

CENTRO DE INVESTIGACIONES EN ÓPTICA A.C.



TESIS DE MAESTRÍA

**Digitalización de objetos a través de técnicas de proyección de luz
estructurada y reconstrucción mediante técnicas de computación suave**

Tesis presentada por:
Ing. Diego Torres Armenta

Como requisito para obtener el grado de:
Maestría en Optomecatrónica

Dirigida por:
Dr. Francisco Javier Cuevas de la Rosa

Fecha:
26 de noviembre de 2010

*A la memoria de mi padre. Que en donde se encuentre
sepa que hizo el mejor de los trabajos.*

Agradecimientos

Agradezco en primer lugar a mi asesor de tesis, Dr. Francisco Cuevas de la Rosa por darme la confianza para trabajar libremente y por sus consejos en cuanto hacia donde encaminar este trabajo de tesis.

Agradecer también a mis sinodales, Dr. Juan Manuel López y Dr. Ramón Rodríguez, por tomarse el tiempo de revisar el presente trabajo y por sus sabios consejos en cuanto el desarrollo del mismo. Al Ing. Juan Rayas por su valioso apoyo en los detalles técnicos involucrados en la adquisición de superficies.

Un agradecimiento también a mis compañeros durante la maestría, especialmente a Giorgio, Víctor M., Frank, Miguel A. y César por hacer que los momentos más estresantes fueran más sencillos de llevar.

De igual manera, le doy las gracias al Centro de Investigaciones en Óptica, por brindarme todo lo necesario para la realización de este trabajo, así como a todos aquellos docentes que tuvieron que ver con mi formación. Se agradece también el apoyo económico proporcionado por CONACYT.

Finalmente, agradecer a mis padres Pascual y Concepción, así como a mis hermanos Leonardo, Rafael, David, Octavio y Alejandra, pues gracias a ustedes todo el esfuerzo tiene sentido.

Resumen

Hoy en día, es cada vez más frecuente la necesidad de muchas compañías de tener una representación digital de sus productos, ya sea para muestra a sus clientes, como un parámetro de calidad o como un proceso de prototipado rápido. En muchos casos, esta digitalización puede llevarse a cabo mediante el diseño en algún software CAD. Sin embargo, este proceso es muy complicado y en ocasiones imposible para objetos que cuentan con geometrías complejas.

Con este fin, se han desarrollado diversas técnicas de medición de topografía de objetos incluyendo máquinas de contacto, así como diversas técnicas ópticas tanto invasivas como no invasivas. Los métodos que hacen uso del contacto físico con el objeto pueden llegar a dañar la muestra bajo análisis, mientras que los métodos ópticos sólo adquieren una pequeña porción de la misma y es necesario contar con arreglos ópticos fijos y contar con un conocimiento vasto de los parámetros de rotación aplicados a la muestra para su completa adquisición y posterior reconstrucción.

El presente trabajo propone realizar la adquisición de nubes densas de puntos mediante la técnica de proyección de franjas, dada su adaptabilidad al tamaño de los detalles a adquirir con la simple modificación de la frecuencia de las franjas proyectadas, además de que, en comparación con otras técnicas, permite la adquisición de una mayor superficie con una sola toma y requiere de un menor tiempo de procesamiento. De manera adicional, se propone el registro de nubes de puntos o reconstrucción del objeto a través de técnicas de computación evolutiva y la implementación del algoritmo de iteración del punto más cercano (ICP).

Se ha probado el algoritmo propuesto en objetos con geometrías complejas dando un alto grado de repetitividad y un error pequeño en comparación con el ICP puro.

Índice General

Resumen.....	ii
Lista de figuras.....	vi
Lista de tablas.....	viii
Capítulo 1 – Introducción	1
1.1 Objetivo de la tesis.....	1
1.2 Contribuciones del trabajo.....	1
1.3 Definición y planteamiento del problema.....	1
1.4 Antecedentes.....	2
1.5 Revisión de la tesis.....	4
1.5.1 Técnicas de digitalización de objetos.....	4
1.5.2 Calibración de sistemas cámara-proyector.....	5
1.5.3 Técnicas de computación suave para búsqueda, optimización y aprendizaje de máquina.....	5
1.5.4 Registro de superficies utilizando algoritmos evolutivos.....	5
1.5.5 Conclusiones y trabajo futuro.....	5
Referencias.....	6
Capítulo 2 – Técnicas de digitalización de objetos	8
2.1 Técnicas estereoscópicas.....	8
2.1.1 Estereoscopía fotográfica.....	8
2.1.1.1 Determinación de pares conjugados.....	10
2.1.2 Estéreo fotométrico.....	12
2.2 Técnicas que utilizan luz estructurada.....	16
2.2.1 Proyección de franjas con perfil cosenoidal.....	18
2.2.2 Métodos de recuperación de fase.....	20
2.2.2.1 Método de demodulación directa.....	20
2.2.2.2 Método de Fourier-Takeda.....	21
2.2.2.3 Método de corrimiento de fase.....	23

2.2.3 Técnicas de desenvolvimiento de fase.....	25
Referencias.....	28
Capítulo 3 – Calibración de sistemas que utilizan visión artificial	31
3.1 Cámara sin distorsión (pinhole).....	31
3.2 Relación entre sistemas coordenados de cámara y de mundo.....	32
3.3 Calibración por Transformaciones Lineales Discretas.....	35
3.4 Calibración de sistemas cámara-proyector.....	36
3.4.1 Ejes ópticos coplanares y paralelos.....	36
3.4.2 Ejes ópticos coplanares y no paralelos.....	37
3.4.3 Ejes ópticos no coplanares y no paralelos.....	38
3.5 Cámara con distorsión geométrica.....	41
3.5.1 Distorsión radial.....	41
3.5.2 Distorsión de de-centro.....	44
3.5.3 Distorsión de prisma delgado y distorsión total.....	46
Referencias.....	48
Capítulo 4 – Técnicas de computación suave para búsqueda, optimización y aprendizaje de máquina	49
4.1 Métodos de búsqueda exhaustivos y analíticos.....	49
4.1.1 Búsqueda del segmento dorado.....	50
4.1.2 Búsqueda de Fibonacci.....	52
4.1.3 Método de Newton.....	53
4.1.4 Método de la secante.....	55
4.2 Métodos de búsqueda y aprendizaje basados en modelos naturales.....	56
4.2.1 Redes neuronales.....	56
4.2.1.1 Entrenamiento de una neurona simple.....	58
4.2.1.2 Entrenamiento de redes multicapas: algoritmo de retro-propagación.....	60
4.2.2 Recocido simulado.....	64
4.2.3 Optimización de enjambre de partículas.....	65
4.2.4 Optimización de colonia de hormigas.....	65
4.2.5 Algoritmos genéticos.....	67
4.2.5.1 Justificación de la convergencia de algoritmos genéticos.....	68
4.2.5.2 Codificación del algoritmo genético.....	70
4.2.5.2.1 Codificación binaria.....	71
4.2.5.2.2 Codificación Gray.....	72
4.2.5.2.3 Codificación IEEE.....	73

4.2.5.2.4 Codificación continua.....	73
4.2.5.3 Operador de cruza.....	74
4.2.5.4 Operador de selección.....	75
4.2.5.4.1 Métodos que no consideran aptitud.....	76
4.2.5.4.2 Método de la ruleta.....	76
4.2.5.4.3 Escalamiento lineal.....	76
4.2.5.4.4 Clasificación (ranking)	77
4.2.5.4.5 Método de torneo.....	78
4.2.5.5 Operador de mutación.....	78
Referencias.....	80
Capítulo 5 – Registro de superficies utilizando algoritmos genéticos	82
5.1 Algoritmo de Iteración del Punto más Cercano.....	82
5.1.1 Árboles k-dimensionales.....	84
5.1.2 Descomposición de valores singulares.....	85
5.2 Descripción del algoritmo evolutivo utilizado.....	88
5.2.1 Muestreo.....	91
5.2.2 Cálculo de la aptitud de cada individuo.....	94
5.2.3 Operadores de selección, cruza y mutación.....	96
5.3 Experimentos.....	99
5.3.1 Registro de una superficie sintética.....	99
5.3.2 Registro de una superficie adquirida digitalmente.....	100
5.4 Limitaciones y consideraciones.....	103
Referencias.....	105
Capítulo 6 – Conclusiones y trabajo futuro	109
6.1 Conclusiones.....	109
6.2 Trabajo futuro.....	111
Apéndice A – Calibración de cámaras	112
A.1 Conversión de un espacio real a uno proyectado.....	112
A.2 Calibración por Transformaciones Lineales Discretas.....	115
Referencias.....	119

Apéndice B – Plan de negocios	120
B.1 Introducción.....	120
B.1.1 Misión.....	120
B.1.2 Visión.....	120
B.1.3 Producto/Idea.....	120
B.2 Resumen ejecutivo.....	121
B.3 Análisis FODA.....	121
B.4 Descripción de la empresa.....	122
B.5 Análisis de mercado.....	123
B.5.1 Perfil de mercado.....	124
B.5.2 Análisis de la competencia.....	126
Referencias.....	128
Apéndice C – Carta de enlace con la industria	129

Lista de figuras

Figura	Página
2.1 Disposición geométrica de un sistema de visión estéreo.....	9
2.2 Dos imágenes del mismo objeto tomadas con una distancia base entre los ejes ópticos de 20mm.....	11
2.3 Superficie reconstruida a partir de las imágenes mostradas en la figura 2.2 con la técnica de estéreo fotográfico al minimizar a ecuación 2.9.....	11
2.4 Geometría de la iluminación y punto de vista de un punto en la superficie.....	13
2.5 Patrones de sombra sobre un objeto con vectores de iluminación a) $L_1 = [0.3, -0.12, 0.94]$, b) $L_2 = [-0.3, -0.12, 0.94]$ y c) $L_3 = [-0.1, 0.32, 0.94]$	15
2.6 Mapas de derivadas de x e y obtenidas de los patrones de sombras con las ecuaciones	
2.21.....	15
2.7 Superficie reconstruida al integrar los mapas de derivadas obtenidos con la ecuación 2.24 después de 5,000 iteraciones.....	16
2.8 Diagrama de la proyección de un punto.....	16

2.9	Geometría de la proyección de franjas.....	18
2.10	Esquema de la aplicación del método de demodulación directa.....	21
2.11	(a) Espectro de Fourier de un patrón de franjas y; (b) espectro seleccionado y trasladado al origen.....	22
2.12	Pasos necesarios para el método de corrimiento de fase. (a) Imágenes de entrada; (b) fase envuelta obtenida a partir de la ecuación 2.37; (c) fase desenvuelta y; (c) reconstrucción del objeto.....	24
2.13	Diagramas de fase (a) envuelta y; (b) desenvuelta.....	25
3.1	Proyecciones de un punto P a un punto p (a) central y; (b) paralela.....	31
3.2	Punto objeto y plano imagen para una lente delgada.....	33
3.3	Coordenadas de mundo, de cámara y de imagen.....	34
3.4	Geometría de la proyección de franjas para ejes ópticos coplanares y paralelos.....	36
3.5	Sistema de proyección de franjas con ejes ópticos coplanares y no paralelos.....	37
3.6	Sistema de proyección de franjas con ejes ópticos no coplanares y no paralelos.....	38
3.7	Vista frontal del sistema de proyección de franjas con ejes ópticos no coplanares y no paralelos.....	39
3.8	Geometría de cámara con proyección en perspectiva y distorsión de lente.....	42
3.9	Deformación de una imagen ideal por el efecto de distorsión radial de lente.....	43
3.10	Distorsión de de-centro (a) Cuadrícula sin distorsión, (b) cuadrícula con distorsión.....	45
3.11	Distorsión tangencial (denotada por la línea punteada).....	45
4.1	Función unimodal.....	50
4.2	Caso en que $f(a_1) < f(b_1)$; x^* representa el mínimo.....	51
4.3	Selección de puntos de evaluación.....	52
4.4	Neurona simple.....	57
4.5	Estructura de una red neuronal multi-capa.....	57
4.6	Neurona simple lineal.....	58
4.7	Adaline.....	59
4.8	Red neuronal de tres capas.....	60
4.9	Codificación binaria de un cromosoma de n variables.....	71
4.10	Representación de la codificación IEEE.....	73
4.11	Operación de cruce de un punto.....	74
4.12	Operación de cruce de dos puntos.....	75
5.1	Esquema de construcción de un árbol kd bidimensional.....	84
5.2	Significado geométrico de SVD: la imagen de un círculo tras una multiplicación matricial es una elipse.....	86
5.3	Convención adoptada para parámetros de transformación positivos.....	89
5.4	Sub espacios muestrales para dos círculos unitarios superpuestos.....	93
5.5	Cruza de generación con nueva información.....	97
5.6	a) Objetivo y, b) nube fija y nube flotante con transformación.....	98
5.7	Resultado obtenido del algoritmo genético.....	99
5.8	Estadística de la prueba.....	99
5.9	Resultado de aplicar ICP a los parámetros obtenidos por el GA.....	100

5.10	a) Fotografía de objeto real y, b) reconstrucción del mismo.....	101
5.11	Nube fija X y nube flotante después de la transformación tP.....	101
5.12	Resultado obtenido del algoritmo genético.....	102
5.13	Estadística de la prueba.....	102
5.14	Resultado de aplicar ICP a los parámetros obtenidos por el GA.....	103
5.15	Una mala solución antes de aplicar el algoritmo ICP.....	103
5.16	Objetivo para una prueba con una curva gaussiana.....	104
5.17	Una solución incorrecta para la curva mostrada en la figura 5.16 antes de ICP.....	104
A.1	Proyecciones de un punto P a un punto p (a) central y; (b) paralela.....	112
A.2	Coordenadas de mundo, de cámara y de imagen.....	114

Lista de tablas

Tabla		Página
5.1	Algoritmo ICP.....	83
5.2	Definición de parámetros para clasificación de superficies.....	92
B.1	Participación al PIB para el estado de Guanajuato, según división económica.....	124
B.2	Participación al PIB de la industria manufacturera.....	124
B.3	Producto bruto de subsectores manufactureros por municipio.....	125

Capítulo 1

Introducción

1.1 Objetivo de la Tesis

El objetivo de este trabajo de tesis consiste en adquirir la topometría de objetos a través de técnicas de proyección de luz estructurada y realizar un alineamiento de las distintas superficies adquiridas a través de técnicas de computación suave. Para llevar a cabo el emparejamiento de las nubes de puntos se plantea la utilización de Algoritmos Evolutivos para tener un primer acoplamiento aproximado y después un ajuste fino a través del Algoritmo de Iteración de Puntos más Cercanos, o ICP por sus siglas en inglés. La solución del problema considera que no se tiene conocimiento previo de los parámetros de rotación y traslación existente entre una nube y otra. La única condición necesaria es la existencia de una superficie de traslape suficiente para unir las superficies bajo análisis.

1.2 Contribuciones del trabajo

- Utilización de algoritmos evolutivos para la reconstrucción de objetos digitalizados a partir de la técnica de proyección de luz estructurada.
- Omisión de la necesidad de puntos de referencia para la reconstrucción gracias a técnicas de computación suave.

1.3 Definición y planteamiento del problema

Uno de los mayores problemas al llevar a cabo reconstrucción de superficies es la necesidad de la utilización de puntos de referencia desde donde pueda hacerse un acoplamiento adecuado de dos o más superficies obtenidas a través de técnicas de adquisición de topometría o técnicas de perfilometría. Estos puntos de referencia son fáciles de identificar si la topografía del objeto presenta particularidades, en caso contrario se requiere de métodos invasivos, como la aplicación de marcas ajenas a la muestra.

Otro inconveniente es la necesidad de rotar la muestra alrededor de un eje referenciado a un plano de calibración fijo, esta situación obliga a que el sistema de adquisición se encuentre asentado, situación que reduce grandemente la practicidad del mismo.

Para solucionar dichos problemas se plantea la utilización de un algoritmo evolutivo para la reconstrucción de las superficies digitalizadas en combinación con una técnica de minimización de distancia entre nubes de puntos conocida como Iteración del Punto más Cercano, con la ventaja de que no se requieran parámetros de la dirección de movimiento o rotación entre las distintas imágenes de rango adquiridas ni marcas ajenas a la muestra.

Se plantea realizar la digitalización a través del método de proyección de franjas, pues éste permite obtener superficies relativamente grandes con respecto a otras técnicas, además de que es fácil de calibrar dados los parámetros ya conocidos provenientes de la proyección (grosor de la franjas, relación geométrica entre proyector y cámara, etc.), además de que, en la etapa de desenvolvimiento de fase, pueden ignorarse áreas sub-muestreadas para su posterior reconstrucción a través de interpolación o con el uso de una perspectiva distinta de la misma área.

1.4 Antecedentes

En la actualidad, la digitalización de objetos se ha vuelto una tarea de vital importancia en la industria para las tareas de modelado de producto y prototipado rápido y, aunque existen diversos sistemas que cumplen con esta necesidad, la mayoría de estos necesitan seguir una trayectoria conocida [Ze-Su *et al.*, 2007] o utilizar puntos de referencia [Percoco y Galantucci, 2007] que, en general, son invasivos para la muestra a digitalizar.

Existen diversas técnicas ópticas para llevar a cabo el digitalizado o reconstrucción de la topografía de un objeto, entre estas se encuentran las que dependen de la diferencia en perspectiva entre dos imágenes tomadas desde diferentes perspectivas, las que necesitan de la proyección de un patrón de luz conocido (luz estructurada) así la combinación de ambas.

A las técnicas que utilizan diferencia en perspectiva entre dos o más imágenes tomadas del mismo objeto, se les conoce como técnicas de fotografía estéreo. Al trabajar con una fotografía puede obtenerse información de la geometría del objeto, es decir, información bidimensional. Al trabajar con dos imágenes, si se trabaja en la zona común entre ellas (zona de traslape), puede tenerse visión estereoscópica o bien, información tridimensional [Horn, 1986]. Estas técnicas emulan a menudo el comportamiento de la visión humana, que depende de la diferencia entre lo que percibe un ojo con respecto al otro, dado el hecho de que este proceso se emula con la colocación de dos cámaras colocadas a una distancia base una de la otra, a esta técnica se le conoce también como estéreo fotográfico [Haralick y Shapiro, 1993]. Esta técnica presenta la ventaja de que no considera condiciones de iluminación, pues en general, las adquisiciones se hacen en el mismo instante, aunque por otro lado, encontrar la correlación entre los puntos comunes en cada imagen suele un proceso tardado y con un costo computacional alto.

Una segunda técnica basada en la adquisición de más de una imagen del mismo objeto es la de estéreo fotométrico, en donde se proyecta luz hacia un objeto desde posiciones conocidas y se obtiene la topografía del mismo a partir de las diferencias que este presenta en las sombras o,

mejor dicho, en la reflectancia de su superficie [Horn, 1986; Klette, 1996]. Para obtener los vectores normales a la escena bajo observación se requiere calcular los mapas de gradiente tanto en x como en y para así llegar a un problema simple de ecuaciones simultáneas. La principal desventaja de aplicar estéreo fotométrico es que requiere de al menos tres imágenes para resolver la profundidad en cada punto, además de implementar un método de integración. Sin embargo, si se tiene una buena calibración y se genera una tabla de búsqueda, puede acelerarse en gran medida este último proceso.

Cuando se habla de la proyección de luz estructurada, se trata de utilizar un haz o un patrón de iluminación conocido para medir su deformación con respecto a un plano u objeto de referencia y obtener la topografía del objeto bajo observación considerando métodos de calibración del sistema de visión para expresar los resultados obtenidos en coordenadas reales.

El caso más simple es la técnica de proyección de un punto de luz para medir distancias entre el sistema de adquisición de imágenes y puntos en la superficie del objeto. Un solo haz de luz se modela como un rayo (o línea) se proyecta sobre la escena. La proyección del rayo y haz se intersecan para encontrar la posición del objeto iluminado en el espacio tridimensional [Klette et al, 1996].

La proyección de una franja o línea de luz representa una extensión de la técnica de proyección de un punto. Esta técnica proyecta una hoja de luz sobre la escena. La idea es intersecar el rayo proyectado del punto de la imagen examinado con el plano de luz. La intersección del plano de luz con la superficie del objeto es visible como una línea de luz en la imagen, lo que permite obtener una mayor cantidad de valores de profundidad y hacer una reconstrucción más rápida comparada con el método de un solo punto [Klette *et al*, 1996]. Estas dos últimas técnicas cuentan con la gran ventaja de que sólo es necesario calibrar la cámara, pues la interpretación de las imágenes adquiridas es extremadamente sencilla en términos de determinar la deformación del patrón proyectado. Lamentablemente, con estos métodos se obtiene una cantidad de puntos muy limitada y suele prolongarse el proceso de adquisición.

La técnica que involucra la proyección de un patrón de franjas es probablemente la más eficiente, ya que se puede adquirir una mayor superficie en una sola imagen. Su funcionamiento consiste en la recuperación de la fase relacionada con el perfil del objeto bajo estudio y, en la mayoría de los casos, un paso adicional de desenvolvimiento de la misma [Cuevas, 2000]. En general, este método es el más veloz, ya que requiere de una sola imagen para reconstruir una superficie. Su mayor desventaja en comparación con otras técnicas es que necesita de la calibración tanto de la cámara como del sistema de proyección.

El registro de superficies consiste en llevar a cabo la alineación de dos nubes de puntos que contienen entre si un área de traslape o superposición, normalmente obtenidas de alguno de las técnicas de digitalización antes mencionadas. El procedimiento de registro se hace a partir propiedades regionales de la forma [Xiao *et al* 2005], comparación entre las propiedades cinemáticas y geometría de la función de distancia de las superficies [Pottman *et al*, 2004] o por minimización de la distancia media entre ambas superficies, siendo el método más utilizado el del

algoritmo ICP [Besl y McKay, 1992; Yaniv, 2002]. También se han utilizado combinaciones de estas con algún algoritmo evolutivo [Chow *et al*, 2004; Lomonosov *et al*, 2006].

En esta tesis se plantea utilizar la técnica de digitalización por proyección de franjas para la adquisición de las nubes de puntos, sin embargo, la fortaleza de la misma se basa en proponer una función objetivo para implementar el algoritmo evolutivo necesario para el registro, utilizando la minimización de la distancia mínima promedio entre los puntos de dos superficies y otras características como la razón de traslape y la discriminación de puntos de acuerdo a tolerancias establecidas. Debe de hacerse notar que, en este caso, la finalidad del algoritmo evolutivo no es la alineación perfecta de las superficies bajo análisis, sino un buen acercamiento que permita la aplicación del algoritmo de Iteración del Punto más Cercano, que tiene la capacidad de realizar un ajuste fino muy confiable, siempre y cuando se tenga un buen pre alineamiento.

1.5 Revisión de la tesis

A continuación se presenta un resumen general de los distintos capítulos contenidos en el trabajo de tesis, para así dar al lector un antecedente del contenido del mismo además de que este pueda enfocarse a la lectura de algún capítulo de su interés particular.

1.5.1 Técnicas de digitalización de objetos

El capítulo 2 comprende un recuento de las técnicas de digitalización ya mencionadas en la sección anterior, este comprende en detalle los arreglos ópticos y las matemáticas involucradas en la determinación de la topografía de objetos.

En primera instancia, se trata la técnica de estéreo fotográfico que utiliza dos imágenes del objeto a reconstruir. En esta sección se analiza el principio fundamental de la obtención de topografía a partir de la disparidad entre los puntos conjugados en la superficie común entre dos imágenes tomadas desde perspectivas diferentes, ya encontrada la disparidad, puede calcularse la profundidad a partir de un cálculo geométrico.

Como segundo tema se describe el caso de la técnica de estéreo fotométrico, basada en la adquisición de imágenes del mismo objeto obtenidas con diferentes fuentes de iluminación [Lee y Kuo, 1993]. El conocimiento a priori de las ubicaciones de al menos dos fuentes proporciona información para obtener la profundidad de objeto con respecto a un plano de calibración, sin embargo, dos tomas no son suficientes para resolver el sistema de manera concreta ya que para ciertos puntos habrá más de una posible solución. Por lo antes mencionado, en este trabajo se utiliza la técnica con tres fuentes de iluminación, esto proporciona suficiente información para resolver el sistema en que se buscan los vectores normales a la iluminación de manera adecuada y con una solución única a partir del método de minimización de energía [Horn, 1986], mismo que requiere como entrada únicamente los planos gradientes correspondientes a las direcciones x e y .

Por último, se hace referencia a las técnicas de proyección de luz estructurada. En esta sección, se analiza la técnica de proyección de un punto de luz, de proyección de una franja de luz y de luz con perfil cosenoidal. Se analiza primero el arreglo necesario para proyección de un punto y se menciona el caso de proyección de una línea o cono de luz como una extensión de la proyección de un punto. Posteriormente, se analiza la técnica de proyección de franjas con perfil cosenoidal; esta técnica se encuentra relacionada con el concepto de interferometría conocido como fase, que será utilizado para obtener la profundidad del objeto bajo análisis. Se describirán distintas maneras de obtener dicha fase y se explicarán los conceptos de fase envuelta y fase desenvuelta, además de que se mostrarán algunas formas de llevar a cabo el desenvolvimiento de la misma.

1.5.3 Calibración de sistemas cámara-proyector

Este capítulo trata de la forma de transformar coordenadas del mundo real a coordenadas máquina desde distintas configuraciones. Dependiendo de la localización de las partes del sistema pueden dividirse en: sistemas de ejes ópticos paralelos y coplanares, ejes ópticos no paralelos y coplanares, ejes ópticos no coplanares y no paralelos. Se exponen además métodos de calibración para sistemas que incluyen proyección de franjas [X. Chen et al, 2009; Zhang y Huang, 2006] que considera tanto las aberraciones en la cámara como la inclinación inducida al proyector con respecto a la misma.

1.5.4 Técnicas de computación suave para búsqueda, optimización y aprendizaje de máquina

En el capítulo 4 se exponen algunas de las técnicas computacionales de búsqueda y optimización así como la justificación para el uso de los mismos, primeramente se exponen algunos métodos convencionales basados en derivación y análisis; en segundo lugar se mencionan algunos modelos basados en la naturaleza. En esta última parte se hace énfasis en los algoritmos evolutivos, desde la justificación de su funcionamiento basada en la teoría de esquemas [Goldberg, 1989; Chambers, 2002] hasta la descripción de los operadores involucrados y algunas variantes de estos.

1.5.5 Registro de superficies utilizando algoritmos evolutivos

En este capítulo se trata la parte central de la tesis: el registro de superficies. Se comienza describiendo el método de registro por la búsqueda iterativa del Punto más Cercano o ICP, por sus siglas en inglés [Yaniv, 2002]. Así mismo se trata la función objetivo a aplicar para evaluar el desempeño dentro del algoritmo evolutivo, la codificación y la parametrización del mismo con el fin de buscar el mínimo global del problema a resolver.

1.5.6 Conclusiones y trabajo futuro

En el capítulo 6 se muestran los resultados obtenidos y se hace especial énfasis en las contribuciones del trabajo desarrollado. Se plantea también una serie de mejoras que pueden hacerse en un futuro próximo para garantizar una mejor y más rápida convergencia del método planteado.

Referencias

- [1] Besl, P. J., McKay N.D., 'A method for registration of 3-D shapes', IEEE Trans. Pattern Anal. Machine Intell., **14**: 239-256 (1992).
- [2] Chambers, L., 'Practical handbook of genetic algorithms: Applications', Chapman and Hall, 2002.
- [3] Chen, X., Xi, J., Jin, Y., Sun, J., 'Accurate calibration for a camera-projector measurement system based on structured light projection', Optics and Lasers in Engineering **47**: 310-319 (2009).
- [4] Chow, C.K., Tsui, H.T., Lee, T., 'Surface registration using a dynamic genetic algorithm', Pattern Recognition, Volume **37**: 105 – 117 (2004).
- [5] Cuevas, F.J., 'Aplicación de redes neuronales en la reconstrucción tridimensional de objetos', Tesis doctoral, Centro de Investigaciones en Óptica, A.C., León, México (2000).
- [6] Goldberg, D., 'Genetic Algorithms in Search, Optimization and Machine Learning', Addison-Wesley, 1989.
- [7] Haralick, R.M., Shapiro, L.G., 'Computer and Robot Vision, Volume II', Addison-Wesley, 1993.
- [8] Horn, B.K.P., 'Robot Vision', MIT Press, 1986.
- [9] Klette, R., Schlüns, K., Koschan, A., 'Computer Vision: Three-Dimensional Data from Images', Springer, 1996.
- [10] Lee, K.M., Kuo, C.C.J., 'Surface reconstruction from photometric stereo images', J. Opt. Soc. Am. A, **10**: 855 – 868 (1993).
- [11] Lomonosov, E., Chetverikov, D., Ekárt, A., 'Pre-registration of arbitrary oriented 3D surfaces using a genetic algorithm', Pattern Recognition Letters, Volume **27**: 101 – 114 (2006).
- [12] G. Percoco, L. M. Galantucci, 'Genetic Point Cloud Alignment for Computer Aided Inspection and Reverse Engineering', Politecnico di Bari, Italy (2007).
- [13] Pottman, H., Leopoldseder, S., Hofer, M., 'Registration without ICP', Computer Vision and Image Understanding, Volume **95**: 54 – 71 (2004).
- [14] Xiao, G., Ong, S.H., Foong, K.W.C., 'Efficient partial surface registration for 3D objects', Computer Vision and Image Understanding, Volume **98**: 271 – 274 (2005).
- [15] Yaniv, Z., 'Rigid Registration: The Iterative Closest Point Algorithm', School of Engineering and Computer Science, The Hebrew University, Israel, 2002.
- [16] Zhang, S., Huang, P.S., 'Novel method for structured light system calibration', Optical Engineering **48**: 3134 - 3142 (2006).

[17] Ze-Su, H. Bing-Rong, L. Hong, *An Improved Polar Scan Matching Using Genetic Algorithm*, Information Technology Journal 6(1): 89-95, 2007.

Capítulo 2

Técnicas de digitalización de objetos

En este capítulo se revisan las técnicas para adquisición de forma y profundidad de objetos. Se hace énfasis en las técnicas que emplean luz estructurada, principalmente en la proyección de franjas, ya que es la que se empleará para la adquisición de superficies que se manejarán en el capítulo 6.

2.1 Técnicas estereoscópicas

Como ya se mencionó anteriormente, las técnicas de visión estéreo son aquellas con que se realiza una medición a partir de varias imágenes del mismo objeto, pero con variación ya sea en el punto de observación o la dirección de iluminación. A la técnica que utiliza dos o más imágenes del mismo objeto, pero con distinta perspectiva, se le conoce como estereoscopia fotográfica [Horn, 1986], mientras que a la que utiliza iluminación desde fuentes colocadas en distintas posiciones se le llama estéreo fotométrico [Horn, 1986; Klette *et al*, 1996].

La estereoscopia fotográfica necesita de al menos dos imágenes del mismo objeto, pero con una distancia de separación entre las cámaras utilizadas (véase figura 2.1). Puede obtenerse la profundidad de cada punto en el área común de las imágenes adquiridas por medio de triangulación, aunque este proceso requiere de una correlación entre las imágenes para encontrar los puntos conjugados en cada una de las cámaras. La profundidad de cada punto está relacionada con la disparidad entre los puntos, misma que se determina por medio de los pares conjugados.

Para el caso el estéreo fotométrico, la reconstrucción se realiza a partir de dos o más tomas del mismo objeto iluminado a partir de fuentes con distintas posiciones. El patrón de sombreado obtenido puede relacionarse con la forma del objeto a partir de los mapas de gradientes parciales resultantes de las matrices de reflectancia y los vectores de iluminación. La profundidad relacionada a cada punto se obtiene al aplicar alguna técnica de integración a los gradientes obtenidos.

2.1.1 Estereoscopia fotográfica

Considérese un sistema de dos cámaras en donde ambos ejes ópticos son paralelos [Horn, 1986; Peng y Guo-Qiang, 2010] y se encuentran separadas por una distancia que conecta los ejes ópticos de las lentes, llamada línea de base o distancia base, denotada por b , que es perpendicular a los ejes ópticos. EL eje x se encuentra orientado de tal manera que este sea perpendicular a la

línea de base. El arreglo básico para llevar a cabo el estéreo fotométrico se muestra en la figura 2.1.

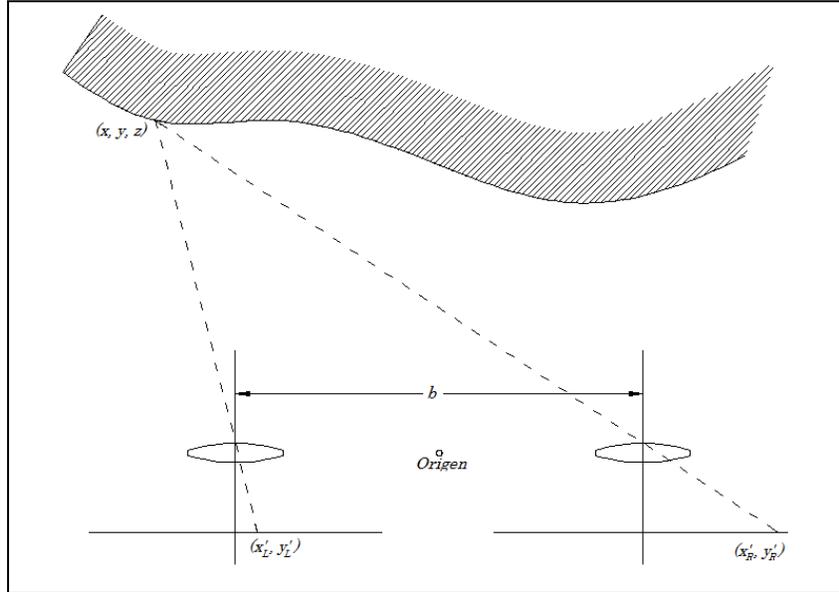


Figura 2.1 Disposición geométrica de un sistema de visión estéreo

Las coordenadas de un punto (x, y, z) se miden de manera relativa a un punto de origen intermedio entre los centros de las lentes. Sean las coordenadas de las imágenes izquierda y derecha (x'_L, y'_L) y (x'_R, y'_R) , respectivamente. Entonces, en la región paraxial se obtiene

$$\frac{x'_L}{f} = \frac{x + b/2}{z} \quad y \quad \frac{x'_R}{f} = \frac{x - b/2}{z} \quad (2.1)$$

mientras que

$$\frac{y'_L}{f} = \frac{y'_R}{f} = \frac{y}{z} \quad (2.2)$$

en donde f es la distancia del centro de la lente al plano imagen en ambas cámaras. Estas tres ecuaciones pueden resolverse para las tres incógnitas x, y y z . La diferencia en coordenadas de imagen $x'_L - x'_R$ se denomina *disparidad* y está dada por

$$\frac{x'_L - x'_R}{f} = \frac{b}{z} \quad (2.3)$$

Se tiene entonces

$$x = b \frac{(x'_L + x'_R)/2}{x'_L - x'_R}, \quad y = b \frac{(y'_L + y'_R)/2}{x'_L - x'_R}, \quad z = b \frac{f}{x'_L - x'_R} \quad (2.4)$$

La distancia z es inversamente proporcional a la disparidad. Distancias a objetos cercanos pueden medirse de manera precisa, mientras que las distancias a objetos lejanos no tanto. Nótese

también que la profundidad es proporcional a b , la distancia entre las lentes. De aquí se concluye que la precisión de la determinación de profundidad incrementa a la par de la distancia de base entre las cámaras, sin embargo, la disparidad debe de compensarse usando distancias focales mayores [Horn, 1986].

2.1.1.1 Determinación de pares conjugados

Como se mencionó anteriormente, una parte fundamental para encontrar la profundidad z de un objeto a partir de un sistema estereoscópico fotográfico es la determinación de la disparidad entre las imágenes percibidas por ambas cámaras. A este proceso se le denomina como localización de pares conjugados entre imágenes y puede llevarse a cabo de diversas maneras, incluyendo comparación de intensidad, correlación y detección de bordes, entre otros [Horn, 1986; Haralick y Shapiro, 1993].

Un punto visible en ambas cámaras da lugar a un par de puntos imagen llamados *par conjugado*. Debe hacerse notar que un punto en la imagen izquierda correspondiente a un punto en la imagen derecha debe encontrarse en una línea en particular, si se considera el sistema mostrado en la figura 2.1, puede asumirse que los dos tienen la misma coordenada y . Esta línea se denomina *línea epipolar*.

Si se tienen líneas epipolares paralelas al eje x , entonces se puede esperar que los niveles de gris de puntos vecinos en una imagen puedan emparejarse con los niveles de gris de puntos vecinos en la otra imagen. Puede considerarse el problema entonces como una comparación de formas de onda de niveles de gris sobre líneas epipolares.

Considérese la geometría simple mostrada en la figura 2.1. Debe encontrarse entonces una función $z(x, y)$ tal que

$$E_L(x'_L, y'_L) = E_R(x'_R, y'_R) \quad (2.5)$$

que al aplicar las igualdades 2.1 y 2.2, puede reescribirse como

$$E_L\left(f \frac{x + b/2}{z(x, y)}, y'\right) = E_R\left(f \frac{x - b/2}{z(x, y)}, y'\right) \quad (2.6)$$

En este punto es conveniente realizar un cambio de variables, para expresar el sistema en función de coordenadas imagen $(x', y')^T$ en lugar de coordenadas reales $(x, y, z)^T$. Sea entonces

$$\frac{x'}{f} = \frac{x}{z} \quad y \quad d(x', y') = \frac{bf}{z} \quad (2.7)$$

Se está buscando una función de disparidad $d(x', y')$ tal que

$$E_L\left(x' + \frac{1}{2}d(x', y'), y'\right) = E_R\left(x' - \frac{1}{2}d(x', y'), y'\right) \quad (2.8)$$

Sin embargo, como los niveles de gris en las imágenes no es exactamente el mismo, no debe insistirse en que se cumpla exactamente la condición $E_L = E_R$. En lugar de eso, se busca minimizar

$$e_i = \int (E_L - E_R)^2 dx' dy' \quad (2.9)$$

La restricción se cumple cuando los valores en una vecindad $E_L(x, y)$ son aproximadamente iguales a los valores de una vecindad del mismo tamaño en $E_R(x, y)$.

En las figuras 2.2 se muestra una reconstrucción a partir de un objeto real con una línea de base $b = 20\text{mm}$. Las imágenes fueron adquiridas con una cámara CCD PixeLink y una lente telefoto con una apertura de F1.8-32C. Las imágenes adquiridas tienen una resolución de 480×480 . Se llevó a cabo una correlación con una ventana 10×10 . La superficie reconstruida se muestra en la figura 2.3.

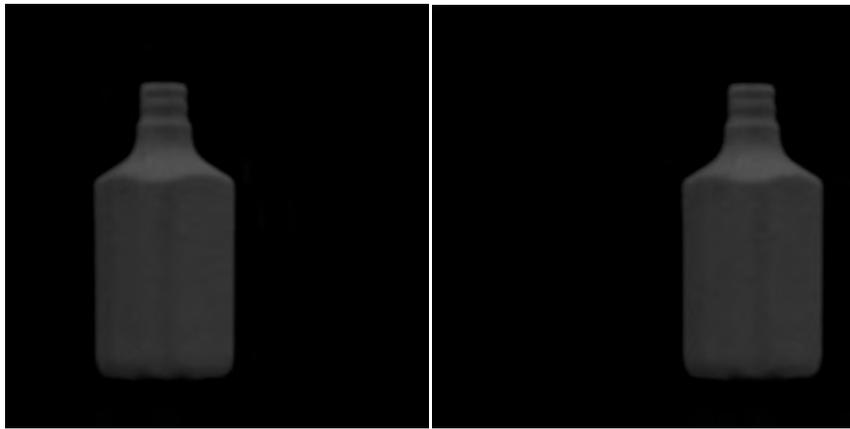


Figura 2.2 Dos imágenes del mismo objeto tomadas con una distancia entre los ejes ópticos de 20mm

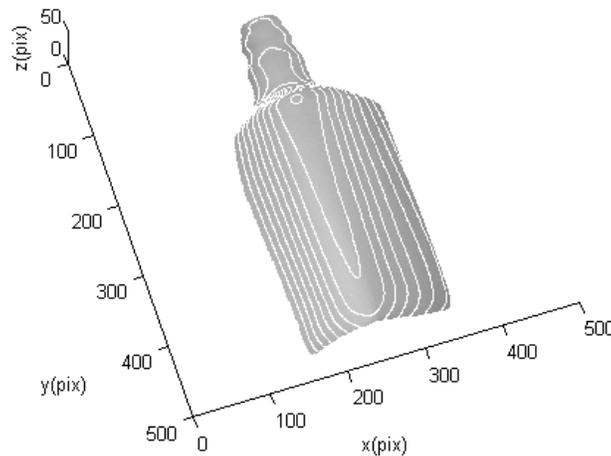


Figura 2.3 Superficie reconstruida a partir de las imágenes mostradas en la figura 2.2 con la técnica de estéreo fotográfico al minimizar la ecuación 2.9

Como puede notarse una de las principales desventajas de la técnica de estéreo fotográfico es la necesidad de encontrar los puntos conjugados para determinar la disparidad entre ambas imágenes, este proceso lleva un tiempo de cómputo bastante prolongado si no se cuentan con

marcas de referencia o si se utilizan imágenes de alta resolución, en cuyo caso deben de usarse ventanas de vecindad de gran tamaño. En la siguiente sección se analiza la técnica de fotometría estéreo que, por lo regular, es más eficiente en la reconstrucción de objetos tridimensionales, pero con la clara desventaja que para obtener un modelo fiel a la realidad se deben de hacer por lo menos tres tomas del mismo objeto.

2.1.2 Estéreo fotométrico

El principio fundamental de la fotometría estéreo es el de aproximar la forma de las superficies a partir de los vectores normales provenientes del patrón de sombras dependientes de la dirección de iluminación. En la actualidad, se han desarrollado algoritmos para solucionar el problema de la obtención de forma a partir del sombreado, entre los métodos más comunes se encuentran los de tira característica [Horn, 1970; Horn, 1975, Frankot y Chellappa, 1989], de Fourier [Pentland, 1988] y el de control óptimo [Dupuis y Oliensis, 1992].

Para este modelo debe asumirse que se tiene una proyección ortogonal, una superficie Lambertiana (de reflectancia constante) y una fuente puntual de luz distante. El mapa de reflectancia es entonces una función que caracteriza la relación entre la irradiancia de la imagen y la orientación de la superficie del objeto.

El mapa de reflectancia $R(p, q)$ es una función no lineal que puede considerarse como líneas de contorno en el espacio gradiente (p, q) . La ecuación básica para el proceso de formación de imagen se expresa como

$$E(x, y) = R(p, q) \quad (2.10)$$

que es conocida como la ecuación de imagen-irradiancia. Puede derivarse que

$$R(p, q) = \begin{cases} \eta \mathbf{l}^T \mathbf{n} & \mathbf{l}^T \mathbf{n} \geq 0 \\ 0 & \mathbf{l}^T \mathbf{n} < 0 \end{cases} \quad (2.11)$$

en donde η es el albedo de la superficie,

$$\mathbf{n} = \frac{(-p, -q, 1)^T}{\sqrt{1 + p^2 + q^2}} \quad (2.12)$$

\mathbf{n} es la normal a la superficie, p y q representan las derivadas de la profundidad de superficie z ,

$$\mathbf{l} = (\cos \tau \sin \sigma, \sin \tau \sin \sigma, \cos \sigma)^T \quad (2.13)$$

es el vector unitario de la dirección de iluminación apuntando hacia la fuente de luz, y τ y σ son los ángulos de inclinación y de punto de vista que se crean con los ejes x y z , respectivamente (véase figura 2.4). Esto también pueden representarse como

$$\mathbf{l} = \frac{(-p_s, -q_s, 1)^T}{\sqrt{1 + p_s^2 + q_s^2}} \quad (2.14)$$

donde p_s y q_s denotan la pendiente de un elemento en la superficie perpendicular a la dirección de iluminación.

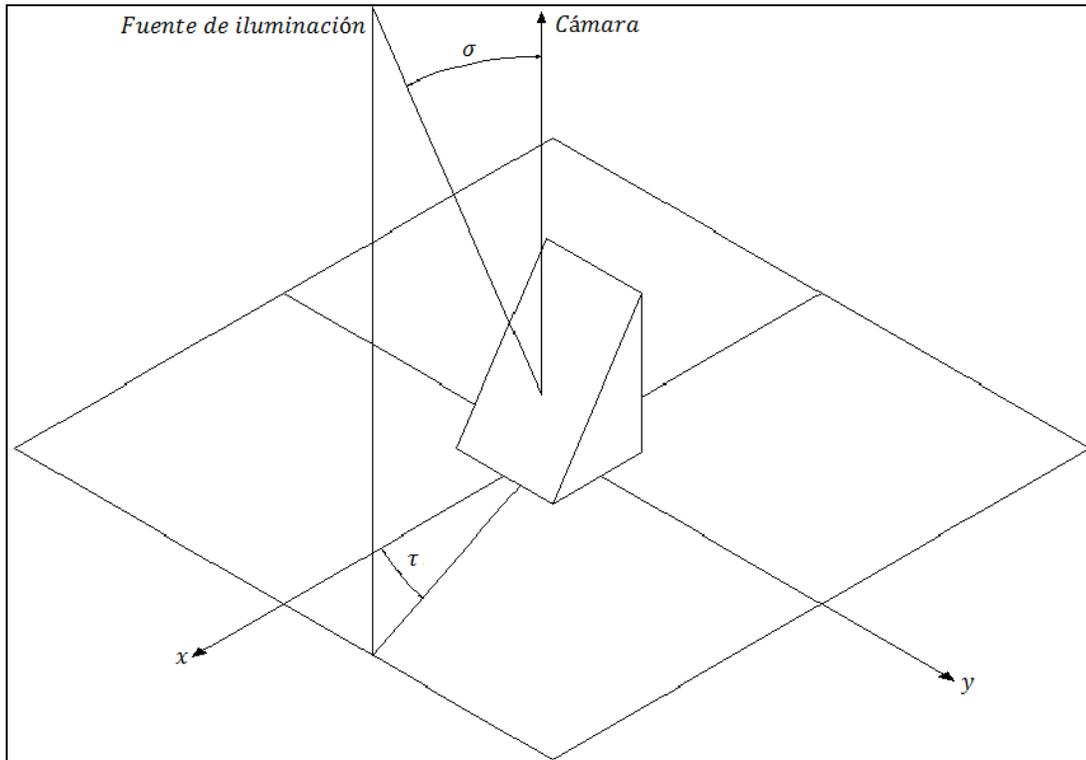


Figura 2.4 Geometría de la iluminación y punto de vista de un punto en la superficie

El estéreo fotométrico convencional es un método algebraico para resolver la ecuación de imagen-irradiancia. Dadas el albedo η , la dirección de iluminación \mathbf{l} , y la imagen de irradiancia E , resta sólo encontrar las incógnitas p y q , así que se requieren al menos dos ecuaciones para solucionar el sistema. Sin embargo, como la ecuación 2.10 es no lineal, puede tener más de una solución y a menudo se necesita una tercera imagen para formar un sistema sobre determinado.

