosaico construido hasta el instante actual, para obtener una estimación de registro de mayor precisión. Se pretende de esta forma explotar las ventajas de ambos métodos de registro, consiguiendo resultados con alta precisión al tiempo que se mantiene un costo computacional limitado. Una vez que han sido determinados con precisión los parámetros de alineamiento del último cuadro procesado, éste es añadido al mosaico mediante un proceso de integración.

Abordando lo ya señalado en el capítulo primero, un mosaico de imágenes consiste en una composición de varias imágenes relacionadas por un mapeo global, para formar una imagen panorámica. Paixao muestra en [Paixao03] algunos ejemplos de Mosaicos de Imágenes encontrados en la literatura así como en la red.

#### 4.2.1. Obedeciendo Algunas Reglas para la Construcción de Mosaicos

Obligatoriamente las dimensiones de las imágenes de entrada deberán de ser iguales, deberán ser en algún formato de imagen como por mencionar algunos *.jpg*, *.gif*, *.png*, etc. Estas imágenes se guardaran en el directorio `\MosIma\fotos\` dentro de la aplicación. Un mosaico terminado será gravado en el directorio `\MosIma\resultados\`, en formato *.jpg* o *.png*. Las imágenes para cada mosaico deberán estar en orden dentro de la secuencia ya que de lo contrario la ampliación aun no esta dotada de una solución para cuando no se encuentren partes correspondientes también deben de ser identificadas por un conjunto de caracteres,

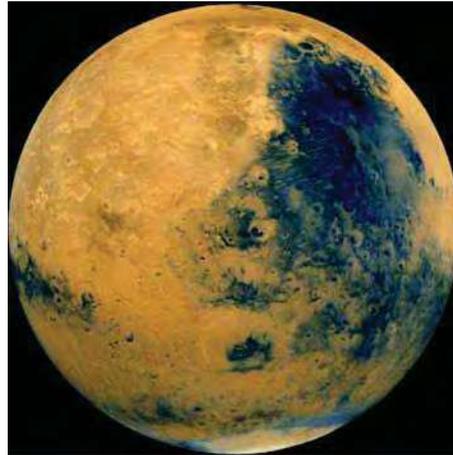


Figura 4.2: Mosaico esférico de la luna, obtenida a partir de imágenes de Viking Orbiter, en 1980

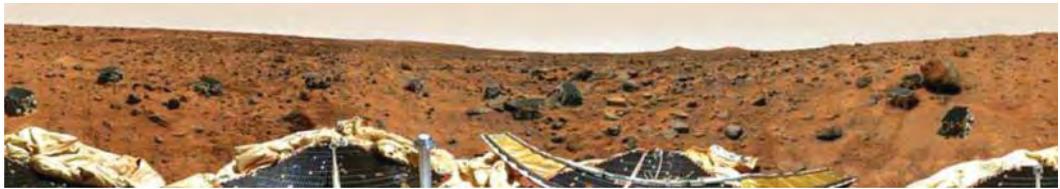


Figura 4.3: Mosaico de la superficie de Marte, obtenido a partir de fotografías de Mars Pathfinder, de Octubre de 1998



Figura 4.4: Imagen panorámica de un edificio (secuencia tomada de [Paixao03]) reconstruido con nuestra aplicación

seguido de un número secuencial y la extensión de la imagen, por ejemplo *fuentes1.jpg*, *fuentes2.jpg*, etc. Las imágenes podrán ser tomadas a color, la aplicación las procesará en tono de gris, para esta versión de nuestra aplicación aún no se dota a esta para el manejo de las imágenes en color, cabe mencionar que si se desea componer un mosaico en color, se deben aplicar los mismos procedimientos para cada banda de color hablando de RGB. Los mosaicos podrán ser construidos de derecha a izquierda o de izquierda a derecha.

---

**Algoritmo 5** Algoritmo General para la Construcción de Mosaicos

---

1. Escoger manual o automáticamente, puntos comunes entre dos imágenes, que deberán ser un mínimo de 4, respetando obligatoriamente el orden de marcado de estos puntos.
2. Estimar una primera aproximación de la homografía  $H_1$  con el método de correspondencias de puntos.
3. Utilizando la primera estimación de la homografía  $H_1$  como estimación inicial para el método de registro en tono de gris, esto se hace para optimizar la homografía y calcular una homografía final.
4. Calcular la imagen final.
5. Empatar las dos imágenes en una única.

---

Esta aplicación permitirá la construcción de mosaicos panorámicos a partir algunos modos de operación por ejemplo podrá resolver imágenes tomadas por cámaras digitales o fotográficas que presenten una escena la cual tenga elementos en común, también, podrá utilizar secuencias de imágenes ya sean de vídeo o de cámara fotográfica, si es un vídeo se extraerán los frames de las imágenes para que estos sean procesados como una secuencia de imágenes.

La construcción de mosaicos de imágenes obedece a reglas de cuyo cumplimiento

depende el resultado final:

- La distancia entre la escena y la cámara debe de ser muy superior a las dimensiones de esta toma.
- Si esta distancia no es significativamente grande, entonces solo puede existir el movimiento de rotación entorno al centro óptico de la cámara, y no puede haber movimiento de traslación.
- En cada conjunto de dos imágenes las imágenes deben interceptarse en una parte de la imagen.

Esta *repetición* permite que sean identificados algunos puntos comunes en las dos imágenes (correspondencias), estos puntos servirán de base para el calculo de la homografía a través del cual una segunda imagen será procesada. Para cada conjunto de dos imágenes es necesario seguir los pasos expuestos por el algoritmo 5. Así sean mas de dos imágenes, el ciclo se repite atendiendo todos los puntos del algoritmo 5 para que las imagenes sean procesadas.

### 4.3. Relación entre Dos Imágenes

Al establecer una relación entre un plano del mundo real y una imagen, también será posible establecer una relación entre dos imágenes diferentes del mismo plano real. Esto se conoce como homografía, el esquema de la definición de una homografía se presenta en la figura 4.5.

De la figura 4.5 se verifica que el punto  $X$  es el mismo. Así mismo, se establece la homografía entre estas dos imágenes:

$$\begin{cases} x_1 = H_1 X \\ x_2 = H_2 X \end{cases} \quad \begin{cases} X = H_1^{-1} x_1 \\ x_2 = H_2 H_1^{-1} x_1 \end{cases} \quad x_2 = H_{12} x_1 \quad H_{12} = H_2 H_1^{-1} \quad (4.1)$$

Analizando la definición 4.1 en conjunto con la figura 4.5, podemos observar que con  $x_1 = H_1 X$  vamos de la imagen 1 a el mundo plano  $X$  y con  $x_2 = H_2 X$  vamos de la

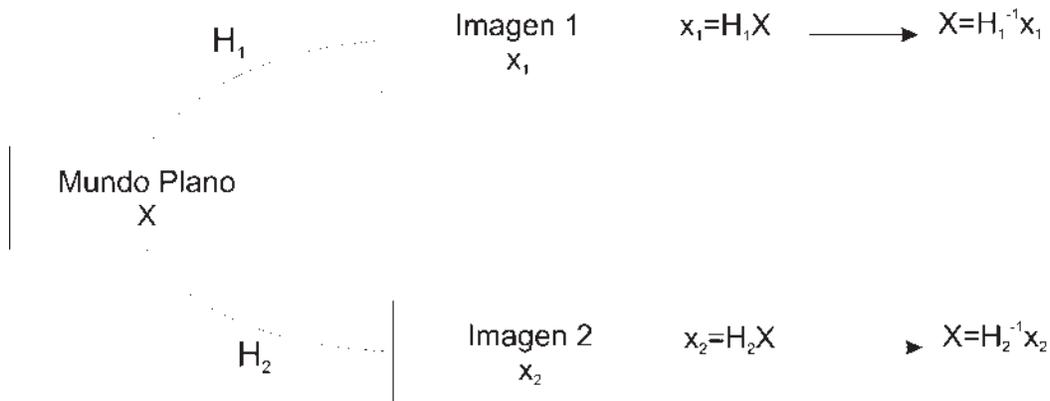


Figura 4.5: Mapeo de un punto de un plano en dos imágenes diferentes

imagen 2 a el mundo plano  $X$ , por consiguiente del mundo plano  $X$  podemos ir a la imagen 1 por medio de la inversa de la homografía como se aprecia por  $X = H_1^{-1} x_1$  sustituyendo  $X$  en  $x_2 = H_2 X$  tenemos que  $x_2 = H_2 H_1^{-1} x_1$ , para empatar la imagen 1 y la imagen 2 basta con relacionar sus homografía.

