REGISTRO DE IMAGENES BIDIMENSIONALES A UN MODELO DE PUNTOS TRIDIMENSIONALES

TESIS

Que para obtener el grado de

MAESTRO EN CIENCIAS EN INGENIERÍA ELÉCTRICA

presenta

Garibaldi Pineda García

Doctor Félix Calderón Solorio

Director de Tesis

Universidad Michoacana de San Nicolás de Hidalgo

División de Estudios de Posgrado de la Facultad de Ingeniería Eléctrica

Agosto 2008

Resumen

La visión por computadora es un campo de las ciencias computacionales que en años recientes ha visto gran interés y desarrollo. Dentro de esta especialidad se encuentra el problema de registro de imágenes, el cual consiste en encontrar la transformación geométrica para alinear las imágenes de forma que se superpongan. El presente trabajo revisa, evalúa y compara distintas técnicas para el registro de imágenes en dos dimensiones a un modelo tridimensional. Lo anterior es debido a que se pretende tener un sistema automático para conocer la posición y orientación en tres dimensiones de un conjunto de fotografías bidimensionales respecto a un conjunto de puntos tridimensionales.

Se brinda una introducción al campo de registro de imágenes en su forma general y después se adentra en el mundo del registro 2D/3D. Se utiliza proyección en perspectiva débil pues esta se aproxima a la perspectiva completa cuando el objeto está a gran distancia de la cámara comparado con su tamaño. El primer método revisado, *SoftPOSIT*, plantea resolver tanto las correspondencias como la transformación proyectiva simultáneamente utilizando un algoritmo basado en recocido simulado. El segundo algoritmo es una adaptación del ya probado *RANSAC*, la novedad se encuentra en que la aleatoriedad del muestreo se ve reemplazada por elecciones ordenadas de acuerdo a la probabilidad de apareamiento. Finalmente se revisa un procedimiento basado en fusión robusta de datos, el cual reduce a cada paso la cantidad de hipótesis a revisar dependiendo de criterios probabilísticos; al terminar el acotamiento de datos se realiza una búsqueda/comprobación mediante *RANSAC*.

A partir de las pruebas realizadas se observó que el algoritmo basado en *RANSAC* se comporta de manera más estable y robusta. El algoritmo que utiliza fusión robusta de información presenta la relación entre la incertidumbre de proyección y la incertidumbre de la ubicación de la cámara. El algoritmo basado en fusión robusta presenta resultados positivos, similares a los obtenidos mediante *RANSAC*. *SoftPOSIT* da resultados inestables y es dependiente de una estimación inicial de la matriz de ubicación de la cámara que pueda guiar el procedimiento a la solución real.

Abstract

Computer vision has recently been the focus of much research and development teams. Image registration, which is the procedure to overlay a two or more images, is studied by computer vision. This paper reviews, evaluates and compares various techniques developed to solve the 2D-3D registration problem. This investigation's motivation was a plan to put together a fully automatic system to obtain the position and orientation in a three-dimensional space of a set of 2D photographies and a known set of 3D points.

A brief introduction to both computer vision and registration as a whole is given. A weak perspective projection was chosen due to the fact that it can closely approximate full perspective, given that the distance from the camera to the object is much larger than the object's size. *SoftPOSIT*, the first reviewed methodology, attempts to solve both pose and correspondences simultaneously by minimization of an energy function in a simulated annealing procedure.An adaptation to the well proven *RANSAC* algorithm was analyzed as a second proposal, the main difference is that random sampling is no longer carried on, instead a probability ordered election is carried on to ensure rapid convergence. A recognition/registration algorithm using a robust data fusion scheme was lastly reviewed, this methodology reduces the data pool by several probabilistic criteria. With the reduced dataset a *RANSAC*-like search is performed in order to compute both real extrinsic camera parameters and final correspondence.

Results from the set of tests made show that the *RANSAC* based algorithm behaves in a robust and stable manner, more so than the other algorithms evaluated. A relationship between pose and projection uncertainties is introduced as a key component of the robust fusion based algorithm, this procedure also behaves in a robust way. *SoftPOSIT* renders unstalbe results which depend mostly on an initial pose matrix that leads the procedure towards convergence.

Contenido

Lista de Figuras

Lista de Tablas

Lista de Algoritmos

Lista de Símbolos

3D	En tres dimensiones, tridimensional.
2D	En dos dimensiones, bidimensional.
$\mathbf{P}$	Punto en tres dimensiones.
$\mathbf{q}$	Proyección en dos dimensiones de un punto tridimensional.
$x,\ y,\ z$	Coordenadas de un punto en tres dimensiones.
$u,\ v$	Coordenadas de un punto en dos dimensiones.
$\boldsymbol{R}$	Matriz de rotación.
$\mathbf{T}$	Vector de traslación, $\mathbf{T} = [T_x, T_y, T_z]^\top$.
f	Distancia focal de la cámara.
s	Escalamiento en perspectiva débil, $s = f/T_z$.
pp	Punto Principal.
$\mathbf{C}$	Centro de proyección, origen de coordenadas de la cámara.
$\mathbf{Cz}$	Dirección del eje óptico.
$\boldsymbol{K}$	Matriz de parámetros internos.
α_u, α_v	Distorsión por el tamaño horizontal y vertical de cada pixel, incluye la distancia focal.
$u_0,\ v_0$	Coordenadas del punto principal.
s_k	Indicador del ángulo entre los pixeles horizontales y verticales.
SVD	Descomposición en Valores Singulares.
E	Función de error.
a_{ij}	Coeficiente de asignamiento entre el punto 2D i y el punto 3D j.
β_T	Variable de control (temperatura) del recocido simulado.

ι Radio de la esfera de observabilidad.

α, β, γ Pseudo-coordenadas para proyección vía algorítmo de Alter.

H_1, H_2 Parámetros para proyección vía algorítmo de Alter.

Ψ_i Vectores de incertidumbre de proyección.

Ξ_c Matriz de covarianza, caracteriza la incertidumbre de obtención de información.

ϕ Fuente de información.

Φ Observación de la fuente de información ϕ.

δ Función de densidad.

Capítulo 1

Introducción

El problema del registro de imágenes se refiere a alinear imágenes geométricamente. Esto puede ser conseguido tras encontrar la transformación que relaciona los dos conjuntos, o bien las posibles correspondencias entre datos. La tarea de registro puede clasificarse por la naturaleza de los datos de entrada. Si se registra una imagen en dos dimensiones (2D) con otra de la misma naturaleza, se le conoce como registro 2D-2D; cuando se trata de registrar imágenes bidimensionales con una imagen tridimensional (3D) se le llama registro 2D-3D y al proceso de registrar puntos en tres dimensiones se le denomina registro 3D-3D.

Otra forma de clasificar el registro es por la naturaleza de la transformación entre las imágenes cuya complejidad del cálculo se incrementa a medida que se agregan grados de libertad. Las transformaciones rígidas son las más simples y están compuestas de translaciones y rotaciones sin deformación de los cuerpos. Las transformaciones afines mapean líneas paralelas a líneas paralelas. Las transformaciones proyectivas mapean líneas a líneas y las transformaciones elásticas aparean líneas a curvas [Hartley03].

Existen gran cantidad de algoritmos utilizados para resolver este problema, desde los completamente automáticos donde solamente es necesario introducir las imagenes a registrar; hasta enteramente interactivos, en los cuales se tiene la guía del usuario para tomar decisiones en cada paso del proceso de registro. [Shapiro01].

En este documento se aborda el problema del registro de imagenes bidimensionales a modelo compuesto por puntos en tres dimensiones, es decir, encontrar la transformación

proyectiva de un objeto. Para este propósito se revisan tres algoritmos que abordan distintas aproximaciones para resolver el problema sin conocer las relaciones entre los puntos del modelo en tres dimensiones y los puntos extraídos de la imagen. Se analizan además un par de metodologías para establecer la proyección con conocimiento a priori de las correspondencias entre puntos del modelo tridimensional y los obtenidos de la imagen.

1.1. Objetivo

El presente trabajo tiene como objetivo comprender, evaluar y comparar algunos algoritmos del estado del arte para resolver el problema del registro automático de imágenes bidimensionales a un modelo compuesto por puntos en tres dimensiones. Dadas fotografías adquiridas mediante una cámara digital comercial y conjunto de puntos en tres dimensiones que modelan las características físicas de un objeto real, se busca la transformación proyectiva entre las fotografías y el conjunto de puntos tridimensionales con el objetivo de conocer la ubicación de la cámara respecto al modelo de forma automática. La figura 1.1 es un ejemplo del objetivo del trabajo, se tienen las fotografías insertadas en un espacio tridimensional con el modelo 3D al centro. Para lograr esta imagen se calcularon las ubicaciones de las distintas fotografías respecto al objeto, mostrado al centro de la figura. Cada fotografía contiene al objeto, un cubo de bloques de Lego, tomados desde distintas ubicaciones, es por esto que se aprecian al rededor del cubo modelado en tres dimensiones.

1.2. Antecedentes

El problema del registro de modelo tridimensional a imagen en dos dimensiones generalmente se resuelve utilizando dos estrategias, la primera consiste en encontrar las correspondencias entre puntos detectados en la imagen y los que componen al modelo; mientras que la segunda técnica implica buscar la matriz de proyección que mapea los puntos en tres dimensiones pertenecientes al modelo a los puntos detectados en la imagen bidimensional. Si se conocen las correspondencias entre características, es decir, puntos detectados en la imagen, y puntos del modelo en tres dimensiones, la forma de estimar la proyección depende del número de pares. En este trabajo se aborda únicamente la perspectiva débil, un modelo de proyección que mapea puntos en tres dimensiones a su imagen bidimensional de forma tal

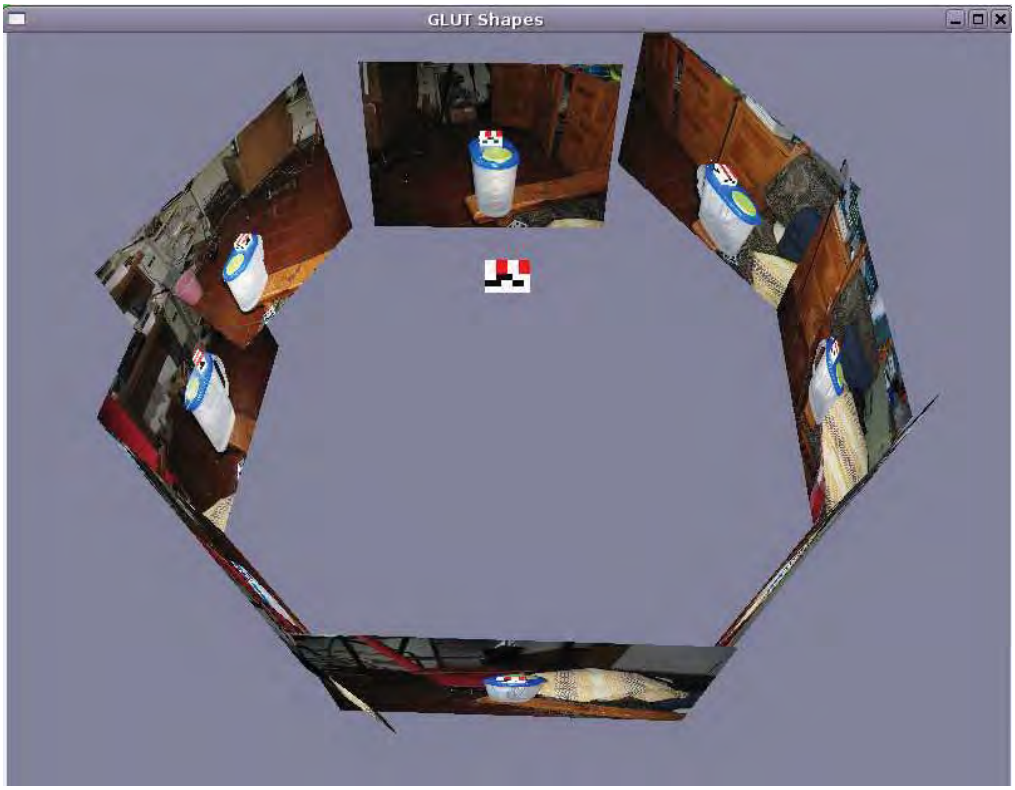


Figura 1.1: Fotografías insertadas en un espacio tridimensional

que la geometría tridimensional no se distorsiona salvo por un escalamiento originado por la distancia de la cámara al objeto. Para conocer la posición y orientación, también conocida como pose o ubicación, del objeto respecto a la cámara se pueden resolver ecuaciones o bien utilizar procesos de minimización [Dementhon95]. Por otro lado si se conoce previamente la ubicación, estimar las correspondencias es relativamente sencillo y solo hace falta verificar la validez de la ubicación, un análisis de esto es presentado en [Grimson91].

El caso más interesante del registro de imagenes es cuando se tiene nada o poco conocimiento previo de correspondencias y/o ubicación. Existen distintas aproximaciones para solucionar la falta de conocimiento de ambas por separado, las formas clásicas son minimizando funciones de error o utilizando estrategias de hipótesis-prueba como el algoritmo de búsqueda robusta desarrollado por Martin A. Fischler y Robert C. Bolles conocido como RANSAC [Fischler81].

En [Shimshoni00a] se describe un método para reconocimiento de objetos tridimensionales basándose en el efecto de pico de probabilidad y verificación de hipótesis de apareamiento. En [Clarkson01] se minimiza una medida de similaridad llamada "foto-consistencia" para registrar imágenes de rostros a superficies tridimensionales, una de las principales limitantes es que se debe tener conocimiento previo de las transformaciones entre imágenes. Con líneas detectadas en la imagen y los ejes conocidos del modelo 3D, en [David03] se logra el registro 2D-3D buscando una matriz de proyección que permita la convergencia de un ciclo de recocido simulado que minimiza una función de error basada en la proyección de las líneas del modelo en perspectiva débil.

En [Dorfler04] se estima la ubicación de objetos mediante *EigenTracking*, además se utilizan mascaras que compensan los distintos contornos que puede tener un objeto al ser rotado, por lo que se cuenta con una "*búsqueda*" jerárquica de espacios normales.

Utilizando refinamiento del espacio de búsqueda, en [Chen04b] se plantea un algoritmo que toma en cuenta las probabilidades de apareamiento entre características para generar hipótesis de poses. Luego se calcula la probabilidad de que estas sean correctas y finalmente lo verifica mediante un procedimiento tipo RANSAC.

1.3. Descripción de la Tesis

En el Capítulo segundo se da una introducción a los conceptos de visión por computadora. Se tratan los componentes de un sistema de visión, algunos modelos matemáticos de cámaras y la relación que guardan entre sí, y se describe también el proceso de registro de imágenes, sus tipos y variaciones. El tercer Capítulo complementa la introducción al registro de imágenes y presenta las posibles transformaciones que serán tomadas en cuenta para el problema de registro de modelo a imagen. Es en ese apartado que se realiza el análisis y comparación del funcionamiento de dos algoritmos con apareamiento manual, *estimación de Alter* y *POSIT*. Posteriormente se analizan tres algoritmos completamente automáticos para resolver el problema de registro 2D-3D, *SoftPOSIT*, *gRANSAC* y un tercero basado en *fusión robusta de información*. *SoftPOSIT* plantea resolver tanto la pose como las co-

rrespondencias simultaneamente, por lo que solamente es necesario proporcionarle puntos extraidos de la imagen, puntos del conjunto tridimensional y la distancia focal f de la cámara al momento de tomar la fotografía.

gRANSAC es una variación del algorítmo *RANSAC* cuya caracteristica principal es ser robusto ante la presencia de ruido y oclusión. Esta modificación introduce el concepto de muestreo probabilistico, que da prioridad a hipótesis con alta probabilidad. Las hipótesis de proyección son generadas mediante pares de triadas de puntos 2D-3D y la estimación de proyección de Alter. La probabilidad de apareamiento se calcula con el modelo probabilístico de la esfera de observabilidad.

El algorítmo basado en *fusión robusta de información* utiliza el modelo probabilístico de la esfera de observabilidad para ordenar hipótesis de apareamiento de triadas. Con algunas de estas hipótesis se proyectan un número de puntos tridimensionales pertenecientes a la envolvente convexa del conjunto 3D. Se considera que la incertidumbre de proyección es proporcional a la incertidumbre de la pose, bajo este precepto se utiliza el algoritmo *Mean Shift* para formar cúmulos, de forma robusta, de hipótesis con mayor probabilidad. Dado que se proyectan n puntos 3D se reduce el número de hipótesis al intersectar los cúmulos correspondientes a cada punto. Finalmente las hipótesis se introducen a un algorítmo tipo *RANSAC* para comprobar su validéz. Los parámetros externos pueden ser calculados posteriormente mediante *POSIT*.

Se realizaron experimentos con cada uno de estos métodos y el planteamiento experimental, los resultados y comparativas se encuentran reflejados en el Capítulo cuarto. Finalmente el quinto Capítulo recoge las conclusiones a las que se llegaron mediante la experimentación, así como ideas sobre posibles cambios a los algoritmos y el futuro de esta rama de la visión por computadora.

Capítulo 2

Visión Artificial y Registro

2.1. Visión Artificial

La visión es la asociación de información de forma, color y movimiento con imágenes percibidas mediante un sensor, esto con el objetivo de comprender el estado físico del medio ambiente. Los componentes comunes, ilustrados en la figura 2.1, a todos los sistemas de visión [Jahne00] son comentados en la siguiente enumeración

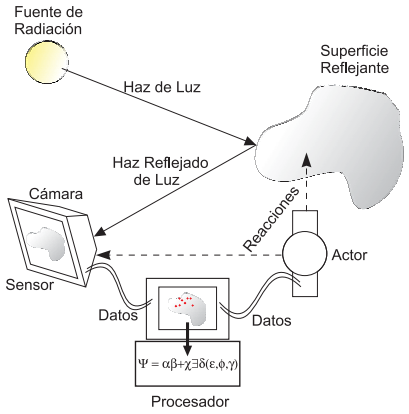


Figura 2.1: Componentes de un sistema de visión artificial típico

1. Fuente de radiación. Emite la radiación de algún tipo, comúnmente electromagnética, gracias a ello se tiene la posibilidad de observar los objetos que no la irradian pero si la reflejan.

2. Cámara. Su función es la de capturar la radiación emitida por la fuente y reflejada por los objetos. Puede ser desde un dispositivo sencillo como una cámara obscura hasta uno sofisticado como un tomógrafo de rayos X.

3. Sensor. El sensor convierte el flujo electromagnético en una señal adecuada para procesamiento futuro. Dependiendo de la aplicación es conveniente utilizar una línea o matriz de sensores.

4. Procesador. Se encarga de procesar la información entrante, esto incluye extraer características importantes para interpretar la escena o conocer propiedades de los objetos en ella. Generalmente existe un sistema de memoria que distingue entre imágenes relevantes y las almacena, ademas de desechar las imágenes sin importancia.

5. Actor. Finalmente se tiene al actor, este reacciona al estimulo visual, por ejemplo esquivar una pelota, guiar un automóvil, vigilar los movimientos de un objeto, etc.

La visión por computadora es la ciencia que se encarga de estudiar máquinas que ven y su meta principal es la de conocer información de un escenario representado en imágenes. Al igual que en la visión natural el inicio del flujo de información es el sensor luminoso [Shapiro01]. En años recientes la fotografía digital ha reducido sus costos y se ha transformado en la vía de captura por excelencia para la visión artificial, esto porque las fotografías pueden almacenarse y recuperarse en forma electrónica, lo que las hace ideales para el procesamiento con computadoras digitales [Zitova03]. Una imagen digital está compuesta por una matriz de pixeles (acrónimo de elementos de imagen, por sus siglas en inglés) y cada uno de ellos tiene un valor discreto que guarda la información de color, profundidad, luminancia, etc. capturada en esa posición.

2.2. Proceso de formación de la imagen

En este trabajo se utiliza el término imagen como sinónimo de fotografía ya que solo se toman en cuenta las imágenes obtenidas mediante cámaras sensibles a la luz visible por el ojo humano. Hacía el siglo XVI se inventa la cámara oscura y los artistas de aquella época comenzaron a retratar la realidad [Shapiro01]. La figura 2.2 muestra una cámara oscura, que sirve para ilustrar el funcionamiento básico de una cámara. Se compone por un cuarto oscuro con un orificio por donde entran los haces de luz (*centro de proyección*), la

pared donde se encuentra el orificio se conoce como *plano principal*. Dentro de la cámara está un plano donde se forma la imagen, este conocido como *plano de la imagen* y es paralelo al plano principal. La distancia del plano principal al plano de la imagen se conoce como distancia focal denotada por f.

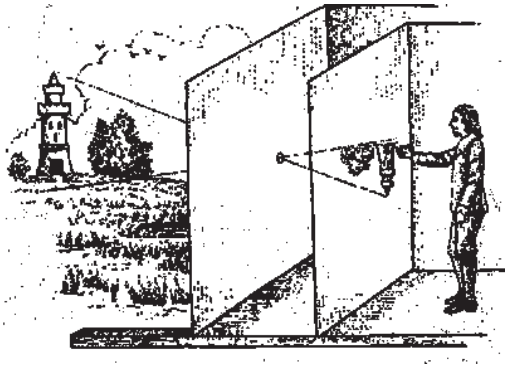


Figura 2.2: Cámara Oscura, Athanasius Kircher, 1646

Al analizar escenas en tres dimensiones se hace necesario establecer sistemas de referencias, en general se definen cinco sistemas de referencias coordenadas correspondientes a los componentes de un sistema de visión. La figura 2.3 ilustra los sistemas coordenados, se tienen uno para el mundo, otro para cada objeto, uno para la cámara y dos para la imagen formada, uno en coordenadas con números reales y el otro con coordenadas de números enteros. A continuación se enlistan descripciones breves de los sistemas de referencia .

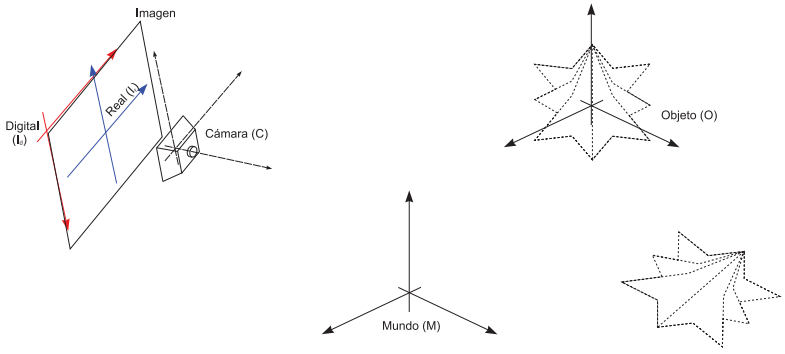


Figura 2.3: Sistemas de Coordenadas

1. Sistema de Referencia del Mundo (M). El primer sistema de referencia es el que da origen al mundo donde se sitúa la escena, sirve como la base para relacionar todos los objetos situados en el mundo.