Basándose en el mapa de reflectancia (Ec. 2.11), se puede tener un punto de vista simple para comprender el proceso de solución. Supóngase que se tienen tres imágenes con direcciones de iluminación \mathbf{l}_1 , \mathbf{l}_2 y \mathbf{l}_3 . En algún punto (x, y) las irradiancias de estas tres imágenes son E_1 , E_2 y E_3 , respectivamente. Se tienen entonces tres ecuaciones de imagen-irradiancia

$$E_i = \eta \mathbf{l}_i^T \mathbf{n}_i \quad i = 1, 2, 3 \quad (2.15)$$

donde \mathbf{n} es la normal a la superficie en el punto (x, y) . Estas ecuaciones pueden reescribirse en forma matricial como

$$\mathbf{E} = \eta \mathbf{L} \mathbf{n} \quad (2.16)$$

donde

$$\mathbf{E} = \begin{bmatrix} E_1 \\ E_2 \\ E_3 \end{bmatrix}, \quad \mathbf{L} = \begin{bmatrix} \mathbf{l}_1^T \\ \mathbf{l}_2^T \\ \mathbf{l}_3^T \end{bmatrix} \quad (2.17)$$

Si \mathbf{l}_1 , \mathbf{l}_2 y \mathbf{l}_3 son linealmente independientes, y \mathbf{L}^{-1} existe, entonces la ecuación 2.15 puede resolverse para \mathbf{n} . Es este punto es importante hacer notar que \mathbf{n} es un vector unitario con dos variables libres y por tanto 2.15 se convierte en un sistema sobre determinado y puede obtenerse su solución por mínimos cuadrados.

Una característica importante del sistema mostrado es que puede aplicarse aún si no se tiene conocimiento *a priori* del albedo η , al resolver 2.15 se obtiene

$$\eta \mathbf{n} = \mathbf{L}^{-1} \mathbf{E} \quad (2.18)$$

La magnitud del lado derecho de 2.17 da el valor del albedo en (x, y) ,

$$\eta = \|\mathbf{L}^{-1} \mathbf{E}\| \quad (2.19)$$

y la normal unitaria correspondiente es entonces

$$\mathbf{n} = \frac{1}{\eta} \mathbf{L}^{-1} \mathbf{E} \quad (2.20)$$

Dado que este método determina la orientación de la superficie en lugar de su profundidad, deben de integrarse las orientaciones para obtener la altura en cada punto. Como las orientaciones de superficie pueden no ser consistentes, puede no existir una superficie apropiada para la reconstrucción. Esto se conoce como el problema de integrabilidad [Frankot y Chellappa, 1989; Horn, 1990].

Antes de integrar las superficies, es necesario conocer los mapas de gradientes en las direcciones x y y , estos pueden calcularse a partir del vector normal a cada punto de la superficie y están denotados por p y q . Si el vector normal está definido como $\mathbf{n} = [n_x, n_y, n_z]$, entonces los mapas de gradientes son

$$p = \frac{n_x}{n_z}, \quad q = -\frac{n_y}{n_z} \quad (2.21)$$

En general, para obtener la altura z a partir de p y q , se utiliza [Horn y Brooks, 1986]

$$z(x, y) = z(x_0, y_0) + \int_{\Gamma} (p dx + q dy) \quad (2.22)$$

en donde $z(x, y)$ denota la altura en el punto (x, y) a través de la ruta de integración Γ desde el punto de referencia $z(x_0, y_0)$.

La expresión 2.22 puede resolverse de manera iterativa si se calculan nuevos mapas de gradiente para cada nueva iteración y se recalcula la altura hasta minimizar el error a un rango aceptable. Los mapas de gradiente en el punto (x, y) para la iteración $n + 1$ estarán dados por

$$\begin{aligned} p^{n+1}(x, y) &= \frac{1}{4} [p^n(x, y + 1) + p^n(x - 1, y) + p^n(x + 1, y) + p^n(x, y + 1)] \\ q^{n+1}(x, y) &= \frac{1}{4} [q^n(x, y + 1) + q^n(x - 1, y) + q^n(x + 1, y) + q^n(x, y + 1)] \end{aligned} \quad (2.23)$$

Mientras que la profundidad después de cada iteración n será

$$z^{n+1}(x, y) = z^n(x, y) + \int (p^n dx + q^n dy) \quad (2.24)$$

En las figuras 2.5 se presenta una simulación de la técnica de fotometría estéreo con vectores de iluminación $L_1 = [0.3, -0.12, 0.94]$, $L_2 = [-0.3, -0.12, 0.94]$ y $L_3 = [-0.1, 0.32, 0.94]$ sobre una superficie esférica generada de manera sintética en MATLAB. Las imágenes mostradas en la figura 2.6 son los mapas de gradiente y se obtienen al aplicar el desarrollo mostrado en 2.15 – 2.21. Para la reconstrucción de la superficie mostrado en la figura 2.7 se utilizó el método de integración por minimización de energía que considera la integración de puntos vecinos para obtener una superficie suave [Horn, 1986] como se mostró en 2.22 – 2.24.

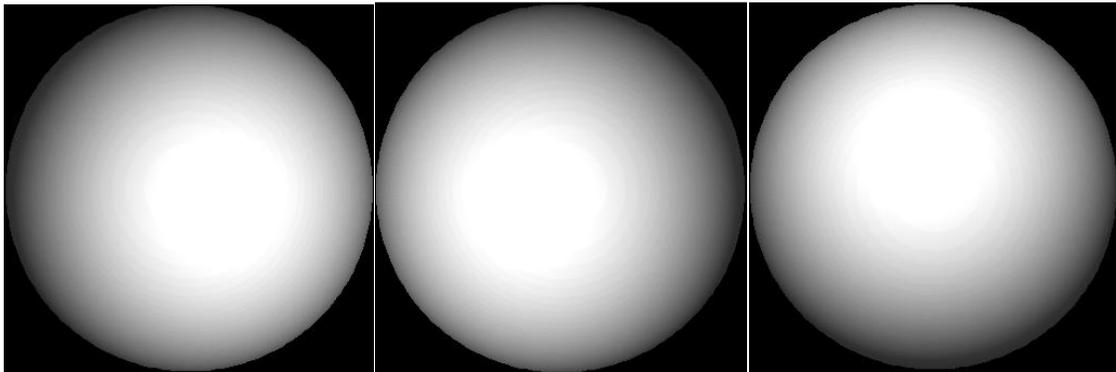


Figura 2.5 Patrones de sombra sobre un objeto con vectores de iluminación a) $L_1 = [0.3, -0.12, 0.94]$, b) $L_2 = [-0.3, -0.12, 0.94]$ y c) $L_3 = [-0.1, 0.32, 0.94]$

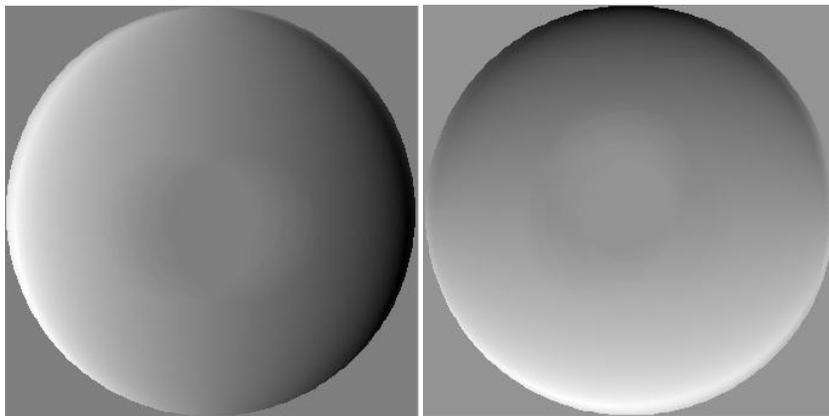


Figura 2.6 Mapas de derivadas de x e y obtenidas de los patrones de sombras con las ecuaciones 2.21

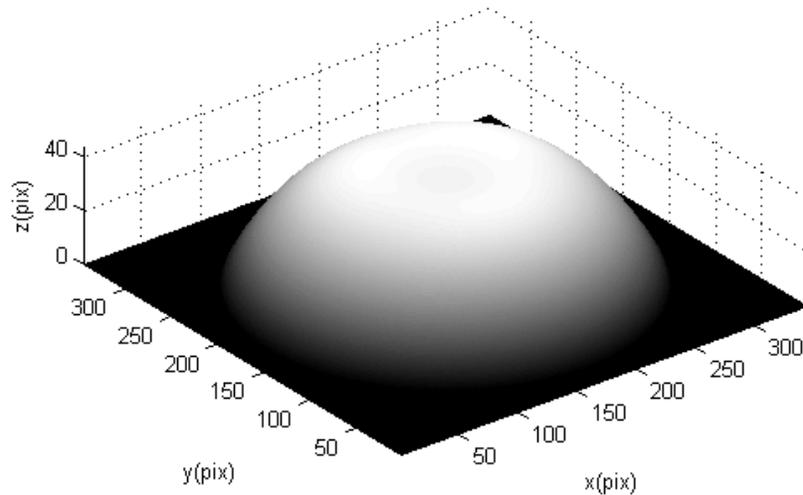


Figura 2.7 Superficie reconstruida al integrar los mapas de derivadas obtenidos con la ecuación 2.24 después de 5,000 iteraciones

La técnica de fotometría estéreo cuenta con la desventaja de requerir al menos tres imágenes para sobre determinar el sistema y resolverlo sin ambigüedades. Sin embargo, los procesos de obtención de gradiente y de integración son relativamente rápidos en comparación con la búsqueda de pares conjugados involucrada en la fotografía estéreo, aunque tampoco es recomendable utilizarla en aplicaciones de tiempo real a menos que se cuente con una tabla de búsqueda para grupos de intensidades en cada imagen.

En la siguiente sección se analizan algunos de los métodos de recuperación de forma a través de técnicas que utilizan luz estructurada. Se hace especial énfasis en las técnicas de proyección de franjas, dada la relativamente amplia superficie que puede obtenerse en comparación con otras técnicas. Se explican también algunas técnicas de demodulación de fase y se presenta un método para hacer más eficiente esta tarea.

2.2 Técnicas que utilizan luz estructurada

A la proyección de patrones de luz sobre una escena se le conoce como *luz estructurada*. Se proyectan patrones de luz sobre objetos que se encuentran el campo visual de la cámara. La distancia de un objeto a la cámara o la localización de un objeto en el espacio pueden determinarse al analizar los patrones observados en las imágenes.

La técnica más simple para medir distancias entre el sistema de adquisición y puntos en una superficie es la proyección de un punto, es decir, un solo haz de luz que se modela como un rayo o línea. El rayo proyectado y el haz de luz se interceptan para encontrar la posición del objeto iluminado en el espacio tridimensional.

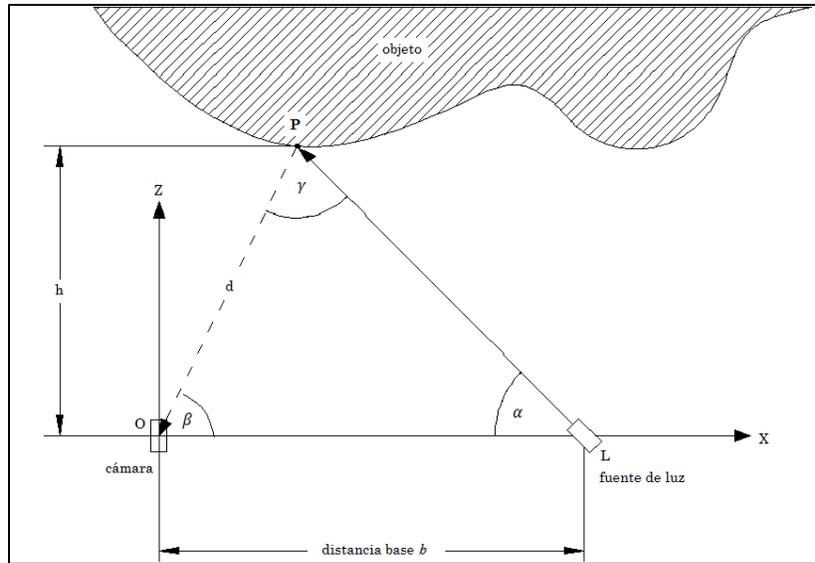


Figura 2.8 Diagrama de la proyección de un punto

Considérese el arreglo mostrado en la figura 2.8, en donde el objeto, la fuente de luz y la cámara se encuentran en el mismo plano. Sea O el centro de proyección de la cámara y el origen del sistema coordenado. La distancia base b se asume constante y conocida. La distancia d entre la cámara y el punto objeto $P = (X_o, Z_o)$ se calcula usando la ley de senos

$$\frac{d}{\sin \alpha} = \frac{b}{\sin \gamma} \quad (2.25)$$

de las relaciones $\gamma = \pi - (\alpha + \beta)$ y $\sin(\pi - \gamma) = \sin \gamma$ se obtiene que

$$\frac{d}{\sin \alpha} = \frac{b}{\sin(\pi - \gamma)} = \frac{b}{\sin(\alpha + \beta)} \quad (2.26)$$

entonces, la distancia d está dada por

$$d = \frac{b \cdot \sin \alpha}{\sin(\alpha + \beta)} \quad (2.27)$$

La ubicación del punto $P = (X_o, Z_o)$ puede representarse en el sistema de coordenadas de cámara por las dos coordenadas polares (d, β) . La transformación al sistema cartesiano está dada por

$$X_o = d \cdot \cos(\beta) \quad y \quad Z_o = h = d \cdot \sin(\beta) \quad (2.28)$$

El eje z coincide con el eje óptico de la cámara y el plano imagen se encuentra en $Z = f$, en donde f es la distancia focal de la cámara. La distancia Z puede encontrarse entonces usando el ángulo β .

El caso general de una triangulación tridimensional requiere que se incluya la dimensión vertical, en este caso, el eje y . Se asumen un sistema XYZ centrado en la cámara y su plano imagen ubicado

en $Z = f$. El punto objeto $\mathbf{P} = (X_o, Y_o, Z_o)$ es proyectado en un punto $p = (x, y)$ en el plano imagen.

En el caso tridimensional la cámara y la fuente de iluminación pueden colocarse en posiciones arbitrarias. Una vez más, el eje z es el eje óptico de la cámara y se asume que el centro óptico de la fuente de luz está ubicado en el eje x . Si se hace un análisis similar al caso 2D se obtiene

$$\frac{X_o}{x} = \frac{Z_o}{f} = \frac{Y_o}{y} \quad (2.29)$$

para el sistema centrado en la cámara. Usando trigonometría se tiene que

$$\tan \alpha = \frac{Z_o}{b - X_o} \quad y \quad Z_o = \frac{X_o}{x} \cdot f = \tan \alpha \cdot (b - X_o) \quad (2.30)$$

que resulta en

$$Z_o = X_o \left(\frac{f}{x} + \tan \alpha \right) = \tan \alpha \cdot b \quad (2.31)$$

La posición tridimensional del punto \mathbf{P} puede calcularse entonces por

$$X_o = \frac{\tan \alpha \cdot b \cdot x}{f + x \cdot \tan \alpha}, \quad Y_o = \frac{\tan \alpha \cdot b \cdot y}{f + x \cdot \tan \alpha}, \quad Z_o = \frac{\tan \alpha \cdot b \cdot f}{f + x \cdot \tan \alpha} \quad (2.32)$$

Así, es posible calcular la ubicación de cualquier punto a partir del conocimiento de las posiciones de la cámara y de la fuente de luz, así como la ubicación de un punto en el plano imagen $p = (x, y)$, esta técnica simplifica el proceso de búsqueda de puntos conjugados.

La técnica de proyección de una franja de luz representa una extensión de la técnica de proyección de un punto. En esta, se proyecta una hoja de luz sobre el objeto; la idea es intersecar el rayo proyectado del punto imagen examinado con la hoja de luz. La intersección del plano con la superficie del objeto se visualiza como una franja de luz en la imagen. Por lo tanto, una mayor cantidad de puntos pueden recuperarse de una sola imagen, lo que resulta en una reconstrucción más rápida comparado con la técnica de proyección de un punto.

2.2.1 Proyección de franjas con perfil cosenoidal

El método de proyección de franjas tiene la ventaja de que requiere, en algunas técnicas, de una sola imagen para llevar a cabo una reconstrucción adecuada del objeto bajo observación. Cuenta además con la ventaja de que pueden medirse objetos de distintas dimensiones a través de la modificación de la frecuencia de las franjas [Joenathan, 1992; Nadeborn y Andrä, 1996], con las únicas limitantes de las resoluciones tanto del sistema de proyección como de la cámara.

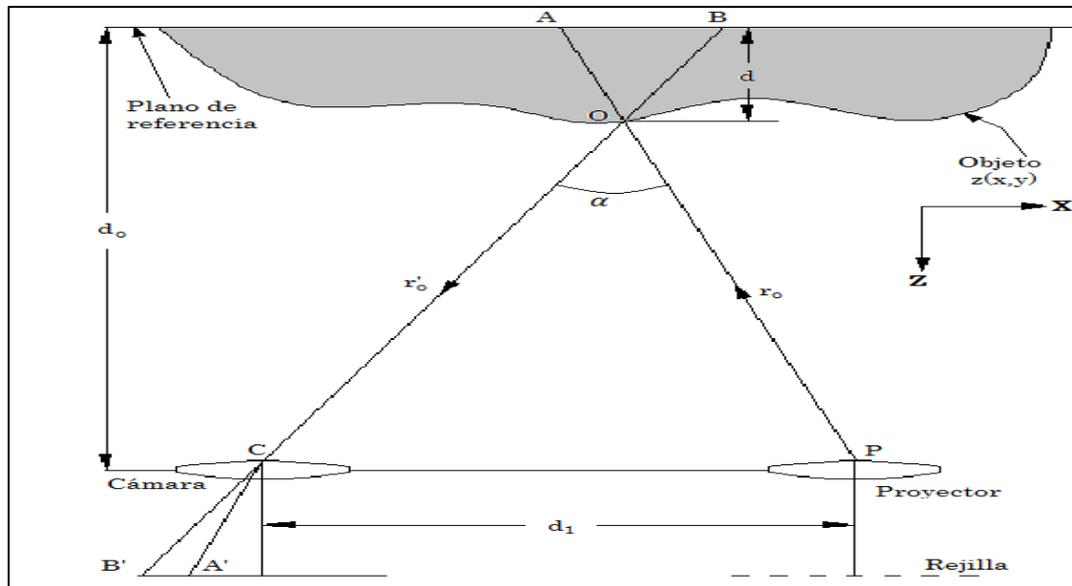


Figura 2.9 Geometría de la proyección de franjas

La relación entre la fase y la intensidad observada en un patrón de perfil cosenoidal puede aproximarse por la siguiente expresión

$$I(x, y) = a(x, y) + b(x, y) \cdot \cos(\varphi(x, y)) \quad (2.33)$$

en donde $a(x, y)$ describe la iluminación de fondo, $b(x, y)$ representa las variaciones de reflectancia del objeto y $\varphi(x, y)$ está dada por

$$\varphi(x, y) = 2\pi f_o(x, y) + \phi(x, y) \quad (2.34)$$

donde f_o corresponde a la frecuencia espacial de la señal portadora y $\phi(x, y)$ es el término asociado de fase, mismo que será utilizado para aproximar la forma del objeto [Quan *et al*, 2007]. Los algoritmos de extracción de fase, como el de la transformada de Fourier [Takeda *et al*, 1982; Takeda y Mutoh; 1983] y el de corrimiento de fase [Snirivasan *et al*, 1984; Morgan, 1982; Greivenkamp, 1984], pueden utilizarse para extraer la fase envuelta de uno o varios patrones de franjas. Un proceso de desenvolvimiento de fase puede usarse después para obtener una distribución continua de fase conteniendo tanto la fase relacionada con la forma del objeto $\varphi(x, y)$, como la fase relacionada con el término de la señal portadora $2\pi f_o(x, y)$.

Considérese el caso en que los ejes ópticos de la rejilla y de la cámara son paralelos a lo largo del eje z (figura 2.9), y en que los ejes x y y son horizontal y vertical al plano de referencia ($z = 0$), respectivamente. Las líneas de la rejilla proyectada son también paralelas al plano de referencia. Los centros de las pupilas del proyector P y la cámara C están situados a una distancia d_o del plano de referencia y se localizan sobre la misma posición z .

Si se coloca un objeto sobre el plano de referencia, se puede calcular la profundidad del mismo utilizando las técnicas de extracción de fase mencionadas anteriormente. Si la distancia d_o es relativamente grande con respecto a d_1 , entonces la rejilla proyectada sobre el objeto será

irregular con cambios de frecuencia en la dirección x , en cuyo caso la fase observada en **O** (ϕ_O) será muy similar a la fase en **A** (ϕ_A) y los puntos **O** y **B** sobre el objeto se proyectarán en el plano imagen en el mismo punto **B'**. La distancia **AB** puede expresarse entonces como

$$\overline{AB} = \frac{\phi_B - \phi_O}{2\pi f} \quad (2.35)$$

donde f es la frecuencia espacial de la rejilla sobre el plano de referencia y ϕ_B es la fase en el punto **B**. Puede hacerse notar de la ecuación 2.35 que la distancia **AB** es proporcional a la diferencia de fase producida por la presencia del objeto ya que en lugar de aparecer un punto proyectado cualquiera r_o como si estuviera en el plano de referencia (en la posición **A**), este se muestra en el punto **B**. Si se analizan los triángulos **PCO** y **ABO** en la figura 2.9, se obtiene la relación

$$z(x, y) = \frac{\overline{AB} \cdot d_o}{d_1 + \overline{AB}} \quad (2.36)$$

Al sustituir la ecuación 2.27 en 2.28 se obtiene la profundidad $z(x, y)$ en términos de la fase y esta queda como

$$z(x, y) = \frac{(\phi_B - \phi_O) \cdot d_o}{2\pi f d_1 + (\phi_B - \phi_O)} \quad (2.37)$$

Si se expresa el periodo de la rejilla proyectada como $p = 1/f$ y se hace $\Delta\phi = \phi_B - \phi_O$, entonces la ecuación anterior puede expresarse en función del ángulo α entre el proyector y la cámara [Lilley, 1999]

$$z(x, y) = \frac{\Delta\phi(x, y)}{2\pi} \frac{p}{\sin \alpha} \quad (2.38)$$

Debe de hacerse notar que si el patrón se genera de forma sintética mediante software, el periodo p aún debe de sufrir una transformación entre coordenadas imagen a coordenadas de mundo, esta tarea puede llevarse a cabo mediante una adecuada calibración del sistema cámara-proyector.

2.2.2 Métodos de recuperación de fase

Como ya se mencionó antes, una parte fundamental de la topometría a través del contorno por medio de la proyección de un patrón de franjas con perfil cosenoidal, es la correcta adquisición de la fase, ya que esta se encuentra asociada directamente con la profundidad de un punto en el objeto. En la siguiente sección se describen algunas técnicas de recuperación de fase, comenzando por el método directo y continuando con los métodos de Fourier y de corrimiento de fase.

2.2.2.1 Método de demodulación directa

La demodulación directa de fase o demodulación espacial síncrona fue desarrollada por Ichioka e Inuiya [1972] y aplicada más tarde por Womack [1984]. Ellos utilizaron una técnica llamada demodulación de cuadratura derivada de la teoría de comunicaciones [Cuevas, 2000] para la determinación tridimensional de la forma de un objeto a partir de la fase recuperada de la proyección de franjas sobre el mismo.

Considérese una vez más la ecuación 2.33, dada por

$$I(x, y) = a(x, y) + b(x, y) \cdot \cos(\varphi(x, y))$$

Tanto la iluminación de fondo, como la reflectancia del objeto pueden eliminarse al aplicar filtros apropiados en el dominio frecuencial [Muñoz-Maciel *et al*, 2007]. Esto se logra mediante el proceso conocido como normalización [Quiroga *et al*, 2001]. De esta manera, el patrón normalizado $I_N(x, y)$ contiene sólo el coseno de la fase

$$I_N(x, y) = \cos[\varphi(x, y)] \quad (2.39)$$

Con la técnica espacial síncrona el patrón normalizado de franjas es multiplicado por el coseno y el seno de una referencia cercana $\psi(x, y)$ quedando

$$\begin{aligned} I_C(x, y) &= I_N(x, y) \cos[\psi(x, y)] \\ &= \frac{1}{2} \{ \cos[\varphi(x, y) - \psi(x, y)] + \cos[\varphi(x, y) + \psi(x, y)] \}, \\ I_S(x, y) &= I_N(x, y) \sen[\psi(x, y)] \\ &= \frac{1}{2} \{ -\sen[\varphi(x, y) - \psi(x, y)] + \sen[\varphi(x, y) + \psi(x, y)] \} \end{aligned} \quad (2.40)$$

Para eliminar los términos alta frecuencia correspondientes al seno y al coseno de la suma de $\varphi(x, y)$ y $\psi(x, y)$, las intensidades I_C e I_S se filtran con un pasa-bajos. La diferencia envuelta se encuentra al aplicar la función arco tangente a las intensidades filtradas, esta se obtiene en un rango de $[-\pi, \pi]$:

$$\Phi_w(x, y) = \arctan \left[\frac{-\tilde{I}_S}{\tilde{I}_C} \right] \quad (2.41)$$

en donde \tilde{I} representa el filtro pasa-bajos aplicado sobre las señales pre multiplicadas I_C e I_S . La fase envuelta es calculada entonces con

$$\varphi_w(x, y) = \arctan \left\{ \frac{\sen[\Phi_w(x, y) + \psi(x, y)]}{\cos[\Phi_w(x, y) + \psi(x, y)]} \right\} \quad (2.42)$$

Hace falta hacer notar que en este punto aún hace falta aplicar un método de desenvolvimiento de fase como se muestra en la figura 2.10.

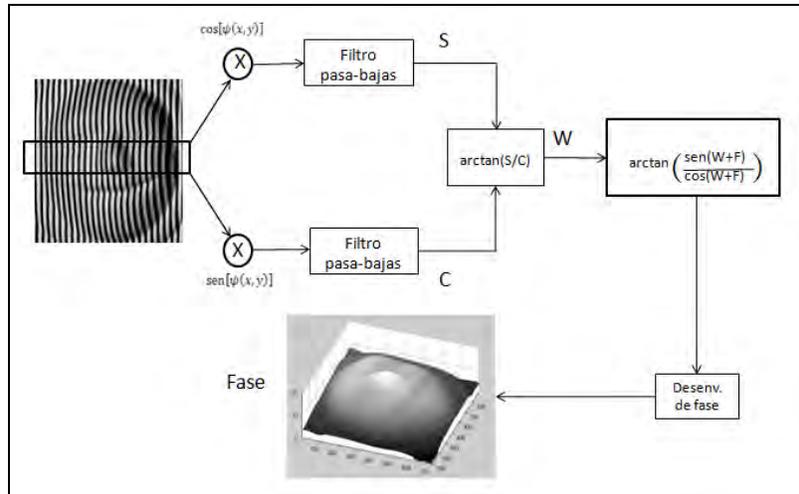


Figura 2.10 Esquema de aplicación del método de demodulación directa

2.2.2.2 Método de Fourier-Takeda

El uso de técnicas de análisis espacial como las transformadas wavelet y de Fourier se han vuelto muy populares en los últimos tiempos debido a que puede obtenerse la información de fase a partir de una sola imagen [Langoju *et al*, 2005]. La técnica de la transformada de Fourier se ha aplicado ampliamente en el procesamiento de señales e imágenes al transformar información de intensidad al dominio de Fourier o dominio de frecuencia. Para aplicar la transformada de Fourier a señales interferométricas se introduce una señal portadora para separar la señal de primer orden de los componentes de orden cero en el dominio frecuencial.

En algunas mediciones ópticas, se encuentra un patrón de franjas representado de la forma

$$g(x, y) = a(x, y) + b(x, y) \cos[2\pi f_0 x + \phi(x, y)] \quad (2.43)$$

donde la fase $\phi(x, y)$ contiene la información deseada y los términos $a(x, y)$ y $b(x, y)$ desean eliminarse; en la mayoría de los casos $a(x, y)$, $b(x, y)$ y $\phi(x, y)$ varían lentamente comparados con el término de frecuencia portadora f_0 .

En la técnica de la transformada de Fourier propuesta por Takeda [Takeda *et al*, 1982] se toma el patrón de franjas adquirido, que por conveniencia se reescribe como

$$g(x, y) = a(x, y) + c(x, y) \exp(2\pi i f_0 x) + c^*(x, y) \exp(-2\pi i f_0 x) \quad (2.44)$$

con

$$c(x, y) = \frac{1}{2} b(x, y) \exp[i\phi(x, y)] \quad (2.45)$$

donde * denota el conjugado complejo.

Después se aplica una transformación con respecto a x de 2.44 por medio de la transformada rápida de Fourier (FFT), lo cual da como resultado

$$G(f, y) = A(f, y) + C(f - f_o, y) + C^*(f + f_o, y) \quad (2.46)$$

en donde las letras mayúsculas denotan el espectro de Fourier y f es la frecuencia espacial en la dirección x . Como las variaciones espaciales de $a(x, y)$, $b(x, y)$ y $\phi(x, y)$ son lentas en comparación con la frecuencia espacial f_o , los espectros de Fourier en 2.46 se separan como se muestra en la figura 2.11(a).

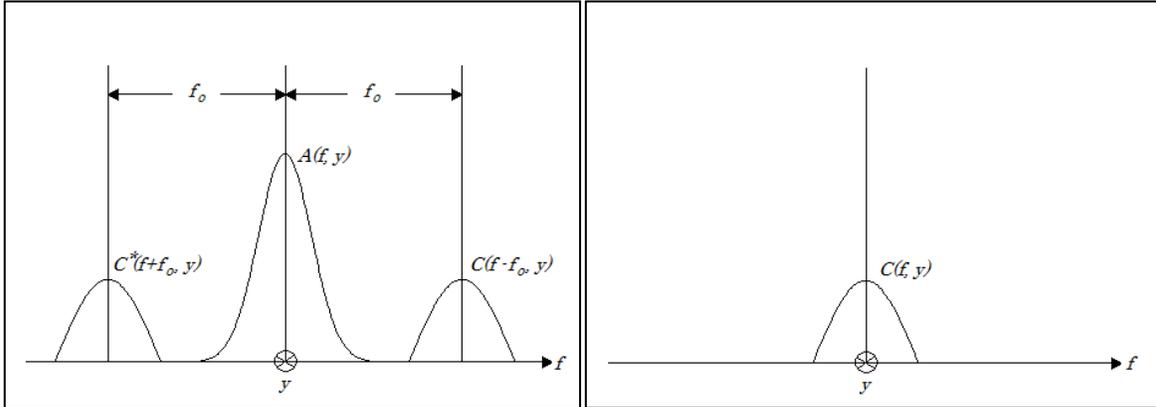


Figura 2.11 (a) Espectro de Fourier de un patrón de franjas y ; (b) espectro seleccionado y trasladado al origen

Se hace uso de cualquiera de los dos espectros correspondientes a la señal portadora, $C(f - f_o, y)$ por ejemplo, y se traslada al origen una distancia f_o a lo largo del eje f para obtener $C(f, y)$ como se muestra en la figura 2.11(b). Nótese que la variación de fondo $a(x, y)$ ya se ha filtrado en esta etapa. Ahora se computa la transformada inversa de Fourier para $C(f, y)$ con respecto a f para obtener $c(x, y)$, definido por la ecuación 2.45. Al calcular el logaritmo de $c(x, y)$ se obtiene

$$\log[c(x, y)] = \log\left[\frac{1}{2}b(x, y)\right] + i\phi(x, y) \quad (2.47)$$

Ahora se tiene la fase $\phi(x, y)$ en la parte imaginaria completamente separada del término $b(x, y)$ en la parte real.

Un segundo enfoque para obtener la fase a partir de $c(x, y)$ es aplicar la función

$$\phi(x, y) = \arctan\frac{\text{Im}(c(x, y))}{\text{Re}(c(x, y))} \quad (2.48)$$

La fase obtenida se encuentra indeterminada a un factor de 2π , es decir se encuentra envuelta y, al igual que en el caso del método directo, es necesario aplicar algún algoritmo de desenvolvimiento de fase.

2.2.2.3 Método de corrimiento de fase

El método de corrimiento de fase es el más popular de los métodos que requieren múltiples imágenes [Yoshizawa, 2009]. En este método, se proyectan N patrones de franjas con la fase inicial modificada por un valor $\Delta\psi$. Estas imágenes con cambio de fase se obtienen en tiempos o espacios de muestreo diferentes.

Para el número de muestras $k = 1, 2, \dots, N$, el cambio de fase está dado normalmente por

$$\Delta\psi = \frac{2\pi}{N} \quad (2.49)$$

Para cada patrón de franjas, se adquieren las N imágenes denotadas por

$$I_k(x, y) = a(x, y) + b(x, y) \cos[\phi(x, y) + (k + 1)\Delta\phi] \quad (2.50)$$

Al usar todas las imágenes, la relación entre el mapa de fase $\phi(x, y)$ y los patrones de intensidad en cada pixel se deducen por la solución de mínimos cuadrados

$$\tan \phi(x, y) = \frac{\sum_{k=1}^N I_k(x, y) \sin[(k - 1)\Delta\psi]}{\sum_{k=1}^N I_k(x, y) \cos[(k - 1)\Delta\psi]} \quad (2.51)$$

En principio, entre mayor sea el número de N muestras tomadas menor es el nivel de ruido aleatorio, que decrece por un factor $1/\sqrt{N}$ y da lugar a medidas muy precisas. En la práctica se utilizan entre tres y siete patrones de franjas.

El algoritmo básico utilizado corresponde al caso en que se utilizan tres pasos ($N = 3$), con un corrimiento de fase $\Delta\psi = \pi/2$. En este caso el mapa de fase se recupera con la ecuación [Wyant *et al*, 1984]

$$\phi(x, y) = \tan^{-1} \left[\frac{I_3(x, y) - I_2(x, y)}{I_1(x, y) - I_2(x, y)} \right] \quad (2.52)$$

Aún así, el número de pasos más utilizado es probablemente $N = 4$ con $\Delta\psi = \pi/2$. Para este método se han desarrollado distintas técnicas de recuperación de fase, entre ellas los métodos de cuatro pasos [Creath, 1988], el de Carré [1966] y el método de cinco pasos ($4 + 1$) [Hariharan, 1987], siendo el primero el más conocido y cuyo mapa de fase está dado por

$$\phi(x, y) = \tan^{-1} \frac{I_4(x, y) - I_2(x, y)}{I_1(x, y) - I_3(x, y)} \quad (2.53)$$

La ventaja del método de corrimiento de fase es la operación pixel por pixel para obtener el valor de fase a medir, para el cual no se sacrifica resolución espacial y la precisión del cálculo de la fase puede mejorarse fácilmente al incrementar el número de imágenes muestreadas. Por otro lado, dada la necesidad de adquirir múltiples imágenes, este método se encuentra altamente influenciado por factores exteriores como vibraciones y no es recomendable para mediciones de objetos móviles. Un análisis completo del error involucrado en múltiples métodos de proyección

de franjas es presentado por Novák *et al* en [39], incluido el método de Carré y el método de N+1 pasos [Surrel, 1993].

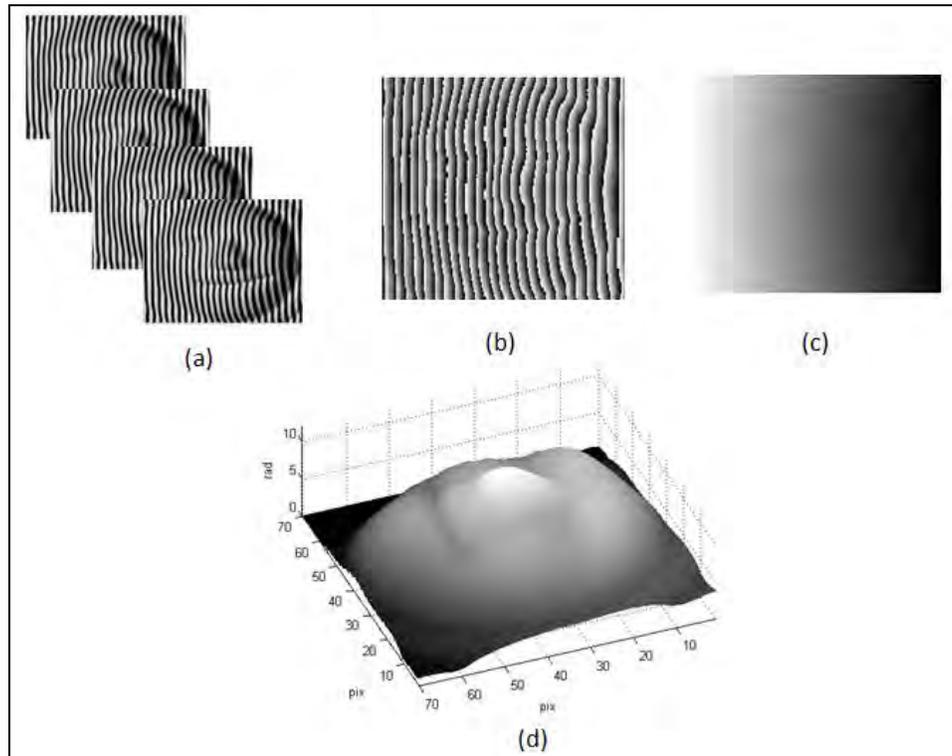


Figura 2.12 Pasos necesarios para el método de corrimiento de fase. (a) Imágenes de entrada; (b) fase envuelta obtenida a partir de la ecuación 2.37; (c) fase desenvuelta y; (c) reconstrucción del objeto

Al igual que en las técnicas anteriores, el método de corrimiento de fase también entrega la fase envuelta y requiere de un desenvolvimiento. En la siguiente sección se describe el concepto de 'fase envuelta' y se muestran algunos métodos para el desenvolvimiento de la misma.

2.2.3 Técnicas de desenvolvimiento de fase

En los métodos de recuperación de fase descritos en las secciones anteriores se ve involucrada la función arco tangente, este hecho induce la restricción de que la fase obtenida se encuentre en el rango $[-\pi, \pi]$, lo que lleva a discontinuidades o saltos de fase en píxeles adyacentes como se muestra en la figuras 2.12(b) y 2.13(a), a este tipo de diagrama se le conoce como '*fase envuelta*'.

Las discontinuidades en la fase ϕ se dan cada vez que esta presenta un cambio de 2π . Si ϕ incrementa, la pendiente de la curva es positiva y viceversa para el caso en que decrece. El término '*desenvolvimiento de fase*' tiene lugar dado que el paso final en la medición de patrones de franjas consiste en integrar la fase a lo largo de una línea o camino contando las discontinuidades de 2π y sumando esa misma cantidad cada vez que el ángulo de fase cambie de 2π a cero o restándosela si este cambia de cero a 2π . La figura 2.13(b) muestra la fase después de ser desenvuelta aplicando estas últimas condiciones.

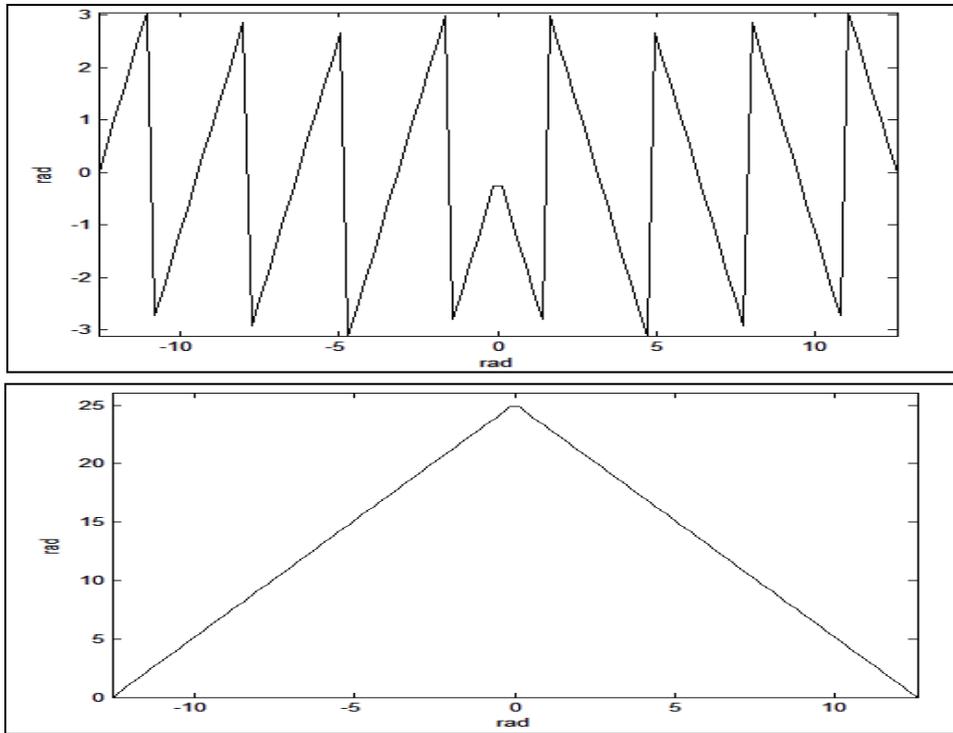


Figura 2.13 Diagramas de fase (a) envuelta y; (b) desenvuelta

El principio básico del desenvolvimiento de fase es el de integrar la fase envuelta ϕ a través de un camino. En cada pixel, el gradiente de fase se calcula con

$$\Delta\phi = \phi_n - \phi_{n-1} \quad (2.54)$$

en donde n es el índice del pixel bajo análisis. Si $|\Delta\phi|$ excede un cierto nivel de umbral ($K \cdot \pi$, por ejemplo), entonces se asume que existe un borde de franja. Este salto de fase se corrige con la adición o sustracción de 2π , dependiendo del signo de $\Delta\phi$. Esta operación puede resumirse de la siguiente manera

$$\begin{aligned} \text{Si} \quad & \phi_n - \phi_{n-1} > K \cdot \pi \\ & \phi_n = \phi_n - 2\pi \\ \text{Pero si} \quad & \phi_n - \phi_{n-1} \leq K \cdot \pi \\ & \phi_n = \phi_n + 2\pi \end{aligned} \quad (2.55)$$

Si el área de la fase efectiva se encuentra en coordenadas rectangulares $x - y$, entonces puede obtenerse un resultado apropiado de la combinación de desenvolvimientos a lo largo de los caminos horizontales x y de los verticales y [Itoh, 1982].

Otro método basado en la integración de las diferencias locales consiste en utilizar las funciones $\text{sen}(x)$, $\text{cos}(x)$ y $\text{artan}(x)$ [Servin y Cuevas, 1995]. La integración puede expresarse de la siguiente manera

$$\phi = \int \Delta\phi_x dx \quad (2.56)$$

A partir de la expresión anterior, puede obtenerse la forma discreta para aplicar el desenvolvimiento que resulta en

$$\phi_d(x_i) = \phi_d(x_{i-1}) + atan \left[\frac{\text{sen}(\phi_e(x_{i-1}) - \phi_d(x_{i-1}))}{\text{cos}(\phi_e(x_{i-1}) - \phi_d(x_{i-1}))} \right] \quad (2.57)$$

en donde ϕ_d representa la fase desenvuelta, ϕ_e es la fase envuelta y el subíndice i denota la posición del píxel bajo análisis.

Los dos métodos de desenvolvimiento mencionados son fáciles de implementar y resultan particularmente útiles al desenvolver fases con variación en un solo sentido. Sin embargo, muchos patrones de franjas pueden encontrarse rodeados por vecindades más complejas y pueden contener píxeles considerados como puntos singulares, donde la modulación de franjas es muy baja o las diferencias de fase entre puntos vecinos es mayor a 2π . En estos casos, el camino de desenvolvimiento se ve interrumpido y la distribución de la fase desenvuelta se desvía de la verdadera fase. Para estos casos, se han propuesto varios algoritmos efectivos y robustos, algunos de estos caen en las siguientes categorías [Yoshizawa, 2009]:

1. Algoritmos seguidores: son métodos para obtener un camino de desenvolvimiento al atravesar el área de fase efectiva con trayectorias de tipo ramificadas que no necesariamente siguen un ciclo cerrado. Para seleccionar el camino más efectivo, se utilizan funciones de costo como el gradiente de fase y el contraste de las franjas. Se clasifican píxeles no confiables, que son removidos de las trayectorias. Cuentan con la particularidad de que son algoritmos muy rápidos [Judge *et al*, 1992; Takeda y Abe, 1996].
2. Algoritmos basados en minimización de energía: se realiza la suma de minimización de la suma de diferencias de fase entre puntos vecinos en que la distribución de fase inicial corresponde a un estado de energía máximo. Se busca la convergencia del estado mínimo de energía al variar las fases en todos los píxeles por $\pm 2\pi$ basándose en alguna regla. Este método requiere varias iteraciones y no depende de la ruta de desenvolvimiento. Se han propuesto diversos algoritmos como el método de recocido simulado [Huntley, 1989], el método de Euler-Poisson [Kerr *et al*, 1996], el método de celdas autómatas [Ghiglia *et al*, 1987] y el sistema regularizador del seguimiento de fase [Servin *et al*, 1998, Cuevas *et al*, 1999].