Tanto  $H_1$  y  $H_2$  como sus inversas son matrices de homografía, al igual que el producto de una de las matrices inversa da otra. Así mismo, una matriz  $H_{12}$  es también una matriz de homografía y relaciona a las dos imágenes.

De lo anterior se concluye que cualquier punto de la imagen 1 puede ser mapeado a la imagen 2 a través de una transformación por medio de la matriz de homografía correspondiente, y viceversa. Esta conclusión es muy importante, por tratase del principio básico para la construcción de mosaicos, partiendo de varias imágenes y conociendo los puntos de correspondencia entre ellas.

La figura 4.6 presenta en forma esquemática una secuencia de imágenes relacionadas por las matrices de homografía. Por ejemplo de la Imagen 1 a la imagen 2 las relaciona la matriz de homografía  $H_{12}$  y de la imagen 2 a la 3 las relaciona la matriz de homografía  $H_{23}$  y todas las imagines estarán relacionadas por la matriz  $H_{15}$

$$x_1 = H_{12}x_2 \quad x_2 = H_{23}x_3 \quad x_3 = H_{34}x_4 \quad x_4 = H_{45}x_5 \quad (4.2)$$

de la ecuación 4.2 se toma el valor de  $x_1$ ,  $x_2$ ,  $X_3$ , y  $x_4$  y lo sustituimos quedando la ecuación

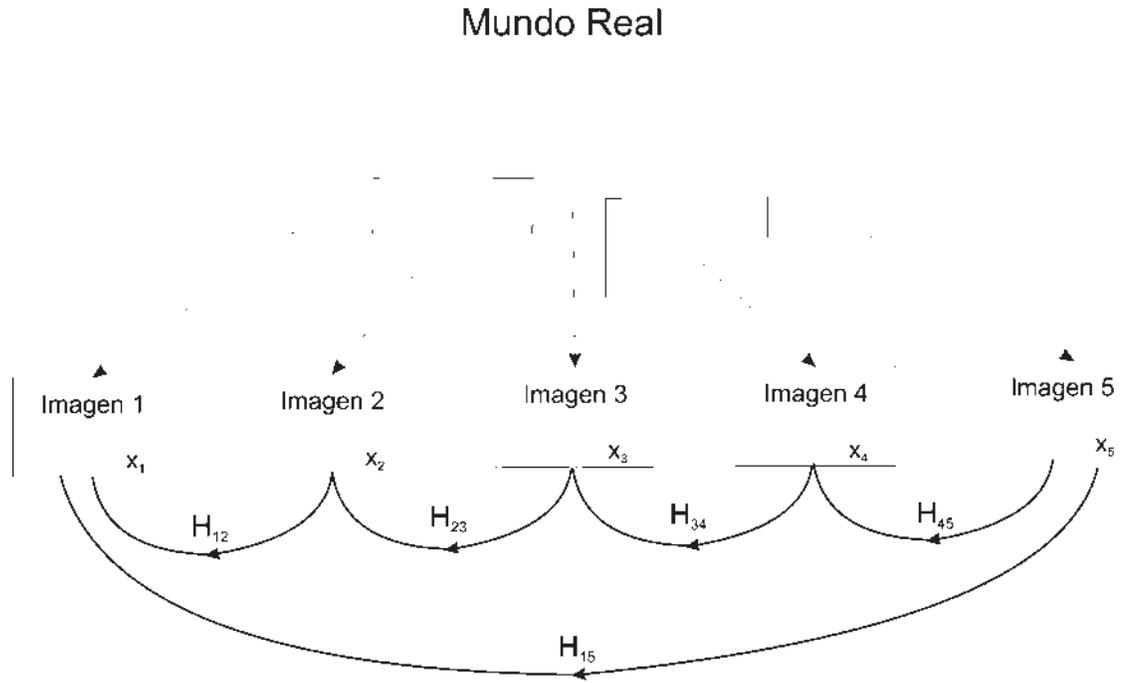


Figura 4.6: Relación de las imágenes con una homografía

4.3, donde  $H_{15}$  relaciona todas las matrices de homografía para la secuencia de imágenes.

$$x_1 = H_{12}H_{23}H_{34}H_{45}x_5 \quad , \text{despues} \quad H_{15} = H_{12}H_{23}H_{34}H_{45} \quad \rightarrow \quad x_1 = H_{15}x_5 \quad (4.3)$$

#### 4.4. Desarrollo Práctico del Trabajo

Ya estudiadas las técnicas correspondientes a cada uno de los problemas planteados al inicio de nuestro trabajo, así como también analizado el resultado de cada etapa planteada. En este capítulo analizaremos el desarrollo práctico que se lleva a cabo para la construcción de los mosaicos de imágenes, ya se definió en 4.2.1 las características que deben presentar las imágenes para ingresar a el sistema así como algunas de las restricciones que se manejan en la aplicación.

Dada la relativa complejidad del proyecto será necesario dividirlo en varias tareas, de acuerdo a la siguiente secuencia:

- Seleccionar de las dos imágenes algunos puntos de control.
- Estimar la correspondencia entre los puntos de control.
- Estimar la matriz  $H$
- Unir las imágenes.
- Construir un mosaico para mas de dos imágenes

#### 4.4.1. Correspondencia entre Puntos

Con esta tarea se pretende relacionar dos imágenes de una secuencia, escogiendo automática o manualmente puntos correspondientes.

La selección de puntos característicos se realiza automáticamente por medio del método KLT la figura 5.1 muestra la selección de los puntos realizada por KLT. Después de esta tarea se realiza la de estimar los puntos correspondientes que para el caso del modelo proyectivo deben de ser 4 puntos correspondientes, estas tareas están descritas en el capítulo 2.

#### 4.4.2. Estimar la Matriz H

Para llevar a cabo la tarea de estimar la matriz H es necesario seguir los pasos mostrados en el algoritmo 6.

Después de leer las dos imágenes se aplican los métodos restantes para estimar la matriz H, los cuales se describen en los capítulos 2 y 3 de este trabajo.

#### 4.4.3. Transformar y Empatar las Imágenes

Después de la estimación de la matriz  $H$ , el resultado final es la homografía entre las imágenes, las dimensiones de la imagen final se calculan a través del producto de todas las matrices de homografía y las coordenadas entre las primera y la ultima imagen como se analizara en el algoritmo 7.

En la figura 4.7 se muestra la construcción de la imagen final y las dimensiones de esta antes de colocar las demás imágenes que compondrán el mosaico final.

