INSTITUTO POLITÉCNICO NACIONAL

CENTRO DE INVESTIGACIÓN EN COMPUTACIÓN

RECONSTRUCCIÓN 3D MULTI-OCULAR

TESIS QUE PARA OBTENER EL GRADO DE
MAESTRO EN CIENCIAS DE LA COMPUTACIÓN

PRESENTA
ING. LUIS ALBERTO HORNA CARRANZA

DIRECTORES DE TESIS
DR. RICARDO BARRÓN FERNÁNDEZ
DR. JOSÉ GIOVANNI GUZMÁN LUGO

MÉXICO, D.F., DICIEMBRE 2010

Índice general

Índice de figuras

SIP-14

INSTITUTO POLITECNICO NACIONAL

SECRETARIA DE INVESTIGACIÓN Y POSGRADO

ACTA DE REVISIÓN DE TESIS

En la Ciudad de México, D.F. siendo las 14:00 horas del día 16 del mes de noviembre de 2010 se reunieron los miembros de la Comisión Revisora de Tesis designada por el Colegio de Profesores de Estudios de Posgrado e Investigación del:

Centro de Investigación en Computación

para examinar la tesis de grado titulada:

"RECONSTRUCCIÓN 3D MULTI-OCULAR"

HORNA	**CARRANZA**	**LUIS ALBERTO**
Apellido paterno	materno	nombre(s)

Con registro:

B	0	8	1	4	2	0

aspirante al grado de: **MAESTRÍA EN CIENCIAS DE LA COMPUTACIÓN**

Después de intercambiar opiniones los miembros de la Comisión manifestaron ***SU APROBACIÓN DE LA TESIS***, en virtud de que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.

LA COMISIÓN REVISORA

Presidente

Dr. Edgardo Manuel Felipe Riverón

Secretario

Dr. Marco Antonio Ramírez Salinas

Primer vocal
(Director de Tesis)

Dr. Ricardo Barrón Fernández

Segundo vocal
(Director de Tesis)

Dr. José Giovanni Guzmán Lugo

Tercer vocal

Dra. Nareli Cruz Cortés

Suplente

Dr. Rolando Quintero Téllez

EL PRESIDENTE DEL COLEGIO

Dr. Luis Alfonso Villa Vargas

INSTITUTO POLITECNICO NACIONAL
CENTRO DE INVESTIGACION
EN COMPUTACION
DIRECCION

INSTITUTO POLITECNICO NACIONAL

SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

CARTA CESIÓN DE DERECHOS

En la Ciudad de México el día 16 del mes de Noviembre del año 2010, el (la) que suscribe HORNA CARRANZA LUIS ALBERTO alumno (a) del Programa de Maestría en Ciencias de la Computación con número de registro B081420, manifiesta que es autor (a) intelectual del presente trabajo de Tesis bajo la dirección de Dr. Ricardo Barrón Fernández y co-dirección de Dr. José Giovanni Guzmán Lugo, cede los derechos del trabajo intitulado RECONSTRUCCIÓN 3D MULTI-OCULAR, al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo a la siguiente dirección miaufix@yahoo.com.mx. Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.

Luis Horna

Luis Alberto Horna Carranza

Resumen

En este trabajo de tesis se trata el problema de la reconstrucción 3D de un escenario a partir de dos imágenes de dicho escenario. El problema de la reconstrucción 3D es un clásico de la visión por computadora, dicho problema sigue siendo hasta hoy día un área importante de investigación tanto por los desafíos que esto plantea como por sus aplicaciones.

El proceso comúnmente usado para la reconstrucción de un escenario se centra en hallar el punto en el que dos regiones de las imágenes del escenario coinciden, a dicho punto se lo denomina disparidad. La principal aportación del presente trabajo de investigación se centra principalmente en el uso de memorias asociativas morfológicas para corregir el proceso de correspondencia, así como el uso de la optimización por mínimos cuadrados para el refinamiento y propagación de las medidas de disparidad.

Capítulo 1

Introducción

La reconstrucción 3D de objetos o ambientes, ya sea a partir de una o varias imágenes ha sido uno de los principales problemas que la visión por computadora ha intentado resolver, siendo la precisión de dicha reconstrucción el problema central con el que los algoritmos para dicha tarea deben lidiar. Dentro de las distintas técnicas para llevar a cabo dicha tarea se pueden encontrar:

- Forma a partir del sombreado (Shape From Shading, SFS)
- Estructura a partir del movimiento (Structure From Motion, SFM)
- Profundidad a partir del enfoque
- Visión Estereoscópica

Cada una de estas técnicas tienen sus fortalezas y debilidades, por ejemplo, SFS es ideal para extraer la estructura de imágenes tales como de rostros humanos u objetos estáticos en un ambiente controlado donde la iluminación es uniforme, sin embargo esto no es posible en situaciones de la vida real como por ejemplo, un auto en un estacionamiento donde la luz proviene de diferentes direcciones tales como el sol o reflexiones de otro autos. Técnicas más robustas como SFM y Visión Estereoscópica usan la correspondencia[1]de las imágenes para estimar la distancia y proveen excelentes resultados en situaciones reales como se muestra en [15] ,[19] ,[22].

[1]por correspondencia se entenderá como la relación existente entre dos puntos en el espacio

Uno de los problemas que pueden surgir en la reconstrucción 3D basado en la disparidad de imágenes es que los modelos recuperados algunas veces se encuentran incompletos, esto se debe a que no es completamente posible encontrar una correspondencia perfecta entre las imágenes procesadas, en algunas ocasiones situaciones tales como un agujero a través de una pared pueden desviar el proceso de reconstrucción como es mencionado en [15]. En este punto debemos preguntarnos qué es lo que falta en el proceso de reconstrucción, a lo cual se podría responder de manera parcial tomando en cuenta el hecho de que los algoritmos empleados para la reconstrucción 3D no tienen conocimiento previo de que es la profundidad.

1.1. Descripción del problema

La situación de la que se parte para llevar a cabo la reconstrucción 3D de un escenario será, contar con dos fotografías de un mismo escenario tomadas desde ángulos diferentes, con estas fotografías se deberá obtener un modelo 3D que represente el escenario en las fotografías

1.2. Justificación

La reconstrucción 3D de un ambiente dado u objeto es una tarea que tiene diversas aplicaciones, como por ejemplo: la medición de desperfectos en piezas producidas industrialmente, generación de mapas 3D de un ambiente, navegación de robots autónomos, generación de objetos 3D para video juegos por citar algunas.

1.3. Motivación

La mayoría de los algoritmos empleados para la reconstrucción basan su funcionamiento en modelar de manera matemática la transformación que tiene lugar al pasar de un mundo tridimensional a una imagen digital bidimensional, y ocupan dicho modelado para poder

llevar a cabo el proceso inverso. Sin embargo no es incluido conocimiento previo de que es la profundidad ni de la relación existente con una imagen 2D.

1.4. Objetivo general

El objetivo principal de la presente tesis es crear una metodología que realice una reconstrucción 3D empleando varias tomas fotográficas hechas desde diferentes posiciones en un ambiente dado, dicho modelo podría tener aplicaciones como: generación de mapas, paseos virtuales, navegación por citar algunos.

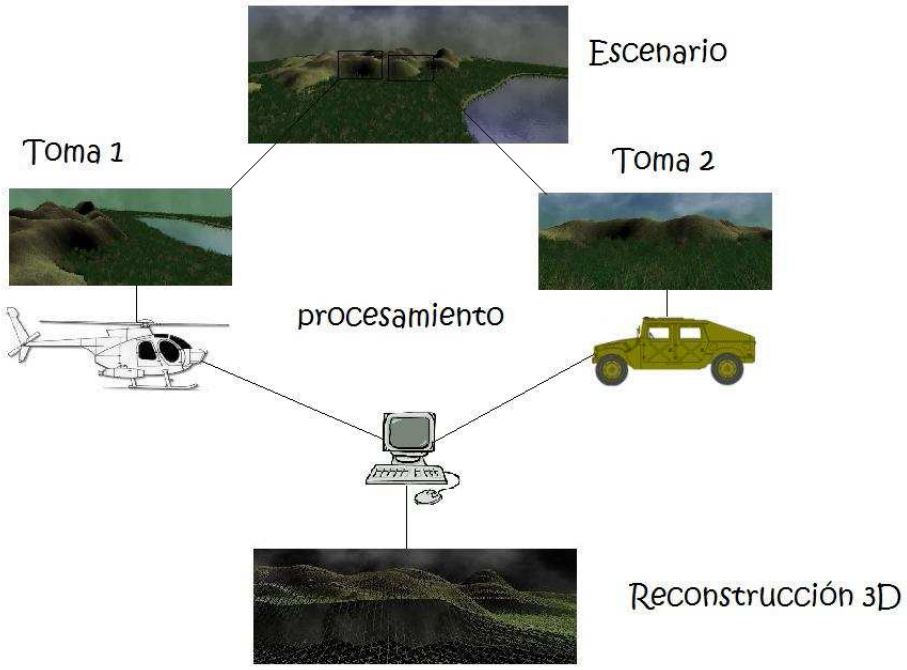


Figura 1.1: Esquema descriptivo de la aplicación

La principal aportación del presente trabajo será emplear un enfoque de reconocimiento de patrones para el aprendizaje de la re-

lación que existe entre los niveles de intensidad de los pixeles de una imagen digital y la profundidad a la que están los puntos de los objetos presentes en una escena dada.

Dicho enfoque se aparta de técnicas tradicionales que emplean la geometría para hallar la relación de dos o más imágenes para establecer la profundidad de objetos en una escena, cabe mencionar que el empleo de un enfoque de aprendizaje es novedoso como es mostrado en [20] y por lo tanto es gran de interés científico incursionar en un área poco explorada.

1.5. Objetivos particulares

Para poder llevar a cabo la reconstrucción 3D de un escenario se he identificado los siguientes problemas como los obstáculos principales para el desarrollo de la tesis actual, por tanto son objetivos particulares deberán ser cumplidos primero:

- Cómputo del mapa de profundidad a partir de dos imágenes usando métodos locales.
- Creación de un modelo 3D a partir del mapa de profundidad.
- Uso de técnicas de reconocimiento de patrones para mejorar el proceso de correspondencia.

1.6. Hipótesis

Para realizar la reconstrucción 3D se emplearán dos imágenes digitales, las cuales serán adquiridas mediante cámaras digitales colocadas en dos lugares distintos de la escena a reconstruir, dichas imágenes cumplirán con las siguientes restricciones:

i) Serán tomadas de modo tal que exista una correspondencia entre ellas.

ii) Serán tomadas durante el mismo periodo de tiempo.

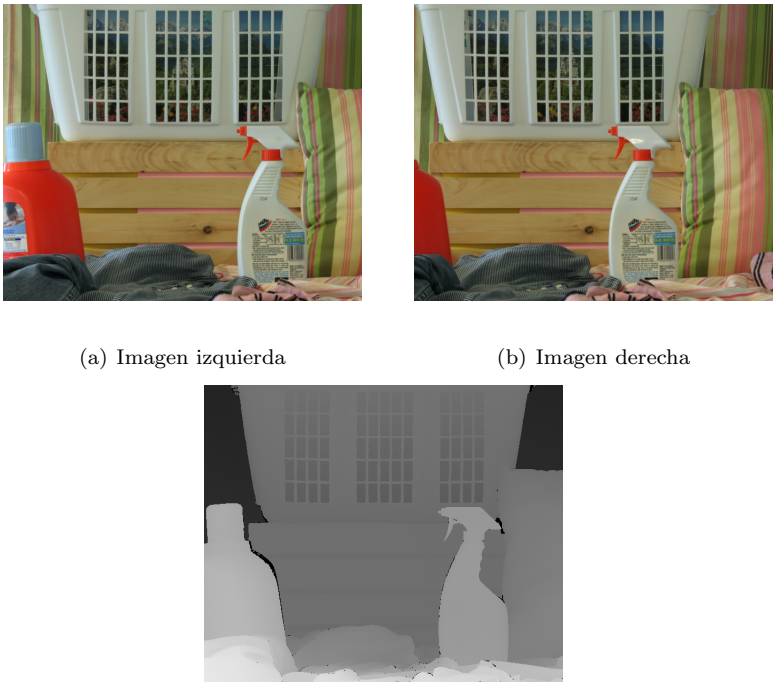
iii) Serán adquiridas por el mismo tipo de dispositivo.

La reconstrucción realizada corresponderá únicamente a la superficie que es representada en las imágenes empleadas, el resultado de esta reconstrucción será un modelo 3D que tendrá las siguientes características:

a) Estará formado por polígonos

b) Tendrá texturas, que correspondan a la escena reconstruida.

Las imágenes que se emplearán para realizar la reconstrucción 3D son las del conocido conjunto de imágenes de Middlebury [1][2][3][4],[65], dicho conjunto provee las vistas izquierda y derecha de un escenario, además incluye un mapa de disparidad de referencia.



(a) Imagen izquierda

(b) Imagen derecha

(c) Mapa de disparidad

Figura 1.2: Imágenes estereoscópicas y su correspondiente mapa de disparidad

1.7. Solución propuesta

El trabajo presentado actualmente intenta dar una nueva forma de plantear la reconstrucción 3D, en lugar intentar resolver de manera tradicional tratando de obtener la transformación inversa para pasar de una imagen bidimensional a un objeto tridimensional.

En esta tesis se trata de resolver el problema empleando un enfoque de reconocimiento de patrones que permita hallar la correspondencia existente entre una imagen y la profundidad correspondiente de los objetos presentes en una escena. El empleo de un enfoque de reconocimiento de patrones está motivado por el hecho de que es una herramienta que permite incorporar conocimiento de situaciones previas a la resolución de problemas.

1.8. Organización de la tesis

La presente tesis se encuentra dividida de la siguiente manera:

- Capítulo 2: Examina el estado del arte, se discuten sistemas y artículos de actualidad relacionados con la tesis que se presenta.
- Capítulo 3: Contiene el marco teórico que da sustento a desarrollo de la tesis.
- Capítulo 4: Describe la metodología para propuesta para la resolución del problema planteado.
- Capítulo 5: Discute los resultados de la tesis.
- Capítulo 6: Propone posibles mejoras o áreas por explorar en el futuro.
- Capítulo 7: Presenta las conclusiones de la tesis.

1.9. Conclusiones

En este capítulo se ha presentado un panorama general del problema de la reconstrucción 3D así como de los desafíos por resolver durante el transcurso del presente trabajo. Una de las principales motivaciones por las cuales se está llevando la investigación en curso es la de dar un aporte novedoso al área de visión por computadora mediante el cual se planteé de manera diferente un problema clásico de visión.

Capítulo 2

Estado del arte

En esta sección se hace una descripción de sistemas similares al propuesto en la presente tesis, los sistemas a los que se hacen referencia son de actualidad e incluyen técnicas para la estimación de la profundidad clásicas como visión estereoscópica, SFM hasta técnicas novedosas que incluyen el uso de reconocimiento de patrones o análisis estadístico de la imagen.

2.1. Reconstrucción de objeto 3d a partir de imágenes calibradas

El sistema presentado en [17], usa una variación de visión estereoscópica, dicho sistema consiste en una única cámara la cual captura imágenes de la rotación de objetos enfrente de ella, la rotación se lleva a cabo de manera constante y a intervalos definidos de modo tal que a cada imagen se le puede asociar una matriz de rotación.

Una vez obtenida la secuencia de imágenes el primer paso que realiza este sistema consiste en obtener puntos de interés en las imágenes, para dicho propósito se emplea el algoritmo detección de bordes pre-

sentado en [12]. En una etapa posterior se realiza la correlación de las disparidades en las imágenes empleando únicamente puntos de interés previamente obtenidos, dicha correlación se lleva a cabo de dos en dos imágenes sucesivas.

Un punto muy importante a destacar es que este sistema realiza la búsqueda de la correlación no solo tomando como referencia una de las imágenes si no que se busca correlación de una imagen I_1 hacia una I_2 y en el sentido inverso. Para poder hallar la correlación primero se tiene que satisfacer que los puntos donde hay correlación deban estar en la misma línea epipolar.

Cuando se han obtenido los puntos donde hay correlación, se procede a agregar dichos puntos a un modelo tomando en cuenta que estos puntos han sufrido rotaciones y translaciones. Es primordial hacer notar que este sistema debe estar bien calibrado para poder realizar la reconstrucción.

2.2. Reconstrucción multi-vista precisa empleando visión binocular robusta

En [19]se presenta un trabajo en el cual se lleva a cabo la reconstrucción 3D de un objeto emplean visión binocular, el primer paso para realizar la reconstrucción es hallar una correspondencia entre las dos imágenes capturadas, para este efecto sus autores emplean la correlación cruzada normalizada, la cual se aplica sobre ventanas de NxN pixeles en ambas imágenes con el objeto de localizar los puntos de correspondencia

De acuerdo a sus autores también se puede variar el tamaño de las ventanas de búsqueda empleando un factor de escala, dicho factor de escala fue determinado por los autores de manera experimental. Una vez obtenidas las correspondencias se emplean para obtener un mapa de profundidad, posteriormente se elimina ruido de alta frecuencia, finalmente del mapa de profundidad se extrae un nube de puntos a partir de la cual se obtiene un modelo 3D.

De acuerdo a los resultados de este método, la reconstrucción de objetos con muchas curvas, como por ejemplo almohadas, ropa, esculturas pequeñas es muy buena, ya que la calidad de los modelos generados es muy cercana al modelo real.

Entre las limitaciones del método presentado se pueden encontrar, regiones con huecos en los modelos generados, esto principalmente debido a que no siempre es posible establecer una correlación perfecta entre las dos imágenes usadas, otra limitación es el uso de un patrón de referencia sobre el cual son puestos los objetos paro poder realizar la reconstrucción y finalmente el método solo fue empleado para la reconstrucción de objetos aislados.

2.3. Empleo de visión-estereoscópica en tiempo real para la navegación de un robot móvil

Aquí [15] se detalla la implementación del sistema de visión de un robot móvil, el sistema de visión consiste en un arreglo de tres cámaras digitales con espacio entre si y distribuidas como un triángulo, de estas tres cámaras de realizan dos cálculo de la disparidad uno para la cámara que queda arriba con la de la izquierda y otro para la cámara de arriba con la cámara de la izquierda, al final se suman ambos cálculos de la disparidad para obtener un solo mapa de disparidad que contenga un número mínimo de errores.

Una vez obtenido el mapa de disparidad se procede a calcular la profundidad de los puntos de la escena que está viendo el robot, con esta información el robot procede a navegar en el ambiente en que se encuentra y va generando un mapa del ambiente empleando un algoritmo de ocupación de malla.

De acuerdo a sus autores el principal problema que tiene el empleo del sistema de visión estereoscópico es el hecho de que hay situaciones en las que no es posible calcular de manera correcta ciertas regiones en el mapa de disparidad lo que se refleja como “picos” en

el mapa reconstruido de la escena donde navegó el robot, otro problema que se presenta son situaciones en las que una de las cámaras está viendo por encima de un objeto y otra de las cámaras ve a través de un hueco en el objeto, esto también produce errores en el cálculo de la disparidad.

2.4. Reconstrucción de profundidad 3D a partir de una sola imagen fija

El trabajo descrito en [20], se aparta de las técnicas tradicionales para la estimación de la profundidad e introduce el área de reconocimiento de patrones, esto con el fin de modelar el problema de la obtención de profundidad en una imagen como un problema de aprendizaje.

La idea principal de este trabajo radica en que dada una imagen esta tiene asociado un mapa de profundidad y que dicha relación se puede extraer o aproximar mediante un mecanismo de aprendizaje, al plantear el problema de esta forma se hace necesario un conjunto de características que describan de manera significativa la relación de la imagen con su mapa de profundidad, el mapa de paridad fue obtenido mediante el uso de un escáner LASER.

La solución tomada en [20] para extraer características es la de muestrear la imagen en diferentes escalas para de este modo obtener tanto características locales como globales de la imagen. Para la obtención de un patrón de salida que corresponda al mapa de profundidad se emplea un mecanismo de aprendizaje basado en el campo aleatorio de Markov, es decir, un clasificador de patrones probabilístico. De acuerdo a sus autores el porcentaje de recuperación de mapas de profundidad correctos es del 66 %.

2.5. Forma a partir del enfoque usando una red neuronal multicapa

Este trabajo [18] describe el uso de redes neuronales para aprender la relación que existe entre la imagen capturada, la superficie enfocada por la cámara y los puntos 3D a los que pertenece, para el funcionamiento de este enfoque los autores emplean sesenta imágenes, de cada imagen se determinan el área que es enfocada, después se estima la forma tridimensional de cada una de las áreas enfocadas, finalmente se hace una composición de las formas obtenidas y esto da como resultado un objeto 3D de a partir del conjunto de fotografías usado.

La principal ventaja de acuerdo del método propuesto en este trabajo, de acuerdo a sus autores, es que el costo computacional de emplear una red neuronal para aproximar la forma es significativamente menor al de enfoques tradicionales de forma a partir del enfoque, otra ventaja de acuerdo a los experimentos llevados a cabo por los autores, es que el método propuesto da mejores resultados que los métodos tradicionales. Las desventajas de este trabajo son que solo se emplearon fotografías de un cono y pintado con un patrón de anillos y fotografías microscópicas de una moneda, es decir solo se empleo para la recuperación de un solo objeto y no de una escena.

2.6. Sobre la creación de mapas de profundidad a partir de visión monocular mediante estructura a partir del movimiento

En [21] se describe un sistema que está en desarrollo para el cálculo de mapas de profundidad para aplicarlo en televisión 3D. Este sistema emplea un algoritmo de SFM para obtener la profundidad en una secuencia de vídeo.

Una característica particular de este sistema es que no se tiene conocimiento previo de la calibración de la cámara ni del movimiento de la cámara, para superar este obstáculo se hacen las siguientes suposiciones entre dos cuadros de vídeo continuos

a) La escena cambia poco.

b) El movimiento de la cámara es paralelo a la escena.

c) La posición de la cámara no cambia de manera abrupta.

Bajo estas suposiciones el principal problema radica en hallar la correspondencia en la disparidad de las imágenes capturadas.

Para hallar la disparidad en una primera etapa el sistema obtiene puntos de interés en la imagen empleando el filtro de Harris[12], dichos puntos son rastreados a lo largo de una secuencia de cuadros de vídeo y se calcula una correlación empleando el método de correlación por bloques. Al final de este proceso se obtiene un mapa de profundidad.

2.7. Reconstrucción 3D urbana detallada en tiempo real a partir de vídeo

En [22] se describe un sistema que realiza la reconstrucción de ambientes urbanos, dicha reconstrucción es guardada como un modelo 3d virtual y posteriormente es georeferenciado con GPS sobre un mapa de modo tal que es posible ver físicamente como es un lugar si la necesidad de estar presente.

El sistema funciona tomando como entrada un vídeo del área a reconstruir, dicho vídeo es capturado desde un vehículo que se desplaza a velocidad constante y a no más de 10 km/h, el sistema de captura de vídeo consiste en ocho cámaras que están alineadas en paralelo, al igual que en [21] la profundidad se obtiene gracias a la disparidad de cuadros de vídeo sucesivos.

El primer paso que realiza el sistema es el de obtener características

en las imágenes, la característica que se extrae de la imagen es el gradiente y dichas características son rastreadas en cuadros de vídeo sucesivos para poder obtener una correspondencia de la disparidad. Finalmente los mapas de profundidad obtenidos son combinados para su posterior integración como un modelo 3D.