2. Sistema de Referencia del Objeto (O). Este sistema coordenado es necesario porque en general cada objeto se modela por separado y mediante las transformaciones respecto a la referencia del mundo (M), se pueden situar de diferentes maneras en la escena.

3. Sistema de Referencia de la Cámara (C). Es un sistema de referencia tridimensional para "anclar" la imagen, es el punto desde donde la cámara tiene visibilidad de la escena. Uno de sus ejes se denomina eje óptico que corresponde a la dirección de observabilidad de la cámara.

4. Sistemas de Referencia de la Imagen (I_r, I_d). El primero es un sistema bidimensional con números reales para la fotografía *real*, se sitúa a f unidades del origen de coordenadas (C) sobre el eje óptico (C_z). El segundo es un sistema bidimensional con coordenadas enteras, trasladadas por congruencia con el almacenamiento digital. De esta relación se puede decir que la imagen digital es una versión de la fotografía real.

2.3. Modelo de Cámara

Se conoce como modelo de cámara a la expresión matemática que permite calcular la proyección de una escena tridimensional en un plano. Un modelo de cámara representa la construcción física de una cámara real de forma matemática para poder ser utilizado en cálculos. Es necesario para poder estudiar el fenómeno que ocurre al adquirir una imagen y sus relación con el objeto fotografiado [Hartley03, Forsyth03]. Existen diversas expresiones para modelar un dispositivo fotográfico, desde los que incluyen cada detalle o error introducido a la proyección por la construcción y óptica de la cámara, hasta los que son puramente ideales. Un modelo de camara se compone por distintos parámetros, y pueden clasificarse en externos e internos.[Shapiro01]

2.3.1. Parámetros Internos

Los parámetros internos obedecen a la forma en que la cámara fué construida y a la óptica al instante de adquirir la imagen. Algunos, como la distancia focal y el radio

de aspecto, pueden ser modificados mediante elementos diseñados exclusivamente para este cometido. La figura 2.4 ilustra los parámetros internos y se introduce el plano de imagen adelantado, una construcción matemática que facilita el modelado de la cámara. El plano de imagen real es el que físicamente se encuentra en la cámara, este fabrica las imágenes invertidas, es por eso que se hace uso del plano de imagen adelantado, pues en este no se presenta la inversión de las imagenes generadas. A continuación se presenta una enumeración de los parámetros internos, así como distorsiones originadas por los lentes y sensores digitales.

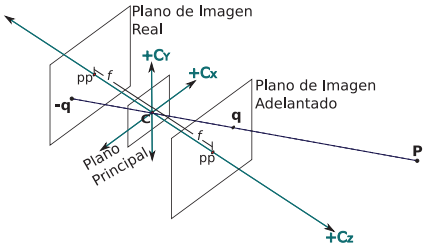


Figura 2.4: Parámetros Internos

1. Punto Principal (pp). Este se encuentra localizado en la intersección del plano de la imagen y el eje óptico, se considera el origen del sistema de coordenadas reales de la imagen.

2. Distorsión por Tamaño del Pixel. Se debe a la diferencia del tamaño horizontal y vertical de cada píxel en el sensor.

3. Distorsión por Relación de Aspecto. Se produce porque la cantidad de píxeles a lo largo del sensor es distinta a la que se tiene a lo alto.

4. Distancia Focal (f). Es la distancia del centro óptico al plano de la imagen.

5. Distorsión por Radial. Es provocada por la geometría del lente, ya que la luz se ve refractada al pasar de un medio a otro con distinta densidad; genera una distorsión

radial, por lo que se requiere un factor que compense dicha anomalía en la imagen. La figura 2.5 ilustra el efecto de esta distorsión la parte superior es una cuadrícula tal y como es, la parte inferior muestra un tipo de distorsión conocida como distorsión de cañon. Esta distorsión tiene como efecto que las líneas se vean transformadas en curvas lo que dificulta las mediciones y la veracidad de la extracción de puntos significativos de la fotografía. El lente está presente en la cámara para enfocar los haces de luz en el plano de la imagen, la distorsión por lentes se da cuando el lente (y los dispositivos auxiliares) no es capáz de realizar una proyección rectilinea.

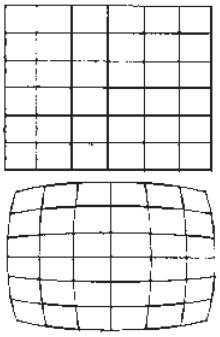


Figura 2.5: Distorsión por Lentes

2.3.2. Parámetros Externos

Una imagen también depende de la posición y orientación de la cámara en el instante de adquirirla, estos son conocidos como parámetros externos o pose de una cámara, estos se ven representados en la figura 2.6

1. Traslación. Es el vector que expresa la posición del sistema de coordenadas de la cámara respecto al del mundo.

2. Rotación. Es la orientación que guarda el sistema de coordenadas de la cámara respecto al del mundo y normalmente se expresa como una matriz de rotación $\boldsymbol{R}$ ejem-

plificada en la ecuación 2.1.

$$\boldsymbol{R} = \begin{bmatrix} \mathbf{R}_1^T \\ \mathbf{R}_2^T \\ \mathbf{R}_3^T \end{bmatrix} = \begin{bmatrix} R_{1x} & R_{1y} & R_{1z} \\ R_{2x} & R_{2y} & R_{2z} \\ R_{3x} & R_{3y} & R_{3z} \end{bmatrix} \tag{2.1}$$

donde $\mathbf{R}_1$, $\mathbf{R}_2$ y $\mathbf{R}_3$ son conocidos como vectores de rotación y son ortonormales entre sí, guardando la condición de que la matriz de rotación sea ortogonal. Estos vectores también representan las coordenadas de los vectores unitarios paralelos a los ejes coordenados rotados en el sistema de referencia actual.

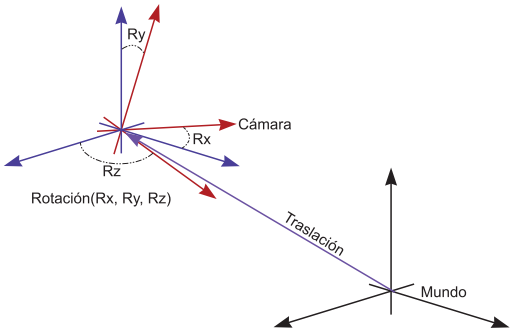


Figura 2.6: Parámetros Externos

En este trabajo el origen de coordenadas del modelo y del mundo se igualan, es decir, existe una transformación unitaria entre dichos sistemas de referencia. Dado que los objetos en el mundo están en un sistema de coordenadas arbitrario y la cámara está en el propio, es necesario unificar todas las referencias poder realizar la proyección. Generalmente se transforman los puntos tridimensionales al sistema de coordenadas de la cámara, esta acción es representada por la ecuación 2.2.

$$\mathbf{P}_c = \begin{bmatrix} x_c \\ y_c \\ z_c \\ w_c \end{bmatrix} = [\boldsymbol{R}|\mathbf{T}]\,\mathbf{P}_m = \begin{bmatrix} R_{1x} & R_{1y} & R_{1z} & T_x \\ R_{2x} & R_{2y} & R_{2z} & T_y \\ R_{3x} & R_{3y} & R_{3z} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ z_m \\ 1 \end{bmatrix} \tag{2.2}$$

donde $\mathbf{P}_c$ es el punto tridimensional en coordenadas de la cámara, $[R|\mathbf{T}]$ es la transforma-

ción entre el sistema de coordenadas del mundo y el sistema de coordenadas de la cámara, y $\mathbf{P}_m$ es el punto en tres dimensiones en coordenadas del mundo. Una vez que los puntos están en el mismo sistema de coordenadas es posible realizar la proyección (se usa el modelo de agujero de alfiler solo como ejemplo). Las ecuaciones de proyección son por su naturaleza no lineales, por lo que se utilizan coordenadas homogéneas para linealizar dicho proceso.

$$\begin{bmatrix} u\prime \\ v\prime \\ w \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_c \\ y_c \\ z_c \\ w_c \end{bmatrix} \tag{2.3}$$

en la ecuación 2.3, f es la distancia focal de la cámara, y $u\prime$, $v\prime$, y w son la proyección del punto $[x_c,\, y_c,\, z_c,\, w_c]^\top$ en coordenadas homogéneas. Finalmente se normalizan las coordenadas homogéneas para poder trabajar con geometría euclidiana (ecuaciones 2.4).

$$u = \frac{u\prime}{w} = \frac{f x_c}{z_c} \qquad v = \frac{v\prime}{w} = \frac{f y_c}{z_c} \tag{2.4}$$

Estado de Coordenadas	Transformación	Matemáticamente
Coordenadas 3D Originales		$x_m\ y_m\ z_m$
	Cambio a Sistema de Coordenadas de la Cámara	$\begin{bmatrix} R_{1x} & R_{1y} & R_{1z} & T_x \\ R_{2x} & R_{2y} & R_{2z} & T_y \\ R_{3x} & R_{3y} & R_{3z} & T_z \end{bmatrix}$
Coordenadas 3D Cámara	$\downarrow$	$x_c\ y_c\ z_c$
	Proyección	$\begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix}$
Coordenadas 2D		$u\ v$

Tabla 2.1: Resumen de Formación de Imagen

2.3.3. Modelo de cámara de agujero de alfiler

El modelo de cámara de agujero de alfiler es el más simple que existe. En este se considera una cámara ideal, sin distorsiones, sin lentes. Tiene un agujero lo suficientemente pequeño para que la luz sature el interior de la cámara y este es conocido como centro óptico y es la única via de entrada para la luz que termina en el plano principal, donde se forma la imagen, la figura 2.7 muestra un bosquejo de una cámara de agujero de alfiler.

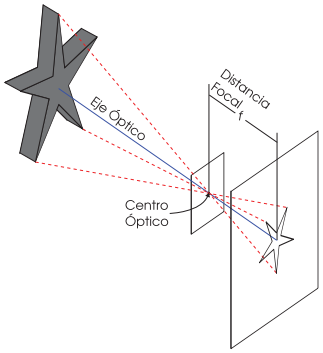


Figura 2.7: Proyección en perspectiva completa

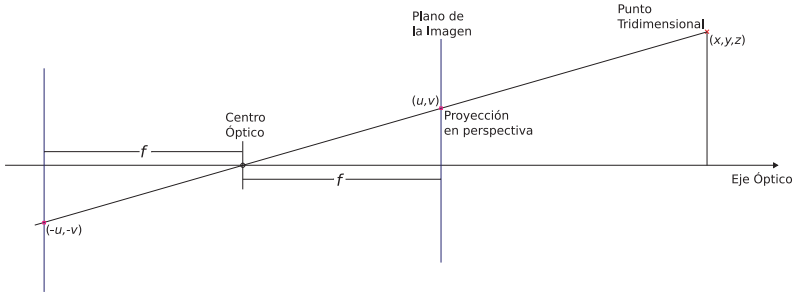


Figura 2.8: Vista lateral de la proyección en perspectiva completa

Para facilitar las ecuaciones se introduce una construcción matemática que traslada el plano de la imagen dos veces la distancia focal ($2f$) sobre el eje óptico hacia el objeto, dando esto como resultado que la imagen formada no esté invertida. La figura 2.8 ilustra lo recien descrito.

$$\begin{bmatrix} u\prime \\ v\prime \\ w \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R_{1x} & R_{1y} & R_{1z} & T_x \\ R_{2x} & R_{2y} & R_{2z} & T_y \\ R_{3x} & R_{3y} & R_{3z} & T_z \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ z_m \\ 1 \end{bmatrix} \tag{2.5}$$

finalmente se tienen las expresiones 2.6 que son las ecuaciones de proyección con el plano de imagen localizado frente a la cámara con un modelo de cámara de agujero de alfiler.

$$u = \frac{u\prime}{w} = \frac{fx_c}{z_c} \qquad v = \frac{v\prime}{w} = \frac{fy_c}{z_c} \tag{2.6}$$

2.3.4. Modelo de cámara de perspectiva débil

Los algoritmos detallados en este documento tienen como base el modelo de perspectiva débil, esto porque este modelo es una buena aproximación a la perspectiva completa cuando el tamaño del objeto es pequeño comparado con la distancia de este a la cámara [Shapiro01]. Este modelo puede considerarse como una proyección ortogonal combinada con un escalamiento y en algunas circunstancias se aproxima en gran medida a la perspectiva completa [Alter92]. La figura 2.9 ilustra la proyección en perspectiva débil utilizando el plano de imagen adelantado, se tienen puntos en tres dimensiones que se proyectan ortogonalmente en un plano normal al eje óptico. Posteriormente se escala dicha proyección de acuerdo a la distancia focal y la distancia del objeto a la cámara.

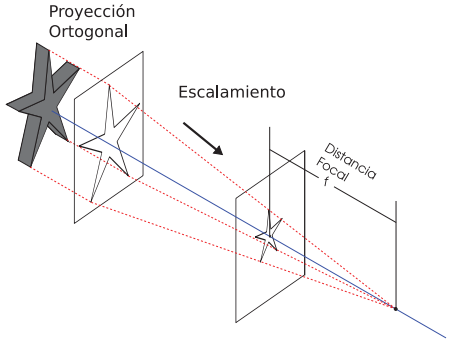


Figura 2.9: Proyección en perspectiva débil

Las siguientes ecuaciones modelan la proyección en perspectiva débil, las operaciones representan el cambio de coordenadas del sistema de referencia del mundo al de la

cámara, la proyección ortogonal y el escalamiento. Tanto la proyección ortogonal como el escalamiento se modelan en una sola matriz.

$$\begin{bmatrix} u \\ v \end{bmatrix} = \boldsymbol{K}\,[\boldsymbol{R}|\mathbf{T}] \begin{bmatrix} x_m \\ y_m \\ z_m \\ 1 \end{bmatrix} = \begin{bmatrix} s & 0 & 0 \\ 0 & s & 0 \end{bmatrix} \begin{bmatrix} R_{1x} & R_{1y} & R_{1z} & T_x \\ R_{2x} & R_{2y} & R_{2z} & T_y \\ R_{3x} & R_{3y} & R_{3z} & T_z \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ z_m \\ 1 \end{bmatrix} \tag{2.7}$$

o bien, en forma compacta

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} sR_1^T & sT_x \\ sR_2^T & sT_y \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ z_m \\ 1 \end{bmatrix} \tag{2.8}$$

El escalamiento, $s = f/T_z$, es inversamente proporcional a la distancia del objeto al origen de coordenadas de la cámara sobre el eje óptico. En pruebas realizadas se encontró que el error introducido al utilizar perspectiva débil es insignificante cuando la razon de la distancia de la cámara al objeto al tamaño del objeto es de 6 a 10. La figura 2.10 muestra el error generado al utilizar perspectiva débil en comparación con la perspectiva completa.

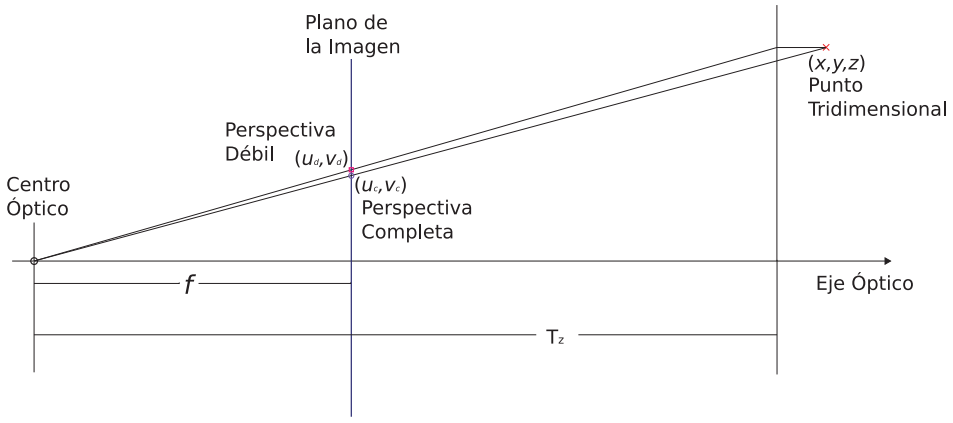


Figura 2.10: Comparación proyección en perspectiva débil - completa

A medida que se aleja un objeto de la cámara el error entre la proyección en

perspectiva completa y perspectiva débil se reduce. En la siguiente sección se trata la razón de la posiblidad de aproximar los tipos de perspectivas recién tratadas.

2.3.5. De perspectiva completa a débil

Sea $\mathbf{P} = [x\ y\ z]^T$ un punto en un espacio tridimensional y $\mathbf{q} = [u\ v]^T$ su proyección en dos dimensiones, usando el modelo de cámara de agujero de alfiler.

$$\begin{bmatrix} w\mathbf{q} \\ w \end{bmatrix} = \boldsymbol{K}\ [\boldsymbol{R}|\mathbf{T}] \begin{bmatrix} \mathbf{P} \\ 1 \end{bmatrix} \tag{2.9}$$

$$\begin{bmatrix} wu \\ wv \\ w \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_1^\top & T_x \\ \mathbf{R}_2^\top & T_y \\ \mathbf{R}_3^\top & T_z \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{2.10}$$

multiplicando ambos lados de la expresión por $\frac{1}{T_z}$, en el lado izquierdo de la ecuación se incluye esta multiplicación en la variable w

$$\begin{bmatrix} wu \\ wv \\ w \end{bmatrix} = \begin{bmatrix} \frac{f\mathbf{R}_1^\top}{T_z} & \frac{fT_x}{T_z} \\ \frac{f\mathbf{R}_2^\top}{T_z} & \frac{fT_y}{T_z} \\ \frac{\mathbf{R}_3^\top}{T_z} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{2.11}$$

sea $s = \frac{f}{T_z}$

$$\begin{bmatrix} wu \\ wv \\ w \end{bmatrix} = \begin{bmatrix} s\mathbf{R}_1^\top & sT_x \\ s\mathbf{R}_2^\top & sT_y \\ \frac{\mathbf{R}_3^\top}{T_z} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{2.12}$$

o bien

$$\begin{bmatrix} wu \\ wv \end{bmatrix} = \begin{bmatrix} s\mathbf{R}_1^\top & sT_x \\ s\mathbf{R}_2^\top & sT_y \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{2.13}$$

$$w = \frac{\mathbf{R}_3 \cdot \mathbf{P}}{T_z} + 1 \tag{2.14}$$

cuando $w = 1$ la ecuación 2.13 se convierte en la ecuación de perspectiva débil

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} s\mathbf{R}_1^\top & sT_x \\ s\mathbf{R}_2^\top & sT_y \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{2.15}$$

Esto es, cuando $\frac{\mathbf{R}_3 \cdot \mathbf{P}}{T_z} \to 0$, o bien, $T_z \gg \mathbf{R}_3 \cdot \mathbf{P}$.

2.4. Calibración de la Cámara

Un paso crítico para realizar tareas de visión computacional es la calibración del dispositivo óptico de entrada. La calibración se refiere a estimar los valores de los parámetros internos y externos del modelo matemático de la cámara. Esto permite compensar las distorsiones presentes en las cámaras debido a su constrcción física y, una vez corregidas, reducir el número de variables a calcular pues restaria conocer la orientación y posición de la cámara. Para realizar esto se escriben las ecuaciones de proyección que ligan un conjunto de puntos tridimensionales y sus proyecciones para despues resolverlas para los parámetros [Trucco98]. Una de las distrosiones que causa mayor problema en las cámaras comunes es la que se origina por la inclusión de un lente para enfocar los haces de luz en el sensor. Esta puede causar que las líneas se proyecten como curvas, con lo que se pudiera estropear algorítmos de detección de líneas en la imagen. La distorsión de la imagen en una cámara real se puede modelar con una matriz de parámetros internos como la especificada en la ecuacion 2.16,

$$\mathrm{K} = \begin{bmatrix} \alpha_u & s_k & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.16}$$

donde α_u y α_v representan la distorsión por el tamaño horizontal y vertical de cada pixel. La distancia focal f va implicita, ya que $\alpha_u = fk_u$ y $\alpha_v = fk_v$, donde k_u y k_v son las constantes de resolución horizontal y vertical respectivamente, estos transforman la distancia focal de unidades reales a pixeles. Los escalares u_0 y v_0 son las coordenadas del punto principal y s_k

representa el ángulo entre los pixeles horizontales y verticales, hoy en día la construcción precisa de los sensores ópticos permite suponer una s_k igual a cero o muy cercana a este valor. Finalmente un modelo de proyección en perspectiva completa tomando en cuenta algunas de las distorsiones generadas por la construcción de la cámara se presenta en la ecuación 2.17,

$$\mathbf{q} = \begin{bmatrix} u\prime \\ v\prime \\ w \end{bmatrix} = \mathrm{K}\,[\mathrm{R}|\mathbf{T}]\,\mathbf{P} = \begin{bmatrix} \alpha_u & s_k & u_0 \\ 0 & \alpha_u & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_1^\top & T_x \\ \mathbf{R}_2^\top & T_y \\ \mathbf{R}_3^\top & T_z \\ \mathbf{0}^\top & 1 \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ z_m \\ 1 \end{bmatrix} \tag{2.17}$$

En los libros [Trucco98, Kwon98, Hartley03, Shapiro01] se presentan métodos para realizar el procedimiento de calibración. Una forma de calcular los parámetros internos es mediante la transformación lineal directa (DLT, por sus síglas en inglés) [Hartley03, Abdel-Aziz71]. Se reacomodan las ecuaciones de proyección de tal forma que se obtiene una expresión de la forma $\boldsymbol{A}\mathbf{m}=\mathbf{0}$, donde $\mathbf{m}$ es un vector que contiene cada uno de los elementos de la matriz $\boldsymbol{K}\,[\boldsymbol{R}|\mathbf{T}]$. Un patrón de calibración es utilizado para tener una geometría tridimensional conocida proyectada en una fotografía, se establecen correspondencias y se calcula el vector $\mathbf{m}$ con ayuda de descomposición en valores singulares (SVD, por sus siglas en inglés) para obtener la mejor solución en el sentido de los mínimos cuadrados y que además cumpla con la restricción $\|\mathbf{m}\| = 1$. Dado que el vector tiene once variables independientes es necesario utilizar cinco y media correspondencias para tener un problema determinado.

Como generalmente se tienen más de seis correspondencias el problema se torna sobredeterminado, por lo que se hace necesario utilizar un procedimiento de minimización como Gauss-Newton o Levenberg-Marquardt para obtener una mejor solución. Estos métodos requieren una aproximación inicial a la solución buscada, en este caso la calculada con SVD.