En este capítulo se repasaron algunas técnicas de medición de forma, se revisaron las ventajas y desventajas que presenta cada uno y se analizaron métodos para la demodulación de los patrones de franjas adquiridos por la proyección de franjas con perfil cosenoidal. Sin embargo, no se mencionó a fondo el hecho de que antes de utilizar cualquiera de las técnicas descritas, hace falta una etapa previa de calibración de cámara. En el capítulo 3 se mencionan distintas técnicas de calibración y se hace énfasis en las que se aplican a sistemas de proyección de franjas, dada su mayor relevancia con respecto a este trabajo de tesis.

Referencias

- [1] Carré, P., *'Installation et utilisation de comparateur photoelectrique et interferential'*, Bureau International des Poids et Mesures, Metrologia **2**: 13 – 23 (1966).
- [2] Creath, K., *'Phase-measurement interferometry techniques'*, Progress in Optics, Ed. E. Wolf (Elsevier Science Publishers B.V.Vol. **XXVI**, 1988), pp. 348 – 393.
- [3] Cuevas, F.J., Servin, M., Rodríguez-Vera, R., *'Depth object recovery using radial basis functions'*, Opt. Comm. **163**: 270 – 277 (1999).
- [4] Cuevas, F.J., *'Aplicación de redes neuronales en la reconstrucción tridimensional de objetos'*, Tesis doctoral, Centro de Investigaciones en Óptica, A.C., León, México (2000).
- [5] Dupuis, P., Oliensis, J., *'Direct method for reconstructing shape from shading'*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 453 – 458 (1992).
- [6] Frankot, R.P., Chellappa, R., *'A method for enforcing integrability in shape from shading algorithm'*, IEEE Trans. Pattern Anal. Mach. Intell., **10**: 439 – 451 (1989).
- [7] Ghiglia, D.C., Mastin, G.A., Romero, L.A., *'Cellular-automata method for phase unwrapping'*, J. Opt. Soc. Am. **4**: 267 – 280 (1987).
- [8] Greivenkamp, J.E., *'Generalized data reduction for heterodyne interferometry'*, Opt. Eng. **23**: 350 – 352 (1984).
- [9] Hariharan, P., Oreb, B.F., Eiju, T., *'Digital phase-shifting interferometry: A simple error-compensating phase calculation algorithm'*, Appl. Opt. **26**: 2504 – 2506 (1987).
- [10] Haralick, R.M., Shapiro, L.G., *'Computer and Robot Vision, Volume II'*, Addison-Wesley, 1993.
- [11] Horn, B.K.P., *'Shape from shading: a method for obtaining the shape of a smooth opaque object from one view'*, Ph. D. dissertation, Massachusetts Institute of Technology, Cambridge, Mass., 1970.
- [12] Horn, B.K.P., *'Obtaining Shape from Shading Information'*, MIT Press, Cambridge, Mass., 1975.
- [13] Horn, B.K.P., *'Robot Vision'*, MIT Press, 1986.
- [14] Horn, B.K.P., Brooks, M.J., *'The variational approach to shape from shading'*, Comput. Vision Graphics Image Process, **33**: 174 – 208 (1986).
- [15] Horn, B.K.P., *'Height and gradient from shading'*, International Journal of Computer Vision **5**: 584 – 595 (1990).
- [16] Huntley, J.M., *'Noise-immune phase unwrapping algorithm'*, Appl. Opt. **28**: 3268 – 3270 (1989).

- [17] Ichioka Y., Inuiya M., '*Direct phase detecting system*', Appl. Opt. **11**: 1507-1514 (1972).
- [18] Itoh, K., '*Analysis of the phase unwrapping algorithm*', Appl. Opt. **21**: 2470 (1982).
- [19] Joenathan, C., Khorana, B.M., '*Phase Measurement by Differentiating Interferometric Fringes*', Journal of Modern Optics **39**: 2075 – 2087 (1992).
- [20] Judge, T.R., Quan, C.G., Bryanstoncross, P.J., '*Holographic deformation measurements by Fourier-transform technique with automatic phase unwrapping*', Opt. Eng. **31**: 533 – 543 (1992).
- [21] Kerr, D., Kaufmann, G.H., Gilizzi, G., '*Unwrapping of interferometric phase-fringe maps by the discrete cosine transform*', Appl. Opt. **35**: 810 – 816 (1996).
- [22] Klette, R., Schlüns, K., Koschan, A., '*Computer Vision: Three-Dimensional Data from Images*', Springer, 1996.
- [23] Langoju, R., Patil, A., Rastogi, P., '*Super-resolution Fourier transform method in phase shifting interferometry*', Optics Express **13**: 7160 – 7173 (2005).
- [24] Lilley, F., '*An Optical 3-D Body Surface Measurement System to Improve Radiotherapy Treatment of Cancer*', Faculty of General Engineering, James Parsons Building, Liverpool. Liverpool John Moores University (1999).
- [25] Morgan, C.J., '*Least-squares estimation in phase-measurement interferometry*', Opt. Letters **7**: 368 – 370 (1982).
- [26] Muñoz-Maciel, J., Peña-Lecona, F.G., Castillo-Quevedo, C., Casillas-Rodríguez, F.J., Durán-Ramírez, V.M., Mora-González, M., Rodríguez-Zavala, J.G., '*Phase tracking with a spatial synchronous method*', Appl. Opt. **46**: 3462 – 3465 (2007).
- [27] Nadeborn, W., Andrä, P., '*A robust procedure for absolute phase measurement*', Optics and Lasers in Engineering **24**: 245 – 260 (1996).
- [28] Novák, J., Nývák, P., Miks, A., '*Multi step phase-shifting algorithms intensive to linear phase shift error*', Optics Communications **281**: 5302 – 5309 (2008).
- [29] Peng, Z., Guo-Qiang, N., '*Simultaneous perimeter measurement for 3D object with a binocular stereo vision measurement system*', Optics and Lasers in Engineering **48**: 505 – 511 (2010).
- [30] Pentland, A., '*Shape information from shading: a theory about human perception*', in Proceedings of the IEEE International Conference on Computer Vision, pp. 404 – 413 (1988).
- [31] Quan, C., Tay, C.J., Chen, L.J., '*A study on carrier-removal techniques in fringe projection profilometry*', Optics and Laser Technology **39**: 1155 – 1161 (2007).
- [32] Quiroga, J. A. Gómez-Pedrero, and A. García-Botella, "*Algorithm for fringe pattern normalization*," Opt. Commun. **197**: 43–51 (2001).

- [33] Servin, M., Cuevas, F.J., 'A novel technique for spatial phase-shifting interferometry', *Journal of Modern Optics* **42**: 1853-1862 (1995).
- [34] Servin, M., Marroquin, J.L., Malacara, D., Cuevas, F.J., 'Phase unwrapping with a regularized phase tracking system', *Appl. Opt.* **37**: 1917-1923 (1998).
- [35] Srinivasan, V., Liu, H.C., Haliou, M., 'Automated phase-measuring profilometry of 3D diffuse objects', *Appl. Opt.* **23**: 3105 – 3108 (1984).
- [36] Surrel, Y., 'Phase stepping: a new self calibrating algorithm', *Appl. Opt.* **32**: 3598 -3600 (1993).
- [37] Takeda, M., Ina, H., and Kobayashi, S., 'Fourier-transform method of fringe-pattern analysis for computer-based topography and interferometry', *J. Opt. Soc. Am.* **72**: 156-160 (1982).
- [38] Takeda, M., Mutoh, K., 'Fourier transform profilometry for the automatic measurement of 3D object shapes', *Appl. Opt.* **22**: 3977 – 3982 (1983).
- [39] Takeda, M., Abe, T., 'Phase unwrapping by a maximum cross-amplitude spanning tree algorithm: A comparative study', *Opt. Eng.* **35**: 2345 – 3251 (1996).
- [40] Womack, K. H., 'Interferometric phase measurement using spatial synchronous detection', *Opt. Eng.* **23**: 391-395 (1984).
- [41] Wyant, C.J., 'Interferometric optical metrology: basic principles and new systems', *Laser Focus*, May: 65 – 71 (1982).
- [42] Yoshizawa, T., 'Handbook of optical metrology: principles and applications', CRC Press, 2009.

Capítulo 3

Calibración de sistemas que utilizan visión artificial

En este capítulo se describen distintas formas de calibrar un sistema de adquisición basándose en la posición y orientación de los componentes. La calibración del sistema significa, en general, determinar los parámetros para realizar una conversión de la cantidad física a su interpretación en la unidad de procesamiento, en este caso una computadora. Al realizar este proceso se consideran también las aberraciones y errores introducidos tanto por la cámara como por el sistema de proyección, ambos relacionados con el uso de sistemas de lentes.

Para llevar a cabo la calibración deben de estimarse los parámetros internos de la cámara (formación de la imagen, aberraciones, etc.) y los parámetros externos inherentes al arreglo experimental (disposición geométrica de los distintos elementos del sistema). En las secciones siguientes se exponen los distintos tipos de arreglos experimentales utilizados en sistemas de proyección de franjas y se mencionan algunos métodos que solucionan algunos de los problemas no considerados en los dichos arreglos.

3.1 Cámara sin distorsión (*pinhole*)

La calibración de un sistema sin distorsión es un proceso para determinar las características ópticas y geométricas internas de la cámara (cinco parámetros intrínsecos) y la posición y orientación del sistema coordinado de la cámara en relación con el sistema mundo (seis parámetros extrínsecos) [Lin y Sung, 2007]. En este modelo idealizado pueden existir dos tipos de proyección: la proyección central o de perspectiva (figura 3.1a) y la proyección paralela, también conocida como ortogonal (figura 3.2b).

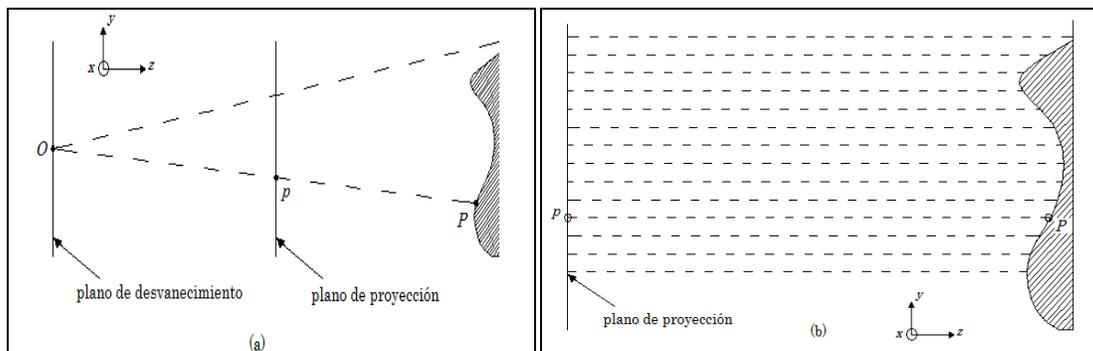


Figura 3.1 Proyecciones de un punto P a un punto p (a) central y; (b) paralela

Para el sistema ideal asumido, la proyección de los puntos $\mathbf{P} = (X, Y, Z)$ del espacio XYZ en los puntos $\mathbf{p} = (x, y)$ del plano imagen xy es una proyección central. Si el sistema coordenado del espacio XYZ está definido como centrado en la cámara, es decir, el punto focal corresponde al centro de proyección y el eje óptico al eje Z (en el caso del estéreo fotográfico se considera cualquiera de las cámaras como referencia), entonces las ecuaciones de proyección pueden derivarse del teorema geométrico de rayos y están dadas por

$$x = \frac{f \cdot X}{Z} \quad y = \frac{f \cdot Y}{Z} \quad (3.1)$$

Si el origen del sistema coordenado está posicionado en el plano imagen (sistema centrado en la imagen), entonces las ecuaciones de proyección son

$$x = \frac{f \cdot X}{f + Z} \quad y = \frac{f \cdot Y}{f + Z} \quad (3.2)$$

El espacio real \mathcal{R}^3 , es decir, el espacio XYZ se describe en coordenadas homogéneas (t, u, v, w) como el espacio proyectado \mathbf{P}^3 .

Si un punto del espacio XYZ debe expresarse en coordenadas proyectadas entonces puede elegirse cualquier valor $w \neq 0$. Un plano $Z = pX + qY + r$ por ejemplo, tiene la proyección

$$v = pt + qu + rw \quad (3.3)$$

en coordenadas homogéneas. Para $w = 0$ se obtiene un plano en el infinito. Las transformaciones de \mathbf{P}^3 son lineales y se conocen como *colineaciones* [Klette et al., 1996], estas se encuentran dadas por

$$(t', u', v', w')^T = \mathbf{A} \cdot (t, u, w, w)^T \quad (3.4)$$

donde \mathbf{A} es una matriz de 4×4 no singular de números reales [Haralick y Shapiro, 1993]. Los sistemas equivalentes y transformaciones correspondientes para modelos de proyección tanto central como paralela se desarrollan en el apéndice A.

El modelo de proyección paralela es particularmente útil cuando los objetos en la escena se encuentran cerca de la cámara pero son pequeños en comparación con la constante de cámara f , si los objetos se encuentran relativamente lejos de la cámara o se usan lentes con distancia focal grande. En cualquier otro caso se recomienda el uso del modelo de proyección central.

3.2 Relación entre sistemas coordenados de cámara y mundo

El siguiente sistema a analizar se denomina sistema coordenado de cámara y se denomina con el índice k . Además de este sistema $X_k Y_k Z_k$ se asume también un sistema de coordenadas reales o coordenadas de mundo $X_w Y_w Z_w$.

La constante de cámara (también llamada distancia focal efectiva) describe la distancia ortogonal entre el centro de la lente en la cámara y el plano imagen. Se denota la constante de cámara como f_k para distinguirla de la distancia focal de la lente f . En este caso ya no se considera un sistema con *pinhole*, sino que se agrega una lente ideal (sin aberraciones) con distancia focal f (figura 3.2).

Geoméricamente, tanto el modelo *pinhole* como el de una lente tienen casi las mismas características. En ambos casos puede considerarse proyección central como modelo para proyectar los puntos de la escena real P en el plano imagen de la cámara.

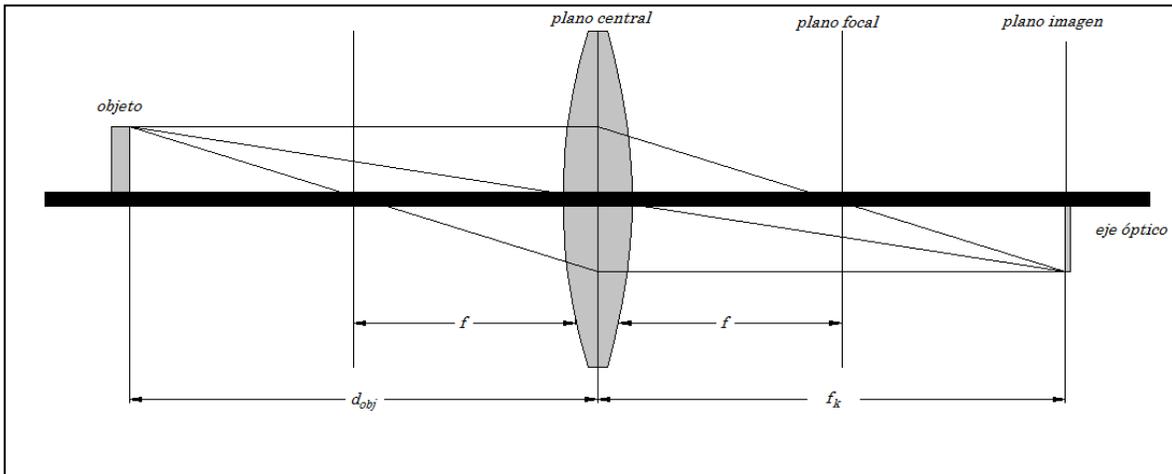


Figura 3.2 Punto objeto y plano imagen para una lente delgada

La diferencia principal entre los modelos es que una lente solo proyecta los puntos bien enfocados en el plano imagen. La distancia entre el objeto y el centro de la lente se conoce como *ancho del objeto*, mientras que la distancia entre el plano imagen y el centro de la cámara se conoce como *ancho de imagen* y corresponde al término en fotogrametría de la constante de cámara. La relación entre el ancho de imagen con respecto a la constante de cámara f_k , la distancia focal f de la lente y el ancho de objeto d_{obj} está dada por la fórmula de lente Gaussiana

$$\frac{1}{f} = \frac{1}{f_k} + \frac{1}{d_{obj}} \quad (3.5)$$

En la figura 3.3 se muestra la situación de adquisición, donde el plano imagen se asume detrás del plano $X_k Y_k$ a una distancia f_k . Las coordenadas de mundo y las de cámara pueden transformarse entre sí a través de una transformación Euclídea, es decir una operación de translación seguida de una rotación.

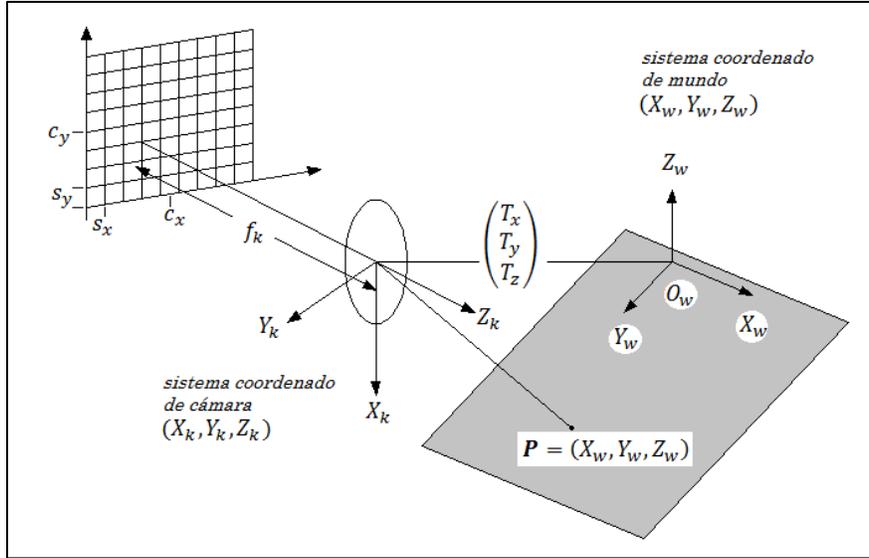


Figura 3.3 Coordenadas de mundo, de cámara y de imagen

La transformación de un punto (X_w, Y_w, Z_w) en coordenadas mundo a un punto en coordenadas centradas en la cámara (X_k, Y_k, Z_k) está descrita por la rotación \mathbf{R} y la translación \mathbf{T} ,

$$\begin{pmatrix} X_k \\ Y_k \\ Z_k \end{pmatrix} = \mathbf{R} \cdot \begin{pmatrix} X_w \\ Y_w \\ Z_w \end{pmatrix} + \mathbf{T}, \quad \text{con } \mathbf{T} = \begin{pmatrix} T_x \\ T_y \\ T_z \end{pmatrix} = \begin{pmatrix} -X_o \\ -Y_o \\ -Z_o \end{pmatrix} \quad (3.6)$$

La matriz de rotación \mathbf{R} representa las tres transformaciones parciales R_x , R_y y R_z con

$$\mathbf{R} = R_x \cdot R_y \cdot R_z = \begin{pmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{pmatrix} \quad (3.7)$$

La matriz \mathbf{R}_x describe la rotación alrededor del eje X_k dado por un ángulo α , conocido como ángulo de recorrido (*pan angle*) y está dada por

$$R_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\text{sen}(\alpha) \\ 0 & \text{sen}(\alpha) & \cos(\alpha) \end{pmatrix} \quad (3.8)$$

La matriz \mathbf{R}_y representa la rotación alrededor del eje Y_k dado por un ángulo β , se le denomina como ángulo de inclinación (*tilt angle*) y está dada por

$$R_y(\beta) = \begin{pmatrix} \cos(\beta) & 0 & \text{sen}(\beta) \\ 0 & 1 & 0 \\ -\text{sen}(\beta) & 0 & \cos(\beta) \end{pmatrix} \quad (3.9)$$

Por último, la matriz \mathbf{R}_z describe una rotación alrededor del eje Z_k por un ángulo γ y se le conoce como ángulo de rodamiento (*roll angle*). Esta rotación está definida por

$$R_z(\gamma) = \begin{pmatrix} \cos(\gamma) & -\text{sen}(\gamma) & 0 \\ \text{sen}(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (3.10)$$

En el proceso de calibración, los parámetros de la matriz \mathbf{R} y del vector \mathbf{T} deben resolverse para determinar la relación entre las coordenadas de mundo y las de cámara. Los parámetros de la transformación Euclídea definen los parámetros extrínsecos de la cámara. Los parámetros intrínsecos se refieren al comportamiento proyectivo de la cámara. Estos parámetros son el punto principal y la constante de cámara (la distancia focal efectiva). El punto principal se define como el punto de intersección entre el eje óptico y el plano imagen, este casi nunca es idéntico al centro de la imagen, que se encuentra en el origen del plano imagen.

3.3 Calibración por Transformaciones Lineales Directas

Una medición directa tanto de parámetros intrínsecos como extrínsecos puede obtenerse de manera indirecta utilizando una técnica de calibración adecuada. El objetivo es obtener la correspondencia entre el modelo de cámara utilizado (parametrizado por los valores observados o calculados) y la imagen adquirida por la misma.

La técnica de Transformaciones Lineales Directas (DLT, por sus siglas en inglés) se basa en el conocimiento de algunas coordenadas en el espacio de puntos geoméricamente configurados que se representan físicamente por marcas en algún objeto de calibración.

Bajo ciertas condiciones, el cálculo automatizado de las localizaciones de las marcas de calibración es una tarea de fácil solución en el análisis de imágenes, asumiendo que las condiciones de iluminación y la coloración de las caras del objeto de calibración y de las marcas permite la fácil detección de las mismas. Entre las marcas de calibración más comunes se encuentran círculos rellenos y localizaciones de precisión a nivel de sub-píxel (puntos medios de las marcas) basadas en el cálculo de centroides.

Para una determinación precisa de los parámetros de calibración resulta beneficioso que las marcas se encuentren igualmente espaciadas en la escena y que la imagen tomada digitalmente sea tan buena como sea posible. Por lo tanto, el tamaño del objeto de calibración debe mantener una relación con la distancia de trabajo de la cámara. Si por ejemplo, quieren obtenerse imágenes a una distancia de 2 – 3m, entonces el objeto de calibración debe elegirse de tal manera que este pueda proyectarse en toda la imagen digital colocada a la distancia mencionada.

El uso de la técnica de DLT es una posibilidad para especificar los parámetros en cuestión. Tiene la ventaja de que deben resolverse casi únicamente ecuaciones lineales [Abdel-Aziz y Karara, 1971].

Dada la naturaleza vectorial del desarrollo de la técnica DLT y el enfoque práctico de esta tesis, el desarrollo y cálculo de los parámetros se hacen en el apéndice A, por si interesara al lector.

$$\overline{AB} = \frac{\phi_B - \phi_O}{2\pi f} \quad (3.12)$$

en donde ϕ_B es la fase relacionada con el plano de referencia, ϕ_O la del objeto bajo estudio y f es la frecuencia de las franjas proyectadas.

3.4.2 Ejes ópticos coplanares y no paralelos

Se dice que los ejes ópticos de un sistema cámara-proyector son no paralelos cuando los elementos se encuentran a una distancia distinta del plano de referencia como se ilustra en la figura 3.5. El plano XOY' es el plano imagen en donde la rejilla es proyectada y el plano XOY es el plano objeto de la cámara. Si se analiza el punto B sobre el plano de referencia, se observa que al desplazarse el punto objeto en la dirección BC , la fase varía en la línea $B'E$, que es la intersección entre el plano FCB y el plano XOY' . La fase cambia entonces de B' a A' cuando el punto en el objeto cambia de B a D .

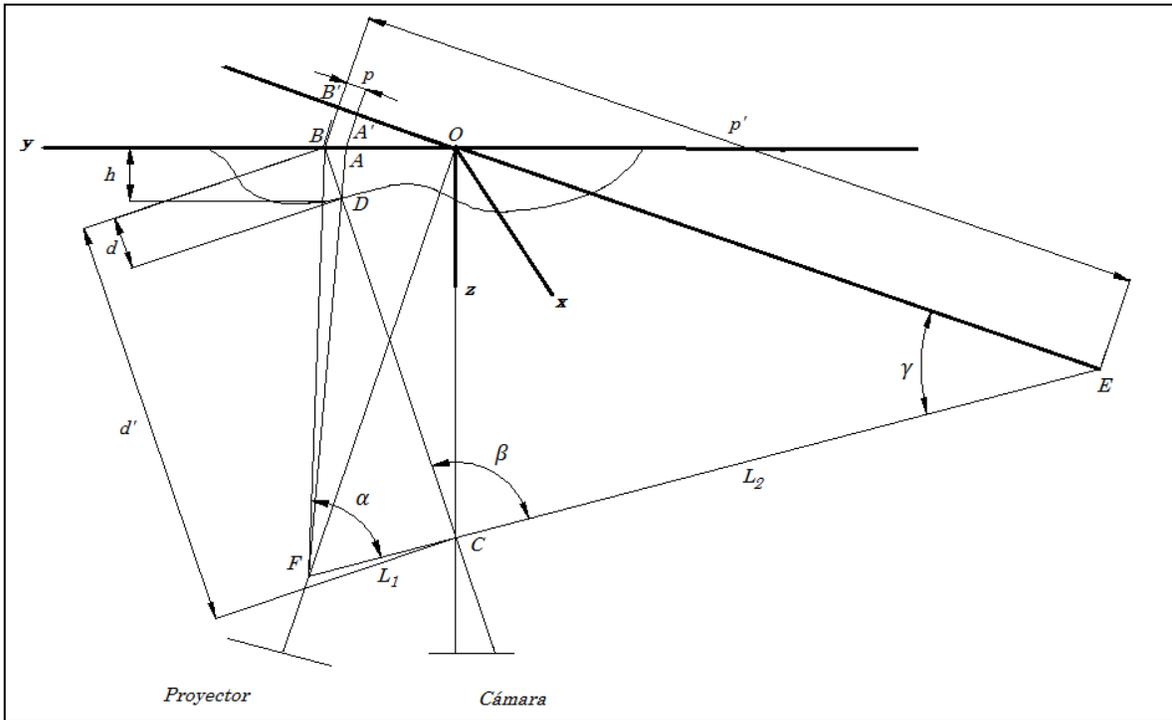


Figura 3.5 Sistema de proyección de franjas con ejes ópticos coplanares y no paralelos

Si se conocen los parámetros y se considera un sistema sin distorsión, entonces para el detector D_B , la altura del punto D es proporcional a la distancia d por una constante k , es decir $h = kd$. LA función de altura para cada punto puede expresarse por

$$h(x, y) = \left[a(x, y) + b(x, y) \frac{1}{p(x, y)} \right]^{-1} \quad (3.13)$$

en donde

$$a = \frac{k}{d'} \left[1 - \frac{p' L_1 \text{sen } \gamma}{d' (L_1 + L_2) \text{sen } \beta} \right] \quad (3.14)$$

$$b = \frac{kp^2 L_1 \text{sen } \gamma}{d'^2 (L_1 + L_2) \text{sen } \beta} \quad (3.15)$$

Para llevar a cabo este tipo de calibración aún hace falta saber los parámetros de la cámara para poder caracterizar los fotodetectores y poder recuperar la forma a partir de las franjas proyectadas. Se recomienda el análisis de varios objetos de calibración como se menciona en la sección 3.3.

3.4.3 Ejes ópticos no coplanares y no paralelos

En la disposición geométrica de este tipo de arreglo, las líneas FO' y CO' (figura 3.6), correspondientes a los ejes ópticos se encuentran en distintos planos en el espacio. El plano $X'O'Y'$ es el plano imagen de la rejilla proyectada y el plano XOY es el plano imagen para la cámara.

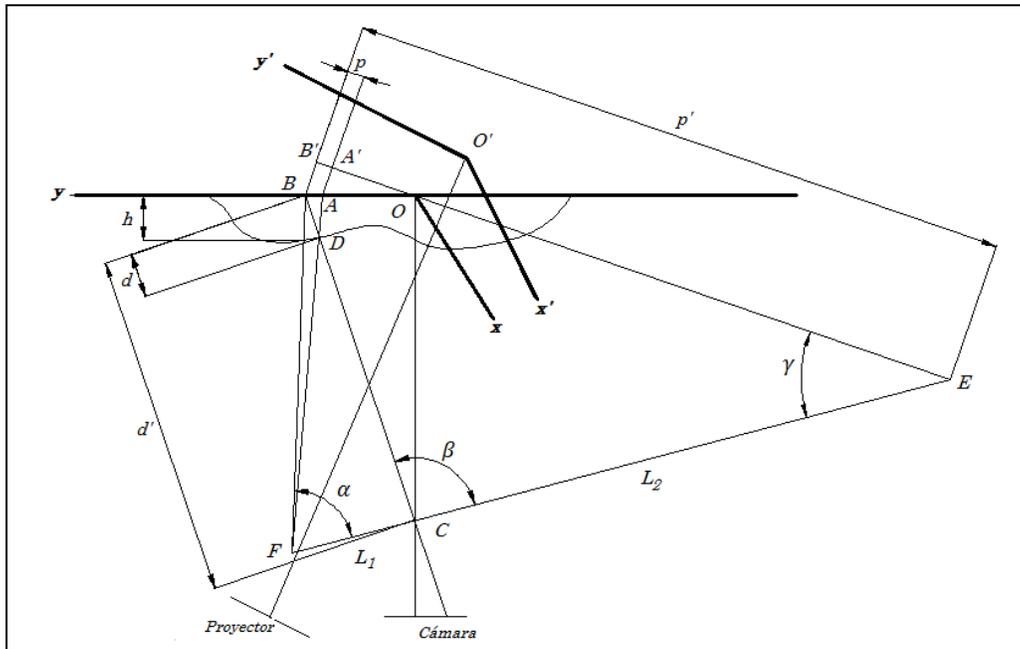


Figura 3.6 Sistema de proyección de franjas con ejes ópticos no coplanares y no paralelos

Al igual que en el sistema con ejes ópticos coplanares y no paralelos, cuando un punto del objeto se mueve en la dirección BC, la fase varía a lo largo de la línea B'E, que es la línea de intersección entre los planos FCD y $XO'Y'$.

La vista frontal de este plano se muestra en la figura 3.7, la notación de algunos puntos se ha cambiado por conveniencia.

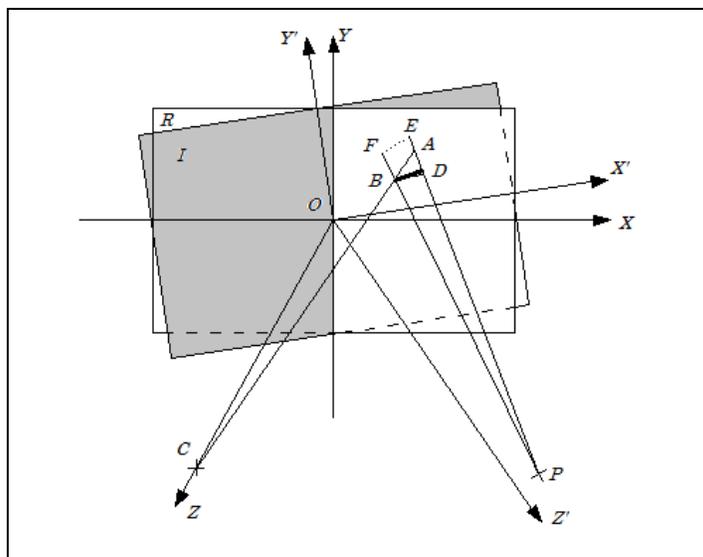


Figura 3.7 Vista frontal del sistema de proyección de franjas con ejes ópticos no coplanares y no paralelos

En la figura 3.7, los centros de proyección de las lentes de la cámara y del proyector se encuentran localizados en C y P , respectivamente; R denota el plano de referencia. Se establece un sistema coordenado $(OXYZ)$, en el que el plano XOY está superpuesto en R , el eje Z pasa por el punto C y las coordenadas de C se encuentran en $(0,0,z_C)$. El plano I es un plano ficticio que pasa por el origen O y es perpendicular al eje óptico de la lente del proyector. Entonces, las franjas proyectadas en I son paralelas y se encuentran separadas por un periodo p . Se establece también el sistema coordenado $(OX'Y'Z')$, en el que el plano $X'OY'$ se encuentra superpuesto en I y el eje Y' es paralelo a las franjas en I , entonces las coordenadas de P son (x'_p, y'_p, z'_p) .

Para establecer la relación entre los dos sistemas coordenados, se asume que el sistema $(OXYZ)$ puede transformarse a $(OX'Y'Z')$ a través de rotaciones alrededor de los ejes X , Y y Z en secuencia, a través de los ángulos α , β y γ , respectivamente. Esto significa que un punto $(OXYZ)$ tiene coordenadas (x, y, z) , sus coordenadas en $(OX'Y'Z')$ puede calcularse con

$$(x', y', z') = (x, y, z)R_x(\alpha)R_y(\beta)R_z(\gamma) \quad (3.16)$$

en donde $R_x(\alpha)$, $R_y(\beta)$ y $R_z(\gamma)$ son las mismas mostradas en la sección 3.2.

En el sistema coordenado $(OXYZ)$, el plano I está determinado por la orientación del proyector. Dado que existen tres grados de libertad para determinar la orientación de un objeto en el espacio tridimensional, los ángulos de rotación α , β y γ son parámetros geométricos básicos que describen la orientación del proyector.

Cuando un patrón de franjas es proyectado sobre el plano de referencia, la fase en el punto B se denota como ϕ_B . Cuando el mismo patrón se proyecta sobre el objeto bajo prueba, la fase en ese punto es ϕ_A . La diferencia de fase es entonces

$$\psi = \phi_A - \phi_B \quad (3.17)$$

La línea \overline{PA} cruza I en el punto E , y \overline{PB} en F . Nótese que las líneas \overline{PE} y \overline{PF} son ambas líneas con la misma fase, se tiene entonces que $\phi_A = \phi_E$ y $\phi_B = \phi_F$, por lo que se obtiene

$$\psi = \phi_E - \phi_F \quad (3.18)$$

Si A es un punto en el objeto con una altura h , entonces las coordenadas del punto A en el sistema coordenado $(OX'Y'Z')$ pueden calcularse como (x'_A, y'_A, z'_A) , en donde

$$\begin{aligned} x'_A &= [x_B \cos \beta \cos \gamma + y_B (\sen \alpha \sen \beta \cos \gamma + \cos \alpha \sen \gamma)] \\ &\quad \times \frac{h + z_C}{z_C} - h(-\cos \alpha \sen \beta \cos \gamma + \sen \alpha \sen \gamma) \\ y'_A &= [-x_B \cos \beta \sen \gamma + y_B (-\sen \alpha \sen \beta \sen \gamma + \cos \alpha \cos \gamma)](h + z_C)/z_C \\ &\quad - h(\cos \alpha \sen \beta \sen \gamma + \sen \alpha \cos \gamma) \\ z'_A &= [x_B \sen \beta - y_B \sen \alpha \cos \beta](h + z_C)/z_C - h \cos \alpha \cos \beta \end{aligned} \quad (3.19)$$

Las coordenadas del punto B son (x'_B, y'_B, z'_B) , donde

$$\begin{aligned} x'_B &= x_B \cos \beta \cos \gamma + y_B (\sen \alpha \sen \beta \cos \gamma + \cos \alpha \sen \gamma) \\ y'_B &= -x_B \cos \beta \sen \gamma + y_B (-\sen \alpha \sen \beta \sen \gamma + \cos \alpha \cos \gamma) \\ z'_B &= x_B \sen \beta - y_B \sen \alpha \cos \beta \end{aligned} \quad (3.20)$$

Las coordenadas para los puntos E y F serán

$$\begin{aligned} x'_E &= x'_P - z'_P (x'_A - x'_P)/(z'_A - z'_P) \\ y'_E &= y'_P - z'_P (y'_A - y'_P)/(z'_A - z'_P) \\ z'_E &= 0 \end{aligned} \quad (3.21)$$

y

$$\begin{aligned} x'_F &= x'_P - z'_P (x'_B - x'_P)/(z'_B - z'_P) \\ y'_F &= y'_P - z'_P (y'_B - y'_P)/(z'_B - z'_P) \\ z'_F &= 0 \end{aligned} \quad (3.22)$$

Utilizando las ecuaciones 3.17 - 3.21 se obtiene que diferencia de fase estará dada por

$$\psi = \frac{2\pi(x'_E - x'_F)}{p} = \frac{2\pi z'_P}{p} \left(\frac{x'_B - x'_P}{z'_B - z'_P} - \frac{y'_A - y'_P}{z'_A - z'_P} \right) \quad (3.23)$$

A partir de este punto son varios los métodos de calibración existentes, algunos de estos incluyen calibraciones por separado de cámara y proyector [Zhang y Huang, 2006], la primera se obtiene al aplicar metodologías como las mencionadas anteriormente en esta sección; la

calibración del proyector se logra al obtener datos de referencia de a partir de la proyección de patrones conocidos sobre un objeto de calibración en distintas posiciones [Ito e Ishi, 1995; Tsai y Hung, 2005]; otros enfocan el problema desde la adquisición de planos de calibración colocados a diferentes distancias del sistema de adquisición y la aplicación de algún tipo de minimización de error [Hongwei *et al.*, 2005; Chen *et al.*, 2008, Cuevas *et al.*, 1999].

3.5 Cámara con distorsión geométrica

En las secciones anteriores se modeló a la cámara basándose en un sistema ideal aunque las cámaras comunes se caracterizan por, de alguna manera, un comportamiento de proyección distorsionado. Los efectos de distorsión pueden describirse matemáticamente y pueden considerarse en el modelado.

3.5.1 Distorsión radial

En la figura 3.8 se muestra la geometría de entre la cámara y el sistema de coordenadas mundo. El centro de proyección O_k es el origen del sistema centrado en la cámara. El eje Z_k en el mismo sistema coordenado coincide con el eje óptico. La proyección de un punto (X_w, Y_w, Z_w) dado en coordenadas de mundo a coordenadas de un dispositivo digital (x_b, y_b) se lleva a cabo en cinco pasos de transformación. En estos pasos se utiliza la siguiente notación:

- (X_w, Y_w, Z_w) denota las coordenadas tridimensionales de un punto P en el sistema de coordenadas de mundo,
- (X_k, Y_k, Z_k) denota las coordenadas tridimensionales de un punto P en el sistema coordenado centrado en la cámara,
- f_k constante de la cámara (distancia focal efectiva),
- (x_u, y_u) son las coordenadas de la imagen sin distorsión (X_k, Y_k, Z_k) asumiendo un sistema de proyección central (modelo de cámara ideal),
- (x_v, y_v) coordenadas válidas con distorsión bajo consideración de una lente con distorsión radial,
- (x_s, y_s) son las coordenadas del sensor bajo consideración del cambio del punto principal, y
- (x_b, y_b) son las coordenadas (finales) en la imagen digital adquirida por el dispositivo.

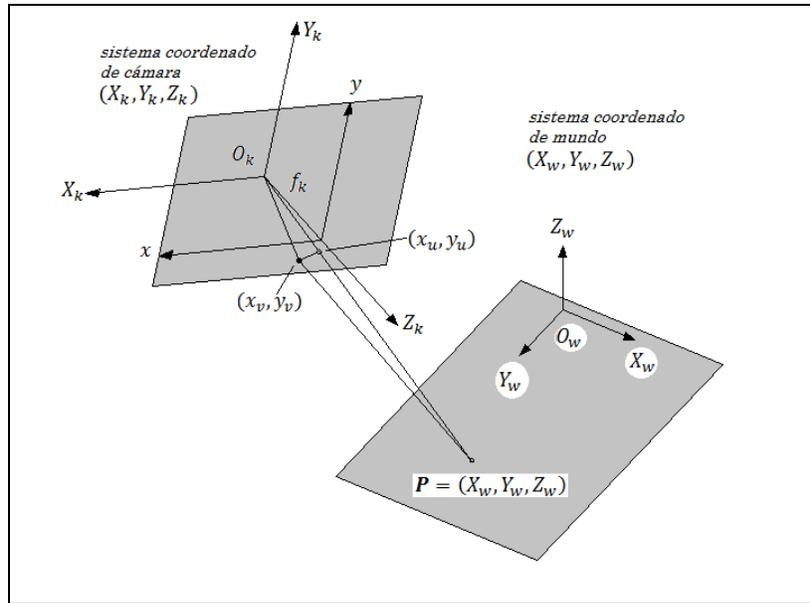


Figura 3.8 Geometría de cámara con proyección en perspectiva y distorsión de lente

Las cinco transformaciones antes mencionadas se describen a continuación:

(i) *Transformación de coordenadas mundo a coordenadas centradas en la cámara.*

Se refiere al mismo proceso descrito en la sección 3.2, esta transformación está dada según la ecuación 3.6 por

$$\begin{pmatrix} X_k \\ Y_k \\ Z_k \end{pmatrix} = \mathbf{R} \cdot \begin{pmatrix} X_w \\ Y_w \\ Z_w \end{pmatrix} + \mathbf{T}, \quad \text{con } \mathbf{T} = \begin{pmatrix} T_x \\ T_y \\ T_z \end{pmatrix} = \begin{pmatrix} -X_o \\ -Y_o \\ -Z_o \end{pmatrix}$$

Una vez más \mathbf{T} denota del vector de translación y \mathbf{R} es la matriz de rotación proveniente de las tres transformaciones parciales \mathbf{R}_x , \mathbf{R}_y y \mathbf{R}_z . Los parámetros para la matriz ortogonal \mathbf{R} y el vector \mathbf{T} deben determinarse durante un proceso de calibración.

(ii) *Proyección sobre el plano imagen*

Asúmase una vez más un sistema de proyección central, la transformación del sistema coordenado de cámara (X_k, Y_k, Z_k) al sistema de imagen sin distorsión (x_u, y_u) está descrito por las ecuaciones de proyección

$$x_u = \frac{f_k \cdot X_k}{Z_k} \quad \text{y} \quad y_u = \frac{f_k \cdot Y_k}{Z_k} \quad (3.24)$$

provenientes de los teoremas de rayos discutidos en la sección 3.1. La constante de cámara f_k debe conocerse para modelar estas transformaciones.

(iii) *Transformación de coordenadas de imagen sin distorsión a coordenadas distorsionadas*

En la figura 3.9 se muestra la influencia de la distorsión en una imagen. Un modelo ideal de la distorsión en la lente lleva a un número infinito de coeficientes de distorsión. En la práctica se tiene ya una buena aproximación para estas distorsiones cuando sólo se consideran dos coeficientes κ_1 y κ_2 . Un valor positivo de estos coeficientes significa que las esquinas de la imagen se proyectan hacia el centro de la misma, el caso contrario (un valor negativo) significa la transformación inversa.

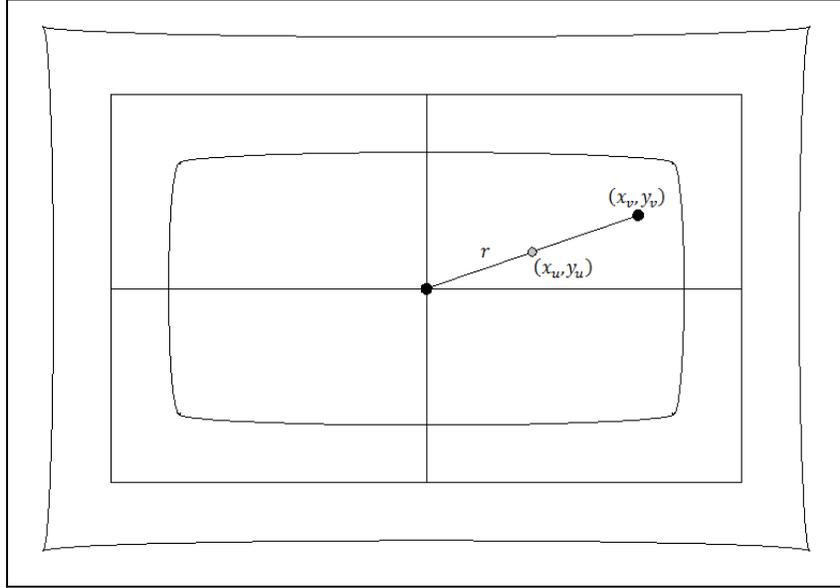


Figura 3.9 Deformación de una imagen ideal por el efecto de distorsión radial de lente

Las coordenadas de la imagen válida (x_v, y_v) pueden determinarse a partir de las coordenadas de la imagen distorsionada (x_u, y_u) usando las ecuaciones

$$x_v = x_u - D_x \quad y \quad y_v = y_u - D_y \quad (3.25)$$

en donde

$$D_x = x_v \cdot (\kappa_1 r^2 + \kappa_2 r^4), \quad D_y = y_v \cdot (\kappa_1 r^2 + \kappa_2 r^4) \quad y \quad r = \sqrt{x_v^2 + y_v^2} \quad (3.26)$$

Los coeficientes de distorsión κ_1 y κ_2 deben de calibrarse. Comenzando con las coordenadas válidas de imagen (x_v, y_v) , puede calcularse una corrección de la localización de los puntos de imagen utilizando las ecuaciones antes mencionadas.

Aún así, para el cálculo inverso (de coordenadas ideales a válidas) la transformación algebraica de estas ecuaciones para los parámetros κ_1 y κ_2 lleva a un sistema de ecuaciones no lineales. Basándose en dichas ecuaciones:

$$x_{vi} = \frac{x_u}{1 + \kappa_1 r_{i-1}^2 + \kappa_2 r_{i-1}^4}, \quad y_{vi} = \frac{y_u}{1 + \kappa_1 r_{i-1}^2 + \kappa_2 r_{i-1}^4} \quad (3.27)$$

con $r_i = \sqrt{x_{vi}^2 + y_{vi}^2}$, para $i \in \{1, \dots, n\}$, la distorsión puede determinarse de manera iterativa al inicializar $r_0 = \sqrt{x_u^2 + y_u^2}$. La demanda de precisión y velocidad de cómputo determinan el máximo número de iteraciones. En general, se adquiere una buena aproximación dentro de no más de ocho iteraciones [Horn, 1986].