2.8. Imagen y profundidad a partir de una cámara convencional con apertura codificada

El trabajo presentado en [23] propone un mecanismo totalmente nuevo para la obtención de la profundidad a partir de una sola cámara, esto se logra remplazando el mecanismo de apertura en la cámara por uno de apertura codificada.

El mecanismo de apertura codificada consiste en un filtro en el que físicamente han sido perforados distintos patrones, el efecto logrado con este mecanismo es el de obtener disparidad dentro de una misma cámara, además las distintas imágenes obtenidas presentan un grado de borrosidad de acuerdo a la distancia de los objetos.

Para estimar un mapa de profundidad se realiza un análisis estadístico del los gradientes de la imagen, con esto se caracteriza a las regiones borrosas y con esta información se estima el kernel (i.e filtro) que produjo una región borrosa dada, de acuerdo al tamaño del kernel estimado se asigna la profundidad en la que se encuentra una determinada región.

2.9. SFS para iluminación oblicua con mejora de precisión por optimización de dirección de luz

En [24] se presenta una novedosa técnica de SFS para que la precisión de la forma recuperada sea muy cercana, los objetos con los que se prueba este método propuesto son sintéticos y reales, sus autores toman como restricciones una proyección ortogonal de la escena y suponen que el objeto fotografiado es un reflector lambertiano, el proceso de reconstrucción comienza tomando muestras de la imagen en cuatro direcciones así como de los gradientes de dicha imagen.

Para la obtención de la profundidad se establece la relación de la profundidad con el mapa de reflectancia del objeto (esto de acuerdo a la ley del coseno de Lambert), con las muestras previamente obtenidas de la imagen se lleva a cabo el método de iteración de Jacobi para obtener cuatro relaciones que posteriormente son empleadas para obtener de manera iterativa la profundidad empleando un procedimiento de error cuadrado mínimo, para cada punto reconstruido la solución inicial de la que parte es cero. De acuerdo a su autor, el método presentado es superior a los métodos SFS actuales ya que tiene la capacidad de superar muy los problemas impuestos por la dirección de la luz ya que emplea métodos de optimización.

Cabe destacar que este método solo funciona para objetos aislados y no para escenas compuestas por múltiples objetos o bien paisajes, ya que este método como todos los basados en SFS dependen de la cantidad de luz que incide sobre un objeto para poder funcionar.

2.10. Representación de ropa mediante SFS con primitivas de sombreado

En este trabajo [25] se presenta un método mediante el cual es posible obtener la forma de la superficie de ropa, sábanas o algún tipo de tela extendida sobre una superficie. El método propuesto es una combinación de técnicas de SFS con una técnica propuesta por el autor.

Este método hace la primera consideración de que las imágenes (ropa, sábanas, telas, etc) se pueden dividir en dos regiones que tienen "dobleces" y regiones que son mas uniformes, el primer paso que se sigue en la metodología propuesta es identificar las regiones de dobleces, estas regiones son identificadas por el sombreado que poseen en sus bordes, aunque sus autores no proporcionan el mecanismo mediante el cual las segmentan. Un punto muy importante es que una vez obtenidos los dobleces estos descompuestos en primitivas más pequeñas y estas son caracterizadas como un patrón que contiene el tipo de primitiva, orientación geométrica de la primitiva, localización, escala y atributos fotométricos, una vez hecho esto la primitivas son interconectadas como un grafo, es decir que los dobleces en la imagen son modelados como un grafo.

Para la recuperación de la forma se realiza un recorrido del grafo, con la información previa de la primitiva esta se busca en un diccionario de primitivas y se le asigna una forma 3D, es decir que la metodología propuesta por el autor tiene conocimiento previo de que es la profundidad, para lograr dicho efecto los autores emplean un clasificador bayesiano que permite identificar el tipo de primitiva y asignar una forma 3D, para las otras áreas de la imagen que no son dobleces se emplean técnicas de SFS tradicionales.

Es muy importante hacer notar que este trabajo funciona únicamente con condiciones controladas de iluminación, se reconstruye únicamente un objeto a la vez y es un trabajo específico para modelar ropa o telas.

2.11. Correspondencia estereoscópica mediante pesos de apoyo geodésicos

Debido a que la correspondencia resulta ser una parte crucial para realizar la reconstrucción 3D se ha incluido a [26].

Este trabajo se centra principalmente en el uso de pesos dinámicos para la ventana que se emplea durante el proceso de correspondencia, los pesos se ajustan midiendo la distancia geodésica desde el pixel del centro hacia sus vecinos, esto significa que aquellos pixeles que son muy similares tendrán pesos más elevados y será menor para aquellos que son muy distintos.

El motivo por el cual se realiza esto según sus autores es que dicho procedimiento ayuda de manera enorme a mejorar el proceso de correspondencia, ya que ayuda a realizar una especie de segmentación.

Entre las ventajas de este método está el hecho de que permite conservar la forma de los objetos, y además usa ventanas pequeñas (de acuerdo al autor de 3x3) lo que reduce su tiempo el tiempo que toma estimar lo que se denomina mapa de disparidad.

Los autores no mencionan que sucede cuando hay cambios de iluminación o como se comporta su método cuando se presentan imágenes ruidosas.

2.12. Sobre-segmentación para correspondencia estereoscópica

El trabajo presentado en [27] presenta una metodología para la estimación de un mapa de disparidad mediante el uso de técnicas de segmentación.

La metodología propuesta funciona realizando un preprocesamiento

a las imágenes antes de realizar el proceso de estimación de la correspondencia; dicho preprocesamiento consiste en dividir las imágenes en regiones que son similares, al realizar dicho proceso de acuerdo a sus autores se logra hacer que el proceso de correspondencia (estimación del mapa de disparidad) se más exacto por que se busca una región en particular y no una ventana predefinida.

Una vez se ha estimado el mapa de disparidad este se refina mediante el uso de propagación de creencia sobre un campo aleatorio de Markov.

Una desventaja es que la no haber un tamaño fijo de ventana el tiempo que se tarda en calcular el mapa de disparidad puede variar, también hay que decir que inicialmente las disparidades que se encuentran corresponden únicamente a las regiones en que se han dividido las imágenes, motivo por el cual sus autores deben recurrir a técnicas sofisticadas como propagación de creencia para poder estimar la disparidad en todos los pixeles de la imagen.

2.13. Correspondencia estereoscópica cooperativa con apoyo local adaptativo

La metodología propuesta en [28] al igual que [26] hace uso de pesos que se adaptan de manera dinámica de acuerdo al contenido de la ventana que se usa para realizar el proceso de correspondencia.

La principal diferencia que tiene con [26] es que en [28] hace uso de optimización cooperativa para estimar el mapa de disparidad, lo que se optimiza es una función global que toma encuentra la disparidad de los vecinos dentro de una ventana. Esto último es muy importante porque le permite a la metodología propuesta refinar el mapa de disparidad, su autor también menciona que una ventaja es el tamaño de las ventanas usadas (3x3).

Es muy importante decir que esta metodología hace uso intensivo aritmética de punto flotante por lo que de acuerdo a su autor

podría ayudar a implementar la metodología en tiempo real.

En este trabajo no se menciona que sucede cuando las imágenes han sido contaminadas con ruido o que ocurre con objetos que son transparentes.

2.14. Correspondencia mediante agregación adaptativa de costo

En [29] se presenta una metodología para la estimación de mapas de disparidad en tiempo real, en particular la idea presentada ha sido diseñada para funcionar en GPU.

La estimación de la disparidad se realiza mediante programación dinámica, la cual es usada para estimar la disparidad donde la correspondencia de una ventana referencia es mejor; en el proceso se correspondencia sus autores han incluido restricciones en cuanto a la suavidad del mapa de disparidad.

Este trabajo también hace uso de pesos adaptativos sin embargo su uso se realiza en una etapa posterior a haber realizado el computo de la disparidad, dicha etapa se denomina agregación y consiste en agregar los costos de las disparidades que se encuentra en el mismo nivel, según los autores de [29] esto realiza una mejora notoria con respecto a no usar pesos adaptativos.

La principal ventaja de este trabajo radica en que funciona en tiempo real, lo que significa que puede tener aplicaciones prácticas, sin embargo sus autores aclaran que las imágenes que se emplean son de 320x240 y por tanto los mapas de disparidad no tienen mucho detalle.

2.15. Pesos adaptativos mediante segmentación

El trabajo descrito en [30] tiene similitudes con el trabajo [27] en el sentido que los dos métodos hacen uso de técnicas de segmentación, sin embargo [30] se distingue por emplear la segmentación en la etapa de estimación de la correspondencia pero si buscar correspondencia de una región segmentada, sino una ventana de tamaño fijo.

Los pesos adaptativos son asignados tomando en cuenta que tanto se parecen los pixeles dentro de una ventana de correspondencia, también se considera si los pixeles pertenecen a la misma región segmentada. Debido a que este método usa segmentación tiene la ventaja de poder conservar muy bien lo detalles finos de los objetos que se encuentran en el escenario.

Otra ventaja de esta metodología que sus autores destacan es que gracias al uso de pesos adaptativos es posible hallar una mejor correspondencia en regiones que son ambiguas; por ejemplo el caso de barras verticales paralelas o bien regiones que tiene muy poca textura o el caso contrario carecen de textura.

Este trabajo no menciona que método se segmentación se emplea, una desventaja muy importante de este método es que requiere usar ventanas muy grandes (51x51) lo podría afectar de manera muy negativo el tiempo que se tarda en estimar un mapa de disparidad.

2.16. Conclusiones

En este capítulo se han discutido trabajos de suma importancia para la presente tesis, en particular de mayor relevancia resultan [18],[19],[20],[23] y [25], principalmente por las siguientes aportaciones de dichos trabajos:

a) Aprendizaje de la relación de la profundidad con la imagen.

b) Obtención de un kernel de desconvolución de una imagen, que permite establecer relación con la profundidad.

c) Planteamiento de métodos robustos para la obtención correspondencia entre dos imágenes.

d) Planteamiento de métodos para la extracción de características locales que permitan establecer la profundidad.

En particular de los puntos mencionados anteriormente resulta de mucho interés para la tesis actual contar con trabajos previos que proporcionan un mecanismo de aprendizaje para establecer la relación de la profundidad de una escena con la imagen de dicha escena.

Otro aspecto importante de este capítulo es que se ha dedicado una parte del estado del arte a la estimación de la disparidad ya que esta parte resulta de suma importancia ya que la profundidad y la disparidad (se muestra en el siguiente capítulo) se pueden expresar una en términos de la otra, es decir que si no se estima la disparidad es imposible determinar la profundidad (al menos para el caso de visión estereoscópica).

Finalmente es importante destacar que todos los métodos para la estimación de disparidad son "locales" es decir hacen uso de ventanas de correspondencia, y son de relevancia para la presente tesis porque se ha decidido ir por el mismo camino.

Capítulo 3

Marco teórico

En este capítulo se presentan algunas de las técnicas encontradas en la literatura para la reconstrucción 3D a partir de ya sea una o varias imágenes digitales, teniendo en común todas estas técnicas un respaldo geométrico y matemático muy sólido, motivo por el cual su revisión es obligada para cualquier trabajo que incursiona en el área de la reconstrucción 3D.

3.1. Noción básica de la formación de una imagen

Una forma común de modelar la formación de una imagen en un sensor de imagen (CCD o CMOS) es ver dicho proceso como una proyección de la escena u objeto sobre un plano, ver [9] y [10]. A continuación se muestra una figura modelando dicha situación.

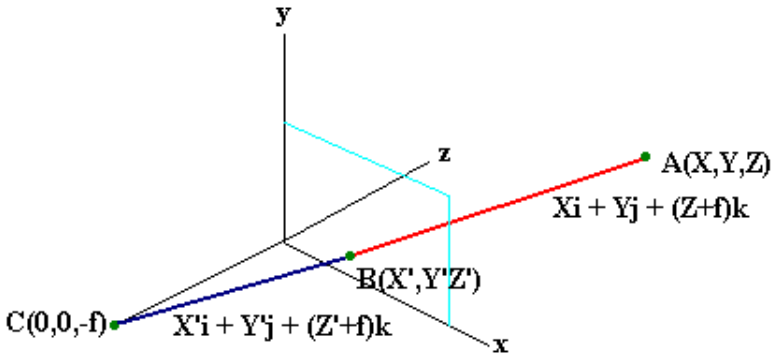


Figura 3.1: Representación de una proyección

A partir de esta representación resulta de gran interés modelar de forma matemática dicho proceso, para este efecto se expresará dicho problema como una transformación.

Los vectores $X'i + Y'i + (Z' + f)k$ y $Xi + Yi + (Z + f)k$ pueden ser definidos uno mediante el otro por un factor de escala, de modo que $X' = Xt$, $Y' = Yt$ y $Z' + f = (Z + f)t$, como el plano de la imagen yace en el origen del sistema coordenado entonces:

$$t = \frac{f}{Z + f} \tag{3.1}$$

De las ecuaciones anteriores es claro que se pierde información al pasar del plano en $\mathbb{R}^3$ al plano en $\mathbb{R}^2$, es decir se pierde la información de la profundidad.

3.2. Imágenes digitales

Una imagen digital es, de acuerdo a [5], una representación discreta de una función $f(x, y)$ donde el valor de dicha función corresponde a la combinación de iluminación reflejada y absorbida por una escena u objeto, dicha representación discreta es el resultado de

un proceso de adquisición como el que se describe en la figura de abajo.

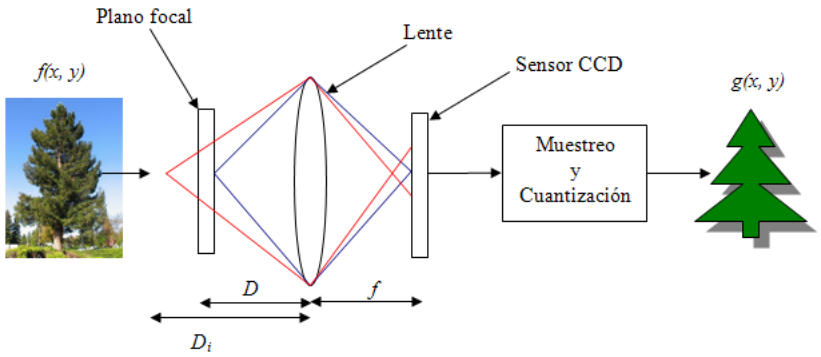


Figura 3.2: Proceso de adquisición de la imagen

La fuente de iluminación puede ser luz visible, luz infrarroja, rayos X, SONAR, RADAR, o un rayo LASER por citar algunos, el caso particular que es de interés en la presente tesis son la imágenes que son resultado de la luz visible, i.e fotos o vídeo capturados por una cámara convencional.

La función $g(x, y)$ será representada como un arreglo bidimensional donde x y y representan las posiciones espaciales dentro de la imagen donde se encuentra un valor de intensidad de color o escala de gris, a dicho punto en el espacio se le denomina *pixel*.

3.2.1. Vecindades de pixeles

Se dirá que un pixel $P_i(x', y')$ es vecino de un pixel $P_j(x, y)$ si se satisface (3.2), ver [6].

$$[\sum_{k=1}^{2} |P_{jk} - P_{ik}|^r]^{\frac{1}{r}} \leq D \tag{3.2}$$

Donde:

- k indica la componente espacial del pixel

- r es un entero positivo

- D es un entero positivo que indica la distancia máxima a la que está ubicado un vecino.

3.3. Filtrado de imágenes digitales

Una vez llevado a cabo el muestreo de la función $f(X, Y)$ y posteriormente la recuperación de la función $g(x, y)$ dicho resultado puede contener ruido. El ruido será una señal no deseada que se encuentra presente en la señal que se está analizando.

Entre los tipos de ruido que se pueden encontrar en una imagen están los siguientes:

- Ruido aditivo

- Ruido substractivo

- Ruido sal y pimienta

- Ruido Gaussiano

Con el propósito de reducir el nivel de ruido en una imagen ésta se filtra de igual forma que toda señal es decir se realiza un convolución con un determinado filtro. La convolución de una imagen se define como:

$$g(x, y) = f(x, y) * K_{mn} \tag{3.3}$$

Donde:

- $f(x, y)$ es la señal con ruido.

- $g(x, y)$ es la señal filtrada.

- K es el filtro con el que es convolucionada la imagen.

- nm es el tamaño del filtro, es decir $n \times m$.

El filtro K_{nm} es una matriz donde cada componente tiene un valor por el cual son multiplicados los pixeles durante la convolución. Es importante resaltar que los filtros que basan su funcionamiento en la convolución se denominan filtros *lineales*, por ejemplo el filtro promedio, mientras filtros que realizan operaciones tales como ordenamientos o comparaciones son conocidos como filtros *no lineales*, por ejemplo los filtros de mediana, mínimo, máximo, etc.

Resulta importante hacer notar que el filtrado en imágenes no solo se realiza para eliminar el ruido presente en la función $f(x, y)$ sino también se emplea para extraer características específicas de una imagen como son las esquinas de los objetos presentes en una imagen, como por ejemplo los filtros de Roberts, Prewitt y Sobel así como el Laplaciano, ver [5].

3.4. Representación de la imagen en el dominio de la frecuencia

Hasta este momento se ha hablado de la imagen únicamente como una función $f(x, y)$ cuyo dominio es el espacio que ocupa la imagen, sin embargo también es posible representar a las imágenes en el dominio de la frecuencia, para lograr este objetivo se emplea la transformada de Fourier.

Transformada de Fourier

$$F(u) = \int_{-\infty}^{\infty} f(x)e^{-2j\pi ux}dx \tag{3.4}$$

Transformada inversa de Fourier

$$f(x) = \int_{-\infty}^{\infty} F(u)e^{2j\pi ux}du \tag{3.5}$$

Donde:

- $f(x)$ es una función continua en el dominio del tiempo o del espacio.

- $F(u)$ es una función continua en el dominio de la frecuencia.

Las ecuaciones (3.4) y (3.5) representa tanto la transformada de Fourier como su inversa para el caso de variables continuas, en el caso discreto la transformada de Fourier está dada por las siguientes ecuaciones:

Transformada de Fourier

$$F(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(x) e^{\frac{-2j\pi ux}{M}} \tag{3.6}$$

Transformada inversa de Fourier

$$f(x) = \frac{1}{M} \sum_{u=0}^{M-1} F(u) e^{\frac{2j\pi ux}{M}} \tag{3.7}$$

Donde:

- $F(u)$ es una función discreta en el dominio de la frecuencia.
- $f(x)$ es una función discreta en el dominio del tiempo o del espacio.

El caso que es de interés en el procesamiento digital de imágenes es la transformada de Fourier para el caso de una función bidimensional, cuya representación matemática es.

Transformada de Fourier para una función discreta bidimensional

$$F(u,v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-2j\pi(ux/M+vy/N)} \tag{3.8}$$

Transformada inversa de Fourier para una función discreta bidimensional

$$f(x,y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) e^{2j\pi(ux/M+vy/N)} \tag{3.9}$$

De igual forma que en el dominio espacial se filtra una imagen digital mediante la convolución de un kernel [1], el filtrado en el dominio de la frecuencia se realiza mediante la multiplicación de la transformada de dicho filtro, esto se puede hacer debido a la propiedad de

la transformada de Fourier en la que la convolución en el dominio espacial equivale a una multiplicación en el dominio de la frecuencia, ver [7].

3.5. Detección de esquinas en las imágenes

Una esquina en una imagen es interpretada como un cambio pronunciado en el nivel de iluminación, es decir, que para hallar dicho cambio es necesario el empleo del operador diferencial gradiente el cual está dado por:

$$\nabla f(x) = \frac{\partial f}{\partial x}(x) = f(x + dx) - f(x) \tag{3.10}$$

Como se definió previamente una imagen es una función $f(x, y)$, para detectar las direcciones de máximo cambio se calcula el gradiente de la imagen, el cual estaría compuesto por derivadas parciales es decir una que va en la dirección en X y otra en Y.

$$\nabla f(x, y) = \frac{\partial f}{\partial x}(x, y) + \frac{\partial f}{\partial y}(x, y) \tag{3.11}$$

de donde:

$$\frac{\partial f}{\partial x}(x, y) = f(x + dx, y) - f(x, y) \tag{3.12}$$

$$\frac{\partial f}{\partial y}(x, y) = f(x, y + dy) - f(x, y) \tag{3.13}$$

Se debe tomar en cuenta una consideración muy importante cuando se calculan las derivadas de una imagen digital, al ser una función discreta, las derivadas serán únicamente aproximaciones, entre dichas aproximaciones se pueden encontrar los operadores diferenciales de Sobel, Prewitt, Roberts y el Laplaciano.

[1] Como kernel se entenderá un arreglo de pesos por el cual son multiplicados los pixeles durante la convolución

3.6. Calibración de la cámara

Como fue mencionado en la sección anterior, la formación de una imagen es interpretada como una proyección sobre un plano motivo por el cual resulta de suma importancia conocer la transformación que está teniendo lugar, con la finalidad de simplificar la representación de dicha transformación se empleara una proyección central (3.3), la cual recibe su nombre del hecho que el origen de la proyección es el origen del sistema coordenado.

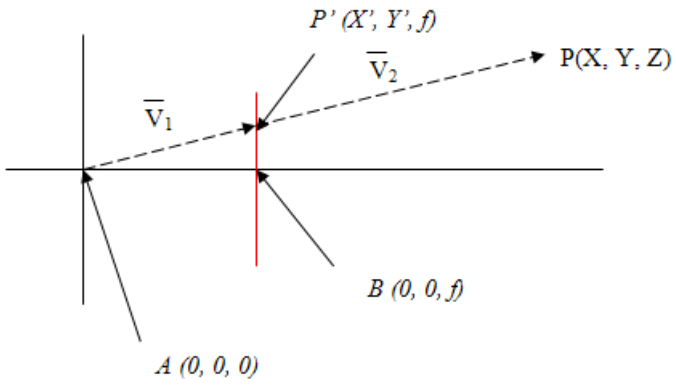


Figura 3.3: Representación de una proyección central

Donde:

- $\overline{V_1} = X'i + Y'j + fk$
- $\overline{V_2} = Xi + Yj + Zk$

Es claro de fig.(3.3) que $\overline{V_1}$ y $\overline{V_2}$ de donde $\overline{V_2} = \overline{V_1}t$, ya que ambos vectores están relacionados por un factor de escala, de modo tal que:

$$\begin{aligned} X' &= Xt \\ Y' &= Yt \\ f &= Zt \end{aligned} \tag{3.14}$$

despejando Z de eq.(3.14) se obtiene:

$$t = \frac{f}{Z} \tag{3.15}$$

Este es el factor de escala con el cual se definen $\overline{V_1}$ y $\overline{V_2}$, de modo tal que es posible reescribir esta transformación como una matriz:

$$M = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{3.16}$$

$$\begin{pmatrix} X' \\ Y' \\ 1 \end{pmatrix} = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \tag{3.17}$$

de la ec.(3.17) es de notar que se han conservado únicamente X' y Y' y la tercer coordenada es 1, esto es la representación en coordenadas homogéneas de (X', Y') sobre el plano donde se ha realizado la proyección.