Una vez realizada la estimación de los parámetros intrinsecos para una cámara en condiciones especificas, estos datos pueden utilizarse para "separar" la matriz de parámetros internos de la matriz de parámetros externos. En el algoritmo POSIT descrito en la sección 3.1.1 se necesita conocer previamente la distancia focal por lo que se hace completamente

necesaría la calibración de la cámara.

En este trabajo se utilizó el *Calibration Toolbox* para el paquete de software para matrices *MATLAB* [Bouguet]. Este software entrega resultados para la distancia focal, el cizallamiento, el punto principal y las distorsiones radiales y tangenciales. Se realizaron pruebas con distintos conjuntos de imagenes utilizando una cámara *Canon PowerShot A630* en su modo manual con número F/2.8, sin zoom y con auto-enfocamiento. El número F es un radio de distancia focal al diámetro de apertura del lente, por lo que puede utilizarse como un indicador de la distancia focal. Los efectos de el enfoque automátizado se descartan pues son mínimos a comparación de la distancia focal, los resultados en promedio arrojados se muestran en la tabla 2.2 y la figura 2.11 ilustra uno de los modelos de distorsión computados.

Distancia focal	2660
Punto principal	[1270, 980]
Skew	0.0
Coeficientes de distorsión radial	$[k_{r0} = -0.225,\ k_{r1} = 0.44,\ k_{r2} = 0]$

Tabla 2.2: Resultados de la Calibracion para F/2.8

La distancia focal es presentada en dos cantidades en el programa, una para el eje horizontal y otra para el vertical, para la cámara utilizada las distancias son prácticamente identicas, por lo que solo se reporta una. El punto principal son las coordinadas de intersección del eje principal con el plano de la imagen. Sea $\mathbf{q}_i = [u, v]^\top$ un pixel proveniente de la proyección ideal con perspectiva completa de un punto $\mathbf{P} = [x, y, z]^\top$ en coordenadas de la cámara, adicionalmente se define $r^2 = u^2 + v^2$. El modelo de la distorsión radial se presenta en la ecuación 2.18, dando como resultado $\mathbf{q}_d$, el pixel que incluye los efectos de la distorsión radial.

$$\mathbf{q}_d = \left(1 + k_{r0}r^2 + k_{r1}r^4 + k_{r2}r^6\right)\mathbf{q}_i \tag{2.18}$$

De los datos proporcionados por el fabricante se tiene que la distancia focal $f = 7.3mm$, y la resolución del plano focal en la dirección horizontal es de $k_u = 9062.94\frac{pixel}{pulgada} = 356.8086\frac{pixel}{mm}$, por lo que $\alpha_u = f * k_u = 2604.70278$; de forma similar para la dirección vertical $k_v = 357.6421\frac{pixel}{mm}$ y $\alpha_v = 2610.7875$. Los valores de α_u y α_v son similares a los obtenidos mediante el *Toolbox de Calibración.*

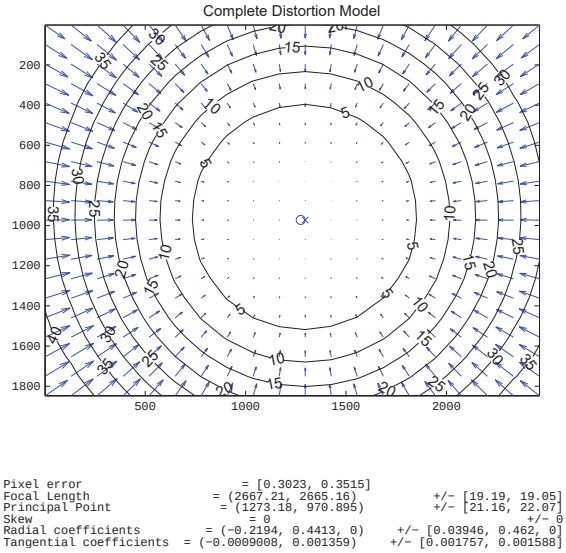


Figura 2.11: Modelo de Distorsión

2.5. Registro

Una de las tareas más importantes en la visión por computadora es la denominada como registro. Esto porque en parte se encarga de encontrar un sistema de referencias común a imágenes capturadas en diferentes situaciones [Zitova03]. El registro de imágenes es un paso esencial cuando se desea obtener mayor información a partir de la combinación de distintas fuentes de datos. Algunos ejemplos donde el registro se considera una etapa básica son la fusión de imágenes, pronostico del clima, sistemas de información geográfica, monitorio de crecimiento de tumores, control de calidad, seguimiento de objetos, etc.

En el proceso de registro requiere de una imagen de referencia y una o más imágenes observadas a alinear geométricamente, en general las imágenes tomadas son diferentes, a menos que las condiciones sean extremadamente controladas. Una clasificación posible de este proceso es de acuerdo a la forma en que se adquieren las imágenes:[Zitova03]

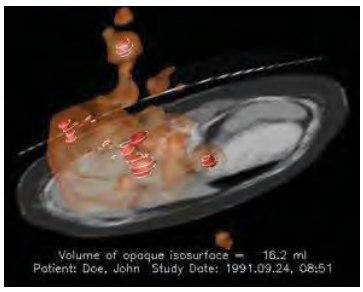
Diferentes Puntos de Vista. Un punto de vista se define como la orientación y posición de la cámara respecto a la escena. Se adquieren imágenes de la misma escena desde

diferentes puntos de vista, el objetivo es obtener mayor información de la escena que la que se tiene con un único punto de vista. La figura 2.12(b) ilustra un conjunto de imágenes aereas registradas para crear un mosaico.

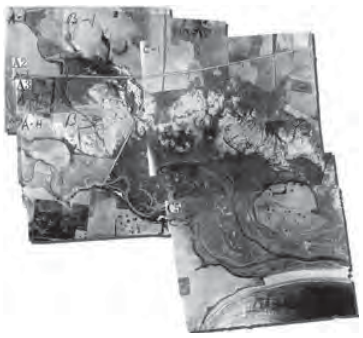
Diferentes Tiempos. Se adquieren imágenes en diferentes tiempos, comúnmente se utilizan intervalos definidos y la meta es conocer la variación de la escena a lo largo del tiempo.

Diferentes Sensores. Las imágenes provienen de distintos tipos de sensores, rayos X, infra-rojos, láser, etc. Se utiliza para generar una mejor y más compleja descripción de la escena, este hecho origina una clasificación conocida como registro multi-modal.

Diferente Naturaleza de Información. Se realiza el registro entre imágenes y un modelo de la escena u objeto. El modelo puede ser una representación digital de la escena, un modelo estándar de la escena, etc. El objetivo es el de establecer la localización de la imagen en el modelo, o viceversa, o bien comparar dicha imagen con algún modelo estándar. La figura 2.12(a) muestra un ejemplo de registro realizado en datos volumétricos y una tomografía.



(a) Registro 2D-3D Datos Médicos



(b) Registro 2D-2D Imágenes Aéreas

Figura 2.12: Ejemplos de Registro de Imágenes

Existen diferentes formas de realizar registro de imágenes y estos dependen, generalmente, de la naturaleza del problema a resolver [Zitova03]. La mayoría de los métodos conocidos para realizar registro de imágenes tienen los siguientes pasos en común

1. Detección de Características.
2. Apareamiento de Características.
3. Estimación de Transformación.
4. Transformación de Imagen.

Cada una de estas etapas esta en entrelazadas y afectan de forma significativa a las otras. A continuación se da un resumen de cada paso listado.

2.5.1. Detección de Características.

Durante esta etapa se buscan objetos salientes en la imagen y esto puede realizarse de forma manual o automática. Recientemente se ha dado un gran énfasis a la investigación de la detección automática; abordándose, principalmente, dos aproximaciones, los métodos basados en áreas con algo (color, profundidad, direccion) en común y los basados en características como puntos, esquinas, líneas, etc. En general se busca que la detección sea invariante a las posibles transformaciones entre las imágenes. Los basados en áreas se utilizan en mayor medida en imágenes como las médicas donde no se tiene mucha diversidad de objetos y se observan grupos de "manchas". En algunas ocasiones la selección de áreas la realiza un usuario experto de forma interactiva. En [Remondino06] se reseñan distintas estrategias tanto para la detección como para la adquisición de información invariante de características.

En imágenes comunes se tienen gran cantidad de objetos con limites bien definidos, es por esto que se favorece la detección basada en características en estos casos. Un tipo de características son las regiones cerradas de un tamaño previamente seleccionado y generalmente se representan por su centro de gravedad pues este es invariante a algunas transformaciones. Otra característica que se utiliza con frecuencia son las líneas, estas pueden pertenecer al contorno de un objeto, el horizonte, carreteras, etc. Usualmente se

representan por un par de puntos, el inicial y final, o bien, por un punto y un vector de dirección. La última característica es el punto y puede obtenerse de una gran cantidad de formas como en la intersección de líneas, centroides de regiones, puntos de inflección en curvas, esquinas, etc. En épocas recientes se han realizado distintas investigaciones para obtener representaciones con mayor información y que logran resultar invariantes a distintas transformaciones.

2.5.2. Apareamiento de Características.

Consiste en encontrar las correspondencias entre las características detectadas en la imagen de referencia y la imagen a registrar. Existen múltiples formas para establecer si una característica detectada en la referencia corresponde o no a otra detectada en la imagen observada, como comparar el tono, la distribución espacial o su descripción simbólica. Un método para realizar el apareamiento consiste en estabecer algún tipo de correlación, para esto se calcula una medida de similaridad entre pares de ventanas y los pares con mayor similaridad se establecen como correspondientes. La correlación sigue siendo utilizada en gran cantidad de estudios pues es relativamente sencillo implementarle en hardware y esto la hace una buena candidata para algoritmos en tiempo real [Zitova03].

Cuando en las imágenes se presenta ruido dependiente de cierta frecuencia, o bien, si se requiere acelerar la velocidad del cómputo se recurre a métodos basados en series de Fourier, estos explotan la representación de la imagen en el dominio de la frecuencia. Recientemente se ha comenzado a investigar con métodos basados en información mutua, la cual es una medida que expresa que tan dependiente es una variable aleatoria de otra. Finalmente se tiene el apareamiento basado en métodos de optimización, que tienen como objetivo encontrar el máximo o mínimo de una función de error o energía interpretada como una medida de similitud [Zitova03].

Otra aproximación que usa características salientes es la que se basa en descriptores invariantes. Las principales condiciones de existencia son la invariancia, el descriptor de una característica correspondiente en la imagen de referencia y la imagen de prueba debe ser el mismo; unicidad, una característica saliente debe tener uno y solo un descriptor;

estabilidad, si bien un descriptor puede deformarse un poco, este debe permanecer similar al original; independencia, todos los elementos de un descriptor (en forma de vector) deben ser independientes. Las características especiales cuyos descriptores tengan mayor similitud se toman como correspondientes, para calcular la medida de similaridad generalmente se utiliza la distancia mínima [Remondino06].

2.5.3. Estimación del Modelo de Transformación.

Una vez conocidas (o estimadas) las correspondencias se procede a calcular los parámetros del modelo de transformación elegido para el problema en particular. La tarea más complicada es encontrar la función que mapea las imágenes sensadas a la de referencia, es decir, conocer las transformaciones esperadas para modelarlas en una expresión matemática. Esta expresión debe ser parametrizada y dependiente de las correspondencias para utilizar esta información en favor de la estimación del modelo de transformación. La forma más directa es la de calcular una transformación global; pero, debido a que las transformaciones pueden ser muy distintas en su naturaleza, en ocasiones es necesario calcular transformaciones locales.

En otra categoría caen las transformaciones elásticas en estas no se buscan los parámetros para una función de transformación, en lugar de esto se considera que las imágenes son conjuntos de láminas elásticas y que sufren distorsiones producidas por fuerzas exteriores. El apareamiento y la estimación se hacen simultáneamente pues se calculan las fuerzas internas mínimas de la malla de láminas elásticas para obtener la deformación presente por fuerzas externas. Partiendo de esta idea han surgido otros esquemas de registro conocidos como registro con fluidos, que utilizan modelos de fluidos viscosos para simular las transformaciones presentes en las imágenes [Zitova03].

2.5.4. Transformación de Imagen.

Finalmente las imágenes deben estar en el mismo sistema de referencias por lo que una de ellas se transforma utilizando el modelo estimado en el paso anterior. Generalmente se transforma la imagen sensada para alinearla con la imagen de referencia, cada uno de los

píxeles de puede ser convertido al nuevo sistema de referencias, sin embargo, esto genera huecos y traslapes. Es por esto que, en algunas ocasiones, es necesario calcular el tono de los píxeles transformados interpolando valores de los píxeles originales utilizando la función de transformación invertida. El método utilizado para la interpolación dependerá de la aplicación a desarrollar, en general se utiliza interpolación bilineal pues se considera adecuada tanto en complejidad computacional como en precisión [Hartley03].

2.6. Conclusiones

En este capítulo se presentó una introducción a la visión por computadora y en la sección 2.2 se da una explicación del proceso de formación de una imagen, así como distintos modelos matemáticos que replican los pasos presentados de forma física para generar imágenes. Se compararon los modelos y se presentó la validez entre de utilizar perspectiva débil en lugar de perspectiva completa bajo condiciones especiales.

En el segundo bloque del capítulo se presentó el problema del registro de imágenes, sus bases y alcances. Se dió una clasificación de dicho problema de acuerdo a la naturaleza de los datos que se desean superponer. Se resumieron cada uno de los pasos comunes para lograr la alineación geométrica de la información visual y se nombraron algunas de las técnicas utilizadas para llevarlos a cabo. En el siguiente capítulo se abordarán distintas soluciones al problema de registro de imagenes en dos dimensiones a un modelo compuesto por puntos en tres dimensiones.

Capítulo 3

Registro de Imagenes en Dos Dimensiones a un Conjunto de Puntos Tridimensionales

El registro de puntos tridimensionales a imagenes de dos dimensiones se refiere al proceso de alinear geométricamente un conjunto de puntos en tres dimensiones con puntos detectados en una imagen bidimensional, es también conocido como registro multimodal por la distinta naturaleza de los datos a registrar [Zitova03]. La medicina es el campo científico donde se ha dado la mayor cantidad de investigación al respecto pues se utilizan sensores que entregan diferentes tipos de imagenes, la industria aeroespacial utiliza estas herramientas para combinar datos de altitud con imágenes. El problema consiste en encontrar la posición y orientación de la cámara con que se adquiere la imagen respecto al objeto fotografiado; las incógnitas pueden resolverse encontrando las correspondencias entre puntos del modelo e imagen para después calcular la pose, o bien, buscándole directamente.

3.1. Obtención de Ubicación de la Cámara a Partir de Correspondencias

Dados cierto número de pares correspondientes de puntos de la imagen y del modelo se pueden calcular los parámetros externos de la cámara al momento de tomar la

fotografía. En esta sección se realiza el análisis de un par de metodologías para calcular la pose. La primera, POSIT, calcula los parámetros externos directamente, mientras que la segunda técnica estima dichos valores pero se expresan con un par de magnitudes signadas que representan la inclinación del plano de la fotografía respecto al plano conformado por tres puntos tridimensionales pertenecientes al modelo.

3.1.1. POSIT

POSIT es un algoritmo lineal diseñado para encontrar los parámetros externos de una cámara, también conocidos como pose de un objeto, respecto a una imagen en perspectiva. El nombre del procedimiento es un acrónimo de *pose a partir de proyección ortográfica y escalamiento iteradas*, por sus siglas en inglés y fué reportado por David DeMenthon en [Dementhon95]. Esto porque se aproxima una matriz de proyección en perspectiva completa mediante una de perspectiva débil a partir de cuatro o más correspondencias y en cada iteración se "corrigen" los puntos de la imagen (perspectiva completa) para que se ajusten mejor a la proyección en perspectiva débil.

3.1.1.1. Pose a partir de proyección ortográfica y escalamiento (POS)

En esta sección del método se calcula la posición y orientación de un objeto respecto a una cámara. Para esto se aproxima una proyección en perspectiva completa desde una perspectiva débil. Se requiere conocer la distancia focal de la imagen de los puntos característicos. En la sección 2.3.5 se obtiene la siguiente expresión para la perspectiva completa (ecuación 2.13)

$$\begin{bmatrix} wu \\ wv \end{bmatrix} = \begin{bmatrix} s\mathbf{R_1}^T & sT_x \\ s\mathbf{R_2}^T & sT_y \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$w = \frac{\mathbf{R_3} \cdot [x\ y\ z]}{T_z} + 1$$

así pues, para cierta w se puede calcular una matriz de proyección dados. La matriz de proyección tiene siete variables independientes, por lo que para calcularla se debe contar

con cuatro pares de puntos correspondientes.

$$\begin{bmatrix} w_1u_1 & w_2u_2 & \cdots & w_nu_n \\ w_1v_1 & w_2v_2 & \cdots & w_nv_n \end{bmatrix} = \begin{bmatrix} s\mathbf{R_1}^T & sT_x \\ s\mathbf{R_2}^T & sT_y \end{bmatrix} \begin{bmatrix} x_1 & x_2 & \cdots & x_n \\ y_1 & y_2 & \cdots & y_n \\ z_1 & z_2 & \cdots & z_n \\ 1 & 1 & \cdots & 1 \end{bmatrix} \tag{3.1}$$

para calcular la matriz de rotación, esta se despeja de la ecuación se utiliza la ecuación 3.1

$$\begin{bmatrix} s\mathbf{R_1}^T & sT_x \\ s\mathbf{R_2}^T & sT_y \end{bmatrix} = \begin{bmatrix} x_1 & x_2 & \cdots & x_n \\ y_1 & y_2 & \cdots & y_n \\ z_1 & z_2 & \cdots & z_n \\ 1 & 1 & \cdots & 1 \end{bmatrix}^{-1} \begin{bmatrix} w_1u_1 & w_2u_2 & \cdots & w_nu_n \\ w_1v_1 & w_2v_2 & \cdots & w_nv_n \end{bmatrix} \tag{3.2}$$

Como se debe garantizar que $\| \mathbf{R_1} \| = \| \mathbf{R_2} \| = 1$ y $\mathbf{R_1} \perp \mathbf{R_2}$ se factoriza el parámetro s de la matriz de proyección agrupando $s\mathbf{R_1}$ y $s\mathbf{R_2}$ en una matriz y aplicando descomposición en valores singulares. Sea $\boldsymbol{A}$ una matriz cuyas columnas sean los vectores de rotación $\mathbf{R}_1$ y $\mathbf{R}_2$ escalados por s, estos vectores son obtenidos de la expresión 3.2. A continuación se presenta la descomposición en valores singulares de la matriz $\boldsymbol{A}$.

$$\boldsymbol{A} = \begin{bmatrix} s\mathbf{R_1} & s\mathbf{R_2} \end{bmatrix} \tag{3.3}$$

$$\boldsymbol{A}^T = \begin{bmatrix} s\mathbf{R_1}^T \\ s\mathbf{R_2}^T \end{bmatrix} \tag{3.4}$$

$$\boldsymbol{A}^T\boldsymbol{A} = \begin{bmatrix} s^2\mathbf{R_1}^T\mathbf{R_1} & s^2\mathbf{R_2}^T\mathbf{R_1} \\ s^2\mathbf{R_1}^T\mathbf{R_2} & s^2\mathbf{R_2}^T\mathbf{R_2} \end{bmatrix} \tag{3.5}$$

Dado que $\| \mathbf{R_1} \| = \| \mathbf{R_2} \| = 1$ y $\mathbf{R_1} \perp \mathbf{R_2}$,

$$\begin{aligned} \boldsymbol{A}^T\boldsymbol{A} &= \begin{bmatrix} s^2(1) & s^2(0) \\ s^2(0) & s^2(1) \end{bmatrix} \\ &= \begin{bmatrix} s^2 & 0 \\ 0 & s^2 \end{bmatrix} \end{aligned} \tag{3.6}$$

finalmente se obtienen los valores singulares

$$\begin{aligned} |\boldsymbol{A}^T\boldsymbol{A} - \lambda I_{2\times 2}| &= 0 \\ |\left(s^2 - \lambda\right) I_{2\times 2}| &= 0 \\ \left(s^2 - \lambda\right)^2 &= 0 \end{aligned} \tag{3.7}$$

de lo que resulta que

$$\lambda_1 = \lambda_2 = s^2 \tag{3.8}$$

y, por lo tanto, los valores singulares son

$$\sigma_1 = \sigma_2 = s \tag{3.9}$$

así pues, la descomposición en valores singulares de la matriz $\boldsymbol{A}$ será

$$\begin{aligned} \boldsymbol{A} &= \begin{bmatrix} s\mathbf{R_1} & s\mathbf{R_2} \end{bmatrix} \\ \begin{bmatrix} s\mathbf{R_1} & s\mathbf{R_2} \end{bmatrix} &= \boldsymbol{U} \begin{bmatrix} s & 0 \\ 0 & s \\ 0 & 0 \end{bmatrix} \boldsymbol{V}^T \\ s\begin{bmatrix} \mathbf{R_1} & \mathbf{R_2} \end{bmatrix} &= \boldsymbol{U} s \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \boldsymbol{V}^T \\ &= s\left(\boldsymbol{U} \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \boldsymbol{V}^T \right) \end{aligned} \tag{3.10}$$

Ya con s, $\mathbf{R_1}$ y $\mathbf{R_2}$ conocidos se calcula el vector de traslación pues se conoce la distancia focal f a priori.

$$T_x = \frac{sT_x}{s} \tag{3.11}$$

$$T_y = \frac{sT_y}{s} \tag{3.12}$$

$$T_z = \frac{f}{s} \tag{3.13}$$

$$\mathbf{R_3} = \mathbf{R_1} \times \mathbf{R_2} \tag{3.14}$$

3.1.1.2. POS con iteraciones

Los puntos tridimensionales se encuentran a tal distancia de la cámara que asumir $w_i = 1$ es una buena aproximación pues la diferencia en profunidad entre los puntos es mínima en comparación con su lejanía a la cámara. Con esta estimación de los valores w_i se calcula la matriz de proyección que mejor ajuste a los puntos correspondientes. La matriz de proyección recién calculada permite mejorar la estimación de los valores w_i, lo que a su vez mejora el siguiente cálculo de la matriz de proyección. Al repetir este proceso se logran empatar los dos modelos de perspectiva. En el algoritmo 1 los valores w_i se inicializan a unos, posteriormente se calcula la matriz de proyección en perspectiva débil para obtener la matriz de proyección en perspectiva completa y finalmente se recalculan los valores w_i, estos pasos se repiten mientras los estimados de la matriz de proyección en perspectiva completa varíen.

