Análisis de la Función de Correlación por Fase para la Estimación de Disparidad

Claudia G. Ambriz González

Maestría en Ingeniería Electrónica
Facultad de Ciencias, Universidad Autónoma de San Luis Potosí

Resumen

La estimación de disparidad es uno de los problemas más importantes por resolver en el área de visión estéreo; para lograr la reconstrucción tridimensional de una escena. Los métodos más recientes se basan en minimizar una función de energía global. Este tipo de métodos ha demostrado buenos resultados en precisión estimando mapas densos; su alto costo computacional los hace inadecuados para aplicaciones en tiempo real. Considerando esta problemática, en este trabajo de tesis se un algoritmo estéreo para la estimación de mapas disparidad densos a partir de dos vistas, basado en la correlación por fase. El método analizado se basa en encontrar un grupo de posibles valores de disparidad (candidatos) a partir de la función de correlación por fase, posteriormente se determina el candidato correcto para cada pixel minimizando una función de error utilizando una técnica de costo agregado. Todo lo anterior con el objetivo de reducir el espacio de búsqueda y el costo computacional. Para probar el algoritmo se utilizó la base de datos Middlebury, y como medida de precisión se calculó el porcentaje de pixeles correspondientes incorrectos. El algoritmo reduce el espacio de búsqueda en promedio un 37 %, con lo que se alcanza a procesar 1.4 fps. Debe mencionarse que esto se consigue con una implementación secuencial del código, sin recurrir a técnicas de procesamiento en paralelo, uso de GPU's o FPGA's.

Dedicatoria

Para mi ...

Nuestra reconpensa se encuentra en el esfuerzo y no en el resultado.
Un esfuerzo total es una victoria completa.

Mohandas K. Gandhi

Agradecimientos

A mis padres y hermanas por apoyarme a lo largo de toda esta trayectoria. Hoy cierro otro ciclo de la mano de Dios con la bendición de tener una familia que me apoya en las buenas decisiones y me jala las orejas en las malas.

A mis asesores Daniel U. Campos Delgado y Ruth M. Aguilar Ponce por ayudarme a sacar el trabajo adelante y darme una nueva motivación.

Al Consejo Nacional de Ciencia y Tecnología CONACYT por el apoyo económico brindado. (267879)

A mi compañero de estudios Isnardo Reducindo Ruiz quien me brindo consuelo, risas y apoyo. ¡Lo logramos!. De igual forma a mi vecina de escritorio Cinthia J. Martínez Sánchez por todas sus aportaciones y desveladas.

Finalmente pero no menos importantes a todas las personas que me motivaron a terminar mis estudios de maestría y me dejaron una enseñanza de vida. Gracias Omar, Martín Luna, Martín Méndez, Will, José Manuel, Gustavo, Vero, p.f Daphnne y Azdrubal.

En la investigación es incluso más importante
el proceso que el logro mismo.
Emilio Muñoz

Índice general

Índice de figuras

Índice de tablas

Capítulo 1

Introducción

1.1. Visión estéreo

El cuerpo humano es considerado en ocasiones como ejemplo de un sistema eficiente y eficaz, una "máquina perfecta". Por lo que dotar a la tecnología con replicas de los procesos que se llevan acabo en el cuerpo humano, es un reto inmediato en la optimización o desarrollo de tecnología. Uno de los procesos más complejos es la forma en que el cuerpo humano es capaz de percibir con perfecta precisión donde están los objetos en relación a él mismo. Especialmente cuando esos objetos se están moviendo hacia o lejos de él en la dimensión de profundidad.

El cuerpo humano logra captar dos vistas por medio de los ojos. Es capaz de ver un poco alrededor de objetos sólidos sin mover la cabeza, hasta percibir y medir el espacio "vacío" con los ojos y el cerebro. Cada ojo capta su propia imagen y las dos imágenes separadas se envían al cerebro para su procesamiento [1]. Cuando llegan las dos imágenes simultáneamente en la parte posterior del cerebro, éste combina las dos imágenes tomando en cuenta diferencias relativas en la posición (disparidad) de los objetos dentro de cada imagen, que a su vez tienen una relación directa con las distancias a la que se encuentran los objetos (profundidad) del observador. De tal modo el cerebro es capaz de interpretar esas diferencias y reconstruir la estructura de la escena que ve el observador [2].

De forma general a la capacidad de un sistema visual de recuperar la estructura tridimensional de una escena, a partir de por lo menos dos imágenes de la misma escena, obtenidas de puntos de vista distintos se le denomina visión estéreo o visión estereoscópica [2].

Desde el punto de vista computacional un sistema estéreo debe resolver dos problemas fundamentales. El primero llamado *correspondencia* (matching) que consiste en determinar qué objeto de la imagen izquierda, corresponde a qué objeto en la imagen derecha. Es necesario resolver este problema para obtener información sobre profundidad y es un problema nada fácil de resolver debido a diferentes situaciones, como por ejemplo variaciones de la intensidad luminosa, oclusiones, ruido y errores de muestreo etc. que se abordaran con detalle más adelante.

El segundo problema que debe ser resuelto es la *reconstrucción*. Nuestra percepción en

3D es posible gracias a la interpretación que el cerebro hace de las diferencias en la posición retinal, llamada disparidad, entre los objetos correspondientes [3]. Las disparidades de todos los puntos de la imagen de referencia forman el llamado mapa de disparidad, que puede ser visto como una imagen. Si la geometría de un sistema estéreo es conocida, entonces el mapa de disparidad puede ser convertido a un mapa tridimensional de la escena observada, a lo que se le llama reconstrucción en 3D [3].

La Figura 1.1 muestra la aplicación de un sistema de visión estéreo con un par de imágenes. Las subfiguras (a) y (b) ilustran el problema de correspondencia, mientras en (c) se aprecia la reconstrucción en 3D de la escena (segundo problema que debe resolver un sistema estéreo).

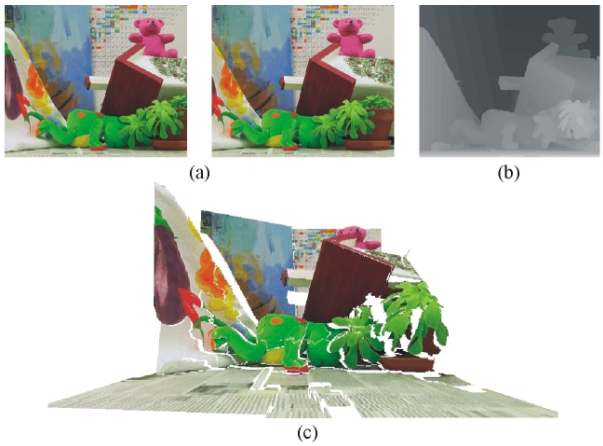


Figura 1.1. Aplicación de un sistema de visión estéreo: (a) Par de imágenes estéreo, (b) Mapa de disparidad, (c) Reconstrucción 3D. [4]

1.1.1. Calibración de cámaras y rectificación de imágenes

Para obtener información sobre la profundidad usando por lo menos dos cámaras, es necesario contar con alguna información de la geometría de las mismas. El proceso por el cual se estiman los parámetros necesarios para evaluar las coordenadas espaciales en base a las de las proyecciones en las imágenes [5], es conocido en visión estéreo como calibración de cámaras. Dentro de este proceso se determinan las líneas de visión de cada una de las cámaras utilizadas, es necesario que estas líneas intersecten el punto X observado de la escena del cual la información de profundidad será procesada (ver Figura 1.2).

La figura 1.2 muestra la geometría general en un sistema estéreo. Los dos centros ópticos F y F' están asociados por una recta llamada línea base. Las líneas de visión que pertenecen a F y F' se intersectan en el punto X y a su vez generan un plano triangular que intersecta cada plano imagen π y π' en líneas epipolares g y g'. Las proyecciónes u y u', respectivamente, del punto X pueden ser encontradas en estas líneas. Todas las posibles posiciones de X caen en la línea FX para la imagen izquierda y en $F'X$ para la imagen derecha. Esto es reflejado en el plano π de la cámara izquierda para FX por la linea g y en π' de la cámara derecha para $F'X$ por la línea g'. Los epipolos e y e' también caen en las líneas g y g' que son intersectadas por la línea base. Cuando π y π' son paralelos a la

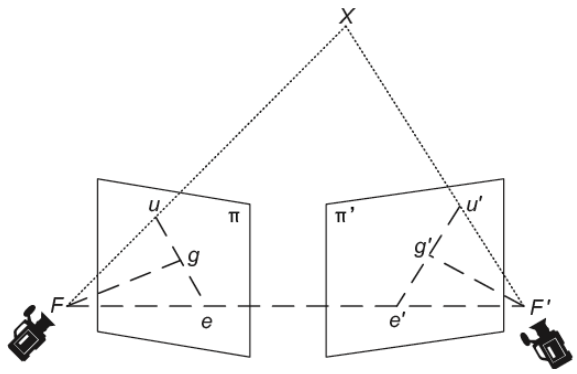


Figura 1.2. Geometría general de un sistema de visión estéreo. [6]

línea base, las líneas epipolares también lo serán. La Figura 1.3 muestra una configuración estéreo simple que es llamada canónica o geometría epipolar estéreo, en la cual la línea base coincide con el eje de coordenadas horizontales y las líneas de visión de las dos cámaras son paralelas. Lo que tiene como consecuencia que los dos epipolos e y e' son infinitos, de tal forma las líneas epipolares corren horizontalmente[7].

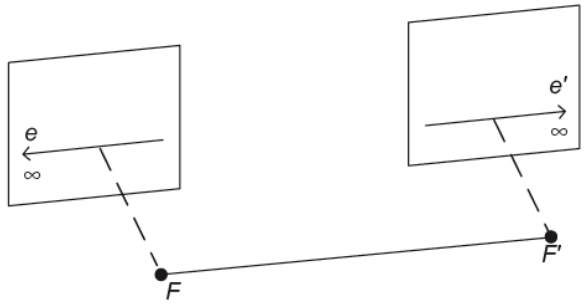


Figura 1.3. Configuración canónica estéreo. [8]

La conclusión de lo anterior para el desarrollo de un proceso automatizado es que los puntos que tienen correspondencia pueden ser encontrados examinando líneas horizontales en lugar de analizar líneas arbitrarias. Lo que simplifica enormemente el proceso de búsqueda de correspondencias, ya que la solución a este problema se transforma de ser bidimensional a unidimensional. En la práctica es difícil lograr una alineación precisa, por ello se requiere no solo de calibración de cámaras sino también de otro proceso llamado rectificación de imágenes, en el que se aplican transformaciónes geométricas a las imágenes para conseguir que las rectas epipolares sean paralelas [9]. El problema de calibración y rectificación es muy extenso y no es de los objetivos de esta investigación, razón por la cual su descripción es breve.

Al rectificar las imágenes antes de buscar las correspondencias se reduce el costo computacional de los algoritmos estéreo, lo que permite aplicaciones prácticas [1], [10], [11]. Sin embargo, debido a las transformaciones geométricas necesarias para lograr la rectificación de las imágenes, estas pierden resolución. Por lo tanto si la resolución es un factor importante en la aplicación, la calibración de las cámaras será la mejor opción.

Existen diferentes métodos propuestos de calibración y rectificación. Como ejemplos de métodos de calibración podemos mencionar el propuesto en [12] que utiliza solo las correspondencias en múltiples imágenes sin tomar en cuenta la posición de las cámaras o la posición de las correspondencias en el espacio 3D. El enfoque estándar de la calibración de distorsión de lentes es un modelo de las desviaciones que existen de una cámara real comparada con una cámara ideal *pinhole*. Dadas múltiples vistas de un conjunto de puntos correspondientes tomadas por la cámara ideal *pinhole*, existen restricciones epipolares entre pares o tercias de estas vistas que en la práctica por el ruido y la distorsión de los lentes no se mantienen exactamente, por lo que se obtiene un error. De tal forma que la calibración se convierte en la búsqueda de los parámetros de la desviación de los lentes que minimizan este error. Es interesante este método de calibración debido a que mejora notablemente la precisión en la reconstrucción 3D de la escena.