Una vez definida esta transformación es necesario mencionar que f (del cap.1) es la distancia focal de la cámara, y usualmente está dada en milímetros (mm), la transformación definida únicamente involucra a f, sin embargo hay que recordar que las coordenadas sobre el sensor que captura la imagen no son continuas, motivo por el cual se requiere establecer a cuanto equivale cada pixel por unidad métrica (mm ,cm ,dm ,m,etc) del mundo que observa la cámara, también se debe tomar en cuenta que los elementos en el sensor no necesariamente tienen la misma dimensión en sus lados, finalmente se debe tomar en cuenta en qué punto se encuentra el centro óptico de la cámara, todo estos parámetros mencionados reciben el nombre de parámetros intrínsecos de la cámara, y es posible representarlos como una matriz:

$$I = \begin{pmatrix} S_x & 0 & O_x \\ 0 & S_y & O_y \\ 0 & 0 & 1 \end{pmatrix} \tag{3.18}$$

donde :

- S_x equivale al ancho del pixel en mm ,cm ,dm ,m,etc a lo que equivale en el mundo que observa.
- S_y equivale al alto del pixel en mm ,cm ,dm ,m,etc a lo que equivale en el mundo que observa.
- O_x el centro óptico de la cámara en X.

- O_y el centro óptico de la cámara en Y.

Ahora incorporando la matriz I en la ec.(3.17):

$$\begin{pmatrix} X' \\ Y' \\ 1 \end{pmatrix} = \begin{pmatrix} S_x & 0 & O_x \\ 0 & S_y & O_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \tag{3.19}$$

3.7. La reconstrucción 3D como un problema de inversión

Recordando que la generación de una imagen digital puede ser vista como una proyección sobre un plano y que durante dicha transformación la información de la profundidad del objeto o escena se pierde, entonces se puede afirmar que la labor de un algoritmo de reconstrucción 3d ya sea a partir de una o varias imágenes consiste en recuperar la profundidad, i.e. un problema de inversión.

Se entenderá como un problema de inversión en el que dado un conjunto de datos $Y_1, Y_2, ..., Y_n$ les debe ser asociada la función o transformación inversa tal que esta nos lleve de regreso al dominio $X_1, X_2, ..., X_n$, es decir:

$$G(Y_i) \Longrightarrow X_i \tag{3.20}$$

3.8. Métodos de reconstrucción

3.8.1. Forma a partir del sombreado

La forma de un objeto así como el material del que está hecho influye en la forma en que este refleja una cantidad determinada de iluminación, también conocida como luminancia, los métodos basados en SFS hacen uso de dicha propiedad para poder realizar la reconstrucción 3D de un determinado objeto.

Para llevar a cabo la reconstrucción en los algoritmos basados en SFS asumen las siguientes consideraciones [8]:

- El objeto observado es un reflector Lambertiano, es decir que la irradiancia del objeto se comporta de acuerdo a la ley del coseno de Lambert, ver la figura (ecuaciones).
- La iluminación proviene de un solo punto, que es paralela al objeto iluminado y que no hay interacción entre los rayos reflejados y distintas partes del objeto u otros objetos en la escena.
- La proyección del objeto en una imagen corresponde a una proyección ortogonal, ver figura (proyección ortogonal).
- La geometría del objeto es diferenciable de manera continua.
- El sensor (de la cámara) es lineal, es decir que los valores de escala de gris corresponden de manera lineal a la irradiancia del objeto fotografiado.

Sin embargo es importante mencionar que también hay algoritmos de SFS los cuales asumen una proyección de perspectiva.

Proceso de reconstrucción en SFS

La premisa principal de SFS es que existe una relación entre la imagen de un objeto y su mapa de reflectancia, dicha relación daría información acerca de la profundidad espacial de dicho objeto.

El mapa de reflectancia del objeto de acuerdo a [8] es la relación que existe entre la irradiancia y la orientación de la superficie de un objeto, usualmente la representación de dicho mapa se hace en el espacio del gradiente (por algún método para obtener dicho gradiente) o bien con un mapa normal de la superficie (el mapa de todos los vectores normales a la superficie), es decir que el mapa de reflectancia es una función ya sea del mapa de gradientes o del mapa normal.

La relación existente entre la irradiancia y el mapa de reflectancia es dada por la ec.(3.21)

$$E(x, y) = R(p, q) \tag{3.21}$$

Donde:

- $E(x, y)$ es la irradiancia

- $R(p, q)$ es el mapa de reflectancia

En particular en SFS se usa el mapa de gradientes de la imagen ya que para poder recuperar la forma de un objeto en SFS es preciso poder determinar el mapa normal de la superficie del objeto a reconstruir, el cual se obtiene empleando la siguiente ecuación.

$$E(x, y) = E_0 P \hat{n} \hat{s} \tag{3.22}$$

Donde:

- $\hat{n}$ es el vector normal a la superficie

- $\hat{s}$ es la dirección de la fuente de iluminación.

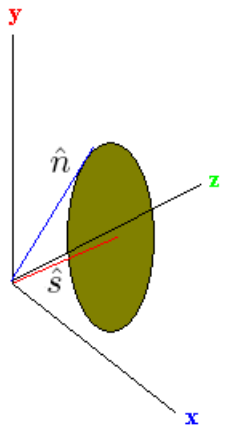


Figura 3.4: Representación del planteamiento del problema en SFS, nótese que el vector $\hat{n}$ puede ocupar cualquier lugar alrededor del perímetro del círculo

Para recuperar la forma del objeto SFS trata de determinar el vector $\hat{n}$,en este punto es necesario hacer la observación que la ec.(3.22) tiene un número infinito de soluciones, de acuerdo a [13],es decir que la reconstrucción que se obtenga no necesariamente corresponde de manera correcta a la realidad, por este motivo el problema de la

obtención de la forma usando SFS permanece siendo uno de los más difíciles de resolver.

3.8.2. Reconstrucción con multi-imágenes

En esta categoría de algoritmos se encuentran aquellos que para poder determinar la profundidad de un objeto emplean la disparidad de las imágenes capturadas de dicho objeto. En esta categoría de algoritmos podemos encontrar:

- Visión estereoscópica.
- Profundidad a partir de enfoque.
- Estructura a partir del movimiento.

En lo sucesivo se usará un sistema de visión estereoscópica binocular como ejemplo para ilustrar como se lleva a cabo la estimación de la profundidad en los algoritmos previamente citados, también se asumirá una proyección central como se muestra en la figura de abajo.

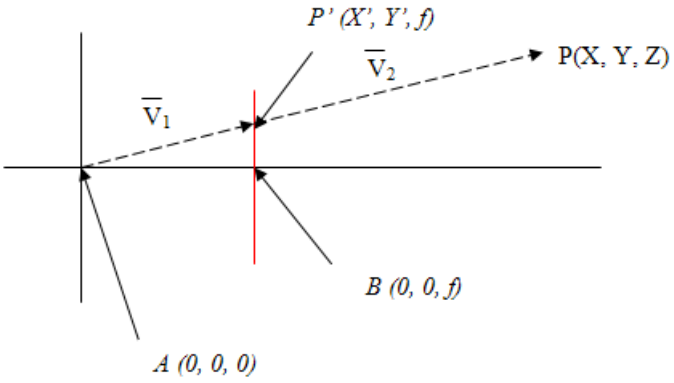


Figura 3.5: Representación de una proyección central

Correlación de imágenes múltiples y visión estereoscópica

La correlación de las distintas imágenes capturadas es la primera fase que llevan a cabo los algoritmos que emplean múltiples imáge-

nes, en esta etapa lo que se busca es encontrar que puntos de una imagen X corresponden con los de una imagen Y. Para llevar a cabo este proceso se toman en cuenta principalmente dos características presentes en las imágenes capturadas.

- Nivel de similitud en las distintas regiones pixel a pixel de cada imagen usada.
- Correlación existente debido a la geometría del modelo de adquisición.

Es muy importante mencionar que el proceso para hallar la correlación entre dos puntos en las imágenes X y Y conlleva realizar una búsqueda, esto significa que se realizan desplazamientos en la imagen Y, dichos desplazamientos reciben el nombre de disparidades; de aquí en adelante se usar correlación para referirse a disparidad y viceversa a menos que se indique otra cosa.

Nivel de similitud en las distintas regiones pixel a pixel en visión estereoscópica

En un sistema de visión estereoscópico donde la escena u objeto es capturado desde dos puntos de vista diferentes y la distancia entre las cámaras es conocida, entonces es posible medir el nivel de semejanza de dos puntos (X_{A1}, Y_{A1}) y (X_{B1}, Y_{B1}) es decir el nivel de correlación que existe entre esos dos puntos.

Con el propósito de poder medir esta disparidad se hace la suposición de que la disparidad entre las dos imágenes varía de manera constante en ambas imágenes capturadas, como en [8], [9], una medida para medir dicha disparidad es:

$$\left|\sqrt{(X_{A1}-X_{B1})^2-(Y_{A1}-Y_{B1})^2}+\sqrt{(X_{A2}-X_{B2})^2-(Y_{A2}-Y_{B2})^2}\right| \tag{3.23}$$

Donde (X_{A2}, Y_{A2}) y (X_{B2}, Y_{B2}) representa puntos en las vecindades de (X_{A1}, Y_{A1}) y (X_{B1}, Y_{B1}) respectivamente, si la evaluación de (3.23) se encuentra debajo de un umbral previamente definido entonces se dice que hay una correspondencia.

Otro mecanismo para evaluar la disparidad entre dos imágenes consiste en realizar la evaluación de la ec.(3.23) de la disparidad donde se ha encontrado una esquina en las imágenes capturadas, este método depende en gran medida del operador de gradiente que se emplee.

Una observación de gran importancia es el hecho de que para poder hallar los puntos de disparidad es necesario realizar una búsqueda en la que se compara el punto P_1 localizado en una de la imagen A y compararlo con un punto P_2 en B a lo largo del eje X, como se describe en [8], [9] [10], al realizar dicha búsqueda es necesario un mecanismo de decisión que permita discriminar entre regiones que presentan una correspondencia, para dicho propósito se emplean medidas de similitud como el uso del error cuadrado [16] o la correlación cruzada normalizada como en [19].

En estos dos métodos,[16] y [19] el objetivo es localizar un mínimo local, es decir esto indicaría que punto P_1 en la imagen A tiene correspondencia con un punto P_2 en la imagen B. Estos dos algoritmos hacen uso de ventanas de tamaño variable para obtener mayor información del área examinada y mejorar el nivel de correlación.

Otros enfoques existentes como [36] hacen uso de autómatas celulares para el mejoramiento del nivel de correspondencia en la disparidad, dicho enfoque no será discutido en el presente documento.

Correlación existente debido a la geometría del modelo de adquisición en visión estereoscópica

En sistemas donde se emplean múltiples cámaras y se conoce la ubicación de dichas cámaras así como los parámetros de las cámaras, es posible hallar una correlación entre dos imágenes haciendo uso de la geometría del modelo de adquisición, es decir se puede introducir la restricción de la línea epipolar.

La restricción epipolar consiste en que dos puntos (X_A, Y_A) y (X_B, Y_B) tiene una correlación únicamente si estos yacen sobre la misma línea

epipolar, ver [9], a continuación se ejemplifica dicha situación.

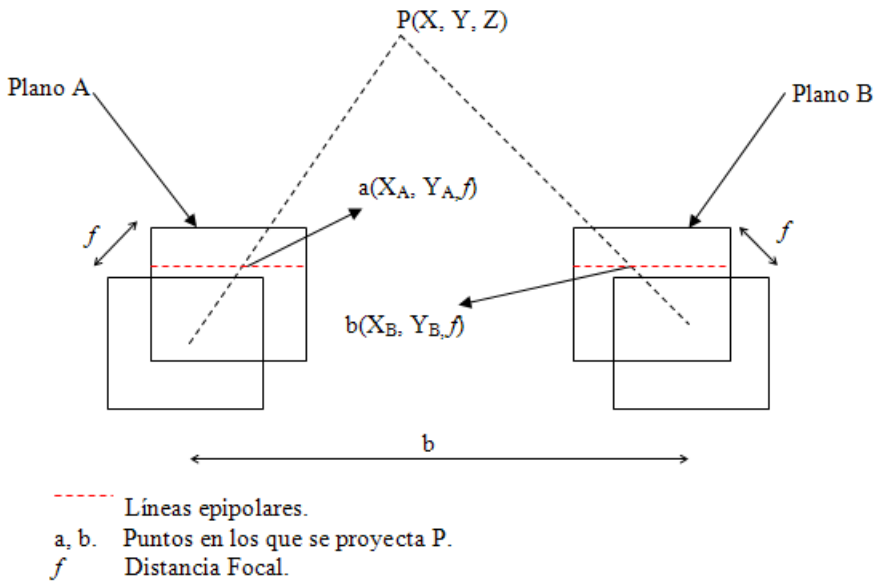


Figura 3.6: Representación de una proyección en dos planos

en el caso descrito en la fig.(3.6), la línea epipolar corresponde de manera directa a la coordenada y en las imágenes, para el caso en que las cámaras estuviesen rotadas, se tendrían que hacer las respectivas consideraciones para determinar la por donde pasan dichas líneas, y sobre ellas hacer la búsqueda de correlación de la disparidad.

Obtención de la profundidad

De la fig.(3.6) el punto $P(X, Y, Z)$ es proyectado en dos planos diferentes, con el fin simplificar la descripción se asumirá una proyección central como en la fig.(3.3). Para los planos A y B el punto P es transformado a los puntos (tX, tY, f) y $(t(X-b), tY, f)$, donde $t = \frac{f}{Z}$ ya que, de la fig.(3.3), los vectores V_1 y V_2 se pueden definir uno mediante el otro por un factor de escala.

Como ambas cámaras están ubicadas en la misma coordenada en el eje Z, entonces la profundidad o coordenada Z del punto P pue-

de ser determinada por $Z = \frac{fb}{X_A - X_B}$ tal que $X_A - X_B$ es la disparidad (el desplazamiento de un punto respecto al otro). Este proceso solo deber ser llevado a cabo una vez que se ha obtenido la correlación de la disparidad en caso contrario no hay forma de garantizar que X_A y X_B tengan una relación.

Tipos de algoritmos de correspondencia

En esta sección se describen brevemente las dos principales vertientes de los algoritmos de correspondencia para la visión estereoscópica, ya que es importante poder diferenciar sus características ventajas y desventajas, la fig.(3.7) muestra las dos principales vertientes.

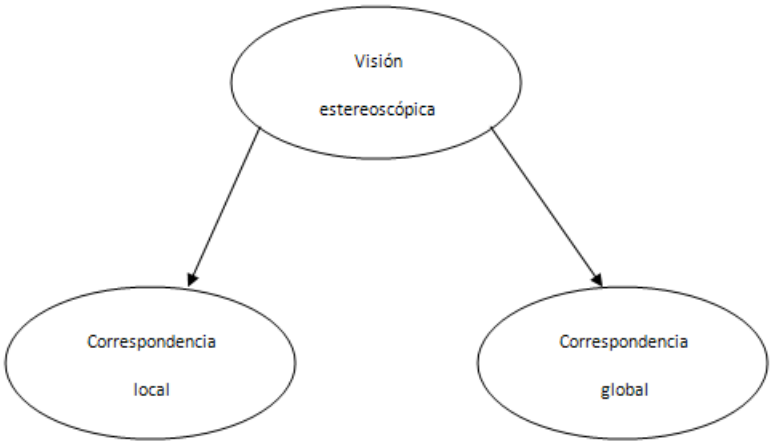


Figura 3.7: clasificación de los algoritmos de visión estereoscópica

Correspondencia local

Los algoritmos de correspondencia local tiene como característica principal el uso de ventanas de correspondencia donde la idea principal es tomar una de las imágenes como referencia y otra como un área de búsqueda. Aunque también es común encontrar aquellos que hallan la correspondencia de una línea completa mediante programación dinámica [11].

Generalmente la correspondencia se halla usando una medida de similitud; por ejemplo: sumatoria de diferencias al cuadrado (SDC), sumatoria de diferencias absolutas (SDA) o correlación cruz normalizada (CCN). Al valor que arrojan estas medidas se le denomina costo, cómo la comparación se realiza pixel a pixel y esto implica un desplazamiento en el área de búsqueda (disparidad) se dice que cada nivel de disparidad tiene asociado un costo, dicho costo se almacenan de igual forma por niveles de disparidad (ver fig.(3.8)).

Figura 3.8: Costos almacenados por nivel de disparidad

Una vez se tienen calculados los costos por disparidad se elige aquella que tiene un mayor o menor coste; por ejemplo: si la medida de similitud es SDC o SDA se elige la disparidad que tiene un menor costo, sin embargo si se usa CCN se elegirá aquella con un mayor costo.

El esquema de selección anterior es conocido como "el ganador se lleva todo", el cual tiene el problema de que no toma en cuenta los costos de los vecinos en un mismo nivel de disparidad. Para poder tomar en cuenta los costos de los vecinos, comúnmente se lleva a cabo el denominado proceso de "agregación" el cual consiste en convolu-

cionar cada nivel de disparidad con filtro A, el cual comúnmente es un filtro promedio aunque en la literatura contemporánea es común encontrar el uso de filtros cuyos pesos varían de forma adaptativa (cómo se mencionó en el estado del arte).

La principal razón para realizar el proceso de agregación es que se hace la suposición de que aquellas disparidades en el mismo nivel, que se encuentran dentro del mismo vecindario deben tener un costo muy similar y por lo tanto afectan de alguna forma a sus vecinos.

Las principales ventajas de los algoritmos de correspondencia local es que son muy fáciles de implementar y son rápidos, por otro lado la desventaja que tienen es que son propensos a tener errores en la estimación de la correspondencia debido a que al usar ventanas o líneas completas no consideran toda la imagen de referencia o la de búsqueda.

Correspondencia global

Los algoritmo de correspondencia global tiene como objetivo hacer uso de toda la imagen de referencia para poder estimar la disparidad. Estos algoritmos surgen como respuesta a las limitaciones de los algoritmos de correspondencia local y hoy en día son los que tienen mayor precisión.

La forma en que la estimación de correspondencia global funciona es plantear el problema como uno de optimización generalmente como uno de minimización de energía, como por ejemplo:

$$E(\theta) = E_{datos}(\theta) + E_{suavidad}(\theta) \tag{3.24}$$

Donde θ es un valor de disparidad y $E_{datos}(\theta)$ es el costo de la disparidad el cual es comúnmente expresado como:

$$E_{datos} \sum_{(x,y)} C(x, y, \theta(x, y)) \tag{3.25}$$

Tal que $C(x, y, \theta(x, y))$ es una función de costo y $\theta(x, y)$ un nivel de disparidad, por obvias razones la función de costo no puede ser CCN en su lugar es común usar únicamente las diferencias al cuadrado o diferencias absolutas.

En cuanto a $E_{suavidad}$ es una restricción que se impone para que los valores de disparidad varíen de manera suave tomando en cuenta los valores de intensidad de la imagen, por lo que $E_{suavidad}$ generalmente se expresa como:

$$E_{suavidad} \sum_{(x,y)} f(\theta(x,y) - \theta(x+1,y)) + f(\theta(x,y) - \theta(x,y+1)) \tag{3.26}$$

En ec.(3.26), f es una función que crece de manera monotónica.

Cabe destacar que $E_{suavidad}$ también puede ser modificada para que se incorpore información de los gradientes de la imagen de referencia.

Una vez se ha definido la función de energía a minimizar queda por emplear algún método para dicha tarea. Los algoritmos de optimización global hace uso generalmente de técnicas como "cortes en grafos", campos aleatorios de Markov o "propagación de creencia", para una descripción profunda de estos métodos ver [11].

3.8.3. Reconstrucción empleando profundidad a partir del enfoque.

A diferencia de los algoritmos que usan visión estereoscópica, la obtención de la profundidad a partir del enfoque consiste en tomar múltiples imágenes de una misma escena u objeto con un diferente grado de enfoque del lente de la cámara que captura la escena, de este modo se logra que la escena sea vista de manera borrosa en un determinado grado, de acuerdo a [14] la relación existente entre la distancia focal y el grado en que un objeto localizado a una determinada profundidad se ve borroso está dado por la ec(3.27):

$$\frac{1}{f} = \frac{1}{D_i} + \frac{1}{D} \tag{3.27}$$

Esta ecuación también es conocida como ley del lente delgado, donde D_i es la distancia a la que se encuentra un objeto con respecto a lente de la cámara, D es la distancia del lente al plano de la imagen y f es la distancia del lente hacia el sensor CCD que captura la imagen.

Esta ecuación modela la situación en la que un objeto está perfectamente enfocado, la fig.(3.9) representa la situación en la que un objeto está fuera de foco.

En figura (3.9) se puede apreciar que el objeto fuera de foco se mapea correctamente en el plano de la imagen I' mientras que en la imagen I un punto del objeto se ha mapeado como un circulo, dicho círculo representa niveles de intensidad en una imagen, en la práctica este círculo varía su intensidad del centro hacia su circunferencia por este motivo es común modelarlo como una campana de Gauss dada por la ec.(3.28).

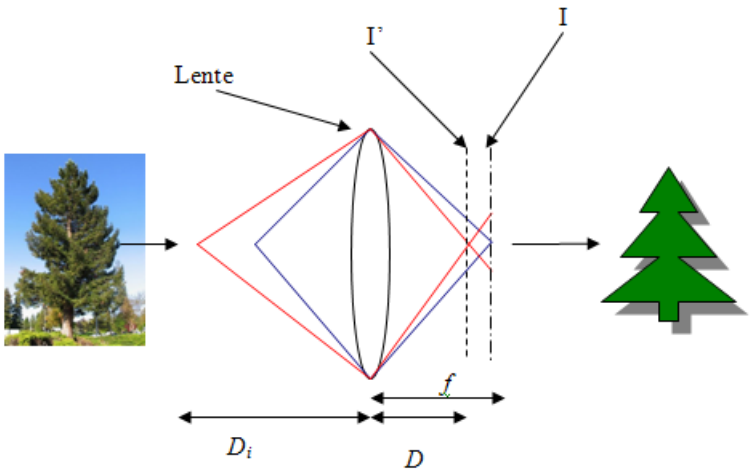


Figura 3.9: Representación de la captura de un objeto fuera de foco

$$h(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{3.28}$$

De modo tal que la intensidad de un punto en la imagen I estaría dado por:

$$I(x,y) = h(x,y) * I'(x,y) \tag{3.29}$$

Es decir que cada una de las imágenes capturadas presentará un distinto grado de borrosidad conforme el lente haya sido ajustado o

la distancia a la que se encuentre el objeto haya cambiado, es decir que características de interés como zonas de alta frecuencia en la imagen (i.e. esquinas) serán afectadas, como en la fig.(3.10).

Figura 3.10: Imágenes capturadas con diferente distancia focal

Por lo mencionado anteriormente para identificar las regiones de interés al reconstruir una escena se requiere del uso del operador de gradiente, el resultado del operador gradiente indicará las regiones donde hubo una mayor respuesta, posteriormente esta información se empleará para inferir la profundidad aproximada de un determinado punto en la escena u objeto, un punto muy importante de destacar de este método es que debe ser calibrado empleando objetos a una distancia conocida y medir los cambios en el gradiente a distancias conocidas, ejemplos ilustrativos del empleo de la técnica descrita se pueden encontrar en [37] y [38].

3.9. Borrosidad en las imágenes

Como se menciono en el apartado anterior cuando se toma una fotografía de un objeto, los puntos que están fuera de foco son mapeados como un círculo de intensidades en la imagen y como resultado la imagen del objeto (conjunto de puntos fuera de foco) se ve "borrosa". También existe la situación en la que un objeto se ve borroso debido a que este o la cámara estaban en movimiento, esta situación en particular no será discutida.