(iv) *Transformación de coordenadas de imagen distorsionada a coordenadas de imagen digital bajo consideración del desplazamiento del punto principal*

Como ya se mencionó antes, el punto principal no necesariamente coincide con el centro de la imagen. Si estos puntos son diferentes, entonces el punto principal exhibe un movimiento circular en el plano imagen mientras el foco del lente de la cámara se altere. Por lo tanto la posición del punto principal (c_x, c_y) en el plano imagen debe de calibrarse con respecto al ajuste de la cámara. Finalmente, estas coordenadas deben de transformarse en coordenadas imagen (x_b, y_b) .

Las coordenadas del punto principal centrado en el sensor se escalan y se convierten en posiciones de fila y de columna de la imagen almacenada. Estas coordenadas pueden determinarse directamente de las coordenadas de imagen distorsionada por

$$x_b = \frac{s_x x_v}{d'_x} + c_x \quad y \quad y_b = \frac{y_v}{d_y} + c_y \quad (3.28)$$

En donde (c_x, c_y) denota las coordenadas del punto principal, Nc_x es el número de elementos del sensor en una fila de la imagen, Nf_x es el número de pixeles en una fila de la imagen digital, $d'_x = d_x \frac{Nc_x}{Nf_x}$ es la distancia horizontal entre dos pixeles en la imagen, d_x es la distancia horizontal entre dos elementos del sensor, d_y es la distancia vertical entre dos elementos del sensor y s_x representa el factor de escalamiento horizontal.

El factor de escalamiento s_x debe de calibrarse, mientras que los parámetros d_x , d_y , Nc_x y Nf_x son casi siempre incluidos en la hoja de datos proporcionada por el fabricante de la cámara y la unidad de adquisición de imágenes.

3.5.2 Distorsión de de-centro

Este tipo de distorsión se presenta principalmente en cámaras con sistemas de más de una lente y sucede cuando los ejes ópticos de estas no se encuentran alineados. Este tipo de distorsión tiene componentes radial y tangencial y tiene una forma como la mostrada en la figura 3.10, mientras que la distorsión tangencial pura se muestra en la figura 3.11.

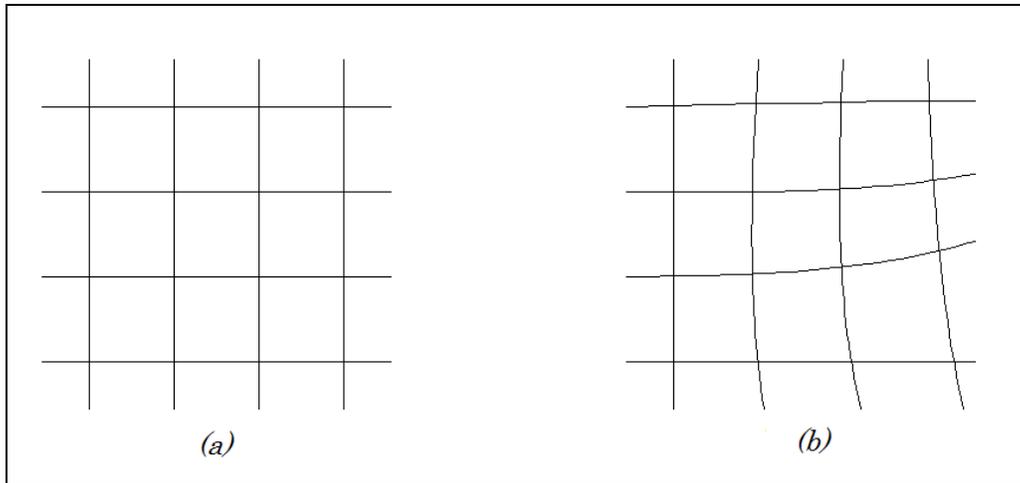


Figura 3.10 Distorsión de de-centro (a) Cuadrícula sin distorsión, (b) cuadrícula con distorsión

Considerando la misma notación que en la sección anterior, las distorsiones radial y tangencial son descritas por las siguientes expresiones:

$$\begin{aligned}\delta_{rd} &= (j_1 r^2 + j_2 r^4 + \dots) \text{sen}(\phi - \phi_o) \\ \delta_{td} &= (j_1 r^2 + j_2 r^4 + \dots) \text{cos}(\phi - \phi_o)\end{aligned}\quad (3.29)$$

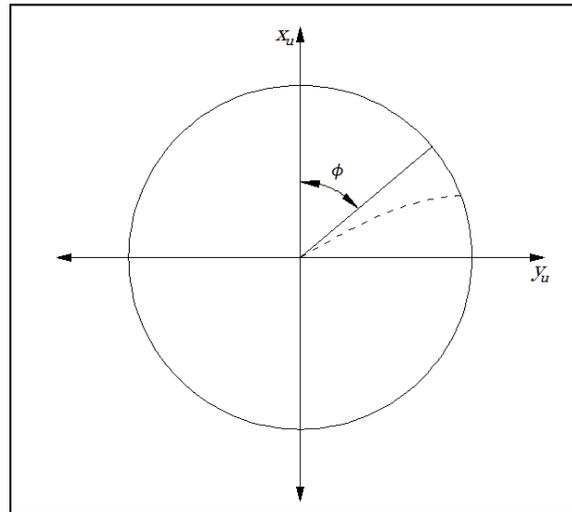


Figura 3.11 Distorsión tangencial (denotada por la línea punteada).

donde ϕ_o es el ángulo entre el eje positivo x_u y una línea de referencia conocida como eje de máxima distorsión tangencial.

La distorsión a lo largo de los ejes x_u y y_u pueden determinarse mediante la siguiente relación

$$\begin{pmatrix} \delta_{xd} \\ \delta_{yd} \end{pmatrix} = \begin{pmatrix} \text{cos}\phi - \text{sen}\phi \\ \text{sen}\phi \text{cos}\phi \end{pmatrix} \begin{pmatrix} \delta_{rd} \\ \delta_{td} \end{pmatrix}\quad (3.30)$$

3.5.3 Distorsión de prisma delgado y distorsión total

La distorsión de prisma delgado es ocasionada por la imperfección en el diseño de lentes y en su manufactura o por inclinaciones entre los elementos ópticos de la cámara. La expresión matemática relacionada con este tipo de distorsión es

$$\begin{aligned}\delta_{rd} &= (i_1 r^2 + i_2 r^4 + \dots) \text{sen}(\phi - \phi_1) \\ \delta_{td} &= (i_1 r^2 + i_2 r^4 + \dots) \text{cos}(\phi - \phi_1)\end{aligned}\quad (3.31)$$

donde ϕ_1 es el ángulo entre el eje positivo x_u y el eje de la máxima distorsión tangencial. Estas relaciones pueden expresarse en coordenadas rectangulares como

$$\begin{aligned}\delta_{xp} &= s_1(x_u^2 + y_u^2) + O[(x_u, y_u)^2] \\ \delta_{yp} &= s_2(x_u^2 + y_u^2) + O[(x_u, y_u)^2]\end{aligned}\quad (3.32)$$

donde

$$s_1 = -i_i \text{sen} \phi_1 \quad y \quad s_2 = i_i \text{cos} \phi_1 \quad (3.33)$$

Nótese que las ecuaciones que modelan las distorsiones de de-centro y de prisma delgado son similares aunque, debe remarcar que el eje de máxima distorsión no es necesariamente el mismo para ambos casos.

La distorsión es la suma de todas las distorsiones antes mencionadas (radial, de de-centro y de prisma delgado). Dado que las distorsiones son expansiones binomiales de las distancias de distorsión en cada punto, entonces para cada una de las distorsiones mencionadas basta con los dos primeros términos para dar una buena exactitud de las mismas [Goodman, 1996; Legarda-Sáenz *et al.*, 2003], ya que los términos con exponente mayor a 3 son muy cercanos a 0. La distorsión total está dada entonces por

$$\begin{aligned}\delta_x(x_u, y_u) &= x_u(\kappa_1 r^2 + \kappa_2 r^2) + [2j_1 x_u y_u + j_2(r^2 + 2x_c^2)] + s_1(x_u^2 + y_u^2) \\ \delta_y(x_u, y_u) &= y_u(\kappa_1 r^2 + \kappa_2 r^2) + [2j_2 x_u y_u + j_1(r^2 + 2x_c^2)] + s_2(x_u^2 + y_u^2)\end{aligned}\quad (3.34)$$

con $r^2 = x_u^2 + y_u^2$.

En este capítulo se revisaron las transformaciones de un sistema coordinado del mundo real a un sistema interpretado por la máquina a partir de distintas transformaciones Euclideas y la parametrización del equipo de adquisición por medio de técnicas de calibración. Se revisaron además las aberraciones inherentes a una cámara con sistema de lentes. Es importante resaltar la importancia de una correcta calibración de un sistema de adquisición con fines metrológicos, pues un mal método puede conducir a importantes errores en el cálculo de las cantidades físicas a medir.

El capítulo 4 hace referencia a métodos computacionales para optimización de problemas, se comienza con las técnicas de búsqueda exhaustiva y analíticas. Se hace también una breve revisión de algunas técnicas de optimización y aprendizaje de máquina basados en modelos naturales. Se hace especial énfasis en la técnica de algoritmos evolutivos, dada su importancia para este trabajo de tesis.

Referencias

- [1] Abdel-Aziz, Y.I., Kahara, H.M., '*Direct linear transformation into object space coordinates in close-range photogrammetry*', Proc. ASP Symposium on Close-Range Photogrammetry, Urbana, Illinois, USA, 1971, pp. 1-18.
- [2] Chen, X., Xi, J., Jin, Y., Sun, J., '*Accurate calibration for a camera-projector measurement system based on structured light projection*', Optics and Lasers in Engineering **47**: 310-319 (2009).
- [3] Cuevas, F.J., Servin, M., Rodriguez-Vera, R., '*Depth object recovery using radial basis functions*', Optics Communications 163: 270 – 277 (1999).
- [4] Goodman, J.W., '*Introduction to Fourier optics*', McGraw-Hill, 2nd Ed., New York, 1996.
- [5] Haralick, R.M., Shapiro, L.G., '*Computer and Robot Vision, Volume II*', Addison-Wesley, 1993.
- [6] Hongwei, G., Haitao, H., Yingjie, Y., Mingyi, C., '*Least squares calibration method for fringe projection profilometry*', Optical Engineering, **44**: 1 – 9 (2005).
- [7] Horn, B.K.P., '*Robot Vision*', MIT Press, 1986.
- [8] Ito, M., Ishi, A., '*Three-level checkerboard pattern (TCP) projection method for curved surface measurement*', Pattern Recognition **28**: 27 – 40 (1995).
- [9] Klette, R., Schlüns, K., Koschan, A., '*Computer Vision: Three-Dimensional Data from Images*', Springer, 1996.
- [10] Legarda-Sáenz, R., Bothe, T., Jüptner, W.P., '*Accurate procedure for the calibration of a structured light system*', Opt. Eng. **43**: 464 – 471 (2003).
- [11] Ling, P.D., Sung, C.K., '*Comparing two new camera calibration methods with traditional pinhole calibration*', Opt. Express **6**: 3012 – 3022 (2007).
- [12] Tsai, M.J., Hung, C.C., '*Development of a high-precision surface metrology system using structured light projection*', Measurement 38: 236 – 247 (2005).
- [13] Zhang, S., Huang, P.S., '*Novel method for structured light system calibration*', Optical Engineering **48**: 3134 - 3142 (2006).

Capítulo 4

Técnicas de computación suave para búsqueda, optimización y aprendizaje de máquina

La optimización se refiere al proceso de determinar las mejores soluciones para ciertos problemas matemáticos, lo cuales son a menudo relacionados con la realidad física [Fletcher, 1987; Chong y Zak, 2001]. Al decir que se buscan las mejores soluciones se implica que existe más de una posible solución y que estas no tienen el mismo valor. La definición mejor es relativa al problema bajo análisis, su método de solución y las tolerancias permitidas.

En palabras más técnicas, el término optimización se refiere al proceso de ajustar las entradas o características de un aparato, proceso matemático o experimento para encontrar la salida o resultado máximo o mínimo [Haupt y Haupt, 2004].

De acuerdo con Haupt los algoritmos de optimización pueden dividirse en seis categorías que no son necesariamente mutuamente exclusivas.

1. La optimización por prueba y error se refiere al proceso de ajustar las variables que afectan la salida sin tener un conocimiento profundo que produce dicha salida.
2. El proceso de optimización se vuelve más difícil entre mayor sea el número de dimensiones a resolver. Muchos métodos de optimización multidimensional se generalizan a una serie de resoluciones unidimensionales.
3. El término optimización dinámica se refiere a que la función de salida es una función del tiempo, mientras que una estática es independiente de este factor. El problema estático es ya de por sí difícil de resolver, pero agregar la dimensión del tiempo incrementa el reto de resolver un problema dinámico.
4. La optimización puede distinguirse ya sea por variables discretas o continuas. Las variables discretas tienen un número finito de valores posibles, mientras que las variables continuas tienen un número infinito.
5. Las variables a menudo tienen límites o restricciones. Una variable restringida a menudo se convierte en una sin restricción a través de una transformación de variables. Muchas rutinas de optimización numérica funcionan mejor con variables no restringidas.
6. Algunos algoritmos buscan minimizar el costo comenzando por un grupo de valores de las variables. Estos buscadores de mínimos a menudo tienden a un óptimo local pero suelen ser rápidos en cuanto a su convergencia.

En las siguientes secciones se hará una revisión de algunos métodos computacionales de optimización en donde se incluyen los métodos exhaustivos y analíticos para continuar con los métodos basados en fenómenos biológicos, de donde se deriva la técnica de algoritmos genéticos, misma que será descrita más a fondo. Se presentan técnicas de optimización no basadas en computación evolutiva para ofrecer un panorama de porqué se seleccionaron técnicas de computación evolutiva.

4.1 Métodos de búsqueda exhaustivos y analíticos

El caso más básico en la optimización es la función de una variable. El método exhaustivo consiste en evaluar la función en un intervalo dado con tamaños de paso definidos. Sin embargo, este sistema es altamente ineficiente y tiene un costo computacional extremadamente alto si no se tiene conocimiento previo de la función a evaluar [Conte, 1980].

El caso mencionado anteriormente se puede extender a funciones multidimensionales, pero el número de operaciones a realizar se encuentra dado por

$$V = \prod_{i=1}^{N_{var}} Q_i$$

En donde V es el número de evaluaciones obtenidas de cada posible valor Q del intervalo seleccionado. Si se considera una función de dos variables x e y donde $0 \leq x, y \leq 10$ y se evalúa a intervalos de 0.1, se requeriría un total de 101^2 evaluaciones.

En las siguientes secciones se presentan alternativas a este tipo de análisis que proveen una mayor eficiencia en la búsqueda de óptimos para funciones de una sola variable.

4.1.1 Búsqueda del segmento dorado

Probablemente el método más sencillo de implementar para problemas en que busca optimizarse una sola dimensión es el de búsqueda de la sección dorada. La única propiedad que se asume de la función objetivo f es que exista un solo mínimo local (función unimodal) en el intervalo de búsqueda $[a_0, b_0]$, como la mostrada en la figura 4.1.

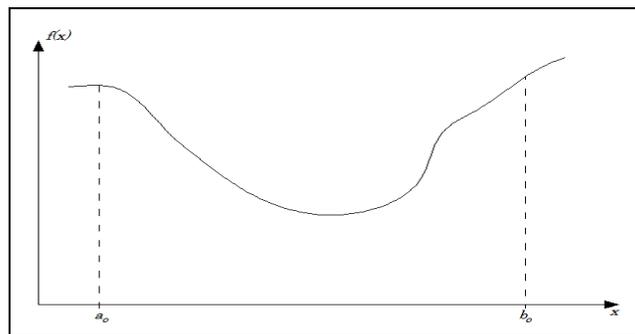


Figura 4.1 Función unimodal

El método discutido consiste en evaluar una función objetivo en diferentes puntos dentro del intervalo determinado por $[a_0, b_0]$, dichos puntos se seleccionan de tal forma que se logre encontrar el mínimo de f en tan pocas evaluaciones como sea posible [Fletcher, 1987]. El objetivo es reducir el rango hasta que el mínimo se encuentre 'encajonado' con suficiente precisión.

Considérese una función unimodal f de una variable en el intervalo $[a_0, b_0]$. Si se evalúa f en un punto intermedio del intervalo, se puede reducir el rango en el que el mínimo está localizado. Se debe de evaluar f en dos puntos intermedios. La selección de dichos puntos debe de hacerse de tal manera que la reducción del rango sea simétrica, en el sentido en que

$$a_1 - a_0 = b_0 - b_1 = \rho(b_0 - a_0) \quad (4.1)$$

donde $\rho < 1/2$.

Ahora se evalúa f en los puntos intermedios. Si $f(a_1) < f(b_1)$, entonces el mínimo debe de encontrarse en el rango $[a_0, b_1]$. Por el otro lado, si $f(a_1) \geq f(b_1)$ significa que el mínimo se encuentra contenido entre $[a_1, b_0]$ (figura 4.2).

Comenzando con el rango reducido de incertidumbre se puede repetir el proceso y encontrar dos nuevos puntos a_2 y b_2 usando el mismo valor de $\rho = 1/2$. Aún así, se busca minimizar el número de evaluaciones de la función objetivo al mismo tiempo que se reduce el ancho del nivel de incertidumbre.

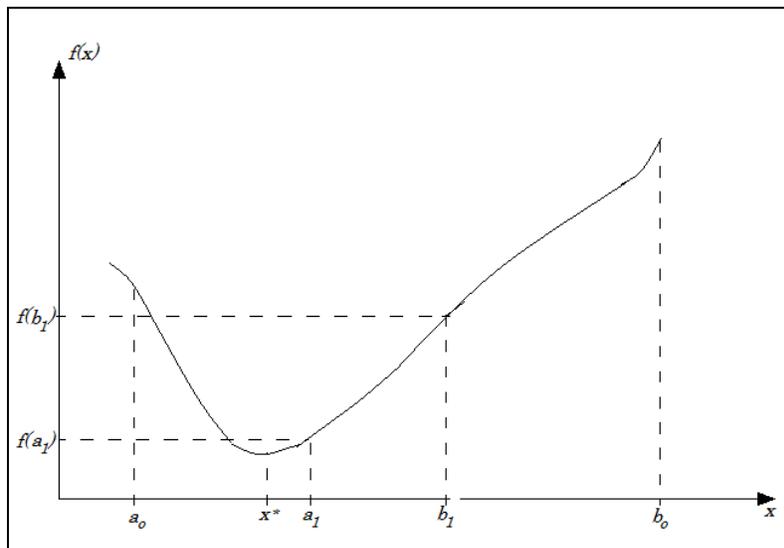


Figura 4.2 Caso en que $f(a_1) < f(b_1)$; x^* representa el mínimo

Para el caso mostrado en la figura 4.2 se sabe que x^* se encuentra en el intervalo $[a_0, b_1]$. Dado que a_1 se sabe en el intervalo de incertidumbre y $f(a_1)$ es conocido, puede hacerse que a_1 coincida con b_2 . Entonces, sólo se necesitará una nueva evaluación de f en a_2 .

El rango a analizar en N iteraciones ve reducido por una razón de

$$(1 - \rho)^N \approx (0.61803)^N \quad (4.2)$$

Esta regla está basada en la regla del la sección dorada planteado por los antiguos griegos en donde se establece que si se divide el rango de la razón de ρ a $\rho - 1$ tiene el efecto de que la razón del segmento más corto al más largo es igual a la razón del más largo a la suma de ambos. De aquí se deriva también que el método se denomine de la búsqueda de la sección dorada.

4.1.2 Búsqueda de Fibonacci

Este método es una variante de la búsqueda del segmento dorado, la diferencia reside en que ahora el factor ρ varía con cada generación, de manera que en la k -ésima etapa del proceso de reducción se utiliza un valor ρ_k .

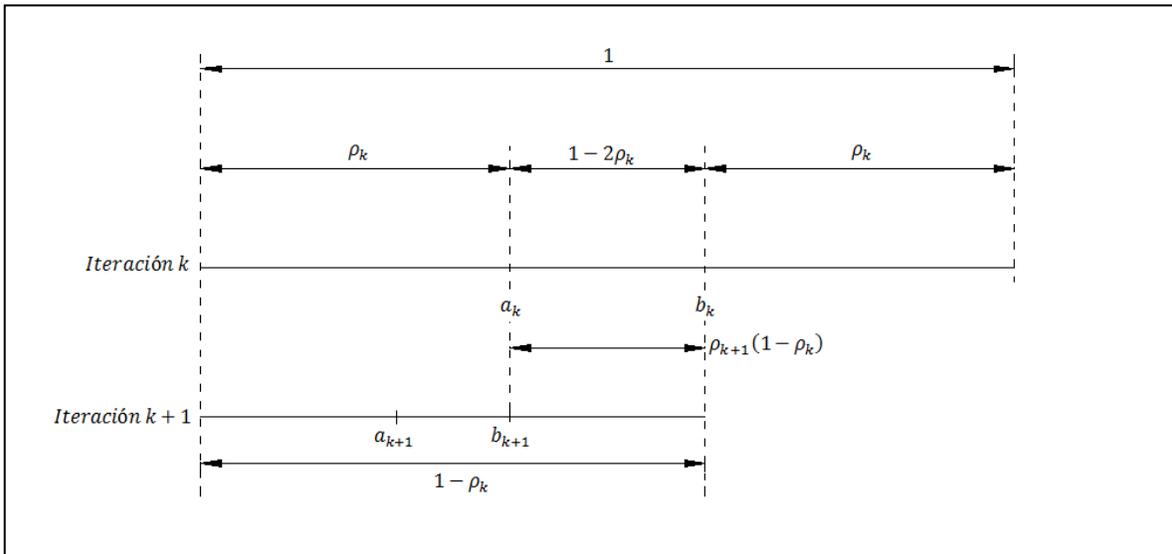


Figura 4.3 Selección de puntos de evaluación

Al igual que en la búsqueda del segmento dorado, el objetivo es escoger valores sucesivos de ρ_k en el intervalo $[0, 1/2]$, de tal manera que solo una nueva función de evaluación se requiera en cada etapa. Para evaluar la estrategia de búsqueda para seleccionar los puntos de evaluación considérese la figura 4.3, de donde puede verse que es suficiente elegir ρ_k de tal manera que

$$\rho_{k+1}(1 - \rho_k) = 1 - 2\rho_k \quad (4.3)$$

de donde se obtiene

$$\rho_{k+1} = 1 - \frac{\rho_k}{1 - \rho_k} \quad (4.4)$$

Para una secuencia ρ_1, ρ_2, \dots , después de N iteraciones del algoritmo, el rango de incertidumbre se reduce por un factor de

$$(1 - \rho_1)(1 - \rho_2) \dots (1 - \rho_N) \quad (4.5)$$

El objetivo del algoritmo es minimizar la expresión 4.5 sujeta a la condición establecida en 4.4 con $k = 1, \dots, N - 1$ dentro del intervalo $0 \leq \rho_k \leq 1/2$ y $k = 1, \dots, N$. En este punto es necesario presentar la secuencia de Fibonacci, F_1, F_2, F_3, \dots . Sea $F_{-1} = 0$ y $F_0 = 1$. Entonces para $k \geq 0$

$$F_{k+1} = F_k + F_{k-1}$$

La solución para el problema de optimización propuesto es

$$\rho_1 = 1 - \frac{F_N}{F_{N+1}}, \quad \rho_2 = 1 - \frac{F_{N-1}}{F_N}, \quad \dots, \quad \rho_N = 1 - \frac{F_1}{F_2} \quad (4.6)$$

En el método de búsqueda de Fibonacci, el rango de incertidumbre se reduce por el factor

$$(1 - \rho_1)(1 - \rho_2) \cdots (1 - \rho_N) = \frac{F_N}{F_{N+1}} \frac{F_{N-1}}{F_N} \cdots \frac{F_1}{F_2} = \frac{F_1}{F_{N+1}} = \frac{1}{F_{N+1}} \quad (4.7)$$

Dado que el método de Fibonacci hace uso de los valores óptimos de ρ_1, ρ_2, \dots , el factor de reducción es menor que aquel del método de la búsqueda del segmento dorado. En otras palabras, el método descrito en esta sección es mejor y resulta en un menor rango de incertidumbre mejor.

4.1.3 Método de Newton

Si ahora se asume que en cada punto evaluado $x^{(k)}$ puede calcularse $f(x^{(k)})$, $f'(x^{(k)})$, y $f''(x^{(k)})$. Puede encontrarse una función cuadrática a través de $x^{(k)}$ que se adapte a su primera y segunda derivadas con sólo una función f . Entonces, dicha función tiene la forma

$$q(x) = f(x^{(k)}) + f'(x^{(k)})(x - x^{(k)}) + \frac{1}{2}f''(x^{(k)})(x - x^{(k)})^2 \quad (4.8)$$

Nótese que $q(x^{(k)}) = f(x^{(k)})$, $q'(x^{(k)}) = f'(x^{(k)})$, y $q''(x^{(k)}) = f''(x^{(k)})$. Entonces, en lugar de minimizar f , se minimiza la aproximación q . La condición necesaria de primer orden para un mínimo de q sostiene

$$0 = q'(x) = f'(x^{(k)}) + f''(x^{(k)})(x - x^{(k)}) \quad (4.9)$$

Si se hace $x = x^{(k+1)}$, se obtiene

$$x^{(k+1)} = x^{(k)} - \frac{f'(x^{(k+1)})}{f''(x^{(k+1)})} \quad (4.10)$$

El algoritmo se detiene cuando se alcanza una condición tal que $|x^{(k+1)} - x^{(k)}| < \epsilon$, en donde ϵ representa la precisión establecida por el problema.

El método de Newton trabaja bien si $f''(x) > 0$ en todas las evaluaciones. Sin embargo, si para alguna x se tiene que $f''(x) < 0$, el método podría fallar en converger al mínimo de la función.

Otra manera de ver este método es como una manera de aproximar la primera derivada de f a cero, de hecho, si se hace $g(x) = f'(x)$, entonces se obtiene la fórmula para la solución iterativa de la ecuación $g(x) = 0$:

$$x^{(k+1)} = x^{(k)} - \frac{g(x^{(k)})}{g'(x^{(k)})} \quad (4.11)$$

Cuando se usa el método de Newton para resolver ecuaciones de la forma $g(x) = 0$ se hace referencia a él como *método de tangentes de Newton*. Si se dibuja una tangente a $g(x)$ en un punto dado $x^{(k)}$, entonces esta línea tangente interseca el eje x en el punto $x^{(k+1)}$, el cual se espera se encuentre más cerca del mínimo x^* de $g(x) = 0$. La pendiente de $g(x)$ en $x^{(k)}$ es

$$g'(x^{(k)}) = \frac{g(x^{(k)})}{x^{(k)} - x^{(k+1)}} \quad (4.12)$$

que al resolver por $x^{(k+1)}$ da la ecuación 4.11.

Esta manera de optimizar una función puede fallar si la primera aproximación de la raíz es tal que la razón $g(x^{(0)})/g'(x^{(0)})$ no es suficientemente pequeña. Por lo tanto, una aproximación inicial a la raíz es muy importante.

Cabe mencionar que el método de Newton puede expandirse al caso multidimensional al expandir la expansión en serie de Taylor de f en un punto $x^{(k)}$, como se hizo en la ecuación 4.8 que, tomando sólo los dos primeros términos puede expresarse como

$$f(x) = f(x^{(k)}) + (x - x^{(k)})^T g^{(k)} + \frac{1}{2}(x - x^{(k)})^T F(x^{(k)})(x - x^{(k)}) \triangleq q(x) \quad (4.13)$$

en donde $g^{(k)}$ y $F(x^{(k)})$ representan el vector gradiente y la matriz Hessiana, respectivamente, y se encuentran dadas por

$$g^{(k)} = \nabla f(x^{(k)}) = \begin{bmatrix} \frac{\partial f}{\partial x_1}(x^{(k)}) \\ \vdots \\ \frac{\partial f}{\partial x_n}(x^{(k)}) \end{bmatrix} \quad (4.14)$$

y

$$F(x^{(k)}) = D^2(x^{(k)}) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_2 \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_n \partial x_1} \\ \frac{\partial^2 f}{\partial x_1 \partial x_2} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_n \partial x_2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_1 \partial x_n} & \frac{\partial^2 f}{\partial x_2 \partial x_n} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix} \quad (4.15)$$

Al aplicar la condición necesaria de primer orden a q , la expresión 4.15 puede reescribirse como

$$0 = \nabla q(x) = g^{(k)} + F(x^{(k)})(x - x^{(k)}) \quad (4.16)$$

Considérese $d^{(k)} = (x - x^{(k)})$, el proceso para obtener la k -ésima iteración en dos pasos puede resumirse entonces de la siguiente manera

1. Resolver $F(x^{(k)})d^{(k)} = -g^{(k)}$ para $d^{(k)}$;
2. Hacer $x^{(k+1)} = x^{(k)} + d^{(k)}$

El paso 1 requiere solucionar un sistema de ecuaciones lineales de $n \times n$. Entonces, al utilizar el método de Newton, es necesaria una técnica eficiente para resolver sistemas de ecuaciones lineales.

4.1.4 Método de la secante

El método de Newton hace uso de las segundas derivadas para minimizar f , como se muestra en la ecuación 4.10. Si no se puede obtener información acerca de la segunda derivada, puede hacerse una aproximación a esta usando información de la primera derivada. En particular, puede aproximarse $f''(x^k)$ por medio de

$$\frac{f'(x^{(k)}) - f'(x^{(k-1)})}{x^{(k)} - x^{(k-1)}} \quad (4.17)$$

Usando la aproximación anterior de la segunda derivada puede obtenerse el algoritmo

$$x^{(k+1)} = x^{(k)} - \frac{x^{(k)} - x^{(k-1)}}{f'(x^{(k)}) - f'(x^{(k-1)})} \quad (4.18)$$

Nótese que el algoritmo requiere de dos puntos de inicio, denotados por $x^{(-1)}$ y $x^{(0)}$. El algoritmo de la secante puede representarse en forma equivalente como

$$x^{(k+1)} = \frac{f'(x^{(k)})x^{(k-1)} - f'(x^{(k-1)})x^{(k)}}{f'(x^{(k)}) - f'(x^{(k-1)})} \quad (4.19)$$

Al igual que en el método de Newton, el de la secante tampoco incluye directamente valores $f(x^{(k)})$, sino que trata de hacer que la derivada f' se aproxime a cero. De hecho, puede interpretarse el método de la secante como un algoritmo para resolver ecuaciones de la forma $g(x) = 0$. Específicamente, el algoritmo de la secante para encontrar una raíz de esta ecuación toma la forma

$$x^{(k+1)} = \frac{g(x^{(k)})x^{(k-1)} - g(x^{(k-1)})x^{(k)}}{g(x^{(k)}) - g(x^{(k-1)})} \quad (4.20)$$

A diferencia del método de Newton, que utiliza la pendiente de g para determinar el siguiente punto, este método hace uso de la 'secante' entre $(k - 1)$ -ésimo y k -ésimo puntos para determinar el $(k + 1)$ -ésimo punto.

Los métodos mencionados en esta sección siguen el principio básico de desplazarse a lo largo de una función hasta encontrar un óptimo. La mayor diferencia entre ellos es decisión de hacia dónde moverse y que tan lejos. Sin embargo, estos algoritmos siguen teniendo el defecto de que suelen culminar al encontrar un mínimo local en lugar de uno global. Como alternativa a estos algoritmos, existe otro grupo de métodos basados en modelos genéticos, este tipo de métodos de búsqueda generan nuevos puntos en el espacio de búsqueda al aplicar diversos operadores a los puntos ya conocidos y evaluados y se mueven de manera estadística hacia puntos más óptimos. La siguiente parte de esta tesis trata de dichos algoritmos.

4.2 Métodos de búsqueda y aprendizaje basados en modelos naturales

Muchas técnicas computacionales toman prestadas ideas de la naturaleza de una manera u otra: las redes neuronales imitan la estructura del cerebro humano, los algoritmos genéticos simulan evolución y las colonias de insectos han inspirado algoritmos de optimización estocástica. Estas técnicas se encuentran caracterizadas por paralelismo, adaptabilidad, retroalimentación y algún elemento de aleatoriedad.

A continuación se presentan algunos de los algoritmos basados en distintos modelos naturales, se hace especial énfasis en la sección de algoritmos genéticos por su relevancia para este trabajo de tesis.

4.2.1 Redes neuronales

Las redes neuronales han encontrado numerosas aplicaciones prácticas desde la cancelación de eco en líneas telefónicas [Rangwala y Dornfeld, 1989] hasta la interpretación de datos obtenidos por electroencefalogramas [Kelly *et al*, 1990]. La esencia de las redes neuronales yace en el peso de las conexiones entre neuronas, estos pesos son a menudo referidos como parámetros de aprendizaje o entrenamiento.

Una red neuronal artificial es un circuito compuesto por circuitos interconectados más simples llamados neuronas. Cada neurona representa un mapa, típicamente con varias entradas y una salida, dicha salida de la neurona es una función de la suma de las entradas, como se ilustra en la figura 4.4 y se denomina como función de activación. La salida de una neurona puede ser aplicada a su vez, como entrada a varias otras neuronas y por lo tanto el símbolo de una neurona simple muestra múltiples flechas saliendo de ella.

La red neuronal consiste de neuronas interconectadas, donde las entradas de cada neurona consisten de salidas ponderadas de otras neuronas. Las interconexiones permiten el intercambio de información entre neuronas. En una red neuronal multi-capa, las neuronas se interconectan en

capas, de tal manera que el flujo de datos ocurra sólo en una dirección. Cada neurona recibe entonces información sólo de neuronas en la capa previa: las entradas a cada neurona son salidas ponderadas de neuronas en la capa anterior. Este tipo de estructura se ilustra en la figura 4.5. La primera capa en la red se denomina como capa de entrada, y la última como capa de salida. Las capas entre la entrada y la salida son llamadas capas ocultas.

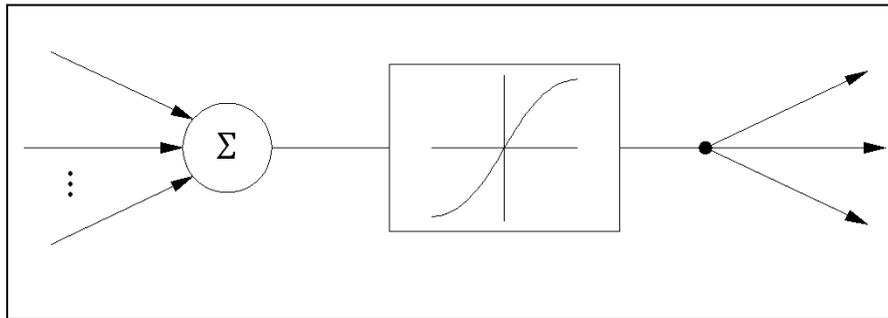


Figura 4.4 Neurona simple

Puede verse a una red neuronal como una implementación de un mapeo de \mathbb{R}^n a \mathbb{R}^m , en donde n es el número de entradas x_1, \dots, x_n y m el de salidas y_1, \dots, y_m . El mapeo implementado por una red neuronal depende de los pesos de las interconexiones en la red. Por lo tanto, puede cambiarse dicho mapeo al ajustar los valores de estos pesos. La información obtenida de los mapeos se almacena en los pesos de las neuronas, la red neuronal es, por lo tanto, a representación distribuida del mapeo.

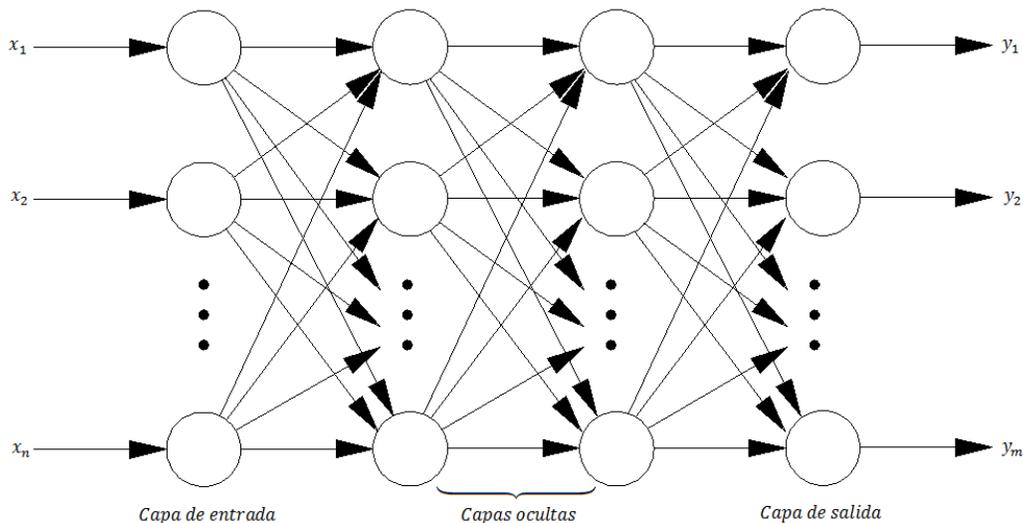


Figura 4.5 Estructura de una red neuronal multi-capas

El problema de entrenamiento puede verse también como el problema de optimización. Pueden usarse técnicas de optimización y de búsqueda para la selección de pesos. Los algoritmos de entrenamiento se encuentran basados de hecho en métodos de optimización. Este método de aprendizaje es conocido como *aprendizaje supervisado*. El aprendizaje no supervisado es referido en [Hertz *et al*, 1991].

4.2.1.1 Entrenamiento de una neurona simple

Para una neurona como la mostrada en la figura 4.6 la función de activación es simplemente la identidad (función lineal con pendiente unitaria). La neurona implementa el siguiente mapeo lineal de \mathbb{R}^n a \mathbb{R}

$$y = \sum_{i=1}^n w_i x_i = \mathbf{x}^T \mathbf{w} \quad (4.21)$$

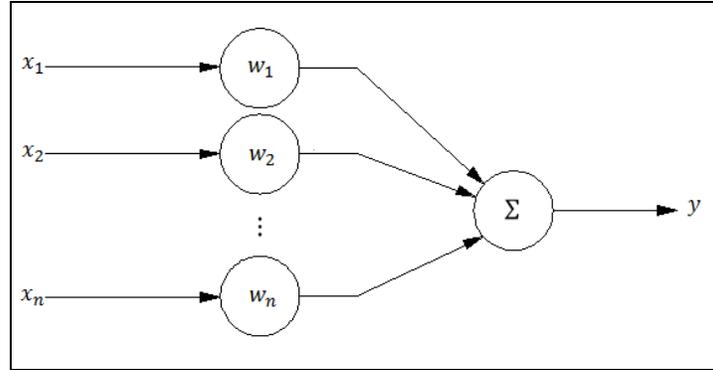


Figura 4.6 Neurona simple lineal

en donde $\mathbf{x} = [x_1, \dots, x_n]^T \in \mathbb{R}^n$ es el vector de entradas, $y \in \mathbb{R}$ es la salida y $\mathbf{w} = [w_1, \dots, w_n] \in \mathbb{R}^n$ es el vector de pesos. Se desea encontrar el valor de los pesos w_1, \dots, w_n de tal manera que la neurona aproxime un mapa $F : \mathbb{R}^n \rightarrow \mathbb{R}$ de manera tan cercana como sea posible. Para esto, se utiliza el grupo de entrenamiento que consta de p pares $\{(\mathbf{x}_{d,1}, y_{d,1}), \dots, (\mathbf{x}_{d,p}, y_{d,p})\}$, en donde $\mathbf{x}_{d,i} \in \mathbb{R}^n$ y $y_{d,i} \in \mathbb{R}, i = 1, \dots, p$. Entonces, para cada i , $y_{d,i} = F(\mathbf{x}_{d,i})$. El problema de entrenamiento puede formularse como el siguiente problema de optimización

$$\text{minimizar } \frac{1}{2} \sum_{i=1}^p (y_{d,i} - \mathbf{x}_{d,i}^T \mathbf{w})^2 \quad (4.22)$$

La función objetivo representa la suma de los errores cuadráticos entre las salidas deseadas $y_{d,i}$ y las salidas correspondientes de la neurona $\mathbf{x}_{d,i}^T \mathbf{w}$. El factor $1/2$ se agrega por conveniencia de notación y no incurre ningún cambio en la minimización.

Si se define la matriz $\mathbf{X}_d \in \mathbb{R}^{n \times p}$ y el vector $\mathbf{y}_d \in \mathbb{R}^p$, la función anterior puede reescribirse de forma matricial como

$$\mathbf{X}_d = [\mathbf{x}_{d,1} \cdots \mathbf{x}_{d,p}] \quad (4.23)$$

$$\mathbf{y}_d = \begin{bmatrix} y_{d,1} \\ \vdots \\ y_{d,p} \end{bmatrix} \quad (4.24)$$

Entonces, el problema de optimización se convierte en

$$\text{minimizar } \frac{1}{2} \|\mathbf{y}_d - \mathbf{X}_d^T \mathbf{w}\|^2 \quad (4.25)$$

Considérese el caso en que $p \leq n$, es decir, cuando se tienen a lo mucho tantos pares como número de pesos. Por conveniencia, se asume el rango $\mathbf{X}_d^T = p$. En este caso, existen un número infinito de puntos que satisfacen $\mathbf{y}_d = \mathbf{X}_d^T \mathbf{w}$. Un posible criterio de selección es el de minimizar la norma de la solución, dada por $\mathbf{w}^* = \mathbf{X}_d (\mathbf{X}_d^T \mathbf{X}_d)^{-1} \mathbf{y}_d$. Un algoritmo eficiente para encontrar la solución es el de Kaczmarz [Widrow y Lehr; 1990], que para este problema toma la forma

$$\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} + \mu \frac{e_k \mathbf{x}_{d,R(k)}}{\|\mathbf{x}_{d,R(k)}\|^2} \quad (4.26)$$

en donde $\mathbf{w}^{(0)} = 0$ y

$$e_k = y_{d,R(k)} - \mathbf{x}_{d,R(k)}^T \mathbf{w}^{(k)} \quad (4.27)$$

En las ecuaciones 4.26 y 4.27, $R(k)$ es el entero único en $\{0, \dots, p-1\}$ que satisface $k = lp + R(k)$ para algún entero l ; es decir, $R(k)$ es el remanente que resulta de dividir k entre p . Una neurona simple con el método de entrenamiento antes descrito es conocida como *Adaline*, que es el acrónimo de '*adaptive linear element*' (figura 4.7),

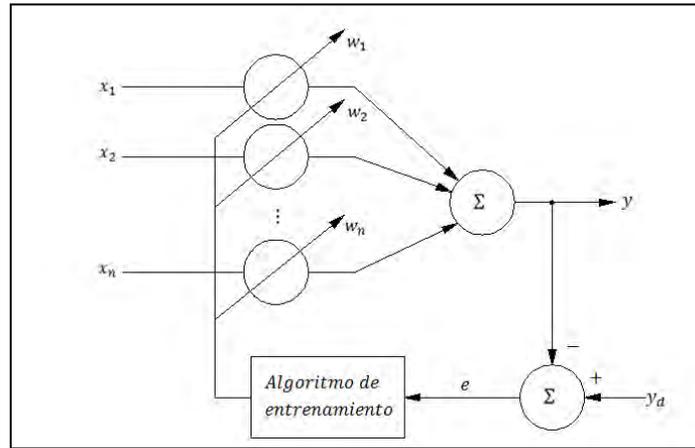


Figura 4.7 Adaline

Considérese ahora el caso en que $p > n$ en donde se tiene mayor número de puntos de entrenamiento que de pesos. Se asume el rango $\mathbf{X}_d^T = p$. En este caso, la función objetivo $\frac{1}{2} \|\mathbf{y}_d - \mathbf{X}_d^T \mathbf{w}\|^2$ es una función convexa cuadrática de \mathbf{w} , pues la matriz $\mathbf{X}_d \mathbf{X}_d^T$ es definida y positiva. Una posible solución a este problema de optimización es usar el método de gradiente, que en este caso queda como

$$\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} + \alpha_k \mathbf{X}_d \mathbf{e}^{(k)} \quad (4.28)$$

donde $\mathbf{e}^{(k)} = \mathbf{y}_d - \mathbf{X}_d^T \mathbf{w}^{(k)}$.

En este caso también se asumió que la función de activación es la de identidad. La derivación y análisis de los algoritmos puede extenderse al caso de una función de activación general diferenciable f_a . Específicamente, la salida de la neurona en este caso está dada por

$$y = f_a \left(\sum_{i=1}^n w_i x_i \right) = f_a(\mathbf{x}^T \mathbf{w}) \quad (4.29)$$

El algoritmo para el caso de un solo par de entrenamiento (x_d, y_d) tiene la forma

$$\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} + \mu \frac{e_k \mathbf{x}_d}{\|\mathbf{x}_d\|^2} \quad (4.30)$$

y el error está dado por $e_k = y_d - f_a(\mathbf{x}_d^T \mathbf{w}^{(k)})$.

4.2.1.2 Entrenamiento de redes multicapas: algoritmo de retro propagación

Considérese una red multicapa como la mostrada en la figura 4.5. Por simplicidad se considera que hay una sola capa oculta. Hay n entradas x_i con $i = 1, \dots, n$ y m salidas y_s , $s = 1, \dots, m$. Existen también l neuronas en la capa oculta, cuyas salidas son z_j , donde $j = 1, \dots, l$. Las entradas x_1, \dots, x_n se distribuyen a las neuronas en la capa oculta. Puede pensarse que las neuronas en la capa de entrada como elementos lineales de una salida y una entrada, siendo su función de activación el mapa identidad. Esta red tendrá una forma como la ilustrada en la figura 4.8, en donde las neuronas de la capa de entrada aparecen como divisores de señal. Se denotan las funciones de activación de las neuronas en la capa oculta con f_j^h , y las funciones de activación en la capa de salida con f_s^o .