Un algoritmo de rectificación basado en una transformación geométrica (warped resampling) se describen en [13], otro método simple y eficaz se propone en [14]. La idea en general del método consiste en utilizar una parametrización polar de la imagen alrededor del epipolo. La propuesta garantiza que para un tamaño mínimo de imagen no existan perdidas de pixeles y como información para lograr la rectificación únicamente necesita la matriz fundamental (matriz que relaciona los puntos correspondientes en las imágenes estéreo algunas veces llamada tensor bifocal). Este método tiene ciertas ventajas sobre los enfoques tradicionales que en algunos casos presentan un pobre rendimiento con imágenes grandes o hasta les es imposible rectificar las imágenes.

1.1.2. Geometría estéreo

En la adquisición de imágenes estéreo existen variables a considerar como son el número de cámaras, el arreglo de estas y su precisión. En este trabajo de tesis la adquisición de las imágenes estéreo no es un aspecto a considerar, ya que se utilizan imágenes sintéticas. Aún así es importante considerar un sistema binocular para situar antecedentes al problema de estimación de disparidad mediante dos vistas.

A continuación se estudiará el caso binocular, que es el más básico y muestra los parámetros principales que deben ser considerados en la geometría estéreo para obtener información sobre la profundidad de un punto en la escena. En la Figura 1.4 se muestra el diagrama de un sistema binocular típico, visto desde arriba donde los planos imagen π y π' son coplanares, y F y F' son los centros ópticos. Los ejes ópticos son paralelos, por esta razón el punto de fijación, definido como el punto donde intersectan los ejes ópticos, se encuentra infinitamente lejos de las cámaras.

Para determinar la posición de los puntos P y Q (figura 1.4(a)) se utiliza la triangulación, esto es, utilizando las intersecciones de las líneas definidas por los centros ópticos y las proyecciones p, p', q, q' de los puntos P y Q. La triangulación depende fundamentalmente de la solución del problema de correspondencia. Si (p, p') y (q, q') son escogidos como pares correspondientes de las imágenes, intersectando las líneas $Fp - F'p'$ y $Fq - F'q'$ conduce a la interpretación de los puntos de la imagen como las proyecciones de P y Q, pero si (p, q') y (q, p') son el par seleccionado como puntos correspondientes, por triangulación resultará en P' y Q'. Observar que ambas interpretaciones, aunque radicalmente diferentes, parten

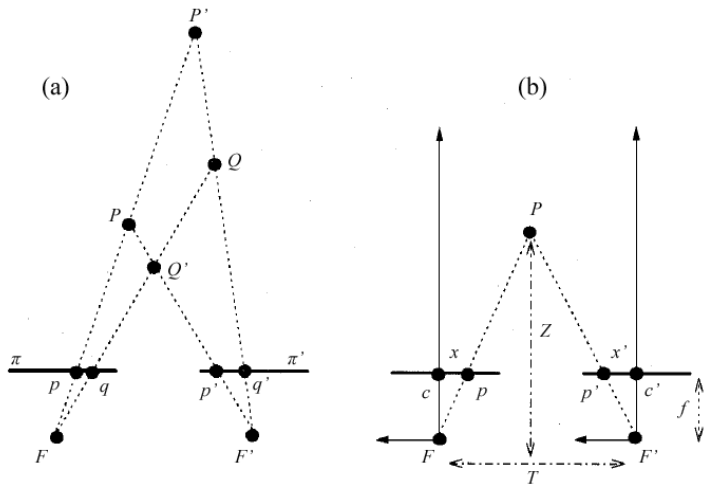


Figura 1.4. Diagrama de un sistema binocular. (a) La reconstrucción 3D depende de la solución del problema de correspondencia. (b) La profundidad se estima a partir de la disparidad de puntos correspondientes [3]

de la misma base.

En las próximas secciónes se tratará más a detalle la solución al problema de correspondencia, por lo pronto supongamos que se ha resuelto y se empieza con la reconstrucción; concentrandonos en recuperar la posición de un solo punto P de sus proyecciones p y p' (figura 1.4(b)). Sea la distancia T entre F y F' la línea base, x y x' las coordenadas de p y p' con respecto a los puntos principales c y c', f la distancia focal y Z la distancia entre P y T, de los triángulos semejantes (p, P, p') y (F, P, F') se puede concluir:

$$\frac{T - (x' - x)}{Z - f} = \frac{T}{Z}. \tag{1.1}$$

Resolviendo para Z

$$Z = f\frac{T}{d}, \tag{1.2}$$

donde $d = x' - x$, la disparidad, que mide la diferencia en la posición retinal entre los puntos correspondientes entre las dos imágenes. A partir de (1.2) se puede concluir que la profundidad es inversamente proporcional a la disparidad [3]. De la misma ecuación podemos apreciar que para un punto de la escena a una distancia fija, si se aumenta T o f, la disparidad aumenta. En otras palabras la línea base o la distancia focal actúan como coeficientes amplificadores de la disparidad. Esta observación es muy relevante en la determinación del error en la medición de la profundidad asociado a una cierta longitud T.

Suponiendo que el sistema ha sido adecuadamente calibrado y alineado, y que las imágenes han sido rectificadas para corregir posibles distorsiones, el proceso de búsqueda de

correspondencias entregará como resultado un mapa de disparidades. Este mapa puede entenderse como una imagen cuyas intensidades no representan luminosidad en la escena, sino más bien la disparidad en cada punto de ella. Los puntos de la escena próximos a las cámaras, es decir aquellos puntos con Z pequeño tendrán disparidades mayores que los puntos lejanos, con Z más grande. Al ser las intensidades de la imagen o mapa de disparidades inversamente proporcionales a las de la imagen o mapa de profundidades Z, visualmente la imagen negativa de un mapa de disparidad es muy similar al mapa de profundidad.

Los resultados que nos interesan de esta investigación son los mapas de disparidad, donde los puntos cercanos a la cámara (puntos con disparidad alta) se representarán en tonalidades de grises claras y los lejanos con intensidades de grises obscuros. El blanco corresponde a la disparidad máxima del intervalo de búsqueda de disparidades y el negro corresponde al mínimo.

1.1.3. Problema de Correspondencia

Después de dar un panorama general de la antesala al problema de correspondencia en visión estéreo, enfoquemonos en explicar los retos que se presentan al intentar encontrar una solución a este problema. Resolverlo es la tarea central de un sistema estéreo y a la vez de las más difíciles, debido a diferentes factores bien documentados en textos como [2, 3, 9, 15, 16]. Estos factores de forma general son:

- Oclusiones que son elementos que no tienen correspondencia. En otras palabras elementos en la escena que aparecen en una de las vistas y que no se perciben en las otras, debido frecuentemente a la forma del objeto o a la presencia de otro. Este factor depende principalmente de la taza de cambio en la profundidad y la separación de los centros ópticos.

- Cambios en las características del elemento entre las diferentes vistas de la misma escena que se deben a alteraciones en la dirección de iluminación (cambio en la intensidad), efectos de escorzo (cambio de tamaño), ruido y errores de muestreo.

- Correspondencias falsas que se producen cuando existen elementos muy similares alrededor del verdadero correspondiente. Supongamos que existe dentro de la escena una textura que presenta un patrón periódico como una pared de ladrillos, esto aumenta el número de puntos similares o correspondencias, lo que ocasiona correspondencias falsas.

Para resolver ambigüedades o aminorar los efectos de los factores antes mencionados se requiere el uso de restricciones adicionales y supuestos. Siendo de las más utilizadas las que se describen a continuación [9, 17]:

- Restricción epipolar: esta restricción permite reducir el espacio de búsqueda de dos dimensiones a una dimensión.

- Restricción de continuidad: En este caso se suponen superficies suaves. Haciendo posible descartar disparidades que no sean similares a disparidades vecinas.

- Restricción de oclusión: Es fácil detectar y eliminar oclusiones, comparando la vista

1 con la vista 2 y viceversa, la vista 2 con la vista 1. De esta manera, es posible descartar aquellos elementos que no tengan correspondencia o no tengan posibilidad de ser correlacionados.

Ahora bien, ignoremos los parámetros de las cámaras y supongamos que la mayoría de los puntos de la escena son visibles en ambas imágenes, y las áreas correspondientes de las imágenes son similares. Las suposiciones anteriores transforman el problema de correspondencia en solo una búsqueda de que elemento en la imagen izquierda corresponde a que elemento en la imagen derecha. Estas suposiciones se mantienen para un sistema estéreo en donde la distancia del punto de fijación a la cámara es mayor a la línea base ($Z > T$). En general estas suposiciones pueden no cumplirse. Lo que tendría como consecuencia que el problema de correspondencia fuera considerablemente más complicado. Ahora bien el problema de búsqueda empezaría con dos importantes decisiones, que elementos van a ser comparados y que medida de similitud se aplicará para determinar su correspondencia.

Existen numerosas opciones que pueden ser utilizadas como elementos a comparar, por ejemplo pixeles (unidad homogénea más pequeña en color que forma parte de una representación bidimensional de una imagen digital [18]) que contienen información de intensidad lumínica. Otros elementos pueden ser bordes, esquinas o hasta regiones completas de las imágenes, lo que presentaría el problema de segmentar las imágenes, etc. Las medidas de similitud aplicadas dependerá en gran parte del elemento que se decida comparar.

Existen distintos métodos para encontrar la correspondencia entre pixeles por ejemplo los métodos basados en correlación o la suma de diferencias al cuadrado (SSD por su acrónimo en inglés, Sum of Squared Differences). Los elementos a emparejar en estos métodos locales son encontrados por medio de valores de intensidad de los pixeles. Analizar por si solo cada pixel no es suficiente, porque se encuentran muchos candidatos con el mismo valor de intensidad, por lo que es necesario examinar los pixeles vecinos. Esto se lleva acabo a través de bloques de pixeles, de tal forma que los elementos a emparejar son ventanas de la imagen de tamaño fijo y el criterio o medida de similitud es la correlación cruzada o SSD entre las ventanas de las dos imágenes [18]. El pixel en correspondencia está dado por la ventana que minimiza el criterio de error respecto a la disparidad d.

Los métodos de correlación y SSD son fáciles de implementar y verificar además de producir mapas densos de disparidad (a cada píxel de una imagen se le hace corresponder otro en la otra imagen), lo cual es muy conveniente a la hora de la reconstrucción de las superficies. Sin embargo se requieren imágenes con textura para que estos métodos trabajen correctamente. Sin embargo, debido a los efectos de movimiento y cambio de iluminación, son muy inadecuados para emparejar imágenes tomadas desde puntos de vista muy diferentes. La eficiencia de este tipo de métodos puede disminuir notablemente por la existencia de oclusiones y correspondencias falsas creadas por el ruido [2]. La implementación de este tipo de algoritmos es sencilla pero nada útil para lograr altas velocidades de procesamiento, debido a la extensa búsqueda que debe realizarse para encontrar los pares de puntos correspondientes, lo que resulta en un alto costo computacional.

1.2. Aplicaciones de visión estéreo

Actualmente diversos campos científicos y técnicos se benefician de la estereoscopía. En el área de robótica ha sido posible dotar a ciertas maquinas de habilidades para identificar el terreno por explorar y adecuar su perfil de movimiento. Goldberg en [19] describe el sistema de visión estéreo utilizado por un vehículo para exploraciones espaciales en el 2004 por la NASA, el cual tiene la habilidad de navegar de manera segura por terreno desconocido y potencialmente hostil.

Una de las aplicaciones prácticas más antigua es la visualización y medición del relieve terrestre mediante fotografías aéreas, actualmente existe software especialmente diseñado para tareas de este tipo, como los de Intergraph y Zeiss que incluyen representaciones del relieve submarino [20, 21].

Por otro lado, se han podido desarrollar también herramientas como CAD (Diseño Asistido por Computador) y CAE (Ingeniería Asistida por Computador) para diseño y visualización de prototipos, principalmente en la industria automotriz. Chrysler, Ford, Opel, Renault, Volvo y otros fabricantes ya usan estas técnicas, con un importante ahorro en tiempo y dinero durante el desarrollo. Los más importantes paquetes y estaciones de diseño por ordenador, como IBM, HP, DEC, Sun o Silicon Graphics , soportan actualmente la visualización estereoscópica mediante gafas LCS, como las de Stereographics o VRex. En [22] se describe con más detalle como se realizan los modelos CAD y las dificultades que presenta la visión estéreo en este tipo de aplicaciones.