Recordando las ecuaciones (3.28) y (3.29), surge una duda muy importante es decir no se sabe ni los pesos ni el tamaño del kernel que produce la borrosidad en la imagen, es de hacer notar que dicho kernel no actúa sobre toda la I' si no que únicamente en algunas regiones más aun si existen más puntos fuera de foco cada uno tendrá asociado regiones borrosas en la imagen I y por tanto un kernel diferente.

Entonces tomando en cuenta lo anterior en una imagen I habrá regiones borrosas que son provocadas por kerneles de distinto tamaño.

3.10. Nubes de puntos y modelos 3D

El resultado de la obtención de la profundidad es un conjunto de puntos en $\mathbb{R}^3$ que no tienen conectividad, a esto se le denominara nube de puntos la cual será definida como:

$$P_i = \{a_1, a_2, a_3, ..., a_i\} \tag{3.30}$$

donde:

- P_i es la nube de puntos.

- i es el número de puntos en la nube de puntos.

- a_k es un punto en $\mathbb{R}^3$, para $1 \leq k \leq i$

A pesar de que P_i es una representación de la escena reconstruida, dicha representación no es la mejor si se requiere desplegarla como un gráfico por generado computadora. Esto es principalmente por dos razones, la primera es que el hardware moderno de gráficos está hecho para dibujar triángulos de manera más eficiente, la segunda razón es que al aplicar una transformación como el escalamiento sobre P_i y posteriormente desplegar el resultado, lo que se obtendría es un conjunto de puntos muy separados entre sí que se asemejan poco a la escena que se quiere reconstruir.

Para resolver los problemas anteriores se hace necesaria otra representación de la escena reconstruida, a esta representación se le denominara modelo 3D, cuya principal característica será que los

puntos de P_i estarán agrupados como n-tuplas que forman polígonos convexos, al modelo 3D se lo definirá como:

$$M_p = \{\rho_1, \rho_2, \rho_3, ..., \rho_p\} \tag{3.31}$$

donde:

- M_p es el modelo 3D.
- p es el número de polígonos en el modelo.
- ρ_k es una n-tupla de puntos de P_i que forman un polígono convexo, para $1 \leq k \leq p$

Las ventajas de esta representación son que: es posible calcular un vector normal para cada polígono con lo cual se puede calcular iluminación y sombras, es posible poner texturas al polígono, transformaciones como el escalamiento resultaran únicamente en polígonos más grandes sin perder la forma del objeto representado por M_p.Abajo se muestra la diferencia entre P_i y M_p.

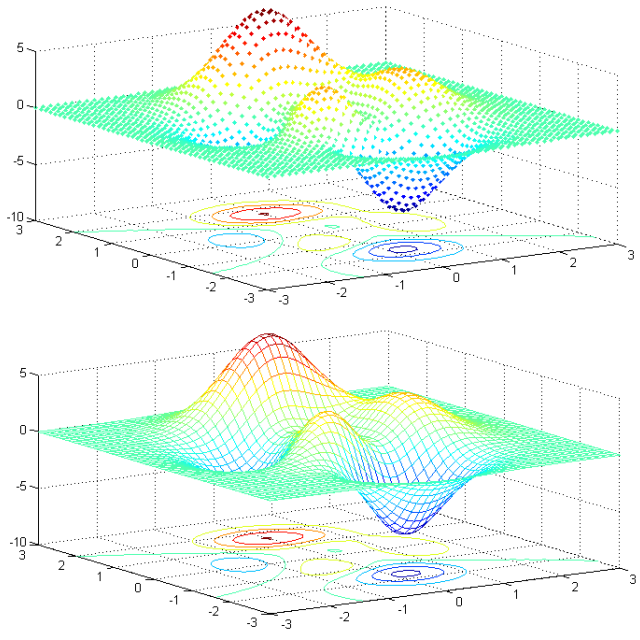


Figura 3.11: Diferencia entre P_i y M_p

Debido a que el hardware moderno para graficar dibuja de manera más eficiente triángulos que cualquier otro tipo de polígono, se reemplazará a M_p por M_T, con T como el número de triángulos; de aquí en adelante se referirá a los triángulos como polígonos.

3.11. Proceso de generación de modelos 3D

Una vez obtenido el mapa de profundidad de una escena, resulta de gran interés convertir dicho mapa en un modelo 3D, el cual pueda ser utilizado por ejemplo: para la exploración de un escenario por un robot, visitas virtuales, generación de mapas o georeferenciado de objetos por citar algunos.

El proceso mediante el cual se transforma una nube de puntos P_i en un modelo M_T puede ser interpretado como:

$$F(P_i) \Longrightarrow M_T \tag{3.32}$$

donde:

- P_i es la nube de puntos.
- M_T es el modelo 3D.
- F es la función que conecta a los puntos en P_i

La función $F(X)$ conecta a los puntos de puntos en P_i en una 3-tupla (a_i, a_j, a_k) de modo tal que se satisface un predicado lógico λ, cabe destacar que cualquier componente de un 3-tupla ρ_j se puede encontrar en otra 3-tupla ρ_k es decir que los vértices pueden ser compartidos entre polígonos, la fig.(3.12) representa como seria M_T a partir del P_i mostrado en la fig.(3.11).

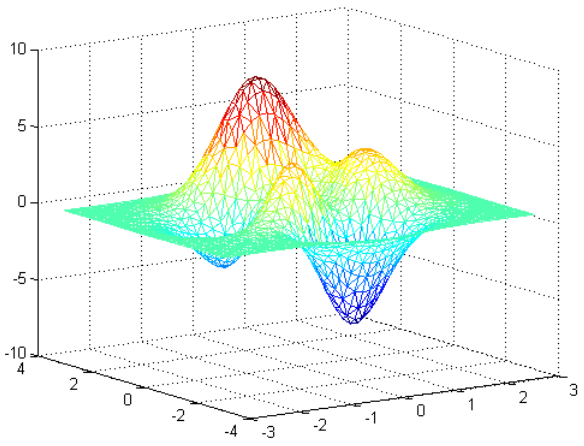


Figura 3.12: Representación de M_T

3.12. Triangulación de Delaunay

Hasta este momento no se ha mencionado como se construye un modelo M_t a partir de una nube de puntos P_i, para poder realizar dicha construcción hay que plantear dos preguntas importantes:

i) ¿Qué criterio debe satisfacer M_t para considerarlo un modelo?

ii) ¿Qué que procedimiento hay que llevar a cabo para construir M_t?

El criterio que deberá satisfacer M_t será el de la triangulación de Delaunay.

Para efectos prácticos del presente trabajo se considerara a la triangulación de Delaunay como un proceso mediante el cual a partir de un conjunto de puntos P_i se puede generar un modelo M_t, recibe su nombre en honor al matemático ruso Boris Nikolaevich Delone quien invento dicha triangulación en 1934. La triangulación de Delaunay tiene la característica de que cada uno de los triángulos que componen a M_t estará circunscrito en un circulo y cada circulo contendrá únicamente un triángulo, es decir que si M_t satisface este

criterio se lo considerará un modelo, es conveniente usar este criterio ya que proporciona un mecanismo de decisión.

El procedimiento para pasar de una nube de puntos a un modelo empleando el criterio de la triangulación de Delaunay se puede llevar a cabo de al menos dos formas:

i) **Algoritmos por incrementos**

En este tipo de algoritmos se van agregando puntos de manera gradual, se revisa si se cumple el criterio de Delaunay, en caso de que se cumpla se agrega otro punto, si no se cumple se reagrupa la triangulación para que cumpla el criterio, y repite el proceso agregando otro punto. En estos algoritmos es común ordenar los puntos con respecto a alguna coordenada, obsérvese por cada punto que se agrega hay que reordenar la triangulación de modo que se requieren $O(n^2)$ verificaciones del criterio de Delaunay.

ii) **Algoritmos divide y vencerás**

En este tipo de algoritmos la nube de puntos P_i se divide en dos de manera recursiva y se crea un triangulación en cada subdivisión de modo que se cumpla con el criterio de Delaunay, al final se juntan los resultados y se obtiene el modelo M_t, el primer algoritmo de este tipo fue propuesto por [41] y requiere $O(nlog(n))$ operaciones.

Excelentes ejemplos que emplean la triangulación de Delaunay así como algoritmos que buscan formas óptimas de encontrar dicha triangulación se pueden encontrar en [42],[43],[44].

3.13. Reconocimiento de patrones

Una de las características del aprendizaje es la capacidad de recordar, situaciones, lugares, rostros, etc. Esta capacidad es sin duda uno de los grandes objetivos que ha intentado alcanzar la Inteligencia artificial por mucho tiempo, entonces el reconocimiento de patrones será el área de la Inteligencia artificial que se dedica al diseño de algoritmos que permitan a las computadoras aprender a partir de datos que se les proporcionen, ver [52], [53].

Para denotar que un algoritmo es capaz de "recordar" un "objeto" o "situación" con base en sus "características" como similar a otros objetos o situaciones, en el reconocimiento de patrones de usan los términos clasificación ,patrón ,rasgo y clase respectivamente donde:

i) **Rasgo**

Es una característica que es particular de un objeto o situación y que ayuda a diferenciarlos de otros, por ejemplo considérese una naranja un rasgo particular puede ser su color, en la practica un rasgo en la computadora puede ser numérico o simbólico (caracteres, frases, etc.).

ii) **Patrón**

Es un conjunto de rasgos que es particular de un objeto o situación, por ejemplo una naranja podría ser caracterizada por: color, tamaño y textura, en la notación de [45] un patrón es un vector, como se muestra abajo.

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \tag{3.33}$$

donde la componente x_i es un rasgo.

iii) **Clase**

Es un conjunto de patrones cuyos rasgos representan al mismo objeto o situación, es decir es un conjunto de muestras, empleando la notación de [45]:

$$C = \{X_1, X_2, ..., X_n\} \tag{3.34}$$

donde X_i es un patrón.

iv) **Clasificación**

Es el mecanismo de decisión mediante el cual dado un patrón X desconocido se lo asocia como perteneciente a una clase C_i.

El funcionamiento de un sistema que haga uso del reconocimiento de patrones (RP) tiene esencialmente dos partes:

i) **Etapa de entrenamiento**

En esta etapa se provee al sistema de RP con un conjunto de entrenamiento que sirva para aprender a diferenciar las distintas clases que se le presentan, a dicho conjunto se lo denominara conjunto fundamental el cual será denotado por:

$$C = \{C_1, C_2, ..., C_n\} \tag{3.35}$$

donde C_i es una clase.

Para poder construir un conjunto fundamental se requiere, primero extraer un rasgos del objeto o situación de estudio, segundo seleccionar los rasgos que sean más representativos que caractericen mejor al objeto o situación y finalmente llevar a cabo el aprendizaje el cual es llevado a cabo empleando un algoritmo en especifico.

ii) **Etapa de recuperación**

En esta etapa al sistema de RP se le presenta un patrón X desconocido que no necesariamente forma parte del conjunto fundamental y el sistema debe clasificarlo de acuerdo a la clase a la que pertenece. Es obvio en este punto que dicho patrón X debe ser el resultado de haber extraído rasgos de un objeto o situación y los rasgos empleados deben ser los mismo que se emplearon en el entrenamiento, entendiéndose por los mismo como que un rasgo en el patrón X representa a lo mismo que el rasgo de un patrón que está en el conjunto fundamental.

Una parte de suma importancia en la extracción de rasgos es conocer si las clases del conjunto fundamental (formado por los patrones con determinados rasgos) son linealmente separables, es decir si es posible definir una frontera que separe a dos o más clases, la importancia radica en que si un conjunto de clases no es linealmente separable significa que al menos dos clases pueden ser confundidas, la figura de abajo muestra un ejemplo empleando un conjunto de patrones de pruebas de la base "Planta iris", ver [56].

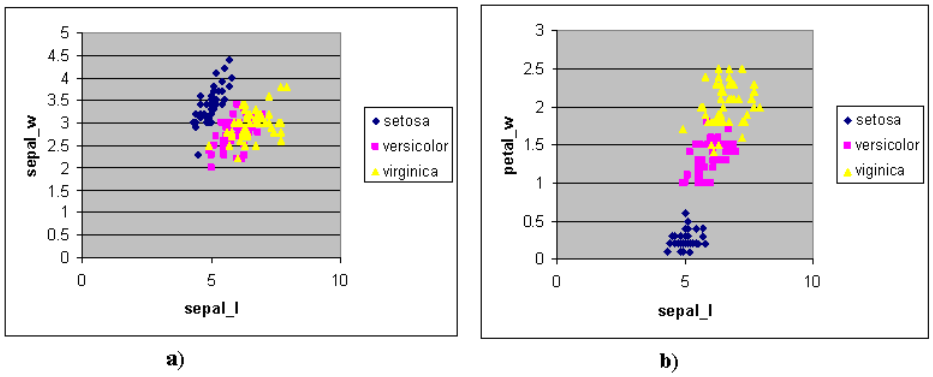


Figura 3.13: Comparación de dos patrones con rasgos distintos, a)largo del sépalo vs ancho del sépalo, b)largo del sépalo vs ancho del pétalo

De esta figura es claro que una mejor selección de rasgos resulta en la facilidad de distinguir las tres clases (es decir la gráfica b)), también es claro que hay dos clases que a pesar de elegir mejores rasgos siempre están mezcladas, para resolver dicho problema

se podrían seguir probando mejores rasgos o aumentar el número de características, aunque no hay garantía de que esto resolvería el problema de manera completa.

Es importante resaltar que una frontera de decisión, puede ser n-dimensional en cuyo caso es un híper-plano también la frontera de decisión puede ser un área o híper-volumen que rodee a las clases, este es un problema que en la actualidad es tema de estudio.

Otro concepto de gran importancia en RP es el tipo de aprendizaje que se emplea, es decir como aprende el sistema, esto se puede dividir en dos categorías:

Aprendizaje supervisado

El sistema es entrenado con un conjunto fundamental donde se sabe a qué clase pertenece cada patrón.

Aprendizaje no supervisado

El sistema es entrenado con un conjunto en el cual no se sabe que clases existen, en este caso en particular se espera que el sistema de RP de forma automática clasifique de manera automática las diferentes clases que pueden existir en el conjunto que se le presentó al sistema, dicha tarea recibe en Ingles el nombre de "Clustering".

Ahora bien para poder evaluar la efectividad de un sistema de RP es necesario medir el porcentaje de elementos correctamente clasificados, para dicho efecto es necesario emplear un conjunto de patrones de prueba, dicha medición recibe el nombre de porcentaje de recuperación (recall en Ingles).

3.14. Enfoques de reconocimiento de patrones

Dado que la presente tesis pretende plantear el problema de la obtención de la profundidad de una escena como un problema de reconocimiento de patrones se hace necesario describir los enfoques más comúnmente empleados para el problema de aproximación de funciones o reconocimiento de patrones en imágenes, en el caso particular del trabajo presente la estimación de profundidad; por este motivo esta sección se concentrara únicamente en dar una descripción de los siguientes enfoques:

i) Estadístico

ii) Neuronal

iii) Asociativo

Es importante destacar que estos tres enfoques son los más reportados en la literatura, ejemplos muy representativos se pueden encontrar en [52], [53],[54],[55],[62],[63].

3.15. Enfoque estadístico/probabilístico

Este enfoque basa su funcionamiento haciendo la consideración de que las distintas clases de patrones que existen pueden ser modeladas como una función de distribución de probabilidad, es decir que se debe realizar un estudio estadístico para hallar, la media, mediana, desviación estándar de una clase en particular .

La manera en la que funciona este enfoque es mediante el cálculo de probabilidades condicionales, es decir las probabilidades de que un patrón X pertenezca a una clase W_i. Debido a esta característica el principal sustento teórico de este enfoque es el teorema de Bayes, el cual está dado por:

$$P(C_i|X) = \frac{P(X|C_i)}{P(X)} \tag{3.36}$$

Donde:

- C_i la clase i.

- X un patrón cualquiera.

Ahora bien para poder clasificar un patrón como perteneciente a una clase C_i o C_j se debe cumplir una de las siguientes condiciones[52]:

i) si $P(C_i|X) > P(C_j|X)$, entonces X pertenece a W_i

ii) si $P(C_j|X) > P(C_i|X)$, entonces X pertenece a W_j

iii) $P(C_i|X) \sim P(C_j|X)$, entonces X no pertenece a ninguna.

De esto último se debe notar que este mecanismo de decisión debe ser aplicado sobre todas las clases del conjunto fundamental.

En este punto deben ser claras dos cosas acerca de este enfoque, primero se requiere un número suficientemente grande de muestras para caracterizar de la mejor manera posible a cada clase C_i, en otras palabras implicaría tener conocimiento demasiado extenso de un problema en particular, segundo existe alguna subclase de patrones que no fue incluida en una clase C_i y se presenta un patrón X que pertenezca a dicha clase esto podría resultar en la imposibilidad de clasificarlo de manera correcta.

3.16. Enfoque neuronal

Una de las metas de la Inteligencia Artificial es emular capacidades humanas como el funcionamiento del cerebro, por este motivo el enfoque neuronal debe su nombre a que basa su funcionamiento en el modelo matemático de neurona propuesto por primera vez por Warren McCulloch y Walter Pitts en 1943 [57], en el cual se modela el comportamiento de una neurona como la contribución de una serie de pesos que actúan como una ponderación de las señales eléctricas que reciben en sus dendritas y la interacción de dicha ponderación con un umbral de activación que permite la excitación del axón, esta interacción es en realidad una función de transferencia, es decir la que permite la activación del axón.

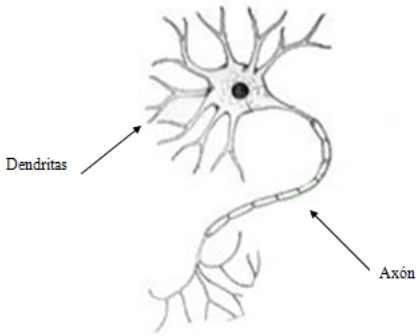


Figura 3.14: Representación de una neurona biológica

En este enfoque el funcionamiento depende de las neuronas artificiales, las cuales son interconectadas para formar las llamadas redes neuronales artificiales (RNA), el efecto que se logra al interconectar elementos simples es la contribución de cada uno de ellos para creación de un tipo de memoria que emula adquirir conocimiento.

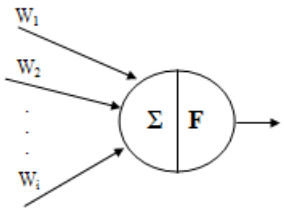


Figura 3.15: Representación McCulloch y Pitts de una neurona

La primera RNA que se desarrollo fue el "perceptron"(propuesto por Frank Rosenblat en 1985), el cual consiste en un arreglo de neuronas, que comparten sus entradas, ver fig.(3.16),su método de entrenamiento consiste en los siguientes pasos:

1.- Por cada neurona, hasta que no haya error en la clasificación:

a.- Inicializar el umbral b_i y vector de pesos W_i con valores aleatorios.

b.- Elegir un patrón P_i.

c.- Evaluar el patrón, si el error e al clasificar no es cero, entonces actualizar $W_i = W_i + eP_i$ y $b_i = b_i + e$.

2.- Regresar al paso 1.

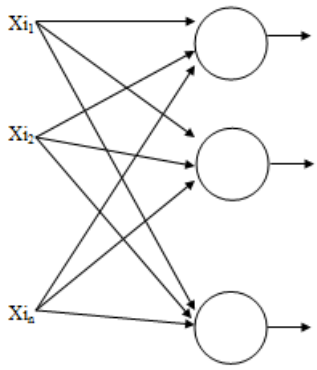


Figura 3.16: Perceptron

De esto último es necesario hacer tercero observaciones, primero una clasificación con error cero no siempre será posible por lo que es preferible definir un "ϵ"mínimo que se debe satisfacer para considerar que se ha hecho una clasificación correcta, para llegar a dicho "ϵ"deben transcurrir un número determinado de iteraciones a esto se le denomina "épocas" que también pueden ser empleadas como un criterio de terminación del entrenamiento, un número excesivo de épocas o un ϵ muy pequeño puede ocasionar el efecto de "sobre-entrenamiento" y pierde capacidad de generalización, tercero el perceptron trata de establecer una frontera de decisión para separar una clase de las demás, sin embargo en el caso de que las clases no sean linealmente separables el perceptron no será capaz de realizar una clasificación correcta, por ejemplo el problema de la XOR, expuesto por Marvin Minsky y Seymour Papert en su libro perceptrons de 1969.

Debido a esta limitación del perceptron las RNA son abandonadas por muchos años, hasta los años ochenta cuando John Hopfield revive el interés en las RNA, posteriormente surge el perceptron multicapa (ver fig.3.17) con el entrenamiento por propagación hacia

atrás (backpropagation) el cual fue descubierto casi de manera simultánea por Rumelhart,Hinton y Williams en 1986, David Parker en 1985 y Yann Le Cun en 1986, para mayor información ver [50], ahora el perceptron puede tener múltiples capas[2], es decir conjuntos sucesivos de neuronas que permiten aprovechar salidas de neuronas anteriores, de este modo es posible atacar problemas que no son linealmente separables. El algoritmo de propagación hacia atrás funciona de la siguiente manera:

- Elegir un parámetro de velocidad r.
- Hasta que el desempeño sea satisfactorio,
- Para cada patrón de entrada P_i,
  - Computar la salida de la red.
  - Computar β para las neuronas en la capa de salida mediante $\beta_z = d_z - o_z$.
  - Computar β para las demás neuronas $\beta_j = \sum_{i=1}^{n} w_{jk} \frac{df(\sigma_k)}{d\sigma} \beta_k$.
  - Computar el cambio en los pesos mediante $\Delta w_{ij} = r o_j \frac{df(\sigma_j)}{d\sigma} \beta_j$.
- Sumar todos los cambios en los pesos para todos los patrones de entrada y actualizar los pesos.

donde:

- $f(\sigma)$ representa la función de transferencia empleada por las neuronas.
- σ_k representa la entrada que recibe una neurona en la capa k.
- w_{ij} representa el peso de la conexión de la salida de una neurona en la capa i y la entrada en una neurona en la capa j.
- d_z representa el valor esperado en la capa de salida.
- β_i representa la propagación del error hacia atrás.
- o_i representa la salida en la capa i.

[2]las capas que están entre la entrada y la salida se denominan capas ocultas

Del algoritmo de "backpropagation" nótese que no se emplean umbrales y únicamente se actualizan los pesos de conexión entre las neuronas, también nótese que al igual que en el perceptron de una sola capa se intenta obtener el error mínimo, es decir que el error lo podemos ver como una híper-superficie donde se intenta hallar un mínimo global, debido a este punto es muy importante mencionar que dicho algoritmo puede quedar atrapado en un mínimo local, la función de transferencia juega un papel muy importante en el desempeño de la red, ejemplos muy empleados son: $\frac{e^{2\sigma}-1}{e^{2\sigma}+1}$ (tangente hiperbólica), $\frac{1}{1+e^{-\sigma}}$, lo que se busca en una función de transferencia es que pueda actuar como un umbral.

Otras variantes del algoritmo de "backpropagation" buscan mejorar la velocidad con la que se llega a una solución por ejemplo[51]:

- Algoritmo de razón de aprendizaje variable.
- Algoritmo de momento.
- Algoritmo de Levenberg-Marquardt.
- Algoritmo de gradiente conjugado.