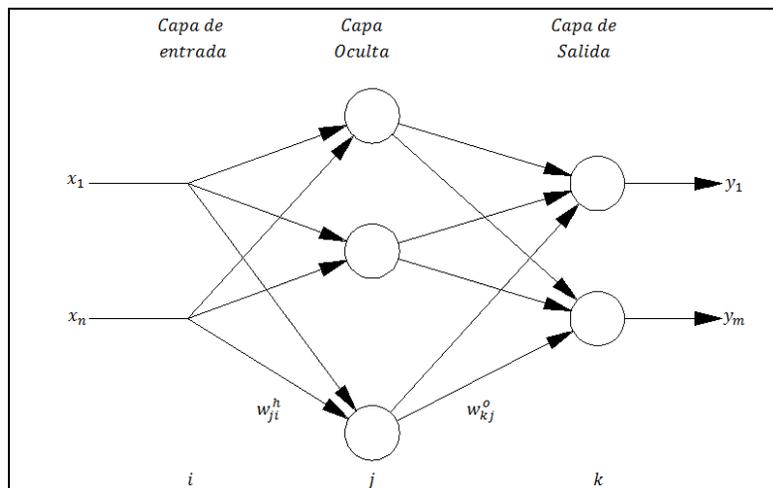


Figura 4.8 Red neuronal de tres capas

Los pesos de las entradas están denotados en la capa oculta por w_{ji}^h y los pesos de las entradas desde la capa oculta hacia la capa de salida por w_{kj}^o . Para encontrar la fórmula explícita para hacer

el mapeo correspondiente, se denota la entrada a la j -ésima neurona in la capa oculta por v_j , y la salida de la misma neurona como z_j . Se tiene entonces

$$\begin{aligned} v_j &= \sum_{i=1}^n w_{ji}^h x_i \\ z_j &= f_j^h \left(\sum_{i=1}^n w_{ji}^h x_i \right) \end{aligned} \quad (4.31)$$

La salida de la s -ésima neurona de la capa de salida es

$$y_s = f_s^o \left(\sum_{j=1}^l w_{sj}^o z_j \right) \quad (4.32)$$

La relación entre las entradas x_i y la s -ésima salida y_s está dada entonces por

$$\begin{aligned} y_s &= f_s^o \left(\sum_{j=1}^l w_{sj}^o f_j^h(v_j) \right) \\ &= f_s^o \left(\sum_{j=1}^l w_{sj}^o f_j^h \left(\sum_{i=1}^n w_{ji}^h x_i \right) \right) = F_s(x_1, \dots, x_n) \end{aligned} \quad (4.33)$$

El mapeo completo implementado por la red neuronal queda de la siguiente manera

$$\begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix} = \begin{bmatrix} F_1(x_1, \dots, x_n) \\ \vdots \\ F_m(x_1, \dots, x_n) \end{bmatrix} \quad (4.34)$$

Ahora considérese el problema de entrenar a la red neuronal. El grupo de entrenamiento consiste de varios pares (x_d, y_d) y su entrenamiento se lleva normalmente a cabo con un par a la vez [Hush y Horne, 1993; Rumelhart *et al*, 1986]. El entrenamiento de la red neuronal involucra el ajuste de los pesos de la red de tal manera que la salida generada para una entrada dada $\mathbf{x}_d = [x_{d1}, \dots, x_{dn}]^T$ este tan cerca de \mathbf{y}_d como sea posible. Formalmente, esto puede formularse como el siguiente problema de optimización:

$$\text{minizar } \frac{1}{2} \sum_{s=1}^m (y_{ds} - y_s)^2 \quad (4.35)$$

en donde y_s , $s = 1, \dots, m$, son las salidas de la red neuronal en respuesta a las entradas x_{d1}, \dots, x_{dn} dadas por la ecuación 4.26. Por simplicidad, se utilizará el símbolo \mathbf{w} para el vector

$$\mathbf{w} = \{w_{ji}^h, w_{sj}^o; i = 1, \dots, n, j = 1, \dots, l, s = 1, \dots, m\}, \quad (4.36)$$

y el símbolo E para la función objetivo a minimizar; esto queda como

$$\begin{aligned}
E(\mathbf{w}) &= \frac{1}{2} \sum_{s=1}^m (y_{ds} - y_s)^2 \\
&= \frac{1}{2} \sum_{s=1}^m \left(y_{ds} - f_s^o \left(\sum_{j=1}^l w_{sj}^o f_j^h \left(\sum_{i=1}^n w_{ji}^h x_i \right) \right) \right)^2
\end{aligned} \tag{4.37}$$

Para resolver este problema de optimización, puede utilizarse cualquier algoritmo, úsese en este caso uno de gradiente con un tamaño de paso fijo. Para formular el algoritmo, necesita computarse las derivadas parciales de E con respecto a cada componente de \mathbf{w} . Para llevar a cabo este paso, se fijarán los índices i, j y s . Para computar la derivada parcial de E con respecto a w_{sj}^o puede escribirse

$$\frac{1}{2} \sum_{p=1}^m \left(y_{dp} - f_p^o \left(\sum_{q=1}^l w_{pq}^o z_q \right) \right)^2 \tag{4.38}$$

en donde para cada $q = 1, \dots, l$ $z_q = f_q^h(\sum_{i=1}^n w_{qi}^h x_{di})$. Usando la regla de la cadena se obtiene

$$\frac{\partial E}{\partial w_{sj}^o}(\mathbf{w}) = -(y_{ds} - y_s) f_s^{o'} \left(\sum_{q=1}^l w_{pq}^o z_q \right) z_j \tag{4.39}$$

En la expresión anterior $f_s^{o'}$ es la derivada de f_s^o . Considérese ahora el término

$$\delta_s = (y_{ds} - y_s) f_s^{o'} \left(\sum_{q=1}^l w_{pq}^o z_q \right) \tag{4.40}$$

Puede pensarse en δ_s como un error de salida escalada, dado que es la diferencia entre la salida actual de la red neuronal y_s y la salida deseada y_{ds} , escalada por $f_s^{o'}(\sum_{q=1}^l w_{pq}^o z_q)$. Al usar la notación de δ_s , la ecuación 4.39 se simplifica en

$$\frac{\partial E}{\partial w_{sj}^o}(\mathbf{w}) = -\delta_s z_j \tag{4.41}$$

Para computar la derivada parcial de E con respecto a w_{ji}^h se inicia con la ecuación

$$E(\mathbf{w}) = \frac{1}{2} \sum_{p=1}^m \left(y_{dp} - f_p^o \left(\sum_{q=1}^l w_{pq}^o f_q^h \left(\sum_{r=1}^n w_{qr}^h x_{dr} \right) \right) \right)^2 \tag{4.42}$$

Al seguir el mismo procedimiento que en el caso anterior, se obtiene

$$\frac{\partial E}{\partial w_{ji}^h}(\mathbf{w}) = - \left(\sum_{p=1}^m \delta_p z_q \right) f_j^{h'}(v_j) x_{di} \quad (4.43)$$

en donde v_j está dado por la ecuación 4.31, $f_j^{h'}$ representa la derivada de f_j^h y δ_p es

$$\delta_p = (y_{dp} - y_p) f_p^{o'} \left(\sum_{q=1}^l w_{pq}^o z_q \right) \quad (4.44)$$

Para formular el algoritmo de gradiente para actualizar los pesos de la red neuronal se escriben las ecuaciones de actualización para los dos grupos de ponderaciones w_{sj}^o y w_{ji}^h de manera separada. Se tiene entonces

$$\begin{aligned} w_{sj}^{o(k+1)} &= w_{sj}^{o(k)} + \eta \delta_s^{(k)} z_j^{(k)} \\ w_{ji}^{h(k+1)} &= w_{ji}^{h(k)} + \eta \left(\sum_{p=1}^m \delta_p^{(k)} w_{pj}^{o(k)} \right) f_j^{h'}(v_j^{(k)}) x_{di} \end{aligned} \quad (4.45)$$

en donde η es el tamaño de paso fijo y

$$\begin{aligned} v_j^{(k)} &= \sum_{i=1}^n w_{ji}^{h(k)} x_i \\ z_j^{(k)} &= f_j^h(v_j^{(k)}) \\ y_s^{(k)} &= f_s^o \left(\sum_{q=1}^l w_{sq}^o z_q^{(k)} \right) \\ \delta_s^{(k)} &= (y_{ds} - y_s^{(k)}) f_s^{o'} \left(\sum_{q=1}^l w_{pq}^o z_q^{(k)} \right) \end{aligned} \quad (4.46)$$

Los pasos descritos en las expresiones 4.34 son comúnmente referidas en la literatura como el algoritmo de retro-propagación. La razón de su nombre se debe a que los errores a la salida $\delta_1^{(k)}, \dots, \delta_m^{(k)}$ se propagan de regreso desde la capa de salida a la capa oculta y se utilizan para actualizar la ecuación de los pesos de la misma. Para el caso de múltiples capas, los errores de salida se propagan hacia atrás de capa en capa para actualizar los pesos en cada una de estas.

En todos los análisis hechos en esta sección se consideró el uso de la función identidad o de escalón unitario, sin embargo, es importante resaltar que existen más funciones de activación, entre las más utilizadas se encuentran la función hiperbólica y la sigmoide, un análisis de estas se encuentran en [Cuevas, 2000].

4.2.2 Recocido simulado

El método de recocido simulado fue introducido en la década de 1980 por Kirkpatrick y sus colegas [Kirkpatrick *et al.*, 1983], que se basaron en ideas formuladas en los años 50's [Metropolis, 1953]. El método consiste en simular el proceso de recocido en el cual una sustancia se calienta por encima de su punto de fusión y luego se enfría de manera gradual para producir una estructura cristalina, lo cual minimiza su función de distribución de energía.

La analogía en algoritmo de este proceso consiste en comenzar con valores aleatorios de las variables de la función de costo. Calentar significa modificar de manera aleatoria los valores de la variable. Mayor calor implica mayores fluctuaciones en la aleatoriedad. Si la salida decrece, entonces el nuevo grupo de variables reemplaza al grupo anterior. En caso contrario, si la salida es mayor entonces esta es aceptada a condición de que

$$r \leq e^{[f(p_{viejo}) - f(p_{nuevo})]/T} \quad (4.47)$$

en donde r es un número aleatorio uniforme y T es una variable análoga a la temperatura. De otra manera, el nuevo grupo de variables es rechazado. Entonces, aún si un grupo de variables lleva a un costo peor, puede aceptarse con una cierta probabilidad. El nuevo grupo se encuentra al tomar un paso aleatorio del grupo viejo

$$p_{nuevo} = d \cdot p_{viejo} \quad (4.48)$$

La variable d puede ser distribuida de manera uniforme o normal alrededor de p_{viejo} . Este grupo de variables controla el tamaño de paso, de tal manera que al inicio del proceso, el algoritmo es forzado a hacer grandes cambios en los valores de las variables. En ocasiones, los cambios mueven al algoritmo lejos del óptimo, lo cual obliga al algoritmo a explorar nuevas regiones del espacio variable. Después de un cierto número de iteraciones, los nuevos grupos de variables dejan de llevar a costos más bajos. En este punto los valores de T y d se decrementan en algún porcentaje y el algoritmo se repite. El algoritmo se detiene cuando $T \approx 0$. El decremento en T es conocido como plan de enfriamiento. Si la temperatura inicial es T_o y la temperatura final T_N , entonces la temperatura en un paso n está dada por

$$T_n = f(T_o, T_N, N, n) \quad (4.49)$$

en donde f decrece con el tiempo. Algunos planes de enfriamiento son

1. Decremento lineal: $T_n = T_o - n(T_o - T_N)/N$.
2. Decremento geométrico: $T_n = 0.99T_{n-1}$.
3. Óptimo de Hayjek: $T_n = c/\log(1 + n)$, donde c es la menor variación requerida para salir de cualquier mínimo local.

Existen aún más variaciones posibles. La temperatura es usualmente disminuida lentamente, de tal manera que el algoritmo tenga oportunidad de encontrar el valle correcto antes de intentar llegar al punto más bajo del mismo.

4.2.3 Optimización de enjambre de partículas

Este método de optimización se formuló en 1995 por Kennedy y Eberhart [22], el proceso detrás del algoritmo fue inspirado por el comportamiento social de algunos animales como las parvadas de pájaros, por ejemplo. El método de optimización de enjambre de partículas (PSO, por sus siglas en inglés) no cuenta con operadores evolutivos como pueden ser la cruce y la mutación. Las filas en una matriz, correspondientes a las variables de la función son llamadas partículas y tienen una codificación continua. Cada partícula se mueve alrededor de la superficie de costo con una velocidad. Las partículas actualizan sus velocidades y posiciones basándose en las mejores soluciones locales y globales:

$$v_{m,n}^{nueva} = v_{m,n}^{vieja} + \Gamma_1 \times r_1 (p_{m,n}^{mejor\ local} - p_{m,n}^{viejo}) + \Gamma_2 \times r_2 (p_{m,n}^{mejor\ global} - p_{m,n}^{viejo}) \quad (4.50)$$

$$p_{m,n}^{nuevo} = p_{m,n}^{viejo} + v_{m,n}^{nueva} \quad (4.51)$$

en donde

$v_{m,n}$ = velocidad de la partícula,

$p_{m,n}$ = variables de la partícula,

r_1, r_2 = números aleatorios uniformes,

Γ_1, Γ_2 = factores de aprendizaje = 2,

$p_{m,n}^{mejor\ local}$ = mejor solución local,

$p_{m,n}^{mejor\ global}$ = mejor solución global.

El algoritmo PSO actualiza el vector de velocidad para cada partícula y agrega esa velocidad a las posiciones de todas las partículas o valores. Las actualizaciones en la velocidad se ven influenciadas tanto por la mejor solución global asociada con el menor costo encontrado por una partícula en todo el algoritmo y la mejor solución local asociada con el menor costo en la población actual. Si la mejor solución local tiene un costo menor que aquel de la solución global actual, entonces la solución local en cuestión reemplaza a la global. La velocidad de partícula evoca métodos de optimización que utilizan información de derivadas, pues la velocidad es la derivada de la posición. La constante Γ_1 se denomina parámetro cognitivo, mientras que Γ_2 lleva el nombre de parámetro social. Las ventajas del PSO son que es fácil de implementar y existen pocos parámetros que ajustar.

4.2.4 Optimización de colonia de hormigas

La base de este algoritmo es el hecho de que las hormigas encuentran el camino más corto hacia una fuente de comida al dejar un rastro de un químico llamado feromona mientras caminan. Las hormigas que elijan el camino más corto crearán un rastro de feromona más fuerte y rápido que aquellas que elijan un camino más largo. Dado que las feromonas más fuertes atraen mejor a las

hormigas, un mayor número de estas elegirán el camino más corto hasta que eventualmente todas las hormigas hayan elegido el camino más corto.

Los primeros algoritmos de optimización de colonia de hormigas (ACO) se diseñaron para resolver el problema del vendedor ambulante, dado que guarda cierto parecido con el problema de encontrar el camino más corto a una fuente de comida [Dorigo y Maria, 1997]. Los intentos iniciales del algoritmo no fueron satisfactorios hasta que este se acopló con un optimizador local. Uno de los problemas que tiene este algoritmo es la convergencia prematura a una solución mucho menos que óptima dado que se utilizó demasiada feromona virtual demasiado rápido. Para evitar este estancamiento, se implementa un proceso llamado evaporación de feromonas. En otras palabras, las feromona asociada con una solución desaparece después de un cierto periodo de tiempo.

El algoritmo comienza al asignar a cada hormiga un punto de partida aleatorio. El siguiente punto se elige por una probabilidad ponderada que es una función de la fuerza de la feromona puesta en el camino y la distancia del punto. La probabilidad de que una hormiga k viaje de un punto m a un punto n está dada por

$$p_{mn}^k = \frac{\tau_{mn}^a / d_{mn}^b}{\sum_q \tau_{mq}^a / d_{mq}^b} \quad (4.52)$$

en donde τ representa la fuerza de la feromona, q son los puntos en el recorrido por los que pasa k antes de llegar a m , a es el peso o ponderación de la feromona; cuando $a = 0$, se selecciona el punto más cercano y b es el peso asignado a la distancia; cuando $b = 0$, se ignora la distancia entre los puntos.

Los caminos cortos con alto valor de feromonas tienen la mayor probabilidad de selección. En los caminos iniciales, se dejan feromonas en rutas ineficientes. Como consecuencia algunas de estas feromonas deben de evaporarse en tiempo o el algoritmo convergerá a una ruta ineficiente. Para evitar esta situación es necesario tener una estrategia de elitismo. Como resultado, a la feromona en el mejor camino se le asigna algún peso al calcular los nuevos niveles de feromonas. La formula de actualización de feromonas está dada por [Bonabeau *et al*, 1999]:

$$\tau_{mn} = (1 - \xi)\tau_{mn} + \sum_{k=1}^{N_h} \tau_{mn}^k + \varepsilon\tau_{mn}^{elite} \quad (4.53)$$

en donde N_h denota el número de hormigas, τ_{mn}^k es el valor de la feromona dejada por la hormiga k entre los puntos m y n , ξ representa la constante de evaporación de feromonas, ε es la constante de peso del camino elite y τ_{mn}^{elite} es el valor de feromona dejado en el mejor camino encontrado en este punto del algoritmo.

4.2.5 Algoritmos genéticos

Un algoritmo genético, o GA (por sus siglas en inglés), es una técnica de búsqueda probabilística cuyas raíces se encuentran en los principios de genética y selección natural. El inicio de los algoritmos genéticos se adjudica a John Holland, quien concibió las ideas básicas a finales de la década de 1960 y principios de la de 1970 y finalmente se popularizaron gracias a uno de sus estudiantes, David Goldberg, quien fue capaz de resolver el difícil problema del control de transmisión en tuberías de gas [Goldberg, 1980]. Desde su creación, los algoritmos genéticos han sido utilizados ampliamente como herramienta en programación de computadoras e inteligencia artificial [Holland, 1992a; Koza, 1992; Mitchell, 1996], optimización [Davis, 1987; Jang, 1997; Then, 1994], entrenamiento de redes neuronales [Kozek *et al*, 1993], entre otras.

El trabajo de Holland mostró la utilidad de los algoritmos genéticos en la optimización de funciones e hizo el primer esfuerzo por encontrar parámetros óptimos para la aplicación del GA, aunque ha sido probablemente Goldberg quien más ha contribuido a esta área con varias aplicaciones [Goldberg, 1989]. Algunas ventajas de los algoritmos genéticos son [Haupt y Haupt, 2004]:

- Optimiza con variables continuas o discretas,
- No requiere información de las derivadas,
- Búsqueda simultánea en un espacio muestral grande de la superficie de costo,
- Puede manejar un amplio número de variables,
- Optimiza variables con superficies de costo muy complejas (puede salir de un óptimo local),
- Proporciona una lista de variables óptimas, no sólo una solución,
- Puede codificar las variables de tal manera que la optimización se lleva a cabo con variables codificadas, y
- Funciona con información generada numéricamente, datos experimentales o funciones analíticas.

La idea general de un algoritmo genético es optimizar (minimizar o maximizar) una función $f(x)$ sujeta a $x \in \Omega$ en donde x es un valor cualquiera restringido al espacio Ω . Se comienza con un grupo de puntos $P(0)$ dentro del espacio Ω , denotado como población inicial. Basado en la evaluación de estos puntos, se crea un nuevo grupo $P(1)$. Para crear este nuevo grupo hace falta aplicar a $P(0)$ algunos operadores llamados cruza y mutación. Este procedimiento se repite de manera iterativa para generar las poblaciones $P(2), P(3), \dots$, hasta que se alcanza un criterio de paro. El objetivo de los operadores aplicados es el de crear una nueva población con una función objetivo en promedio mejor que la población anterior [Davis, 1989].

Cabe destacar que los algoritmos genéticos en realidad no trabajan directamente con puntos del grupo Ω , sino con codificaciones del mismo, es decir, se necesita un mapeo de Ω a un grupo de cadenas de valores compuestas por símbolos, en general, de la misma longitud. A estas cadenas se les conoce como cromosomas. Cada cromosoma consiste de elementos de un cierto grupo de símbolos, llamados alfabeto. Se denota a la longitud de los cromosomas como L , y es el número

de símbolos en la cadena. Para cada cromosoma corresponde un valor de la función objetivo, referido como la aptitud del cromosoma. Para cada cromosoma x , se denota la aptitud relacionada por $f(x)$.

Una vez que se ha establecido el esquema de representación, es decir, la longitud, el alfabeto y codificación del cromosoma, el siguiente paso es inicializar la población $P(0)$ de manera aleatoria. Durante cada iteración k del proceso, se evalúa la aptitud $f(x^{(k)})$ de cada miembro $x^{(k)}$ de la población $P(k)$. Después de que la aptitud de toda la población se ha calculado, se puede crear la nueva población $P(k + 1)$.

En las siguientes secciones se explica de manera más detallada el proceso completo necesario para la implementación de un algoritmo genético.

4.2.5.1 Justificación de la convergencia de algoritmos genéticos

El porqué del funcionamiento de los algoritmos genéticos está basado en argumentos heurísticos. Los mecanismos utilizados en esta técnica mimetizan la noción de la ‘sobrevivencia el más apto’. Se comienza con una población de cromosomas, y de manera selectiva se escogen los más aptos para reproducción, mismos que formarán la nueva generación al combinar información codificada en ellos. De esta manera, el objetivo es asegurar que los miembros más aptos de la población sobrevivan, y el contenido de su información se conserve y combine para producir aún una mejor descendencia.

Para analizar al algoritmo genético de una manera más cuantitativa, necesitan definirse algunos términos. Por conveniencia, se consideran sólo cromosomas codificados con alfabeto binario. Un esquema es un grupo de cromosomas con ciertas características en común. De manera específica, para el caso en que se utiliza un alfabeto binario, un esquema es un grupo de cromosomas que contienen 1s y 0s en lugares particulares. Un esquema se representa usando una notación de cadena en un alfabeto extendido $\{0,1,*\}$. Por ejemplo, la notación $1 * 01$ representa el esquema

$$1 * 01 = \{1001, 1101\},$$

y la notación $0 * 101 *$ representa el esquema

$$0 * 101 * = \{001010, 001011, 011010, 011011\}$$

En la notación de esquema, los números 0 y 1 denotan los valores binarios fijos en los cromosomas que pertenecen al esquema, mientras que el símbolo * denota que no importa el valor. Un cromosoma pertenece a un esquema en particular si para todas las posiciones $j = 1, \dots, L$ el símbolo en la j -ésima posición del cromosoma es el mismo que el símbolo en la misma posición del esquema. Un esquema con r número de * contiene 2^r . También, cualquier cromosoma de longitud L pertenece a 2^L esquemas.

La idea clave al explicar porqué el algoritmo genético funciona es la observación de que si un esquema tiene cromosomas con mejor aptitud que el promedio, entonces el número promedio de

cromosomas que corresponden a este esquema en la piscina de apareamiento $M(k)$ es mayor que el número de cromosomas que coinciden con el esquema en la población $P(k)$. Para cuantificar esta notación, denótese un esquema cualquiera como H y sea $e(H, k)$ el número de cromosomas en $P(k)$ que se adaptan a H ; es decir, $e(H, k)$ es el número de elementos en el grupo $P(k) \cap H$. Sea ahora $f(H, k)$ la aptitud promedio de los cromosomas en $P(k)$ que coinciden con el esquema H . Esto significa que si $P(k) \cap H = \{x_1, \dots, x_{e(H,k)}\}$, entonces

$$f(H, k) = \frac{f(x_1) + \dots + f(x_{e(H,k)})}{e(H, k)} \quad (4.54)$$

Si N es el número de cromosomas en la población y $F(k)$ la suma de los valores de aptitud de los cromosomas en $P(k)$, entonces $\bar{F}(k)$ denota la aptitud promedio de los cromosomas de la población y está dada por

$$\bar{F}(k) = \frac{F(k)}{N} = \frac{1}{N} \sum f(x_i^{(k)}) \quad (4.55)$$

Finalmente, sea $m(H, k)$ el número de cromosomas en $M(k)$ que corresponden al esquema H , en otras palabras, el número de elementos en el grupo $M(k) \cap H$. Entonces, dado un esquema cualquiera, el valor esperado $\mathcal{M}(H, k)$ de $m(H, k)$ está dado por

$$\mathcal{M}(H, k) = \frac{f(H, k)}{\bar{F}(k)} e(H, k) \quad (4.56)$$

Si se analiza la relación 4.56 puede notarse que si un esquema tiene cromosomas con mejor desempeño que el promedio, entonces el número esperado de cromosomas que se adaptan a dicho esquema en la piscina de apareamiento $M(k)$ es mayor que el número de cromosomas que coinciden con el esquema en la población $P(k)$.

Ahora se analiza el efecto de los operadores evolutivos en los cromosomas en la piscina de apareamiento. Se define el orden $o(S)$ de un esquema S es el número de símbolos fijos (no los símbolos *), es decir, si la longitud de este esquema es L , entonces $o(S)$ es L menos el número de símbolos * en S . Por ejemplo: $o(1 * 01) = 4 - 1 = 3$ y $o(0 * 1 * 10) = 6 - 2 = 4$. La longitud $l(S)$ de un esquema S es la distancia entre el primer y último símbolos fijos. Por ejemplo: $l(1 * 01) = 4 - 1 = 3$ y $l(0 * 1 * 10) = 6 - 2 = 4$.

Primeramente se considera el efecto de la cruce en la piscina de apareamiento. Entonces, dado un cromosoma $M(k) \cap H$, la probabilidad que se elige para cruce si ninguno de sus descendientes se encuentra en H se encuentra restringida en su límite superior por

$$p_c \frac{l(H)}{L - 1} \quad (4.57)$$

Puede concluirse entonces que dado un cromosoma $M(k) \cap H$, la probabilidad de que este sea seleccionado o no para cruce, o al menos uno de sus descendientes se encuentre en H después de la operación de cruce, está limitada en su límite inferior a

$$1 - p_c \frac{l(H)}{L-1} \quad (4.58)$$

El efecto del operador de mutación se considera a continuación. Dado un cromosoma en $M(k) \cap H$, la probabilidad de que este permanezca en H después de aplicar la operación de mutación está dada por

$$(1-p_m)^{o(H)} \quad (4.59)$$

Si p_m es suficientemente pequeño, la expresión anterior es aproximadamente igual a

$$p_m o(H) \quad (4.60)$$

Considerando los operadores de mutación y cruce sobre un esquema H , el valor esperado $\varepsilon(H, k+1)$ de $e(H, k+1)$ está dado por

$$\varepsilon(H, k+1) \geq \left(1 - p_c \frac{l(H)}{L-1} - p_m o(H)\right) \frac{f(H, k)}{\bar{F}(k)} e(H, k) \quad (4.61)$$

El teorema anterior indica cómo cambia el número de cromosomas en un esquema particular de una población a la siguiente. Este cambio está influenciado por tres factores correspondientes al desempeño promedio del esquema, el efecto del operador de cruce y la influencia de la operación de muta. Puede notarse que entre mayor sea el desempeño de un esquema, mayor será su descendencia en la siguiente población; el parámetro correspondiente a la mutación depende del orden del esquema, es decir, entre menor sea la cantidad de símbolos fijos dentro del esquema, su descendencia será mayor; por último, para la mutación es importante la longitud del esquema y por lo mismo es importante cuidar de la posición de los valores importantes para reducir este parámetro.

4.2.5.2 Codificación del algoritmo genético

Una de los aspectos más importantes de un algoritmo genético es el tipo de codificación o estructura que tiene un cromosoma o, como se conoce en literatura, el genotipo. Refiriéndose de manera análoga a la terminología utilizada en la biología, puede definirse también al fenotipo como el organismo en su totalidad, en el caso de un algoritmo genético, el fenotipo corresponde al resultado producido por el genotipo en su ambiente o bien, un grupo de parámetros desconocidos o un vector solución.

Muchos autores han señalado que los algoritmos genéticos son, probablemente, en su punto más efectivo cuando la codificación es tan cercana como sea posible al espacio de búsqueda. Para muchos problemas de ciencia e ingeniería esto implica el uso de números en la forma de base 10. En este aspecto existen varias posibilidades para elegir la que mejor se adapte al problema en cuestión.

Aunque una codificación binaria funcionará a menudo muy bien, como Goldberg ha señalado [Goldberg, 1989], un algoritmo genético es típicamente robusto con respecto a la codificación utilizada.

4.2.5.2.1 Codificación binaria

En la sección 4.2.5.1 se justificó el funcionamiento de un algoritmo genético. Para dicha justificación se utilizó un alfabeto de dos valores: 0 y 1. Este tipo de estructura se conoce como codificación binaria, dado el hecho de que tiene sólo dos valores posibles.

El problema principal de este tipo de codificación reside en elegir el número correcto de *bits* o posiciones que tendrá cada variable, además de que, en la mayoría de los casos, es necesario un paso extra dedicado a la decodificación del cromosoma. Al seleccionar el tamaño de cada variable debe de tomarse en cuenta el rango del espacio muestral a explorar y el tamaño de paso deseado. Si se tiene un rango definido entre los límites $[a, b]$ y se desean sólo valores enteros, el número de bits a utilizar estará definido por

$$l = \text{ceil}\{\log_2(b - a)\} \quad (4.62)$$

en donde el operador *ceil* indica que debe de tomarse el siguiente entero superior.

Para el caso en el que se desean valores con parte decimal debe de definirse un tamaño paso Δx , que está definido por

$$\Delta x = \frac{b - a}{2^l - 1} \quad (4.63)$$

de donde se obtiene que el tamaño de la cadena n necesario para representar esta variable se encuentra dado por

$$l = \text{ceil}\left\{\log_2\left(\frac{b - a}{\Delta x} + 1\right)\right\} \quad (4.64)$$

Una codificación típica de n objetos variables $X = (x_1, x_2, \dots, x_n)$ dentro de una cadena binaria es realizada como se muestra en la figura 4.9.

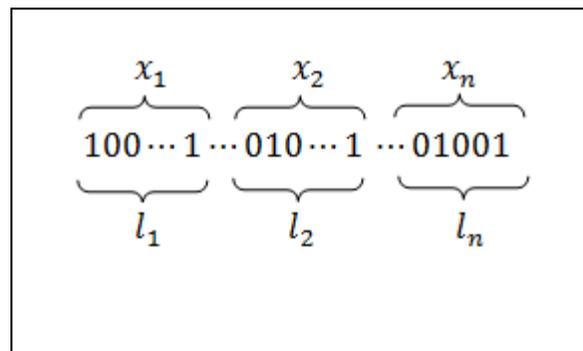


Figura 4.9 Codificación binaria de un cromosoma de n variables

La decodificación de binario a decimal, para cada variable x_i , $i = 1, \dots, n$, con largo de cadena l_i debe de hacerse de la forma

$$x_i = a + \frac{b - a}{2^{l_i} - 1} \left(\sum_{j=0}^{l_i-1} a_{i(l_i-j)} 2^j \right) \quad (4.65)$$

en donde el término $l_i - j$ representa un barrido hacia la izquierda, es decir del bit menos significativo (LSB) hacia el más significativo (MSB).

4.2.5.2.2 Codificación Gray

Dado que la codificación de un algoritmo genético debe de ser tan parecida al espacio de búsqueda como sea posible, entonces un cambio pequeño en el fenotipo debería de indicar un cambio igualmente pequeño en el genotipo. Para una codificación binaria no sucede esto. Dado un genotipo con $n = 6$ y un fenotipo r con $0 \leq r \leq 63$ entonces el valor $011111 = 31$ tendrá que cambiar todos sus seis bits para incrementar el valor del fenotipo por uno: $100000 = 32$.

La codificación Gray mitiga esta situación al asegurar que cualquier par de puntos adyacentes en el espacio del problema difiera por sólo un bit en el espacio de representación [Doran, 2007]. Para muchos problemas esta propiedad de adyacencia es útil para mejorar el desempeño y parecería lo más sensato adaptar el uso de una codificación Gray para la mayoría de los problemas con variables enteras, reales o complejas. En la tabla 4.1 se muestra una lista de equivalencias entre binario y Gray para $l = 4$.

Binario	Gray	Binario	Gray
0000	0000	1000	1100
0001	0001	1001	1101
0010	0011	1010	1111
0011	0010	1011	1110
0100	0110	1100	1010
0101	0111	1101	1011
0110	0101	1110	1001
0111	0100	1111	1000

Tabla 4.1 Comparación entre códigos binario y Gray

Para realizar una conversión de código binario a Gray, deben de usarse operadores OR exclusivos (XOR), una forma de ver esta conversión sería como sigue [Doran, 2007]:

$$G_i = B_{i+1} \oplus B_i, \quad i = l - 1, \dots, 0 \quad (4.66)$$

en donde B_l se toma como 0.

De manera análoga, para transformar una codificación Gray a una binaria es necesario aplicar

$$B_i = B_{i+1} \oplus G_i, \quad i = l - 1, \dots, 0 \quad (4.67)$$

La codificación Gray ofrece una secuencia que no es muy apropiada para manejo humano pero que tiene el potencial de ser más natural para una máquina.

4.2.5.2.3 Codificación IEEE

La codificación más común de la IEEE considera un tamaño por variable de 64 bits. El primer bit denota el signo del valor, los siguientes 11 bits representan la característica o valor entero y, finalmente, los 52 bits restantes son la mantisa o parte decimal de la variable. Esta representación se ilustra en la figura 4.10.

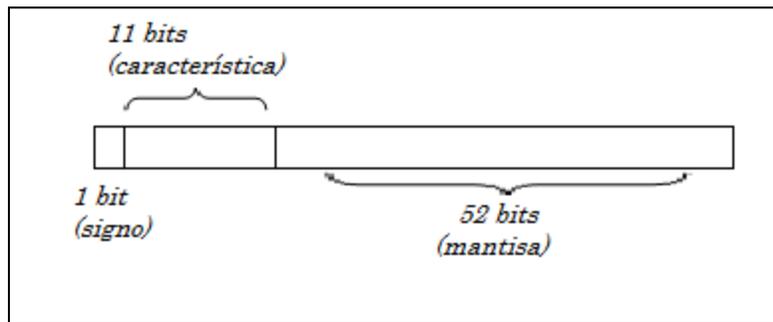


Figura 4.10 Representación de la codificación IEEE

Un valor x escrito en este tipo de codificación estará dado por

$$x = (-1)^s 2^{c-1023} (1 - f) \quad (4.68)$$

en donde s representa el signo, c es el valor en decimal de la característica y f la mantisa, que se calcula de la siguiente forma

$$f = \sum_{i=1}^m f_m \left(\frac{1}{2}\right)^m \quad (4.69)$$

Debe destacarse que la distribución de los bits puede estar sujeta a consideración del usuario para los valores que quieran representarse y que tampoco es necesario usarlos todos.

4.2.5.2.4 Codificación continua

Aunque la codificación binaria (y sus derivadas) son útiles para resolver muchos problemas de optimización, tiene la limitante de su cuantización. Cuando se requiere resolver un problema en donde los valores de las variables son continuos y se quiere saber con toda la precisión de la máquina es necesario implementar este tipo de codificación. Si se utilizara una codificación

binaria, dado que cada variable requiere varios bits para ser representada, al crecer el número de variables, también aumenta el tamaño del cromosoma.

Cuando las variables se encuentran cuantizadas de manera discreta, el algoritmo genético binario funciona de manera muy eficiente. Por otro lado, cuando las variables son continuas, es más lógico representarlas por números de punto flotante. Además, como el algoritmo binario tiene precisión limitada por la representación binaria de las variables, usar valores de punto flotante en lugar de binarias tiene la ventaja de requerir menor espacio de almacenamiento, pues un solo valor flotante representa la variable en lugar de N_{bits} enteros.

La codificación continua es inherentemente más rápida que la binaria, pues los cromosomas no tienen que ser decodificados antes de evaluar la función de costo.

4.2.5.3 Operador de cruza

La cruza se refiere al proceso de intercambiar información entre dos soluciones en una manera similar a como ocurre en la naturaleza dada la reproducción sexual. En el nivel más básico de este operador, se recurre primero a un operador de selección, los pares de individuos elegidos se someten a cruzarse con una probabilidad p_c . Se genera un número aleatorio R_c en el rango 0-1, y se da lugar a la cruza sólo si $R_c \leq p_c$, de otro modo los individuos no se modifican. Los valores típicos de p_c se encuentran en el rango de 0.4 a 0.9.

Sin la cruza, la aptitud promedio de la población, f_{prom} , aumentará o disminuirá (dependiendo de si se maximiza o minimiza) hasta que esta sea igual a la aptitud del miembro más apto, f_{opt} . Después de este punto sólo puede mejorarse a través de mutación.

Uno de los métodos consiste en elegir de manera aleatoria un punto dentro de las cadenas e intercambiar toda la información a la derecha de este punto entre los dos individuos. Este método es el utilizado más comúnmente en algoritmos genéticos sencillos o con longitudes de cromosoma cortas, el principio básico de funcionamiento se ilustra en la figura 4.11.

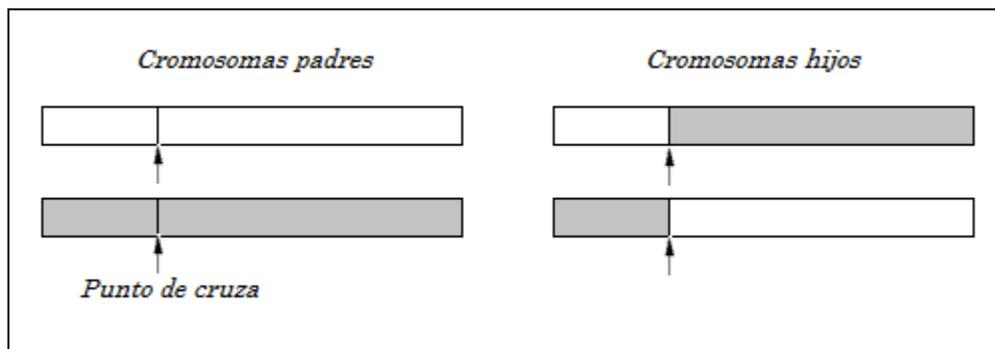


Figura 4.11 Operación de cruza de un punto

Esta operación puede expandirse a casos con múltiples puntos de cruza, la figura 4.12 ilustra el caso en que se lleva a cabo la cruza de dos puntos.

Después de que se lleva a cabo la cruce, se reemplazan los padres en la piscina de apareamiento por sus descendientes. La población ha sido entonces modificada pero aún mantiene el mismo número de elementos.

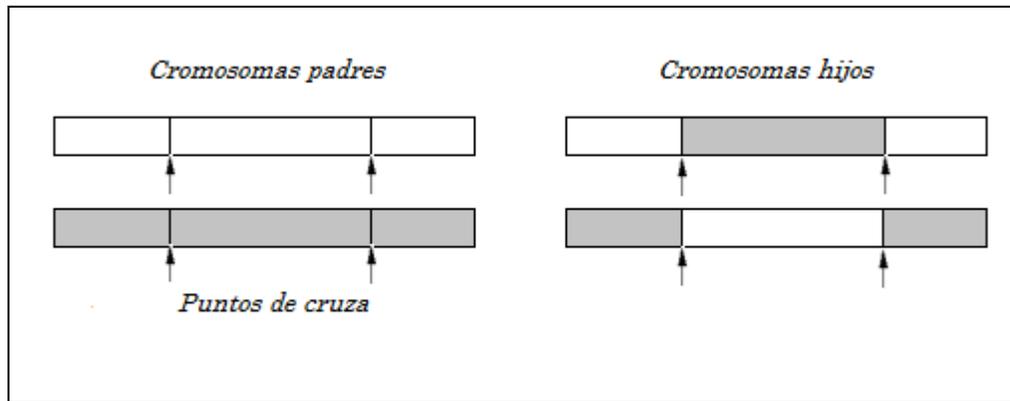


Figura 4.12 Operación de cruce de dos puntos

A continuación debe de aplicarse el parámetro de mutación, sin embargo, en la siguiente parte se explicará antes un poco más acerca del operador de selección y se describirán algunos métodos utilizados por este parámetro.

4.2.5.4 Operador de selección

Como se mencionó en la sección anterior, antes de poder implementar el operador de cruce, es necesario seleccionar a los padres que darán lugar a la nueva población. Es aquí en donde entran los distintos operadores de selección.

La selección intenta aplicar presión sobre la población en una manera similar a la selección natural encontrada en los sistemas biológicos [Baker, 1987]. Los individuos con peor desempeño son sacados de la población y los mejores, o más aptos, tienen una oportunidad mayor que el promedio de proporcionar la información que contienen a la siguiente generación

El método más simple de selección consiste en elegir a los individuos de entre el mejor 50% y el resto se ignora. Este método es práctico pero no es el más común. Una razón para esto es que aunque permite reproducirse a los mejores individuos y detener a los peores, no hace distinción entre 'bueno' y 'muy bueno'. También, en lugar de sólo permitir avanzar soluciones pobres a la siguiente generación con una menor probabilidad, simplemente las aniquila [Gorger-Schleuter, 1989]. Por esta razón se necesitan métodos que hagan mejores distinciones y que permitan que algunos elementos 'no tan buenos' tengan alguna oportunidad de proporcionar información a la siguiente generación. Algunos de estos métodos se describen a continuación.

4.2.5.4.1 Métodos que no consideran aptitud

Un método muy sencillo consiste en comenzar por la parte superior de la lista y aparear los cromosomas en un sistema de dos a la vez hasta que se alcanza el número de cromosomas a seleccionar para reproducción N_k . Los cromosomas madre tienen números de fila en la matriz de población dados por $ma = 1,3,5, \dots$ y el padre tiene los números $pa = 2,4,6, \dots$. Esta técnica no modela bien el modelo natural pero es muy simple programarla e implementarla.

Una segunda técnica consiste en utilizar un generador de números aleatorios uniforme para seleccionar los cromosomas. Los números de fila de los padres se encuentran usando

$$\begin{aligned} ma &= \text{ceil}\{N_k * \text{rand}(1, N_k)\} \\ pa &= \text{ceil}\{N_k * \text{rand}(1, N_k)\} \end{aligned} \quad (4.70)$$

en donde el operador *ceil* redondea el valor al siguiente entero superior.

4.2.5.4.2 Método de la ruleta

Con este enfoque la probabilidad de selección de cada individuo es proporcional a su aptitud. La analogía con una ruleta se da porque puede imaginarse toda la población formando una rueda de ruleta con el tamaño de ranura de cada cromosoma o individuo proporcional a su costo. La ruleta se gira y la 'bola' ficticia se lanza dentro. La probabilidad de que la bola termine en alguna ranura en particular es proporcional al arco de la ranura a , por lo tanto, a la aptitud correspondiente al individuo.

Para implementar este operador, la ruleta está representada por la suma de las aptitudes de la población y la bola es número aleatorio entre 0 y dicha suma. Para determinar en qué ranura cayó la bola, la aptitud de los miembros de la población se suman en orden hasta que la suma exceda el número aleatorio.

En ocasiones, suele implementarse el parámetro de selección como una probabilidad. La probabilidad de que un cromosoma k sea elegido para aparearse está dada, de acuerdo a su aptitud $f^{(k)}$ y el tamaño poblacional N_{pob} por

$$p_{sel}^{(k)} = \frac{f^{(k)}}{\sum_{i=1}^{N_{pob}} f_i} \quad (4.71)$$

En este caso el número aleatorio será generado en el rango de 0 a 1, dado que la suma de todas las probabilidades debe de ser igual a 1.

4.2.5.4.3 Escalamiento lineal

El escalamiento lineal ajusta los valores de aptitud de todos los cromosomas de tal manera que el mejor individuo obtiene un número específico de descendientes esperados. Los otros valores son alterados para asegurar que el número total correcto de nuevas cromosomas es producido.

De esta manera, se previene la reproducción prematura de los individuos con desempeño muy sobresaliente.

El factor de escalamiento s especifica el número de descendientes esperados para el mejor individuo y es típicamente en el rango de 1.2 a 2, dando control sobre la presión de selección. El valor esperado de descendencia para una cromosoma k está dado por

$$E(f^{(k)}) = 1 + \frac{(s - 1)(f^{(k)} - f_{prom})}{(f_{mejor} - f_{prom})} \quad (4.72)$$

Puede notarse que $E(f_{mejor}) = s$ y $E(f_{prom}) = 1$. Aún existe un problema para individuos con bajo desempeño, que tendrán un número negativo de prole. Puede resolverse asignando cero a estos cromosomas, pero esto requeriría que todos los otros valores de aptitud se cambiarán otra vez para mantener el promedio correcto. También se arriesga a perder diversidad. Una alternativa es reducir el factor de escalamiento de tal manera que el peor individuo tenga un puntaje de cero

$$s = 1 + \frac{(f_{mejor} - f_{prom})}{(f_{prom} - f_{peor})} \quad (4.73)$$

Cabe hacer notar que al aumentar el factor de escalamiento s , también crece el radio de convergencia. Con presiones de selección bajas, la razón de convergencia es proporcional a s .

4.2.5.4.4 Clasificación (ranking)

La selección por clasificación fue sugerida en un intento por superar las deficiencias de los enfoques basados directamente en la aptitud [Barker, 1985]. Primero, debe de ordenarse la población de acuerdo a sus valores de aptitud, entonces se asigna un nuevo valor, relacionado de manera inversa a la clasificación o rango del individuo.

El primer método consiste en una clasificación lineal. Al miembro más apto se le asigna una aptitud s , entre 1 y 2. El peor cromosoma recibe $2 - s$. Los valores para las aptitudes de los individuos intermedios k se obtienen por interpolación, una vez que se encuentran ordenados:

$$f^{(k)} = s - \frac{2 \cdot k(s - 1)}{(N_{pop} - 1)} \quad (4.74)$$

en donde N_{pop} representa el tamaño de la población.