La medicina es uno de los campos en los que la estereoscopía proporciona más ayuda para la enseñanza, la interpretación de imágenes para el diagnóstico o como ayuda en las intervenciones quirúrgicas. Se usa para visualizar imágenes o modelos del interior del cuerpo humano a partir de imágenes reales obtenidas por medio de TAC (Tomografía Asistida por Computador) o RMN (Resonancia Magnética Nuclear). Técnicas como la radiografía estereoscópica permiten situar claramente cuerpos extraños o anomalías en el interior del paciente. En [23] se demuestra la viabilidad y utilidad de usar un sistema en análisis de tres dimensiones basado en imágenes de estación de trabajo quirúrgico que se ha modificado para permitir la presentación de imágenes estereoscópicas.

1.3. Objetivos de la tesis

El objetivo principal de este estudio es analizar y evaluar la eficiencia de la correlación por fase para la estimación de disparidad. Para evaluar la técnica propuesta se analizaran las siguientes características que debe cumplir la estimación de disparidad:

- Se analizara la robustez del algoritmo por medio de la evaluación del algoritmo usando diversas escenas que nos permitirán medir su desempeño para imagenes con precisión entera y subpixelica. Además analizaremos el desempeño del algoritmo en presencia de objetos con volumen y sin él.
- Se analizara el desempeño del algoritmo por medio de la variación de los parametros del algoritmo tales como: tamaño de la ventana de correlación, numero de candidatos.

Se medirá el impacto de estos en la rapidez y precisión del algoritmo.

- Se implementará el algoritmo y se medirá el tiempo de procesamiento para darnos un indicativo de la velocidad del algoritmo. Además, se establecerá la reducción promedio del espacio de búsqueda.

1.4. Organización de la tesis

En primer lugar se presenta una introducción al área de visión estéreo y a la terminología básica para la comprensión de las secciones posteriores. En el primer capítulo también se presentan algunas de las aplicaciones más recientes de la visión estéreo, haciendo distinción entre aplicaciones que no están sujetas a restricciones de tiempo y aquellas en tiempo real.

En el segundo capítulo se presenta un panorama general del estado del arte. Primero se presenta la clasificación más actual para los algoritmos estéreo, posteriormente se presentan los algoritmos más recientes que trabajan en tiempo real y se termina describiendo la función de correlación por fase. Enseguida se presentan algunas de sus aplicaciones de forma general como introducción a la técnica en la que se basa el algoritmo propuesto. Finalmente se describen algunos algoritmos estéreo que utilizan la correlación por fase para la estimación de disparidad. La metodología del trabajo desarrollado se encuentra descrita en el tercer capítulo, el cual se inicia con una pequeña introducción a modo de resumen de los puntos de interés de la terminología básica y aspectos relevantes para nuestra implementación. Los resultados experimentales y su análisis se presentan en el capítulo cuatro. El quinto y último capítulo discute los alcances del nuevo algoritmo y se proponen nuevas líneas de investigación a partir del trabajo realizado.

Capítulo 2

Panorama general del estado del arte

2.1. Clasificación de los métodos de estimación de disparidad

Muchos son los algoritmos y las técnicas para lograr obtener un mapa de disparidad aproximado. En los últimos años varias publicaciónes han propuesto una clasificación y actualización del estado del arte para algoritmos estéreo [25] y facilitan una plataforma de comparación de los distintos métodos existentes [26].

Brown y otros [25] presentan una revisión de los avances en visión estéreo, haciendo una distinción entre distintos métodos e implementaciones. El articulo sé concentra en métodos de correspondencia, métodos que tratan con oclusiones e implementaciones en tiempo real. La clasificación que hace para presentar los distintos algoritmos que se han propuesto en la década anterior al año 2003 es la siguiente:

1. **Métodos de correspondencia:** son métodos que toman pixeles de una imagen y se dedican a encontrar sus pixeles correspondientes en el par estéreo. Existen distintos métodos para encontrar la correspondencia entre pixeles los cuales pueden dividirse en globales y locales [26, 30, 29].

    - Locales: los métodos locales son menos precisos, aplican restricciones sobre un pequeño número de pixeles alrededor de un pixel de interés [32, 33, 34].

        - *Block Matching*: Son los métodos locales más populares, buscan la máxima puntuación de correspondencia o el error mínimo sobre una región pequeña o una ventana centrada alrededor del pixel x, en la imagen de referencia y otra ventana centrada alrededor del pixel $x + d$ en la otra imagen, típicamente usando variantes de la correlación cruzada, la suma de diferencias al cuadrado (SSD), la suma de diferencias absolutas (SAD) o métricas robustas.

        - *Gradient-Based Optimization*: Minimizan una función que usualmente es la

suma de diferencias al cuadrado sobre una región pequeña a través de una búsqueda dirigada por la información del gradiente.

  - *Feature Matching*: Realizan una correspondencia con base en características especificas en lugar de valores de intensidad.

- Globales: Se entiende por métodos globales aquellos que aplican restricciones sobre toda la imagen o sobre líneas enteras de esta. Son los que buscan minimizar una función de energía global formada por distintos términos. Algunos de estos términos penalizan las diferencias entre el par de imágenes estéreo y otros penalizan cambios abruptos en los valores de disparidad, particularmente en áreas de un mismo objeto donde la disparidad se supone homogénea. Minimizar estas funciones de energía no es una tarea trivial, comúnmente se realizan mediante técnicas iterativas de alto costo computacional, como por ejemplo técnicas de descenso de gradiente, cadenas de Markov vía Monte Carlo (MCMC), etc [31, 35].

  - *Dynamic Programing*: Es un método matemático con el cual la complejidad computacional de problemas de optimización puede ser reducida, dividiendo el problema de optimización en problemas más pequeños y simples. Una función global es calculada en etapas, con ciertas restricciones entre cada una. Para correspondencia estéreo la restricción de orden epipolar permite que la función de costo global pueda ser determinada como la trayectoria de costo mínimo a través de una imagen conocida como Disparity Space Image (DSI) formada de una representación que se construye de las posibles correspondencias para cada punto. El costo de la trayectoria óptima es la suma de los costos de las trayectorias parciales obtenidas de manera recursiva.

  - *Intrinsic Curves*: Utiliza una representación diferente de los renglones de la imagen, llamada curvas intrínsecas o *intrinsic curves*, que son representaciones vectoriales de distintas propiedades de los pixeles de un renglón. En general el método mapea renglones epipolares a curvas intrinsecas para convertir el problema de búsqueda de correspondencias a un problema de *Nearest-Neighbor-Lookup* (dado un conjunto de N puntos en un espacio n-dimensional y una posición q, se busca dentro de N el punto p que resulte con la mínima distancia Euclidiana de q).

  - *Graph Cuts*: El corte de grafos se basa en armar un grafo a partir de los datos de las imágenes y buscar un corte mínimo. Dependiendo como se arma el grafo, el resultado obtenido es la minimización de una cierta expresión de energía. Este procedimiento se puede considerar análogo al de hallar el mejor camino en una imagen bidimensional, con programación dinámica, pero extendido a un espacio tridimensional con coherencias en las dos dimensiones.

2. **Métodos que tratan con oclusiones:** Existen métodos que se han enfocado en la solución de regiones ocluidas con el fin de alcanzar mayor precisión. Según el objetivo del método se pueden clasificar en tres grupos que a continuación se describen brevemente.

- Métodos que detectan oclusiones : Los enfoques más sencillos que intentan manejar el problema de las oclusiones se limitan a intentar detectar las oclusiones antes o después de resolver las correspondencias. Estas regiones son entonces interpoladas según la información de disparidades vecinas para producir mapas densos de profundidad, o simplemente descartadas en aplicaciones que requieren sólo un mapa parcial de profundidad.
  - *Discontinuidades en mapas de disparidad*: Algoritmos que tratan las discontinuidades como si fueran oclusiones.
  - *Left-Right Matching*: Aquellos algoritmos que estiman dos mapas de disparidad, uno utilizando la imagen izquierda como referencia y otro utilizando la imagen derecha como referencia, de tal forma que las inconsistencias entre ambos mapas resultantes son consideradas regiones ocluidas de la escena.
  - *Intensidad de bordes*: Evitan las oclusiones con base a segmentos correspondientes delineados por la intensidad de los bordes, partiendo de la suposición de que los límites de las regiones de oclusión y las intensidades de los bordes son coincidentes; lo cual en este tipo de algoritmos sirve para detectar y descartar las oclusiones pero esta misma suposición es utilizada para modelar las regiones ocluidas.
  - *Restricción de orden*: La restricción de orden en visión estéreo se refiere a que para un punto m y otro n en la imagen izquierda existen puntos correspondientes m' y n' en la imagen derecha, respectivamente. De tal forma que si m está a la izquierda de n entonces m' debe encontrarse también a la izquierda de n'. Esta restricción es utilizada en los algoritmos para detectar oclusiones. Se verifica la suposición y en caso de no cumplirse se asume que se trata de una oclusión.
- Métodos que reducen la sensibilidad a oclusiones.
  - *Criterio de similitud robusto*: Aquellos métodos que utilizan medidas de similitud robustas como *least median of squares* (LMS), *least trimmed squares* (LTS), estimadores M, Hough transforms etc.
  - *Regiones adaptivas de apoyo*: Estos métodos varían el tamaño de la ventana de correlación que utilizan para determinar la similitud de pixeles. Las ventanas son inicializadas con tamaños muy pequeños y aumentan o fijan su tamaño según el aumento de las regiones donde no es posible conocer la disparidad.
- Métodos que modelan la geometría de las oclusiones.
  - *Modelaje de oclusiones globales*: Se realiza un modelado de las oclusiones y se incluye está información en la búsqueda de correspondencias, por lo general se utiliza programación dinámica (por su acrónimo en inglés DP) para su implementación.
  - *Múltiples cámaras*: El uso de múltiples cámaras asegura que cada punto en la escena es visible en por lo menos dos cámaras.

- *Vision activa*: Aquellos métodos que mueven la cámara o el espacio de correspondencias para asegurar que las regiones ocluidas encuentren correspondencia.

3. **Métodos en Tiempo Real:** Se definen como aquellos algoritmos que logren alcanzar una velocidad igual o mayor a las 30 cuadros por segundo. Este tipo de algoritmos surgen gracias al desarrollo tecnológico y abren la posibilidad de nuevas aplicaciones en visión estéreo y por tal razón Brown les otorga una nueva categoría.

Por otro lado, Scharstein y Szeliski en [26] dan una clasificación más descriptiva, incluyendo el desempeño de los distintos algoritmos que incluyen en su estudio. Su clasificación se basa en la observación de cuatro etapas que son fundamentales en la mayoría de los algoritmos estéreo. Las cuatro etapas a las que se hace referencia son:

1. Costo de estimación de correspondencias.
2. Costo agregado.
3. Estimación de disparidad y método de optimización.
4. Refinamiento de la disparidad.

Es importante recalcar que las etapas y la secuencia en que se realizan dependen de cada algoritmo en particular. Cabe señalar que lo escrito en esta sección se basa principalmente en las clasificaciones y reseñas que se encuentran totalmente descritas en los artículos [25] y [26]. Ademas de que la clasificación presentada por Scharstein y Szeliski es únicamente para sistemas binoculares (dos vistas).

2.2. Métodos eficientes para la estimación de disparidad

La eficiencia de los algoritmos depende de la aplicación a la que van dirigidos, por lo que es importante aclarar que en esta sección nos referimos como algoritmos eficientes a aquellos que trabajan en tiempo real o casi tiempo real, y además mantienen resultados de buena calidad. Se pretende dar un panorama general del estado del arte, lo más actual posible, de los algoritmos eficientes en la estimación de disparidad.

El problema de correspondencia no solo presenta el reto de alcanzar precisión, sino también bajo costo computacional, especialmente cuando se requieren resultados densos (un valor de disparidad por pixel). Los algoritmos de *block matching* suelen aplicar una búsqueda exhaustiva para solucionar el problema de correspondencia, esto es evidente al observar el conjunto de desplazamientos candidatos d que suelen ser muy grandes. Por ejemplo si el valor del desplazamiento máximo que se desea encontrar es de 14 pixeles, el pixel correspondiente a x debe de estar en una ventana de 28×28 pixeles centrada en $x + d$. Por lo tanto habrá 784 posibles candidatos en el espacio de búsqueda, para reducir el espacio de búsqueda muchos algoritmos utilizan distintas restricciones por ejemplo la restricción epipolar, que implica que el correspondiente de un punto en una imagen debe estar en la recta epipolar del punto en la otra imagen [27, 28]. Esta restricción reduce el espacio de

búsqueda a una recta o dicho de otra forma, reduce la búsqueda de correspondencias a una dimensión.