Finalmente un punto importante por mencionar es que el desempeño de una RNA depende tanto del número de épocas en que se llevo a cabo el entrenamiento como del número de neuronas y el tipo de conexión que existe entre ellas, estos tres aspectos siguen siendo un tema de investigación actual ya que no se ha encontrado un criterio que especifique restricción sobre ellos.

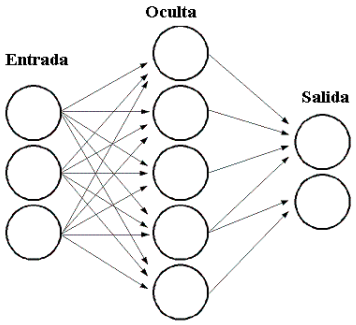


Figura 3.17: Representación de un perceptron multi-capa con una capa oculta

3.17. Enfoque asociativo

Como fue mencionado con anterioridad una de las características que es deseable un sistema de RP es la capacidad de recordar, en otras palabras dado un patrón X asociarlo a un patrón Y, es precisamente esta capacidad la que proveen las "memorias asociativas", se puede ver a una memoria asociativa M como una caja negra (ver fig.3.18) la cual recibe un patrón de entrada X y responde con un patrón de salida Y.

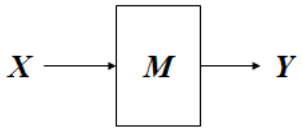


Figura 3.18: Representación de una memoria asociativa

Donde:

- X es un patrón desconocido.

- Y es un patrón que se encuentra almacenado en la memoria M.

La memoria asociativa es una matriz M, y los patrones X y Y son vectores, de manera formal la operación de una memoria asociativa está dado por:

$$y^{\mu} = M \cdot x^{\mu} \tag{3.37}$$

donde y^{μ} y x^{μ} son patrones que están asociados, es decir que el conjunto de patrones que están asociados forman un conjunto fundamental de la forma $C = \{(x^{\mu}, y^{\mu}) | \mu = 1, 2, \ldots, p\}$ siendo p el número de asociaciones existentes. En las memorias asociativas existen dos conceptos muy importantes , el primero es el de una memoria "auto-asociativa" el cual es un caso en el que $x^{\mu} = y^{\mu}$, el segundo caso es el de la memoria "hetero-asociativas" es decir $x^{\mu} \neq y^{\mu}$.

Existen distintos tipos memorias asociativas, la principal diferencia que existe es la manera en que operan en su fase de entrenamiento (generación de la memoria) y la fase de recuperación, a continuación se provee una breve descripción de los modelos que han tenido mayor relevancia en el desarrollo del enfoque asociativo.

Learnmatrix

Este modelo de memoria asociativa fue desarrollado por Karl Steinbuch en 1961, la principal característica es que únicamente puede almacenar patrones binarios, es decir cuyos rasgos están únicamente en el dominio $B = \{0, 1\}$. La fase de aprendizaje de esta memoria está dada por $M + \Delta M$, donde:

$$\Delta M_{ij} = \begin{cases} \epsilon & \text{si } x_i^\mu = 1 \text{ y } y_j^\mu = 1 \\ -\epsilon & \text{si } x_i^\mu = 0 \text{ y } y_j^\mu = 1 \\ 0 & \text{en otro caso} \end{cases} \tag{3.38}$$

Donde M es una matriz que es inicializada con ceros, es decir que ΔM es una matriz que se genera por cada asociación en el conjunto fundamental, la fase de recuperación está dada por:

$$y_i^\mu = \begin{cases} 1 & \text{si } \sum_{j=1}^n m_{ij} \cdot x_i^\mu = \bigvee_{h=1}^p (\sum_{j=1}^n m_{hj} \cdot x_i^\mu) \\ 0 & \text{en otro caso} \end{cases} \tag{3.39}$$

Linear asociator

Esta propuesta fue desarrollada por James A. Anderson y Teuvo Kohonen de manera independiente alrededor del año de 1972, a diferencia de la "learnmatrix" este modelo tiene la capacidad de trabajar con patrones cuyos rasgos pertenece al dominio de los número reales, la fase de aprendizaje está dada por $M + \Delta M$, donde:

$$\Delta M = y^\mu \cdot (x^\mu)^t \tag{3.40}$$

En (3.41) y^μ es un vector columna y $(x^\mu)^t$ es un vector fila, ΔM es solamente la matriz que representa la asociación (x^μ, y^μ),es decir que se deben sumar todas las matrices generadas para las p asociaciones. La fase de recuperación está dada por:

$$M \cdot x^\omega = y^\omega \cdot ((x^\omega)^t \cdot x^\omega) + \sum_{\mu \neq \omega} y^\mu \cdot ((x^\mu)^t \cdot x^\omega) \tag{3.41}$$

De la ec.(3.41),ver [59], la recuperación de un patrón es perfecta únicamente posible si el patrón x^ω es un vector normalizado y si al mismo tiempo es normal hacia todos los x^μ del conjunto fundamental.

Memoria Hopfield

Esta memoria que fue desarrollada por Jhon Hopfield en 1986 fue originalmente concebida como una red neuronal recurrente, es decir que tienen conexiones tanto hacia adelante como hacia atrás, fue probado que dicha red neuronal funcionaba como una memoria auto-asociativa. En este modelo los patrones deben estar en el dominio $A = \{-1, 1\}$, la fase de aprendizaje está dada por $M + \Delta M$, donde:

$$\Delta M = y^{\mu} \cdot (x^{\mu})^{t} - I \tag{3.42}$$

de la ec.(3.42) nótese que se substrae la matriz identidad, también note que a pesar de que el dominio de los patrones es $\{-1, 1\}$ la matriz M puede tener números enteros. La fase de recuperación de esta memoria consiste en dos etapas, en la primera etapa se realiza $M \cdot x^{\omega} = y'^{\omega}$, debido que el resultado no necesariamente da como resultado un patrón en el dominio $\{-1, 1\}$ se requiere aplicar la siguiente regla:

$$y_i^{\omega} = \begin{cases} +1 & \text{si } y'^{\omega}_i > 0 \\ -1 & \text{si } y'^{\omega}_i < 0 \end{cases} \tag{3.43}$$

Memorias morfológicas

Este modelo fue producto del trabajo de Ritter, Sussner y Díaz-de-León, el nombre de estas memorias proviene del hecho de que emplean operadores de mínimo y máximo y la relación de estos con los operadores morfológicos de apertura y dilatación, también pueden trabajar con patrones reales.

En este modelo se definen memorias tanto para el caso hetero-asociativo y auto-asociativo, también se definen dos tipos de memorias las max y min, estas deben su nombre al papel que juegan los operadores máximo ($\bigvee$) y mínimo ($\bigwedge$) respectivamente. las fases de aprendizaje están dadas por $M = \bigvee_{\mu=1}^{p} \Delta M_{\mu}$ para memorias max y $W = \bigwedge_{\mu=1}^{p} \Delta W_{\mu}$ para me-

morias min, donde:

$$\Delta M_\mu = y^\mu \triangle (x^\mu)^t$$
$$\Delta W_\mu = y^\mu \triangledown (x^\mu)^t$$

en donde $\triangledown$ representa la aplicación del producto máximo el cual es $\Delta M_{ij} = y_i^\mu - x_j^\mu$ y $\triangle$ representa aplicación del producto mínimo el cual es $\Delta W_{ij} = y_i^\mu - x_j^\mu$, esto para el caso de vectores columna por vectores fila. Para la fase de recuperación se realiza:

$$y = M \triangle (x^\omega)$$
$$y = W \triangledown (x^\omega)$$

donde $\triangle$ representa la aplicación del producto mínimo tal que $y_i = \bigwedge_{j=1}^{n} (M_{ij} + x_j^\omega)$ y $\triangledown$ representa la aplicación del producto máximo tal que $y_i = \bigvee_{j=1}^{n} (W_{ij} + x_j^\omega)$.

3.18. Conclusiones

En este capítulo se han introducido los conceptos, necesarios del funcionamiento de herramientas cruciales por emplear en la presente tesis de particular importancia resultan la calibración de la cámara, el grado de borrosidad en las imágenes y el enfoque de RP asociativo.

La calibración de la cámara es sumamente importante porque permite establecer una relación del mundo real con las coordenadas en una imagen, lo que permite emplearla como un dispositivo de medición y más aun permite establecer sus limitaciones.

El grado de borrosidad en las imágenes proporciona un mecanismo para relacionar la profundidad a la que se localiza un objeto con lo borrosa que se ve una imagen, esto es explotado por técnicas como "Shape from focus", aunque se requieren muchas imágenes para establecer la profundidad con mucha exactitud, en el estado del arte se proporcionan algunos ejemplos.

De gran relevancia en la presente tesis será el RP que proveerá una

Capítulo 4

Metodología propuesta

La presente tesis tiene como objetivo principal realizar estimación de la profundidad de un escenario empleando dos imágenes tomadas por dos cámaras separadas entre sí, tal como se describió en el capítulo de marco teórico.

En el enfoque que se pretende desarrollar se incorporarán las siguientes técnicas:

- Uso de memorias asociativas como mecanismo para verificar la estimación de la disparidad.
- Empleo de mínimos cuadrados para propagar la estimación de la profundidad en las áreas indefinidas.

El motivo de incluir estas dos técnicas es el de investigar cómo pueden contribuir a estimar la profundidad y proveer un método alternativo al que se encuentra reportado en el estado del arte.

4.1. Principales desafíos

Para poder llevar a cabo la reconstrucción 3D de un escenario se han identificado los siguientes problemas como los obstáculos principales para el desarrollo de la tesis actual.

- Búsqueda de correspondencia entre las dos imágenes empleadas.
- Propagación de la disparidad en regiones donde no fue posible obtener una correspondencia.
- Creación de un modelo 3D a partir del mapa de profundidad que ha sido estimado.

Estos problemas son lo que siempre están presentes en cualquier proceso que estime la profundidad de un escenario mediante dos o más imágenes. El motivo principal por el que se presentan los dos primeros problemas es debido a que en la práctica no es posible encontrar una correspondencia perfecta, esto se produce por variaciones de la iluminación, ruido en los sensores de las cámaras, obstrucción de regiones en el escenario que no están presentes en las dos imágenes, obstrucción completa de una de las cámaras, etc. Por estos motivos se puede decir que en general obtener la correspondencia entre dos imágenes no es una tarea fácil y está caracterizada por la falta de información completa.

4.2. Acerca del ambiente a reconstruir

En el presente trabajo se emplearán imágenes estereoscópicas del conocido conjunto de imágenes de Middlebury [1][2][3][4],[65]; las razones para emplear este conjunto radican en que dicho conjunto ha sido cuidadosamente elaborado por sus autores, las imágenes presentan muy poco ruido, la correspondencia entre ellas es relativamente fácil de encontrar, ahorran tiempo al no tener que capturar la imágenes y lo más importante es que son un conjunto imágenes que es de referencia en el área de visión estereoscópica. La figura de abajo muestra algunas de las imágenes contenidas en el conjunto de Middlebury.

(a) Imagen dolls

(b) Imagen Moebius

(c) Imagen Reindeer

(d) Imagen Dwarves

Figura 4.1: Imágenes del conjunto Middlebury

Es de hacer notar que todas las imágenes corresponden a escenarios que no se encuentran al aire libre.

4.3. Proceso de correspondencia

El proceso mediante el cual se estima la correspondencia entre dos imágenes se realiza de acuerdo a lo mostrado en el capítulo de marco teórico, es decir:

- definir el tamaño del intervalo de búsqueda.
- elegir un tamaño de la ventana a usar para medir correspondencia.
- establecer un error máximo para la medida de correspondencia.
- escoger el tamaño de la ventana de agregación.

- Para cada pixel en la imagen izquierda
    - Se obtienen los vecinos de éste, en el caso particular de esta tesis solo se usan los vecinos que se encuentran a la izquierda y derecha.
    - Para cada pixel en la imagen derecha, cuya coordenada en Y es la misma del pixel de la imagen izquierda
        - Recorrer la imagen izquierda a lo largo de la dirección en X se obtienen los vecinos de los pixeles.
        - Realizar la medida de correspondencia de las dos ventanas centradas en los pixeles de la izquierda y derecha.
        - Guardar la medición en un arreglo temporal,.

    Realizar proceso de selección de la mejor medición de correspondencia y opcionalmente realizar proceso de agregación.

Cabe mencionar que en el presente trabajo se han probado dos medidas de correspondencia, las cuales son:

- sumatoria de diferencias al cuadrado (SDC), dada por:

$$SDC = \sum_{i=1}^{n} (Pizq_i - Pder_i)^2 \tag{4.1}$$

  Donde $Pizq_i$ es un pixel en la imagen izquierda y $Pder_i$ es un pixel en la imagen derecha.

- Correlación cruz normalizada (CCN), dada por:

$$CCN \frac{1}{n-1} \sum_{i=1}^{n} \frac{\left(Pizq_i - \overline{Pizq}\right)\left(Pder_i - \overline{Pder}\right)}{\sigma_{Pizq}\sigma_{Pder}} \tag{4.2}$$

  Donde $Pizq_i$ es un pixel en la imagen izquierda y $Pder_i$ es un pixel en la imagen derecha, y σ es la desviación estándar de cada ventana. Es muy importante resaltar que la CCN toma los valores en el intervalo $(-1, 1)$ en donde -1 significa que la ventanas comparadas son completamente diferentes y 1 significa una correspondencia perfecta.

Las imágenes siguientes muestran el resultado de emplear estas dos diferentes medidas, es de importancia resaltar que en estas imágenes no se ha empleado el proceso de agregación.

Figura 4.2: Mapa de disparidad calculado usando CCN

Figura 4.3: Mapa de disparidad calculado usando SDC

Nótese que de las imágenes mostradas la que a primera vista provee mejores resultados es la que emplea la medida de CCN, esto es debido a que la definición de CCN provee un mecanismo mediante el cual decidir si dos ventanas que se comparan son iguales, por ejemplo suponga una situación en la que una ventana de la imagen izquierda simplemente no tiene una correspondencia en la imagen derecha, si se usa SDC, es claro que elegir la mejor medición no ayudara mucho pues simplemente ninguna medición es la que se busca pues simplemente no existe una correspondencia, una clara opción sería modificar el error máximo permitido en la SDC, sin embargo esto tampoco resolvería el problema pues al hacer esto también se eliminarían correspondencias correctas.

Ahora bien, si se emplea CCN se tiene un mecanismo de decisión que permite discriminar entre ventanas que no tiene nada que ver de aquellas que en las que si existe un alto grado de correspondencia, y esto está garantizado por la definición de la CCN. Por este motivo en el presente trabajo se ha decidido emplear la CCN como medida de correspondencia.

Cabe aclarar que dicho proceso de correspondencia se realiza dos veces una tomando la imagen izquierda como referencia y otra tomando la imagen derecha como referencia (se denomina verificación de consistencia izquierda-derecha []). Una vez se han estimado los dos mapas de disparidad se verifica que las disparidad de los dos mapas coincidan, esto se realiza para detectar aquellas regiones que se encuentran obstruidas y por lo tanto se desconoce su disparidad.

4.4. Verificación de la disparidad

Una vez que se ha calculado la disparidad sería deseable algún mecanismo para poder verificar dichas estimaciones, ya que no se dispone un conocimiento previo con el cual corroborar la mediciones realizadas se ha recurrido al empleo de técnicas de reconocimiento de patrones debido a que el proceso de correspondencia bien puede ser visto como una tarea de RP donde se tiene un patrón desconocido (ventana de referencia) y un conjunto de entrenamiento (venta-

nas de con la que se compara la ventana de referencia). Elecciones comúnmente empleadas para resolver un problema de RP son redes neuronales, maquinas de soporte vectorial, clasificadores basados en métricas, memorias asociativas.

De los cuatro enfoques de RP mencionados, inmediatamente se pueden descartar las redes neuronales y maquinas de soporte vectorial pues el tiempo de su entrenamiento es variable y además no se garantiza una convergencia perfecta. Los clasificadores basados en métricas quedan automáticamente descartados por los motivos expuestos para emplear CCN.

Ahora bien las memorias asociativas morfológicas son una historia completamente diferente ya que estas tienen recuperación perfecta para el conjunto fundamental, esta ventaja resulta muy atractiva además de que el tiempo que requiere para su entrenamiento es proporcional al número de patrones en el conjunto fundamental y a su dimensión, dicho tiempo es el mismo siempre y cuando no se cambie el número de patrones en el conjunto fundamental. Otro punto muy importante que tienen las memorias asociativas morfológicas es que el resultado que se obtiene a su salida es idealmente un patrón del conjunto fundamental, lo cual es exactamente la tarea de hallar la correspondencia.

Por los motivos expuestos se ha empleado una memoria asociativa morfológica como un mecanismo para verificar los resultados del proceso de correspondencia, el proceso mediante el cual se realiza la verificación funciona de la siguiente forma:

- Para cada pixel en la imagen izquierda
- Se obtienen los vecinos de éste, en el caso particular de esta tesis solo se usan los vecinos que se encuentran a la izquierda y derecha.
- Para cada pixel en la imagen derecha, cuya coordenada en Y es la misma del pixel de la imagen izquierda
    - Recorrer la imagen izquierda a lo largo de la dirección en X se obtienen los vecinos de los pixeles.

- A cada ventana obtenida concatenar la disparidad en la que se encuentra, y guardar todas la ventanas en una memoria asociativa.
- Usar la ventana de referencia con la disparidad concatenada (la disparidad que fue calculada previamente) como patrón de entrada de la memoria asociativa.
- Realizar una diferencia de la disparidad calculada previamente con la disparidad calculada con la memoria asociativa, si la diferencia cae bajo un umbral entonces se considera la disparidad calculada como correcta, en caso contrario se marca como una región donde la disparidad es desconocida.

A continuación se muestra la comparación de las disparidades calculadas tanto empleando la memoria asociativa como el resultado de no hacerlo.

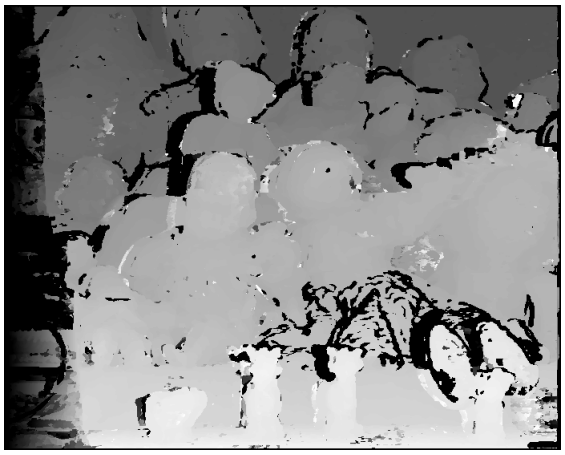


Figura 4.4: Mapa de disparidad sin verificar por la memoria asociativa

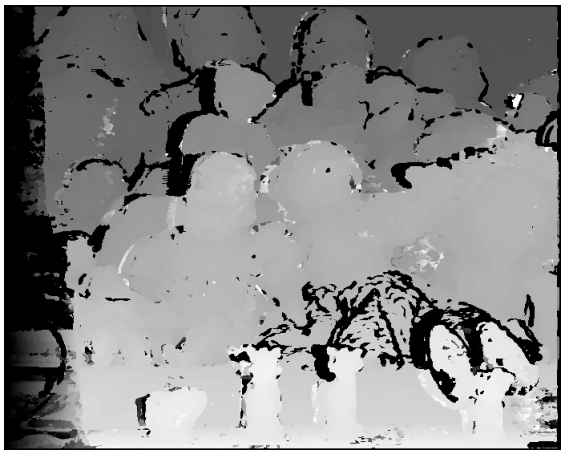


Figura 4.5: Mapa de disparidad verificado por la memoria asociativa

Note que las principales diferencias ocurren cerca de las regiones de la imagen izquierda tales que estas no existen en la imagen derecha.

4.5. Eliminación de ruido en el mapa de disparidad

Como se ha visto hasta el momento lo mapas de disparidad obtenidos contienen una gran cantidad de ruido, el ruido que se ve se encuentra principalmente en las esquinas de los objetos, esto se debe a que esas regiones son discontinuidades en la disparidad y establecer una medición correcta es muy difícil principalmente debido a obstrucciones que están presentes en una imagen pero no en otra, la imagen de abajo muestra el ruido que se contiene el mapa de disparidad. En este punto cabe hacer la aclaración que por ruido se referirá a los artefactos que se manifiestan en los mapas de disparidad como regiones que no coinciden con la disparidad que las rodea.

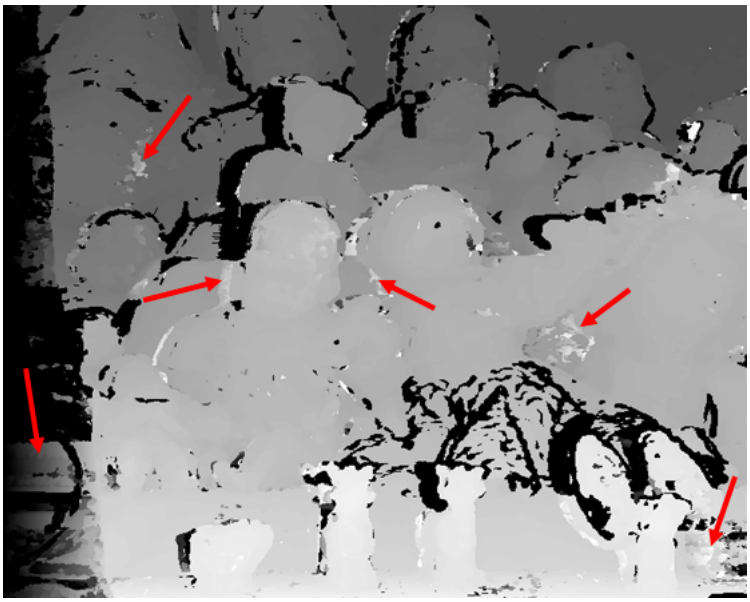


Figura 4.6: Las flechas rojas indican algunas regiones donde hay ruido

Para eliminar el ruido una opción sería emplear ya sea un filtro de mediana, promedio, máximo o mínimo, la imagen de abajo muestra el resultado de aplicar los distintos filtros de $7x7$.