Si se hace $s = 2$, el peor individuo no tiene oportunidad de reproducirse. En principio, s puede incrementarse más allá de 2 para obtener una mayor presión de selección, pero entonces algunos de los peores cromosomas obtendrían un número negativo de descendientes. Un método más eficiente de alcanzar mayores presiones de selección, que también ofrece oportunidad a los peores miembros de la población, es utilizar una clasificación no lineal.

La clasificación no lineal más utilizada es la exponencial. En este esquema, el mejor individuo recibe una aptitud de 1 y el segundo una aptitud s , típicamente 0.99. El tercer individuo recibe una aptitud de s^2 y así sucesivamente hasta el último cromosoma, que recibe s^{N-1} . Los nuevos valores de aptitud necesitan dividirse por su promedio para obtener el número esperado de descendientes para cada miembro.

4.2.5.4.5 Método de torneo

Este enfoque emula de manera muy parecida la competición por apareo en la naturaleza. Su funcionamiento consiste en elegir de manera aleatoria un pequeño subgrupo de cromosomas, normalmente 2 ó 3, de la piscina de apareamiento, y el cromosoma con el mejor desempeño se convierte en padre para la siguiente generación. El torneo se repite hasta que se alcance la cantidad de padres necesarios para crear la siguiente generación. Este método funciona mejor para tamaños de población grandes pues no debe de ordenarse la población de acuerdo a su función de costo.

Un cromosoma con desempeño igual al promedio tendrá expectativas de ganar la mitad de sus torneos y de ser elegido la mitad de las veces que el mejor cromosoma. Algunos modelos como el ASPARAGOS [Gorges-Schleuter, 1989] utilizan una población distribuida espacialmente con torneos sólo entre los miembros cercanos. Aunque al utilizar la selección por torneo debe de considerarse el error de muestreo implicado.

4.2.5.5 Operador de mutación

La mutación se refiere, en general, a modificar un valor de la cadena que representa a un cromosoma con una probabilidad p_m . La forma más común de aplicar este operador consiste en pasar un alelo o variable del cromosoma y decidir si se modifica basándose en un valor aleatorio. Al igual que en la cruce, para decidir si se lleva a cabo la mutación, debe de generarse primero un valor aleatorio R_m en el intervalo 0-1, para cada alelo se aplica el operador sólo si $R_m \leq p_m$. En la mayoría de los casos p_m oscila entre 0.01 y 0.2 [Fogarty, 1989].

El operador de muta tiene la particularidad de que introduce nueva información, en otras palabras, mantiene la diversidad de la población. La cruce no puede generar más información que aquella con que se ha iniciado el algoritmo.

En el caso de una codificación binaria (estándar, Grey o IEEE), la mutación consiste en cambiar el valor del alelo de 1 a 0 y viceversa. Sin embargo, debe de tomarse en cuenta que la posición que se altera puede modificar de manera significativa, por ejemplo, en una cadena de 10 bits con codificación binaria, modificar el bit menos significativo alterará este valor por ± 1 , mientras que un cambio en el más significativo introducirá una variación de ± 512 .

Una manera de lidiar con esta situación es confinar la mutación a los bits de menor orden, aunque es pierde gran parte de la exploración del espacio solución, pues estos bits ayudan a diversificar en generaciones tempranas.

Otras posibilidades son la de sesgar la mutación a los individuos menos aptos para incrementar la exploración sin degradar a los cromosomas con mejor desempeño [Davis, 1989], o hacer que p_m sea una función del número de generaciones con su probabilidad disminuyendo mientras se avanza [Fogarty, 1989].

Para una codificación continua, la forma más sencilla de mutar es la de sumar o restar una cantidad aleatoria con distribución normal

$$p'_n = p_n + \sigma N_n(0,1) \quad (4.75)$$

en donde σ representa la desviación estándar de la distribución normal y $N_n(0,1)$ representa la distribución normal estándar con media 0 y varianza 1. Esta técnica es difícil de usar, pues debe de elegirse un buen valor de σ , la adición de una variable puede ocasionar que ésta se salga de sus límites, lo que lleva un mayor tiempo computacional.

Otro enfoque para esta situación es el de elegir el número de elementos a mutar a partir de p_m con $n_{mut} = N_{pop} \cdot p_m$ y elegir de manera aleatoria n_{mut} elementos de la población total (denotados por su índice de fila y columna) para cambiar por valores aleatorios dentro del rango de búsqueda.

Con esta sección se concluye el capítulo 4, en donde se revisaron distintas técnicas de computacionales para la optimización de problemas. Puede notarse la gran diferencia entre los métodos analíticos que requieren un conocimiento casi acertado del problema a optimizar y las técnicas que imitan procesos naturales que son más bien aleatorias y probabilísticas. El siguiente capítulo hace uso de la teoría revisada en los capítulos anteriores para solucionar el problema en cuestión: el registro de superficies contando con poco conocimiento de la relación entre estas.

Referencias

- [1] Baker, J.E., '*Reducing bias and inefficiency in the selection algorithm*', Proceedings of the 2nd International Conference of Genetic Algorithms, Grefenstette, J.J., Ed. Lawrence Earlbaum, 1987.
- [2] Bonabeau, E., Dorigo, M., Theraulaz, G., '*Swarm intelligence from natural to artificial systems*', New York: Oxford University Press, 1999.
- [3] Chong, E.K.P., Zak, S.H., '*An introduction to optimization*', Wiley-Interscience, 2nd Ed., 2001.
- [4] Conte, S.D., de Boor, C., '*Elementary numerical analysis: an algorithmic approach*', New York: McGraw-Hill Book Co., 3rd ed., 1980.
- [5] Cuevas, F.J., '*Aplicación de redes neuronales en la reconstrucción tridimensional de objetos*', Tesis doctoral, Centro de Investigaciones en Óptica, A.C., León, México (2000).
- [6] Davis, L., '*Genetic Algorithms and Simulated Annealing*', Research notes in Artificial Intelligence, (London), Pitman, 1987.
- [7] Davis, L., '*Adapting operator probabilities in genetic algorithms*', in Schaffer, J.D., Proceedings of the 3rd International Conference of Genetic Algorithms, Morgan Kaufmann, pp. 61 – 69, 1989.
- [8] Doran, R.W., '*The Gray code*', J. Univ. Comp. Science **13**: 1573 – 1597 (2007).
- [9] Dorigo, M., Maria, G., '*Ant colony system: a cooperative learning approach to the traveling salesman problem*', IEEE Trans. Evol. Comput. **1**: 53- 66 (1997).
- [10] Fletcher R., '*Practical methods of optimization*', Wiley-Interscience, 2nd Ed., 1987.
- [11] Fogarty, T.C., '*Varying the probability of mutation in genetic algorithm*', in Schaffer, J.D., Proceedings of the 3rd International Conference of Genetic Algorithms, Morgan Kaufmann, pp. 104 – 109, 1989.
- [12] Goldberg, D.E., '*Adaptive control of gas pipeline systems*', Unpublished manuscript,(seen on [Goldberg, 1989]), 1980.
- [13] Goldberg, D.E., '*Genetic Algorithms in Search, Optimization & Machine Learning*', Addison-Wesley, 1989.
- [14] Gorger-Schleuter, '*M. ASPARAGOS: An analysis of selection schemes used in Genetic Algorithms*', Foundations of Genetic Algorithms, Rawlings, G.J.E., Ed. Morgan Kaufmann, 1989.
- [15] Haupt, R.L., Haupt, S.E., '*Practical genetic algorithms*', Wiley-Interscience, 2nd Ed., 2004.
- [16] Hertz, J., Krogh, A., Palmer R.G., '*Introduction to the Theory of Neural Computation, vol. 1*', Santa Fe Institute Studies in the Sciences of Complexity, Redwood City, CA: Addison-Wesley, 1991.
- [17] Holland, J.H., '*Genetic Algorithms*', Sci. Am. **267**: 66-72 (1992a).

- [18] Holland, J.H., *'Adaptation in Natural and Artificial Systems: An Introductory Analysis with Application to Biology, Control, and Artificial Intelligence'*, Cambridge, MA: MIT Press, 1992b.
- [19] Hush, D.R., Horne, B.G., *'Progress in supervised neural networks: What's new since Lippmann'*, IEEE Signal Processing Magazine, pp. 8 – 39, 1993.
- [20] Jang, J.S.R., Sun, C.T., Mizutani, E., *'Neuro-Fuzzy and Soft Computing: A computational approach to learning and machine intelligence'*, Upper Saddle River, NJ: Prentice-Hall, 1997.
- [21] Kelly, M.F., Parker, P.A., Scott, R.N., *'The application of neural networks to myoelectric signal analysis: A preliminary study'*, IEEE Trans. Biomedical Engineering, **37**: 221 – 230 (1990).
- [22] Kennedy, J., Eberhart, R.C., *'Particle swarm optimization'*, Proc. IEEE Int. Conf. on Neural Networks, IV. Piscataway, NJ: IEEE Service Center, pp. 1942 – 1948 (1995).
- [23] Kirkpatrick, X., Gelatt Jr., C.D., Vecchi, M.P., *'Optimization by simulated annealing'*, Science **220**: 671 – 680 (1983).
- [24] Koza, J.R., *'Genetic Programming: On the programming of Computers by Means of Natural Selection'*, Cambridge, MA: MIT Press, 1992.
- [25] Kozek, T., Roska, T., Chua, L.O., *'Genetic algorithm for CNN template learning'*, IEEE Trans. Circuit and Syst.-I: Fund. Theory and Appl., **40**: 392 – 402 (1993).
- [26] Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., *'Equation of state calculations by fast computing machines'*, J. Chem. Phys. **21**: 1087 – 1092 (1953).
- [27] Mitchell, M., *'An Introduction to Genetic Algorithms'*, Cambridge, MA: MIT Press, 1996.
- [28] Rangwala, S.S., Dornfeld, D.A., *'Learning and optimization of machining operations using computing abilities of neural networks'*, IEEE Trans. Sys., Man and Cybernetics, **19**: 299 – 314 (1989).
- [29] Rumelhart, D.E., McClelland, J.L., Group, P.R., *'Parallel Distributed Processing, Explorations in the Microstructure of Cognition, Volume 1: Foundations'*, Cambridge MA: MIT Press, 1986.
- [30] Then, T.W., Chong, E.K.P., *'Genetic algorithms in noisy environments'*, Proc. 9th IEEE Symp. on Intelligent Control, pp. 225 - 230, 1994.
- [31] Widrow, B., Lehr, M.A., *'30 years of adaptive neural networks: Perceptron, madaline, and backpropagation'*, Proceedings of the IEEE, **78**: 1415 – 1442 (1990).

Capítulo 5

Registro de superficies utilizando algoritmos evolutivos

Este capítulo lidia con el punto central de esta tesis: el registro o alineamiento de superficies. Una vez que se ha adquirido la forma de un objeto a partir de las técnicas mencionadas en el capítulo 2, es necesario hacer dos o más tomas para tener la reconstrucción completa del objeto. En el caso en que el arreglo óptico se encuentre limitado a una mesa giratoria y que sus componentes se encuentren fijos, es fácil llevar a cabo una reconstrucción a partir de los parámetros radiales del arreglo [Ze-Su *et al.*, 2007]. Sin embargo, este tipo de disposiciones se encuentran limitadas, por lo general, a un ambiente de laboratorio y no suelen utilizables en un ambiente industrial.

El problema de registro ha tratado de resolverse desde diferentes perspectivas: basándose en las propiedades cinemáticas y la geometría [Pottman *et al.*, 2004], buscando los parámetros de transformación geométrica basándose en propiedades estadísticas de los vértices [Xiao *et al.*, 2005], la implementación de dos fases de algoritmos genéticos [Percoco y Galantucci, 2002] y utilizando un algoritmo de iteración del punto más cercano (ICP, por sus siglas en inglés) basado en la correlación de características locales [Chavarria y Sommer, 2008]. Aún así, estos algoritmos fallan en un punto vital para el trabajo de registro desde varias vistas: sólo sirven para grandes áreas de traslape entre superficies. Un análisis comparativo entre algunos métodos previos se ofrece en [Matabosch *et al.*, 2004].

En este trabajo de tesis se propone llevar a cabo un registro de superficies sin necesidad de contar con una parametrización de las posiciones relativas de cada adquisición de superficies ni de utilizar marcas invasivas sobre el objeto bajo análisis. Para llevar a cabo esta tarea se plantea el uso de un algoritmo genético en combinación con el algoritmo de Iteración del Punto más Cercano, o ICP por sus siglas en inglés. El primero se usa para realizar un pre-alineamiento de las superficies, suficiente para que el segundo pueda resolver de manera eficaz el problema. A continuación se presenta el algoritmo ICP.

5.1 Algoritmo de Iteración del Punto más Cercano

El algoritmo ICP fue implementado por primera vez en 1992 [Besl y McKay, 1992; Zhang, 1992]. El ICP se basa en una optimización iterativa no lineal que permite la alineación de un conjunto de datos para un registro fino. Varias mejoras se han agregado al algoritmo, como la adición de elementos al algoritmo básico con el fin de procesar los datos atípicos para una mejor

comparación de los sub-conjuntos [Zhang, 1994]. Otro enfoque fue propuesto por [Chen y Medioni, 1992] en donde no se minimiza la distancia entre los puntos más cercanos, sino que se utiliza información de su orientación. La descripción del algoritmo general se hace a continuación.

Considérense dos grupos de puntos $\{x_i\}$ y $\{p_i\}$ en donde no se conoce nada de su emparejamiento y se quiere computar la transformación rígida entre ellos. Si se obtienen los parámetros de transformación, entonces se obtiene una solución de forma cerrada al minimizar la siguiente función objetivo

$$f(R, t) = \frac{1}{N} \sum_{i=1}^N \|p_{x,i} - R(p_{p,i}) + T\|^2 \quad (5.1)$$

en donde N es la cantidad de puntos a emparejar, $R(\cdot)$ y T representan los parámetros de transformación geométrica de rotación y de traslación, respectivamente. Algunos métodos para minimizar la ecuación 5.1 pueden encontrarse en [Lorusso *et al.*, 1995].

Asúmase que la transformación es pequeña, casi una identidad. Esto significa que la distancia entre el punto en el sistema coordenado X y su localización transformada P es pequeña. Dada esta suposición, un probable emparejamiento para el punto p_p es el punto más cercano en $\{x_i\}$:

$$d(p_p, \{x_i\}) = \min_{p_x \in \{x_i\}} \|p_p - p_x\| \quad (5.2)$$

Desafortunadamente, no existe garantía de que la transformación es casi una identidad, así que tomar el punto más cercano en $\{x_i\}$ conducirá por lo normal a una solución errónea. Una propuesta común en esta situación es usar iteraciones esperando converger a la solución correcta, esto es el algoritmo ICP y se ilustra en la Tabla 5.1 [Yaniv, 2001].

<u>Algoritmo de Iteración del Punto más Cercano</u>
1. Crear una correspondencia entre las nubes de puntos, los puntos más cercanos se emparejan.
2. Computar el registro rígido dado el emparejamiento.
3. Aplicar la transformación a los datos y computar la distancia media entre las nubes.
4. Si el cambio en la distancia media no es menor que un umbral $d\mu$ o se alcanzado el número máximo de iteraciones, terminar el algoritmo.

Tabla 5.1 Algoritmo ICP

El mayor costo computacional al implementar un algoritmo ICP es el emparejamiento de los puntos, pues debe de implementarse la ecuación 5.2 para cada punto de la nube $\{R_i\}$ sobre todos los puntos de la nube $\{L_i\}$. En la mayoría de los casos, prefiere implementarse el uso de árboles k -

dimensionales, que proveen una representación más sencilla para la búsqueda de puntos y, por tanto, más eficiente [Moore, 1991]. Después de que se encuentra la correspondencia entre las nubes, debe de encontrarse el registro rígido, para llevar a cabo este paso, una de las técnicas comunes el uso de cuaterniones para determinar las matrices de transformación R y T [Besl y McKay, 1992; Surmann *et al.*, 2003]. Sin embargo, un método más robusto y fácil de implementar es el del uso de la descomposición de valores singulares (SVD) [Arun *et al.*, 1987]. La implementación de la creación de árboles *k-dimensional* y SVD suele acelerar el tiempo de ejecución del algoritmo ICP, pero no asegura la convergencia al óptimo global, sin embargo, se explican en las secciones siguientes, pues es la manera en que se implementan en este trabajo.

5.1.1 Árboles *k-dimensional*es

Un árbol *k-dimensional* (abreviado como árbol *kd*) es una estructura de datos cuyo fin es almacenar un número finito de puntos de un espacio *k-dimensional* y fue analizado por primera vez por Friedman y ampliado por Bentley [Friedman *et al.*, 1977; Bentley, 1980].

Si un espacio euclídeo contiene N puntos, el árbol *kd* correspondiente tendrá el mismo número de nodos, es decir, se tiene un espacio $O(N)$ y k indicará la dimensionalidad del espacio. Cada nodo consistirá de dos apuntadores hijos, información del nodo y una clave, ya sea un solo valor flotante que representa una coordenada o un par de flotantes representando una dimensión o un rectángulo. Un esquema de reconstrucción de un árbol *kd* se muestra en la figura 5.1.

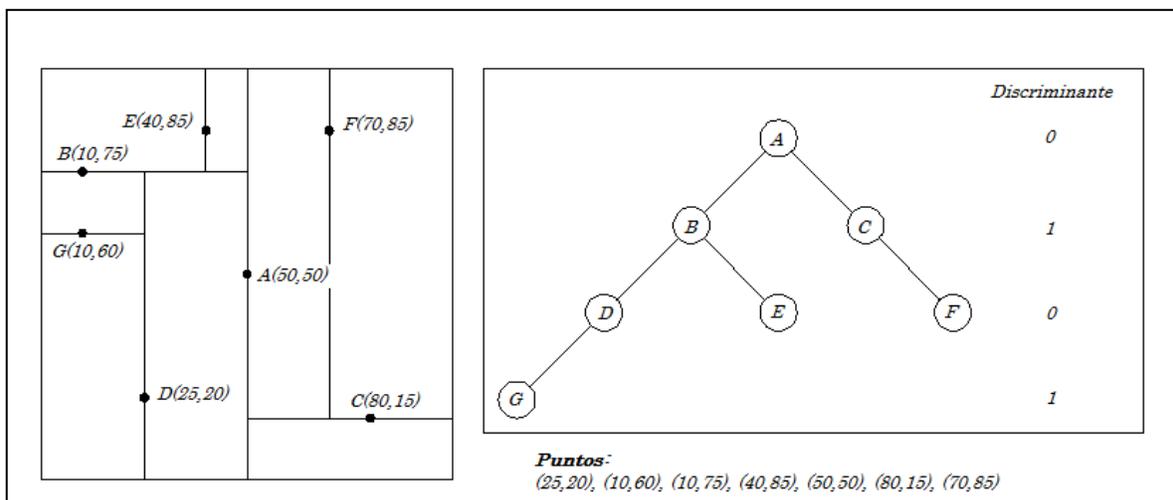


Figura 5.1 Esquema de construcción de un árbol *kd* bidimensional

Para construir un árbol binario, primero debe elegirse un punto coordenada para dividir el resto de los puntos. Por ejemplo, si se elige el eje x como base para dividir el resto de los puntos, todos los puntos menores que el valor raíz de x seleccionado se encontrarán a la izquierda de la raíz y lo contrario para aquellos que sean mayores o iguales. A continuación se elige un valor y como discriminante para los hijos de la raíz y se considera un nuevo valor x para los nietos de la raíz. Suele nombrarse a los discriminantes de acuerdo al número de dimensión considerado, por

ejemplo el discriminante x será 0, para y es 1 y así sucesivamente para árboles de mayores dimensiones.

La figura 5.1 representa un árbol con un pivote o raíz en $x = 50$, como los valores extremos son $x = 10$ y $x = 80$, se eligen los puntos $(10,75)$ y $(80,15)$ como hijos de la raíz. Nótese que se eligió $(10,75)$ y no $(10,60)$, pues el valor de y del punto seleccionado es mayor; una alternativa a este método es el de crear un árbol balanceado, en que los puntos correspondientes a los nodos se proporcionan de manera aleatoria. A continuación debe de tomarse el siguiente discriminante, es decir y (denotado por 1). En este caso, para el nodo marcado como C sólo existe un posible hijo F en donde $y_F > y_C \otimes x_C > x_A$. El nodo D queda a la izquierda del nodo B dado que $y_D < y_B$ y se hace lo propio para el nodo E . Este proceso debe de repetirse hasta tener clasificados los N puntos que conforman el espacio a analizar, intercambiando el discriminante entre cada nueva generación de hijos obtenida.

Para realizar una búsqueda del vecino más cercano se genera un árbol para sólo uno de los espacios, mientras que en el otro espacio debe de buscarse el punto más cercano uno a uno. Sin embargo, dado que el espacio se encuentra particionado por discriminantes y que los puntos emparejados son removidos, esta búsqueda tomará menor tiempo que si se busca en el espacio entero de la lista en representación de árbol.

5.1.2 Descomposición de Valores Singulares

La descomposición de valores singulares (Singular Value Decomposition, SVD) ha sido anteriormente aplicada para establecer correspondencias entre conjuntos 3D [Arun *et al.*, 1987; Rieger, 2005]. Concretamente, se obtiene un sistema de referencia común mediante el cálculo de la transformación existente entre el conjunto de datos [Ballesta *et al.*, 2007]. La descomposición SVD es una generalización de la eigen-descomposición que puede usarse para analizar matrices rectangulares (la eigen-descomposición sólo está definida para matrices cuadradas). Por analogía con la eigen-descomposición, que descompone una matriz cualquiera en dos matrices simples, la idea general de SVD es descomponer una matriz rectangular en tres matrices simples: dos ortogonales y una diagonal [Abdi, 2007].

Una matriz positiva semi-definida puede obtenerse como el producto de una matriz por su traspuesta. Esta matriz es cuadrada y simétrica, pero además sus eigenvalores son todos positivos o nulos, y los eigenvectores correspondientes a diferentes eigenvalores son ortogonales en pares. Si \mathbf{X} es una matriz positiva semi-definida, su eigen-descomposición se expresa como

$$\mathbf{X} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T \quad (5.3)$$

en donde \mathbf{U} es una matriz ortonormal ($\mathbf{U}^T\mathbf{U} = \mathbf{I}$) y $\mathbf{\Lambda}$ es una matriz diagonal que contiene los eigenvalores de \mathbf{X} .

La descomposición SVD usa la eigen-descomposición de una matriz positiva semi-definida para derivar una descomposición similar aplicable a todas las matrices rectangulares compuestas por números reales. Entonces si \mathbf{A} es una matriz rectangular su descomposición SVD es

$$\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^T \quad (5.4)$$

siendo \mathbf{U} los eigenvectores normalizados de la matriz $\mathbf{A}\mathbf{A}^T$, es decir, $\mathbf{U}^T\mathbf{U} = \mathbf{I}$. Las columnas de \mathbf{U} se denominan como vectores singulares izquierdos de \mathbf{A} ; \mathbf{V} representa a los eigenvectores normalizados de la matriz $\mathbf{A}^T\mathbf{A}$ ($\mathbf{V}^T\mathbf{V} = \mathbf{I}$) y sus columnas son los vectores singulares derechos de \mathbf{A} ; finalmente \mathbf{D} es la matriz diagonal de valores singulares, $\mathbf{D} = \mathbf{\Lambda}^{1/2}$ con $\mathbf{\Lambda}$ siendo la matriz diagonal de los eigenvalores de $\mathbf{A}\mathbf{A}^T$ y $\mathbf{A}^T\mathbf{A}$, que de hecho son los mismos [Golub y Van Loan, 1989].

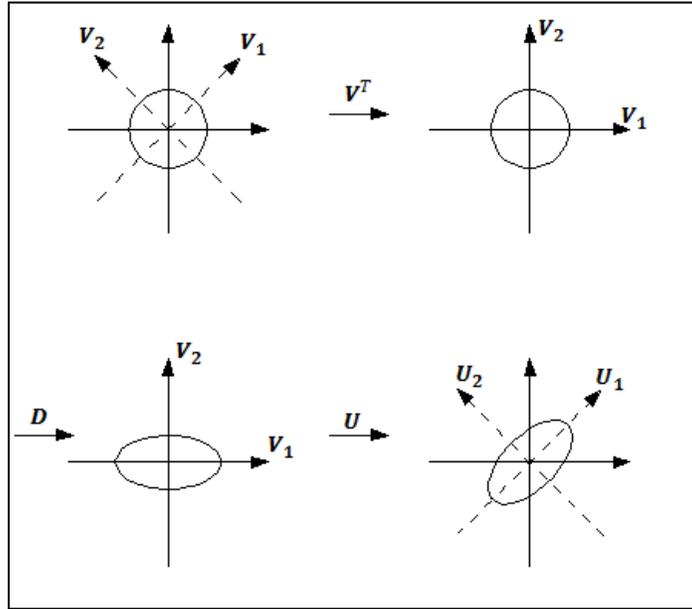


Figura 5.2 Significado geométrico de SVD: la imagen de un círculo tras una multiplicación matricial es una elipse

Para una matriz \mathbf{A} de tamaño $M \times N$ con $M \geq N$, la ecuación 5.4 puede escribirse como

$$\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^T = \sum_{i=1}^N d_i \mathbf{U}_i \mathbf{V}_i^T \quad (5.5)$$

En la representación anterior el i -ésimo elemento distinto de cero de la matriz diagonal \mathbf{D} está representado por d_i , mientras que \mathbf{U}_i y \mathbf{V}_i son la i -ésima columna de las matrices \mathbf{U} y \mathbf{V} , respectivamente. El tamaño de las matrices \mathbf{U} , \mathbf{V} y \mathbf{D} será de $M \times M$, $N \times N$ y $M \times N$. Una interpretación geométrica de SVD para $M = N = 2$ se presenta en figura 5.2, de donde puede observarse que primero se rota \mathbf{V}_i para hacerla coincidir con el i -ésimo eje coordenado, estirado por un factor d_i y rotado de regreso en la dirección de \mathbf{U}_i [Muller *et al.*, 2004].

Una consecuencia directa de esta interpretación es que el valor singular mayor, d_1 , mide la magnitud de \mathbf{A} . El rango de \mathbf{A} es el número de valores singulares distintos de cero. Entonces, si $\text{rango}(\mathbf{A}) = r$, es posible reescribir en su forma reducida

$$\mathbf{A} = \mathbf{U}_+ \mathbf{D}_+ \mathbf{V}_+^T = \sum_{i=1}^r d_i \mathbf{U}_i \mathbf{V}_i^T \quad (5.6)$$

donde \mathbf{D}_+ es la matriz diagonal de tamaño $r \times r$ con los r valores singulares distintos de cero en \mathbf{A} , \mathbf{U}_+ y \mathbf{V}_+ consisten de las primeras r columnas de \mathbf{U} y \mathbf{V} , respectivamente.

El cuaternión unitario para la matriz de rotación a buscar está dado por $\vec{q}_R = [q_0 \ q_1 \ q_2 \ q_3]^T$, en donde $q_0 \geq 0$ y $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$. La matriz de rotación generada por el cuaternión unitario es

$$\mathbf{R} = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 - q_0 q_2) \\ 2(q_1 q_2 - q_0 q_3) & q_0^2 + q_2^2 - q_1^2 - q_3^2 & 2(q_2 q_3 - q_0 q_0) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 - q_0 q_1) & q_0^2 + q_3^2 - q_1^2 - q_2^2 \end{bmatrix} \quad (5.7)$$

Sea ahora $\vec{q}_T = [q_4 \ q_5 \ q_6]^T$ el vector de traslación. El vector completo de registro estará denotado por $\vec{q} = [\vec{q}_R | \vec{q}_T]^T$. Sea $P = \{\vec{p}_i\}$ un conjunto de puntos adquiridos para ser alineados (nube flotante) con un conjunto fijo $X = \{\vec{x}_i\}$, en donde $N_x = N_p$ y en donde a cada punto \vec{p}_i corresponde otro punto \vec{x}_i con el mismo índice. La función objetivo expresada en la ecuación 5.1 puede reescribirse como

$$f(\vec{q}) = \frac{1}{N_p} \sum_{i=1}^{N_p} \|\vec{x}_i - \mathbf{R}(\vec{q}_R) + \vec{q}_T\|^2 \quad (5.8)$$

Los centros de masa de la nube flotante $\vec{\mu}_p$ y de la nube fija $\vec{\mu}_x$ están dados por

$$\vec{\mu}_p = \frac{1}{N_p} \sum_{i=1}^{N_p} \vec{p}_i \quad y \quad \vec{\mu}_x = \frac{1}{N_x} \sum_{i=1}^{N_x} \vec{x}_i \quad (5.9)$$

La matriz de covarianza cruzada Σ_{px} entre los conjuntos P y X será

$$\Sigma_{px} = \frac{1}{N_p} \sum_{i=1}^{N_p} [(\vec{p}_i - \vec{\mu}_p)(\vec{x}_i - \vec{\mu}_x)^T] = \frac{1}{N_p} \sum_{i=1}^{N_p} [\vec{p}_i \vec{x}_i] - \vec{\mu}_p \vec{\mu}_x^T \quad (5.10)$$

Los componentes cíclicos de la matriz anti-simétrica $\mathbf{B}_{ij} = (\Sigma_{px} - \Sigma_{px}^T)_{ij}$ se utilizan para formar el vector columna $\mathbf{\Delta} = [\mathbf{B}_{23} \ \mathbf{B}_{31} \ \mathbf{B}_{12}]^T$. Este vector es usado para formar la matriz simétrica

$$Q(\Sigma_{px}) = \begin{bmatrix} \text{tr}(\Sigma_{px}) & & \\ \mathbf{\Delta} & \mathbf{\Delta}^T & \\ & \Sigma_{px} + \Sigma_{px}^T - \text{tr}(\Sigma_{px})\mathbf{I}_3 & \end{bmatrix} \quad (5.11)$$

en donde \mathbf{I}_3 es la matriz identidad de tamaño 3×3 y $\text{tr}(\cdot)$ representa el operador trazo, es decir, la suma de los elementos en la diagonal principal y está dado por

$$tr(C) = c_{11} + c_{22} + \dots + c_{nn} = \sum_{i=1}^n c_{nn} \quad (5.12)$$

El eigenvector unitario $\vec{q}_R = [q_0 \ q_1 \ q_2 \ q_3]^T$ correspondiente al máximo eigenvalor de la matriz $Q(\Sigma_{px})$ se selecciona como la rotación óptima. El vector de traslación óptimo estará dado entonces por

$$\vec{q}_T = \vec{\mu}_X - \mathbf{R}(\vec{q}_R)\vec{\mu}_P \quad (5.13)$$

Algunos de los inconvenientes con que cuenta el algoritmo ICP son, según [Xiao *et al.*, 2005]: (a) el costo computacional al buscar los puntos más cercanos, aún con árboles k-dimensionales, es alto cuando se tiene un gran número de vértices; (b) no existe una garantía de que se llegará a la solución correcta; (c) se requiere de una buena estimación inicial de los parámetros de transformación para evitar que el proceso de iteración quede atrapado en un mínimo local.

Una posible mejora del enfoque del algoritmo ICP es la implementación de métodos de optimización no lineales, como un algoritmo genético, que tiene la capacidad de salir de mínimos locales. Los algoritmos genéticos han sido usados para aplicaciones relacionadas con el registro de nubes de puntos [Brunnstrom *et al.*, 1996], incluyendo el uso de algoritmos genéticos dinámicos [Chow *et al.*, 2004] con un operador de mutación particular utilizado al final del algoritmo para obtener un ajuste fino de la posición de las nubes.

5.2 Descripción del algoritmo genético utilizado

Como se mencionó al inicio de este capítulo, el planteamiento es utilizar el algoritmo ICP para llevar a cabo el registro o alineamiento de dos nubes de puntos, sin embargo, la sola utilización de este algoritmo tiende a converger a un mínimo local si no existe un buen pre-alineamiento. Con el fin de lograr este primer alineamiento se utiliza un algoritmo genético que incluye un pseudo-ICP, en donde se calculan las distancias mínimas entre puntos vecinos en las superficies, con la diferencia de que no se reemplaza o elimina ningún punto en las nubes una vez que este se ha emparejado, esto con la finalidad de ahorrar tiempo de cómputo gracias a la cualidad de trabajo matricial de MATLAB, pues puede llevarse a cabo la asignación de una operación vectorial en un solo paso y la exclusión de puntos implicaría la programación de un ciclo *for*.

Dado el hecho de que no se calcula la distancia real mínima entre dos puntos adyacentes (dos puntos de una nube pueden tener el mismo vecino más cercano), es indispensable incluir operadores que tomen en cuenta la proporción entre las áreas de traslape o superposición.

Si se definen dos superficies, una denominada nube fija, que será aquella a la que desea alineársele la segunda nube, denotada por el término nube flotante, entonces $X = \{x_i\}$ con $i = 1, \dots, M$ será el conjunto de puntos correspondiente a la nube fija o modelo y $P = \{p_j\}$ con

$j = 1, \dots, N$ los puntos dentro de la nube flotante. La función objetivo a optimizar queda definida entonces como

$$f(\chi) = |d_{OL} - Yd_{OV} - \varepsilon|^2 \quad (5.14)$$

en donde χ representa a un cromosoma conteniendo los parámetros de rotación y de traslación, d_{OL} es la distancia mínima promedio entre puntos que no se encuentran en la superficie traslapada y d_{OV} es la distancia mínima promedio entre los puntos en el área de superposición. El factor Y es un elemento de normalización dependiente de los puntos considerados como alineamiento exitoso dada una cierta tolerancia y ε es un factor de error debido al muestreo. La estimación de estos factores se detalla en las siguientes secciones, dado que algunas de estas necesitan del conocimiento previo de otros parámetros.

Debe de tomarse en cuenta además, que para eje se considera un factor de escalamiento, pues la convergencia del algoritmo se ve acelerada si se utilizan tamaños normalizados en todas las direcciones, ya que no se necesita recalculer nuevos rangos para el parámetro de mutación.

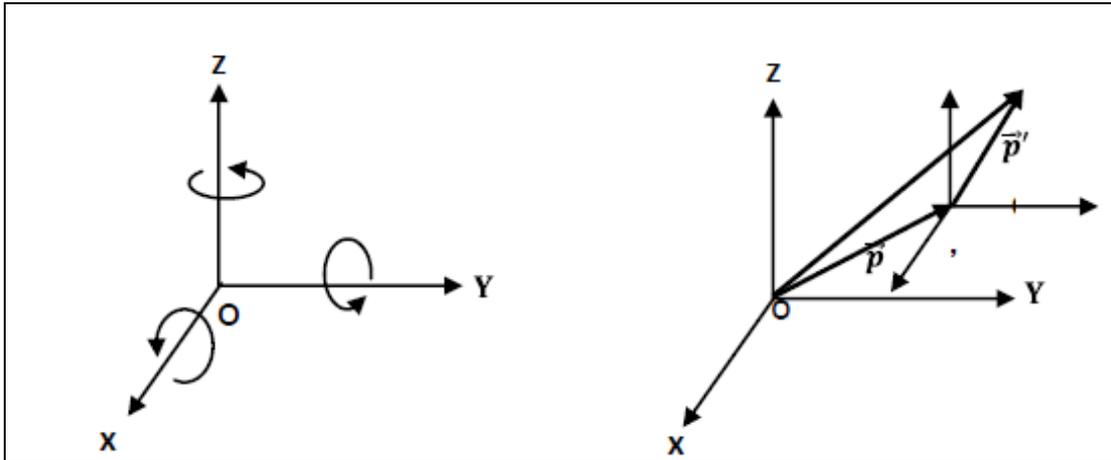


Figura 5.3 Convención adoptada para parámetros de transformación positivos

La matriz de transformación \mathbf{R} está definida como la multiplicación de las tres matrices de rotación parciales para cada eje. Las matrices que definen \mathbf{R} son

$$R_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\text{sen}(\alpha) \\ 0 & \text{sen}(\alpha) & \cos(\alpha) \end{pmatrix}$$

$$R_y(\beta) = \begin{pmatrix} \cos(\beta) & 0 & \text{sen}(\beta) \\ 0 & 1 & 0 \\ -\text{sen}(\beta) & 0 & \cos(\beta) \end{pmatrix}$$

$$R_z(\gamma) = \begin{pmatrix} \cos(\gamma) & -\text{sen}(\gamma) & 0 \\ \text{sen}(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (5.15)$$

Entonces, la matriz \mathbf{R} estará definida como

$$\mathbf{R} = R_y \cdot R_z \cdot R_x = \begin{pmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{pmatrix} \quad (5.16)$$

en donde

$$\begin{aligned} r_1 &= \cos(\beta) \cos(\gamma) \\ r_2 &= \text{sen}(\beta) \text{sen}(\alpha) - \cos(\beta) \text{sen}(\gamma) \cos(\alpha) \\ r_3 &= \cos(\beta) \text{sen}(\gamma) \text{sen}(\alpha) + \text{sen}(\beta) \cos(\alpha) \\ r_4 &= \text{sen}(\gamma) \\ r_5 &= \cos(\gamma) \cos(\alpha) \\ r_6 &= -\cos(\gamma) \text{sen}(\alpha) \\ r_7 &= -\text{sen}(\beta) \cos(\gamma) \\ r_8 &= \text{sen}(\beta) \cos(\gamma) \cos(\alpha) + \cos(\beta) \text{sen}(\alpha) \\ r_9 &= \cos(\beta) \cos(\alpha) - \text{sen}(\gamma) \text{sen}(\beta) \text{sen}(\alpha) \end{aligned} \quad (5.17)$$

Así, se obtiene un sistema sobre determinado con 9 ecuaciones y 3 incógnitas para el cálculo de las rotaciones α , β y γ . Nota: aunque el algoritmo ICP utiliza más bien los valores r_1 - r_9 para su solución, se calculan los ángulos de rotación para una mejor comparación con el algoritmo genético.

El vector de traslación \mathbf{T} se encuentra definido por

$$\mathbf{T} = \begin{pmatrix} T_x \\ T_y \\ T_z \end{pmatrix} \quad (5.18)$$

La convención de parámetros positivos tanto de rotación como de traslación se muestra en la figura 5.3.

Entonces, la transformación p' de un punto p estará dada por

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \mathbf{R} \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \mathbf{T} \quad (5.19)$$

Debe recordarse que la transformación concerniente a \mathbf{R} depende del orden en que se realiza la multiplicación de las matrices 5.15, por simplicidad se considera que $\mathbf{R} = R_y R_z R_x$.

El algoritmo genético busca los parámetros de transformación descritos en las ecuaciones 5.16 y 5.18; la codificación adoptada es la siguiente: $\chi = [\alpha, \beta, \gamma, T_x, T_y, T_z]$. Esta codificación tiene lugar, basándose en la teoría de esquemas descrita en el capítulo anterior, debido a que los esquemas más importantes son los correspondientes a los ángulos de rotación y por tanto, es esencial mantener una longitud corta entre estos para no afectarlos de manera significativa dado el operador de cruza. La codificación considera que los primeros tres parámetros se encuentran expresados en grados y los parámetros correspondientes a las traslaciones en pixeles. A pesar de que se comenzó con una codificación de tipo binaria Gray, esta no representó cambios

significativos en comparación con una codificación continua, así que se prefirió el uso de la segunda dado que no requiere conversiones entre sistemas y, por tanto, tiene un tiempo de ejecución más corto.

5.2.1 Muestreo

Uno de los procesos más importantes para asegurar la convergencia y velocidad del algoritmo es el muestreo. Algunos tipos de muestreo comienzan distinguiendo información de los bordes y agregan más características conforme se encuentran correspondencias [Bolles y Horaud, 1984], aproximaciones por medio de superficies más simples [Tsuji y Nakamura, 1975; Holland, 1976]; y segmentaciones de acuerdo a características específicas, como discontinuidades [Grimson, 1986, Perkins, 1978]. En general, estos últimos son mejores en el sentido de que son más sensibles con respecto a cambios pequeños en las superficies y contienen menor número de puntos [Fan, 1988].

En particular, los algoritmos de muestreo para superficies tridimensionales requieren [Kuffner, 2004]: 1) una distancia métrica definida en el espacio configurado, 2) una función para generar una muestra en el espacio, y 3) un método para interpolar los puntos muestreados.

Para este trabajo el punto 1) se encuentra definido por una métrica en píxeles, pues la normalización de los datos es un dato indispensable para acelerar la ejecución del algoritmo; el punto 2) se logra a través de tres distintos muestreos, uno basado en las características geométricas de las nubes a registrar y los otros dependientes de la densidad de puntos y las superficies traslapadas; finalmente, el punto 3) resulta irrelevante, ya que en todo momento se cuenta con la información completa de las nubes.

A continuación se describen los tres tipos de muestreos que tienen lugar a lo largo de la ejecución del algoritmo genético. El primero, basado en las características geométricas de las superficies y el segundo, cuyo objetivo es reducir drásticamente la cantidad de puntos, se realizan una sola vez para cada nube, al principio del algoritmo. El último muestreo es dependiente de las áreas de superposición y, por tanto, debe de llevarse a cabo antes de calcular la aptitud de cada individuo de la población.

El primer muestreo consiste en determinar qué datos son relevantes para la reconstrucción a partir de los mapas gradientes de los datos a acoplar. Entonces, dado un mapa de profundidad $Z = Z(x, y)$, su mapa gradiente estará dado por

$$G = \sqrt{\left(\frac{\partial Z}{\partial x}\right)^2 + \left(\frac{\partial F}{\partial y}\right)^2} \quad (5.20)$$

en donde $\frac{\partial Z}{\partial x}$ y $\frac{\partial F}{\partial y}$ son las derivadas parciales de primer orden de la superficie Z . Pueden utilizarse operadores de Sobel para calcular las derivadas, aunque esto requiere de la aplicación de una

convolución con un filtro Gaussiano dada su esencialidad como pasa altas y su sensibilidad al ruido [Jin y DuanSheng, 2002; Castleman, 1996].

A continuación es necesario calcular la desviación estándar para determinar la dispersión de los datos en el mapa de gradiente, este cálculo se define por

$$\sigma = \sqrt{\frac{\sum(G_{ij} - \bar{G})}{N}} \quad (5.21)$$

donde G_{ij} representa a un valor gradiente en el punto (i, j) , \bar{G} es el promedio de los gradientes y N es el tamaño de la muestra, en este caso la cantidad de puntos en el espacio Euclídeo de cada nube de puntos. Para cada nube X_G y P_G se obtienen entonces un nuevos tamaños de muestra N_G^X y N_G^P . La razón de muestreo dada la aplicación de la dispersión del espacio gradiente estará determinada por: $r_G = N_G/N$.

Para una muestra uniforme con distribución normal se sabe que el 99% de los datos se encontrarán dentro de tres desviaciones estándar a partir de la media. Esto no es necesariamente cierto para las superficies a analizar, sin embargo, este parámetro funciona como un buen discriminante para remover puntos en que no existe un cambio significativo de gradiente. Al aplicar este muestreo en superficies con grandes cambios de gradiente, se eliminan secciones planas en donde no se tiene información relevante, mientras que si las superficies a registrar son más bien planas, no se elimina ninguna información.

El siguiente paso consiste en realizar un muestreo uniforme para reducir de manera drástica el tamaño de la muestra (se comienza con superficies de más de 20,000 puntos). Así, de manera general, el número de puntos a conservar en las muestras X_U, P_U será

$$N_U = \text{ceil}(r_U \cdot N_G) \quad (5.22)$$

En donde r_U representa el porcentaje de puntos que desea conservarse de la primera muestra y el operador *ceil* indica que debe de redondearse al siguiente entero superior.

Sub-espacio	Descripción
S_{OV}^x	Representa los puntos de la nube fija contenidos en el área de traslape.
S_{OL}^x	Son los puntos de la nube fija que no se encuentran dentro de los límites de superposición.
S_{OV}^p	Espacio que contiene los puntos de la nube flotante en el área de traslape.
S_{OL}^p	Denota a los puntos de la nube flotante que no entran en la superficie de traslape.

Tabla 5.2 Definición de parámetros para clasificación de superficies

El último muestreo depende de si existe o no un área superpuesta entre las nubes una vez que se han aplicado los parámetros de transformación. Antes de llevar a cabo esta reducción del espacio de búsqueda es necesario definir otros sub-espacios, como se muestra en la tabla 5.2.

La definición de estos sub-espacios dadas una nube fija $X_U = \{x_i\}$ y una flotante $P_U = \{p_i\}$ depende del establecimiento de los límites en el espacio euclídeo dependientes de los máximos y mínimos de cada nube, estos límites se eligen de la siguiente manera

$$\begin{aligned} l_i &= \max\{\min(X_U), \min(P_U)\} \\ l_s &= \min\{\max(X_U), \max(P_U)\} \end{aligned} \quad (5.23)$$

en donde l_i y l_s son vectores que contienen los límites $[x_{min}, y_{min}, z_{min}]$ y $[x_{max}, y_{max}, z_{max}]$, respectivamente, que encierran a las dos superficies dentro del rango en que existe superposición. Una vez definidos estos límites, los sub-espacios descritos en la tabla 5.2 pueden verse como

$$\begin{aligned} S_{OV}^x &= \{X_U: l_i \leq X_U \leq l_s\} \\ S_{OL}^x &= \{X_U: X_U < l_i, \quad X_U > l_s\} \\ S_{OV}^p &= \{P_U: l_i \leq P_U \leq l_s\} \\ S_{OL}^p &= \{P_U: P_U < l_i, \quad P_U > l_s\} \end{aligned} \quad (5.24)$$

Por ejemplo, para dos círculos de radio 1, siendo que el centro de uno de ellos se encuentra en el origen y el del segundo se encuentra en (1,0), estos sub-espacios se verían como se muestra en la figura 5.4.