En términos de precisión, los algoritmos estéreo en el estado del arte pueden ser evaluados por medio del porcentaje promedio de error (igual que en la ultima columna "average porcentage of bad pixels" en el sistema de evaluación en linea proporcionada por Scharstein y Szeliski en `http://vision.middlebury.edu/stereo/`) para 4 pares de imágenes estéreo *benchmark* (Tsukuba, Venus, Teddy y Conos) y son clasificados dentro de 3 clases: muy buena calidad (tasa de error por debajo de 7.0), buena calidad (tasa de error entre 7.0 y 11.0) y mala calidad (tasa de error sobre 11.0).

Yang y otros en [36] presentan un algoritmo global basado en *Hierarchical Belief Propagation* que actualiza el costo por pixel de forma adaptiva, siendo un algoritmo global la precisión de los mapas de disparidad resultantes es bastante alta, la estructura de las escenas y los bordes de los objetos son detectados de manera correcta. La eficiencia de este algoritmo se debe al uso del paralelismo de hardware gráfico. El artículo presenta resultados integrando el algoritmo a un sistema estéreo con entrada de video en tiempo real, donde el tamaño de las imágenes de la escena real es de 320×240 pixeles con 16 niveles de disparidad; la velocidad que alcanza el sistema es de aproximadamente de 16 cuadros por segundo (fps, por sus siglas en inglés). El método propuesto y la velocidad lograda, coloca a este algoritmo como uno de los más eficientes actualmente. En la tabla de evaluación de Scharstein y Szeliski este método se encuentra con un Avg. Rank de 70.9 y un porcentaje de error de 10.7 que confirma su eficiencia.

Otros ejemplos de algoritmos eficientes son el propuesto por Salmen, Schlipsing y otros en [37], donde se describe un algoritmo que utiliza programación dinámica (DP) para la estimación de disparidad y que a diferencia de los métodos tradicionales introduce la idea de explotar la información obtenida del uso de DP mediante multi-trayecto *backtracking*, con esto el número de disparidades incorrectas se reduce en 40 % comparado a los métodos tradicionales de DP, mientras que la complejidad no aumento de manera significativa. La velocidad reportada del algoritmo en imágenes de tamaño 384×288 es de 0.2 segundos en una PC de 1.8 GHz. Kosov, Seidel y otros en [38] proponen una técnica adaptiva multi-nivel que es combinada con un enfoque de "multiresolución" (*multigrid*) para lograr el desempeño en tiempo real, con lo que se reduce el esfuerzo computacional y asegura que la calidad de reconstrucción se mantiene casi igual. La velocidad lograda utilizando una PC de 2.38 GHz Intel Pentium CPU, ejecutando código en C++ y utilizando imágenes de 384×288 pixeles es de 0.6 fps hasta 2.15 fps dependiendo del tipo de ciclos que se utilice en el algoritmo (V,W, O). Este método se encuentra con un Avg. Rank de 61.6 y un porcentaje de error de 9.05.

Los métodos mencionados anteriormente son globales pero también existen métodos locales como el descrito en [39] implementado sobre una unidad de procesamiento gráfico (GPU), el cual aumenta la eficiencia Pareto del compromiso entre precisión y velocidad. Se proponen dos algoritmos estéreo: peso adaptivo con paso exponencial (ESAW por su acrónimo en inglés Exponential Step Size Adaptive Weight) el cual disminuye la complejidad computacional sin sacrificar precisión en la disparidad, permite que la información de costo se propague de pixeles distantes al pixel de interés en pocas iteraciones. Además, se propone una propagación de mensaje con paso exponencial (ESMP por su acrónimo en inglés Exponential Step Size Message Propagation) que es una extensión del ESAW e

incorpora un termino de suavizado comúnmente utilizado en métodos globales como *Belief Propagation*. Este método se encuentra con un Avg. Rank de 59.2 y un porcentaje de error de 8.2. Otra propuesta local es presentada por Yang y Pollrfeys en [40] implementada con una tarjeta gráfica NVIDIA GeForce4, logra estimar mapas de disparidad en 71.4 segundos con imágenes de 640 $\times$ 480 pixeles. Para lograr buenos resultados en discontinuidades, así como en áreas de baja textura se utiliza un enfoque multi-resolucion, el cual eficientemente combina el cálculo de SSD para ventanas de diferentes tamaños.

Los algoritmos propuestos en esta última década son bastantes, la tendencia ha sido utilizar programación en paralelo para implementar métodos globales en tiempo real o casi en tiempo real, debido a su demanda computacional. Los métodos globales son algoritmos de alta calidad pero requieren de hardware más complejo y por lo tanto de alto costo, mientras los métodos locales pueden alcanzar altas velocidades sin la necesidad de requerir hardware fuera de lo más popular en el mercado. A pesar de obtener resultados de buena calidad el compromiso que existe entre precisión y velocidad es una limitante en cualquier caso, y la elección de cualquier método sigue dependiendo de las necesidades de su aplicación.

2.3. Métodos basados en correlación por fase

Existen diferentes estrategias que se han utilizado para reducir el costo computacional de los algoritmos. En este sentido, se han propuesto métodos basados en el dominio de la frecuencia [41], donde la idea clave es ganar velocidad al expresar un criterio de correspondencia, como el SSD, en términos de una convolución para lograr hacer eficiente su cálculo en el dominio de la frecuencia realizando un producto en lugar de una convolución. Otra estrategia es la de costo agregado [50], la cual trata de reemplazar el costo de asignar una disparidad d a un píxel dado $p(x)$, con el costo promedio de asignar d para todos los pixeles de una ventana cuadrada centrada en el píxel $p(x)$. Este enfoque de ventana cuadrada, supone implícitamente que todos los pixeles de la ventana tiene un valor de disparidad similar al píxel $p(x)$, que resulta en malos resultados en áreas cercanas a valores de disparidad discontinuas, como los bordes de los objetos, donde el supuesto anterior no se cumple, por lo que se pierde precisión pero se gana velocidad.

Aoki y otros en [42] utilizan la correlación por fase para encontrar correspondencia entre huellas digitales. En este trabajo, se define propiamente la función de correlación por fase o Phase Only Correlation (POC) y describen tres propiedades de ella (invarianza en corrimiento, invarianza en cambios de luminosidad e inmunidad al ruido). Señalan también una de las principales ventajas de la correlación por fase comparada a la correlación tradicional, ya que se obtiene un impulso sin distorsión al comprar 2 imágenes idénticas, a comparación del pico acompañado de ruido de una correlación tradicional. Este estudio abre camino para posteriores aplicaciones en registro de imágenes con precisión sub-pixel como el descrito en [43]. Nuevamente Akoi y otros en [44] describen un método de búsqueda de correspondencias entre dos imágenes utilizando la función POC como medida de similitud, en lugar de utilizar medidas tradicionales como SAD o SSD, debido a la precisión y robustez que proporciona. Utilizar la función POC como medida de similitud en los algoritmos estéreo tiene un desempeño eficiente pero el costo computacional es un problema para aplicaciones en tiempo real, lo que motivo publicaciones como [45] que describen

métodos para reducir el costo computacional utilizando la función POC en una dimensión y de banda limitada para algoritmos locales que utilizan la comparación entre ventanas. Dando como resultado una técnica de bajo costo computacional y con mejores resultados que las medidas tradicionales.

Capítulo 3

Metodología

El objetivo en visión estéreo, es recuperar la información de profundidad o tercera dimensión, a partir de imágenes en dos dimensiones. El proceso para la recuperación de profundidad, incluye el cálculo de la correspondencia, esto es, qué pixel de la imagen izquierda, corresponde a qué pixel de la imagen derecha [3].

Que se puede expresar como $f\,(x, y) = g\,(x - d, y)$, donde d representa un desplazamiento entero y f y g son la imagen izquierda y derecha con líneas epipolares paralelas (rectificadas) respectivamente.

La solución de este problema es una tarea fundamental y nada fácil de resolver en visión estéreo. Debido a diferentes situaciones que se pueden presentar y ocasionar correspondencias falsas. Como por ejemplo regiones homogéneas, donde un pixel puede tener correspondencia con muchos otros pixeles de la región. Otro caso es el ruido que genera diferencias artificiales, o bien el caso de oclusiones que son regiones que se ven en una imagen y que no se ven en la otra, porque las oculta un objeto.

Este proyecto de tesis toma bases metodológicas del trabajo realizado por A. Alba y E. Arce-Santana [30]. Utiliza el hecho de que un desplazamiento circular en el dominio del tiempo se traduce en un cambio de fase en el dominio de la frecuencia, según la propiedad de traslación de la transfromada de Fourier [46]. Sea $\mathcal{F}(\omega)$ la transformada de Fourier de $f(x)$, y $g(x) = f(x + d)$ donde $d > 0$ es un número entero, entonces la transformada de Fourier de $g(x)$ está dada por $\mathcal{G}(\omega) = e^{j\omega d}\mathcal{F}(\omega)$ y el espectro cruzado normalizado $R(\omega)$ entre $\mathcal{F}$ y $\mathcal{G}$ está dado como:

$$R(\omega) = \frac{\mathcal{F}(\omega)\mathcal{G}^*(\omega)}{|\mathcal{F}(\omega)\mathcal{G}^*(\omega)|} = e^{-j\omega d} \tag{3.1}$$

La transformada inversa de Fourier $r(x)$ de $R(\omega)$ es llamada *phase-only correlation* (POC) o correlación por fase. Esta técnica es usada comúnmente en registro de imágenes [42, 47, 48, 49] y ha sido aplicada en métodos locales para la estimación de disparidad [44, 50, 51] demostrando ser una técnica eficiente en la solución del problema de correspondencia.

Además en este proyecto de tesis se introduce una técnica para reducir el espacio de búsqueda, implementando un método local que se basa en encontrar la posición de múltiples

picos resultantes de la función POC, lo cual contrasta con la mayoría de los métodos de *block matching* basados en correlación por fase, que utilizan solo el pico más significativo de la función POC calculado sobre una ventana. La técnica que se propone parte de la idea de que si una ventana más grande es utilizada, habrá varios objetos con distintos desplazamientos, entonces la función POC mostrará varios picos e intuitivamente algunos de esos picos corresponderán a los desplazamientos de los distintos objetos que se encuentran en la ventana.

La función POC nos ayuda a identificar la magnitud y dirección de algún cambio en la posición de un objeto. Considere un par de imágenes $f(x, y)$ y $g(x, y)$ de tamaño $nc \times nr$, que representan la misma escena en la cual existe un desplazamiento uniforme para todos los objetos de la imagen, incluyendo el fondo. Si se aplica la función POC entre ellas y el desplazamiento es nulo, ver figura 3.1, el resultado será un único impulso ubicado en el origen. Entiéndase por origen la posición donde el desplazamiento es cero y para fines ilustrativos localizado en $(\frac{nc}{2}, \frac{nr}{2})$, que corresponde al centro de la imagen. Si el desplazamiento es uniforme y distinto de cero, la función POC produciría un impulso localizado en $(\frac{nc}{2} - d_x, \frac{nr}{2} - d_y)$, donde d_x y d_y representan los desplazamientos en x y y respectivamente (ver figura 3.2).

Ahora bien, si distintos desplazamientos se observan para distintos objetos, la función POC mostrará varios impulsos, donde cada impulso estará relacionado con algún desplazamiento. En la figura 3.3 se muestra una imagen sintética $f(x, y)$, que contiene 3 objetos distintos. A cada objeto se le aplicó un desplazamiento distinto para generar una segunda imagen $g(x, y)$, y al aplicar la función POC entre estas dos imágenes se obtienen 3 impulsos, que se pueden relacionar con el desplazamiento de cada objeto. Tome en cuenta que al aumentar los objetos en movimiento dentro de la escena, el ruido en la función POC también aumenta debido a los bordes, posibles oclusiones y regiones homogéneas.

El objetivo de este trabajo de tesis es analizar el algoritmo para estimar mapas de disparidad densos, de un par de imágenes estéreo rectificadas horizontalmente, basado en el hecho de que las posiciones de los máximos (picos) más significativos de la función POC, están relacionadas con los desplazamientos de los objetos de una escena. Para el caso de imágenes rectificadas los pixeles correspondientes siempre se encontrarán en la misma fila, por lo que el problema de correspondencia puede resolverse con una comparación de renglón a renglón.