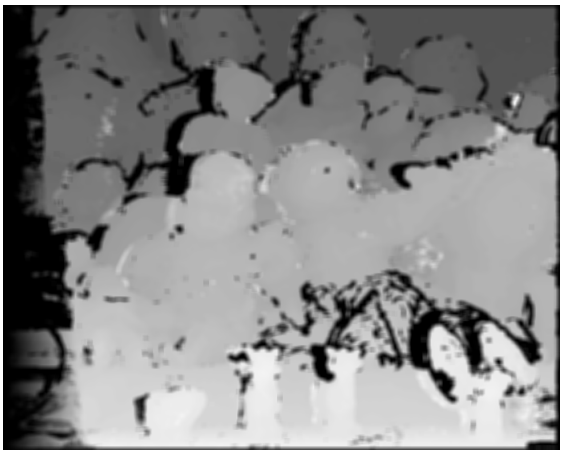


Figura 4.7: Resultado filtro promedio

Figura 4.8: Resultado filtro mediana

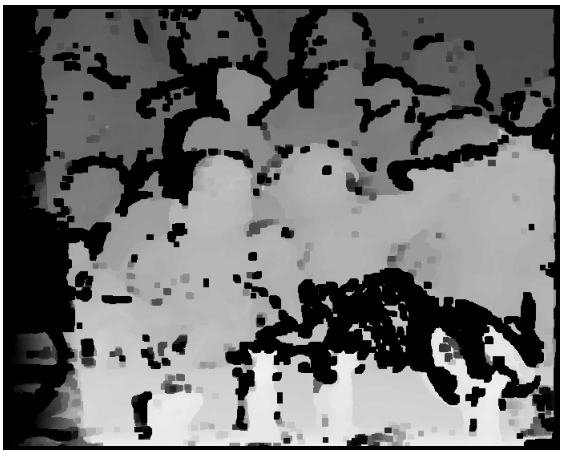


Figura 4.9: Resultado filtro mínimo

Figura 4.10: Resultado filtro máximo

Como se puede apreciar ninguno de los filtros da un resultado que elimine las regiones ruidosas, además provoca que las regiones que están correctas se "contaminen". Debido a que las regiones ruidosas se caracterizan por tener disparidades muy distintas o bien una disparidad muy distinta del resto resulta natural el tratar de identificar aquellas disparidades que no están relacionadas con las que se encuentran alrededor, con este objetivo se ha decido emplear un criterio estadístico para identificar las regiones ruidosas eliminarlas lo más posible.

El criterio que se emplea es el siguiente:

- se obtienen los vecinos alrededor de un punto en el mapa de disparidad; el punto y los vecinos deben tener una disparidad conocida.
- se calcula la media del conjunto de vecinos obtenidos.
- si el punto se encuentra más allá de un umbral establecido, entonces se elimina y se marca con una disparidad desconocida.

Una vez se aplicó el filtro propuesto (de ahora en adelante filtro estadístico) se aplica un filtro mínimo de 3x3 para eliminar aquellos pixeles ruidosos que quedan aislados. La imagen de abajo muestra el resultado de aplicar el filtro propuesto.

Figura 4.11: Resultado del filtro propuesto

Se puede notar que se pierde una gran cantidad de información del mapa de disparidad, pero se mantienen las áreas que son correctas y están libres de "contaminación" de regiones ruidosas.

4.6. Propagación de la disparidad

Debido a las regiones que quedan marcadas con una disparidad/-profundidad desconocida se hace necesario un mecanismo mediante el cual rellenar dichas regiones, en la literatura se encuentran comúnmente reportados propagación de creencia y cortes en grafos para propagar la disparidad, estos son considerados métodos globales de optimización y se encuentran entre los mejores conocidos para dicha tarea.

En el presente trabajo de tesis se ha optado por adaptar la técnica de "coloreado"propuesta en [64], dicha técnica se emplea para colorear imágenes en blanco y negro de manera automática a partir de algunas marcas hechas sobre la imagen que se quiere colorear, esta tarea está directamente relacionada con el objetivo de propagar la disparidad en las regiones desconocidas, más aun el criterio mediante el cual se realiza el proceso de coloreado toma en cuenta la intensidad de la imagen para poder propagar.

El proceso de propagación mediante la técnica de "coloreado"se realiza optimizando la siguiente ecuación:

$$D(d) = \sum_{r} \left(d(r) - \sum_{r \in N(r)} w_{rs} d(s) \right)^2 \tag{4.3}$$

Donde:

$$w_{rs} = e^{-\frac{(I(r)-I(s))^2}{2\sigma_r^2}} \tag{4.4}$$

$I(r)$ es la intensidad de la imagen de referencia.

Para propagar la disparidad de tal modo que se incluya el color, se ha modificado la ec.(4.4) incluyendo los canales de color de la imagen como se muestra en ec.(4.5):

$$w_{rs} = e^{-\frac{(R(r)-R(s))^2}{2\sigma_R^2} - \frac{(G(r)-G(s))^2}{2\sigma_G^2} - \frac{(B(r)-B(s))^2}{2\sigma_B^2}} \tag{4.5}$$

Donde R,G y B son los canales de la imagen que se emplea como referencia para propagar las medidas de disparidad, los resultados que se obtienen con ec.(4.5) difieren de los obtenidos con ec.(4.4) (más adelante se muestra un ejemplo).

El proceso de "coloreado" asume que las regiones en la imagen cuyas intensidades son similares deben tener un color similar, adaptando esta restricción al problema de propagar la disparidad se asume entonces que regiones con un color similar deben tener una disparidad similar, la figura de abajo muestra el resultado de la optimización.

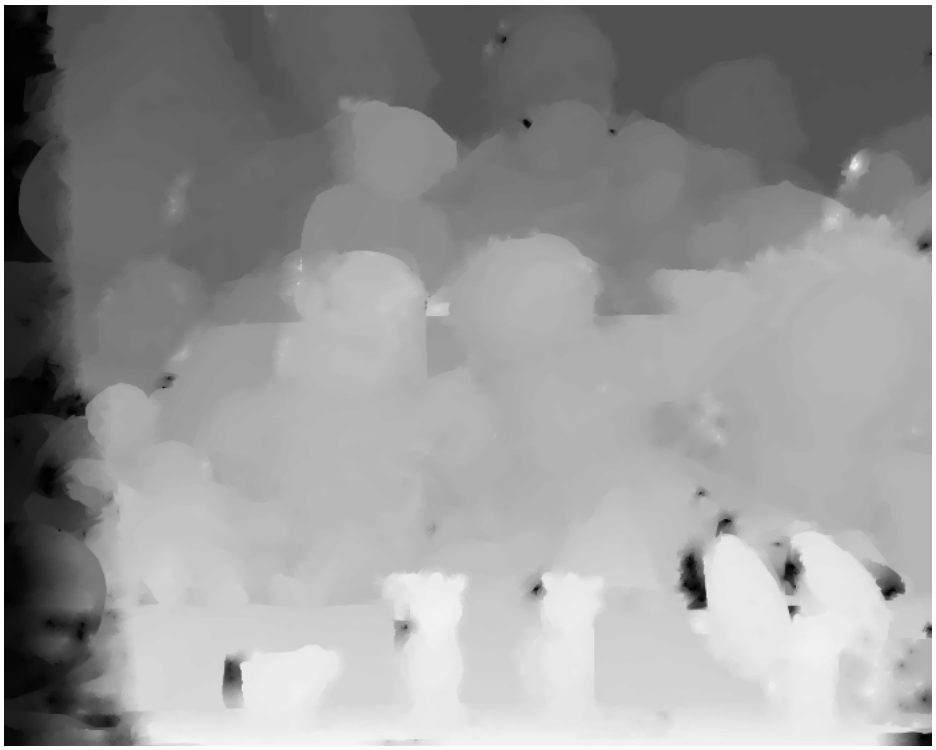


Figura 4.12: Resultado del proceso de optimización

El resultado que se obtiene es un mapa de disparidad en el que las zonas de la misma intensidad tienen disparidades similares. Se debe notar que las regiones de la imagen que se encuentran hacia la derecha son incorrectas, por este motivo al terminar el proceso de optimización una parte de la derecha se marca como disparidad desconocida y se vuelve a repetir el proceso de optimización, la imagen de abajo muestra el resultado del proceso de optimización.

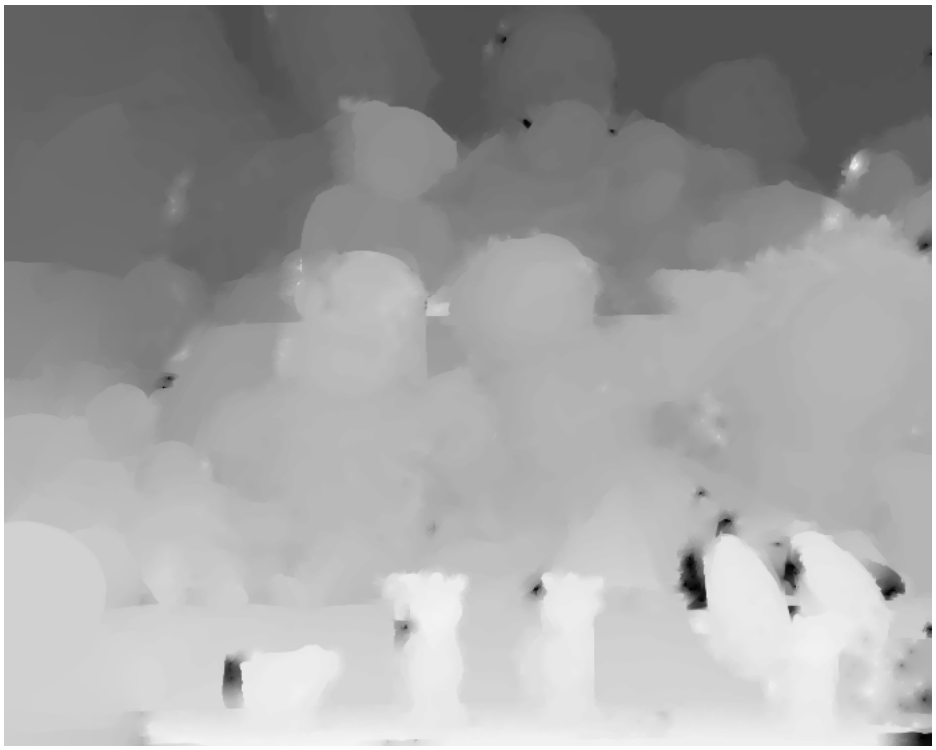


Figura 4.13: Resultado del proceso de optimización

Una observación importante que se debe hacer es que el algoritmo de coloreado es capaz de conservar la forma de los objetos lo cual es muy importante para poder usar el mapa de disparidad para realizar la reconstrucción 3D del escenario.

Como se mencionó anteriormente los resultados que se obtienen al usar las ec.(4.4) y ec.(4.5) presentan diferencias principalmente en la finesa con la que se siguen los contornos de los objetos en una escena, la fig.(4.14) muestra dichas diferencias. La imagen de la izquierda se obtuvo optimizando ec.(4.4) y la derecha con ec.(4.5).

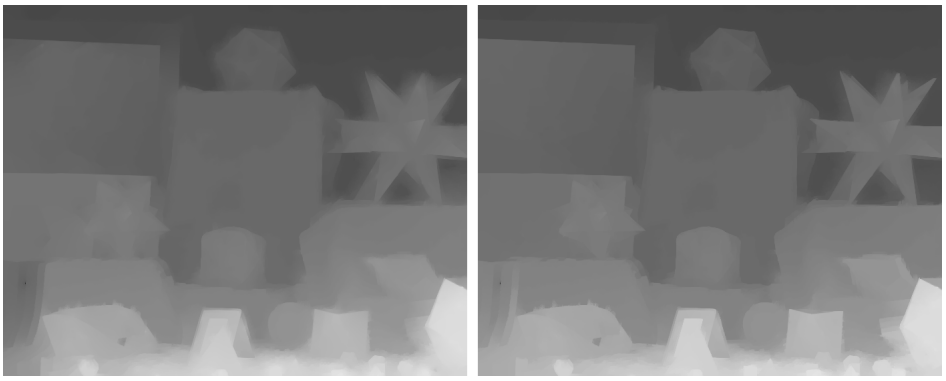


Figura 4.14: diferencias entre ec.(4.4) y ec.(4.5)

Nótese como en la fig.(4.14) la imagen de la izquierda los objetos tienen sus bordes un poco difusos, mientras que en la derecha se aprecian unos bordes mejor; esta es la razón por la cual se prefiere emplear la ec.(4.5) para el proceso de optimización.

Una vez se ha refinado el mapa de disparidad el proceso que sigue es realizar su conversión a mapa de profundidad. Este proceso se realiza tal como se describió en el capítulo de marco teórico. La figura de abajo muestra un escenario reconstruido en 3D.

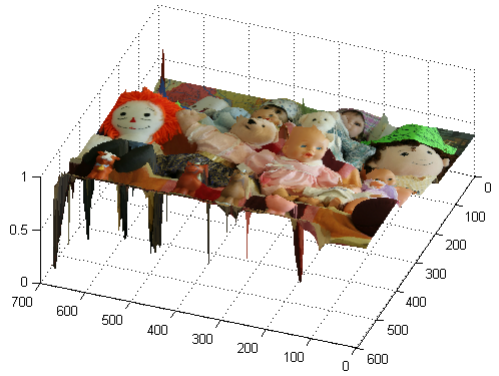


Figura 4.15: Resultado de la reconstrucción 3D

4.7. Resumen

Hasta el momento no se ha dicho de manera explícita cual es el proceso para obtener un mapa de disparidad mediante la metodología propuesta por lo que a manera de resumen la fig.(4.16) resulta muy ilustrativa.

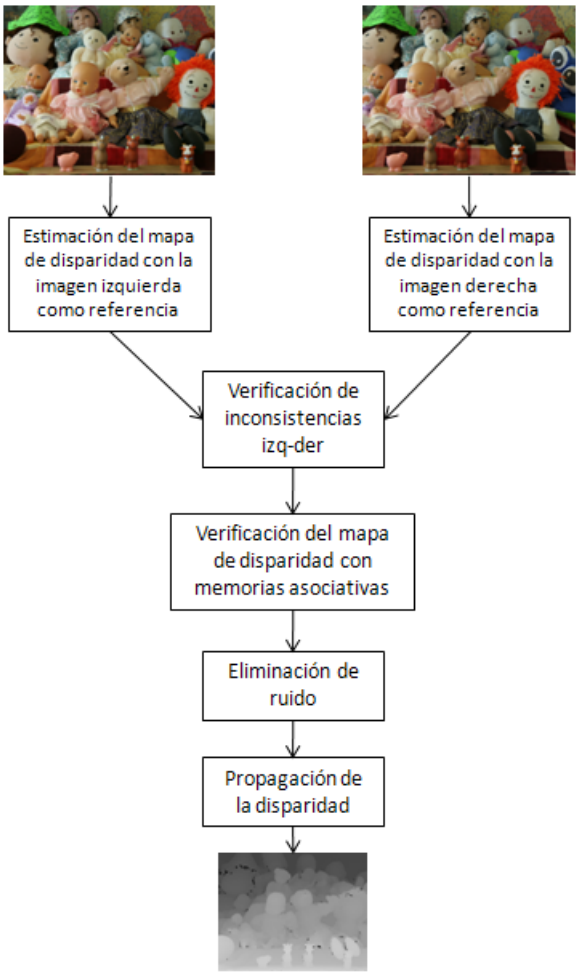


Figura 4.16: Metodología para el cálculo del mapa de disparidad

Cabe señalar que en la fig.(4.16) se ha omitido la reconstrucción 3D por que la parte más importante es la estimación de la disparidad antes de crear un modelo 3D.

4.8. Conclusiones

En este capítulo se ha mostrado cómo se puede utilizar un método completamente numérico para propagar la disparidad calculada, así como el uso de memorias asociativas puede ayudar para verificar el resultado del proceso de correlación. Cabe destacar que hasta lo que se tiene de conocimiento hasta este momento, la metodología presentada aquí hace uso por primera vez del algoritmo de coloreado para propagar mediciones de disparidad.

Lo que resulta de mucha importancia es que las técnicas empleadas para estimación del mapa de profundidad, su verificación y optimización son relativamente simples y se evita por completo el empleo de técnicas muy elaboradas como son la representación del mapa de disparidad como un grafo. También es de resaltar que mediante el proceso de optimización se ha podido establecer una relación entre el mapa de profundidad y las intensidades de la imagen del escenario a reconstruir.

Finalmente hay que destacar que se ha visto experimentalmente que el criterio estadístico propuesto ha sido exitosamente usado para la eliminación de regiones ruidosas y el filtro mínimo aplicado después del estadístico ha ayudado a eliminar gran cantidad de ruido.

Capítulo 5

Resultados

En este capítulo se discuten y muestran los resultados del presente trabajo de investigación. Las pruebas que se muestran se realizan empleando el conocido conjunto de imágenes estereoscópicas de Middlebury, cuyas razones fueron expuestas en el capítulo anterior.

5.1. Acerca de la implementación desarrollada

La implementación de los algoritmos ha sido realizada en C/C++ y Matlab, la principal razón para realizar esto ha sido la facilidad con la que se puede realizar experimentación y la facilidad para leer imágenes de distintos formatos. A continuación se detallan las partes que han sido desarrolladas.

- función para calcular mapa de disparidad, implementada en C/C++ como un plugin para Matlab.
- biblioteca para el uso de memorias asociativas morfológicas, implementada en C/C++.
- filtro para la eliminación de ruido del mapa de disparidad, implementado en Matlab.

- optimización del mapa de disparidad, implementada en Matlab como un derivado del código[64].

La principal razón para emplear Matlab en la optimización del mapa de disparidad es que implementar la solución de mínimos cuadrados en C/C++ resulta complicada, se requiere tiempo para su implementación y validación de su estabilidad numérica.

5.2. Pruebas sobre el conjunto de Middlebury

A continuación se muestran los resultados de la metodología propuesta empleando el conjunto de imágenes de Middlebury, las regiones más claras significan una disparidad mayor y las más obscuras una disparidad menor, las áreas de color negro representan regiones donde la disparidad es desconocida.

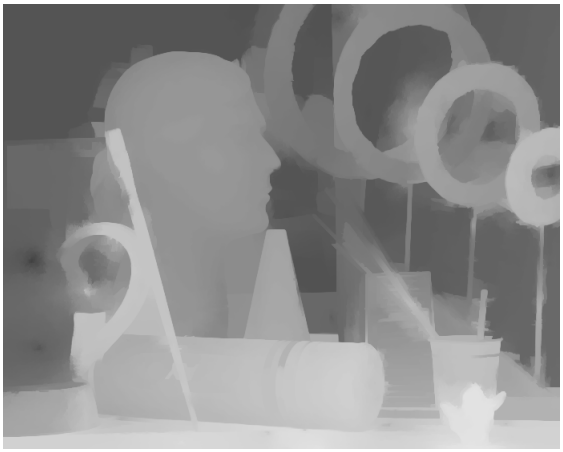


Figura 5.1: Imagen Art

En la figura anterior se puede observar como hacia la parte derecha de la imagen se encuentran área con son consistentes con las que están en sus cercanías, esto se sebe principalmente a que esas áreas no pudieron ser encontradas al realizar el proceso de cálculo de la disparidad, también se pueden observar áreas en las que ciertos objetos han propagado su disparidad hacia otros objetos.

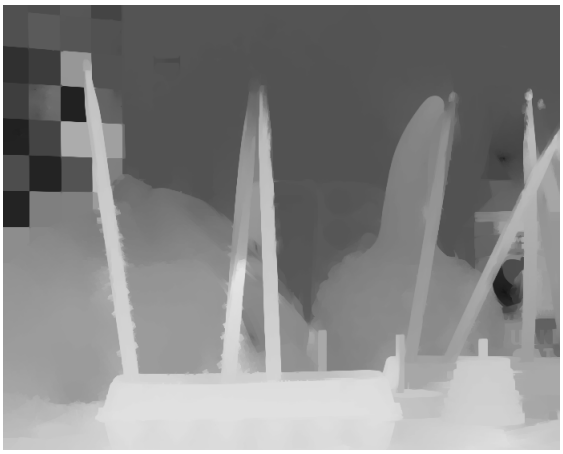


Figura 5.2: Imagen Drumsticks

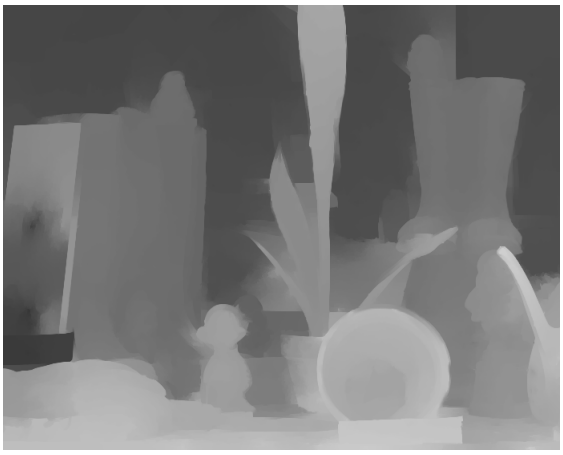


Figura 5.3: Imagen Dwarves

En las fig.(5.2) y fig.(5.3) se puede apreciar detalles finos como las baquetas se han podido mantener bastante bien y también detalles como las hojas de la planta no han sido confundidos con el fondo, los errores se puede notar hacia la parte izquierda de los mapas de disparidad.

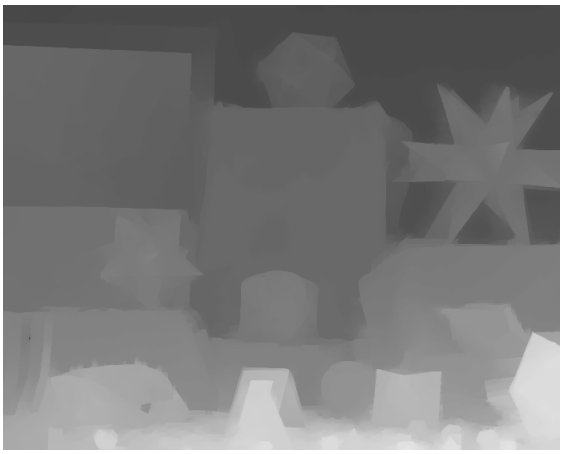


Figura 5.4: Imagen Moebius

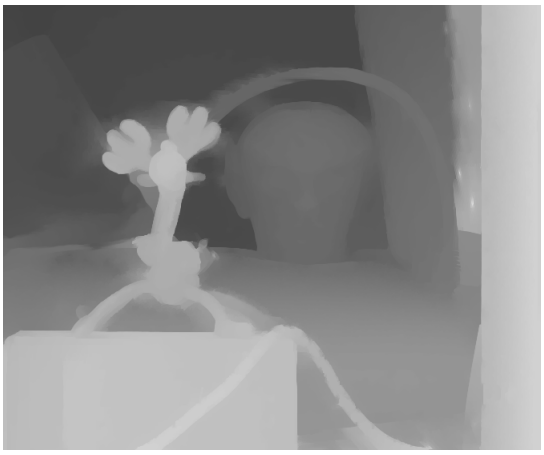


Figura 5.5: Imagen Reindeer

En estos dos mapas de disparidad se pueden observar regiones que han sido contaminadas por otros objetos; por ejemplo en la fig.(5.4) abajo de la estrella más grande (parte superior derecha) se puede ver como la estrella ha contaminado el fondo con su disparidad, y en la fig.(5.5) se puede observar en la parte inferior derecha de la cabeza humana ha sido contaminada con la disparidad de la cabeza.

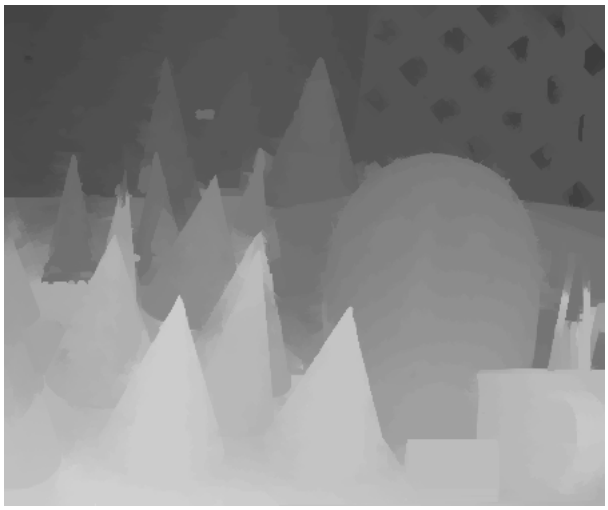


Figura 5.6: Imagen Cones

Figura 5.7: Imagen Laundry

Los mapas de disparidad mostrados arriba conservan detalles tales como las formas de los conos y la máscara. Otro detalle pequeño que se ha conservado bastante bien ha sido el gatillo del aspersor en la imagen "laundry". La conservación de detalles finos ha sido una característica que se ha observado en los experimentos, esto se atribuye principalmente a la función que asigna los pesos durante el

proceso de optimización, dicha función es una modificación (y una aportación de la presente tesis) de la función propuesta en [64].