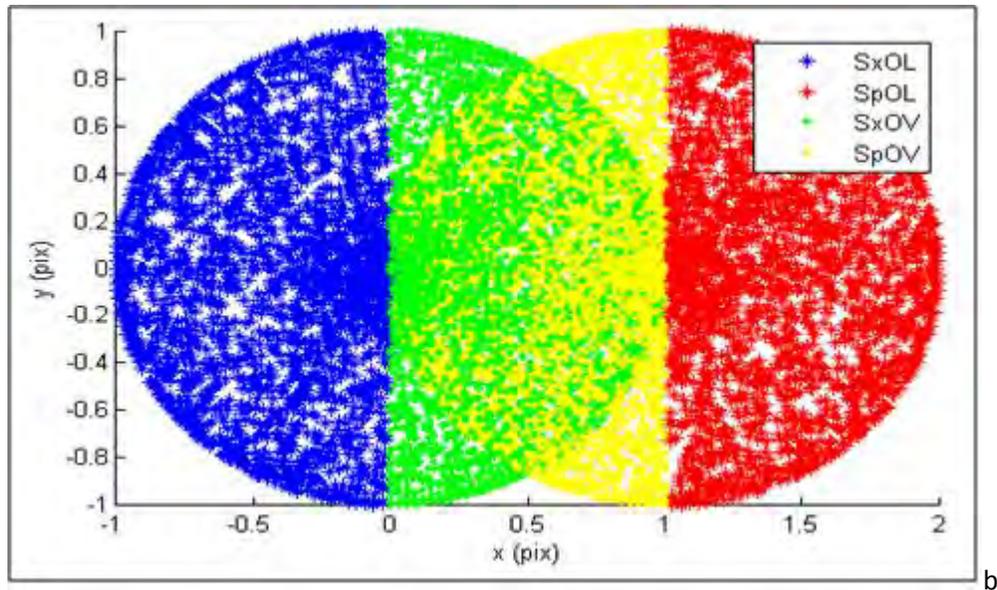


Figura 5.4 Sub espacios muestrales para dos círculos unitarios superpuestos

Cabe hacer notar que si no existe una superficie de traslape, los tamaños de S_{OV}^x y S_{OV}^p serán $N_{OV}^x = N_{OV}^p = 0$ y los espacios S_{OL}^x y S_{OL}^p tendrán tamaños $N_{OL}^x = N_U^x$ y $N_{OL}^p = N_U^p$, respectivamente.

A continuación se procede a realizar un muestreo aleatorio en que se elige una densidad para las áreas de traslape de al menos el doble de aquella utilizada en los puntos que no se encuentran dentro de este parámetro. Así, $r_{OV} = 2r_{OL}$; este criterio se aplica para tener un sobre-muestreo en el área de interés, pues el parámetro Y en la ecuación 5.14 depende de la cantidad de puntos que se alineen exitosamente en esta región.

La cantidad de puntos para cada sub-muestra será entonces

$$\begin{aligned} N_{OV}^{xr} &= r_{OV} \cdot N_{OV}^x \\ N_{OV}^{pr} &= r_{OV} \cdot N_{OV}^p \\ N_{OL}^{xr} &= r_{OL} \cdot N_{OL}^x \\ N_{OL}^{pr} &= r_{OL} \cdot N_{OL}^p \end{aligned} \quad (5.25)$$

La razón de muestreo total para cada nube, debido a los diferentes pasos estará dada por

$$r = \frac{N_{OV}^r + N_{OL}^r}{N} \quad (5.26)$$

que al aplicar las relaciones dadas en las ecuaciones 5.22-5.26, puede reescribirse como

$$r = \frac{r_{OV} \cdot (r_U \cdot N_G - N_{OL})}{N} \quad (5.27)$$

De esta manera, si se desea conservar una razón de muestreo r , basta con fijar los parámetros r_U y r_{OV} , pues no se tiene control sobre N_G , ya que este es dependiente de la geometría de la nube, sin embargo, este parámetro se conoce y es constante para todas las generaciones. Resta considerar la cantidad N_{OL} , que se llega a conocer hasta que se realiza el tercer muestreo, sin embargo, se considera que es suficientemente pequeña con relación al tamaño total de la muestra N y, por tanto, puede descartarse.

5.2.2 Cálculo de la aptitud de cada individuo

Una vez que se tiene una muestra representativa de cada una de las nubes a registrar (muestreo por características y uniforme), puede procederse a crear la primera generación de individuos, tomando en cuenta las siguientes restricciones: 1) dado que las distancias en los ejes se encuentran normalizadas, no habrá desplazamientos mayores a 1, 2) existe un área de traslape de al menos 25% entre las superficies adquiridas, y 3) dado 2), el valor absoluto de cualquier ángulo de rotación no será mayor a 120° .

Un cromosoma estará denotado por $\chi_i^{(k)}$ y se referirá al i -ésimo cromosoma de la generación k para $i = 1, \dots, N_{pop}$ y $k = 1, \dots, N_{gen}$, en donde N_{pop} representa el tamaño de la población y N_{gen} es el número máximo de generaciones a generar. Entonces, la primera generación $\chi^{(1)}$ estará formada por N_{pop} cromosomas codificados de la forma

$$\chi_i = [\alpha_i, \beta_i, \gamma_i, T_{xi}, T_{yi}, T_{zi}], i = 1, \dots, N_{pop} \quad (5.28)$$

Los parámetros α, β y γ corresponden a la rotación alrededor de los ejes x, y y z , respectivamente, siguiendo la convención mostrada en la figura 5.3 y se generarán de manera aleatoria en el rango $[-120^\circ, 120^\circ]$. Los últimos tres parámetros son traslaciones sobre cada eje, se generan también de manera aleatoria en el rango $[-1, 1]$.

El cálculo de la aptitud para cada individuo se calcula a partir de la función objetivo mostrada en la ecuación 5.13 y está dada por

$$f(\chi) = |d_{OL} - Yd_{OV} - \varepsilon|^2$$

Sea X_U la nube de puntos fija después de aplicar los dos primeros muestreos y tP_U la nube flotante después de ser muestreada y de haber sufrido la transformación geométrica dados los parámetros en χ . La distancia euclídea $d(q_1, q_2)$ entre dos puntos $q_1 = (x_1, y_1, z_1)$ y $q_2 = (x_2, y_2, z_2)$ está definida por

$$d(q_1, q_2) = \|q_1 - q_2\| = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \quad (5.29)$$

Entonces, el parámetro de error relacionado con el muestreo será la distancia euclídea promedio entre puntos vecinos multiplicada por el parámetro de muestreo r . Este error se resta de la función de aptitud para asegurar que haya una tolerancia suficiente para permitir que la función tienda a 0 cuando existe un registro perfecto, situación que en realidad no es posible dado el muestreo aleatorio que tiene lugar a continuación. El cálculo de este parámetro se hace en un solo sentido, pues la cercanía con otros puntos vecinos es muy similar en cualquier dirección, el error es constante para toda la ejecución y estará dado por

$$\varepsilon = \frac{r}{N} \sum_{i=1}^{N-1} d(x_{i+1}, x_i) \quad (5.30)$$

Si después de realizar la última operación de muestreo a X_U y tP_U , existen puntos en los sub-espacios S_{OV}^x y tS_{OV}^p , es decir, si hay posible traslape entre las nubes, entonces d_{OV} será la distancia mínima promedio entre los puntos de ambos sub-espacios sin reemplazo, en otras palabras, no se removerá ningún punto de las muestras. Esta distancia estará determinada por

$$d_{OV} = \frac{1}{N} \sum_{i=1}^N \min\{d(s_i, S)\} \quad (5.31)$$

En la ecuación 5.28, s denota el sub-espacio en el área de traslape cuya longitud sea menor y S el de mayor tamaño, en cuyo caso N será el tamaño de s . Esta distinción se hace porque, por principio de cuentas, las nubes no necesariamente serán del mismo tamaño y esta distinción será aún más marcada después del muestreo aleatorio. Se considera el tamaño mínimo dada la mayor velocidad de procesamiento que esto implica.

El operador de emparejamiento exitoso Y dependerá también de las distancias mínimas entre los puntos de una nube y otra. Considérese una variable λ cuyo valor inicial es 0 y se incrementa

en dos cada vez que se tiene un acoplamiento exitoso, mismo que se dará si $\min\{d(s_i, S)\} \leq \varepsilon$, el valor de Y estará dado entonces por

$$Y = \frac{\lambda}{N_{OV}^{xr} + N_{OV}^{pr}} \quad (5.32)$$

Puede observarse que Y alcanzará un valor máximo de 1 si ambos sub-espacios son del mismo tamaño y todos sus puntos se acoplan con éxito. En caso de que Y sea 0, solo contribuirán a la función objetivo la distancia d_{OL} y el ajuste por muestreo ε .

De manera similar, d_{OL} se calculará como una distancia mínima promedio entre las nubes que no se encuentren dentro del rango de superposición. El cálculo de este parámetro será como sigue

$$d_{OL} = \frac{1}{N} \sum_{j=1}^N \min\{d(s_i, S)\} \quad (5.33)$$

En este caso se hacen las mismas consideraciones en cuanto a la selección de las variables que como se hizo para d_{OV} .

De la función objetivo, puede verse que si no existe un área de traslape, d_{OL} tiende a crecer, mientras que entre mayor sea la proporción de emparejamientos exitosos, mayor será la cantidad restada a d_{OL} en función de d_{OV} .

5.2.3 Operadores de selección, cruza y mutación

Otro aspecto muy importante para la convergencia del algoritmo genético es la correcta elección de los operadores que van a actuar sobre la población. La dificultad de este proceso reside en que alguna combinación puede funcionar muy bien para un problema en particular pero fallar por completo para un problema distinto [Goldberg, 1989]. La mejor elección de los operadores se hace si se tiene algún conocimiento *a priori* de la función que se desea optimizar [Goldberg, 1989; Haupt y Haupt, 2004; Chambers, 1995].

Para el caso de registro de nubes de puntos, se sabe que pueden existir gran cantidad de mínimos locales, por lo que lo más conveniente es elegir un operador de selección que aplique poca presión sobre la población, ya que aún los individuos con menos aptitud pueden contener información valiosa para la optimización de las distancias.

El tipo de selección que presiona lo suficiente sin dejar de considerar individuos con poco desempeño es el de clasificación (ranking) lineal [Chambers, 1995], pues este permite que gran parte de la población tenga prole dependiendo del valor de linealización s que se elija [Baker, 1989], para $s = 2$ se tiene que alrededor del 75% de la población está involucrada en dar descendencia a la siguiente generación, mientras que con $s = 1.1$ se considera un 94%.

Puede recordarse del capítulo 4 que el primer paso para hacer una clasificación es el de ordenar las aptitudes de los individuos (en este caso de menor a mayor) para asignar una nueva aptitud. El

primer individuo recibe una aptitud igual a s y las aptitudes de los demás k cromosomas estará dada por

$$f^{(k)} = s - \frac{2 \cdot k(s - 1)}{(N_{pop} - 1)} \quad (5.34)$$

El parámetro s se mantiene constante durante todo el algoritmo, esto implica darle oportunidad de tener prole a la mayoría de los cromosomas.

Una vez que se ha adaptado la nueva función de aptitud, se utiliza el método de ruleta para seleccionar a los padres que darán lugar a la nueva generación. Se considera también un elitismo, es decir, conservación de los mejores individuos de la población, dado por un porcentaje de la población total. El número de individuos a conservar estará dado entonces por

$$N_{elite} = r_{elite} \cdot N_{pop} \quad (5.35)$$

Debido a que el cromosoma tiene una longitud muy corta, se elige el método de cruce de un punto. Sin embargo, para introducir nueva información se utiliza la combinación propuesta por [Radcliff, 1991], en donde, para el punto de cruce, se eligen nuevos valores p_{new} que están dados por

$$p_{new} = \xi p_{mn} + (1 - \xi)p_{fn} \quad (5.36)$$

en donde ξ es un valor aleatorio en el intervalo $[0,1]$, p_{mn} es la n -ésima variable en el cromosoma padre y p_{fn} es la n -ésima variable del cromosoma madre. Entonces, se generan nuevas variables p_{new} para cada alelo del cromosoma y estas se cruzan como si se tratara de nuevos padres. Este proceso se ilustra en la figura 5.5.

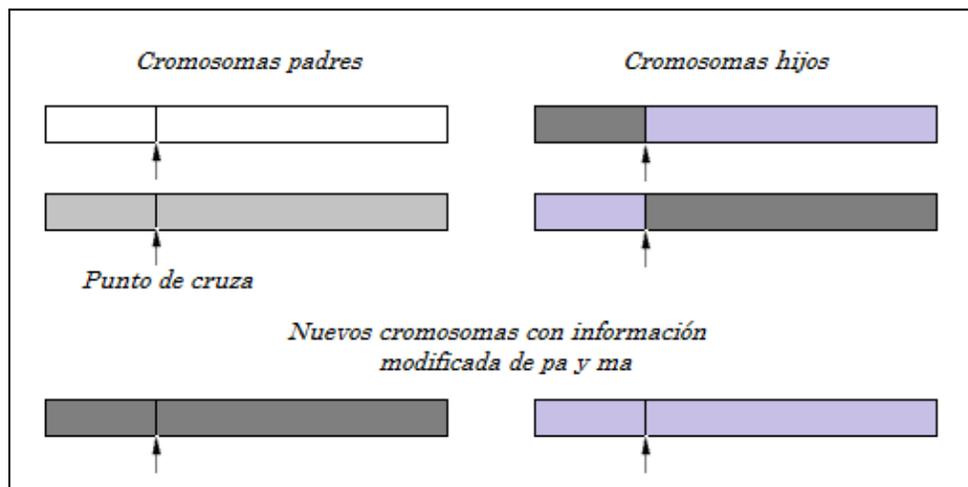


Figura 5.5 Cruza con generación de nueva información

Finalmente, se aplica una mutación simple en que al alelo a mutar se agrega un valor aleatorio ξ comprendido en el intervalo $[m_{min}, m_{max}]$. Para este problema, en que el algoritmo es muy sensible a cambios, sobre todo en los tres primeros parámetros de su codificación, se requieren

primero variaciones grandes y estas deben irse reduciendo no sólo en función del número de generación en que se encuentre, sino también de la aptitud de la población en comparación con el peor desempeño registrado. La solución es asignar una mutación dinámica que estará dada por

$$m_{new} = \xi \cdot \left[2 - \frac{k}{N_{pop}} - \left(1 - \frac{f_{prom}^{k-1}}{f_{max}} \right) \right] \quad (5.37)$$

en donde ξ es un valor aleatorio comprendido en el intervalo $[m_{min}, m_{max}]$, k es el número de generación en que se encuentra el algoritmo, f_{prom}^{k-1} es la aptitud promedio de la generación anterior y f_{max} es la aptitud máxima o peor en la población general. El parámetro 2 se elige para que el mínimo sea 0 y esto se espera dentro de las últimas generaciones.

Hasta este punto se ha detallado la implementación del algoritmo ICP y la parametrización del algoritmo genético. En la siguiente sección se presentan algunas pruebas que comprueban que la metodología planteada presenta una buena convergencia y es viable para aplicaciones reales.

5.3 Experimentos

El algoritmo descrito a lo largo del capítulo se probó en dos diferentes aplicaciones: a) para registrar una superficie sintética generada en MATLAB, y b) para acoplar dos superficies provenientes de una misma adquisición, seccionadas también de manera sintética. Ambas pruebas se realizaron con un tamaño de población $N_{pop} = 100$ y con límite de generaciones $N_{gen} = 120$. El parámetro para detener el algoritmo se estableció cuando $f_{min}/f_{max} < 0.005$ o cuando $|\bar{f} - f_{min}| < 0.001$. Se utilizó una selección por clasificación lineal con $s = 1.2$ y los intervalos de muta se establecieron en $[-40, 40]$ y $[-0.5, 0.5]$ para los ángulos y los desplazamientos, respectivamente.

5.3.1 Registro de una superficie sintética

En este experimento se generó una superficie sintética en MATLAB con función *peaks* con un tamaño de 200×200 , misma que fue seccionada en dos partes: $X_1 = \{X: 1 \leq X \leq 130\}$ y $X_2 = \{X: 80 \leq X \leq 200\}$, por convención con lo explicado en el capítulo, se denominará a la nube fija X_1 como X y a la nube flotante como P . En este caso se obtiene $N^x = 26,000$ y $N^p = 24,200$, con un área común de 25% entre ellas.

Se utilizó una razón de elitismo de $r_{elite} = 0.15$ y se conservó un 5% de la muestra ($r = 0.05$).

Se generaron parámetros de transformación aleatorios para la nube flotante, quedando estos como: $\alpha = -5.8354$, $\beta = 3.6331$, $\gamma = 82.4747$, $T_x = 0.4626$, $T_y = 0.2581$ y $T_z = 0.8410$. El objetivo a alcanzar y las nubes después de la transformación se muestran en la figura 5.6.

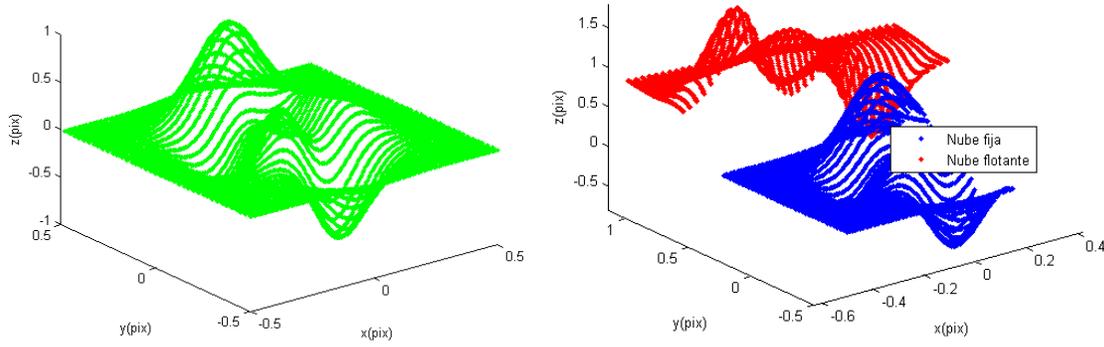


Figura 5.6 a) Objetivo y, b) nube fija y nube flotante con transformación

El algoritmo calculó un error inicial de 0.4756 y alcanzó un criterio de paro en la generación 80 al cumplirse la condición $f_{min}/f_{max} < 0.005$ con un valor $f_{min} = 0.001868$. En la figura 5.7 se muestra el resultado obtenido del algoritmo genético, mientras que la figura 5.8 es una comparación de la aptitud mínima contra el promedio de las aptitudes en cada generación.

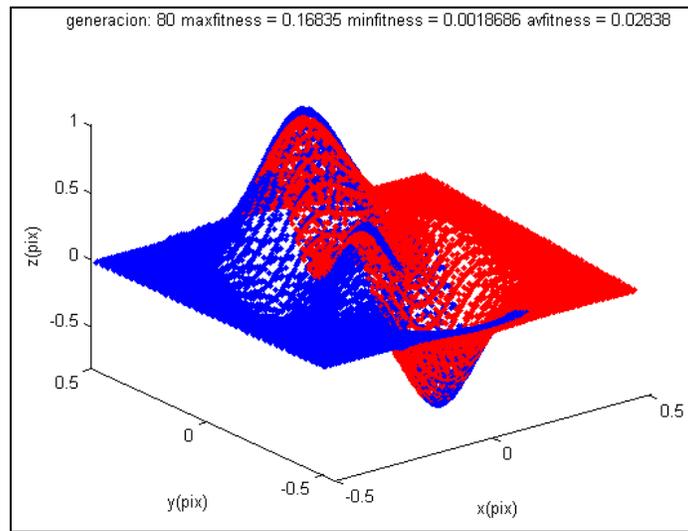


Figura 5.7 Resultado obtenido del algoritmo genético

El algoritmo concluyó en un tiempo de 219.83 segundos con un tiempo promedio por generación de 2.7479. Los parámetros encontrados fueron: $\alpha = -0.6583$, $\beta = -8.8356$, $\gamma = -85.8152$, $T_x = 0.1638$, $T_y = -0.5948$ y $T_z = -0.7896$.

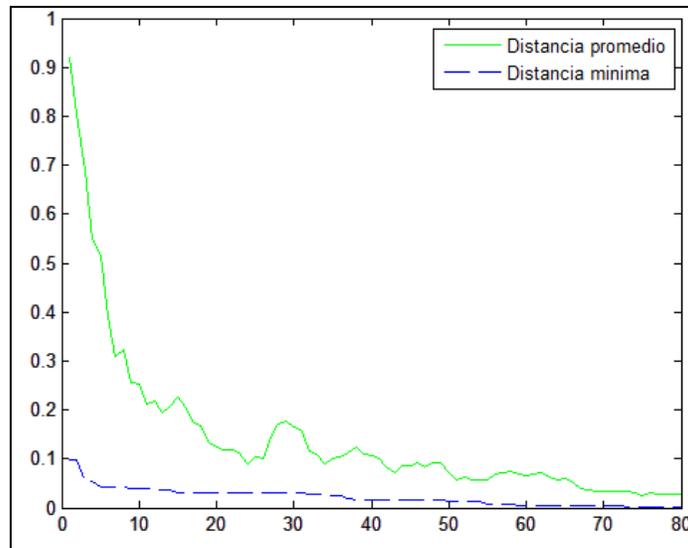


Figura 5.8 Estadística de la prueba

El resultado después de aplicar el algoritmo ICP se muestra en la figura 5.9, este tuvo un tiempo de corrida de 19.32 segundos y obtuvo como resultados los parámetros siguientes: $\alpha = 6.1894$, $\beta = -3.2874$, $\gamma = -83.1392$, $T_x = -0.4280$, $T_y = -0.2349$ y $T_z = -0.8172$.

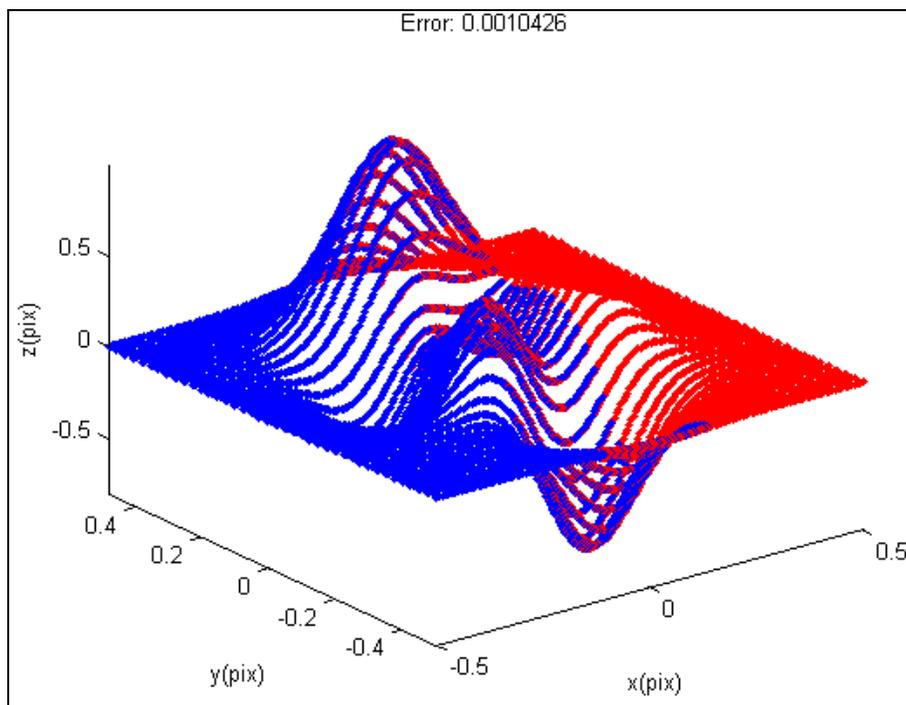


Figura 5.9 Resultado de aplicar ICP a los parámetros obtenidos por el GA

5.3.2 Registro de una superficie adquirida digitalmente

Para la segunda prueba, se adquirió un mapa de profundidad de una superficie real. La adquisición se hizo a través de un sistema cámara-proyector en que los ejes ópticos fueron no paralelos y no coplanares. Se utilizó una cámara CCD PixelLink con una lente telefoto con una apertura de F1.8-32C y un proyector DELL con una resolución VGA de 640 x 480. La calibración del sistema de proyección se hizo a través de diversos planos colocados a 20, 30 y 40cm, sobre los cuales se proyectaron patrones de franjas para obtener los parámetros internos y externos tanto de la cámara como del proyector.

El método utilizado para la adquisición fue el de corrimiento de fase y se utilizaron 4 pasos. El desenvolvimiento se llevó a cabo por el método de relación de tangente, seno y coseno explicado en el capítulo 2 y que está denotado por la ecuación 2.47 como

$$\phi_d(x_i) = \phi_d(x_{i-1}) + atan \left[\frac{sen(\phi_e(x_{i-1}) - \phi_d(x_{i-1}))}{cos(\phi_e(x_{i-1}) - \phi_d(x_{i-1}))} \right]$$

Las imágenes adquiridas por la cámara tienen un tamaño de 640 x 480 pixeles, es decir, se cuenta con una nube de puntos inicial con un tamaño de 307,200 puntos. En la figura 5.10 se muestra una fotografía del objeto real, así como una vista de la reconstrucción hecha a partir del método de corrimiento de fase.

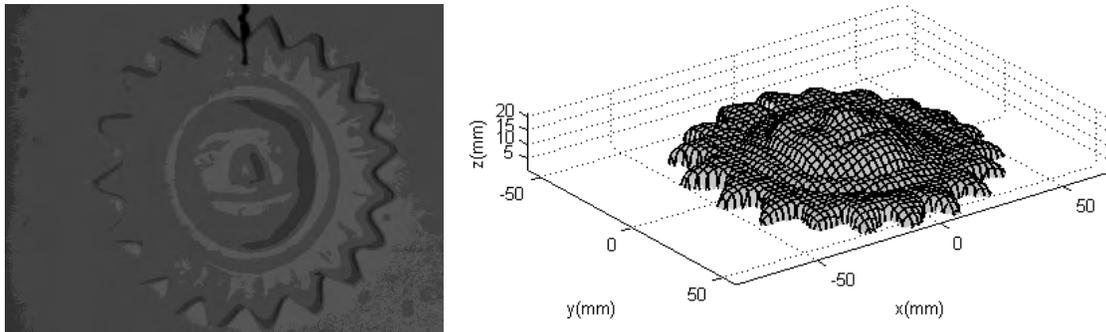


Figura 5.10 a) Fotografía de objeto real y, b) reconstrucción del mismo.

Debido a que muchos puntos en la imagen no son puntos correspondientes a la superficie de interés, el tamaño N total de la muestra se reduce a 137,092 puntos. En esta ocasión se conserva el 1% del tamaño de la muestra. Una vez más se divide la muestra en dos espacios, correspondientes al 25% del área de traslape. Las nuevas nubes tienen tamaños $N^x = 93,980$ y $N^p = 108,337$ puntos. Se vuelve a utilizar una razón de elitismo $r_{elite} = 0.15$. El objetivo está ilustrado por la figura 5.10b y los parámetros de transformación para la nube P se generan de forma aleatoria, quedando: $\alpha = 39.1182$, $\beta = 2.8947$, $\gamma = 111.0243$, $T_x = -0.5068$, $T_y = -0.2547$ y $T_z = 0.7538$.

La nube fija y la nube flotante después de la transformación se muestran en la figura 5.11.

Angles: 39.11824 2.894733 111.0243 Displacements: -69.7417 -29.3442 15.978

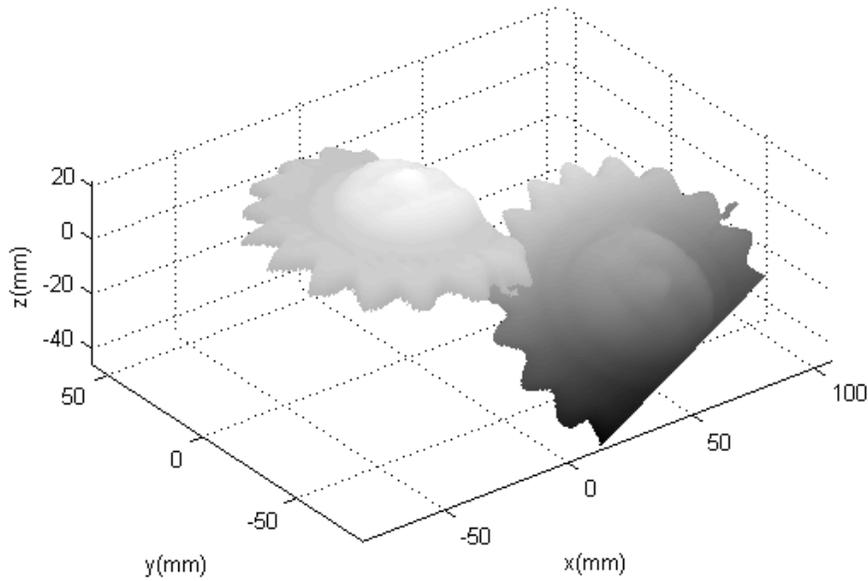


Figura 5.11 Nube fija X y nube flotante después de la transformación tP

El error inicial calculado por el algoritmo fue de 0.4709 y alcanzó las 120 generaciones antes de detenerse por algún criterio. El resultado obtenido después de la aplicación del algoritmo se muestra en la figura 5.12 y la estadística de la prueba se plasma en la figura 5.13.

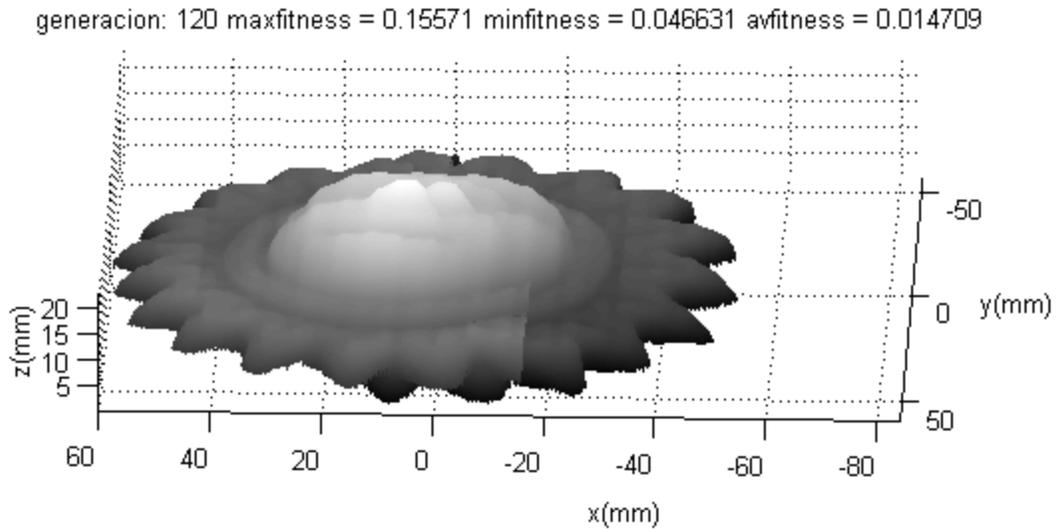


Figura 5.12 Resultado obtenido del algoritmo genético

El algoritmo llegó a su fin en un tiempo de 385.73 segundos, con un tiempo promedio de 3.21 segundos por generación. Los parámetros encontrados fueron: $\alpha = -30.3857$, $\beta = -1.6836$, $\gamma = -107.7356$, $T_x = 0.3861$, $T_y = 0.3074$ y $T_z = -0.7896$.

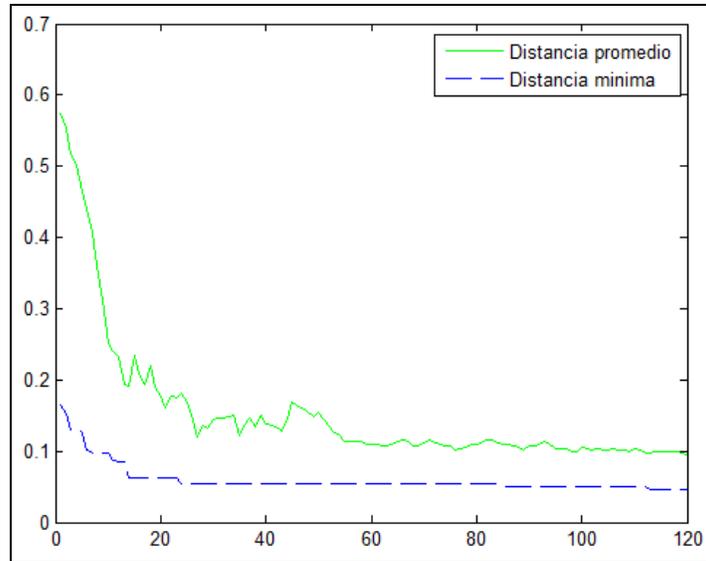


Figura 5.13 Estadística de la prueba

El resultado después de aplicar el algoritmo ICP se muestra en la figura 5.14, este tuvo un tiempo de corrida de 24.73 segundos y obtuvo como resultados los parámetros siguientes: $\alpha = -38.8472$, $\beta = -3.1821$, $\gamma = -109.3874$, $T_x = 0.4072$, $T_y = -0.2619$ y $T_z = -0.7961$.

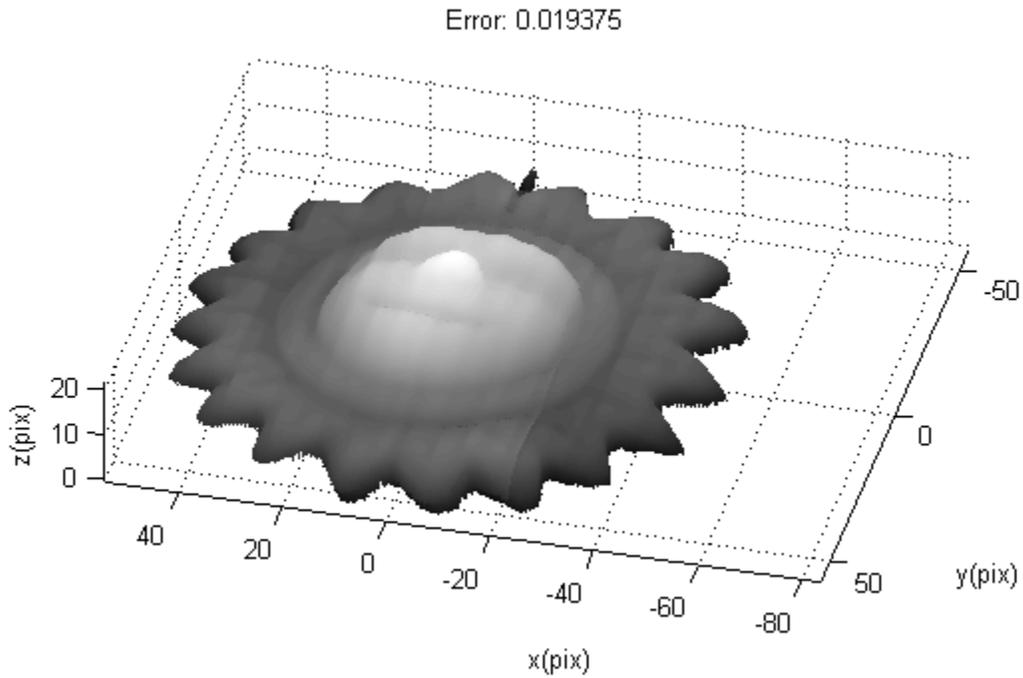


Figura 5.14 Resultado de aplicar ICP a los parámetros obtenidos por el GA

5.4 Limitaciones y consideraciones

En las aplicaciones presentadas hasta el momento, se han utilizado superficies que no cuentan con mucha simetría en los espacios particionados. Sin embargo, de aquí puede verse la mayor limitación que tiene el algoritmo: puede presentar una solución errónea cuando existe simetría entre las diferentes secciones. Una solución alternativa a que ha llegado a converger para la superficie 'Sol' se muestra en la figura 5.15.

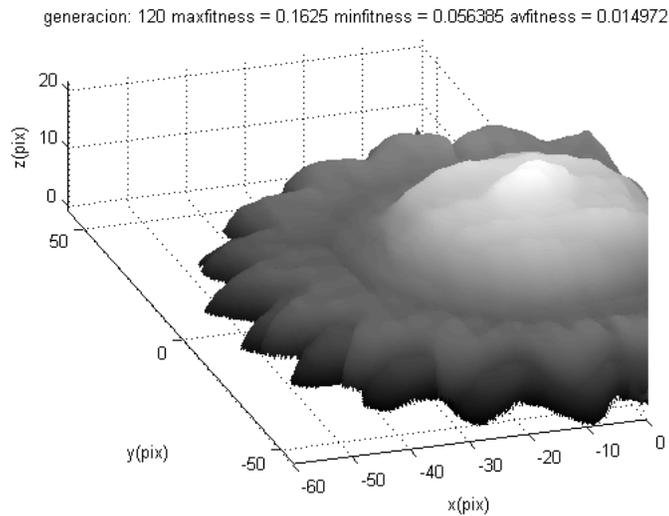


Figura 5.15 Una mala solución antes de aplicar el algoritmo ICP

El tipo de solución mostrado en la figura 5.15 tiene lugar cuando, debido a los parámetros de mutación, el valor absoluto de la rotación alrededor de alguno de los ejes es muy cercano a 180° . Sin embargo, no pueden descartarse parámetros de rotación con esta magnitud, pues aún puede existir alguna solución válida aunque exista un área de traslape mínima.

Esta limitante puede observarse más claramente en la figura 5.17. El objetivo a registrar es una superficie gaussiana como la mostrada en la figura 5.16.

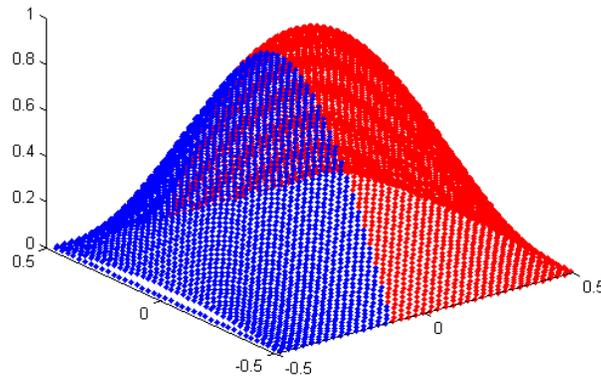


Figura 5.16 Objetivo para una prueba con una curva gaussiana

generacion: 120 maxfitness = 0.069086 minfitness = 5.649e-005 avfitness = 0.016272

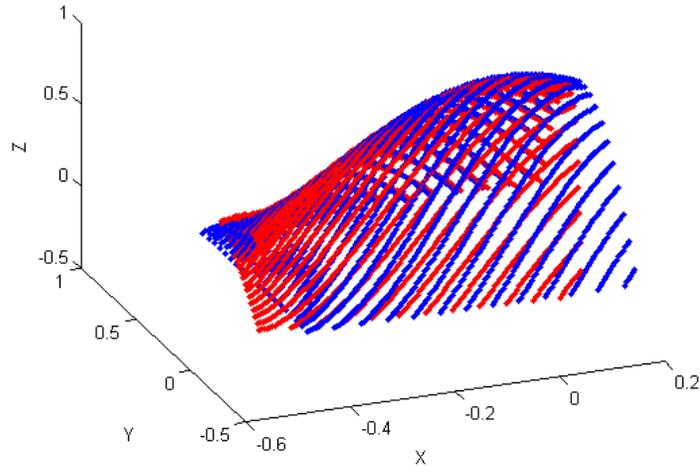


Figura 5.17 Una solución incorrecta para la curva mostrada en la figura 5.16 antes de ICP

El tiempo de reconstrucción es también relativamente alto (alrededor de 300 segundos para dos nubes con un procesador Centrino Duo a 1.2GHz) si se considera que una superficie completa toma más de de dos adquisiciones. Sin embargo, el tiempo de adquisición es rápido si se considera que los métodos de proyección de franjas adquieren una superficie grande en comparación con otras técnicas de proyección de luz estructurada, así que, en general, el planteamiento propuesto no demanda mucho tiempo por parte del usuario y resulta confiable para superficies que puedan distinguirse por ciertas diferencias geométricas.

Referencias

- [1] Abdi, H., *'Encyclopedia of Measurements and Statistics: Singular Value Decomposition and Generalized Singular Value Decomposition'*, Ed. Neil Salkind. Thousand Oaks, CA., 2007.
- [2] Arun, K.S., Huang, T.S., Blostein, S.D., *'Least square fitting of two 3D point sets'*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **9**: 698 – 700 (1987).
- [3] Baker, J.E., *'An analysis of the effects of selection in genetic algorithms'*, Ph.D. thesis, Vanderbilt University, Nashville, 1989.
- [4] Ballesta, M., Reinoso, O., Gil, A., Julia, M., Paya, L., *'Métodos para el alineamiento de mapas de características visuales'*, Universidad Miguel Hernández, Alicante, España, 2007.
- [5] Bentley, J.L., *'Multidimensional Divide and Conquer'*, Communications of the ACM, **23**: 214 – 229 (1980).
- [6] Besl, P. J., McKay N.D., *'A method for registration of 3-D shapes'*, IEEE Trans. Pattern Anal. Machine Intell., **14**: 239-256 (1992).
- [7] Bolles, R.C., Horaud, P., *'3DPO: A three-dimensional part orientation system'*, Int. J. Robotic Res., **5**: 3 – 26 (1986).
- [8] Brunnstrom, K., Stoddart, A.J., *'Genetic algorithms for free-form surface matching'*, Proceedings of the 13th International Conference on Pattern Recognition, **4**: 689 – 693 (1996).
- [9] Castleman, K.R., *'Digital Image Processing'*, Prentice Hall, 1996 (Appendix 3: Mathematical Background).
- [10] Chambers, L.D., *'The Practical Handbook of Genetic Algorithms: New Frontiers Volume II'*, CRC Press, 1995.
- [11] Chavarria, M.A., Sommer, G., *'Correlation ICP algorithm for pose estimation based on local and global features'*, VISAPP **2**: 528 – 534 (2008).
- [12] Chen, Y., Medioni, G., *'Object modeling by registration of multiple range images'*, Image Vision Comput., **10**: 145 – 155 (1992).
- [13] Chow, C.K., Tsui, H.T., Lee, T., *'Surface registration using a dynamic genetic algorithm'*, Pattern Recognition, **37**: 105 – 117 (2004).
- [14] Fan, T.J., *'Describing and recognizing 3-D objects using surface properties'*, Ph.D. dissertation, IRIS-237, Inst. Robotics Intell Syst., Univ. of Southern California, L.A., 1988.
- [15] Friedman, J.H., Bentley, J.L., Finkel, R.A., *'An Algorithm for finding best matches in logarithmic expected time'*, ACM Trans. on Mathematical Software, **3**: 209 – 226 (1977).
- [16] Goldberg, D.E., *'Genetic Algorithms in Search, Optimization & Machine Learning'*, Addison-Wesley, 1989.

- [17] Golub, G.H., Van Loan, C.F., *'Matrix computations'*, The John Hopkins University Press, Baltimore, Maryland, 2nd ed., 1989.
- [18] Grimson, W.E.L., *'The combinatorics of local constraints in model-based recognition and localization from sparse data'*, J. ACM, **33**: 658 – 686 (1986).
- [19] Haupt, R.L., Haupt, S.E., *'Practical genetic algorithms'*, Wiley-Interscience, 2nd Ed., 2004.
- [20] Holland, S.W., *'A programmable computer vision system based on spatial relationships'*, General Motors Publ. GMR-2078. Detroit: General Motors, 1976.
- [21] Jing, C., DuanSheng, C., *'Robust face recognition using Gradient Map'*, HuaQiao University, Quanzhou, P.R. China, 2002.
- [22] Kuffner, J.J., *'Effective sampling and Distance Metrics for 3D Rigid Body Path Planning'*, Proc. IEEE Int. Conf. on Robotics and Automation, pp. 1 – 6 (2004).
- [23] Lorusso, A., Eggert, D., Fisher, R., *'A comparison of four algorithms for estimating 3D rigid transformations'*, Proceedings of the 4th British Machine Vision Conference, pp. 237 – 246, Birmingham, England, 1995.
- [24] Matabosch, C., Salvi, J., Pinsach, X., Pages, J., *'A comparative survey on free-form surface registration'*, Institut d'Informatica i Aplicacions, Universitat de Girona, Spain, 2004.
- [25] Moore, A.W., *'Efficient Memory-based Learning for Robot Control, PhD Thesis'*, Technical Report No. 209, Computer Laboratory, University of Cambridge, 1991.
- [26] Muller, N., Magaia, L., Herbst, B.M., *'Singular Value Decomposition, Eigenfaces, and 3D Reconstructions'*, SIAM Review, **46**: 518 – 545 (2004).
- [27] Percoco, G., Galantucci, L. M., *'Genetic Point Cloud Alignment for Computer Aided Inspection and Reverse Engineering'*, Politecnico di Bari, Italy, 2002.
- [28] Perkins, W.A., *'A model-based vision system for industrial parts'*, IEEE Trans. Comput, **27**: 126 – 143 (1978).
- [29] Pottman, H., Leopoldseder, S., Hofer, M., *'Registration without ICP'*, Computer Vision and Image Understanding, Volume **95**: 54 – 71 (2004).
- [30] Radcliff, N.H., *'Forma analysis and random respectful recombination'*, Proc. 4th Int. Conf. on Genetic Algorithms for real parameter optimization, San Mateo, CA: Morgan Kauffman, 1991.
- [31] Rieger, J., *'On the classification of views of piecewise smooth objects'*, Image and Vision Computing, **5**: 91 – 97 (2005).
- [32] Surmann, H., Nüchter A., Hertzberg, J., *'An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments'*, J. of Robotics and Autonomous Systems **45**: 181 – 198 (2003).

- [33] Tsuji, S., Nakamura, A., '*Recognition of an object in a stack of industrial parts*', Proc. 4th Int. Joint Conf. Artificial Intell., Los Altos: William Kaufmann, pp. 811 – 818, 1975.
- [34] Xiao, G., Ong, S.H., Foong, K.W.C., '*Efficient partial surface registration for 3D objects*', Computer Vision and Image Understanding, Volume **98**: 271 – 274 (2005).
- [35] Yaniv, Z. '*Rigid Registration: The Iterative Closest Point Algorithm*', School of Engineering and Computer Science, The Hebrew University, Israel, 2001.
- [36] Ze-Su, H. Bing-Rong, L. Hong, '*An Improved Polar Scan Matching Using Genetic Algorithm*', Information Technology Journal **6**: 89-95 (2007).
- [37] Zhang, Z., '*On local matching of free-form curves*', Proc. of BMVC, pp. 347 – 356, 1992.
- [38] Zhang, Z., '*Iterative point matching for registration of free-form curves and surfaces*', Int. J. of Comp. Vision, 13: 119 – 152 (1994).