Algoritmo 1 POS con ITeraciones

POSIT(N *puntos* $3d[P]$, N *puntos* $2d[q]$, *distancia focal* f)

1 $w_i \leftarrow 1$
2 **mientras** $[\boldsymbol{R}|\mathbf{T}]_k \neq [\boldsymbol{R}|\mathbf{T}]_{k+1}$
3 $[s\mathbf{R_1}^T,\ sT_x;\ s\mathbf{R_2}^T,\ sT_y] \leftarrow [\boldsymbol{P}]^{-1}[w_i \boldsymbol{q}_i]$
4 $\boldsymbol{U}, \boldsymbol{S}, \boldsymbol{V}^T \leftarrow \text{SVD}([s\mathbf{R_1}, s\mathbf{R_2}])$
5 $s_1 \leftarrow \text{PROMEDIO}(\boldsymbol{S})$
6 $[\mathbf{R_1}, \mathbf{R_2}] \leftarrow \boldsymbol{U}\,\boldsymbol{I}\,\boldsymbol{V}^T$
7 $T_x \leftarrow \frac{sT_x}{s_1},\ T_y \leftarrow \frac{sT_y}{s_1},\ T_z \leftarrow \frac{f}{s_1}$
8 $\mathbf{R_3} \leftarrow \mathbf{R_1} \times \mathbf{R_2}$
9 $w_i \leftarrow \frac{\mathbf{R_3 P_i}}{T_z} + 1$
10
11 **regresar** $[\boldsymbol{R}|\mathbf{T}]$

3.1.2. Estimación de Pose con Tres Pares

En gran cantidad de estudios de visión por computadora se pretende conocer la pose de un objeto. Para lograr esto se requieren de, como mínimo, tres pares de puntos correspondientes. A continuación se presenta una metodología desarrollada por T. D. Alter y reportada en [Alter92] para averiguar los parámetros de proyección entre los pares antes mencionados utilizando perspectiva débil.

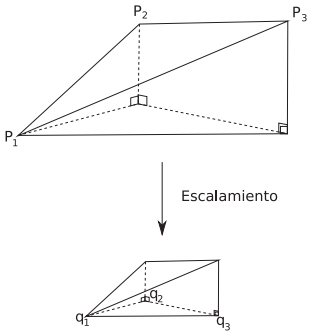
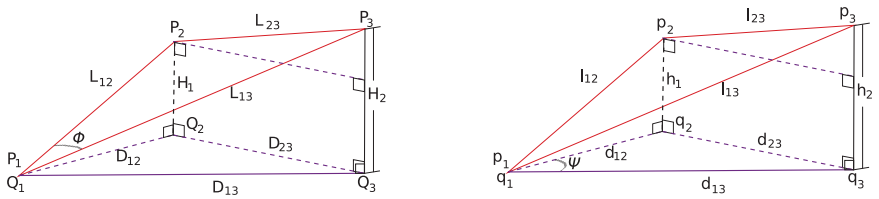


Figura 3.1: Interpretacion de Alter de la Perspetiva Débil

La perspectiva débil puede verse como una proyección ortogonal escalada proporcionalmente a la distancia del objeto a la cámara (sección 2.3.4). En la figura 3.1 se muestra este tipo de perspectiva, cada piramide es una proyección ortogonal y la piramide inferior es un escalamiento de la piramide superior. El problema a resolver es encontrar las inclinación del plano formado por los puntos tridimensionales respecto al plano que forman sus proyecciones en perspectiva débil.

Las figuras 3.2(a) y 3.2(b) son un acercamiento a la piramide superior e inferior, respectivamente. Las variables l_{jk}, L_{jk}, d_{jk} y D_{jk} son las distancias entre los puntos j y k, en el caso de l y L son puntos tridimensionales y bidimensionales para d y D. Los puntos $\mathbf{P}_j$ son parte del modelo tridimensional y los $\mathbf{q}_j$ son sus proyecciones utilizando perspectiva débil. Las cantidades H_j y h_j representan la inclinación entre el plano formado por los puntos tridimensionales y el plano de proyección. Como la pirámide inferior es la superior escalada s veces se puede establecer que $l_{jk} = sL_{jk}$, $d_{jk} = sD_{jk}$, $h_j = sH_j$, etc.



(a) Pirámide Superior (b) Pirámide Inferior

Figura 3.2: Detalle del Proceso de Proyección en Perspectiva Débil

De la figura 3.2(b) se pueden establecer las siguientes relaciones entre triángulos

$$h_1^2 + d_{12}^2 = (sL_{12})^2 \tag{3.15}$$

$$h_2^2 + d_{13}^2 = (sL_{13})^2 \tag{3.16}$$

$$(h_1 - h_2)^2 + d_{23}^2 = (sL_{23})^2 \tag{3.17}$$

Sumando la ecuación 3.15 con 3.16 y restándoles 3.17

$$2h_1h_2 = \left(d_{23}^2 - d_{12}^2 - d_{13}^2\right) + s^2\left(L_{12}^2 + L_{13}^2 - L_{23}^2\right) \tag{3.18}$$

para quitar h_1 y h_2 de la ecuación 3.18 esta se eleva al cuadrado y se sustituyen 3.15 y 3.16 resultando en

$$4\left(s^2L_{12}^2 - d_{12}^2\right)\left(s^2L_{13}^2 - d_{13}^2\right) = \left(-\left(d_{12}^2 + d_{13}^2 - d_{23}^2\right) + s^2\left(L_{12}^2 + L_{13}^2 - L_{23}^2\right)\right)^2 \tag{3.19}$$

al descomponer y reorganizar los términos de la ecuación 3.19 se obtiene

$$\begin{aligned} & \left(4L_{12}^2L_{13}^2 - \left(L_{12}^2 + L_{13}^2 - L_{23}^2\right)^2\right)s^4 \\ - \;& 2\left(2L_{12}^2d_{13}^2 + 2L_{13}^2d_{12}^2 - \left(L_{12}^2 + L_{13}^2 - L_{23}^2\right)\left(d_{12}^2 + d_{13}^2 - d_{23}^2\right)\right)s^2 \\ + \;& \left(d_{12}^2d_{13}^2 - \left(d_{12}^2 + d_{13}^2 - d_{23}^2\right)^2\right) = 0 \end{aligned} \tag{3.20}$$

que tiene la forma $as^4 - 2bs^2 + c = 0$ y utilizando la fórmula general se puede conocer s

$$\begin{aligned} s &= \sqrt{\frac{2b \pm \sqrt{(2b)^2 - 4ac}}{2a}} \\ &= \sqrt{\frac{b \pm \sqrt{b^2 - ac}}{a}} \end{aligned} \tag{3.21}$$

De esta forma la variable s puede tener cuatro valores distintos. Tras evaluar si los términos a, b, c y $b^2 - ac$ son positivos en [Alter92] se prueba que la proyección ortogonal corresponde a los valores de s cuando el término $\pm\sqrt{b^2 - ac}$ toma el signo positivo.

Una vez calculada s se pueden conocer h_1 y h_2 (de las ecuaciones 3.15 y 3.16), sin embargo, los signos de estas ultimas variables se desconocen por la forma en que se construyó el marco para la solución de s y el modelo geométrico. Así pues se elige mantener el signo de h_1 y escoger el signo de h_2 para mantener la igualdad 3.18. Sea σ el signo calculado para h_2

$$\sigma = \begin{cases} 1 & \text{si } d_{12}^2 + d_{13}^2 - d_{23}^2 \leq s^2 \left(L_{12}^2 + L_{13}^2 - L_{23}^2\right) \\ -1 & \text{de otra forma} \end{cases} \tag{3.22}$$

Como h_1 y h_2 son cantidades signadas y se calculan mediante una raíz cuadrada se tiene aun la incertidumbre del signo que toman. Esta ambigüedad representa la posibilidad de que el objeto tridimensional se encuentre de un lado u otro de un plano paralelo al de la imagen proyectada. En la figura 3.3 el cubo en cyan es el reflejo del azul marino respecto

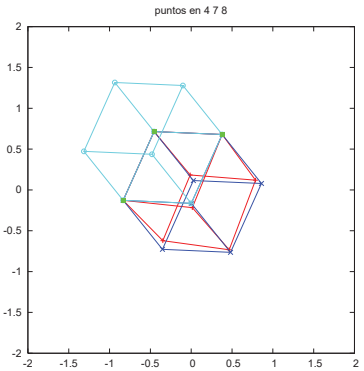


Figura 3.3: Proyecciones con parámetros de Alter.

al plano formado por los puntos verdes, el cubo rojo es la proyección del modelo tridimensional original, las unidades son pixeles. A modo de resumen, el método para calcular los

parámetros de proyección es

$$\begin{aligned}
a &= 4L_{12}^2L_{13}^2 - \left(L_{12}^2 + L_{13}^2 - L_{23}^2\right)^2 \\
b &= 2L_{12}^2d_{13}^2 + 2L_{13}^2d_{12}^2 - \left(L_{12}^2 + L_{13}^2 - L_{23}^2\right)\left(d_{12}^2 + d_{13}^2 - d_{23}^2\right) \\
c &= d_{12}^2d_{13}^2 - \left(d_{12}^2 + d_{13}^2 - d_{23}^2\right)^2 \\
\sigma &= \begin{cases} 1 & \text{si } d_{12}^2 + d_{13}^2 - d_{23}^2 \leq s^2\left(L_{12}^2 + L_{13}^2 - L_{23}^2\right) \\ -1 & \text{de otra forma} \end{cases} \\
s &= \sqrt{\frac{b + \sqrt{b^2 - ac}}{a}} \\
(h_1,\ h_2) &= \left(\sqrt{s^2L_{12}^2 - d_{12}^2},\ \sigma\sqrt{s^2L_{13}^2 - d_{13}^2}\right)
\end{aligned} \tag{3.23}$$

3.1.2.1. Proyección de un cuarto punto tridimensional

Finalmente se presenta una metodología para proyectar un cuarto punto tridimensional $\mathbf{P}_4$ a la imagen. Un punto en tres dimensiones puede expresarse en función de otros tres, es decir, se crea un nuevo sistema de coordenadas con los vectores formados por los punto y un tercero que es el producto cruz de ese par.

$$\mathbf{P}_4 = \alpha\left(\mathbf{P}_2 - \mathbf{P}_1\right) + \beta\left(\mathbf{P}_3 - \mathbf{P}_1\right) + \gamma\left(\mathbf{P}_2 - \mathbf{P}_1\right) \times \left(\mathbf{P}_3 - \mathbf{P}_1\right) + \mathbf{P}_1 \tag{3.24}$$

por lo que se pueden conocer las pseudo-coordenadas α, β y γ resolviendo el sistema de ecuaciones. Ahora se "retro-proyectan" los puntos de la imagen al espacio en tres dimensiones, o bien, se definen los puntos tridimensionales desde el marco de referencia de la cámara

$$\mathbf{P}_1 = \frac{1}{s}(u_1, v_1, w) \tag{3.25}$$

$$\mathbf{P}_2 = \frac{1}{s}(u_2, v_2, h_1 + w) \tag{3.26}$$

$$\mathbf{P}_3 = \frac{1}{s}(u_3, v_3, h_2 + w) \tag{3.27}$$

donde u_i y v_i son las coordenadas en dos dimensiones del punto tridimensional $\mathbf{P}_i$, sustituyendo las ecuaciones 3.25, 3.26 y 3.27 en 3.24

$$\begin{aligned}\mathbf{P}_4 &= \frac{\alpha}{s}(u_{21}, v_{21}, h_1) + \\ &\quad \frac{\beta}{s}(u_{31}, v_{31}, h_2) + \\ &\quad \frac{\gamma}{s^2}(v_{21}h_2 - h_1v_{31}, h_1u_{31} - u_{21}h_2, u_{21}v_{31} - v_{21}u_{21}) + \\ &\quad \frac{1}{s}(u_1, v_1, w) \qquad (3.28) \\ &= \frac{1}{s}\left(\alpha u_{21} + \beta u_{31} + \gamma\frac{v_{21}h_2 - h_1v_{31}}{s} + u_1,\right. \\ &\quad \alpha v_{21} + \beta v_{31} + \gamma\frac{h_1u_{31} - u_{21}h_2}{s} + v_1, \\ &\quad \left.\alpha h_1 + \beta h_2 + \gamma\frac{u_{21}v_{31} - v_{21}u_{21}}{s} + w\right) \qquad (3.29)\end{aligned}$$

donde $u_{ij} = u_j - u_i$ y $v_{ij} = v_j - v_i$. Una vez definida la coordenada tridimensional en coordenadas de la cámara se escala y se proyecta ortográficamente, o bien, se proyecta con perspectiva débil para obtener el punto $\mathbf{q}_4$ en dos dimensiones como se expresa en la ecuación 3.30.

$$\begin{aligned}\mathbf{q}_4 = &\left(\alpha u_{21} + \beta u_{31} + \gamma\frac{v_{21}h_2 - h_1v_{31}}{s} + u_1,\right. \\ &\left.\alpha v_{21} + \beta v_{31} + \gamma\frac{h_1u_{31} - u_{21}h_2}{s} + v_1\right) \qquad (3.30)\end{aligned}$$

En las siguientes secciones se analizan tres distintas metodologías para el registro automático de imagenes bidimensionales a un modelo tridimensional. La primera, SoftPOSIT, se basa en la minimización de una función para estimar tanto correspondencias como la ubicación simultaneamente. El segundo algoritmo es una variación de RANSAC con muestreo guiado. El último procedimiento se basa en la fusión robusta de información para estimar hipótesis de correspondencias con alta probabilidad de ser correctas.

3.2. SoftPOSIT

SoftPOSIT es un algoritmo desarrollado por Daniel Dementhon y reportado en [David02], cuyo objetivo es el de resolver tanto las correspondencias como la matriz de proyección que relacionan a un modelo tridimensional con su imagen en perspectiva. Se basa en dos algoritmos que resuelven los problemas mencionados anteriormente por separado. El primero de ellos es el POSIT [Dementhon95] que resuelve el problema de encontrar la matriz de proyección cuando se conocen las correspondencias y el segundo algoritmo [Gold95] encuentra las correspondencias y los parámetros de la cámara para conjuntos de la misma naturaleza, es decir, puntos de imágenes o puntos de modelos tridimensionales. Para lograr su cometido, SoftPOSIT minimiza una función de error bajo un esquema de recocido simulado.

3.2.1. ubicación de la cámara sin conocer las correspondencias

En las secciones 3.1.1 y 3.1.2 se analizaron metodologías para encontrar los parámetros externos de la cámara con conocimiento previo de las correspondencias entre puntos del modelo tridimensional e imagen. En este apartado se analiza un procedimiento para resolver tanto el problema de la ubicación de la cámara como el de las correspondencias simultaneamente. Para esto emplea un algoritmo similar al procedimiento conocido como Maximización de la Esperanza (Expectation-Maximization, EM por sus siglas en inglés) [Dempster77], donde se separa un problema grande en dos (o más) de menor complejidad. En este caso se estiman las mejores correspondencias (suponiendo que la ubicación se conoce), para después estimar la mejor pose con las relaciones recién computadas. Esto se repite en un marco similar al recocido simulado [Cerny85, Kirkpatrick83] para tratar de evitar mínimos locales.

Si no se conocen las correspondencias cada punto en la imagen puede ser empatado con cualquiera del modelo tridimensional. Sean $\mathbf{q}_i = [u_i \; v_i]^T$ un punto bidimensional proveniente de la imagen, $\mathbf{P}_j = [x_i \; y_i \; z_i]^T$ un punto en tres dimensiones obtenido del modelo del objeto y w_j (ecuación 2.13) el valor que "corrige" a un punto de la imagen para la

proyección en perspectiva débil del punto $\mathbf{P}_j$, es decir,

$$w_j \mathbf{q}_i = \begin{bmatrix} s\mathbf{R_1}^T & sT_x \\ s\mathbf{R_2}^T & sT_y \end{bmatrix} \begin{bmatrix} \mathbf{P}_j \\ 1 \end{bmatrix} \tag{3.31}$$

Cuando ambos lados de la ecuación son iguales la matriz de proyección es la correcta y la distancia entre el punto corregido de la imagen y el punto del modelo proyectado es cero, así pues se busca minimizar esta distancia al cuadrado.

$$d_{jk}^2 = \left(\begin{bmatrix} s\mathbf{R_1}^T & sT_x \end{bmatrix} \begin{bmatrix} \mathbf{P}_j \\ 1 \end{bmatrix} - w_j u_i \right)^2 + \left(\begin{bmatrix} s\mathbf{R_2}^T & sT_y \end{bmatrix} \begin{bmatrix} \mathbf{P}_j \\ 1 \end{bmatrix} - w_j v_i \right)^2 \tag{3.32}$$

Sea a_{ij} el coeficiente de asignamiento entre un punto $\mathbf{P}_j$ del modelo en tres dimensiones de un objeto y sus proyección en dos dimensiones $\mathbf{q}_i$.

$$a_{ij} = \begin{cases} 1 & \text{si } \mathbf{q}_i \text{ es la proyección de } \mathbf{P}_j \\ 0 & \text{de otra forma} \end{cases} \tag{3.33}$$

Ahora se esboza una función de error donde se incluye tanto la distancia al cuadrado, como las correspondencias para N puntos de la imagen y M puntos del modelo tridimensional.

$$E = \sum_{i=1}^{N} \sum_{j=1}^{M} a_{ij} d_{ij}^2 \tag{3.34}$$

Dado que se busca minimizar la expresión 3.34 y pudiera llegarse a la solución trivial para las correspondencias, es decir, $a_{ij} = 0 \; \forall \{i, j\}$, se introduce una variable α para alejar el mínimo de esta solución (ecuación 3.35).

$$E = \sum_{i=1}^{N} \sum_{j=1}^{M} a_{ij} (d_{ij}^2 - \alpha) \tag{3.35}$$

3.2.1.1. Estimación de Correspondencias

La primer parte del algoritmo calcula una serie de probabilidades de correspondencias entre puntos, estas dependerán de la distancia entre la proyección del punto 3d y un punto 2d. Las probabilidades se guardan en una matriz de asignamiento y serán usadas para estimar la proyección en perspectiva débil que mejor les ajuste. En general una parte de los puntos 3d no serán proyectados por estar ocultos para el área visual de la cámara, además cuando se detectan puntos 2d en una imagen se presentan datos apócrifos. Es por esto que se agregan a la matriz de asignamiento un renglón y una columna extras, estos pueden verse como auxiliares para cuando no se pueda decidir o no se encuentre una mejor correspondencia.

Se busca establecer una correspondencia entre punto 2d y 3d si su distancia es la más pequeña, o bien, minimizar una función de error del tipo

$$E = \sum_{i=1}^{N} \sum_{j=1}^{M} a_{ij} f\left(d_{ij}\right) \tag{3.36}$$

Un punto de la imagen solamente puede corresponder a un punto 3d (y viceversa), es decir, las correspondencias tienen una restricción de dos vías. Esto puede expresarse mediante las siguientes restricciones a la matriz se correspondencias,

$$\sum_{i=1}^{N+1} a_{ij} = 1 \text{ para } 1 \leq j \leq M \tag{3.37}$$

$$\sum_{j=1}^{M+1} a_{ij} = 1 \text{ para } 1 \leq i \leq N \tag{3.38}$$

Este problema discreto se transforma en uno continuo utilizando un marco de recocido simulado, lo que evita introducir condiciones “extrañas” en la función de error. Para lograr lo expresado en las ecuaciones 3.37 y 3.38 en cada ciclo la matriz se ve sometida a un procedimiento creado por Sinkhorn y reportado en [Sinkhorn64] que transforma una matriz con elementos todos positivos en una (doblemente estocástica). Esto lo logra normalizando renglones y columnas alternadamente. Para asegurar que todos los elementos de la matriz

de asignamiento sean positivos estos se inicializan a los siguientes valores

$$a_{ij} = \begin{cases} exp(-\beta(d_{ij}^2 - \alpha)) & \text{si } 1 \leq i \leq N \text{ y } 1 \leq j \leq M \\ \gamma = \text{ constante pequeña} & \text{de otra forma} \end{cases} \tag{3.39}$$

El término β_T es una variable de control para el recocido simulado y se incrementa en cada paso del algoritmo. A medida que el valor de β_T aumenta, los términos a_{ij} que corresponden a las distancias más pequeñas tienden a 1 y los demás tienden a 0; pasando de la aproximación continua a el problema discreto original.

La combinación de la exponenciación de los términos de la matriz y el procedimiento de Sinkhorn tiene una similitud con el criterio de activación *softmax* [Bridle90] y en [Gold95] se le da el nombre de *softassign*. En general un algoritmo para encontrar correspondencias inexactas basado en recocido simulado y softassign tendría la forma presentada en el pseudo-código del algoritmo 2. La estimación de las correspondencias se hace calculando las distancias al cuadrado entre proyecciones de puntos 3D y los puntos 2D detectados.