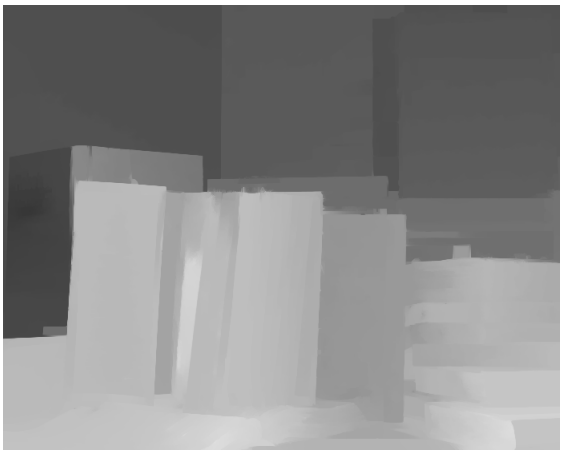


Figura 5.8: Imagen Books

Nótese que a pesar de que se ha eliminado gran cantidad de ruido de los mapas de disparidad estimados aun hay regiones en las cuales hay errores ocasionados por el ruido. Es importante decir que el método de optimización es propenso a producir estas regiones debido a que las propaga, por lo cual es muy deseable desarrollar en el futuro un mecanismo mediante el cual se puedan eliminar aun más la regiones ruidosas.

5.3. Comparación de resultados obtenidos con los mapas de disparidad reales

En esta sección se muestra la comparación de los mapas de profundidad obtenidos en el presente trabajo con los mapas reales que provee el conjunto de imágenes de Middlebury, las imágenes a la izquierda son los resultados de la metodología propuesta y las imágenes de la derecha son los mapas de disparidad reales que proporciona el conjunto de Middlebury.

Para el cálculo de los mapas de disparidad se ha empleado una ventana de correspondencia de $5x5$, un filtro estadístico de $11x11$, y un filtro mínimo de $3x3$.

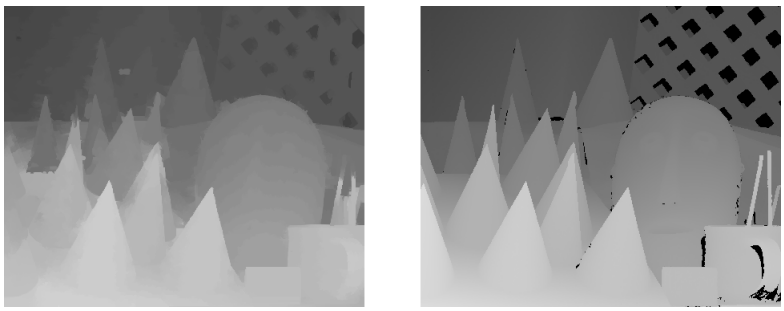


Figura 5.9: Imagen Cones

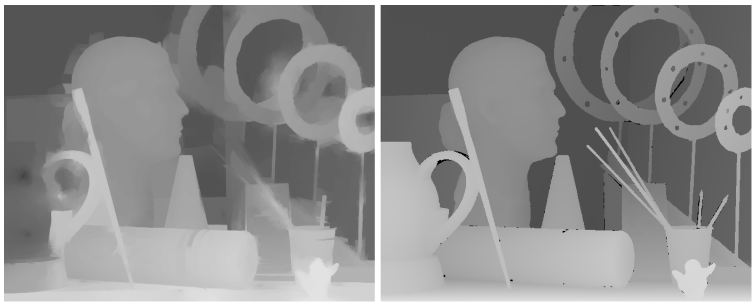


Figura 5.10: Imagen Art

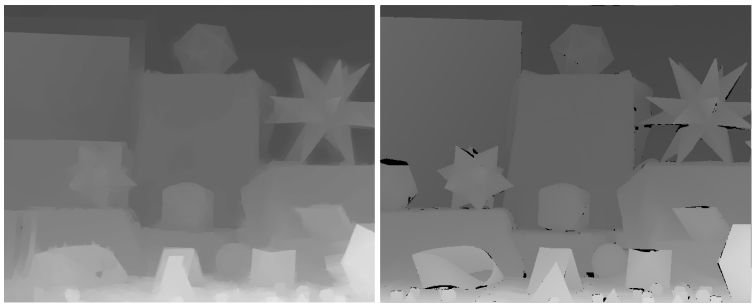


Figura 5.11: Imagen Moebius

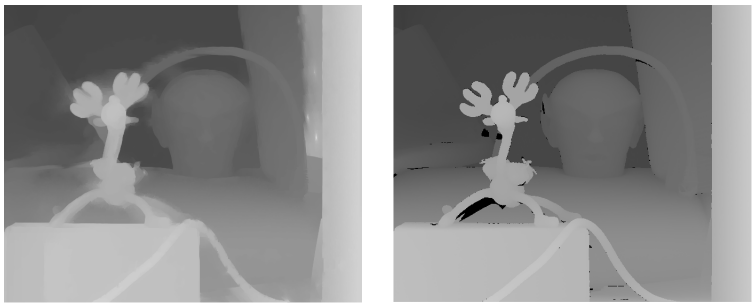


Figura 5.12: Imagen Reindeer

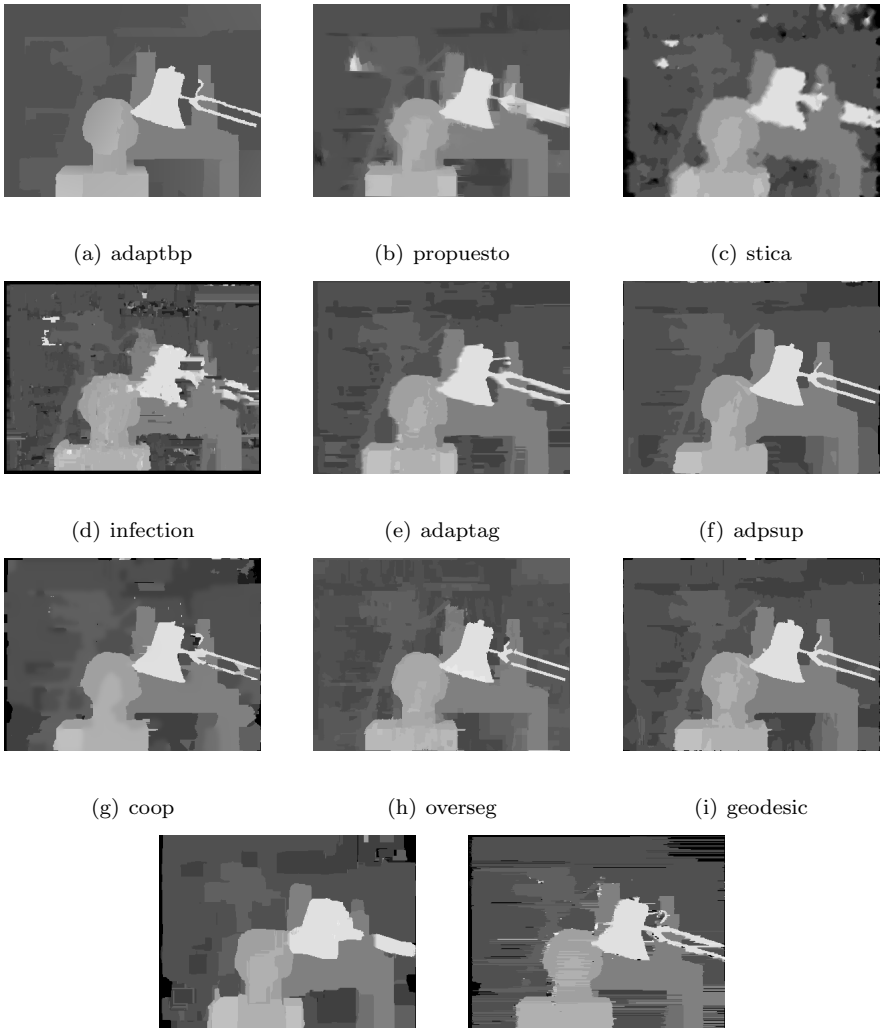
Nótese que los resultados obtenidos por la metodología propuesta son bastante cercanos a los resultados de los mapas disparidad de Middlebury. Cabe destacar que las ventanas empleadas son pequeñas y aún así los resultados que se obtienen son bastante cercanos, lo que parece ser que ayuda a éste resultado es el proceso de propagación de la disparidad.

5.4. Comparación numérica

Aunque en la comparación visual mostrada en la sección anterior se aprecian buenos resultados es necesario realizar una evaluación numérica de los resultados obtenidos, de igual forma es más que obligatorio realizar una comparación con los resultados obtenidos por otros autores.

En esta sección se muestran tanto las imágenes obtenidas por la metodología propuesta como por otros autores, la evaluación de resultados ha sido realizada mediante la aplicación web en [66], la principal razón para emplear dicha aplicación es porque la evaluación de resultados se realiza por una tercera parte de manera objetiva, las imágenes empleadas en la comparación son "tsukuba", "teddy", "venus" y "cones". Las comparaciones se han hecho tomando en cuenta los algoritmos "adaptbp"[32] (el mejor al momento de haber hecho esta tesis), "infection" [33] y"stica" [34], estos dos últimos algoritmos son realizados por investigadores del país, los algoritmos "adaptag"[29], "adpsup"[30], "coop"[28], "overseg"[27],

"geodesic"[26], "SSD+MF" y "SO" [31] han sido seleccionados por que se tratan de algoritmos locales para la estimación de mapas de disparidad.



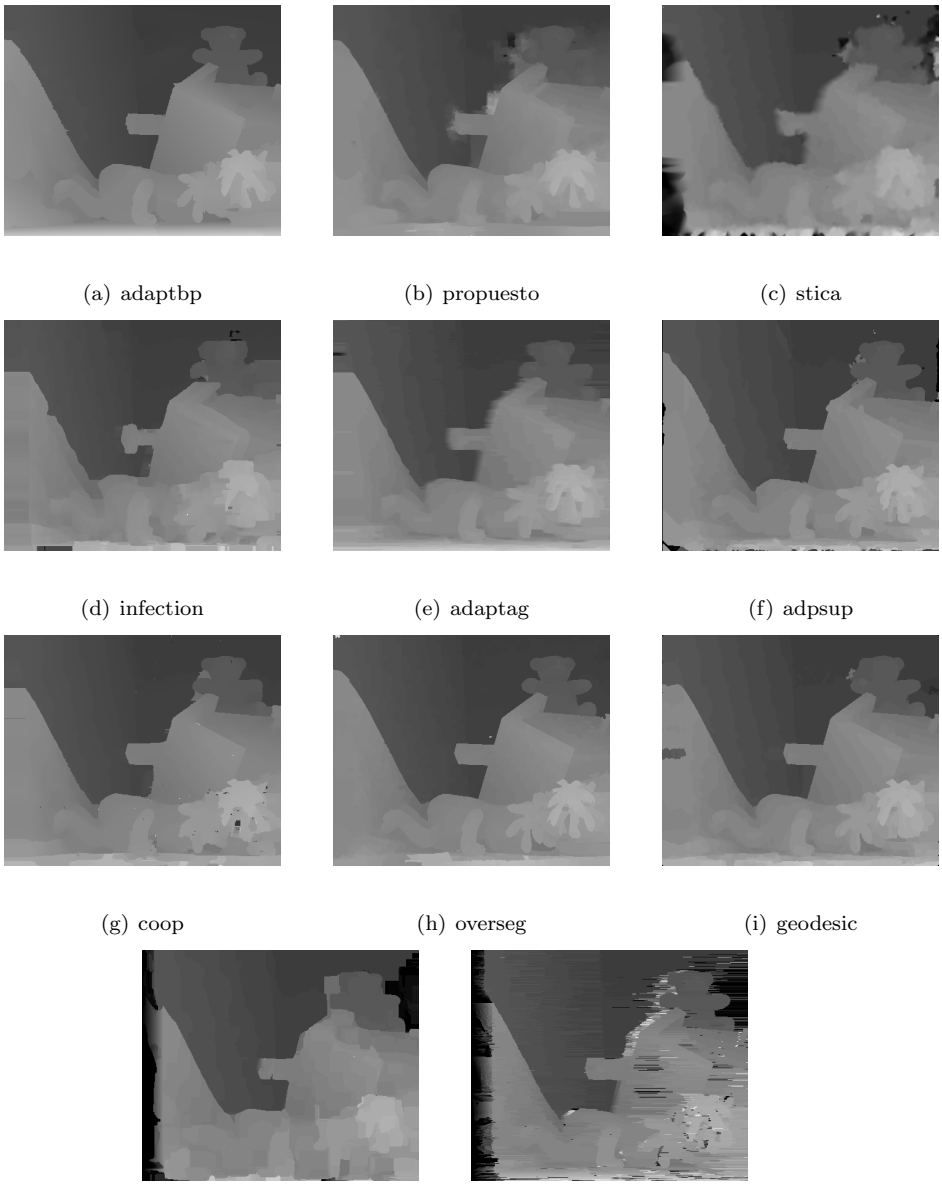
(a) adaptbp (b) propuesto (c) stica

(d) infection (e) adaptag (f) adpsup

(g) coop (h) overseg (i) geodesic

(j) SSD+MF (k) SO

Figura 5.13: Comparación imagen "tsukuba"



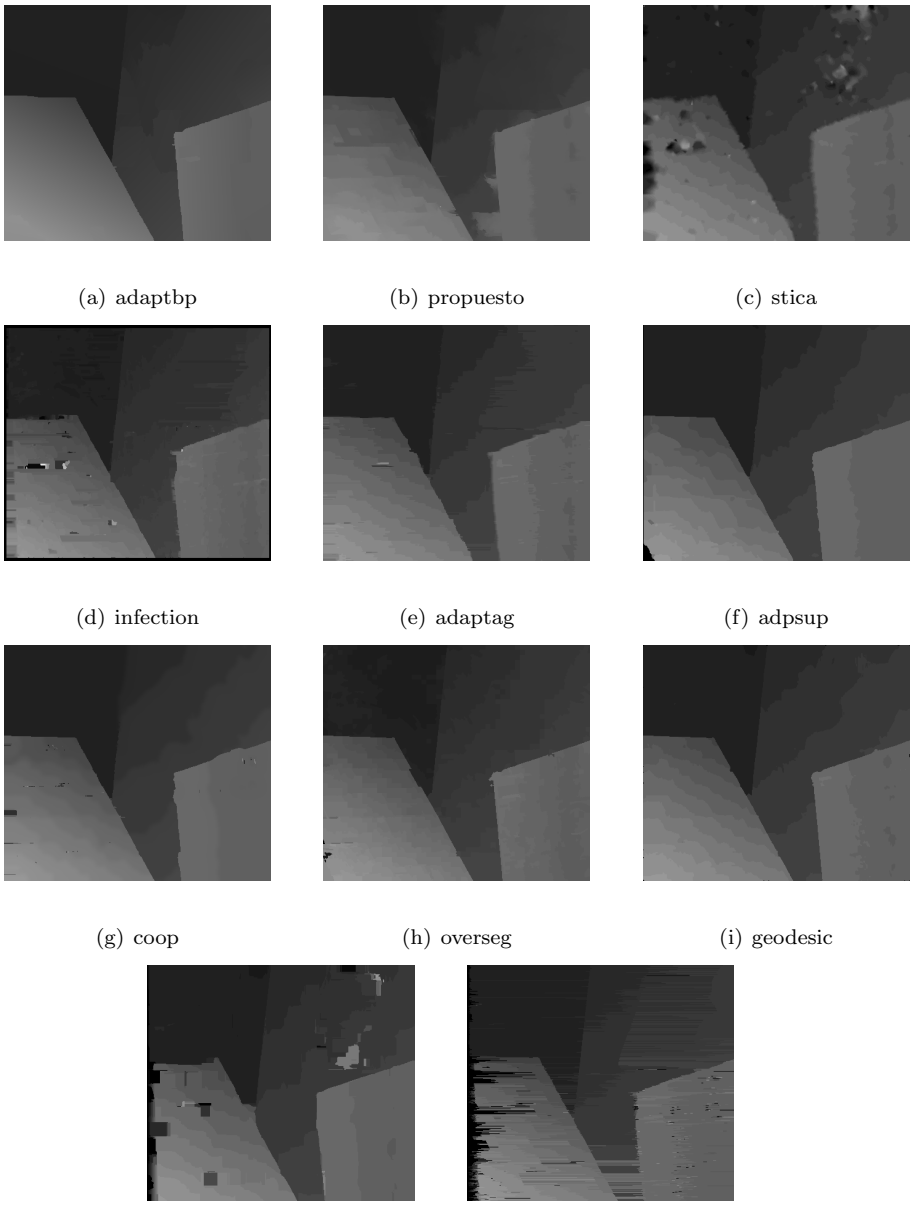
(a) adaptbp (b) propuesto (c) stica

(d) infection (e) adaptag (f) adpsup

(g) coop (h) overseg (i) geodesic

(j) SSD+MF (k) SO

Figura 5.14: Comparación imagen “teddy”



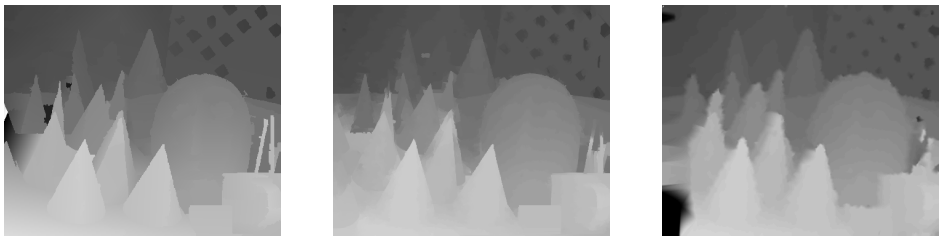
(a) adaptbp (b) propuesto (c) stica

(d) infection (e) adaptag (f) adpsup

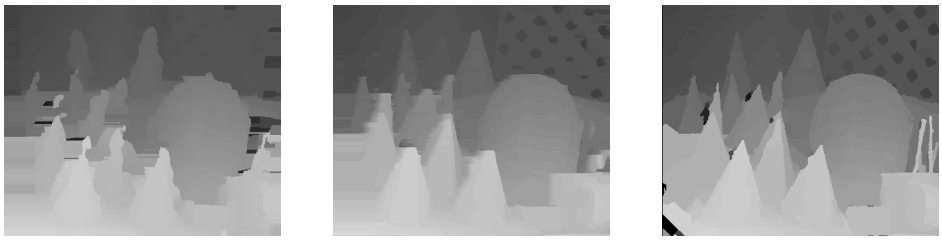
(g) coop (h) overseg (i) geodesic

(j) SSD+MF (k) SO

Figura 5.15: Comparación imagen “venus”

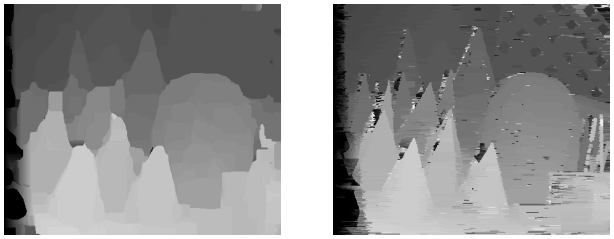


(a) adaptbp (b) propuesto (c) stica



(d) infection (e) adaptag (f) adpsup

(g) coop (h) overseg (i) geodesic



(j) SSD+MF (k) SO

Figura 5.16: Comparación imagen “cones”

La tab.(5.1) muestra el porcentaje de error de cada algoritmo en las imágenes de prueba, cabe destacar que el método "adaptbp" es el mejor que tiene mejor desempeño (al menos al momento en que se ha realizado la presente tesis).

Algoritmo	Tsukuba	Venus	Teddy	Cones	Lugar
adaptbp	1.11	0.10	4.22	2.48	1
geodesic	1.45	0.14	6.88	2.94	16
overseg	1.69	0.51	6.74	3.19	27
adpsup	1.25	0.25	8.43	3.77	30
coop	2.91	0.60	7.92	3.59	47
adaptag	2.05	1.92	7.23	6.41	71
propuesto	7.56	3.10	17.3	7.42	90
SSD+MF	5.23	3.74	16.5	10.06	94
SO	5.08	9.44	16.9	13.0	95
stica	7.70	8.19	15.8	9.80	98
infection	7.95	4.41	17.7	14.3	101

Tabla 5.1: Tabla comparativa de la metodología propuesta

Es muy importante resaltar que los resultados que se han obtenido emplean ventanas de tan solo $5x5$ cuando comúnmente se usan ventanas de $11x11$,$15x15$ o mayores. Otro aspecto que se debe tomar en cuenta es que mediante el proceso de eliminación de ruido en el mapa de disparidad se pierde gran cantidad de información y aun así la presente metodología, mediante el uso del proceso de optimización es capaz de compensar la falta de información.

5.5. Reconstrucción 3D

Cómo se puede apreciar hasta este punto se ha dedicado una gran cantidad de tiempo a presentar resultados del cálculo del mapa de disparidad ya que ,como se mostró en el capítulo de marco teórico, la profundidad de los objetos en un escenario dependen de la disparidad. Por este motivo se puede asegurar que el problema de reconstrucción 3D se resuelve una vez se ha obtenido un mapa de disparidad que tenga una calidad aceptable y este se transforma en mapa de disparidad.

Las fig.(5.17), (5.18), (5.19), (5.20),(5.21) y (5.22) muestran las reconstrucciones 3D de algunas de las imágenes del conjunto de Middlebury, las imágenes simplemente se han usado como textura sobre el mapa de profundidad con el fin de poder apreciar los objetos con sus colores y texturas correspondientes, el proceso mediante el cual se transforma el mapa de profundidad en un modelo 3D es mediante la triangulación de Delaunay (cómo se mostro en marco teórico); los modelos que se muestran a continuación han sido obtenidos con Matlab.

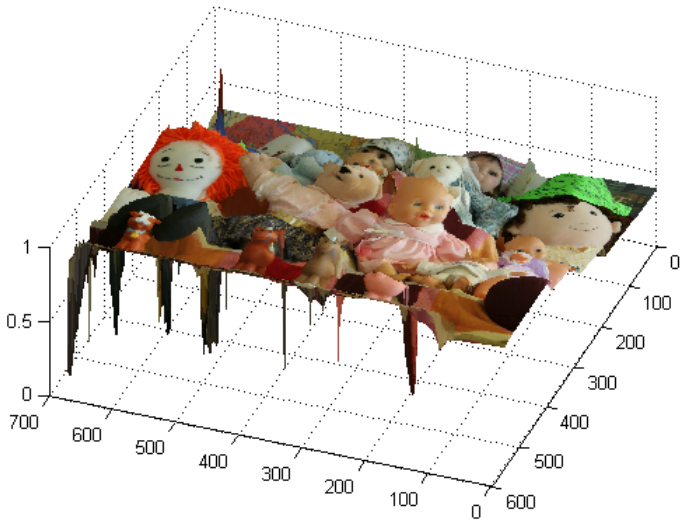


Figura 5.17: Resultado de la reconstrucción 3D imagen Dolls

En la fig.(5.17) se pueden observar detalles finos como la nariz del oso y de los caballos, también nótese como la pared que se encuentra al fondo se ve bastante plana. Esta es una de las imágenes donde se obtienen mejores resultados ya que hay una cantidad muy limitada de ruido que afecta al mapa de profundidad, esta imagen también es muy buena para apreciar como los diferentes peluches que se encuentran ordenados por filas, motivo por el cual la figura ha sido rotada y es posible apreciar cómo se obstruyen los diferentes objetos en el escenario.