Capítulo 6

Conclusiones y trabajo futuro

En este trabajo de tesis se analizaron diversas técnicas de visión robótica para la recuperación de forma o topografía de objetos. Las primeras revisadas fueron aquellas que necesitan más de una imagen del objeto bajo estudio, es decir las técnicas estereoscópicas fotográfica y fotométrica.

Para el caso del estéreo fotográfico se capturan dos o más imágenes desde distintas perspectivas del objeto. El cálculo de la forma del objeto se hace a través de la determinación de la localización de los puntos conjugados entre las imágenes. Estos puntos conjugados se encuentran al analizar la correlación de los atributos de un píxel dentro de una de las imágenes en una vecindad cercana contra otra vecindad del mismo tamaño en otra imagen. Este proceso suele ser tardado y con un alto costo computacional, por lo que no es una buena opción para aplicar en tiempo real. Una forma de acelerar el proceso de encontrar los pares conjugados consiste en aplicar marcas ajenas a la muestra, sin embargo, esto resulta invasivo para la muestra en sí, situación que no es consistente con el objetivo de esta tesis.

La técnica de estéreo fotométrico requiere de al menos dos imágenes del mismo objeto iluminado con fuentes colocadas en ubicaciones distintas. El principio de esta técnica es el de obtener los vectores normales a cada punto de la superficie a través del patrón de sombreado generado por cada fuente. Esta técnica presenta la limitación de que sólo funciona para superficies Lambertianas, es decir, que la irradiancia detectada sea proporcional al ángulo entre el vector normal a cada punto del objeto y la ubicación de la fuente de iluminación. El estéreo fotométrico considera la solución de un sistema de ecuaciones que requiere tan sólo de la inversión de una matriz y el conocimiento de la colocación de cada fuente de iluminación, sin embargo, como ya se mencionó, el resultado arrojado son los vectores normales a cada punto de la superficie y no la profundidad del objeto en sí, así que es necesario un proceso de integración de gradientes. La principal limitante de esta técnica es la necesidad de adquirir al menos tres imágenes, pues el uso de sólo dos puede ocasionar puntos indeterminados en la solución, además de que sólo se pueden obtener los puntos que se encuentren iluminados en todas las imágenes. Al igual que el método de estéreo fotográfico, la fotometría estéreo consume un largo tiempo de procesamiento (debido a la integración de gradientes) y, por lo tanto, no es adecuada para el enfoque propuesto por este trabajo de tesis.

Las técnicas de proyección de luz estructurada consisten en proyectar un patrón de iluminación conocido y calcular de manera geométrica la deformación de dicho patrón con respecto a un plano de calibración. Las técnicas más simples son: proyección un haz, de una línea de luz, de una matriz de puntos y de franjas o luz con perfil cosenoidal. Las primeras tres presentan grandes

limitaciones en cuanto a la superficie que pueden adquirir en una imagen, pues se obtiene sólo la porción del objeto que se encuentra iluminada. Este problema se ve resuelto al utilizar la técnica de proyección de luz con perfil cosenoidal.

La proyección de un patrón con perfil cosenoidal requiere, por lo general, de una sola imagen para llevar a cabo la reconstrucción tridimensional del objeto y suele aplicarse en aplicaciones de tiempo real. La reconstrucción puede realizarse al comparar el patrón de luz proyectado sobre el objeto contra el patrón sobre un plano de calibración. Una vez que se ha determinado dicha diferencia basta con saber los parámetros internos y externos del arreglo experimental para encontrar los parámetros geométricos que determinarán la posición de cada punto objeto. La mayor complicación de este sistema es la calibración, pues al contar con aberraciones tanto por parte de la cámara como del sistema de proyección, este proceso implica encontrar los parámetros para los sistemas ópticos de ambos dispositivos.

La idea básica de las técnicas de proyección de franjas es cuantificar y demodular las franjas para obtener la fase relacionada, por lo que las técnicas de proyección de franjas de luz estructurada requieren además de un proceso adicional de desenvolvimiento de fase, pues para obtener dicha fase se utilizan funciones *arctan* y, por tanto, la respuesta obtenida se encuentra en el intervalo $[-\pi, \pi]$.

El registro de superficies se considera como un problema de búsqueda y optimización, pues se busca minimizar una función dada en un espacio de búsqueda determinado. Se presentaron algunos métodos convencionales de optimización con el fin de ilustrar las limitaciones de estos en comparación con técnicas de computación evolutiva. En la presente tesis se plantea una alternativa a las actuales soluciones de registro de nubes de puntos. Se considera uno de los métodos más utilizados: el algoritmo ICP. Sin embargo, el ICP tiene ciertas limitantes en cuanto a la solución del problema cuando no se cuenta con una pre-alineación de las nubes, pues debido a su naturaleza, presenta una tendencia a converger a mínimos locales. Como alternativa a este enfoque se propuso realizar el alineamiento previo de las superficies por medio de un algoritmo genético. Los algoritmos genéticos presentan la particularidad de que pueden salir de óptimos locales, esto a través de los parámetros de cruce y mutación, siendo este último el que tiene una mayor contribución al poder modificar por completo la información contenida en un individuo, dando lugar a una mayor diversidad.

Para este trabajo se propone una función de aptitud que considera, al igual que el algoritmo ICP, las distancias mínimas promedio entre las superficies a alinear, con la diferencia de que en el enfoque propuesto no se reemplazan ni eliminan puntos. El hecho de conservar los puntos en ambas superficies permite una mayor velocidad de ejecución del algoritmo, pues se aprovecha la facultad de MATLAB para llevar a cabo operaciones vectoriales en una sola línea. Otro aspecto importante para la función objetivo propuesta es la consideración del área de traslape entre ambas superficies. El parámetro de traslape permite dar un peor desempeño a posibles soluciones que no cumplen con una razón mínima de superposición. Además, se considera el concepto de

alineamiento exitoso, que tiene lugar cuando la distancia mínima entre dos puntos es menor que una tolerancia determinada por el algoritmo.

Se obtuvieron buenos resultados y una excelente convergencia para el alineamiento de superficies que no presentan una razón de simetría muy grande entre sí. Sin embargo, para aquellas que no cumplen con este requisito, la respuesta tiende a ser la superposición total entre ambas nubes de puntos, razón que limita la aplicabilidad de la técnica propuesta.

6.1 Trabajos a futuro

Como primera mejora al algoritmo, se propone agregar la técnica de determinación de flujo óptico con el fin de tener una mejor parametrización del rango en que operan las variables del algoritmo y así lograr acelerar su convergencia.

Otra posible mejora es la integración de redes neuronales para el aprendizaje de discriminantes en cuanto a superficies simétricas. Una parametrización adecuada de la razón de superposición entre nubes alineadas puede ayudar a alcanzar una mejor solución en estos casos.

El área de inteligencia artificial aplicada a tareas de metrología óptica está cobrando un gran auge, sobre todo en las técnicas de demodulación de franjas. Por ejemplo, una posible aplicación consiste en aplicar programación genética para encontrar funciones que demodulen exitosamente franjas cerradas en aplicaciones de control de calidad.

Una línea de investigación propuesta para el presente trabajo es la de adquisición y registro de superficies macroscópicas. Este nuevo trabajo debe incluir además factores de escalamiento como parte de la codificación del cromosoma y un nuevo método de muestreo, pues se espera que las nubes de puntos adquiridas contengan cantidades de puntos en el rango de millones, con el fin de obtener una buena resolución y una reconstrucción fiel al modelo original.

Apéndice A

Calibración de cámaras

En este apéndice se desarrollan algunos de los métodos descritos en el capítulo 3. La primera sección incluye la proyección de modelos de proyección central y paralela, en esta se describen las transformaciones del espacio real \mathcal{R}^3 al espacio proyectado \mathcal{P}^3 .

En la segunda parte se describe de forma matemática el método de calibración por Transformaciones Lineales Directas o DLT, por sus siglas en inglés. Este modelo cuenta con la ventaja de que pueden obtenerse los parámetros intrínsecos y extrínsecos de la cámara resolviendo únicamente ecuaciones lineales.

A.1 Conversión de un espacio real a uno proyectado

Como se mencionó en el capítulo 3, en el modelo idealizado de una cámara sin distorsión o pinhole pueden existir dos tipos de proyección: la proyección central o de perspectiva (figura A.1a) y la proyección paralela, también conocida como ortogonal (figura A.1b).

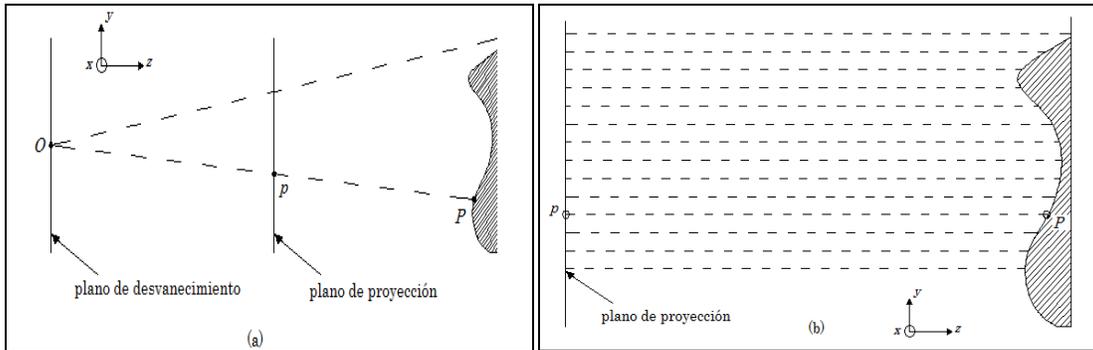


Figura A.1 Proyecciones de un punto P a un punto p (a) central y; (b) paralela

Si el sistema coordenado del espacio XYZ está definido como centrado en la cámara, es decir, el punto focal corresponde al centro de proyección y el eje óptico al eje Z , entonces las ecuaciones de proyección están dadas por

$$x = \frac{f \cdot X}{Z} \quad y = \frac{f \cdot Y}{Z} \quad (A.1)$$

Si el origen del sistema coordenado está posicionado en el plano imagen (sistema centrado en la imagen), entonces las ecuaciones de proyección son

$$x = \frac{f \cdot X}{f + Z} \quad y = \frac{f \cdot Y}{f + Z} \quad (A.2)$$

El espacio real \mathcal{R}^3 , es decir, el espacio XYZ se describe en coordenadas homogéneas como el espacio proyectado \mathcal{P}^3 . Las coordenadas homogéneas (t, u, v, w) de los puntos proyectados en este espacio se interpretan como sigue

Caso en que $w \neq 0$: El punto proyectado (t, u, v, w) se identifica por

$$(X, Y, Z) = (t/w, u/w, v/w) \quad (A.3)$$

del espacio XYZ . Los valores $X = t/w, Y = u/w$, y $Z = v/w$ son las coordenadas no homogéneas del punto proyectado (t, u, v, w) .

Caso en que $w = 0$: Un punto proyectado $(t, u, v, 0)$ se identifica por un punto en infinito del espacio XYZ . El conjunto de puntos $(t, u, v, 0)$ del espacio proyectado forma el plano en infinito con la ecuación de plano $w = 0$.

Si un punto del espacio XYZ debe expresarse en coordenadas proyectadas entonces puede elegirse cualquier valor $w \neq 0$. Un plano $Z = pX + qY + r$ por ejemplo, tiene la proyección

$$v = pt + qu + rw \quad (A.4)$$

en coordenadas homogéneas. Para $w = 0$ se obtiene un plano en el infinito. Las transformaciones de \mathcal{P}^3 son lineales y se conocen como *colineaciones* [Reinhard *et al.*, 1996], estas se encuentran dadas por

$$(t', u', v', w')^T = \mathbf{A} \cdot (t, u, v, w)^T \quad (A.5)$$

donde \mathbf{A} es una matriz de 4×4 no singular de números reales. La más importante ventaja de usar coordenadas proyectivas homogéneas es que las transformaciones pueden describirse de manera uniforme por tales matrices.

Para un sistema centrado en la cámara y de proyección central, la proyección del espacio real \mathcal{R}^3 al plano imagen \mathcal{R}^2 es una de las transformaciones antes mencionadas. Si se asume que la constante de cámara f es la distancia entre el plano de proyección y el plano de desvanecimiento, entonces un punto en la escena $(X, Y, Z) = (t/w, u/w, v/w)$ es proyectado en el punto imagen (x, y, f) de acuerdo a la ecuación general de proyección homogénea

$$(t', u', v', w')^T = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{f} & 0 \end{pmatrix} \cdot (t, u, v, w)^T \quad (A.6)$$

con

$$x = \frac{t'}{w'} = \frac{t}{v/f} = \frac{f \cdot X}{Z}, \quad y = \frac{u'}{w'} = \frac{u}{v/f} = \frac{f \cdot Y}{Z}, \quad \frac{v'}{w'} = f \quad (\text{A.7})$$

En ocasiones es más apropiado utilizar proyección paralela como aproximación a la escena observada. Para este caso todo punto $\mathbf{P} = (X, Y, Z)$ del espacio \mathcal{R}^3 se proyecta en un punto $\mathbf{p} = (x, y)$ en el plano imagen, con las ecuaciones de proyección $x = X$ y $y = Y$. En este caso, el centro de proyección se transforma en un punto en el infinito como se muestra en la figura A.1(a). Para la ecuación general homogénea correspondiente a la proyección paralela

$$(t', u', v', w')^T = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & f \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot (t, u, v, w)^T \quad (\text{A.8})$$

en coordenadas homogéneas, un punto en la escena $(X, Y, Z) = (t/w, u/w, v/w)$ es proyectado como punto imagen (x, y, f) por medio de

$$x = \frac{t'}{w'} = \frac{t}{w} = X, \quad y = \frac{u'}{w'} = \frac{u}{w} = Y, \quad \frac{v'}{w'} = \frac{f \cdot w}{w} = f \quad (\text{A.9})$$

Sean x_b y y_b las coordenadas de los píxeles en la imagen digital en donde estos píxeles pueden considerarse el resultado final del proceso de adquisición. Los puntos de objeto $\mathbf{P} = (X_w, Y_w, Z_w)$ en coordenadas mundo son proyectados en los puntos de la imagen digital $\mathbf{p} = (x_b, y_b)$ especificando los índices de fila y columna durante la adquisición (figura A.2).

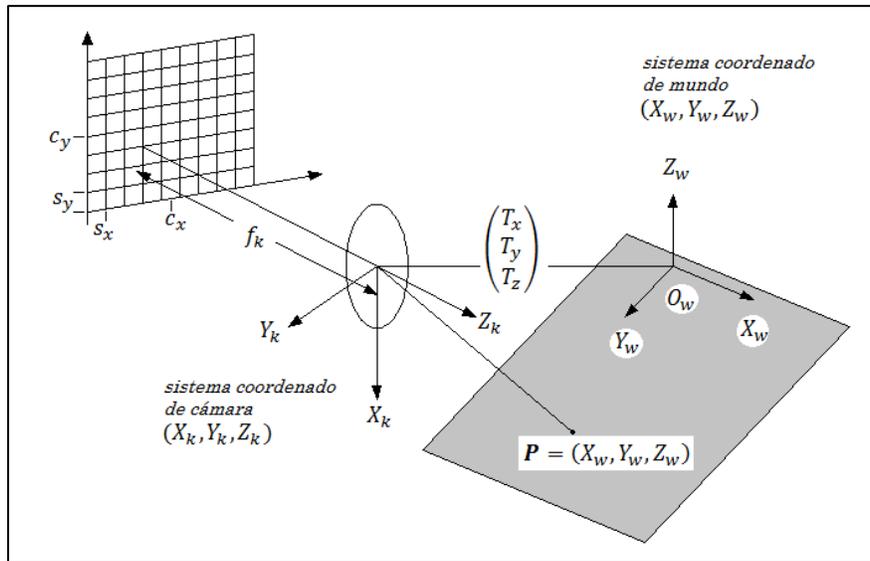


Figura A.2 Coordenadas de mundo, de cámara y de imagen

El mapeo descrito puede dividirse en varias transformaciones de coordenadas y otros mapeos, y está descrito por la siguiente transformación lineal [Klette *et al.*, 1998]

$$\begin{pmatrix} x_b - c_x \\ y_b - c_y \\ -f_k \end{pmatrix} = \begin{pmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{pmatrix} \begin{pmatrix} X_w - X_o \\ Y_w - Y_o \\ Z_w - Z_o \end{pmatrix} \quad (\text{A.10})$$

En esta ecuación

- (x_b, y_b) denota las coordenadas en la imagen digital de los puntos proyectados (X_w, Y_w, Z_w) ,
- (c_x, c_y) son las coordenadas de la intersección del eje óptico con el plano imagen,
- f_k es la distancia entre el centro de la lente y el plano imagen (distancia focal efectiva),
- r_i son los coeficientes de la matriz de rotación $\mathbf{R} = \mathbf{R}_x \cdot \mathbf{R}_y \cdot \mathbf{R}_z$ de la transformación Euclidea
- (X_w, Y_w, Z_w) son las coordenadas de mundo del punto de escena proyectado, y
- (X_o, Y_o, Z_o) son las coordenadas del centro de proyección en coordenadas mundo.

La transformación de un punto (X_w, Y_w, Z_w) en coordenadas mundo a un punto en coordenadas centradas en la cámara (X_k, Y_k, Z_k) está descrita por la rotación \mathbf{R} y la translación \mathbf{T} ,

$$\begin{pmatrix} X_k \\ Y_k \\ Z_k \end{pmatrix} = \mathbf{R} \cdot \begin{pmatrix} X_w \\ Y_w \\ Z_w \end{pmatrix} + \mathbf{T}, \quad \text{con } \mathbf{T} = \begin{pmatrix} T_x \\ T_y \\ T_z \end{pmatrix} = \begin{pmatrix} -X_o \\ -Y_o \\ -Z_o \end{pmatrix} \quad (\text{A.11})$$

Las transformaciones euclídeas correspondientes para \mathbf{R} se describieron en la sección 3.2. Para determinar la relación entre coordenadas de mundo y de cámara deben de resolverse los parámetros de la matriz \mathbf{R} y del vector \mathbf{T} .

Una alternativa al método descrito anteriormente es la calibración por Transformaciones Lineales Directas. En este, hace falta contar con patrones con posiciones conocidas y, partir de dichas posiciones, resolver los parámetros de la cámara a través de ecuaciones lineales.

A.2 Calibración por Transformaciones Lineales Directas

Una medición directa tanto de parámetros intrínsecos como extrínsecos puede obtenerse de manera indirecta utilizando una técnica de calibración adecuada. El objetivo es obtener la correspondencia entre el modelo de cámara utilizado (parametrizado por los valores observados o calculados) y la imagen adquirida por la misma.

La técnica de Transformaciones Lineales Directas (DLT, por sus siglas en inglés) se basa en el conocimiento de algunas coordenadas en el espacio de puntos geoméricamente configurados que se representan físicamente por marcas en algún objeto de calibración.

Bajo ciertas condiciones, el cálculo automatizado de las localizaciones de las marcas de calibración es una tarea de fácil solución en el análisis de imágenes, asumiendo que las condiciones de iluminación y la coloración de las caras del objeto de calibración y de las marcas permite la fácil detección de las mismas. Entre las marcas de calibración más comunes se

encuentran círculos rellenos y localizaciones de precisión a nivel de sub-píxel (puntos medios de las marcas) basadas en el cálculo de centroides.

Para una determinación precisa de los parámetros de calibración resulta beneficioso que las marcas se encuentren igualmente espaciadas en la escena y que la imagen tomada digitalmente sea tan buena como sea posible. Por lo tanto, el tamaño del objeto de calibración debe mantener una relación con la distancia de trabajo de la cámara. Si por ejemplo, quieren obtenerse imágenes a una distancia de 2 – 3m, entonces el objeto de calibración debe elegirse de tal manera que este pueda proyectarse en toda la imagen digital colocada a la distancia mencionada.

El uso de la técnica de DLT es una posibilidad para especificar los parámetros en cuestión. Tiene la ventaja de que deben de resolverse casi únicamente ecuaciones lineales [Abdel-Aziz y Karara, 1971].

Se puede desarrollar la ecuación A.10 para obtener las ecuaciones lineales para $x_b - c_x$ y $y_b - c_y$:

$$\begin{aligned} x_b - c_x &= -f_k \frac{r_1(X_w - X_o) + r_2(Y_w - Y_o) + r_3(Z_w - Z_o)}{r_7(X_w - X_o) + r_8(Y_w - Y_o) + r_9(Z_w - Z_o)} \\ y_b - c_y &= -f_k \frac{r_4(X_w - X_o) + r_5(Y_w - Y_o) + r_6(Z_w - Z_o)}{r_7(X_w - X_o) + r_8(Y_w - Y_o) + r_9(Z_w - Z_o)} \end{aligned} \quad (A.12)$$

Estas ecuaciones pueden reescribirse en la forma

$$\begin{pmatrix} X_w & Y_w & Z_w & 1 & 0 & 0 & 0 & 0 & x_b X_w & x_b Y_w & x_b Z_w \\ 0 & 0 & 0 & 0 & X_w & Y_w & Z_w & 1 & x_b X_w & x_b Y_w & x_b Z_w \end{pmatrix} \begin{pmatrix} L_1 \\ L_2 \\ \vdots \\ L_{11} \end{pmatrix} = \begin{pmatrix} X_w \\ Y_w \end{pmatrix} \quad (A.13)$$

en donde

$$\begin{aligned} L_1 &= \frac{c_x r_7 - d_x r_1}{D}, & L_2 &= \frac{c_x r_8 - d_x r_2}{D}, & L_3 &= \frac{c_x r_9 - d_x r_3}{D} \\ L_4 &= -L_1 X_o - L_2 Y_o - L_3 Z_o \\ L_5 &= \frac{c_y r_7 - d_y r_4}{D}, & L_6 &= \frac{c_y r_8 - d_y r_5}{D}, & L_7 &= \frac{c_y r_9 - d_y r_6}{D} \\ L_8 &= -L_5 X_o - L_6 Y_o - L_7 Z_o \\ L_9 &= \frac{r_7}{D}, & L_{10} &= \frac{r_8}{D}, & L_{11} &= \frac{r_9}{D} \end{aligned} \quad (A.14)$$

En las ecuaciones A.14, d_x y d_y representan los factores de conversión para los ejes de la cámara X_k y Y_k , respectivamente desde un sistema de coordenadas reales a uno en píxeles. El factor D está dado por

$$D = -(X_0 r_7 + Y_0 r_8 + Z_0 r_8) \quad (A.15)$$

Los parámetros L_1 - L_{11} se conocen como parámetros de transformación lineal directa (DLT). Para una solución no ambigua al menos seis puntos de calibración (X_w, Y_w, Z_w) y sus correspondientes coordenadas imagen de los centros de las marcas de calibración se requieren para resolver el sistema con respecto a los once parámetros de transformación L_1, \dots, L_{11} . En el caso de seis puntos se tienen doce ecuaciones lineales, lo que conduce a un sistema sobre determinado.

Los parámetros de transformación L_n adquiridos contienen los parámetros intrínsecos y extrínsecos. El punto principal está dado por

$$c_x = \frac{L_1 L_9 + L_2 L_{10} + L_3 L_{11}}{L_9^2 + L_{10}^2 + L_{11}^2} \quad y \quad c_y = \frac{L_5 L_9 + L_6 L_{10} + L_7 L_{11}}{L_9^2 + L_{10}^2 + L_{11}^2} \quad (A.16)$$

La distancia focal efectiva f_k puede calcularse con respecto a sus componentes

$$f_{kx}^2 = -c_x^2 + \frac{L_1^2 + L_2^2 + L_3^2}{(L_9^2 + L_{10}^2 + L_{11}^2)^2} \quad y \quad f_{ky}^2 = -c_y^2 + \frac{L_5^2 + L_6^2 + L_7^2}{(L_9^2 + L_{10}^2 + L_{11}^2)^2} \quad (A.17)$$

con respecto a ambos ejes del sistema coordenado de la imagen, respectivamente. Puede obtenerse

$$f_k = \frac{f_{kx} + f_{ky}}{2} \quad (A.18)$$

Los parámetros extrínsecos de la cámara también pueden determinarse con L_1, \dots, L_{11} . El cálculo de los ángulos en las matrices de transformación \mathbf{R}_x , \mathbf{R}_y y \mathbf{R}_z es posible en dos pasos: en el primero, se calculan los coeficientes de la matriz de rotación general $\mathbf{R} = r_i$, $i = 1, \dots, 9$ basándose en las ecuaciones para $L_1 - L_3$, $L_5 - L_7$ y $L_9 - L_{11}$. En el segundo paso, se determinan los ángulos individuales de recorrido, inclinación y rodamiento a partir de los valores de los coeficientes r_i de esta matriz \mathbf{R} .

A continuación, las coordenadas del centro de proyección (X_0, Y_0, Z_0) pueden determinarse de las ecuaciones 3.17 y 3.18. Al resolver estas ecuaciones para X_0 , Y_0 y Z_0 , se obtienen dos ecuaciones con tres incógnitas. Por lo tanto, el problema puede resolverse al analizar dos puntos de calibración y sus marcas proyectadas. Se obtiene que

$$\begin{pmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{pmatrix} \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix} = \begin{pmatrix} A \\ B \end{pmatrix} \quad (A.19)$$

de donde

$$\begin{aligned} a_1 &= (c_x - x_b)r_7 - f_k r_1, & b_1 &= (c_y - y_b)r_7 - f_k r_4, \\ a_2 &= (c_x - x_b)r_8 - f_k r_2, & b_2 &= (c_y - y_b)r_8 - f_k r_5, \end{aligned}$$

$$a_3 = (c_x - x_b)r_9 - f_k r_3, \quad b_3 = (c_y - y_b)r_9 - f_k r_6.$$

$$A = X_w a_1 + Y_w a_2 + Z_w a_3,$$

$$B = X_w b_1 + Y_w b_2 + Z_w b_3 \tag{A.20}$$

Los parámetros necesarios para la transformación lineal descrita por la ecuación A.6 pueden determinarse hasta que se han resuelto todas las ecuaciones desde A. 14 hasta A. 20.

Referencias

- [1] Abdel-Aziz, Y.I., Kahara, H.M., '*Direct linear transformation into object space coordinates in close-range photogrammetry*', Proc. ASP Symposium on Close-Range Photogrammetry, Urbana, Illinois, USA, 1971, pp. 1-18.
- [2] Haralick, R.M., Shapiro, L.G., '*Computer and Robot Vision, Volume II*', Addison-Wesley, 1993.
- [3] Klette, R., Schlüns, K., Koschan, A., '*Computer Vision: Three-Dimensional Data from Images*', Springer, 1996.

Apéndice B

Plan de negocios

B.1 Introducción

El presente plan de negocios se realiza con el objetivo primordial, de contar con una herramienta de sofisticación empresarial que demuestre la factibilidad a nivel industria, de la investigación e innovación tecnológica del trabajo de tesis desarrollado.

El procedimiento utilizado para alcanzar este objetivo se desarrollo bajo bases reales de investigación de mercado, trabajo de campo, análisis de la situación actual e interpretación de los datos recopilados.

B.1.1 Misión

Ser una compañía que otorga servicios de metrología óptica, en el área de visión tridimensional de objetos, a través del desarrollo de sistemas optomecatrónicos para digitalizado e ingeniería inversa. Muchos de estos servicios enfocados en la innovación del calzado, el estudio de antropometría humana, de sistemas biomecánicos y necesidades de ingeniería.

B.1.2 Visión

Ser la mejor empresa en México con la capacidad de venta de equipos y servicios de digitalizado y metrología óptica, así como el desarrollo de sistemas optomecatrónicos cuyo objetivo es la innovación tecnológica, aplicados en la tecnología del calzado, la investigación de antropometría humana, de sistemas biomecánicos y de soluciones de ingeniería a la industria.

B.1.3 Producto / Idea

- Sistemas ópticos móviles (Scanner láser) para la digitalización de la antropometría del pie para desarrollo de calzado ortopédico y de confort. Esto permite la visión tridimensional de la forma del pie en un ordenador, ya sea para el estudio podológico del pie de cualquier persona de cualquier edad o el diseño de calzado a la medida.
- Escáneres basados en técnicas de luz estructurada y de visión estéreo para el análisis y adquisición de piezas para ensamble, control de calidad y prototipado.
- Dispositivos robustos que pueden ser transportados de manera fácil.
- Mantenimiento económico y de fácil uso.

B.2 Resumen ejecutivo

Con la finalidad de facilitar el entendimiento rápido del presente trabajo se presenta el siguiente resumen ejecutivo.

Compañía: Opto-Mechatronics Corporation es una compañía que otorga servicios de metrología óptica, a través del desarrollo de sistemas opto-mecatrónicos para digitalizado e ingeniería inversa. Muchos de estos servicios están enfocados en la innovación del calzado, el estudio de antropometría humana, de sistemas biomecánicos, necesidades de ingeniería y más. Las oficinas centrales están ubicadas en Av. Juan Nepomuceno 804 Int. 203, Col. Valle del Campestre en León Guanajuato, México.

Estado Actual: Opto-Mechatronics Corporation está en la etapa de investigación, desarrollo del sistema de la página de internet y del sistema de logística. Cuenta con un plan definido de negocios, proyecciones financieras a 4 años y de un Director General. Se está en negociaciones con inversionistas potenciales en la rama del calzado, así como también en la búsqueda del financiamiento gubernamental para la investigación científica y desarrollo tecnológico.

Servicio: Opto-Mechatronics Corporation ofrece actualmente el servicio de digitalización de la antropometría del pie para desarrollo de calzado ortopédico y de confort enfocado al mercado de la innovación tecnológica en el calzado, ofreciendo calidad, solución de problemas integrales a la industria del calzado e innovación tecnológica. Los ingresos de este corporativo son dos, 1. El servicio de digitalización de objetos y mediciones ópticas, y 2. El desarrollo de sistemas opto-mecatrónicos con la capacidad de solucionar problemas industriales.

Descripción del Mercado: Para poner en marcha el proyecto de servicio en digitalización de la antropometría del pie para desarrollo de calzado ortopédico y de confort, se enfoca al mercado de la innovación tecnológica en el calzado; se seleccionó la ciudad de León, Guanajuato, entre otras características por su desarrollo industrial en esta rama y concentración de fabricas de calzado en el área a nivel nacional. Además, dado su desarrollo industrial y especializació, se consideran otras ciudades del estado de Guanajuato para servicios de digitalización de piezas y venta de escáneres ópticos.

B.3 Análisis FODA

El análisis FODA se refiere a identificar las fortalezas, oportunidades, debilidades y amenazas para conocer la situación presente de la empresa. Las amenazas y oportunidades representan factores externos a la organización, mientras que las fortalezas y debilidades se identifican en la estructura interna de la misma.

- Fortalezas

Conocimientos técnicos y tecnológicos para la investigación y desarrollo de nuevos productos.

Contactos con centros de investigación para pruebas y desarrollo.

- Oportunidades.

Pocas empresas en la región bajío en el país.

Sinergia de varias tecnologías.

Venta de equipos o de servicio de digitalización.

- Debilidades

Se cuenta con pocos recursos para arrancar la empresa.

No se conoce el nivel de aceptación de las empresas locales hacia nuevas tecnologías.

- Amenazas.

Costos de operación elevadas en el comienzo de la empresa.

Propuestas tradicionales en el desarrollo del calzado.

Conocimiento mínimo del producto y servicio.

B.4 Descripción de la empresa

En este apartado se incluye la información básica del contexto y el concepto del negocio. Se explica, de manera general qué será y que hará la empresa.

- Clientes potenciales.

El concepto del proyecto es enfocado al mercado del diseño del calzado, de acuerdo a características de empresas que tengan como objetivo primordial la innovación en la tecnología del calzado. En este rubro se puede mencionar marcas como Brantano, Coqueta y Audaz, entre otras.

Pequeñas y medianas empresas dedicadas en la fabricación de suelas, así como también podólogos con el interés del estudio clínico del pie.

- Medio de distribución.

Por medio de una cartera de clientes, realizando ofertas y todo lugar donde se pueda realizar demostraciones del trabajo como: cursos, talleres o conferencias.

Venta directa de un equipo de ventas.

- Modo de operación.

Desarrollo de sistemas de visión tridimensionales a partir de metrología óptica y algoritmos computacionales, además de la integración de equipo y herramienta de visión computacional.

- Estructura de la organización.

Tres investigadores dedicados al desarrollo de dispositivos de visión artificial, metrología óptica e ingeniería electrónica. Uno de estos investigadores con el cargo de gerente, y los otros dos como comité ejecutivo.

Un equipo de trabajo de ingenieros altamente calificados, de aproximadamente tres personas, siendo uno de ellos el enlace con el comité ejecutivo.

- PLAN DE ACCION:

- Comprobar los resultados de las pruebas de la simulación en un prototipo físico.
- Realizar el plan de negocio.
- Realizar un estudio a fondo de mercado.
- Establecer un equipo de trabajo dedicado a la implementación del sistema de direccionamiento y de visión del robot.
- Capitalizar el proyecto por medio de apoyos otorgados para profesionistas emprendedores.

B.5 Análisis de mercado

En esta sección se presenta la definición del mercado objetivo, así como su situación a partir del último reporte disponible. Se presentan además los principales competidores de acuerdo a su giro y a su ubicación geográfica. Finalmente, se propone un plan de marketing para captar el mayor porcentaje posible de clientes en cada zona.

B.5.1 Perfil de mercado

Los clientes potenciales más importantes son todas aquellas empresas manufactureras en la zona del Bajío del estado de Guanajuato que cuenten con departamentos de ingeniería inversa, prototipado, control de calidad y modelado de piezas para su posterior producción. En esta categoría se consideran principalmente las industrias de manufactura de textiles, metal-mecánica y de plásticos. Adicionalmente, se consideran áreas de biomédica en donde se requiera la cuantización de las medidas antropométricas para pacientes particulares.

Se considera que se tiene un mercado amplio, pues los clusters de equipo médico, óptico y de medición, además del de maquinaria y equipo, son considerados como candidatos con potencial futuro debido a su dinamismo en el mercado mundial y por la disponibilidad de capital intelectual en el estado de Guanajuato [1].

En 2004, la industria manufacturera contribuyó con el 20.2% del PIB estatal, siendo este de \$38'802,028. Además, esta división presentó un crecimiento promedio de 5.6% en el periodo comprendido entre 1995-2001. Ese mismo año (2004), el estado de Guanajuato contaba con 4,932 establecimientos dedicados a la fabricación de productos de cuero, piel y materiales sucedáneos, excepto prendas de vestir; 2,809 empresas dedicadas a la fabricación de productos metálicos y 1,173 consultorios generales y de especialidades [2]. La participación en las divisiones económicas para la entidad se muestra en la tabla B.1.

Tabla B.1
Participación al PIB para el estado de Guanajuato, según división económica

<i>División económica</i>	<i>% Participación</i>
Manufacturera	20.2
Comercial	20
Servicios	32
Transporte	13.15
Otras	14.65

Fuente: Elaboración propia a partir de [5]

Las contribuciones al PIB de la industria manufacturera de las áreas de interés y su segmentación de acuerdo al tamaño de la empresa se muestran en la tabla B.2. De acuerdo con estadísticas del CICEG, en 2004 se contaban con 2,848 empresas en el sector cuero-calzado-proveeduría, con una participación del 9.16% del PIB correspondiente al sector manufacturero, de donde el 19.78% está conformado por micro empresas, 29.94% por pequeñas empresas, 21.59% por medianas y 29.69% por grandes empresas [3], este subsector tuvo una participación en el sector manufacturero ese año de un 26.36%, del cual, la producción de calzado representa un 9.17%.

Tabla B.2
Participación al PIB de la industria manufacturera

<i>Sector</i>	<i>Participación %</i>	<i>Micro empresa %</i>	<i>Pequeña empresa %</i>	<i>Mediana empresa %</i>	<i>Gran empresa %</i>
Producción de calzado.	9.17	19.78	29.94	21.59	29.69
Productos metálicos, maquinaria y equipo.	40.6	1.65	2.29	1.63	94.43
Producción de sustancias químicas y artículos de plástico o hule.	22.26	6.41	11.35	6.65	75.59

Fuente: Elaboración propia a partir de [5]

En el estado de Guanajuato, la industria metal-mecánica se encuentra centrada principalmente en: León, Celaya, Salamanca, Irapuato y Silao [4]. Este subsector tuvo en Guanajuato durante el año 2004 una participación del 0.94% en la industria metálica básica y de 40.56% en la manufactura de productos metálicos, maquinaria y equipo. De este último porcentaje, el 1.65% corresponde a micro empresas, 2.29% a pequeñas empresas y 1.63% y 94.43% a medianas y grandes empresas, respectivamente [5].

El sector de producción de sustancias químicas y artículos de plástico o hule presentó una participación del 22.6% en 2004, en donde a la rama de producción de artículos de plástico o hule corresponde un 1.32%: el 15.99% de micro empresas, 54.69% correspondió a pequeñas empresas, 26.29% a medianas empresas y 3.03% a grandes empresas.

Tabla B.3
Producto bruto de subsectores manufactureros por municipio

	<i>León</i>	<i>Celaya</i>	<i>Irapuato</i>	<i>Salamanca</i>	<i>Silao</i>
Alimentos, bebidas y tabaco.	11%	40%	70%	8%	2%
Textiles, prendas de vestir e industria de cuero.	65%	1%	11%	0%	0%
Químicos, derivados del petróleo, caucho y plástico.	12%	15%	7%	92%	0%
Productos metálicos, maquinaria y equipo.	7%	40%	6%	0%	98%

Fuente: Elaboración propia a partir de [5]

Las áreas de enfoque para varios municipios dependerán de la especialización que estos tengan en cada sector (tabla B.3). El municipio de León genera un 65% de su PIB gracias a la industria de textiles, prendas de vestir e industria del cuero, Celaya genera 40% por medio de la fabricación de productos metálicos, maquinaria y equipo, mientras que Silao obtiene el 98% de su PIB de este mismo sector; Irapuato, por su lado, genera 11% gracias a los textiles y 6% a la manufactura de productos metálicos. Finalmente, Salamanca genera el 92% de su PIB gracias a la industria de químicos y derivados de petróleo, caucho y plástico.

Basándose en los datos obtenidos en el perfil de mercado obtenido, pueden hacerse las siguientes conclusiones:

- En el municipio de León conviene enfocarse a la industria manufacturera de calzado. En este rubro no existe gran distinción entre las participaciones hechas por tamaño de empresa, así que todas son clientes potenciales.
- En el caso de Celaya, debe de enfocarse a buscar clientes en el subsector de manufactura de productos metálicos, maquinaria y equipo, esencialmente a las grandes empresas.

- Para el municipio de Irapuato, no es conveniente buscar una concentración en cuanto a la oferta de servicios, pues la mayor especialización de este se encuentra en la industria de alimentos, misma en que no se perciben clientes potenciales.
- La oferta en el municipio de Salamanca debe de ser limitada, pues aunque gran parte de su producción se debe a la producción de químicos, derivados del petróleo, caucho y plástico, sólo el 1.32% de la producción del estado pertenece a un subsector importante para el negocio planteado (producción de plástico/hule).
- Silao es una fuente importante de clientes, pues casi toda su producción está centrada en productos metálicos, maquinaria y equipo. Sin embargo, la mayor parte de esta participación corresponde a la industria automotriz (34.11% del 40.56% del sector) y es difícil captar estos clientes dado el largo proceso de aprobación por parte de los corporativos.

Hasta este punto se han identificado los posibles clientes en los municipios más significativos dentro del estado, sin embargo, se necesita también conocer a los competidores tanto directos como indirectos que operan en la región. La siguiente sección trata este punto.

B.5.2 Análisis de la competencia

Se consideran como competidores todos aquellos negocios o compañías dentro de la República que presten servicios de digitalización, venta de digitalizadores ya sea de contacto o sin él y venta de software de modelado geométrico.

Las principales empresas dentro de este rubro se encuentran localizadas en Puebla (Shlager de México), San Nicolás, N.L. (CAD Avshmeip S.A.), México D.F. (CIMCO), Querétaro, (3d MetriSoft) y Guadalajara: 3C Metrology, Apliatec, Soluciones en diseño y manufactura (SDM), Tecnologías computarizadas para manufactura (TCM) [6-13].

De las empresas mencionadas anteriormente, aquellas localizadas en Querétaro y Guadalajara son los competidores más cercanos a la región que se planea cubrir. En particular, la empresa Tecnologías computarizadas para manufactura es la única que expresa en su sitio web (<https://tcm.com.mx>) de manera explícita su disponibilidad de servicios para la ciudad de León, aunque, por otro lado, ésta se dedica exclusivamente a la venta de software y capacitación para el uso del mismo.

Otra categoría de competencia son aquellas empresas que comercian escáneres comerciales, como el GOM Atos (CAD Avshmeip S.A., CIMCO), los escáneres de las líneas Handy-Scan 3D (SDM) y escáneres de contacto (3C Metrology, 3D MetriSoft).

Finalmente, se encuentran las empresas que ofrecen servicios de digitalización (Shlager de México, Apliatec) y venta de software de modelado 3D (3C Metrology, Apliatec, CIMCO, Shlager de México, SDM y TCM).

Dado que se planea localizar la base de operaciones en el municipio de León, se cuenta con ventaja para proporcionar servicios dentro de esta ciudad, además de en Silao e Irapuato por su

cercanía. La ciudad de Celaya se sigue considerando un importante cliente a pesar de encontrarse cerca de Querétaro, en donde se encuentra un importante competidor (3d MetriSoft), aunque en este punto deberá de descartarse Salamanca, pues además de encontrarse muy cerca de Querétaro, no representa una fuente importante de clientes dado su giro de especialización.

Referencias

- [1] Puente, E., Herrera, F., '*Generación de nuevas alternativas de desarrollo para el estado (de Guanajuato)*', Observatorio Estratégico-Tecnológico ITESM-FEMSA, 2004 (Recuperado el 22 de septiembre de 2010).
- [2] Instituto Nacional de Estadística y Geografía (INEGI), '*Información estadística*', (Recuperado el 23 de septiembre de 2010).
- [3] Cámara de la Industria del Calzado del Estado de Guanajuato (CICEG), '*Situación de la industria del calzado en México*', 2009. Recuperado el 23 de septiembre de 2010. Disponible en <http://www.ciceg.org/estadisticas/situacion.pdf>.
- [4] Portal del municipio de Irapuato, '*Negocios – Industria*', 2009. Recuperado el 23 de septiembre de 2010. Disponible en <http://www.irapuato.gob.mx/Negocios/negociosindustria.php>.
- [5] Arriaga, R., Leyva, E., Estrada, J.L., 'Perfil y estructura industrial de Guanajuato y Querétaro: un análisis de la producción, el empleo y los salarios', *Análisis Económico* **44**: 135 – 189 (2005).
- [6] 3C Metrology, '*[[3C Metrology]]*'. Recuperado el 26 de septiembre de 2010. Disponible en <http://www.3cmetrology.com.mx/>.
- [7] 3D Metrisoft, '*3D MetriSoft – Laboratorio de Medición Acreditado, Calibración de Máquinas de Coordenadas*', Recuperado el 26 de septiembre de 2010. Disponible en <http://www.3d-metrisoft.com/>.
- [8] Apliatec, '': *Apliatec : Sistemas de Visión 3D* ':'. Recuperado el 26 de septiembre de 2010. <http://www.apliatec.com.mx/>.
- [9] CIMCO, '*Especialistas en CAD/CAM, DNC, Digitalización 3D, Ingeniería Inversa, Inspección y Control de Calidad*'. Recuperado el 27 de septiembre de 2010. Disponible en <http://www.apliatec.com.mx/>.
- [10] Shlager de México, '*Shlager de México Servicios Integrales de Metrología*'. Recuperado el 27 de septiembre de 2010. Disponible en <http://www.apliatec.com.mx/>.
- [11] '*Soluciones en Diseño y Manufactura*'. Recuperado el 27 de septiembre de 2010. Disponible en <http://sdm-net.com/v9x/>.
- [12] TCM, '*Tecnologías Computarizadas para Manufactura S.A. de C.V.*'. Recuperado el 27 de septiembre de 2010. Disponible en <http://sdm-net.com/v9x/>.
- [13] CAD Avshmeip, '*SolidWorks, Mastercam, Logopress3 y más. CAD AVSHMEIP*'. Recuperado el 27 de septiembre de 2010. Disponible en <http://cadavshmeip.com/>.

Carta de enlace con la industria



INSTITUTO NACIONAL DE ANTROPOLOGÍA E HISTORIA

MUSEO NACIONAL DE ANTROPOLOGÍA

Laboratorio de Conservación

Proco de la Reforma y Gandhi

Cd. Chapultepec Polanco

11500 México, D.F.

Tel: 55530090

México D.F., a 11 de mayo de 2010

A QUIEN CORRESPONDA:

Por este medio se hace constar que Diego Torres Armenta, estudiante de Optomecatrónica en el Centro de Investigaciones en Óptica, participó en el proceso técnico de reproducción de las piezas del ajuar funerario de K'inich Janaab' Pakal, efectuados en el Laboratorio de Conservación del Museo Nacional de Antropología del 28 de abril al 8 de mayo del presente año, como parte de la colaboración interdisciplinaria del programa de conservación y restauración "Proyecto Máscaras Funerarias" de la Coordinación Nacional de Museos y Exposiciones, INAH.

El equipo especializado del CIO a cargo de la reproducción de los objetos, lo constituyen el Dr. Ramón Rodríguez-Vera, el Dr. Francisco Cuevas de la Rosa, la Ing. Berenice Salazar y el Ing. Juan Antonio Rayas.

La presente constancia se extiende a petición del interesado y para los fines que al mismo convengan.

Atentamente,

Lic. Sofia Martínez del Campo Lanz
Coordinadora del "Proyecto Máscaras Funerarias"
CNME/INAH

c.c.p. Expediente.