Algoritmo 2 Correspondencias con Recocido Simulado

```
CORRESPONDENCIAS(M puntos 3d[P], N puntos 2d[q])
1  β_T ← β_T0
2  mientras β_T < β_Tfinal
3    a^0_ij ← exp(−β_T(d^2_ij − α)) si 1 ≤ i ≤ N y 1 ≤ j ≤ M
4    mientras a^0_ij ≠ a^1_ij
5      a^1_ij ← NORMALIZARENGLONES(a^0_ij) si 1 ≤ i ≤ N
6      a^0_ij ← NORMALIZACOLUMNAS(a^1_ij) si 1 ≤ j ≤ M
7    β_T ← ACTUALIZA(β_T)
8    resto del algoritmo ...
9
10 regresar a_ij
```

3.2.1.2. Pose

Para estimar la pose que mejor se ajuste a las correspondencias calculadas se minimiza la función de error caracterizada en la ecuación 3.35. Sean $\mathbf{O_1}$ y $\mathbf{O_2}$ el primer y segundo renglón de la matriz de proyección en perspectiva débil, respectivamente, $\mathbf{P}_j = [x_j\ y_j\ z_j\ 1]^T$ un punto tridimensional en coordenadas homogéneas, $\mathbf{q}_i = [u_i\ v_i]^T$ un punto bidimensional y a_{ij} el "peso" calculado para al par i,j de acuerdo a lo visto en el apartado anterior. Así pues la función de energía puede reescribirse como

$$\begin{aligned} E &= \sum_{i=1}^{N}\sum_{j=1}^{M} a_{ij}(d_{ij}^2 - \alpha) \\ E &= \sum_{i=1}^{N}\sum_{j=1}^{M} a_{ij}\left(\left[\sqrt{(\mathbf{O_1}\cdot\mathbf{P_j} - w_j u_i)^2 + (\mathbf{O_2}\cdot\mathbf{P_j} - w_j v_i)^2}\right]^2 - \alpha\right) \\ E &= \sum_{i=1}^{N}\sum_{j=1}^{M} a_{ij}\left((\mathbf{O_1}\cdot\mathbf{P_j} - w_j u_i)^2 + (\mathbf{O_2}\cdot\mathbf{P_j} - w_j v_i)^2 - \alpha\right) \end{aligned}$$

se toman las primeras derivadas de E respecto a las componentes de los vectores O_1 y 0_2 y se igualan a cero. En el caso de O_1

$$\begin{aligned} \frac{\partial E}{\partial O_1} &= \sum_{i=1}^{N}\sum_{j=1}^{M} a_{ij}\left[2\left(\mathbf{O_1}\cdot\mathbf{P_j} - w_j u_i\right)\left(\mathbf{P_j}\right)\right] = 0 \\ &= \sum_{i=1}^{N}\sum_{j=1}^{M} a_{ij}\left(\mathbf{O_1}\cdot\mathbf{P_j}\right)\mathbf{P_j} - \sum_{i=1}^{N}\sum_{j=1}^{M} a_{ij} w_j u_i\left(\mathbf{P_j}\right) \\ &= \sum_{i=1}^{N}\sum_{j=1}^{M} a_{ij}\left(\mathbf{P_j}\mathbf{P_j}^T\right)\mathbf{O_1} - \sum_{i=1}^{N}\sum_{j=1}^{M} a_{ij} w_j u_i\left(\mathbf{P_j}\right) \end{aligned} \tag{3.40}$$

finalmente

$$\mathbf{O_1} = \left[\sum_{i=1}^{N}\sum_{j=1}^{M} a_{ij}\left(\mathbf{P_j}\mathbf{P_j}^T\right)\right]^{-1} \sum_{i=1}^{N}\sum_{j=1}^{M} a_{ij} w_j u_i \mathbf{P_j} \tag{3.41}$$

Similarmente para O_2

$$\mathbf{O_2} = \left[\sum_{i=1}^{N}\sum_{j=1}^{M} a_{ij}\left(\mathbf{P_j}\mathbf{P_j}^T\right)\right]^{-1} \sum_{i=1}^{N}\sum_{j=1}^{M} a_{ij} w_j v_i \mathbf{P_j} \tag{3.42}$$

Después de calcular los vectores O_1 y O_2 se procede a *extraer* los parámetros para la cámara en proyección de perspectiva completa tal y como se muestra en la sección 3.1.1. Al introducir la minimización de O_1 y O_2 en el algoritmo esbozado en el pseudo-código

2 se obtiene lo que en [David02] se denomina SoftPOSIT. El pseudo-código 3 representa el algoritmo SoftPOSIT, en primera instancia se calculan las proyecciones de los puntos tridimensionales P con una matriz inicial RT_0. Posteriormente se calculan las distancias al cuadrado entre proyecciones y puntos detectados en la imagen q, estas medidas serviran como indicadores de probabilidad de las correspondencias. Se estiman correspondencias normalizando alternadamente columnas y renglones, con estas correspondencias se calculan los parámetros de proyección usando POSIT.

3.3. RANSAC Guiado por Probabilidad

En esta sección se aborda una aproximación con un algoritmo que busca el modelo con un procedimiento similar al RANSAC [Fischler81] pero la selección de datos observados no es del todo aleatoria. En lugar de escoger datos al azar se utiliza un resultado reportado en [Ben-Arie90], el cual establece que existen altas probabilidades de que la proyección de un ángulo o la relación de distancia entre dos pares de puntos sean iguales o cercanos a los que guardan en su forma tridimensional. Para mantener al mínimo el costo computacional de calcular el modelo para cada iteración del RANSAC se utiliza un método que utiliza solamente tres puntos correspondientes para obtener la pose.

3.3.1. Efecto de Pico de Probabilidad

El efecto de picos de probabilidad fue reportado en [Ben-Arie90] y dice que existe una gran posibilidad de que una medida geométrica en tres dimensiones se mantenga cercana a la misma cantidad (salvo el escalamiento) al ser proyectada utilizando perspectiva débil. Se observa que la función de probabilidad tiene un máximo cuando la cantidad medida en tres dimensiones es parecida a la encontrada en el espacio bidimensional. Se basa en un modelo probabilista llamado esfera de observabilidad que garantiza que cualquier punto de vista sea igualmente probable. Se toman como casos de estudio el ángulo entre dos líneas y el cociente entre las distancias de dos líneas.

3.3.1.1. La Esfera de Observabilidad

La esfera de observabilidad es un modelo probabilista que caracteriza las posibilidades de que una característica de un objeto sea visible desde un punto perteneciente a una

Algoritmo 3 SoftPOSIT (correspondencias y pose)

SoftPOSIT(M *puntos* $3d[P]$, N *puntos* $2d[q]$, *distancia focal* f, RT_0)

1 $\beta_T \leftarrow \beta_{T0}$
2 $a_{ij}^0 \leftarrow \gamma$ si $i = N+1$ o $j = M+1$
3 **mientras** $\beta_T < \beta_{Tfinal}$
4 $\quad a_{ij}^0 \leftarrow exp(-\beta_T(d_{ij}^2 - \alpha))$ si $1 \leq i \leq N$ y $1 \leq j \leq M$
5 $\quad$ **mientras** $a_{ij}^0 \neq a_{ij}^1$
6 $\quad\quad a_{ij}^1 \leftarrow$ normalizaRenglones(a_{ij}^0) si $1 \leq i \leq N$
7 $\quad\quad a_{ij}^0 \leftarrow$ normalizaColumnas(a_{ij}^1) si $1 \leq j \leq M$
8 $\quad \beta_T \leftarrow$ actualiza(β_T)
9 $\quad L \leftarrow \sum_{i=1}^{N}\sum_{j=1}^{M} a_{ij}\left(\mathbf{P_j}\mathbf{P_j}^T\right)$
10 $\quad \mathbf{O_1} = [L]^{-1}\sum_{i=1}^{N}\sum_{j=1}^{M} a_{ij}w_j u_i \mathbf{P_j}$
11 $\quad \mathbf{O_2} = [L]^{-1}\sum_{i=1}^{N}\sum_{j=1}^{M} a_{ij}w_j v_i \mathbf{P_j}$
12 $\quad [U, S, V^T] \leftarrow$ svd$([\mathbf{O_1}(1:3), \mathbf{O_2}(1:3)])$
13 $\quad s_1 \leftarrow$ promedio(S)
14 $\quad [\mathbf{R_1}, \mathbf{R_2}] \leftarrow UIV^T$
15 $\quad T_x \leftarrow \frac{\mathbf{O_1(4)}}{s_1}$
16 $\quad T_y \leftarrow \frac{\mathbf{O_2(4)}}{s_1}$
17 $\quad T_z \leftarrow \frac{f}{s_1}$
18 $\quad \mathbf{R_3} \leftarrow \mathbf{R_1} \times \mathbf{R_2}$
19 $\quad w_j \leftarrow \frac{\mathbf{R_3}\cdot\mathbf{P_j}}{T_z} + 1$
20 **regresar** a_{ij}

esfera. Se utiliza el modelo de proyección en perspectiva débil para aproximar la proyección en perspectiva completa, al usar este modelo la orientación de la cámara respecto al objeto tiene mayor importancia que la distancia entre ellos. Una punto de un modelo es visible si se puede trazar un rayo desde este hasta el punto de vista sin interrupciones, así pues una región de visibilidad es la unión de los puntos desde donde este el punto 3d es visible. Dado que una característica del objeto 3d es la unión de sus puntos la región de visibilidad de una característica es la unión de las regiones de visibilidad de los puntos que la componen.

Cada orientación del observador respecto al objeto y es igualmente probable que las demás, esto puede expresarse como una esfera que envuelve al objeto con superficie "equiprobable" cuyas normales representan una orientación del observador, esto se modela como una esfera ilustrada en la figura 3.4.

Figura 3.4: Esfera de Observabilidad

Se define la probabilidad de observar una la característica a de un objeto tridimensional desde la esfera con radio ι como

$$P(a) = \frac{F(a)}{4\pi\iota^2} \tag{3.43}$$

La cara plana de un objeto tridimensional será visible prácticamente desde la mitad de la esfera de observabilidad, pues el tamaño del objeto es muchas veces menor que su distancia al observador. Así pues la probabilidad de que esta cara sea vista será

$$\begin{aligned} P(a) &= \frac{F(a)}{4\pi\iota^2} \\ &\simeq \frac{2\pi\iota^2}{4\pi\iota^2} \\ &= \frac{1}{2} \end{aligned} \tag{3.44}$$

Supóngase que un objeto tridimensional convexo esta compuesto de caras planas $a_1, a_2, \ldots, a_n$ cuyas normales son $t_1, t_2, \ldots, t_n$. La probabilidad de observar simultáneamente

el conjunto C de m caras será proporcional a la intersección de sus regiones de visibilidad, es decir,

$$\begin{aligned} P(a_1, a_2, \ldots, a_m) &= \frac{1}{4\pi\iota^2} \bigcap F(a_i) \; \forall a_i \in C \\ &= \int\int \prod_{i=1}^{m} A(v(s) \cdot t_i)\, ds \end{aligned} \tag{3.45}$$

donde $v(s)$ es el vector que representa la dirección del punto de vista normal al diferencial de área ds y

$$A(v(s) \cdot t_i) = \begin{cases} 1 & \text{si } v(s) \cdot t_i \geq 0 \\ 0 & \text{de otra forma} \end{cases} \tag{3.46}$$

Otras características como ejes o vértices del modelo pueden ser construidas a partir de la combinación de dos o más caras. Por ejemplo un eje se puede definir como la región de intersección entre dos caras planas y podrá observarse desde la región de visibilidad de ambas caras pues esta línea pertenece a cada una. Similarmente un punto puede establecerse como la intersección de tres caras, por lo que será visible desde las regiones de observabilidad de las caras que lo definen.

De esta manera, la probabilidad de observar una característica compuesta por la intersección de m caras con normales $T = \{t_1, t_2, \ldots, t_m\}$ será

$$P(a_1, a_2, \ldots, a_m) = \frac{1}{4\pi\iota^2} \bigcup_{i=1}^{m} F(a_i) \tag{3.47}$$

$$= \int\int E(v(s), T)\, ds \tag{3.48}$$

donde $v(s)$ es el vector que representa la dirección del punto de vista normal al diferencial de área ds y

$$E(v(s), T) = \begin{cases} 1 & \text{si } \exists\, v(s) \cdot t_i \geq 0 \\ 0 & \text{de otra forma} \end{cases} \tag{3.49}$$

En [Ben-Arie90] se parametriza el ángulo proyectado (β) respecto al original (α) y la dirección, descompuesta en dos ángulos (σ, τ), normal al plano de proyección.

$$\beta = f(\alpha, \sigma, \tau) \tag{3.50}$$

Combinando esta parametrización y la esfera de observabilidad se propone que la probabilidad de observar un ángulo proyectado (ecuación 3.52) es proporcional a la razón del área que generan los ángulos de parametrización sobre la esfera (ecuación 3.51) y la superficie total de la esfera de observabilidad.

$$dA = \iota^2 sin\sigma d\sigma d\tau \tag{3.51}$$

$$\Delta p = \frac{1}{4\pi} sin\sigma d\sigma d\tau \tag{3.52}$$

Posteriormente en [Ben-Arie90] se utilizan técnicas de integración numérica para caracterizar una función de densidad de probabilidad para el cociente de ángulos. Con esto se concluye que la probabilidad de que un ángulo sea proyectado con un valor cercano al original es grande. Adicionalmente se calcula una función de densidad condicional de un ángulo proyectado dado el ángulo "real" del modelo.

Para el caso de la proyección de líneas se parametriza la línea proyectada ($\mathbf{b}$) con el vector de dirección normal al plano de proyección ($\mathbf{v}$), la línea original ($\mathbf{a}$) y un escalamiento (s) correspondiente a la distancia de la cámara al objeto. Dado que se utiliza la esfera de observabilidad y la distancia entre el plano de proyección y el objeto es la misma para todas las direcciones el efecto del escalamiento se descarta pues es el mismo para toda la esfera.

$$\mathbf{b} = s\left[\mathbf{a} - \mathbf{v}(\mathbf{a} \cdot \mathbf{v})\right] \tag{3.53}$$

Inicialmente se calcula la función densidad de probabilidad para la razón de dos distancias, una proveniente del modelo y su proyección en la imagen. Posteriormente se calcula otra para la razón del cociente de un par distancia en la imagen al cociente de un par de distancias en el modelo, esto con el objetivo de obtener una función de densidad de probabilidad similar a la razón de ángulos.

Finalmente se caracteriza una función de densidad de probabilidad conjunta para ángulos y razón de distancias. Posteriormente se evalúan criterios de compatibilidad entre

un par de líneas que puedan originar ángulos. Tras dicho análisis se establece que dichos criterios son proporcionales a la probabilidad de observancia, por lo que se puede utilizar la función de densidad de probabilidad conjunta para ángulos y la razón de distancias para calcularlos.

3.3.2. RANSAC

RANSAC es un algoritmo desarrollado, por Fischler y Bolles [Fischler81], para encontrar los parámetros de un modelo que mejor se ajuste a un conjunto de datos observados. En [Hartley03] se estudia su uso aplicado a problemas de visión por computadora. La principal ventaja de este algoritmo es su robustez ante la presencia de datos apócrifos y su mayor problema es que no se puede establecer un límite superior en el tiempo de ejecución.

Se tiene un conjunto D con n datos observados y un modelo parametrizado que pudiese explicar algunos de estos datos, el objetivo es estimar los parámetros del modelo para que se. Se elige aleatoriamente un subconjunto s, o bien una muestra, con el mínimo número datos m para calcular los parámetros del modelo. Se calcula un error sobre el conjunto total de datos para determinar si los argumentos recién computados logran que el modelo abarque una cierta cantidad de datos, con lo que los parámetros pueden tomarse como buenos y, en su caso, reemplazar los calculados anteriormente.

En el algoritmo 4

D	$\rightarrow$	Datos observados
M	$\rightarrow$	Modelo parametrizado
ERA	$\rightarrow$	Error limite para considerar un dato como apócrifo
MDP	$\rightarrow$	Mínimo número de datos .^{ex}plicados"para aceptar parámetros

3.3.3. RANSAC Guiado

El algoritmo RANSAC es uno de los más utilizados en el campo de visión por computadora, pues en general la información visual contiene ruido u oclusión, siendo este En [Fischler81] se menciona que una de las mejoras más evidentes para el RANSAC es no usar un criterio aleatorio para elegir las muestras, es precisamente esta idea la que se implementa en esta modificación al marco general del algoritmo. Se utilizan los coeficientes de compatibilidad mencionados en [Ben-Arie90] para elaborar una lista ordenada, en cada iteración del RANSAC se reclasifican los elementos de la lista y se elimina el coeficiente recién utilizado. Para establecer el concenso se proyectan los puntos tridimensionales res-

Algoritmo 4 RANSAC General

```
RANSAC(D, M, ERA, MDP)
 1  iteraciones ← 0
 2  parametros ← ninguno
 3  error ← maxError
 4  mientras iteraciones < maxIteraciones
 5    s ← SUBCONJUNTOALEATORIO(D)
 6    p ← CALCULAPARAMETROS(s)
 7    d ← REMOVERAPOCRIFOS(M, p, D, s, ERA)
 8    si |d| > MDP
 9       entonces
10                e ← CALCULAERROR(M, p, d, s)
11                si e < error
12                   entonces
13                            error ← e
14                            parametros ← p
15
16
17    INCREMENTA(iteraciones)
18
19  RETURNparametros
```

tantes como se menciona en la sección **??**. Se utiliza la distancia entre puntos proyectados y puntos detectados para establecer las correspondencias, se impone la restricción de que un punto detectado corresponda únicamente a un punto proyectado. Al final de los ciclos del método se obtiene una lista de correspondencias que serán utilizadas para computar la matriz de proyección.

3.4. Registro Basado en Fusión Robusta de Información

En este capítulo se analiza una aproximación al registro de modelo a imagen basada en la fusión robusta de información. Esto es posible pues se pueden generar las hipótesis de apareamiento más probables gracias al efecto de pico de probabilidad. Además, para estimar las hipótesis se utiliza un procedimiento de búsqueda conocido como desplazamiento de la media (mean shift)[Fukunaga75], lo que agrega un comportamiento robusto al algoritmo. Dicho procedimiento necesita el grado de certidumbre de la fuente de información la cual también es calculada.

3.4.1. Estimación de la Incertidumbre de Proyección

Para poder utilizar el algoritmo Mean Shift se necesita conocer la incertidumbre con que se obtienen los datos. En este caso se proyecta el mismo punto con diferentes hipótesis de parámetros de proyección y se estima la incertidumbre de acuerdo a lo reportado en [Shimshoni99].

Se trata de establecer como afectan los errores de proyección de $\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3$ cuando se estima la posición de un cuarto punto tridimensional proyectado utilizando los parámetros calculados utilizando tres pares como fue revisado en la sección 3.1.2. Para ello se considera que los puntos tridimensionales tomados para estimar los parámetros no se proyectan exactamente a la localización de sus pares en la imagen. Transformando la ecuación 3.30

$$\begin{aligned}\mathbf{q}_4 = \quad & [(u_1, v_1) \centerdot (1-\alpha-\beta, -\gamma H_2 + \gamma H1) + (u_2, v_2) \centerdot (\alpha, \gamma H_2) + (u_3, v_3) \centerdot (\beta, -\gamma H_1)\,, \\ & (u_1, v_1) \centerdot (\gamma H_2 - \gamma H1, 1-\alpha-\beta) + (u_2, v_2) \centerdot (-\gamma H_2, \alpha) + (u_3, v_3) \centerdot (\gamma H_1, \beta)]\end{aligned} \quad (3.54)$$

Sean $\epsilon_\mathbf{1}$, $\epsilon_\mathbf{2}$, $\epsilon_\mathbf{3}$ los vectores de error de proyección de los puntos de la imagen. Proyectando

Algoritmo 5 RANSAC Guiado por Probabilidad

```
gRANSAC(D2d, D3d, Modelo, Apocrifo, MinAcepta)
 1  Pares ← APAREA(D2d, D3d)
 2  Mc ← CALCULACOEFICIENTES(Pares)
 3  Pares ← ORDENA(Pares, Mc)
 4  iteraciones ← 0
 5  parametros ← ninguno
 6  error ← maxError
 7  mientras iteraciones < maxIteraciones
 8    par ← PARACTUAL(Pares)
 9    p ← CALCULAPARAMETROS(par, D2d, D3d)
10    d ← REMOVERAPOCRIFOS(Modelo, Apocrifo, p, D2d, D3d)
11    si |d| > MDP
12       entonces
13                e ← CALCULAERROR(Modelo, p, d)
14                si e < error
15                   entonces
16                            error ← e
17                            parametros ← pcorrespondencias ← d
18
19
20    INCREMENTA(iteraciones)
21
22  RETURNparametros, correspondencias
```

el cuarto punto con los puntos base de la imagen perturbados ($\mathbf{q_k} + \epsilon_\mathbf{k}$) y restando la proyección inicial ($\mathbf{q_k}$)

$$\mathbf{q}_4(\mathbf{q}_1 + \epsilon_1, \mathbf{q}_2 + \epsilon_2, \mathbf{q}_1 + \epsilon_1) - \mathbf{q}_4(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3) = \tag{3.55}$$

$$\begin{aligned}[\epsilon_\mathbf{1} \centerdot (1 - \alpha - \beta, -\gamma H_2 + \gamma H1) + \epsilon_2 \centerdot (\alpha, \gamma H_2) + \epsilon_3 \centerdot (\beta, -\gamma H_1)\,,\\ \epsilon_\mathbf{1} \centerdot (\gamma H_2 - \gamma H1, 1 - \alpha - \beta) + \epsilon_2 \centerdot (-\gamma H_2, \alpha) + \epsilon_3 \centerdot (\gamma H_1, \beta)]\end{aligned} \tag{3.56}$$

Tomando como notación para los vectores de la ecuación 3.56

$$\begin{aligned}\Psi_1 &= (1 - \alpha - \beta, -\gamma H_2 + \gamma H1)\\ \Psi_2 &= (\alpha, \gamma H_2)\\ \Psi_3 &= (\beta, -\gamma H_1)\\ \Psi_1^\perp &= (\gamma H_2 - \gamma H1, 1 - \alpha - \beta)\\ \Psi_2^\perp &= (-\gamma H_2, \alpha)\\ \Psi_3^\perp &= (\gamma H_1, \beta)\end{aligned} \tag{3.57}$$

entonces la ecuación 3.56 será

$$\left(\epsilon_\mathbf{1} \cdot \Psi_1 + \epsilon_2 \cdot \Psi_2 + \epsilon_3 \cdot \Psi_3, \epsilon_\mathbf{1} \cdot \Psi_1^\perp + \epsilon_2 \cdot \Psi_2^\perp + \epsilon_3 \cdot \Psi_3^\perp\right) \tag{3.58}$$

Si se toma la restricción de que $\|\epsilon_\mathbf{1}\| = \|\epsilon_\mathbf{2}\| = \|\epsilon_\mathbf{3}\| = \epsilon$ y además que H_1 y H_2 son constantes, se tiene que los ϵ_i que maximizan el error en la coordenada u son

$$\epsilon_\mathbf{i} = \epsilon \frac{\mathbf{\Psi_i}}{\|\Psi_i\|} \tag{3.59}$$

Como $\Psi_i \cdot \Psi_i^\perp = 0$ la componente v del error es también cero, y la región de incertidumbre es circular. Así pues se concluye que si el error de los puntos de la imagen detectados ($\mathbf{q_1}$, $\mathbf{q_2}$, $\mathbf{q}_3$) al de los proyectados mediante el modelo de alter ($\mathbf{P}_1$, $\mathbf{P}_2$, $\mathbf{P}_3$) es ϵ entonces la región de incertidumbre de proyección de un cuarto punto $\mathbf{P}_4$ es circular y está caracterizada por sus coordenadas afines extendidas α, β, γ y por los parámetros de proyección H_1 y H_2. La magnitud ϵ es generada por el error en la detección de las características y generalmente se manejan uno o dos píxeles. Posteriormente se analiza el error introducido por variaciones en el cálculo de H_1 y H_2 [Shimshoni99], lo que resulta en que las regiones de incertidumbre se tornan elípticas.

Finalmente, para calcular el radio del área de incertidumbre se recalcula el punto q_4 pero utilizando los puntos de la imagen con el error máximo agregado, esto es $\mathbf{q_1} + \epsilon_\mathbf{1}$, $\mathbf{q_2} + \epsilon_\mathbf{2}$ y $\mathbf{q_3} + \epsilon_\mathbf{3}$. La distancia del punto original al punto perturbado es el radio de la región circular de incertidumbre.