---

**Algoritmo 6** Algoritmo General para Estimar la Matriz  $H$ 

---

1. Leer las dos imágenes  $I_1$  e  $I_r$  y calcular los puntos de control (determinados por el usuario) en cada imagen usando el algoritmo KLT mostrado por Lucas en [Lucas81].
  2. Calcular el conjunto de puntos de control que empatan las intensidades del vecindario (usando correlación) con la ecuación (2.7).
  3. Eliminar datos atípicos con RANSAC y calcular la matriz  $H$  inicial con mínimos cuadrados analizados en 2.3.
  4. Refinar la matriz  $H$  con el registro en tono de gris analizado en 3.7.
  5. Regresar a 1 si hay mas imágenes sino terminar.
  6. Regresar  $H$
- 



Figura 4.7: Dimensión del mosaico final  $605 \times 799$  usando imágenes de  $640 \times 416$ , con la imagen inicial en ella

---

**Algoritmo 7** Algoritmo para Conocer las Dimensiones del Mosaico Final y Colocar la Imagen Origen en Ella

---

Teniendo como entradas la imagen 1, la ultima imagen y la matriz de transformación  $H$  ( $H \times H \times \dots \times H$ ).

1. Calculamos las dimensiones de las imágenes  $[nl1, nc1] = dimension(imagen1)$  y  $[nl2, nc2] = dimension(imagenultima)$ .
  2. Transformamos las coordenadas cartesianas a coordenadas homogéneas de la ultima imagen con ecuación (3.3).
  3. Verificamos las coordenadas exteriores de la imagen  $[x_{min}, x_{max}, y_{min}, y_{max}]$ .
  4. Creamos un vector  $delta = [(1 - x_{min}), (1 - y_{min})]$ .
  5. Construimos la imagen final con  $I_{final}(y_{max} - y_{min} + 1, x_{max} - x_{min} + 1) = -1$
  6. Copiamos la imagen 1 a la imagen a regresar con  $I_{final}(1 + delta(2) : nl1 + delta(2), 1 + delta(1) : nc1 + delta(1)) = I_1(x, y)$ .
- 

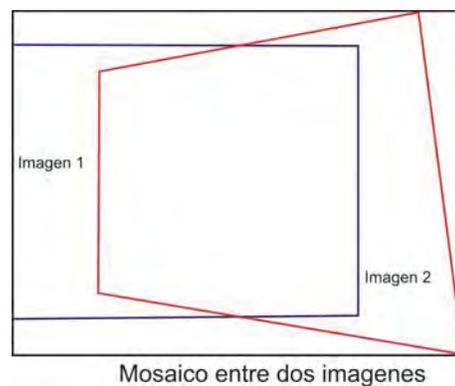


Figura 4.8: Definición del área del mosaico, en relación a la imagen referencia (azul) y la imagen transformada (roja)

Un mosaico puede ser construido de izquierda a derecha, siendo sus dimensiones obtenidas de la comparación entre las coordenadas de la primera imagen y las coordenadas de transformación de la segunda imagen. Encontrando los máximos y los mínimos de esta comparación se define la dimensión de la imagen final, como se puede ejemplificar en la figura.

Como los valores resultantes de la lectura de una imagen son siempre positivos, se iniciara la imagen final con valores de  $(-1)$  los cuales funcionarían como una máscara, de modo que los datos que se añadan de las nuevas imágenes puedan ser identificados fácilmente.

Los pasos para unir dos o más imágenes son los descritos en el algoritmo 8. De la ecuación 4.4 se define  $H$  como la matriz de homografía,  $p_{coor}$  como la matriz de coordenadas homogéneas,  $p_{cx}$  son las coordenadas cartesianas en  $x$  y  $p_{cy}$  son las coordenadas cartesianas en  $y$ .

$$\begin{aligned} p_{aux} &= H \times p_{coor} \\ p_{cx} &= \frac{p_{aux}(1)}{p_{aux}(3)} \\ p_{cy} &= \frac{p_{aux}(2)}{p_{aux}(3)} \end{aligned} \tag{4.4}$$

El procedimiento se realiza sucesivamente colocando las imágenes para construir un mosaico final.

En la figura 4.9 se muestra la imagen (a), que es la definición de las dimensiones del mosaico final con la imagen incida en él, este es el paso presentado en el algoritmo 7, la figura (b) contiene la imagen inicial y una segunda imagen ya empataadas, (c) presenta el mosaico final, lo que serían los pasos presentados en el algoritmo 8.

---

**Algoritmo 8** Algoritmo para Conocer las Dimensiones del Mosaico Final y Colocar la Imagen Origen en Ella

---

Teniendo como entradas la imagen 1 después de ser procesada por el algoritmo 7, la imagen 2 y la matriz de transformación  $H$  ( $H \times H \times \dots \times H$ ).

1. Calculamos las dimensiones de las imágenes  $[nl1, nc1] = dimension(imagen1)$  y  $[nl2, nc2] = dimension(imagenultima)$ .
  2. Creamos una matriz  $imsal$  de las mismas dimensiones que la imagen 1 y le colocamos el valor de  $-1$  así  $imsal = -1$ .
  3. Creamos una matriz  $im2sal$  de las mismas dimensiones que la imagen 1 y colocamos la imagen 2 definiendo  $im2sal(1 : nl2, 1 : nc2) = imagen2(i, j)$ .
  4. Colocamos en un vector  $Pcoor$  las coordenadas homogéneas.
  5. Convertimos las coordenadas homogéneas a coordenadas cartesianas a través de la matriz de homografía con la ecuación (4.4).
  6. Mapeados la imagen 2 con las coordenadas cartesianas calculadas, con la función de interpolación de la sección 3.2.1.
  7. Ponemos en  $-1$  todos los puntos que no pertenezcan a la imagen 2.
  8. Sobreponemos la imagen 2 a través de las máscaras calculadas con  $imsal(i, j) = imsal(i, j).*(imsal(i, j) = -1) + im2sal(i, j).*(imsal(i, j) == -1)$
  9. Terminamos.
-