Es importante considerar que objetos grandes en movimiento mostrarán picos significativos en la función POC, pero también introducirán una mayor cantidad de ruido que puede traslapar los picos correspondientes a objetos más pequeños. Si hay un gran número de objetos en movimiento en la escena, la función POC será más compleja y ruidosa, por lo tanto los desplazamientos correctos serán más difíciles de recuperar. En una imagen grande es más probable que exista un mayor número de objetos, cuyo tamaño es más pequeño respecto a la imagen completa, por lo que la estimación de desplazamientos será menos confiable. Por esta razón es importante dividir las imágenes en sub-imágenes más pequeñas y estimar un conjunto de candidatos (dados por los picos de la POC) para cada una de ellas. Muy probablemente cada una de las sub-imágenes tendrá menos objetos respecto a la imagen completa, por lo tanto la función POC será más robusta, en el sentido que los picos más significativos corresponderán con mayor probabilidad a los desplazamientos correctos.

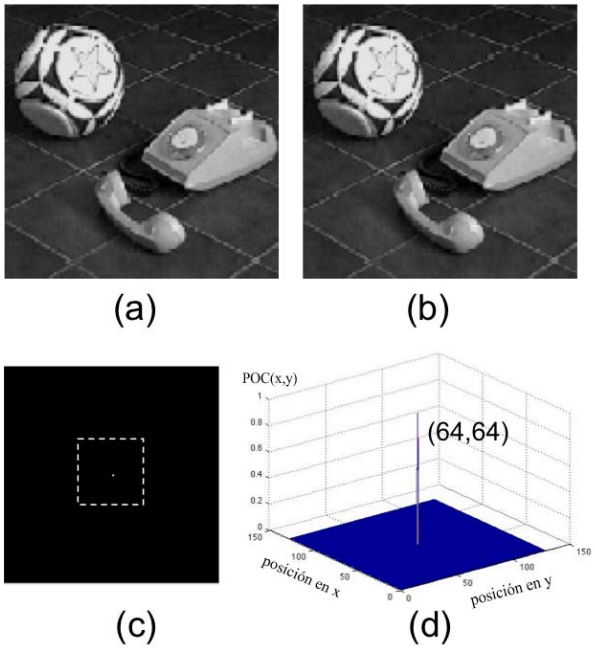


Figura 3.1. POC entre dos imágenes idénticas: (a) Imagen $f(x, y)$ de tamaño 128×128, (b) $f(x, y)$ de tamaño 128×128, (c) POC en 2D, (d) POC en 3D. Note que el pico de la POC esta en la posición $(\frac{nc}{2}, \frac{nr}{2})$ que corresponde a un desplazamiento de $(0, 0)$

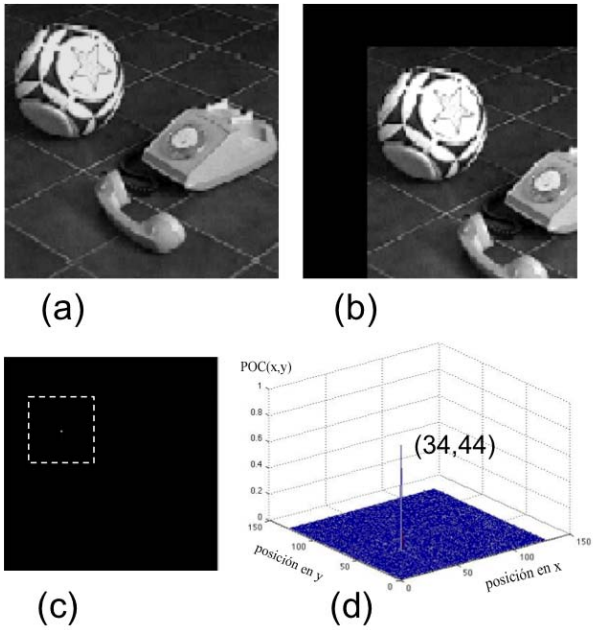


Figura 3.2. POC entre una imágen y su versión desplazada: (a) Imagen $f(x, y)$ de tamaño 128×128, (b) $g(x, y)$ de tamaño 128×128 desplazada 30 pixeles en x y 20 pixeles en y , (c) POC en 2D, (d) POC en 3D. Note que el pico de la POC esta en la posición $(\frac{nc}{2} - d_x, \frac{nr}{2} - dy)$ que corresponde a un desplazamiento de $(30, 20)$

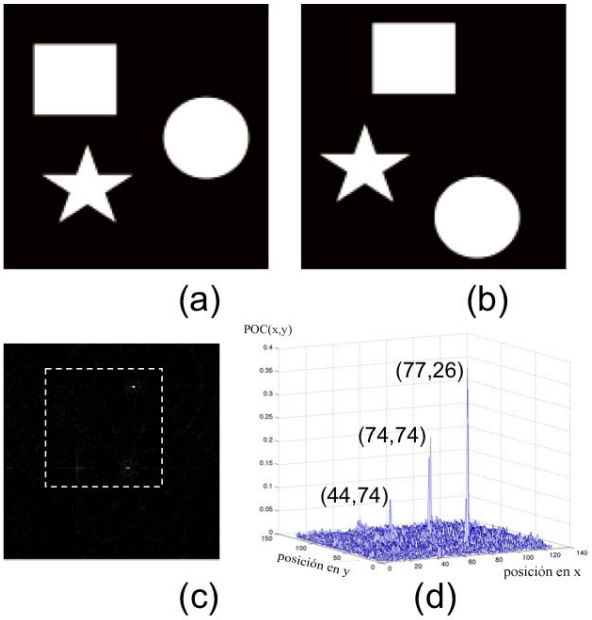


Figura 3.3. POC entre imágenes con objetos iguales y con distintos desplazamientos para cada uno: (a) Imagen $f(x, y)$ de tamaño 128×128, (b) $g(x, y)$ de tamaño 128×128 cada figura con un desplazamiento en pixeles de: círculo $(x+13, y-38)$, rectángulo $(x+13, y+13)$ y la estrella $(x-20, y+13)$, (c) POC en 2D, (d) POC en 3D. Note que los picos de la POC están en las posiciónes $(\frac{nc}{2} - d_x, \frac{nr}{2} - dy)$ que corresponden a los desplazamientos de cada objeto.

3.1. Método Básico

Considere un par de imágenes estéreo f y g rectificadas (los pixeles correspondientes pertenecen al mismo renglón) en escala de grises. De forma que el modelo que describe la imagen izquierda $f(x, y)$ en términos de la imagen derecha $g(x, y)$ es el siguiente:

$$f\,(x, y) = g\,(x - d, y) \tag{3.2}$$

donde $d \geq 0$ es el desplazamiento que tiene cada pixel, notese que la imagen formada por esos desplazamientos es el mapa de disparidad $d(x, y)$.

Un algoritmo de búsqueda exhaustiva común de *block matching* [52], encuentra para cada pixel, el valor de disparidad $d \in \{0, ..., D\}$, que minimiza una función de error local $\Phi(x, y, d)$, que mide la diferencia entre una vecindad de g y su vecindad similar en f. Por lo tanto, la disparidad para cada pixel está dada por:

$$d(x, y) = \underset{d' \in \{0,...,D\}}{\arg\min} \left\{\Phi(x, y, d')\right\} \tag{3.3}$$

donde D es un rango dado de disparidad. Como medidas de error podemos utilizar SAD y SSD, respectivamente definidas como:

$$\Phi_{SAD}(x, y, d') = \sum_{k=-w}^{w} \sum_{l=-w}^{w} \left|f(x + d' + l, y + j) - g(x + l, y + k)\right| \tag{3.4}$$

$$\Phi_{SSD}(x, y, d') = \sum_{k=-w}^{w} \sum_{l=-w}^{w} \left(f(x + d' + l, y + j) - g(x + l, y + k)\right)^2 \tag{3.5}$$

donde w determina el tamaño de la ventana de correlación, que afecta directamente los resultados. Una ventana pequeña producirá mapas ruidosos de disparidad, debido a ambigüedades en regiones homogéneas. Mientras que una ventana grande, resultará en bordes poco claros y en un aumento de costo computacional. Sin embargo, es más conveniente tener ventanas de tamaño fijo ya que se pueden utilizar técnicas de costo agregado, para hacer eficiente el cálculo de la medida de error para un gran número de pixeles.

La búsqueda exhaustiva que se describió, requiere el cálculo de D valores de $\Phi(x, y, d')$ por pixel. Para reducir el espacio de búsqueda proponemos utilizar un conjunto de candidatos de disparidad, los cuales se obtienen calculando la correlación por fase para cada renglón y encontrando la posición de los picos más altos de la función POC. Solo un subconjunto de todos los picos resultantes, con altura mayor a cero es considerado. Una vez que se tiene el conjunto de candidatos para un renglón dado, se realiza la búsqueda del mejor candidato para cada pixel de ese renglón. A continuación se presenta el algoritmo básico que se propone para la estimación de disparidad.

El método propuesto consta de tres etapas. La primera de ellas es el cálculo de la función POC por renglones entre el par de imágenes estéreo, la segunda es la búsqueda de los posibles valores de disparidad (candidatos) y por último el *matching* o proceso de elección del mejor candidato para cada pixel.

1. **Correlación por fase del par de imágenes estéreo.**

   Dado un par de imágenes estéreo $f(x, y)$ y $g(x, y)$, con $0 \leq x < nc$ y $0 \leq y < nr$, que representan la imagen izquierda y derecha, respectivamente. Se obtiene la correlación por fase $r_i(x)$ entre los renglones $f_i(x) = f(x, y)$ y $g_i(x) = g(x, y)$ con $0 < x < nc$ y y fija en i. Esto es:

   - Para i desde 0 hasta nr:

   $$r_i(x) = \frac{1}{N} \sum_{k=0}^{N-1} R_i(k) e^{-\frac{j2\pi xk}{N}} \tag{3.6}$$

   $$R_i(x) = \frac{F_i(k) G_i^*(k)}{|F_i(k) G_i^*(k)|} \tag{3.7}$$

   donde $R_i(k)$ es el espectro cruzado normalizado entre los renglones $f_i(x)$ y $g_i(x)$, $F_i(k)$ es la transformada discreta de Fourier del i-ésimo renglón y $G_i(k)^*$ es el complejo conjugado de la transformada discreta de Fourier, del renglón correspondiente $g_i(x)$.

2. **Búsqueda de candidatos.**

   Para cada $r_i(x)$ obtenido de la etapa anterior se forma un conjunto de nk candidatos, donde cada candidato es la posición de alguno de los máximos m más significativos de la función POC $r_i(x)$. Particularmente se escogen aquellos valores positivos de m y en las posiciones d_n con valor menor o igual al valor máximo de disparidad D. Nótese que debido a estas restricciones es posible encontrar solo n número de candidatos y no nk como se desea. Esto es:

   - Para y desde 0 hasta nr y mientras n sea menor o igual a nk:

     Se encuentra el máximo m más significativo de $r_i(x)$ y se localiza su posición d_n. Si $d_n <= D$ y $m > 0$ entonces se almacena d_n en el conjunto $candidatos[n]$, se vuelve cero su altura ($m = 0$) y se incrementa n.

3. **Elección del mejor candidato.**

   Para cada pixel (x, y), se encuentra el candidato d_n con el menor error dado como:

   $$d(x, y) = \underset{d_n = d_1, \ldots, d_n}{\arg\min} \{\Phi(x, y, d_n)\} \tag{3.8}$$

   donde n es el número de candidatos encontrados para el renglón y, $d(x, y)$ es el mapa de disparidad y $\Phi(x, y, d_c)$ es una función que mide la diferencia entre una vecindad de tamaño $2 * wsad \times 2 * wsad$ de $f(x, y)$ y una vecindad similar de $g(x - d_n, y)$. Note que la búsqueda se realiza solamente hasta n candidatos y no a través de todo el rango de disparidad. Esta minimización puede realizarse de la siguiente forma:

   I. Para cada $n = 1, \ldots, m$

      - Si $\Phi(x, y, d_n) < Error_{mnimo}(x, y)$,
        entonces $d(x, y) = d_n$ y $Error_{mnimo}(x, y) = \Phi(x, y, d_n)$.