3.4.2. Fusión Robusta de Información

La fusión de datos es un campo nuevo que se dedica a combinar datos de distintas fuentes para obtener resultados más confiables que los obtenidos si se analiza de manera individual dicha información. Esta aproximación al procesamiento de información es similar a lo que los seres humanos realizan, pues cuentan con varias fuentes de información y estas se combinan para "entender" mejor el mundo. La fusión de información se puede clasificar de acuerdo al nivel al que se realiza la combinación de la fusión.

Fusión a bajo nivel. Se caracteriza por combinar datos crudos obtenidos de sensores, como fotografías, escáneres infra-rojos, sonares, etc. Para combinar este tipo de datos generalmente se hace un registro de las diversas fuentes.

Fusión a nivel medio. La combinación de la información se realiza a nivel de características, por ejemplo en una imagen las esquinas, contornos, etc. y generalmente se utiliza para obtener una lista de características relevantes.

Fusión a nivel alto. Esta se realiza a nivel de decisiones o niveles de confianza de expertos, para combinar este tipo de información se utilizan esquemas de votación, estadística, lógica difusa, etc.

En este trabajo la fusión de información se utiliza para combinar y validar diferentes hipótesis de apareamiento de puntos provenientes de una imagen bidimensional y puntos que caracterizan el modelo tridimensional proyectado en la imagen.

Sean $\mathbf{x}_i \in \mathbb{R}^p \, i = 1, ..., n$ datos de dimensión p y cada uno asociado a una matriz de covarianza Ξ_{ci} que caracteriza la incertidumbre con que se obtuvo el dato. Los datos pueden provenir de un número indeterminado N de fuentes de información; se considera que

solamente un subconjunto de los datos es producido realmente por las fuentes, los demás son datos apócrifos y el objetivo es caracterizar todas las fuentes de información. Cuando se logra establecer una relación de un dato a la fuente de información, se asume que el dato es una representación insesgada de la fuente.

En el campo de visión artificial se han propuesto varios esquemas para manejar una o varias fuentes de información, sin embargo, estos métodos solo resuelven parte de las tareas de fusión de datos. Las tareas que están aún sin resolver son determinar el número de fuentes de información y tomar en cuenta la incertidumbre de adquisición de datos. El algoritmo propuesto en [Chen04b] se considera robusto pues resuelve los siguientes problemas relacionados con la fusión de información

1. Determinar el número de fuentes de información.
2. Incorporar la incertidumbre de adquisición de datos al proceso de fusión de datos.
3. Los datos apócrifos no son tomados en cuenta.

3.4.3. Esquema de Fusión de Información

Se considera un punto tridimensional proyectado con los parámetros reales de la cámara como la fuente de información. Las demás proyecciones, originadas por los parámetros calculados a partir de el apareamiento probabilístico descrito en el apartado 3.3.1 son observaciones con cierto grado de confiabilidad de la misma fuente. Se asume que cuando la certidumbre de proyección es alta la proyección real estará dentro de la región de incertidumbre del punto observado. Bajo este supuesto se establece que la moda de la distribución de probabilidad generada por las distintas observaciones de la fuente de información debe ser la fuente real. Así pues se utiliza estimación no paramétrica basada en núcleos como estrategia de fusión de datos. El establecer la incertidumbre de proyección como *"peso"* de los kernels al tiempo de estimar la moda se crea un comportamiento robusto en el algoritmo.

Cuando todos los datos $\mathbf{\Phi}_i \in \mathbb{R}^p \, i = 1, ..., n$ provienen de una sola fuente la distribución de probabilidad se caracteriza por tener una sola moda. Se asume que cada uno de

los datos contiene p variables aleatorias y que la incertidumbre con que se adquirió está representada por una matriz de covarianza Ξ_{ci}. El centro del cúmulo de datos puede obtenerse minimizando la suma de distancias al cuadrado y pesando estas por la inversa de la matriz de covarianza (distancia de Mahalanobis [Wand94]) para que las distancias más alejadas con mayor incertidumbre de adquisición impacten menos en el mínimo resultante.

$$\hat{\phi} = \text{argmin}_{\phi} \sum_{i=1}^{n} (\phi - \mathbf{\Phi}_i)^{\top} \Xi_{c_i}^{-1} (\phi - \mathbf{\Phi}_i) \tag{3.60}$$

derivando la expresión anterior (ecuación 3.60)

$$\begin{aligned}
\frac{\partial}{\partial \phi}\hat{\phi} &= \frac{\partial}{\partial \phi} \sum_{i=1}^{n} (\phi - \mathbf{\Phi}_i)^{\top} \Xi_{c_i}^{-1} (\phi - \mathbf{\Phi}_i) \\
&= \sum_{i=1}^{n} \left[\frac{\partial (\phi - \mathbf{\Phi}_i)^{\top}}{\partial \phi} \Xi_{c_i}^{-1} (\phi - \mathbf{\Phi}_i) + (\phi - \mathbf{\Phi}_i)^{\top} \frac{\partial \Xi_{c_i}^{-1} (\phi - \mathbf{\Phi}_i)}{\partial \phi} \right] \\
&= \sum_{i=1}^{n} \left[\left(\frac{\partial (\phi - \mathbf{\Phi}_i)}{\partial \phi} \right)^{\top} \Xi_{c_i}^{-1} (\phi - \mathbf{\Phi}_i) + (\phi - \mathbf{\Phi}_i)^{\top} \frac{\partial \Xi_{c_i}^{-1} (\phi - \mathbf{\Phi}_i)}{\partial \phi} \right] \\
&= \sum_{i=1}^{n} \left[\mathbf{I}^{\top} \Xi_{c_i}^{-1} (\phi - \mathbf{\Phi}_i) + (\phi - \mathbf{\Phi}_i)^{\top} \Xi_{c_i}^{-1} \mathbf{I} \right] \\
&= \sum_{i=1}^{n} \left[\Xi_{c_i}^{-1} (\phi - \mathbf{\Phi}_i) + \Xi_{c_i}^{-1} (\phi - \phi_i) \right] \\
&= 2 \sum_{i=1}^{n} \Xi_{c_i}^{-1} (\phi - \mathbf{\Phi}_i)
\end{aligned} \tag{3.61}$$

igualando a cero para encontrar el mínimo

$$2 \sum_{i=1}^{n} \Xi_{c_i}^{-1} (\mathbf{x} - \mathbf{\Phi}_i) = 0 \tag{3.62}$$

$$\sum_{i=1}^{n} \Xi_{c_i}^{-1} \mathbf{x} - \sum_{i=1}^{n} \Xi_{c_i}^{-1} \mathbf{\Phi}_i = 0 \tag{3.63}$$

$$\mathbf{x} \sum_{i=1}^{n} \Xi_{c_i}^{-1} = \sum_{i=1}^{n} \Xi_{c_i}^{-1} \mathbf{\Phi}_i \tag{3.64}$$

$$\mathbf{x} = \left[\sum_{i=1}^{n} \Xi_{c_i}^{-1} \right]^{-1} \sum_{i=1}^{n} \Xi_{c_i}^{-1} \mathbf{\Phi}_i \tag{3.65}$$

Si bien utilizar la inversa de la incertidumbre como peso para encontrar el centro del cúmulo brinda cierta protección ante ciertos datos apócrifos, esto no sirve si se encuentra

un valor atípica cuya incertidumbre sea pequeña.

El caso general es cuando se tienen múltiples fuentes de información y el número de estas es desconocido. Una observación puede provenir de $N < n$, o bien ser una observación atípica sin relación alguna a las fuentes de información. Ahora la distribución de probabilidad será multimodal, una moda se definirse como el máximo local de la función y puede obtenerse buscando los ceros de su gradiente. El libro [Wand94] es un tratado de estimación no paramétrica basada en núcleos de la función de densidad de probabilidad. La técnica empleada para la fusión de información se basa en estos conceptos. La forma general para un estimador basado en núcleos de la función de densidad p-dimensional δ es

$$\hat{f}\left(\phi;\mathrm{H}\right)=\frac{1}{n}\sum_{i=1}^{n}K_{\mathrm{H}}\left(\phi-\mathbf{\Phi}_i\right) \tag{3.66}$$

donde $\mathbf{\Phi}$ son las observaciones de las variables aleatorias, H_{Ξ} es una matriz simétrica definida positiva de $p\times p$ que cumple la función de establecer el ancho de banda para el estimador,

$$K_{H_{\Xi}}\left(\phi\right)=\left|H_{\Xi}\right|^{-\frac{1}{2}}K\left(H_{\Xi}{}^{-\frac{1}{2}}\phi\right) \tag{3.67}$$

K es el núcleo utilizado del estimador. Un paso crítico es la elección de la matriz H_{Ξ} y en este algoritmo también asegura el comportamiento robusto ante datos atípicos

$$H_{\Xi i}=\chi^2_{\gamma,p}\Xi_{ci} \tag{3.68}$$

donde $\chi^2_{\gamma,p}$ es el valor ji cuadrada con p grados de libertad y un nivel de confianza γ. Así pues, cuando la medición $\mathbf{\Phi}_i$ proviene de la fuente ϕ_k la región elipsoidal en $\mathbb{R}^p$ centrada en $\mathbf{\Phi}_i$ contiene a $\hat{\phi}_k$ con probabilidad de γ. Con esto el estimador para la densidad de probabilidad será

$$\hat{\delta}\left(\phi\right)=\frac{c_{p,k}}{n\left(\chi^2_{\gamma,p}\right)^{p/2}}\sum_{i=1}^{n}\left|\Xi_{ci}\right|^{-\frac{1}{2}}k\left(\frac{1}{\chi^2_{\gamma,p}}\left(\phi-\mathbf{\Phi}_i\right)^{\top}\Xi_{c_i}{}^{-1}\left(\phi-\mathbf{\Phi}_i\right)\right) \tag{3.69}$$

Al igual que en el caso de fuente de información única se busca el centro de cada

cúmulo, sin embargo, ahora se hace uso de la función de densidad por lo que se buscan los máximos, las modas, de dicha función. Haciendo uso del gradiente de la función de densidad

$$\nabla\hat{\delta}(\phi) = \frac{c_{p,k}}{n\left(\chi^2_{\gamma,p}\right)^{p/2}} \sum_{i=1}^{n} |\Xi_{ci}|^{-\frac{1}{2}} \frac{2}{\chi^2_{\gamma,p}} \Xi_{ci}^{-1}(\phi - \mathbf{\Phi}_i)\, k\prime\left(\frac{1}{\chi^2_{\gamma,p}}(\phi - \mathbf{\Phi}_i)^\top \Xi_{ci}^{-1}(\phi - \mathbf{\Phi}_i)\right) \quad (3.70)$$

$$= \frac{c_{p,k}}{n\left(\chi^2_{\gamma,p}\right)^{p/2+1}} \sum_{i=1}^{n} |\Xi_{ci}|^{-\frac{1}{2}} \Xi_{ci}^{-1}(\phi - \mathbf{\Phi}_i)\, k\prime\left(\frac{1}{\chi^2_{\gamma,p}}(\phi - \mathbf{\Phi}_i)^\top \Xi_{ci}^{-1}(\phi - \mathbf{\Phi}_i)\right) \quad (3.71)$$

Sea $g(x) = -k\prime(x)$, esta función sigue cumpliendo con las características necesarias para servir como perfil de un núcleo pues $k(x)$ es monotónicamente decreciente en el intervalo $0 \leq x \leq 1$, por lo que $k\prime(x) \geq 0$ para el mismo lapso. Definase también la matriz $\mathbf{W}_i = |\Xi_{ci}|^{\frac{1}{2}}\Xi_{ci}$ y la función

$$I_\gamma(\mathbf{x}) = g\left(\frac{\phi^\top \Xi_{c_\phi}^{-1}\phi}{\chi^2_{\gamma,p}}\right)$$

$$\begin{aligned}
\nabla\hat{\delta}(\phi) &= \frac{c_{p,k}}{n\left(\chi^2_{\gamma,p}\right)^{p/2+1}} \sum_{i=1}^{n} -\mathrm{W}_i^{-1}(\phi - \mathbf{\Phi}_i)\, I_\gamma(\phi - \mathbf{\Phi}_i) \\
&= \frac{c_{p,k}}{n\left(\chi^2_{\gamma,p}\right)^{p/2+1}} \sum_{i=1}^{n} \left[I_\gamma(\phi - \mathbf{\Phi}_i)\, \mathrm{W}_i^{-1}\mathbf{\Phi}_i - I_\gamma(\phi - \mathbf{\Phi}_i)\, \mathrm{W}_i^{-1}\phi\right] \\
&= \frac{c_{p,k}}{n\left(\chi^2_{\gamma,p}\right)^{p/2+1}} \left[\sum_{i=1}^{n} I_\gamma(\phi - \mathbf{\Phi}_i)\, \mathrm{W}_i^{-1}\mathbf{\Phi}_i - \sum_{i=1}^{n} I_\gamma(\phi - \mathbf{\Phi}_i)\, \mathrm{W}_i^{-1}\phi\right] \quad (3.72)
\end{aligned}$$

Las modas deben estar en los máximos de la función, por lo que

$$\phi = \left(\sum_{i=1}^{n} I_\gamma(\phi - \mathbf{\Phi}_i)\, \mathrm{W}_i^{-1}\right)^{-1} \left(\sum_{i=1}^{n} I_\gamma(\phi - \mathbf{\Phi}_i)\, \mathrm{W}_i^{-1}\mathbf{\Phi}_i\right) \quad (3.73)$$

La ecuación 3.73 es similar a la obtenida para el caso de una fuente de información (ecuación 3.65). La gran diferencia, la que le brinda el comportamiento robusto al estimador, es que se incluye la función $I_\gamma(\phi)$ y esta toma valores distintos de cero únicamente en la región de confianza de ϕ; de esta manera las estimaciones se realizan solamente de forma local y cada una de las modas puede conocerse de forma separada. Cuando se utiliza el núcleo de Epanechnikov la función $I_\gamma(\phi)$ tiene la forma

$$I_\gamma(\phi) = \begin{cases} 1 & \phi^\top \Xi_{c_\phi}^{-1} \phi \leq \chi^2_{\gamma,p} \\ 0 & \phi^\top \Xi_{c_\phi}^{-1} \phi > \chi^2_{\gamma,p} \end{cases} \tag{3.74}$$

3.4.4. Desplazamiento de la Media (Mean Shift)

La expresión 3.73 calcula los máximos de la distribución de probabilidad y para resolverla se emplea un método iterativo conocido como *Mean Shift*. Este procedimiento fue reportado por primera vez en [Fukunaga75] y se utiliza para encontrar la moda de una distribución de probabilidad desconocida. En el contexto del algoritmo presentado en esta sección se utiliza para realizar la fusión de los datos. Originalmente el procedimiento utilizaba ventanas de tamaño uniforme, mientras que la versión analizada en este trabajo se calcula el tamaño de la ventana proporcionalmente a la incertidumbre de la obtención del dato a analizar, esto con el fin de agregar un comportamiento robusto al algoritmo. La incertidumbre de proyección de un punto tridimensional con una hipótesis de apareamiento se consdiera proporcional a la incertidumbre de la misma hipótesis.

Se elige un punto de inicio, se calcula la media ponderada de los datos que contenga la ventana centrada el punto elegido. Se traza un vector de desplazamiento del punto anterior a la media, para posteriormente desplazar el centro de la ventana utilizando el vector recién calculado. Estos pasos se repiten con lo que se debe encontrar una de las n modas cuando el vector de desplazamiento se considere invariante.

El cálculo de la media ponderada tiene una relación directa con la expresión 3.73, de tal suerte que la obtención del centro para la nueva ventana puede darse mediante

$$\mathbf{\Phi}_{k+1} = \left(\sum_{i=1}^{n} I_\gamma \left(\mathbf{\Phi}_k - \mathbf{\Phi}_i \right) \mathrm{W}_i^{-1} \right)^{-1} \left(\sum_{i=1}^{n} I_\gamma \left(\mathbf{\Phi}_k - \mathbf{\Phi}_i \right) \mathrm{W}_i^{-1} \mathbf{\Phi}_i \right) \tag{3.75}$$

con lo que el comportamiento robusto se mantiene gracias a la función I, que restringe los datos a analizar a los que se encuentran dentro de la ventana de confianza; y la matriz W_i, que sopesa el aporte de cada dato de acuerdo a la incertidumbre de su adquisición. Una desventaja del método es que se basa únicamente en el gradiente de la densidad para encontrar máximos locales, lo que introduce el problema de estancamiento en mínimos. Para

Algoritmo 6 Mean Shift

```
MeanShift(puntos, Tolerancia)
1  media ← puntos[x]
2  mientras norma(v) < Tolerancia
3    v ← calcularVectorDesplazamiento(media, puntos)
4    media ← desplazaVentana(media, v)
5    moda ← media
6  RETURN moda
7
```

superar estas barreras se debe perturbar el punto hasta ahora considerado como una moda y reiniciar el procedimiento. Si nuevamente se llega al mismo punto de convergencia esto es un indicador de que realmente se encontró una moda, de lo contrario el método debe encontrar otro candidato a máximo local.

El punto de convergencia del algoritmo 6 representan la moda de la distribución de probabilidad oculta, este procedimiento puede llevar a obtener n distintas modas, dependiendo de donde inicie. Esto porque se van calculando las medias locales y desplazando su ventana de influencia. Los puntos iniciales que convergen a cada moda detectada se consideran como pertenecientes al área de atracción de dicha moda. Una ventaja de este procedimiento es que las zonas de atracción pueden tener forma aleatoria pues cada punto se procesa por separado.

3.4.5. Descripción del Algoritmo

El algoritmo utiliza una especie de reducción del espacio de búsqueda para encontrar los parámetros de proyección adecuados y a partir de ellos las correspondencias entre el modelo y la imagen. El primer paso que se realiza es la generación de triadas en la imagen y en el modelo, de estas se obtienen tanto el ángulo entre el par de líneas que forman y el radio de las magnitudes de dichas líneas. El siguiente paso será calcular la probabilidad de que un

par ángulo-radio del modelo sea proyectado a un par ángulo-radio en la imagen, a diferencia de la aproximación tomada en la sección 3.3 se crea una tabla de búsqueda log(radio) vs. ángulo para evitar calcular cada una de las probabilidades [Olson93, Shimshoni00b].

Una vez generada la información de probabilidad se procede a tomar k puntos de la envolvente convexa del modelo tridimensional. Cada uno de estos se proyecta con los modelos generados mediante el apareamiento probabilístico de tripletas anteriormente mencionado. Además de generar las imágenes bidimensionales de los puntos del modelo se obtiene su región de incertidumbre. En otros algoritmos donde se generan cúmulos de poses en espacios de seis dimensiones, este algoritmo lo hace en 2D, y esto lo hace bajo el supuesto que la incertidumbre de la pose es proporcional a la incertidumbre de la proyección. Con estos datos se procede a iniciar la rutina para encontrar la moda y su región de atracción. Las hipótesis de proyección que estén dentro de la región de atracción son consideradas correctas y para generar un mejor conocimiento se consideran realmente correctas las hipótesis que se encuentren en todas las k zonas de atracción.

Finalmente se realiza una búsqueda del estilo RANSAC pero guiando la adquisición de las muestras de acuerdo a la probabilidad de apareamiento. De este último paso se desprenden tanto los parámetros extrínsecos de la cámara como las correspondencias entre modelo e imagen, concluyendo así el proceso del registro.

3.5. Conclusiones

Se presentó el problema del registro multimodal, es decir, de datos bidimensionales pertenecientes a una imagen con datos tridimensionales pertenecientes a un modelo. Se establecieron los límites de las transformaciones a tomar dado el modelo de proyección en perspectiva débil. Se introduce, también, un par de metodologías para calcular los parámetros extrínsecos de la cámara.

Finalmente, se presentaron tres distintas aproximaciones para resolver el registro de modelo a imagen. La primera basada en minimización de una función de costos, la segunda una busqueda tipo RANSAC con muestreo ordenado y la tercera una técnica basada en la fusión robusta de información. Esta última se caracteriza porque en sus pasos se acota

Algoritmo 7 Registro Basado en Fusión Robusta de Información

```
FusionRegistration(Datos2d, Datos3d)
 1  triadas ← GeneraTriadas(Datos2d, Datos3d)
 2  tablas ← calculaTablas(triadas)
 3  ligaTriadasTablas(triadas, tablas)
 4  referencias ← aleatorios(convexHull(Datos3d), k)
 5  para (cada ref_i en referencias)
 6    proys ← ProyectaMasIncertidumbre(tablas, triadas, ref_i)
 7    [modas_i, zonasAtraccion_i] ← MeanShift(proys)
 8
 9  hipotesis = Interseccion(triadas, modas, zonasAtraccion)
10  [parametros, correspondencias] ← gRANSAC(Datos2d, Datos3d, hipotesis)
11  RETURN[parametros, correspondencias]
12
```

la cantidad de datos a analizar en la siguiente etapa del algorítmo. En el siguiente capítulo se presentan experimentos y resultados realizados con estas técnicas.

Capítulo 4

Experimentos y Resultados

En este capítulo se da cuenta de los experimentos realizados para evaluar el desempeño tanto de algunos de los componentes de los algoritmos revisados en el capítulo 3, como de dichos procedimientos en su implementación final. En este trabajo se considera un modelo tridimensional a un conjunto de puntos en $\mathbb{R}^3$ y sus proyecciones en $\mathbb{R}^2$ son manejadas como puntos de imagen. En algunas ocasiones se utilizaran líneas que unen tanto los puntos en 3D como en 2D, esto con el fin de mejorar la visualización de los resultados. Los experimentos se realizaron en una computadora con procesador *Pentium D* de doble núcleo a 3.00 GHz, 2 GBytes de memoria RAM, disco duro SATA II y utilizando Linux como sistema operativo.

4.1. Pose a Partir de Correspondencias

Los algoritmos revisados en los apartados 3.1.1 y 3.1.2 calculan los parámetros proyección asumiendo que las correspondencias son previamente conocidas, es decir, se hace un registro no automático.

4.1.1. POSIT

Se realizaron experimentos tanto con datos sintéticos como reales, en las pruebas hechas el algoritmo converge en pocas iteraciones ($4 \sim 6$) y se obtienen resultados favorables en las condiciones establecidas; como son que el objeto se encuentre a suficiente distancia de la cámara para que la perspectiva débil sea aproximable a la perspectiva completa, que

Coordenadas	P_0	P_1	P_2	P_3	P_4	P_5	P_6	P_7
x_{3D}	0	10	10	0	0	10	10	0
y_{3D}	0	0	10	10	0	0	10	10
z_{3D}	0	0	0	0	10	10	10	10
u_{2D}	0.81818	1.50000	1.29232	0.70490	0.97485	1.78722	1.50000	0.81818
v_{2D}	2.52273	2.52273	1.75808	1.75808	2.43134	2.43134	1.55849	1.55849

Tabla 4.1: Datos de entrada para evaluar POSIT

el objeto se localice prácticamente sobre el eje óptico de la cámara, que los puntos no sean coplanares en el espacio tridimensional y que se cuente con cuatro o más correspondencias.