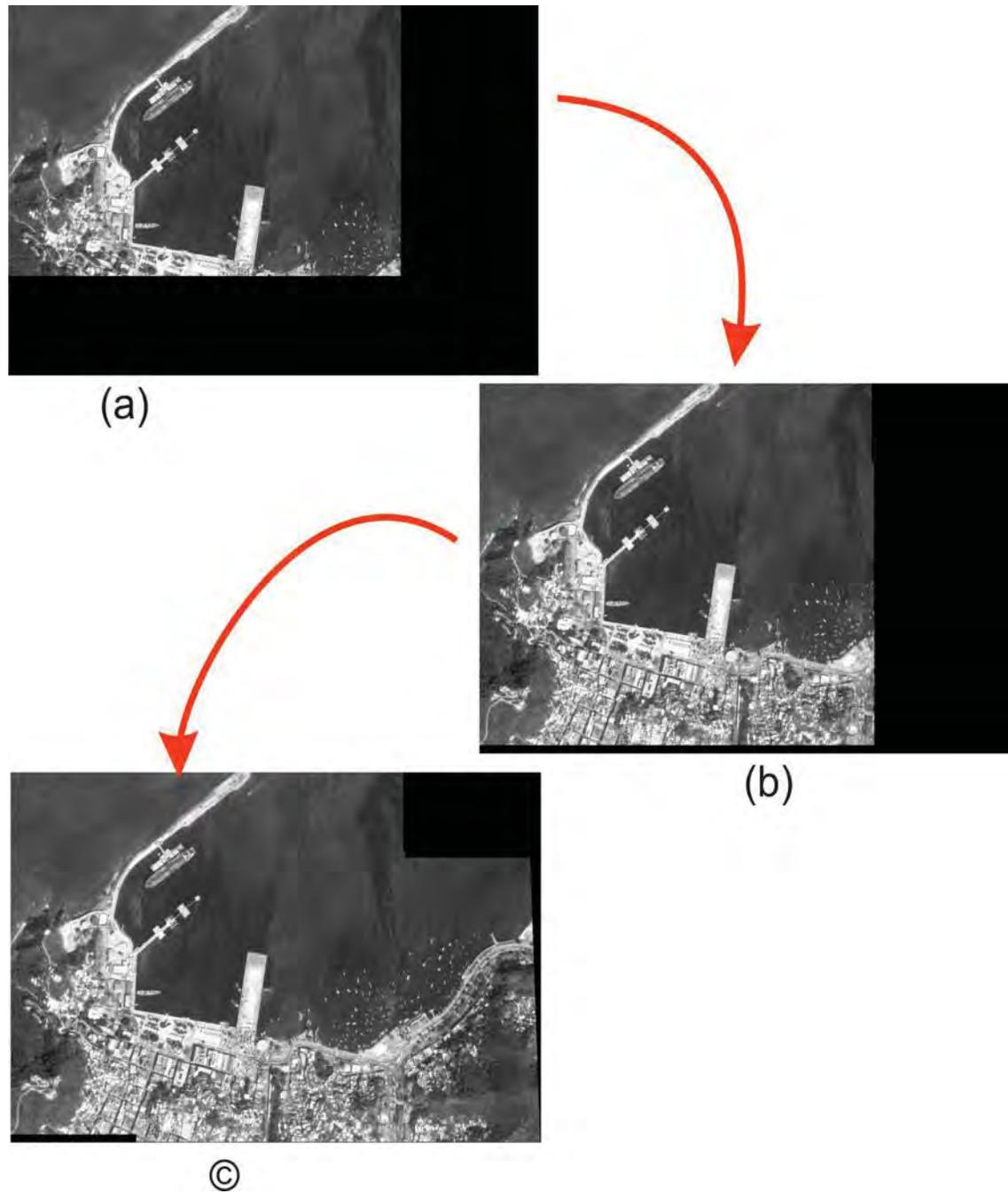


Figura 4.9: (a) Mosaico con la Imagen Inicial, (b) Mosaico con la secuencia de imágenes 1 y 2, (c) Mosaico final



## Capítulo 5

# Resultados

### 5.1. Registro Basado en Puntos Característicos

Los métodos de selección de puntos característicos habitualmente evalúan una medida de calidad en un área de selección y escogen finalmente un número determinado de puntos que obtienen los más altos valores de interés. Debido a esta forma de operar y a que las imágenes frecuentemente presentan áreas con diferente textura, un hecho que sucede frecuentemente en la práctica es que los puntos seleccionados tienden a concentrarse en las áreas de mayor textura de la imagen.

#### 5.1.1. Selección de Puntos Característicos

En la figura 5.1 se muestra como a el método de selección de puntos característicos KLT se le puede dar un número determinado de puntos a seleccionar, ventaja con la que otros algoritmos cuentan, en la imagen se muestran 20 puntos seleccionados para ambas imágenes.

#### 5.1.2. Búsqueda de Correspondencias

En este apartado se detalla con un ejemplo práctico el comportamiento del coeficiente de correlación el cual es un método que considera que todos los puntos de la vecindad de los píxeles alrededor del punto característico sufre el mismo desplazamiento como muestra



Figura 5.1: Puntos característicos marcados con el método KLT

Black en [Black96].



Figura 5.2: Puntos correspondientes seleccionados por el coeficiente de correlación

Se puede observar como en las imágenes las líneas marcadas en ambas imágenes indican que puntos son correspondientes, para este caso son cuatro los puntos que se indican pero si observamos detenidamente las imágenes se encuentran algunos otros puntos que son correspondientes en las dos imágenes al igual que podemos observar como en el círculo punteado de la imagen izquierda hay dos puntos que no tienen correspondencia alguna con los puntos marcados en la imagen derecha.

### 5.1.3. Determinación de la Transformación de la Imagen

Ya analizados los problemas de selección de puntos y búsqueda de correspondencias le asignaría el turno a el problema de determinar la posición de la imagen para la construcción de mosaicos de imágenes, una vez determinados los parámetros de desplazamiento, se plantea a continuación la estimación del conjunto de parámetros de movimiento, lo que habitualmente se conoce como posición del objeto. La siguiente etapa del proyecto se centra en determinar los parámetros del modelo de movimiento utilizado para este caso el modelo proyectivo.

#### RANSAC

En la figura 5.3 se puede observar como RANSAC elimina las falsas correspondencias o datos atípicos entregados por el coeficiente de correlación dejando para estas imágenes solo puntos correspondientes en ambas imágenes.



Figura 5.3: Puntos Correspondientes después de eliminar los datos atípicos por RANSAC

#### Mínimos Cuadrados

El trabajo que le queda hacer a mínimos cuadrados es estimar la  $H$  Homografía después de que RANSAC le entrega datos completamente limpios de falsas correspondencias por lo cual se dice que el trabajo realizado por mínimos cuadrados es muy limpio, este método

se encarga de resolver el sistema sobredeterminado que tenemos en 2.8. Hasta este momento tenemos nuestra primera estimación de  $\Theta_0$  la cual se pasara a el método de registro para mejorar con el método de optimización que minimiza la función de error SSD-ARC.

## 5.2. Construcción del Mosaico

La construcción del mosaico implica la estimación de los parámetros de alineamiento de cada cuadro de la secuencia respecto al cuadro en el que se realizó la extracción de los puntos característicos (cuadro de referencia), así como la integración de los cuadros en la imagen mosaico.

La imagen mosaico se inicializa al cuadro de referencia y a partir de aquí, los siguientes cuadros de la secuencia se van integrando sucesivamente en el mosaico transformándolos de acuerdo a los parámetros de alineamiento estimados.

Como ya se ha referido, para la estimación de los parámetros de alineamiento se utilizan dos estrategias diferentes y complementarias. En primer lugar se calcula la homografía que relaciona el cuadro actual  $f_k$  con el cuadro de referencia  $f_{ref}$  haciendo uso de las correspondencias entre puntos suministradas por el seguidor. Como esta primera estimación de la homografía puede estar influenciada negativamente por los pequeños errores de seguimiento ocurridos, se establece a continuación un segundo paso de registro mediante un método directo con el fin de mejorar la precisión de la estimación. Este segundo procedimiento realiza un registro entre el cuadro actual y la siguiente imagen a integrar en el mosaico mediante el método descrito en [Paixao03]. En este segundo paso se utiliza la homografía suministrada por el primer paso de registro basado en correspondencias como estimación inicial. Los parámetros de registro obtenidos como resultado de este segundo proceso de registro serán los que se usen finalmente para integrar el cuadro actual en el mosaico.

El paso de registro es un proceso mucho más costoso computacionalmente que la estimación basada en correspondencias, pero ofrece una estimación más precisa al hacer uso de información de la imagen completa.