Para elegir el valor de disparidad para cada pixel y formar el mapa disparidad $d(x, y)$ se utiliza una técnica de costo agregado, para hacer más eficiente el cálculo del error mínimo entre ventanas de tamaño $2 * wsad + 1$, donde la idea en general es calcular el error por columnas, de tal forma que no se necesite volver a calcular para cada pixel el error de toda la ventana completa, si no que sólo basta con sumar al error anterior ($E[wsad]$), el error de la columna que falta ($Ec[x + wsad]$) y restar el error de aquella columna que ya no forma parte de la ventana de correlación ($Ec[x - wsad - 1]$) para obtener el error total de la ventana de tamaño $2 * wsad + 1$. En la Figura 3.4 se hace un esquema para ilustrar lo mencionado anteriormente.

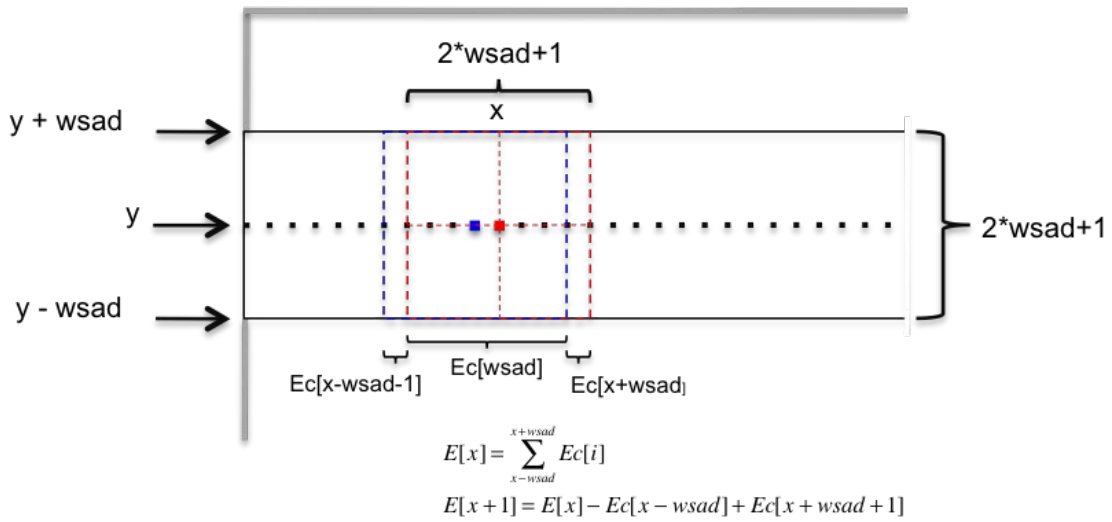


Figura 3.4. Esquema de la técnica de costo agregado

3.2. Optimización del método básico

3.2.1. Suavizado

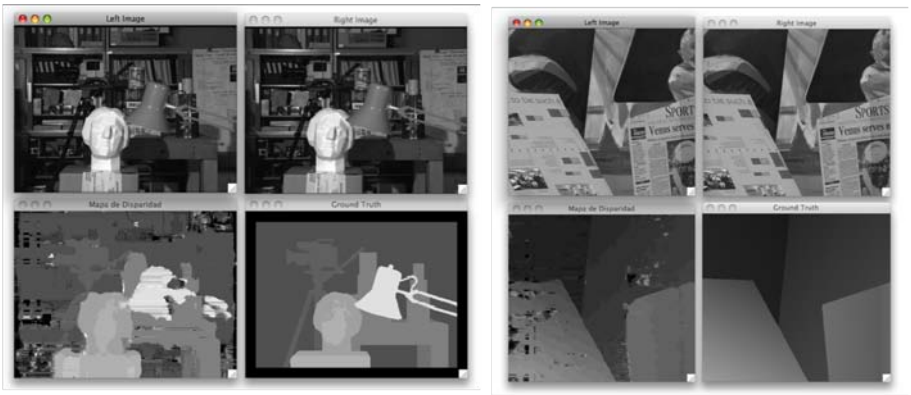
La reducción del espacio de búsqueda induce una cierta cuantización de los valores de disparidad. Bajo la restricción de continuidad (sección 1.1.3), esta cuantización introduce también cierto grado de regularización en el mapa de disparidad pero sólo en dirección horizontal. Debido a esto el algoritmo básico puede suprimir valores de disparidad en los bordes de los objetos, causando una especie de artefactos en forma de líneas horizontales en los mapas resultantes. Este problema puede observarse en la Figura 3.5, donde se muestran los resultados obtenidos con el método básico, utilizando como entrada el par de imágenes estéreo de Tsukuba [51] y Venus [51], junto con el mapa de disparidad ideal (Ground Truth). En la Figura 3.6, se hace un acercamiento al mapa de disparidad obtenido con el algoritmo básico, donde resaltan los artefactos a los que nos referimos. La presencia de estos, se debe a que en algunas ocasiones el valor de disparidad adecuado no se encuentra entre los candidatos obtenidos a partir de la POC. La solución al problema es introducir regularidad entre los conjuntos de candidatos de líneas adyacentes. Lo cual se puede lograr implementando un filtro Gaussiano con media cero y desviación estándar σ, después de

calcular la POC por renglones (paso No.3a del método básico). El filtro se aplica en dirección vertical (por columnas) centrado en el pixel y_i de la columna, donde $i = 0, ..., nr - 1$, con un kernel:

$$G(y_i, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{y_i^2}{2\sigma^2}} \tag{3.9}$$

Después del filtrado, se pueden encontrar las posiciones de los máximos de la función POC por renglones, que corresponden a los candidatos de disparidad, asegurando regularidad entre los conjuntos de candidatos entre líneas.

Cabe mencionar que se han obtenido buenos resultados al utilizar un filtro con desviación estándar en un intervalo de 5 a 20, algunas de las pruebas para determinar el valor adecuado de σ se muestran en la Figura 3.7. También se observa en la Figura 3.8 dos gráficas de la POC de 6 distintos renglones del mapa de disparidad obtenido del par de imágenes estéreo de Tsukuba; una aplicando el filtro con $\sigma = 10$ (figura 3.8(a)) y la otra sin él (Figura 3.8(b)). En la primera gráfica los valores de la función POC tienen cambios pronunciados mientras que en la segunda gráfica se aprecian valores muy parecidos de renglón a renglón, debido al filtro que suavizó la función POC. Debajo de cada gráfica se presenta el mapa de disparidad resultante utilizando el filtro (figura 3.8(c)) y sin utilizarlo (figura 3.8(d)). La implementación del filtro regulariza el conjunto de disparidades candidatas entre renglones vecinos, lo cual disminuye algunos de los artefactos observados.



(a) (b)

Figura 3.5. (a) Mapas de disparidad de Tsukuba y (b) Venus obtenidos con el método básico: (Imágenes superiores) Par estéreo y (Imágenes inferiores) Mapa de disparidad estimado e ideal

3.2.2. División de las imágenes completas en sub-imágenes

Anteriormente en el capítulo se mencionó la importancia de dividir imágenes grandes en sub-imágenes, con el fin de volver la función POC más robusta y lograr aumentar la probabilidad de que los picos más significativos (máximos) correspondan a los desplazamientos correctos. Con esta idea se decidió dividir las imágenes y formar sub-imagenes de tamaño W que posiblemente se traslapen δ pixeles. W debe de ser como mínimo el doble

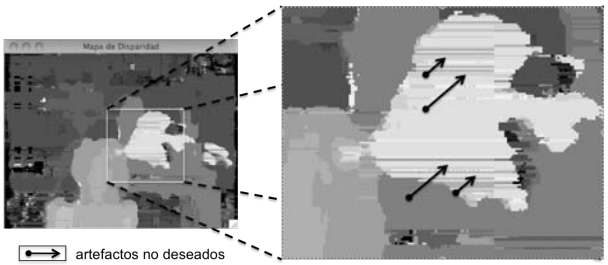


Figura 3.6. Acercamiento del artefacto causado por suprimir valores de disparidad

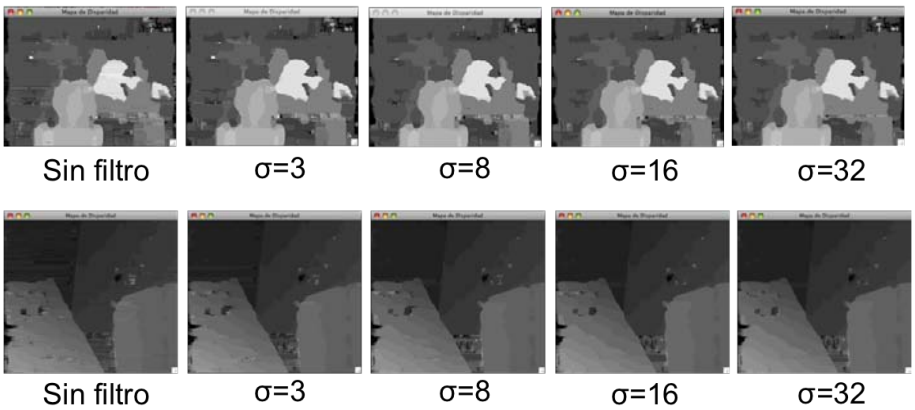
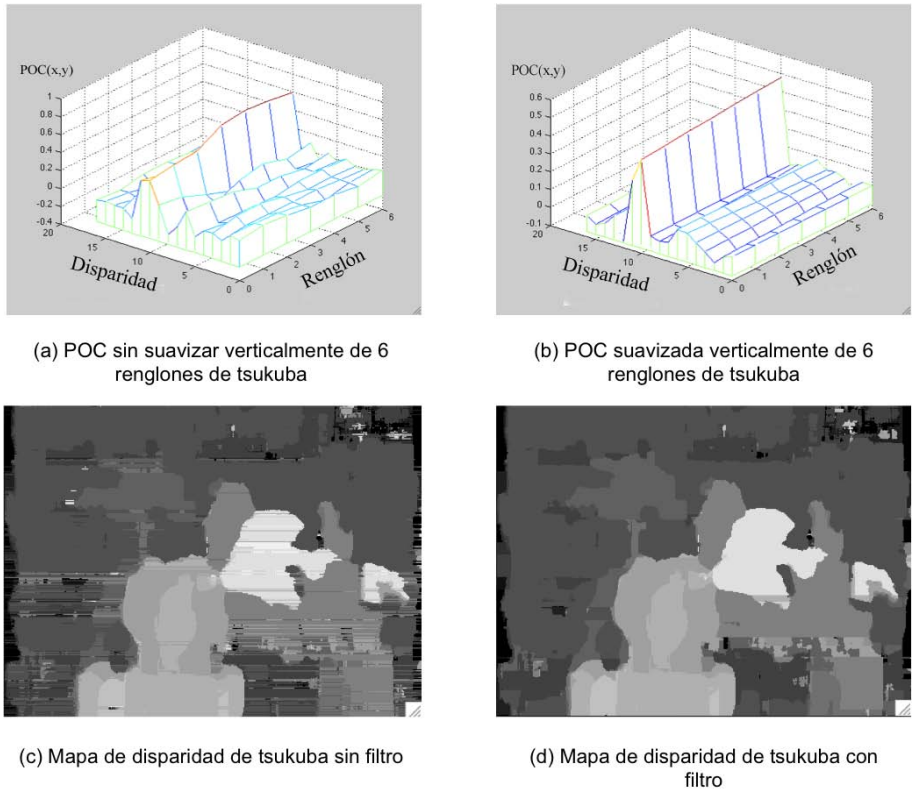


Figura 3.7. Filtrado con diferentes valores para la desviación estándar



(a) POC sin suavizar verticalmente de 6 renglones de tsukuba

(b) POC suavizada verticalmente de 6 renglones de tsukuba

(c) Mapa de disparidad de tsukuba sin filtro

(d) Mapa de disparidad de tsukuba con filtro

Figura 3.8. Función POC y Mapa de disparidad de Tsukuba con y sin filtro de suavizado

de la disparidad máxima y δ toma un valor de al menos $W/2$, de tal forma se asegura que puedan resultar valores en todo el rango de disparidad. Además el traslape entre las sub-imagenes ayuda a reducir la probabilidad de una mala estimación de candidatos debido al doble cálculo que se realiza. Al implementar esta propuesta debe considerarse el compromiso entre velocidad y precisión debido a que es inmediato pensar que esta propuesta aumenta el costo computacional, pero reduce el porcentaje de error. En el capítulo siguiente se presentaran a detalle resultados aplicando el método básico, con las imágenes completas y considerando las divisiones de las imágenes, que posteriormente se discutirán en el último capítulo.