En uno de los experimentos con información sintética se plantea recuperar la matriz de proyección a partir de las correspondencias entre un modelo tridimensional y sus proyecciones en perspectiva débil. La tabla 4.1 muestra los datos utilizados para realizar la prueba, las columnas indican el número de punto tridimensional P_i y las primeras tres filas (x_{3D}, y_{3D}, z_{3D}) son las coordenadas en tres dimensiones; las ultimas dos filas (u_{2D}, v_{2D}) son las coordenadas de la proyección del punto P_i en perspectiva completa mediante la matriz especificada en la ecuación 4.2.

La distancia focal fué elegida al azar, y para la cámara en el ejemplo fué $f = 3$ por lo que la matriz de transformación utilizada fue

$$
\begin{aligned}
M &= K[R|\mathbf{T}] \qquad (4.1)\\
M &= \left[\begin{array}{ccc} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{array}\right] \left[\begin{array}{c|c} \mathbf{R}_1^\top & T_x \\ \mathbf{R}_2^\top & T_y \\ \mathbf{R}_3^\top & T_z \end{array}\right] \\
M &= \left[\begin{array}{ccc} 3 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 1 \end{array}\right] \left[\begin{array}{ccc|c} 1.0 & 0.0 & 0.0 & 12.0 \\ 0.0 & -0.70711 & -0.70711 & 37.0 \\ 0.0 & 0.70711 & -0.70711 & 44.0 \end{array}\right] \\
&= \left[\begin{array}{cccc} 3.0 & 0.0 & 0.0 & 36.0 \\ 0.0 & -2.12132 & -2.12132 & 111.0 \\ 0.0 & 0.70711 & -0.70711 & 44.0 \end{array}\right] \qquad (4.2)
\end{aligned}
$$

y la matriz calculada a partir del procedimiento fue

$$\widehat{M} = \begin{bmatrix} 1.0 & -2.0258e-13 & 2.0142e-13 & 12.0 \\ -8.4687e-16 & -0.70711 & -0.70711 & 37.0 \\ 2.8567e-13 & 0.70711 & -0.70711 & 44.0 \end{bmatrix}$$

que es prácticamente la misma, salvo por la inclusión de los efectos de la distancia focal en la matriz original, y esto se comprueba en la figura 4.1 donde los puntos proyectados por ambas matrices se superponen.

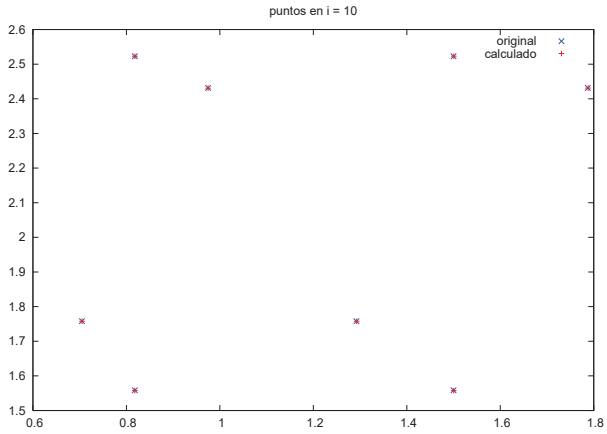


Figura 4.1: Puntos proyectados matriz original y recuperada

El suma de las distancias entre parámetros de la matriz calculada y la original fué de $2.65482e^{-4}$, la suma de las distancias entre puntos proyectados con la matriz calculada y la original fué de $1.81818e^{-5}$. Se realizaron pruebas para conocer empíricamente a que distancia la perspectiva débil era comparable con la perspectiva completa, encontrándose que esto se presenta cuando la cámara se encuentra a una distancia superior a 6 veces el tamaño del objeto. Los puntos de la imagen son "detectados" manualmente y los correspondientes tridimensionales son medidas realizadas al objeto de estudio. La figura 4.2 es una fotografía de un ortoedro construido a partir de bloques de *Lego*, se ilustra la proyección de puntos a partir los parámetros externos calculados mediante POSIT. Los puntos en verde son las proyecciones y los rojos los detectados en la imagen.

La ventaja principal de este método es que se calculan directamente los parámetros extrínsecos de la cámara y se hace con buena precisión. Una de las desventajas es que es un método iterativo, aun cuando se realizan pocos ciclos para obtener la pose, otra limitante

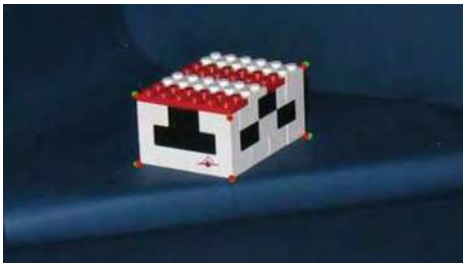


Figura 4.2: Puntos detectados (rojo) vs. puntos calculados (verde)

es que se requiere un mínimo de cuatro correspondencias para calcular los parámetros.

4.1.2. Estimación de Parámetros de Alter

Las pruebas realizadas para evaluar el desempeño de este procedimiento se realizaron únicamente con datos sintéticos, para llevar a cabo los experimentos se tomaron en cuenta las mismas restricciones que en el caso del POSIT. En uno de los experimentos se modeló un cubo y se proyectó con perspectiva débil, utilizando tres correspondencias se estimaron los parámetros como se indica en el apartado 3.1.2. La figura 4.3 muestra el cubo tal como se proyectó (con perspectiva débil) en color rojo, los puntos correspondientes en verde, las proyecciones calculadas mediante Alter son los cubos en cyan y azúl, las unidades son en pixeles.

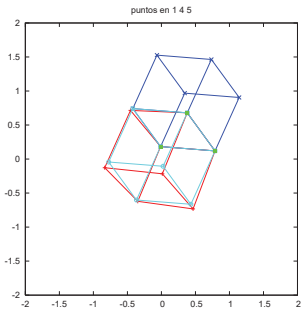


Figura 4.3: Proyección utilizando los parámetros de Alter

Al calcular los parámetros de proyección de Alter se tiene la incertidumbre de un signo, por lo que corresponde al usuario buscar el adecuado, esto se observa en la figura 4.3 el

Modelo	0.81818	1.50000	1.29232	0.70490	0.97485	1.78722	1.50000	0.81818
	2.52273	2.52273	1.75807	1.75807	2.43134	2.43134	1.55849	1.55849
Alter	0.76107	1.50000	1.44383	0.70490	1.04829	1.78722	1.73105	0.99212
	2.54719	2.52273	1.73361	1.75807	2.45581	2.43134	1.64222	1.66668
Distancia	0.05711	0.00000	0.15151	0.00000	0.07344	0.00000	0.23105	0.17394

Tabla 4.2: Comparación de Proyección Modelo vs. Alter

cubo azúl representa una mala elección del signo de los parámetros, mientras que el cubo en cyan la adecuada. Si bien la estimación se hace con pocas operaciones matemáticas y con tan solo tres correspondencias, las proyecciones de puntos adicionales son, en ocasiones, lejanas a las esperadas. Utilizando los datos mostrados en la tabla 4.1 se calcularon parámetros de proyección utilizando distintas triadas de puntos correspondientes, los puntos proyectados con el modelo de cámara de perspectiva débil y los proyectados mediante los parámetros de alter se muestran en la tabla 4.1.2. La distancia entre los puntos se muestra en el tercer renglon, la suma de estas distancias fué de 0.95237. Estos errores variaron al elegir distintas triadas correspondientes, sin embargo, el máximo de estos no fué mayor a 2.

Utilizando una fotografía del cuboide mencionado en el apartado 4.1.1 se seleccionaron tres correspondencias aleatorias y se calcularon los parámetros de proyección. La figura 4.4 es un fragmento de la fotografía con los puntos detectados indicados mediante cruces azules y las proyecciones marcadas por cruces verdes. Se observa que algunas proyecciones no estan del todo cercanas a los puntos detectados.

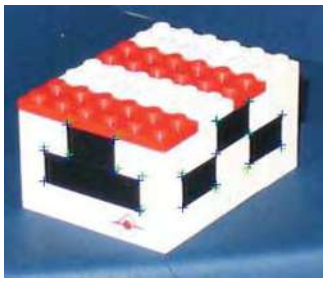


Figura 4.4: Puntos detectados (azul) vs. puntos calculados (verde)

4.2. Registro Automático

A continuación se presentan resultados de algoritmos completamente automáticos para estimar correspondencias y calcular los parámetros de proyección a partir de un conjunto de puntos tridimensionales y sus proyecciones en dos dimensiones.

4.2.1. SoftPOSIT

Los experimentos realizados para evaluar el desempeño del algoritmo se realizaron tanto con datos reales como sintéticos. Primeramente se evaluó la funcionalidad del procedimiento fijando las correspondencias y después el estimado inicial de la matriz de proyección se igualaba a la matriz original. En ambos casos el algoritmo funcionó de acuerdo a lo esperado, es decir la matriz calculada terminó siendo igual a la original y se encontraron correspondencias correctas.

Posteriormente se evaluaron los efectos de los parámetros del recocido simulado. Se observó una gran dependencia entre estos y las estimaciones iniciales de la matriz de transformación. En general si no se tiene conocimiento alguno de la matriz de proyección se recomienda utilizar valores pequeños para las variables, esto es β inicial debe ser aproximadamente 0.003 y su actualización $\gamma \approx 1.05$ y mantener el error estimado de detección de características $alpha \approx 0.0001$. Cuando se conoce razonablemente bien la matriz de transformación los valores de β pueden incrementar de acuerdo a la certidumbre de los valores conocidos, es decir, si se esta completamente seguro β incluso podría llegar a valores muy altos, por ejemplo 100.

Los resultados de los experimentos se tornan erráticos a medida que se varía el estimado inicial de la matriz de proyección. En una de las pruebas la más ligera modificación al vector de traslación originaba que el algoritmo convergiera a una matriz y correspondencias erróneas. Tras profundizar en la investigación se descartó la real funcionalidad del procedimiento por su comportamiento inestable. Utilizando los puntos de ejemplo del apartado 4.1.1 presentados en la tabla 4.1 no se logró convergencia tras explorar 30 matrices de rotación iniciales, incluida la buscada.

La figura 4.5 se muestran las proyecciones iniciales en color verde, las proyecciones

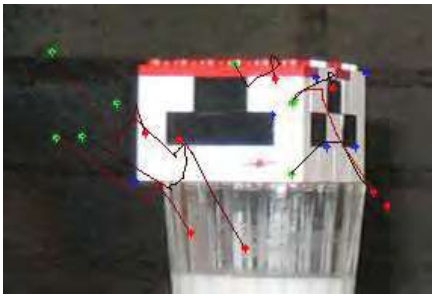


Figura 4.5: Terminación errónea de SoftPOSIT con datos reales (inicio → verde, final → rojo)

finales en rojo y el "camino" que siguió el algoritmo en líneas cafés. Los puntos detectados en la imagen se muestran en color azul, se utilizó un modelo del objeto truncado, es decir, se forzó a que hubiera cero oclusión y cero ruido.

En [David03] no se hace mención de la necesidad de insertar este algoritmo en un esquema de búsqueda con criterios de terminación probabilísticos, sin embargo, esto se hace necesario para compensar la falta de convergencia del algoritmo y Daniel DeMenthon lo reporta en [DeMenthon01]. Un dato curioso es que en algunas de las pruebas se obtenían hasta un 80 % de las correspondencias acertadas y aun así la matriz de transformación presentaba grandes diferencias, esto porque las correspondencias incorrectas tienen un cierto peso al calcular la matriz.

4.2.2. gRANSAC

En las pruebas realizadas se utilizaron únicamente puntos generados sinteticamente, esto con la finalidad de evaluar el desempeño ante ruido y oclusión de datos. Los puntos del modelo tridimensional utilizados son creados aleatoriamente en un espacio $\mathbb{R}^3$ con límites $[-100:100, -100:100, -100:100]$ y matrices de proyección se componen de una matriz de rotación aleatoria diseñada a partir de ángulos de Euler con acotamientos de 0 a 2π y un vector de traslación T cuya norma se mantuviese $\|T\| \geq 800$ para garantizar que la proyección en perspectiva débil fuese equivalente a la perspectiva completa.

Para comprobar la efectividad del algoritmo se generaron conjuntos de 10 datos

tanto tridimensionales como sus imágenes, o bien no se agregó ruido ni oclusión a la *"fotografía"*. Inicialmente se manejó una selección aleatoria con lo que la cantidad de iteraciones del algoritmo variaba significativamente, posteriormente se ordenó el muestreo y la eficiencia del algoritmo incrementó notablemente. En pruebas realizadas para medir el mínimo de iteraciones necesarias para encontrar las correspondencias correctas, en un promedio de 10 iteraciones se logra alcanzar el objetivo.

Se generaron conjuntos con 3, 4, 5 ... 20 puntos en dos y tres dimensiones, y se utilizaron para verificar el tiempo de generación de hipótesis entre conjuntos bi y tridimensionales. Se generan triadas bidimensionales y tridimensionales, posteriormente para cada triada en dos dimensiones se crean listas incluyendo cada triada en tres dimensiones calculando la probabilidad de ser pareja y ordenandolas de acuerdo a esta medida. La figura 4.6 muestra los resultados, se observa una progresión exponencial del tiempo, esto impidió realizar pruebas con mayor número de puntos. El crecimiento exponencial se debe a que, para N puntos 2D y M puntos 3D, se tienen $\binom{N}{3}\binom{M}{3}$ cálculos de probabilidades de apareamiento.

Para las siguientes mediciones se realizaron pruebas con conjuntos sintéticos de 10, 12 ... 30 puntos en tres dimensiones y proyectandolos con una matriz de proyección en perspectiva débil aleatoria. Se hicieron pruebas de registro con cinco mil iteraciones de gRANSAC. Inicialmente los conjuntos no contenían ni ruido ni oclusion, posteriormente se introdujo ruido del 10, 20, 30, 40 y 50 por ciento, y eliminaron datos verídicos en un 10, 20, 30, 40 y 50 por ciento para simular la oclusión de puntos en la imagen. Se generaron 50 conjuntos para cada condición y se utilizaron para evaluar distintos aspectos del funcionamiento del algoritmo. La figura 4.7(a) ilustra los resultados para los conjuntos con 10 puntos tridimensionales y los varios niveles de oclusión y ruido. El color rojo significa que se obtuvo cero por ciento de correspondencias correctas y el azul que se obtuvieron todas. La tabla 4.3 contiene los datos para el experimento.

Realizar experimentos con más de 12 puntos 2D/3D fué prácticamente imposible pues los tiempos de generación de listas de probabilidad son extremadamente largos. Así pues se da un cambio en la forma de almacenar la probabilidad de apareamiento de

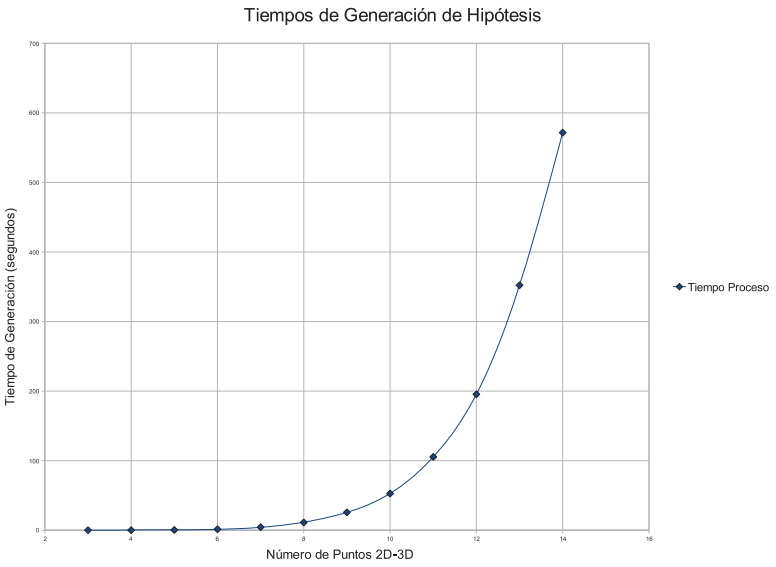
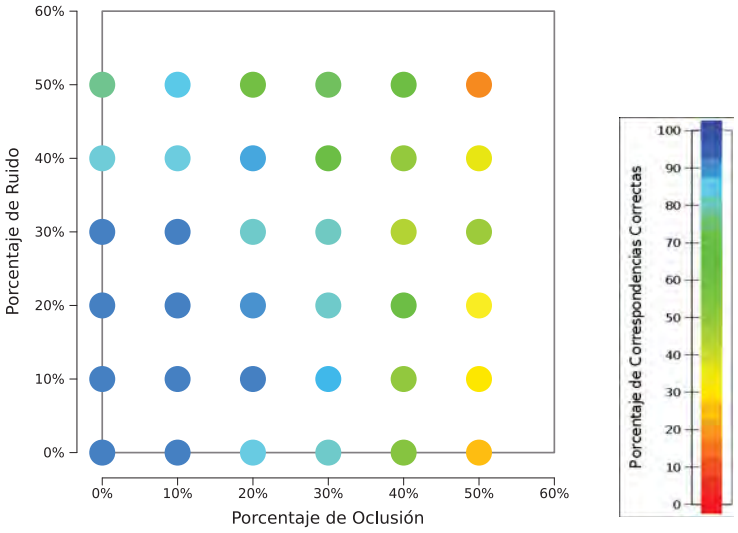


Figura 4.6: Tiempos de Generación de Hipótesis

triadas, por lo que se recurrió a una estructura similar a la propuesta en [Chen04b]. En esta estructura se generan listas con las triadas de dos y tres dimensiones, se genera una tabla de busqueda para los puntos tridimensionales en un indexadas por ángulo y radio de distancias.

La tabla de busqueda contiene 400 entradas correspondientes a 20 divisiones para el ángulo y 20 para el radio de distancias, cada entrada contiene una lista de las triadas de puntos en tres dimensiones cuyos ángulos y radios de distancias correspondan a los límites de la entrada. A cada triada de puntos bidimensionales se les asigna una lista de las entradas de la tabla ordenadas por probabilidad de apareamiento. Los tiempos de generación de listas de probabilidad utilizando tablas de busqueda se ven reflejados en la figura 4.8(a), una comparación de los tiempos de creación de listas se presenta en la figura 4.8(b)

La tabla 4.2.2 resume los resultados para diez puntos 3D con distintos niveles de oclusión y ruido utilizando tablas de búsqueda probabilística. Utilizar tablas de busqueda es una mejor opción pues el tiempo de estimación de probabilidad de apareamiento se ve drasticamente reducido, lo que permite utilizar el algoritmo con mucho mayor número de puntos. Utilizando los datos presentados en la tabla 4.1 se generaron falsos positivos pues



(a) (b)

Figura 4.7: Porcentajes de correspondencias correctas a diferentes grados de oclusión y ruido

se trata de un cubo y la proyección de este puede es identica al observarse desde distintos puntos de vista. Para probar el algoritmo se agregaron un par de puntos salientes que distinguieran las proyecciones y estos se ven reflejados en la tabla 4.2.2.

Con este conjunto de datos se obtuvieron el 100 % de correspondencias y se calculó la matriz de proyección utilizando POSIT, con lo que se recuperó la matriz de proyección. Usando una fotografía del cuboide se tomaron 18 puntos del modelo, los correspondientes a los visibles en la fotografía. Esta fue una prueba sin ruido u oclusión, se encontraron 13 correspondencias correctas y con ellas se calculó la matriz de proyección mediante POSIT y los parámetros de proyección de Alter. La figura 4.9 ilustra los resultados del proceso, las cruces azules simbolizan los puntos detectados y las cruces verdes y rojas las proyecciones con la matriz calculada mediante POSIT y los parámetros de Alter respectivamente.

Posteriormente se evaluó la misma fotografía con el conjunto completo de puntos pertenecientes al modelo tridimensional y 18 puntos detectados en la imagen, es decir, se introdujo un 47 % de oclusión. En esta prueba no se encontraron suficientes correpondencias correctas para calcular los parámetros de proyección de Alter ni la matriz de parámetros

% Oclusión	% Ruido	% Correspondencias Correctas	% Oclusión	% Ruido	% Correspondencias Correctas
0	0	100	30	0	90.13
0	10	100	30	10	96.24
0	20	100	30	20	90.22
0	30	100	30	30	89.47
0	40	91.23	30	40	77.44
0	50	85.38	30	50	81.95
10	0	100	40	0	57.89
10	10	100	40	10	55.26
10	20	100	40	20	78.07
10	30	100	40	30	48.24
10	40	91.81	40	40	54.38
10	50	93.56	40	50	67.54
20	0	92.39	50	0	27.19
20	10	100	50	10	33.68
20	20	98.68	50	20	35.78
20	30	90.13	50	30	52.63
20	40	97.36	50	40	38.94
20	50	65.13	50	50	20

Tabla 4.3: Resultados de correspondencias correctas a diferentes grados de oclusión y ruido para conjuntos de datos de 10 puntos tridimensionales

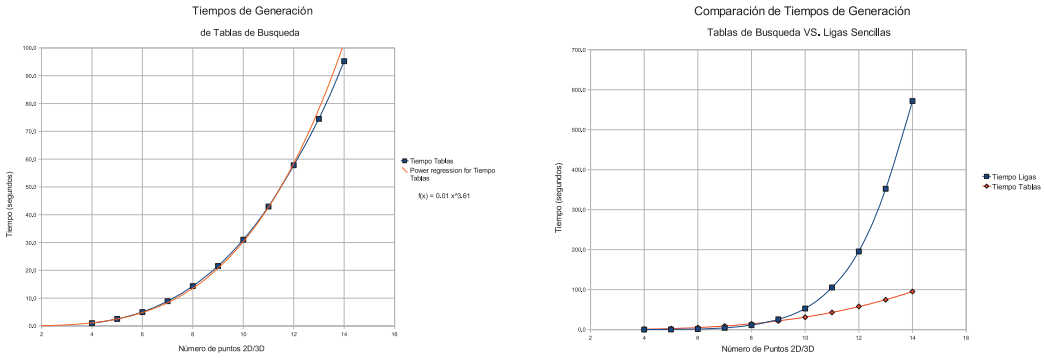
externos mediante POSIT. La figura 4.10 muestra los resultados del proceso.