### 5.3. Mosaicos de Mas de Dos Imágenes

Cuando un mosaico es construido para mas de dos imágenes, se utiliza un procedimiento recursivo para unir varias imágenes, este procedimiento es el mismo utilizado para dos imágenes. La diferencia mas significativa es que todas las imágenes son reverenciadas a la primera, y es preciso multiplicar las diversas matrices  $H$ , resultantes del proceso de estimación de esta, la figura 4.8 ejemplifica como se define la dimensión final del mosaico.

Partiendo del principio de que la cámara nunca realiza un movimiento de retorno, ni en dirección de la primera imagen, la dimensión final del mosaico es obtenida de la comparación entre coordenadas de la primera imagen y las coordenadas de la ultima imagen, después de haber sido transformada a través del producto de todas las matrices  $H$ , de acuerdo con el esquema de la figura 4.8.

En este apartado se muestra el proceso de construcción de varios mosaicos de acuerdo al método desarrollado en este capítulo. La Tabla 5.1 recoge los valores de los parámetros que son comunes a todos los experimentos presentados en este apartado.

Tabla 5.1: Valores de los diferentes parámetros durante los experimentos.

parámetro	Valor
$\sigma$ de derivada	0.5
numero de puntos a seleccionar KLT	500
$\mu$ SSD-ARC	20

#### 5.3.1. Mosaico Posgrado FIE

La Figura 5.4 muestra algunos cuadros de la secuencia “Posgrado FIE” tomada por el autor de este trabajo con una cámara digital Kodak DC7630 de 6.1 megapíxeles en modo video del cual se extrajo una secuencia de imágenes de 10 cuadros para trabajar con estas imágenes, esta secuencia es utilizada para realizar la construcción del primer mosaico que se presenta. Se trata de una secuencia de 10 cuadros de imagen de un tamaño de  $320 \times 240$  píxeles. La secuencia registra mediante un movimiento progresivo de la cámara. Se trata de una escena plana registrada bajo un movimiento arbitrario de la cámara, que experimenta movimientos simultáneos de traslación, y rotación en diversos ejes.

La Figura 5.6 muestra el mosaico final obtenido para esta secuencia. En esta figura se observa cómo se ha conseguido un un la construcción de un mosaico con algunos detalles que saltan a la vista, por ejemplo centrando nuestra vista del centro a la derecha de la figura apreciamos algunas imprecisiones en la imagen, esto se debe a la imprecisión a del método RANSAC ya que estamos hablando que RANSAC trabaja a nivel píxel en la imagen y al empatar las imágenes se notan los bordes de las imágenes empatadas. En la figura 5.5 se aprecia el mosaico integrado mediante el método de RANSAC y el método de minimización GNLM con la función de error SSD-ARC, de esta forma se refina el resultado obtenido por la selección de puntos con el método RANSAC y mínimos cuadrados. Observando este mosaico se aprecia que tiene los errores al empatar las imágenes prácticamente desaparecen, no se notan estos errores por que el método de registro en tono de gris usa toda la información de la imagen para estimar la homografía y además de que lo hace a nivel subpixel.



Figura 5.4: Secuencia de imágenes del Posgrado de la FIE

Para demostrar la necesidad de la introducción del proceso adicional de registro en tono de gris, la Figura 5.6 muestra el mosaico que resulta de la integración de los 10 cuadros de la secuencia con un alineamiento basado exclusivamente en las posiciones de los puntos correspondientes suministradas por RANSAC y mínimos cuadrados.



Figura 5.5: Imagen Panorámica del Edificio del Posgrado de la FIE estimado con GNLM-SSD-ARC



Figura 5.6: Imagen Panorámica del Edificio del Posgrado de la FIE estimado con RANSAC y mínimos cuadrados

### 5.3.2. Mosaico de las Tarascas de Morelia

El mosaico de la figura 5.8 fue creado con dos imágenes que fueron tomadas por el autor de este trabajo con una cámara digital Kodak DC7630 de 6.1 megapíxeles, las dimensiones de estas imágenes son de  $640 \times 480$ , las cuales podemos ver en la figura 5.7. En la figura (a) de 5.8 en la parte superior derecha, se presentan tres acercamientos a esta imagen en donde se nota como al empatar las imágenes no coinciden correctamente, por ejemplo para el primero de los acercamientos se puede ver como el borde entre las imágenes es muy notorio y sobre todo en la parte derecha no empata con la parte izquierda, en la imagen del acercamiento del centro se aprecia como la esquina de la banqueta no se intercepta correctamente y por último se puede apreciar como los diques de la calle no siguen la misma línea de unión, esto debido a lo ya mencionado con RANSAC, trabaja a nivel píxel y sus errores son más notorios al una las imágenes. También podemos apreciar de la figura 5.7 que hay partes en una imagen que no existen en la otra, por ejemplo, en la imagen a la izquierda hay tres vehículos que en la imagen de la derecha ya no están, esto genera ruido a nuestra imágenes, el cual nuestro método lo resulte sin ningún problema.



Figura 5.7: Par de Imágenes de Dimensiones  $640 \times 480$  para Construir el Mosaico Tarascas de Morelia

En la parte inferior de la figura 5.8 parte (b) se realizan los mismo acercamientos pero al mosaico creado por el método de optimización GNLM con la función de error SSD-ARC y se puede apreciar como los errores al empatar las imágenes del mosaico estimado desaparecen en relación con los errores dados por RANSAC y mínimos cuadrados, se puede

observar también como las imágenes se acoplan perfectamente una sobre otra sin los detalles bruscos ya analizados.