Capítulo 4

Resultados

En este capítulo se expone una serie de experimentos que permiten evaluar el desempeño y calidad de ambas estrategias presentadas en el capítulo tres para la estimación de disparidad. Se utilizaron imágenes de entrada en escala de grises, tomando como referencia la imagen izquierda. La implementación de los algoritmos propuestos se realizó usando OpenCV versión 2.2 [53]. Las pruebas se realizaron en un procesador 2.53 Ghz Intel Core2Duo con 4 Gb en RAM y sistema operativo MacOS X versión 10.5.8

Se midió el desempeño del algoritmo por medio del conjunto de imágenes utilizado por Scharstein y Szeliski [26]. Se utilizó este conjunto de imágenes conocido como banco de prueba Middlebury debido a que la comunidad las ha adoptado como un estándar para la comparación y evaluación de los algoritmos estéreo. El banco de pruebas incluye la imagen izquierda, derecha y mapa de disparidad ideal así como la máxima disparidad de cada escena.

La evaluación de precisión de nuestro algoritmo se tomó como medida el porcentaje de pixeles correspondientes incorrectos, propuesta también por Scharstein y Szeliski en [26], está se obtiene al comparar el mapa de disparidad resultante contra el mapa de disparidad ideal (conocido como Ground Truth, GT), omitiendo aquellos valores iguales a cero del GT que representan disparidades desconocidas. De esta manera, se define el porcentaje de pixeles correspondientes incorrectos (Bad Matching Percentage) entre el mapa de disparidad resultante del algoritmo propuesto $d_{AP}(x, y)$ y el Ground Truth $d_{GT}(x, y)$,

$$BMP = \frac{1}{N} \sum_{(x,y)} (|d_{AP}(x, y) - d_{GT}(x, y)| > \delta_d), \tag{4.1}$$

donde δ_d es la tolerancia de error. Para nuestras pruebas $\delta_d = 1.0$ que coincide con publicaciones como [54, 55, 56, 57].

Primero se presentarán resultados para cada par de imágenes estéreo utilizando la imagen completa (método básico), con y sin la etapa de suavizado, para posteriormente mostrar los resultados con las divisiones de las imágenes de igual forma con y sin la etapa de suavizado. Se presentan los parámetros que obtienen el mejor desempeño en términos de precisión. También, introducimos los parámetros que nos representan el mejor compromiso

entre tiempo de procesamiento y precisión. Finalmente con base a los resultados anteriores se darán los parámetros óptimos del algoritmo propuesto y se adjuntará la comparación con los algoritmos actuales por medio de la plataforma en línea de Middlebury.

Las pruebas que se realizaron para cada par de imágenes estéreo fueron para obtener el número de candidatos y el tamaño de ventana de correlación que lograrán mantener un equilibrio entre la velocidad con la que se estima el mapa de disparidad y su precisión. Se obtuvieron las medidas de precisión y tiempo de ejecución variando los siguientes parámetros del algoritmo: número de candidatos, tamaño de la ventana de medida de similitud. El número de candidatos se varia de 1 hasta la disparidad máxima, mientras que el tamaño de la ventana se vario de $3 \times 3, \ldots, 17 \times 17$. Posteriormente para la etapa de suavizado se probaron valores de 1 hasta 35 para determinar la mejor varianza σ del filtro gaussiano. Los resultados para las escenas de tsukuba, venus, teddy y conos se presentan a continuación.

4.1. Escena Tsukuba

El par estéreo conocido como Tsukuba fue proporcionado por Y. Obta y N. Nakamura de la Universidad de Tsukuba [54]. Este par estéreo también es conocido como *head and lamp*. El par estéreo es de tamaño 384×288. Tsukuba tiene una disparidad máxima de 16 pixeles y tiene la característica que todos sus desplazamientos son valores enteros. Es uno de los pares estéreo más sencillos y populares en el estado del arte para llevar acabo experimentos.

El mínimo BMP que se obtuvo para esta escena es de 11.19 % con 15 candidatos y una ventana de correlación de 15×15 en un tiempo de $315.94ms$. El mapa estimado con los parámetros anteriores se muestra en la figura 4.1 junto con su respectivo GT y el par de imágenes estéreo.

Analizando los resultados de precisión y tiempo para el método básico utilizando como entrada el par de imágenes de Tsukuba, se obtuvieron las gráficas que se muestran en la figura 4.2. A partir de la gráfica 4.2 (a), donde se expone el BMP contra el número promedio de candidatos, se puede observar que al aumentar el número de candidatos el porcentaje de error disminuye, lo cual es de esperarse debido a que entre más grande sea el conjunto de candidatos, las opciones de posibles valores de disparidad para cada pixel aumentan y así también la probabilidad de estimar el desplazamiento correcto para cada pixel. Es importante resaltar que el número promedio de candidatos no es el número de candidatos que se da como parámetro de entrada al algoritmo, si no es el promedio de candidatos que se necesitó por renglón para estimar el mapa de disparidad. Para este par de imágenes con 15 candidatos como parámetro de entrada en el algoritmo se obtuvo el mínimo BMP, el cual no refleja una reducción importante del espacio de búsqueda, recordando que el mayor desplazamiento que se puede encontrar en la escena es de 16. Recordemos que el parámetro de entrada referente al número de candidatos nos permite fijar solo el limite superior. Sin embargo, el número de candidatos real puede ser menor al indicado. Por lo cual se midió la cantidad de candidatos tomados en cada iteración para encontrar el promedio. En este caso el número de candidatos promedio es de 10 dando como resultado una reducción del espacio de búsqueda de aproximadamente 38 %. Por otro lado, en la gráfica de la figura

4.2 (c) se observa un aumento del tiempo respecto al número de candidatos promedio, lo cual es lógico por el aumento de computo al tener que estimar el candidato que obtenga el error mínimo por pixel. Ahora bien, el tamaño de la ventana de correlación que se utiliza para la estimación del error mínimo por pixel es otro parámetro que debe ser considerado. En la gráfica de la figura 4.2 (b) se observa como el BMP es alto para ventanas de tamaño pequeño y disminuye conforme el tamaño de la ventana aumenta, esto debido a que la escena de Tsukuba presenta pocos objetos (ver figura 4.1) para los que una ventana de tamaño entre 11 y 16 es suficiente, observe que el porcentaje de error tiende a estabilizarse entre los mismos valores. El tiempo como es de esperarse debido al aumento de computo por ventanas más grandes aumenta proporcionalmente al tamaño de la ventana de correlación.

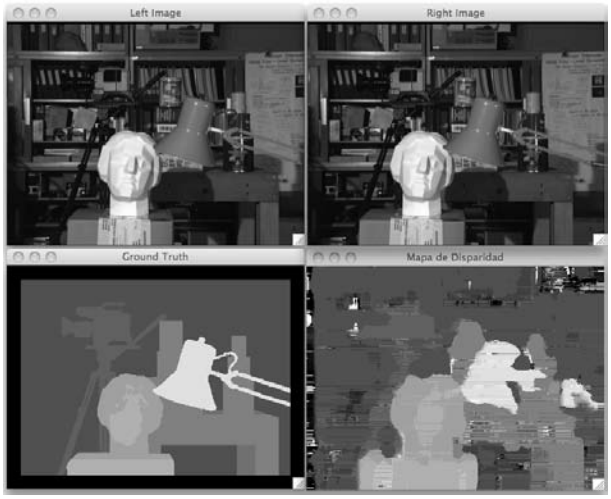


Figura 4.1. Par de imágenes estéreo, Ground Truth y mapa de disparidad con el mínimo BMP estimado con el método básico para la escena de Tsukuba.

4.1.1. Etapa de suavizado

En la figura 4.3 se muestran las gráficas de la variación del tiempo de cómputo (b) y el BMP (a) respecto al aumento de los valores de sigma. El BMP del mapa de disparidad estimado después de la etapa de filtrado con una varianza de 26 es de 9.57 % con un tiempo de $434.2ms$, lo que reduce el error en 1.62 % respecto al mapa estimado sin el filtrado pero aumenta en $118.26ms$ el tiempo.

4.1.2. Sub - imágenes con y sin etapa de suavizado

Los resultados de precisión y tiempo de procesamiento para el método básico aplicando la propuesta de formar sub-imágenes se pueden observar en la tabla de la figura 4.3, donde Tsukuba obtiene un $BMP = 13.19\,\%$ con una ventana de correlación de 17×17, 13 candidatos, un ancho de sub-imágenes de 64 pixeles con un traslape entre ellas de 64 pixeles y un tiempo de ejecución de $374.1ms$. Haciendo una comparación de estos resultados con los obtenidos del método básico (ver figura 4.1) resulta que el BMP aumenta en un 2 %

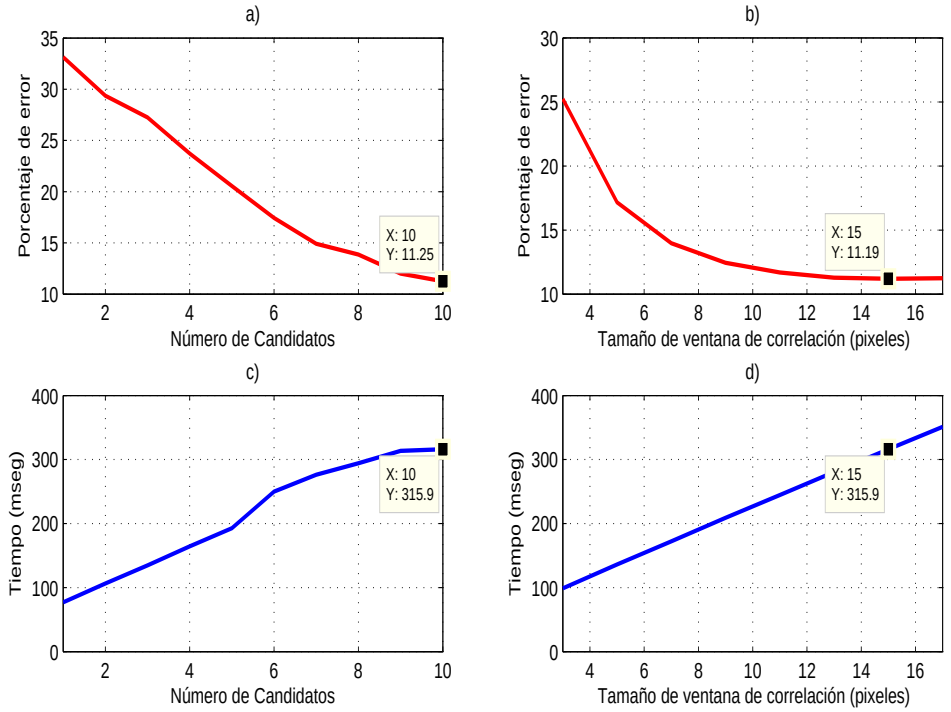


Figura 4.2. Desempeño del algoritmo propuesto para la escena Tsukuba: (a) Porcentaje de error vs. No. promedio de candidatos, (b) Porcentaje de error vs. Tamaño de ventana de correlación, (c) Tiempo vs. No. promedio de candidatos, (d) Tiempo vs. Tamaño de ventana de correlación.

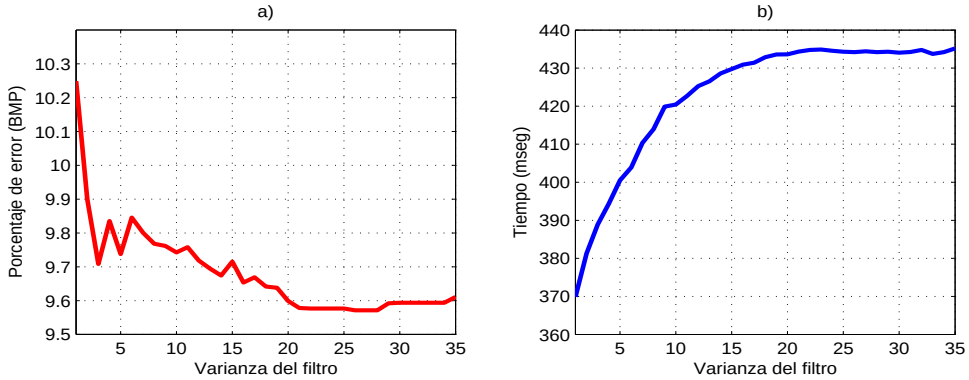


Figura 4.3. Desempeño del algoritmo propuesto con respecto de la etapa de filtrado para la escena Tsukuba: (a) Porcentaje de error vs. Varianza del filtro, (b) Tiempo vs. Varianza del filtro.