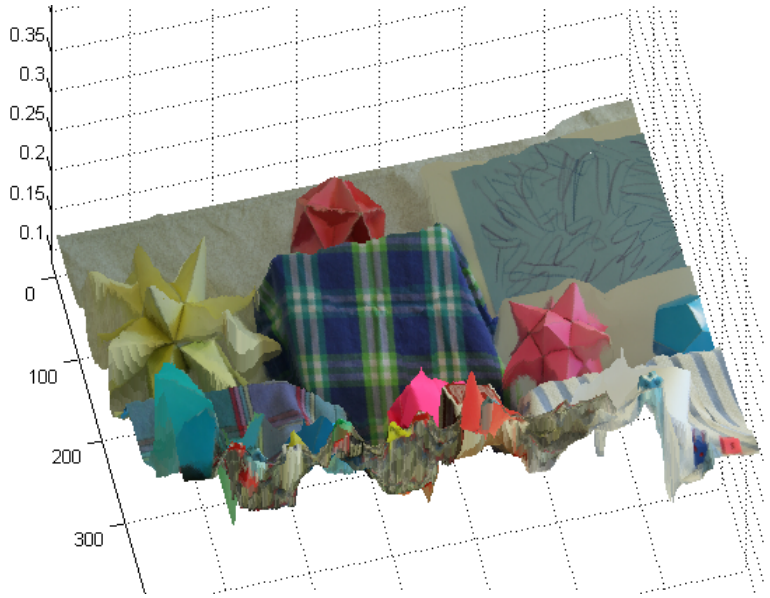


Figura 5.18: Resultado de la reconstrucción 3D imagen Moebius

La figura anterior en particular es buena para poder observar detalles un poco complicados como son los picos de las estrellas así como las arrugas en las telas. También es importante hacer notar errores como en el cuadro con garabatos donde se aprecian regiones que no son completamente planas.

Figura 5.19: Resultado de la reconstrucción 3D imagen Reindeer

La fig.(5.19) muestra como se ha conservado bastante bien la forma de la cuerda que está enfrente de la caja de cartón, también se puede observar que se han conservado detalles como los de la nariz de la cabeza humana.

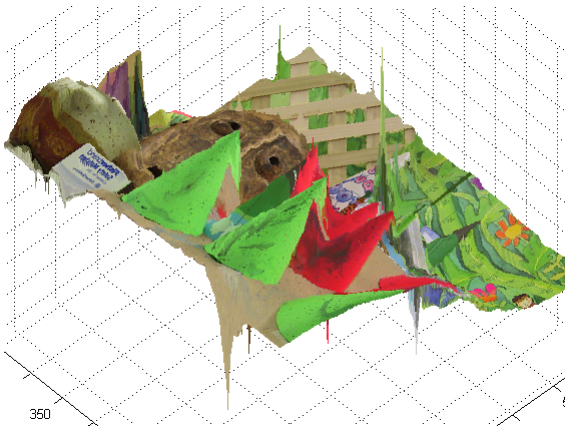


Figura 5.20: Resultado de la reconstrucción 3D imagen Cones

La figura de "Cones" se ha rotado especialmente para que se pueda apreciar como los diferentes conos se obstruyen entre sí así como también a la máscara que se aprecia en el fondo, de esta mascara se

pueden apreciar detalles finos como los ojos que han conservado su relieve circular.

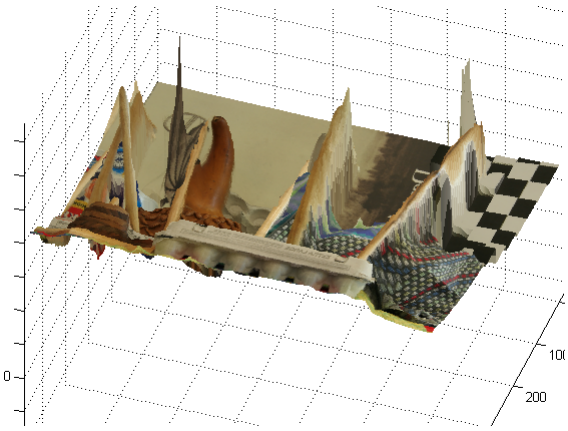


Figura 5.21: Resultado de la reconstrucción 3D Drumstick

La fig.(5.21) ha sido particularmente difícil de reconstruir especialmente el fondo donde se aprecian errores, sin embargo detalles como los huecos en la caja de huevos se han conservado al igual como detalles un poco más complicados como los de la tela que está en la imagen.

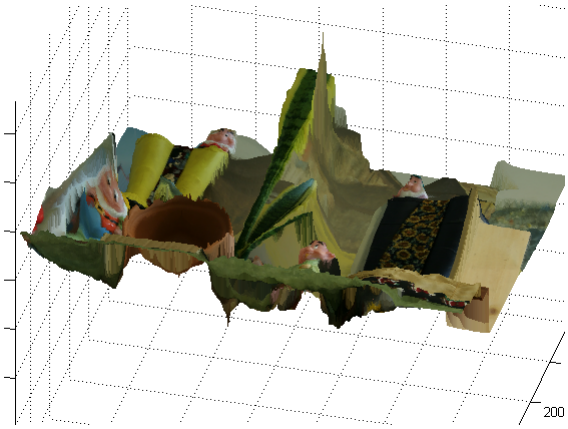


Figura 5.22: Resultado de la reconstrucción 3D imagen Dwarves

Esta última figura ha sido rotada de tal forma que se aprecie el

fondo de la maceta, y al mismo tiempo que la maceta obstruya a las botas amarillas.

Cómo se puede observar en los modelos 3D las regiones ruidosas se manifiestan como picos en lugares donde no deberían estar, motivo por el cual una mejor eliminación de dichas inconsistencias se planea dejar como trabajo futuro.

5.6. Conclusiones

La tarea de reconstrucción 3D ha resultado ser compleja, llena de obstáculos y demuestra porque aún hoy en día sigue siendo un tema de investigación actual. En el presente trabajo de tesis se ha realizado un esfuerzo por aplicar técnicas novedosas y al mismo tiempo simples para la reconstrucción 3D. Es claro que los resultados presentados todavía presentan regiones donde existe ruido aunque se debe resaltar que los resultados obtenidos son muy cercanos a los mapas de disparidad reales.

Capítulo 6

Conclusiones y Trabajo futuro

6.1. Conclusiones

A lo largo de este trabajo se han encontrado muchos obstáculos para poder calcular un mapa de disparidad que en la medida de lo posible, esté libre de ruido. Esto se debe a la naturaleza del problema de la reconstrucción 3D en donde se hace frente a información incompleta, ruidosa e inexacta.

Una de las principales lecciones aprendidas es que es posible utilizar técnicas relativamente simples para obtener mapas de disparidad lo que pensamos podría ayudar a la implementación futura de un sistema de reconstrucción 3D en tiempo real.

Entre los puntos a favor que se han observado en la metodología propuesta se encuentra el hecho de que es posible obtener resultados bastante aceptables mediante el uso de ventanas de correspondencia pequeñas como aquellas descritas en los resultados, esta

característica es sumamente importante por que ayuda a demostrar que con poca información es posible calcular el mapa de disparidad. También se ha mostrado como un criterio estadístico sencillo ayuda en gran medida a eliminar regiones ruidosas respetando a aquellas que son correctas.

Cabe destacar que se ha observado que el algoritmo de coloreado logra conservar bastante bien la forma de los objetos que se encuentran en un escenario. Otro resultado importante es el uso del algoritmo de coloreado para la propagación de medidas de disparidad, esto resulta muy interesante en especial porque con un mapa de disparidad incompleto se puede obtener uno con calidad aceptable, también hay que resaltar que (al momento de realizar la presente tesis) es la primera vez que se emplea el algoritmo de coloreado para propagar medidas de disparidad y además se obtienen resultados bastante aceptables.

Una observación muy importante sobre el algoritmo de coloreado es que este puede ser considerado como un método de optimización global por lo que realizar mejoras en la forma en que se asignan pesos mejora los resultados tal como se mostro en el capítulo de metodología.

En cuanto a las desventajas de la metodología propuesta se ha notado que cuando un existen pequeñas regiones ruidosas que no se eliminaron completamente estas tienen a propagarse ligeramente, motivo por cual se cambio la función para asignar pesos. Otra dificultad que se notó es el hecho de que en ocasiones las regiones que tienen mayor información antes de realizar el proceso de optimización contaminan ligeramente a aquellas donde había poca información.

Finalmente, nos parece correcto decir que mediante el proceso de optimización que se ha empleado, ha sido posible establecer una relación entre la imagen del escenario y la profundidad de éste lo cual fue uno de los objetivos principales de esta tesis.

6.2. Trabajo futuro

En este capítulo se discute brevemente las posibles mejoras al trabajo de investigación expuesto en la presente tesis.

Cómo se menciono en el capítulo de resultados, la eliminación total del ruido no se consigue completamente lo que aparezcan ciertas regiones que son incorrectas en el mapa de disparidad que se obtiene, también resulta de gran interés mejorar la precisión de la corrección que realiza la memoria asociativa ya que esto ayudaría a eliminar más ruido.

Otro aspecto muy importante que queda por probar es el uso de la metodología propuesta en imágenes al aire libre y evaluar que tanto ayuda el proceso de eliminación de ruido que se ha propuesto.

También queda por experimentar si es posible dividir la imagen en regiones pequeñas a optimizar esto con el fin tanto de reducir tiempos de ejecución como memoria requerida por el proceso de optimización. De igual forma queda por probar la optimización del mapa de disparidad empleando mínimos cuadrados no lineales.

Dado que el proceso para calcular la disparidad implica hacer una comparación pixel a pixel desde la imagen de referencia con la imagen donde se realiza la comparación, resultaría muy interesante no usar todos los pixeles sino únicamente un conjunto reducido, esto tendría como resultado un mapa de disparidad que está incompleto pero con la ayuda del algoritmo de coloreado quizás sea posible obtener un mapa de disparidad con una calidad aceptable y al mismo reducir el área donde se realiza la comparación y por consiguiente el tiempo que tarda el algoritmo en calcular un mapa de disparidad.

Entre las mejoras posibles a la metodología propuesta queda por incluir en el proceso de agregación un mecanismo que tome encuentra una de las imágenes a color (ya sea la izquierda o derecha), esto con el fin de usar pesos en la ventana de agregación que se adapten dependiendo de que tanto se parecen los pixeles dentro de la ventana de agregación tomando como referencia una de las imágenes estereoscópicas.

Cabe destacar el tamaño del filtro estadístico afecta de manera muy notoria el resultado del proceso de optimización por lo que sería deseable poder desarrollar un mecanismo mediante el cual el tamaño de dicho filtro se pueda adaptar se manera automática y por consiguiente podría ayudar a mejorar la eliminación de ruido.

Otra mejora importante es desarrollar un mecanismo mediante el cual sea posible hacer que los mapas de disparidad varíen de una forma más suave lo que posiblemente ayudaría a obtener resultados más competitivos.

Finalmente y de mayor interés queda por demostrar si es posible obtener la disparidad únicamente empleando memorias asociativas, de ser posible se podría crear un método completamente nuevo para la estimación de la disparidad.

Referencias

[1] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. International Journal of Computer Vision, 47(1/2/3):7-42, Abril-Junio 2002.

[2] D. Scharstein and R. Szeliski. High-accuracy stereo depth maps using structured light. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2003), vol. 1, pp.195-202, Madison, WI, Junio 2003.

[3] D. Scharstein and C. Pal. Learning conditional random fields for stereo. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), Minneapolis, MN, Junio 2007.

[4] H. Hirschmüller and D. Scharstein. Evaluation of cost functions for stereo matching. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), Minneapolis, MN, Junio 2007

[5] R. C. Gonzalez, R. E. Woods, Digital Image Processing, Prentice Hall, ISBN: 0–201–18075–8.

[6] Cornelio Yáñez Márquez, Juan Luis Díaz de León Santiago, Normas y Métricas de Minkowski, Serie: VERDE, No. 89,2003.

[7] H. P. Hsu, Análisis de Fourier, Prentice Hall, ISBN: 968–444–356–0.

[8] R. Klette,K. Schlns,A. Koschan, Computer Vision Three- Dimensional Data from Images, Springer, 1998 , ISBN: 981–3083–71–9, pp 2–11.

[9] R. Hartley, A. Zisserman, Multiple View Geometry in Computer Vision, Cambridge University Press, 2006, ISBN: 0–521–54051–8.

[10] O. Faugueras,Three Dimensional Computer Vision a Geometry View Point, Pearso Education, 2006, ISBN: 0–521–54051–8.

[11] B. Cyganek, J. Paul Siebert, An introduction to 3D Computer Vision Techniques and Algorithms, Wiley,2009, ISBN: 978–0–470–01704–3.

[12] C. Harris ,M. Stephens,A Combined Corner and Edge Detector,Plessey Research Roke Manor, United Kingdom, 1988.

[13] E. Prados, O. Faugeras, Shape From Shading, En: Mathematical Models in Computer Vision: The Handbook, Springer, 2005, ISBN: 978–0387–28831–4.

[14] H. Farid,Range Estimation by Optical Diferentiation,Ph.D Thesis, University of Pennsylvania, 1997.

[15] D. Murray, J. Little, Using real-time stereo vision for mobile robot navigation, Vol. 8 , No. 2, pp. 161–171 ,2000, ISSN: 0929—5593.

[16] Y. Shirai,Three-dimensional Computer Vision, Springer, 1987.

[17] N. Grandón-Pastén,Diego Aracena–Pizarro,Clésio Luis Tozzi, Reconstrucción de Objeto 3D a Partir de Imagenes Calibradas, Ingeniare. Revista chilena de ingeniera, vol. 15, No 2, 2007, pp. 158—168.

[18] M. Asif, Tae-Sun Choi, Shape From Focus Using Feedfordward Neural Networks, IEICE trans. inf & syst., Vol.E83-D ,No. 4 Abril 2000

[19] D. Bradley, T. Boubekeur, W. Heidrich, Accurate Multi-View Reconstruction Using Robust Binocular Stereo and Surface Meshing, IEEE Conference on Computer Vision and Pattern Recognition, Junio 2008, ISSN: 1063–6919

[20] A. Saxena, S. H. Chung, Andrew Y. Ng, 3–D Depth Reconstruction from a Single Still Image, International Journal of Computer Vision, Vol. 78 , No. 2–3 , Julio 2008 , pp. 143-–167.

[21] P. Li, D. Farin, P. H. N. de With , Rene Klein Gunnewiek, On Creating Depth Maps from Monoscopic Video using Structure From Motion,Video Coding and Architecture group, Eindhoven University of Technology, Digital Signal Processing group, Philips Research Eindhoven.

[22] M. Pollefeys, D. Nist , J.M. Frahm, A. Akbarzadeh, P. Mordohai , B. Clipp, C.Engels , D. Gallup , S.-J. Kim, P. Merrell, C. Salmi, S. Sinha, B. Talton, L.Wang, Q. Yang, H. Stewnius, R. Yang, G. Welch, H. Towles, Detailed Real–Time Urban 3D Reconstruction From Video, International Journal of Computer Vision, Vol. 78, No. 2–3, Julio 2008.

[23] A. Levin, R. Fergus, F. Durand,W. T. Freeman, Image and depth from a conventional camera with a coded aperture,International Conference on Computer Graphics and Interactive Techniques ACM SIGGRAPH 2007, 2007, ISSN:0730–0301.

[24] O. Ikeda, Shape-from-Shading for Oblique Lighting with Accuracy Enhancement by Light Direction Optimization, Hindawi Publishing Corporation EURASIP Journal on Applied Signal Processing, Vol. 2006, 2006, articulo 92456.

[25] F. Han, S.-C. Zhu, Cloth Representation by Shape from Shading with Shading Primitives, Department of Statistics, UCLA Department of Statistics Papers (University of California, Los Angeles), 2005, articulo 2005040101.

[26] A. Hosni, M. Bleyer, M. Gelautz, and C. Rhemann. Local stereo matching using geodesic support weights. ICIP 2009.

[27] L. Zitnick and S.B. Kang. Stereo for image-based rendering using image over-segmentation. IJCV 2007.

[28] R. Brockers. Cooperative stereo matching with color-based adaptive local support. CAIP 2009.

[29] L. Wang, M. Liao, M. Gong, R. Yang, and D. Nistér. High-quality real-time stereo using adaptive cost aggregation and dynamic programming. 3DPVT 2006.

[30] F. Tombari, S. Mattoccia, and L. Di Stefano. Segmentation-based adaptive support for accurate stereo correspondence. PSIVT 2007.

[31] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. IJCV 2002.

[32] A. Klaus, M. Sormann and K. Karner. Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. ICPR 2006.

[33] G. Olague, F. Fernández, C. Pérez, and E. Lutton. The infection algorithm: an artificial epidemic approach for dense stereo correspondence. Artificial Life, 2006.

[34] H. Audirac, A. Beloiarov, F. Núñez, and J. Villegas. Dense disparity map based on STICA algorithm. Expo–Forestal, México, 2005.

[35] W. Wang, Y. Wang, L. Huo, Q. Huang, W. Gao , Symmetric segment-based stereo matching of motion blurred images with illumination variations,Key Lab. of Intell. Inf. Process., Chinese Acad. of Sci., Beijing.

[36] C. Georgoulas, L. Kotoulas, G. Ch. Sirakoulis,I. Andreadis, A. Gasteratos, Real–time disparity map computation module,Microprocessors and Microsystems 32 (2008) 159—170.

[37] M. Niederöst, J. Niederöst, J. Scucka, Shape From Focus: Fully Automated 3D Reconstruction and Visualization of Microscopic Objects, Proceedings of 6th International Conference on Optical 3-D Measurement Techniques,2003, Zurich, Switzerland.

[38] M. Asif, T. Choi, Shape from Focus Using Multilayer Feedforward Neural Networks,IEEE Transactions on Image Processing, Vol. 10, No. 11, Nov 2001.

[39] A. Levin, Blind Motion Deblurring Using Image Statistics,School of Computer Science and Engineering The Hebrew University of Jerusalem.

[40] S. Kyu Kang, J. Hong Min, J. Ki Paik, Segmentation-based spatially adaptive motion blur removal and its application to surveillance systems, International Conference on Image Processing, Vol.1, 2001, pp. 245–248.

[41] L. Guibas, J. Stolfi, Primitives for the Manipulation of General Subdivisions and the Computation of Voronoi Diagrams, ACT TOG, 4(2), Abril, 1985.

[42] S. Fortune, A Sweepline Algorithm for Voronoi Diagrams, Algorithmica, 2:153–174, 1987.

[43] M. Bern, D. Eppstein. Mesh Generation and Optimal Triangulation. Computing in Euclidean Geometry (Ding-Zhu Du and Frank Hwang, editors), Lecture Notes Series on Computing, vol. 1, pp. 23–90. World Scientific, Singapore, 1992.

[44] P. Su, L. Robert , S. Drysdale. A Comparison of Sequential Delaunay Triangulation Algorithms. Proceedings of the Eleventh Annual Symposium on Computational Geometry, pp 61–70. Association for Computing Machinery, Junio 1995.

[45] F. Sánchez Garfias, J. L. Díaz de León Santiago, C. Yáñez Márquez. Reconocimiento automático de patrones: conceptos básicos, Serie Verde, ISBN:970–36–0044–1, CIC–IPN, México, 2003.

[46] F. Sánchez Garfias, J.L. Díaz de León Santiago, C. Yáñez Márquez. Reconocimiento de patrones: enfoque asociativo, Serie Verde, ISBN:970–36–0046–8, CIC–IPN, México, 2003.

[47] F. Sánchez Garfias, J.L Díaz de León Santiago, C. Yáñez Márquez. Reconocimiento de patrones: enfoque neuronal, Serie Verde, ISBN:970–36–0047–6, CIC–IPN, México, 2003.

[48] F.Sánchez Garfias, J.L Díaz de León Santiago, C. Yáñez Márquez. Reconocimiento de patrones: enfoque probabilístico estadístico, Serie Verde, ISBN:970–36–0048–4, CIC–IPN, México, 2003.

[49] P. Henry Winston, Artificial Intelligence,Addison-Wesley,1993,ISBN:0–201–53377–4.

[50] J. Anderson, Redes Neurales, Alfaomega, 2007, ISBN:978–970–15–1265–4.

[51] R. Rojas, Neural Networks: A Systematic Introduction, Springer-Verlag, 1995, ISBN: 3–540–60505–3.

[52] M. Friedman, A. Kandel, Introduction to Pattern Recognition: Statistical, Structural, Nerual and Fuzzy logic aproaches, World Scientific, 2000, ISBN:9810233124.

[53] J.P. Marques de Sá, Pattern Recognition: Concepts, Methods and Applications, Springer, 2001, ISBN:3–540–42297–8.

[54] C. T. Leondes, Image Processing and Pattern Recognition, Academic Press, 1998, ISBN:0-12-443865–2.

[55] B. Javidi, Image Recognition and Classification: Algortihms, Systems and Applications, 2002, ISBN:0–8247–0783–4.

[56] http://archive.ics.uci.edu/ml/

[57] W. McCulloch y W. Pitts, A logical calculus of the ideas immanent in nervous activity, 1943,Bulletin of Mathematical Biophysics, vol.5 pp.115-133.

[58] J.L Díaz de León Santiago, C. Yáñez Márquez,Lernmatrix de Steinbuch, Serie Verde, ISBN:970–18–6688–6, CIC-IPN, México,2001.

[59] J.L Díaz de León Santiago, C. Yáñez Márquez, Linear Associator de Anderson-Kohonen, Serie Verde, ISBN:970-18-6690-8, CIC-IPN, México, 2001.

[60] J.L Díaz de León Santiago, C. Yáñez Márquez, Memoria Asociativa Hopfield, Serie Verde, ISBN:970-18-6692-4, CIC-IPN, México, 2001.

[61] J.L Díaz de León Santiago, C. Yáñez Márquez, Memorias Morfológicas Autoasociativas, Serie Verde, ISBN:970-18-6698-3, CIC–IPN, México, 2001.

[62] R. Barrón Fernández, Memorias Asociativas y Redes Neuronales Morfológicas para la recuperación de patrones, 2006,CIC–IPN, tesis de doctorado.

[63] C. Yáñez Márquez, Memorias Asociativas Basadas en Relaciones de Orden y Operaciones Binarias, 2003, CIC–IPN, ISSN:1405-5546, Computación y Sistemas Vol. 6 No.4 pp. 300 - 311.

[64] A. Levin , D. Lischinski , Y. Weiss, Colorization using optimization, ACM SIGGRAPH 2004, August 08-12, 2004, Los Angeles, California.

[65] http://vision.middlebury.edu/stereo/data/

[66] http://vision.middlebury.edu/stereo/eval/