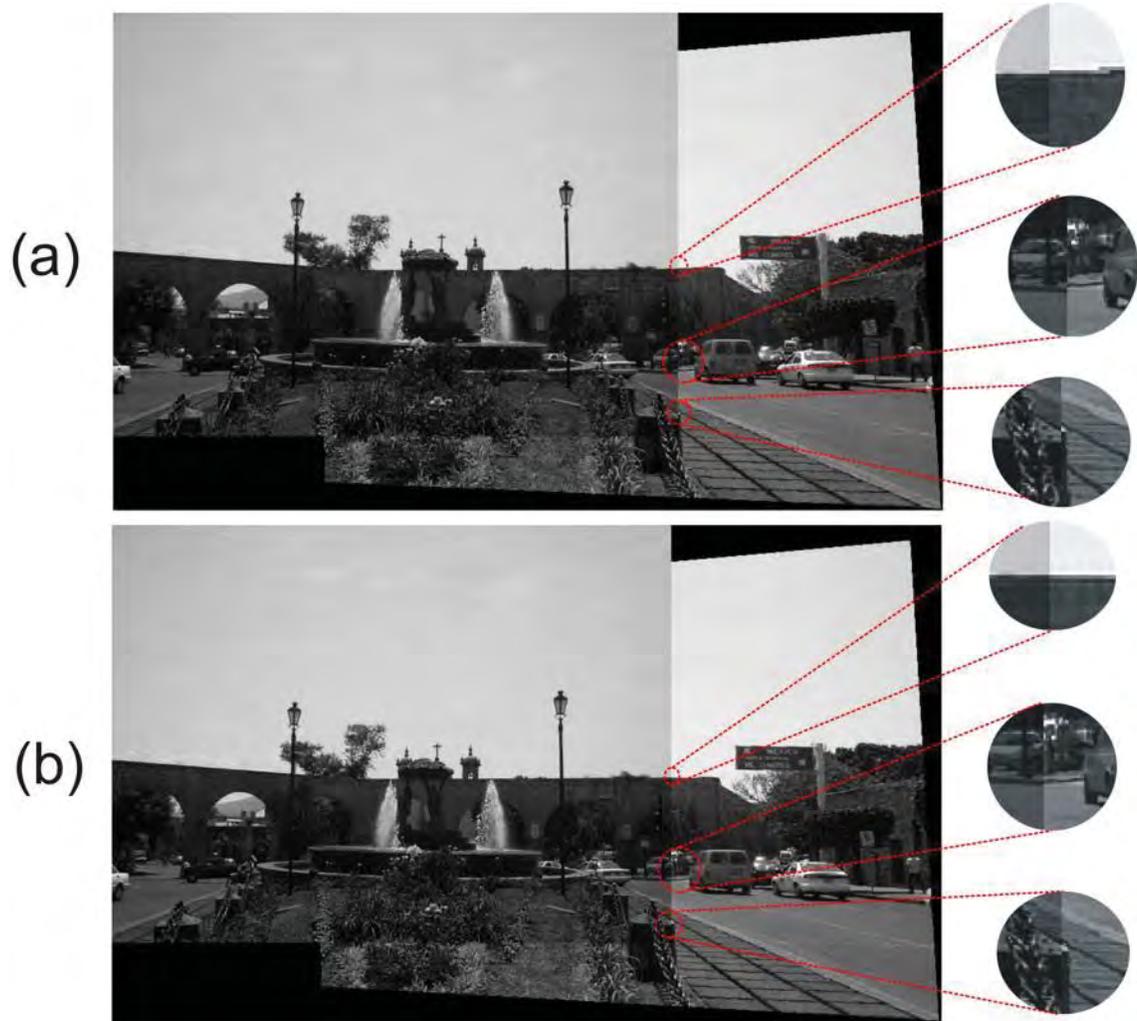


Figura 5.8: Comparando el Método RANSAC (a) con la Función de Optimización SSD-ARC (b) con Imágenes Reales (Tarascas de Morelia)

### 5.3.3. Mosaico con Imágenes Extraídas de Google Earth ©

La Figura 5.10 muestra el mosaico obtenido para la secuencia “Google Earth” extraídas del software Google Earth ©. Esta secuencia, está formada por 3 cuadros de tamaño  $1094 \times 711$  píxeles (La Figura 5.9 muestra los cuadros de esta secuencia). Se trata en este caso de una escena tomada por satélite, que es registrada mediante un movimiento traslación y

rotación. Por tanto se cumplen también ahora las condiciones teóricas de validez del modelo de registro homográfico.

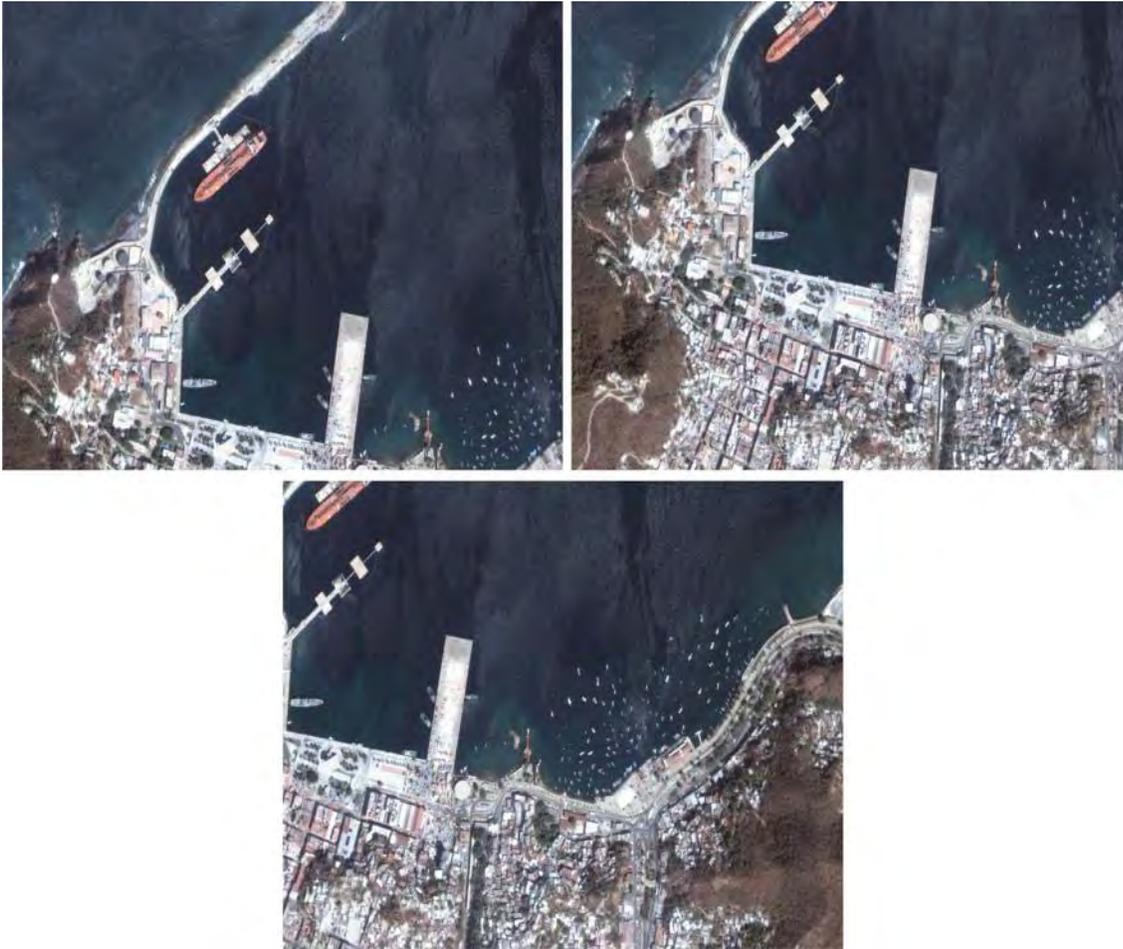


Figura 5.9: Imágenes Extraídas de Google Earth ©

La figura 5.9 muestra las tres imágenes tomadas de Google Earth © para construir el mosaico de imágenes, en estas se pueden apreciar partes de las imágenes que no aparecen en las otras, esto es ruido para el manejo de imágenes, este ruido es manejado sin ningún problema por nuestra aplicación ya que hacemos un registro robusto a datos atípicos, como se puede apreciar en la figura 5.10 que no se notan las uniones entre las imágenes.



Figura 5.10: Imagen Panorámica de las Imágenes Extraída de Google Earth ©



Figura 5.11: Secuencia de imágenes de la Fachada del Vaticano

#### 5.3.4. Mosaico de la Plaza de San Pedro

En la figura 5.11 se muestra la secuencia de imágenes utilizada para crear el mosaico panorámica del la fachada del vaticino en la figura 5.12 esta secuencia fue tomada por el autor de este trabajo con una cámara digital BENQ de 3 megapixeles con un tamaño de  $640 \times 480$ .

Algunos detalles que se pueden apreciar en la figura 5.11, son que en las imágenes superiores hay un elemento que no aparece en la imagen inferior de la secuencia, esto es en el balcón al centro de la imagen, existe un estandarte del cual se ve una parte, esto se maneja como ruido o datos atípicos al igual se tratan las partes de las imágenes que no aparecen en las tres, en la figura 5.12 se puede apreciar el mosaico construido y también que no aparece el estandarte que aparece en las dos imágenes de la figura 5.11, esto se debe a que la imagen inferior de 5.11 se sobreponer a las otras dos imágenes al crear el mosaico.