4.2.3. Fusión Robusta

Se analizó el conjunto de datos presentados en la tabla 4.2.2, se proyectaron mil veces tres puntos pertenecientes a la envolvente convexa del conjunto de puntos tridimensionales. Esta información se analiza mediante el procedimiento Mean-Shift que arroja la moda de la función de densidad oculta y las observaciones pertenecientes a la región de atracción

Tiempo (segundos)	% Oclusion	% Ruido	% Correctas
31.016	0.0	0.0	100.0
73.215	0.0	40.0	90.0
9.244	40.0	0.0	70.0
30.908	40.0	40.0	60.0

Tabla 4.4: Resumen Resultados gRANSAC con Tablas de Búsqueda



(a) Tiempos de Generación Tablas

(b) Comparación Tiempos de Generación

Figura 4.8: Tiempos de Creación de Listas

Coordenadas	P_0	P_1	P_2	P_3	P_4
x_{3D}	1.0	-13.0	0.0	10.0	10.0
y_{3D}	17.0	7.0	0.0	0.0	10.0
z_{3D}	3.0	5.0	0.0	0.0	0.0
u_{2D}	0.72357	-0.06606	0.81818	1.50	1.29232
v_{2D}	1.27224	1.88364	2.52273	2.52273	1.75807
Coordenadas	P_5	P_6	P_7	P_8	P_9
x_{3D}	0.0	0.0	10.0	10.0	0.0
y_{3D}	10.0	0.0	0.0	10.0	10.0
z_{3D}	0.0	10.0	10.0	10.0	10.0
u_{2D}	0.70490	0.97485	1.78722	1.50	0.81818
v_{2D}	1.75807	2.43134	2.43134	1.55849	1.55849

Tabla 4.5: Nuevo Conjunto de Datos

de la fuente de información, es decir, la moda. En este experimento las mil proyecciones iniciales se vieron reducidas a 542 hipótesis, las cuales se utilizaron como entrada para una busqueda mediante gRANSAC. Tras estos pasos se encontraron el 100 % de correspondencias correctas, se utilizó tanto proyección de Alter como POSIT para reproyectar los puntos, se encontraron diferencias iguales a las reportadas en el apartado 4.2.2.

Utilizando la fotografía ilustrada en la figura 4.9 presentada en la sección 4.2.2 y un subconjunto de puntos tridimensionales, correspondientes a los visibles en la imagen. Se realiza un procedimiento similar al comentado en el párrafo anterior, tras "filtrar" las

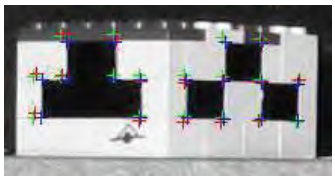


Figura 4.9: Puntos detectados (azul) vs. puntos calculados (verde-POSIT, rojo-Alter)

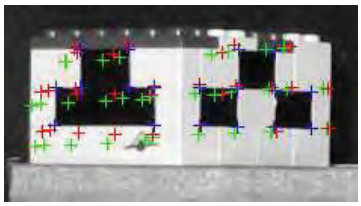


Figura 4.10: Puntos detectados (azul) vs. puntos calculados (verde-POSIT, rojo-Alter)

proyecciones mediante Mean Shift se termina con un total de 852 de un total de mil. Tras evaluar la factibilidad de las hipótesis se encuentran 11 correspondencias de un total de 18, con las que se calculan la matriz de proyección mediante POSIT y los parámetros de Alter con tres de estas. La figura 4.11 muestra los resultados del proceso.

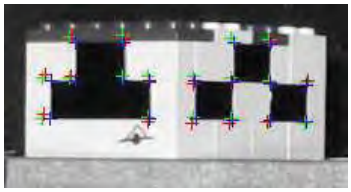


Figura 4.11: Puntos detectados (azul) vs. puntos calculados (verde-POSIT, rojo-Alter)

Posteriormente se analizó el conjunto completo de puntos en tres dimensiones contra los detectados en la imagen, el procedimiento no arrojó correspondencias correctas.

Capítulo 5

Conclusiones

5.1. Conclusiones Generales

Tras la revisión del funcionamiento de los distintos algoritmos se presentan las siguientes conclusiones, en cuanto a algoritmos no automáticos el POSIT presenta gran eficiencia y logra calcular directamente los parámetros externos de la cámara siempre y cuando se la fotografía del objeto entre en la categoría de perspectiva débil. Este método iterativo puede servir como el paso final para conocer los parámetros externos basandose en las correspondencias estimadas con algún otro algoritmo.

La estimación de parámetros de Alter es rápida y presenta una forma de proyectar puntos tridimensionales en la imagen, sin embargo, estos parámetros no se relacionan directamente con los externos de la cámara. La pequeña cantidad de operaciones que se requieren para proyectar lo hacen una gran herramienta para algoritmos del estilo de RANSAC, pues cada iteración es ligera. Los algoritmos automáticos se presentan en apartados individuales por ser el núcleo de este trabajo. Con las pruebas realizadas el método que proporciona mejores y más estables resultados ha sido el algoritmo gRANSAC, sin embargo, el algoritmo basado en fusión robusta es una aproximación bastante prometedora.

5.2. SoftPOSIT

La convergencia del procedimiento como se plantea en el algoritmo 3 no se garantiza, en las pruebas realizadas fué realmente dificil encontrar matrices iniciales para las cuales se resolvieran tanto las correspondencias como la pose.

Para solucionar dicho problema en [DeMenthon01] se propone establecer una búsqueda en un espacio de seis dimensiones, tres ángulos de Euler para la matriz de rotación y tres coordenadas para la traslación. Las búsquedas son terminadas anticipadamente si la probabilidad de encontrar la pose, dadas el número de correspondencias actual y el número de iteraciones, es menor a un umbral determinado. Para conocer la probabilidad se caracteriza una función de probabilidad utilizando datos de correspondencias y pose conocidos.

Buscar en un espacio de seis dimensiones sin acotaciones es una tarea grande, así que se deben establecer reducciones a los margenes de búsqueda para la traslación. El algoritmo termina siendo una búsqueda con criterios de terminación anticipada y gran carga computacional en cada paso. Podría utilizarse en ambientes controlados donde conocer a groso modo la traslación y la rotación no fuese complicado, como en líneas de producción.

5.3. gRANSAC

RANSAC se ha caracterizado por ser el algoritmo robusto más utilizado en visión computacional, además existen multiples optimizaciones que pueden incrustarse al esquema general como los presentados en este trabajo. Estas consisten primariamente en eliminar el aspecto aleatorio de las muestras a evaluar y sustituirlo por un orden probabilistico de acuerdo a las caracteristicas geométricas del modelo tridimensional y su relación con las detectadas en la imagen. El problema generado con esto es que se aumenta el número de cálculos necesarios y, por lo tanto, mayor tiempo de ejecución. Para solventar esto se introducen tablas de busqueda y con estas se reduce notablemente el número de cómputos. Se observó un comportamiento robusto, tal y como se esperaba, con efectividad arriba del 90 % de correspondencias correctas hasta con un 30 % de oclusión y ruido.

Método	Ventajas	Desventajas
SoftPOSIT	Resuelve simultaneamente pose y correspondencias, presenta cierta protección a ruido y oclusión.	Requiere una buena matriz de proyección inicial para convergir, puede alejarse de la solución aun con matrices cercanas a la real.
gRANSAC	Robusto a ruido y oclusión, muestreo guiado.	Se evaluan gran cantidad de hipótesis para garantizar el comportamiento robusto.
Fusión Robusta	Robusto a ruido y oclusión, número de hipótesis reducido.	Sensible a la elección de puntos de la envolvente convexa tridimensional, Mean Shift puede llegar a ser pesado computacionalmente.

Tabla 5.1: Comparativa de Métodos

5.4. Fusión Robusta de Datos

La aproximación de fusión de información es interesante como concepto para tratar de unificar las distintas aproximaciones a este problema. Quizá la más grande aportación de este algoritmo es el concepto de proporcionalidad entre la incertidumbre de proyección y la incertidumbre de la pose. Esto permite realizar cúmulos en dos dimensiones y reducir el número de hipótesis a comprobar mediante un procedimiento tipo RANSAC.

5.5. Comparación

La tabla 5.1 presenta una comparación entre las ventajas y desventajas de cada método revisado en este trabajo para resolver el problema del registro automático de modelo tridimensional a imagen en dos dimensiones.

5.6. Trabajos Futuros

Otra posibilidad para realizar el registro de imagen a modelo es obtener información de la forma del objeto visto en una fotografía para posteriormente registrar estos datos tridimensionales con los del modelo de referencia. En [Robinson04, Ononye02, Kontsevich94] se presentan distintas metodologías para estimar la forma a partir de imágenes con distintas adecuaciones. En [Ng98] se presenta un sistema completo para la reconstrucción de esce-

nas tridimensionales, utiliza múltiples sensores y realiza registro multimodal entre datos de profundidad, color y reflectancia.

Unas de las más nuevas y prometedoras aproximaciones son las que utilizan descriptores robustos como SIFT [Lowe99], SURF [Bay06], LESH [Sarfraz08] o GLOH [Mikolajczyk05]; estos se caracterizan por ser invariantes ante ciertas transformaciones además de reducir la cantidad de datos a evaluar, cabe mencionar que la información individual es mucho más rica que la simple posición y color. En [Forsyth91] se demuestran distintas aplicaciones de visión por computadora con descriptores invariantes, una de las cuales involucra reconocer objetos para posteriormente estimar la pose de los mismos. El libro [Ponce07] revisa múltiples técnicas para el reconocimiento de objetos utilizando descriptores complejos, en particular contiene el trabajo [Gordon07] que plantea insertar objetos en ambientes de realidad aumentada y para ello utiliza registro basado en descriptores robustos.

- Utilizar descriptores complejos para reducir el conjunto de datos a evaluar o establecer mejores medidas de error o relaciones entre caracteristicas.

- Los algoritmos presentados en este trabajo pueden ser fácilmente paralelizados, por lo que las aplicaciones en tiempo real no se descartan.

- Profundizar en el concepto de fusión de información para incluir otras técnicas en el proceso del registro.

- Estudiar la extensión de la esfera de observabilidad, conocida como espacio de observabilidad, para no limitar a perspectiva debil los algoritmos.

Referencias

[Abdel-Aziz71] Abdel-Aziz, Y. I. y Karara, H. M. Direct linear transformation into object space coordinates in close-range photogrammetry. *Symp. Close-Range Photogrammetry*, págs. 1–18, 1971.

[Alter92] Alter, T. D. 3D pose from three corresponding points under weak-perspective projection. Inf. Téc. AIM-1378, 1992.
URL `citeseer.ist.psu.edu/article/alter92pose.html`

[Arie88] Arie, J. B. The properties of viewed angles and distances with application to 3-D object recognition. *En International Conference on Pattern Recognition*, págs. I: 309–312. 1988.
URL `http://dx.doi.org/10.1109/ICPR.1988.28229`

[Bay06] Bay, H., Tuytelaars, T., y Van Gool, L. Surf: Speeded up robust features. págs. 404–417. 2006. doi:10.1007/11744023_32.
URL `http://dx.doi.org/10.1007/11744023_32`

[Ben-Arie90] Ben-Arie, J. The probabilistic peaking effect of viewed angles and distances with application to 3-d object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(8):760–774, 1990. ISSN 0162-8828. doi:http://doi.ieeecomputersociety.org/10.1109/34.57667.

[Bouguet] Bouguet, J.-Y. Matlab calibration toolbox. Internet - http://www.vision.caltech.edu/bouguetj.

[Bridle90] Bridle, J. S. Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. págs. 211–217, 1990.

[Cerny85] Cerny, V. Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of Optimization Theory and Applications*, 45(1):41–51, January 1985. doi:10.1007/BF00940812.
URL `http://dx.doi.org/10.1007/BF00940812`

[Chen04a] Chen, H. *Projection Based Robust Estimators for Computer Vision*. Tesis Doctoral, Graduate School—New Brunswick, Rutgers, The State University of New Jersey, October 2004.

[Chen04b] Chen, H., Shimshoni, I., y Meer, P. Model based object recognition by robust information fusion. *icpr*, 03:57–60, 2004. ISSN 1051-4651. doi: http://doi.ieeecomputersociety.org/10.1109/ICPR.2004.1334468.

[Clarkson01] Clarkson, M. J., Rueckert, D., Hill, D. L., y Hawkes, D. J. Using photo-consistency to register 2d optical images of the human face to a 3d surface model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1266–1280, 2001. ISSN 0162-8828. doi: http://doi.ieeecomputersociety.org/10.1109/34.969117.

[Comaniciu00] Comaniciu, D. I. *Non-Parametric Robust Methods for Computer Vision*. Tesis Doctoral, Graduate School—New Brunswick, Rutgers, The State University of New Jersey, January 2000.

[Coren01] Coren, S., Ward, L. M., y Enns, J. T. *Sensación y Percepción*. McGraw Hill Interamericana, 5ª ed$^{\underline{\text{ón}}}$., 2001.

[David02] David, P., DeMenthon, D., Duraiswami, R., y Samet, H. Softposit: Simultaneous pose and correspondence determination. *En ECCV (3)*, págs. 698–714. 2002.
URL `citeseer.ist.psu.edu/article/david02softposit.html`

[David03] David, P., DeMenthon, D., Duraiswami, R., y Samet, H. Simultaneous pose and correspondence determination using line features. *cvpr*, 02:424, 2003. ISSN 1063-6919. doi: http://doi.ieeecomputersociety.org/10.1109/CVPR.2003.1211499.

[Dementhon95] Dementhon, D. F. y Davis, L. S. Model-based object pose in 25 lines of code. *Int. J. Comput. Vision*, 15(1-2):123–141, 1995. ISSN 0920-5691. doi:http://dx.doi.org/10.1007/BF01450852.

[DeMenthon01] DeMenthon, D., David, P., y Samet, H. Softposit: An algorithm for registration of 3d models to noisy perspective images combining softassign and posit. Inf. téc., mayo 2001.
URL `citeseer.ist.psu.edu/dementhon01softposit.html`

[Dempster77] Dempster, A. P., Laird, N. M., y Rubin, D. B. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977. doi:10.2307/2984875.
URL `http://dx.doi.org/10.2307/2984875`

[Doignon07] Doignon, C. *Scene Reconstruction, Pose Estimation and Tracking*, cap. An Introduction to Model-Based Pose Estimation and 3-D Tracking Techniques, págs. 359–382. I-Tech Education and Publishing, Vienna, Austria, June 2007. Edited by Rustam Stolkin.

[Dorfler04] Dorfler, P. y Schnurr, C. Robust pose estimation for arbitrary objects in complex scenes. págs. 455–462. 2004.

[Fischler81] Fischler, M. A. y Bolles, R. C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981. ISSN 0001-0782. doi: http://doi.acm.org/10.1145/358669.358692.

[Fitzgibbon03] Fitzgibbon, A. W. Robust registration of 2D and 3D point sets. *Image and Vision Computing*, 21(12-13):1145–1153, dic. 2003.
URL `http://www.sciencedirect.com/science/article/B6V09-4B0P1F2-1/2/3`

[Forsyth91] Forsyth, D., Mundy, J., Zisserman, A., Coelho, C., Heller, A., y Rothwell, C. Invariant descriptors for 3d object recognition and pose. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(10):971–991, 1991. ISSN 0162-8828. doi: http://doi.ieeecomputersociety.org/10.1109/34.99233.

[Forsyth03] Forsyth, D. A. y Ponce, J. *Computer Vision: A Modern Approach*. Prentice Hall, 2003.
URL `http://www.cs.berkeley.edu/ daf/book.html`

[Fukunaga75] Fukunaga, K. y Hostetler, L. D. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Trans. Information Theory*, 21(1):32–40, ene. 1975.

[Gold95] Gold, S., Lu, C. P., Rangarajan, A., Pappu, S., y Mjolsness, E. New algorithms for 2D and 3D point matching: Pose estimation and correspondence. *En* G. Tesauro, D. Touretzky, y T. Leen, eds., *Advances in Neural Information Processing Systems*, tomo 7, págs. 957–964. The MIT Press, 1995.
URL `citeseer.ist.psu.edu/gold97new.html`

[Gold96] Gold, S. y Rangarajan, A. A graduated assignment algorithm for graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(4):377–388, 1996.
URL `citeseer.ist.psu.edu/gold96graduated.html`

[Gordon07] Gordon, I. y Lowe, D. G. *Toward Category-Level Object Recognition*, tomo 4170/2006 de *Lecture Notes in Computer Science*, cap. What and Where: 3D Object Recognition with Accurate Pose, págs. 67–82. Springer Berlin / Heidelberg, January 2007. Book Toward Category-Level Object Recognition.

[Grimson91] Grimson, W. E. L. y Huttenlocher, D. P. On the verification of hypothesized matches in model-based recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(12):1201–1213, 1991.
URL `citeseer.ist.psu.edu/grimson89verification.html`

[Hartley03] Hartley, R. y Zisserman, A. *Multiple view geometry in computer vision*. Cambridge University Press, Cambridge, UK, 2ª ed$\underline{\text{ón}}$., 2003.

[Horaud94] Horaud, R., Christy, S., y Dornaika, F. Object pose: The link between weak perspective, paraperspective, and full perspective. *En Technical Report*. 1994.

[Jahne00] Jahne, B. y Haussecker, H., eds. *Computer vision and applications: a guide for students and practitioners*. Academic Press, Inc., Orlando, FL, USA, 2000. ISBN 0-12-379777-2.

[Kirkpatrick83] Kirkpatrick, S., Gelatt, C. D., y Vecchi, M. P. Optimization by simulated annealing. *Science, Number 4598, 13 May 1983*, 220, 4598:671–680, 1983. URL `citeseer.ist.psu.edu/kirkpatrick83optimization.html`

[Kontsevich94] Kontsevich, L. L., Petrov, A. P., y Vergelskaya, I. S. Reconstruction of shape from shading in color images. *J. Opt. Soc. Am. A*, 11(3):1047, 1994. URL `http://josaa.osa.org/abstract.cfm?URI=josaa-11-3-1047`

[Kwon98] Kwon, Y.-H. Camera calibration. Internet - http://www.kwon3d.com/theory/calib.html, 1998. Http://www.kwon3d.com/theory/calib.html.

[Lowe99] Lowe, D. G. Object recognition from local scale-invariant features. *iccv*, 02:1150, 1999. doi: http://doi.ieeecomputersociety.org/10.1109/ICCV.1999.790410.

[Mikolajczyk05] Mikolajczyk, K. y Schmid, C. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 27(10):1615–1630, 2005.
URL `http://lear.inrialpes.fr/pubs/2005/MS05`

[Ng98] Ng, K., Sequeira, V., Butterfield, S., Hogg, D., y Goncalves, J. An integrated multi-sensory system for photo-realistic 3d scene reconstruction, 1998. URL `citeseer.ist.psu.edu/article/ng98integrated.html`

[Olson93] Olson, C. F. Probabilistic Indexing: Recognizing 3D Objects from 2D Images Using the Probabilistic Peaking Effect. Inf. Téc. CSD-93-733, 93. URL `citeseer.ist.psu.edu/olson93probabilistic.html`

[Ononye02] Ononye, A. y Smith, P. Estimating the shape of a surface with non-constant reflectance from a single color image. pág. Poster Session. 2002.

[Ponce07] Ponce, J., Hebert, M., Schmid, C., y Zisserman, A., eds. *Toward Category-Level Object Recognition*, tomo 4170/2006 de *Lecture Notes in Computer Science*. Springer Berlin and Heidelberg, January 2007.

[Raykar06] Raykar, V. C. y Duraiswami, R. Fast optimal bandwidth selection for kernel density estimation. *En* J. Ghosh, D. Lambert, D. B. Skillicorn, y J. Srivastava, eds., *SDM*. SIAM, 2006. ISBN 0-89871-611-X.
URL `http://www.siam.org/meetings/sdm06/proceedings/054raykarvc.pdf`

[Remondino06] Remondino, F. Detectors and descriptors for photogrammetric applications. págs. xx–yy. 2006.

[Robinson04] Robinson, A., Alboul, L., y Rodrigues, M. A. Methods for indexing stripes in uncoded structured light scanning systems. *En WSCG*, págs. 371–378. 2004.

[Romero07] Romero, L. y Calderon, F. *Scene Reconstruction, Pose Estimation and Tracking*, cap. A Tutorial on Parametric Image Registration. I-Tech Education and Publishing, 2007.

[Rosenzweig92] Rosenzweig, M. R. y Leiman, A. I. *Psicología fisiológica*. McGraw Hill, segunda edición ed$^{\underline{\text{ón}}}$., 1992.

[Sarfraz08] Sarfraz, S. y Hellwich, O. Head pose estimation in face recognition across pose scenarios. *En Proceedings of VISAPP'08*, págs. 235–242. VISAPP, Madeira, Portugal, January 2008. Best Student Paper Award.

[Shapiro01] Shapiro, L. G. y Stockman, G. C. *Computer Vision*. Prentice Hall, 2001.
URL `http://www.cse.msu.edu/ stockman/Book/book.html`

[Shimshoni96] Shimshoni, I. A fast method for estimating the uncertainty in the location of image points in 3d recognition. págs. I: 590–594. 1996.

[Shimshoni99] Shimshoni, I. On estimating the uncertainty in the location of image points in 3d recognition from match sets of different sizes. 74(3):163–173, June 1999.

[Shimshoni00a] Shimshoni, I. y Ponce, J. Probabilistic 3d object recognition. 36(1):51–70, January 2000.

[Shimshoni00b] Shimshoni, I. y Ponce, J. Probabilistic 3d object recognition. *International Journal of Computer Vision*, 36(1):51–70, 2000.
URL `citeseer.ist.psu.edu/248140.html`

[Sinkhorn64] Sinkhorn, R. A relationship between arbitrary positive matrices and doubly stochastic matrices. *The Annals of Mathematical Statistics*, 35(2):876–879, June 1964.

[Trucco98] Trucco, E. y Verri, A. *Introductory Techniques for 3-D Computer Vision*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1998. ISBN 0132611082.

[Various01] Various. Geometry and transformations. Internet, October 2001. Http://www.geometer.org/mathcircles/.

[Wand94] Wand, M. P. y Jones, M. C. *Kernel Smoothing (Monographs on Statistics and Applied Probability)*. Chapman & Hall/CRC, December 1994. ISBN 0412552701.

[Zitova03] Zitova, B. y Flusser, J. Image registration methods: a survey. *Image and Vision Computing*, 21(11):977–1000, October 2003. doi:10.1016/S0262-8856(03)00137-9.
URL `http://dx.doi.org/10.1016/S0262-8856(03)00137-9`