Figura 5.12: Imagen Panorámica de la Fachada del Vaticano

## 5.4. Otros Ejemplos

Las Figuras 5.13 y 5.14 muestran el resultado de la construcción de otros mosaicos obtenidos de la tesis [Paixao03] correspondientes en esta ocasión a escenas de un edificio, estas imágenes tienen las siguientes dimensiones  $640 \times 480$  y fueron obtenidas de la tesis de Paixao en [Paixao03], para trabajar con nuestra aplicación.



Figura 5.13: Imagen Panorámica de un Edificio Elaborada de Derecha a Izquierda

La figura 5.13 es un mosaico de imágenes creado de derecha izquierda eso quiere decir que se tomo como referencia la imagen de la derecha y al contrario el mosaico de la figura 5.14 que se tomo la imagen de la izquierda como referencia para construir el mosaico a partir de esta.



Figura 5.14: Imagen Panorámica de un Edificio Elaborada de Izquierda a Derecha

## Capítulo 6

# Conclusiones

En este capítulo se extraen las conclusiones acerca del trabajo realizado, descrito a lo largo del presente documento. Estas conclusiones se refieren tanto a aspectos generales de la labor desarrollada y de las materias tratadas, como a aspectos concretos de los algoritmos propuestos y su aplicación a los problemas que han sido objeto de estudio. Asimismo se proponen una serie de líneas de actuación que se consideran de interés para el futuro.

Se ha desarrollado una aplicación de construcción de mosaicos panorámicos basada en el seguimiento de conjuntos sucesivos de puntos característicos, que son seleccionados en determinados cuadros de la secuencia de imágenes. El algoritmo propuesto en el Capítulo 2, ha resultado eficaz para la aplicación desarrollada.

Los planteamientos multirresolución pretenden solucionar el problema del registro con grandes desplazamientos cuando no se tiene ninguna estimación inicial del desplazamiento o cuando la que se tiene es muy imprecisa. Las conclusiones generales que se extraen de la labor realizada pueden resumirse en los siguientes puntos:

- Como primera aproximación al problema general del movimiento de la cámara entre escenas se ha realizado una revisión bibliográfica de los diferentes planteamientos que han sido objeto de interés entre la comunidad investigadora en los últimos años. Este estudio, junto con la labor llevada a cabo respecto a la evaluación de técnicas y el desarrollo, implementación y experimentación de los diferentes algoritmos, proporciona una experiencia valiosa sobre la que fundamentar futuros trabajos en esta línea.

- La revisión bibliográfica demuestra que el tema del seguimiento visual de objetos ha suscitado un interés importante durante los últimos años y sigue motivando en la actualidad la actividad investigadora de una comunidad científica numerosa. A pesar de haber sido estudiado planteamientos muy diversos y debido a la complejidad del problema, parece evidente que su solución robusta en el caso general sigue abierta. Los algoritmos propuestos sólo son efectivos para aplicaciones concretas y estableciendo adecuadas restricciones a las posibilidades de movimiento de los objetos (o la cámara) y las características de la escena.

Para que cualquier aplicación práctica que involucra sistemas de seguimiento pueda ser resuelta satisfactoriamente debe atenderse de forma obligatoria a los muy diversos aspectos implicados: localización inicial en la imagen de los objetos de interés para la aplicación; selección de un conjunto de puntos característicos con propiedades adecuadas; definición de modelos que establezcan relaciones entre estos puntos de acuerdo a las características del objeto y su movimiento; implementación de mecanismos de estimación del desplazamiento sufrido por los puntos a lo largo de la secuencia; establecer procedimientos para determinar la nueva posición y orientación de la imagen a partir del desplazamiento observado para sus puntos.

Se consideran como aspectos relevantes algunos puntos citados a continuación. Respecto al algoritmo general de seguimiento cabe destacar:

- El método de selección de puntos característicos utilizado en este trabajo se basa en la selección de puntos con buenas propiedades que lo distingan, considerando la estimación de desplazamiento o búsqueda de correspondencia que hará uso de puntos seleccionados.

En relación con la aplicación de construcción de mosaicos, cabe concluir que:

- Se ha desarrollado con éxito esta aplicación de construcción de mosaicos panorámicos basado en registro de imágenes realizado en dos etapas, la primera realizando registro de imágenes basado en puntos característicos, en esta etapa se selecciona un conjunto de puntos característicos, la cual hace uso de las posiciones suministradas por los

puntos correspondientes y un segundo paso basado en registro en tono de gris que hace uso del contenido de la imagen.

- La introducción de técnicas complementarias de registro continuo, aunque aumenta el coste computacional, consigue una mejora importante de la calidad del mosaico. Esto ha hecho factible el uso de registro basado en puntos con un número reducido de puntos característicos y técnicas de localización de correspondencias que no requieren una precisión subpixel.
- Respecto a métodos de construcción de mosaicos basados exclusivamente en técnicas de registro en tono de gris, la introducción del registro de puntos ha conseguido eludir la falta de capacidad que los métodos basados en registro de tono de gris manifiestan en el manejo de grandes desplazamientos entre cuadros evitando recurrir a soluciones más costosas de uso frecuente como el registro piramidal. Las correspondencias entre puntos suministradas, permiten plantear un paso previo de registro que sirve como punto de partida muy aproximado para plantear un registro de tono de gris con plena resolución.
- Se ha implementado un método de selección de puntos que se basa en buenas propiedades que lo distinguen, totalmente autónomo y que permite definir el número de puntos a seleccionar, considerando la estimación de desplazamiento o búsqueda de correspondencia que hará uso de puntos seleccionados. De esta forma se ha conseguido aumentar la precisión del proceso de registro basado en correspondencias de los puntos característicos.
- Se ha implementado el método de registro de tono de gris optimizando una función de error la cual es insensible a datos atípicos, con la ayuda de la función de error propuesta la estimación de la homografía final es muy buena ya que tienen un buen comportamiento ante datos atípicos.

## 6.1. Trabajo futuro

Con relación a las propuestas de mejora de los algoritmos ya desarrollados se pueden apuntar algunas líneas de actuación muy concretas cuyo estudio se considera de interés. En cuanto al algoritmo general de seguimiento desarrollado, se propone:

- El estudio de la posibilidad de extensión del algoritmo al caso de modelado en 3D, para realizar la reconstrucción de panoramas en 3D.
- Una futura contribución a realizar será la elaboración de una aplicación con entorno gráfico para la construcción de mosaicos de imágenes completamente autónomos como mosaicos con ayuda de un operador en base a la selección de puntos de control.
- Se implementarán algoritmo de mejoramiento de la imagen final, como por ejemplo algoritmos de blending entre las imágenes.

# Referencias

- [Amil03] Amil, C. P. y Martínez, V. J. Notas: Registro de imágenes, January 2003.
- [Arias04] Arias, E. M. O. y Cruz, R. M. W. Arquitectura hardware para creación de mosaicos de video en tiempo real. *International Conference on Reconfigurable Computing and FPGAs.*, 2004.
- [Atienza04] Atienza, V. V. L. y Valiente, G. J. M. *Contribución al seguimiento visual basado en puntos característicos. Aplicación al seguimiento de caras y la construcción de mosaicos.* Tesis Doctoral, Universidad Politecnica de Valencia, Julio 2004.
- [Black96] Black, M. y Rangarajan, A. On the unification of line process, outliers rejection, and robust statistics with applications in early vision. *International Journal of Computer Vision 19*, págs. 57–92, 1996.
- [Brown92] Brown, L. G. A survey of image registration techniques. *ACM Computing Surveys*, págs. 325–376, 1992.
- [Calderón03] Calderón, F. y Marroquin, J. Un nuevo algoritmo para el calculo de flujo optico y su aplicación al registro de imágenes. *Computación y Sistemas 6*, págs. 213–226, 2003.
- [Calderón04] Calderón, F. y Romero, L. Non-parametric image registration as a way to obtain an accurate camera calibration. *In Springer: Advances in Artificial Intelligence*, págs. 584–591, 2004.

- [Calderón06] Calderón, F., Romero, L., y Flores, J. J. Ga-ssd-arc-nlm for parametric image registration. *En CIARP*, págs. 227–236. 2006.
- [Censi99] Censi, A., Fusiello, A., y Roberto, V. Image stabilization by features tracking. *Proceedings of the 9th Int. Conf. on Image Analysis and Processing*, 1999.
- [Chapra99] Chapra, S. C. y Canale, R. P. *Métodos numéricos para ingenieros*. McGraw-Hill, 1999.
- [Colombo99] Colombo, C. y Bimbo, A. D. Real-time head tracking from the deformation of eye contours using a piecewise affine cameraon. 1999.
- [Domingo04] Domingo, M. Vision por computador, ago. 2004. [Http://www.ing.puc.cl/dmery](http://www.ing.puc.cl/dmery).
- [Donald95] Donald, H. y Pauline, M. B. *Gráficas por computadora*. Prentice Hall, Hispanoamericana S.A., 1995.
- [Fischler81] Fischler, A. y Bolles, R. Random sample consensus: A paradigm for model fitting with analysis and automated cartography. *Comm. Assoc. Comp.*, págs. 381–395, 1981.
- [García01] García, R., Batlle, J., Cufí, X., y Amat, J. Positioning an underwater vehicle through image mosaicking. *Int. Conf. On Robotics and Automation*, 3:2779–2784, 2001.
- [Gee96] Gee, A. y Cipolla, R. Fast visual tracking by temporal consensus. *Image and Vision Computing*, págs. 105–114, 1996.
- [Gennery80] Gennery, D. B. *Modelling the Environment of an Exploring Vehicle by means of Stereo Visions*. Tesis Doctoral, Stanford University, Junio 1980.
- [Gennery92] Gennery, D. Visual tracking of known three-dimensional objects. *International Journal of Computer Vision*, 1992.

- [Gonzalez95] Gonzalez, W. R., R.C. *Digital Imagen Processing*. Addison-Wesley, 1995.
- [Hampel86] Hampel, F., Ronchetti, E., Rousseeuw, P., y Stahel, W. *Robust Statistics: The Approach Based in Influence Functions*. John Wiley and Sons, 1986.
- [Hartley04] Hartley, R. I. y Zisserman, A. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, 2<sup>a</sup> ed<sup>ón</sup>., 2004.
- [Huber03] Huber, P. J. *Robust Statistics*. Wiley Series in Probability and Statistics, 2003.
- [Kitchen82] Kitchen, L. y Rosenfeld, A. Grey-level corner detection. 1982.
- [Lane94] Lane, R. A. y Thacker, N. A. Stereo vision research: An algorithm survey. Technical report 94/16, University of Sheeld Electronic Systems Group, 1994.
- [Lowe92] Lowe, D. G. Robust model-based motion tracking through the integration of search and estimation. 1992.
- [López99] López, C. E. *Registro Automático de Imágenes de Bandas Espectrales Alejadass*. Tesis Doctoral, Universidad Complutense de Madrid, Agosto 1999.
- [Lucas81] Lucas, B. y Kanade, T. An iterative image registration technique with an application to stereo vision. *Proc. of the 7th Int. Joint Conf. on Artificial Intelligence*, 1981.
- [Marapane92] Marapane, B., Suresh y Trivedi, M., Mohan. Multiprimitive hierarchical (mph) stereo analysis. 1992.
- [McLachlan88] McLachlan, G. J. Mixture models. 1988.
- [Miravet98] Miravet, C., Santamaría, J., Coiras, E., Ureña, E. J., J., y Saraúa, A. Generación semi-automática de mosaicos. aplicación de técnicas de fusión de imágenes. *Teledetección 1998*, 1998.
- [Moravec80] Moravec, H. Robot rover visual navigation. *UMI Research Press,2*, 1980.

- [Nocedal99] Nocedal, J. y Wright, S. J. *Numerical Optimization*. Springer-Verlag New York, Inc, 1999.
- [Odone04] Odone, F. y Fusiello, A. Applications of 2d image registration, research memorandum rm/99/15, 2004.  
URL <http://www.sci.univr.it/fusiello/papers/RM-99-15.ps.gz>
- [Paixao03] Paixao, C. *Construcción de Mosaicos de Video*. Tesis Doctoral, Departamento de Engenharia de Electrónica e Telecomunicações e de Computadores, INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA, Octubre 2003.
- [Pajares03] Pajares, G., Cruz, J. M. d. l., Molina, J. M., Cuadrado, J., y A., L. *Imágenes Digitales Procesamiento práctico con Java*. Alfaomega-Ra-Ma, 2003.
- [Rehg91] Rehg, a. W. A., J.M. Visual tracking with deformation models. *Proc. IEEE Int. Conf. on Robotics and Automation*, págs. 844–850, 1991.
- [Romeny94] Romeny, B. T. H. *Geometry-Driven Diffusion in Computer Vision*. Kluwer Academic Publishers, 1994.
- [Shi94] Shi, J. y Tomasi, C. Good features to track. *Proceedings IEEE Conf. On Computer Vision and Pattern Recognition (CVPR94)*, 1994.
- [Shum98] Shum, H.-Y. y Szelinski, R. Construction and refinement of panoramic mosaics with global and local alignment. *Proc. 6th Conf. on Computer Vision (ICCV 98)*, págs. 953–958, 1998.
- [Soatto95] Soatto, M., Kosacka, S., y Igarashi, S. *An initiation to 3-D Vision From Images to Geometric Models*. Springer, 1995.
- [Stiefelhagen96] Stiefelhagen, R., J., Y., y Waibel, A. A model-based gaze tracking system. *Proceedings of IEEE International Joint Symposia on Intelligence and Systems*, págs. 304–310, 1996.

- [Szelinski94] Szelinski, R. Image mosaicing for tele-reality applications. *Proc. IEEE Workshop on Applications of Computer Vision (WACV 94)*, págs. 44–53, 1994.
- [Szeliski97] Szeliski, Y. H.-S., Richard. Creating full view panoramic image mosaics and enviroment maps. *Microsoft Research.*, 1997.
- [Tekalp95] Tekalp, M. *Digital Video Processing*. Prentice-Hall, 1995.
- [Tomasi98] Tomasi, T., Fusiello, A., y Roberto, V. Robust feature tracking in underwater video sequences. *Proc. of the OCEANS 98 IEEE/OS Conference*, págs. 46–50, 1998.
- [Trucco98] Trucco, R. y Verri, A. *Introductory techniques for 3-D Computer Vision*. Prentice Hall, 1998.
- [Weng93] Weng, J. J. 3-d motion analysis from image sequences using point correspondences. *Handbook of Pattern Recognition and Computer Vision*, págs. 395–441, 1993.
- [William92] William, H. P., Teukosky, S. A., Vetterling, W. T., y Flannery, B. P. *Numerical Recipes in C - The Art of Scientific Computing*. Cambridge University Press, 1992.
- [Zappala99] Zappala, A., Gee, A., y Taylo, M. Document mosaicing". *IVC, vol. 17*, págs. 589–595, 1999.