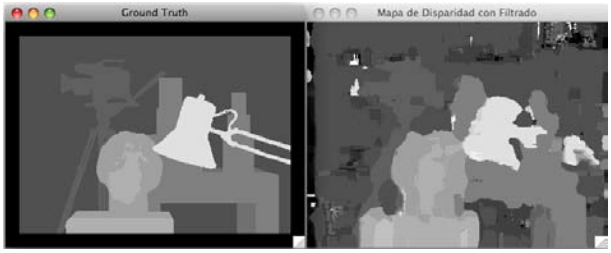


Figura 4.4. Mapa estimado con el mínimo BMP después de la etapa de filtrado para la escena Tsukuba.

mientras que el tiempo aumenta 58.16ms por lo que para Tsukuba no es una opción esta propuesta de formar sub-imágenes sin el filtrado.

Ahora bien por último en la figura 4.5 se puede observar como el BMP disminuye conforme se aumenta la varianza de la etapa de filtrado. El BMP se logra reducir a un 11.01 % el cual es menor al obtenido con el método básico pero mayor al obtenido aplicando el filtrado en un 1.44 % y registra un tiempo de 433.71ms, que es menor al tiempo registrado con el método básico con la etapa de filtrado por 407.59ms.

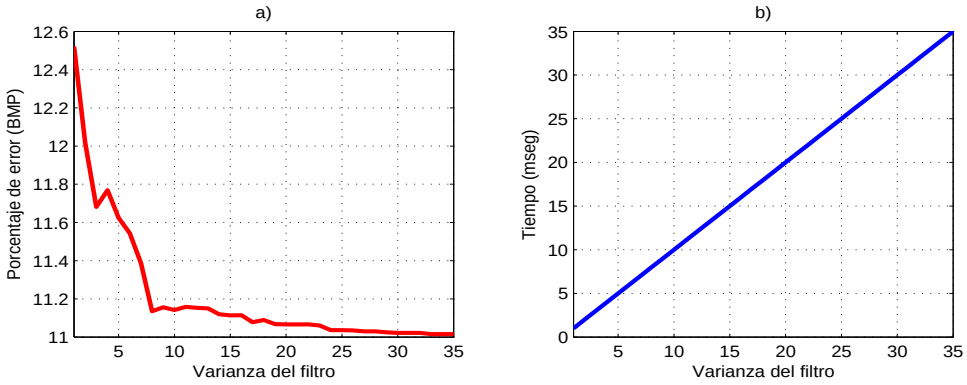


Figura 4.5. Desempeño del algoritmo propuesto considerando sub-imágenes y la etapa de filtrado para la escena Tsukuba: (a) Porcentaje de error vs. Varianza del filtro, (b) Tiempo vs. Varianza del filtro.

4.2. Escena Venus

Par de imágenes estéreo de tamaño 434×383, son vistas de escenas planas y son parte de una secuencia de nueve imágenes creadas por Daniel Scharstein, Padma Ugbabe y Rick Szeliski [54]. El GT tiene una precisión de un octavo de pixel y su disparidad máxima es de 20 pixeles. El mapa estimado con los parámetros anteriores se muestra en la figura 4.6 junto con su respectivo GT y el par de imágenes estéreo.

El mínimo BMP que se obtuvo para esta escena es de 11.33 % con 16 candidatos y una ventana de correlación de 17×17 en un tiempo de 694.16ms. El número promedio de candidatos utilizado por renglón para estimar el mapa de disparidad es de 12 reduciendo el

espacio de búsqueda aproximadamente en un 40 %. Los resultados obtenidos y expuestos en las gráficas de la figura 4.7 son muy parecidos a los de Tsukuba debido a que ambas imágenes representan escenas sencillas por el pequeño rango de disparidad. Además Venus no tiene objetos con volumen, como la cabeza en Tsukuba, donde el rostro tiene distintos planos de profundidad (nariz y ojos), lo cual aumenta la probabilidad de una correcta estimación de los valores de disparidad.

La gráfica 4.7 (a) muestra la relación entre el número de candidatos promedio y el BMP, que disminuye conforme aumentan los candidatos por la misma razón que se mencionó para la escena anterior. En la gráfica 4.7 (c) se puede observar el aumento de tiempo conforme aumenta el número de candidatos promedio, de esta gráfica y de la misma en Tsukuba es interesante resaltar el cambio que tiene el tiempo de procesamiento. Para Venus el punto donde se observa un cambio en la pendiente es en 8 candidatos, mientras que en Tsukuba sucede en 5. Una explicación para este cambio puede ser que al momento de graficar el número de candidatos promedio se deja fijo el valor de la ventana de correlación con el valor que corresponde al BMP mínimo, en este caso 17, de tal forma que para este valor se repite el número de candidatos promedio, por lo que se tiene que realizar un promedio de los tiempos para estas combinaciones repetidas, resultando en un aumento de tiempo.

En las gráficas 4.7 (b) y (c) se aprecian las mismas relaciones que se explicaron en la escena anterior, entre el tamaño de ventana de correlación contra el BMP y el tiempo de procesamiento respectivamente.

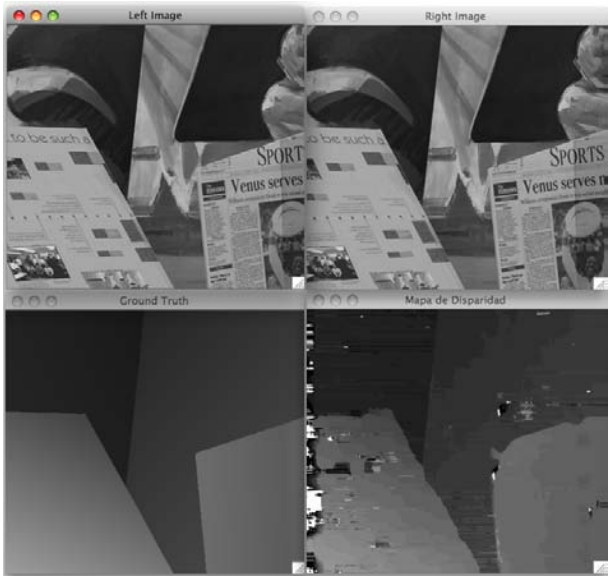


Figura 4.6. Par de imágenes estéreo, Ground Truth y mapa de disparidad con el mínimo BMP estimado con el método básico para la escena Venus.

4.2.1. Etapa de suavizado

En la figura 4.8 se muestran las gráficas de la variación del tiempo de cómputo (b) y el BMP (a) respecto al aumento de los valores de varianza en el filtro gausiano.

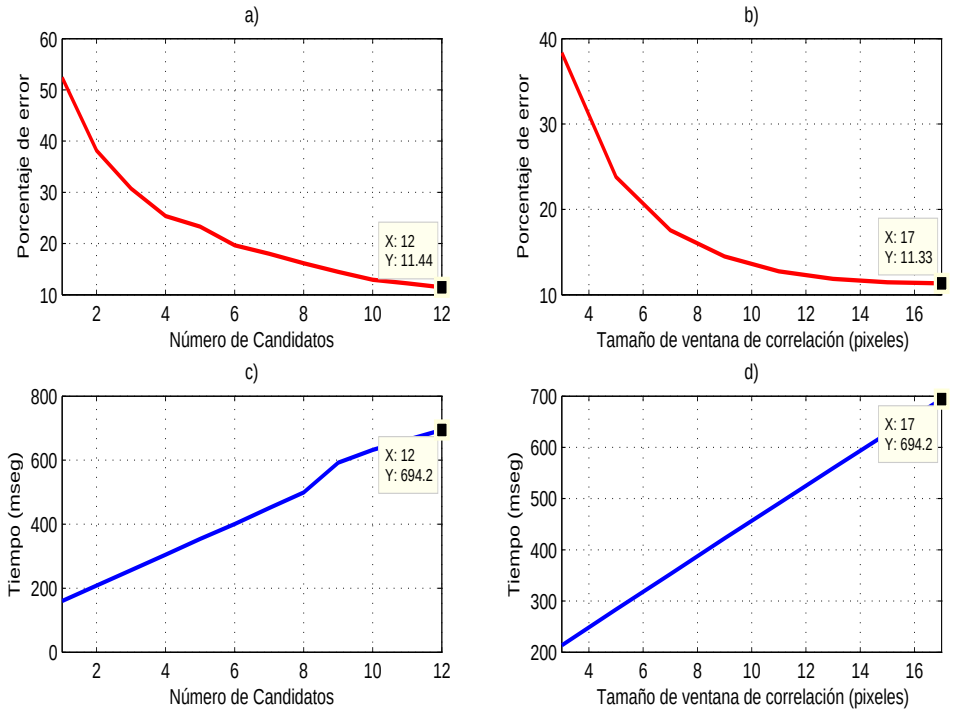


Figura 4.7. Gráficas comparativas de desempeño del algoritmo propuesto para la escena Venus: (a) Porcentaje de error vs. No. promedio de candidatos, (b) Porcentaje de error vs. Tamaño de ventana de correlación, (c) Tiempo vs. No. promedio de candidatos, (d) Tiempo vs. Tamaño de ventana de correlación.

El BMP del mapa de disparidad estimado después de la etapa de filtrado con una varianza de 9 es de 8.53 % con un tiempo de 841.3ms, lo que reduce el error en 2.8 % respecto al mapa estimado sin el filtrado pero aumenta en 147.14ms el tiempo.

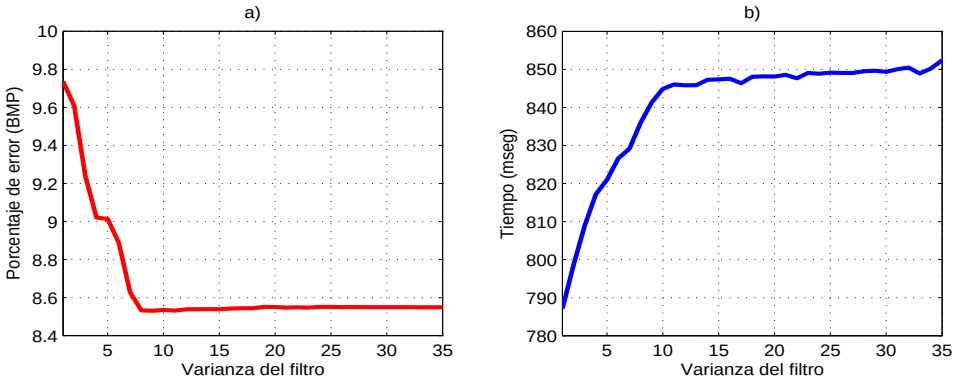


Figura 4.8. Desempeño del algoritmo propuesto con respecto de la etapa de filtrado para la escena Venus: (a) Porcentaje de error vs. Varianza del filtro, (b) Tiempo vs. Varianza del filtro.

Figura 4.9. Mapa estimado con el mínimo BMP después de la etapa de filtrado para la escena Venus

4.2.2. Sub - imágenes con y sin etapa de suavizado

Los resultados de precisión y tiempo para el método básico aplicando la propuesta de formar sub-imágenes se pueden observar en la tabla de la figura 4.3, donde Venus obtiene un $BMP = 13.43\,\%$ con una ventana de correlación de 17×17, 16 candidatos, un ancho de sub-imágenes de 64 pixeles con un traslape entre ellas de 40 pixeles y un tiempo de ejecución de 1075ms. Haciendo una comparación de estos resultados con los obtenidos por el método básico (ver figura 4.1) observamos un efecto negativo en la precisión dado que el BMP aumenta en 2.1 % mientras que el tiempo de ejecución aumenta en 380.84ms representando un aumento del 35.42 %

Ahora bien por último en la figura 4.10 se puede observar como el BMP aumenta conforme el valor de la varianza en el filtro de suavizado también aumenta. El BMP se logra reducir a un 10.87 %, el cual es menor al obtenido con el método básico pero mayor al obtenido aplicando el filtrado en un 2.34 %, además registra un tiempo de 1237ms que es mayor al tiempo registrado con el método básico con la etapa de filtrado por 395.7ms.

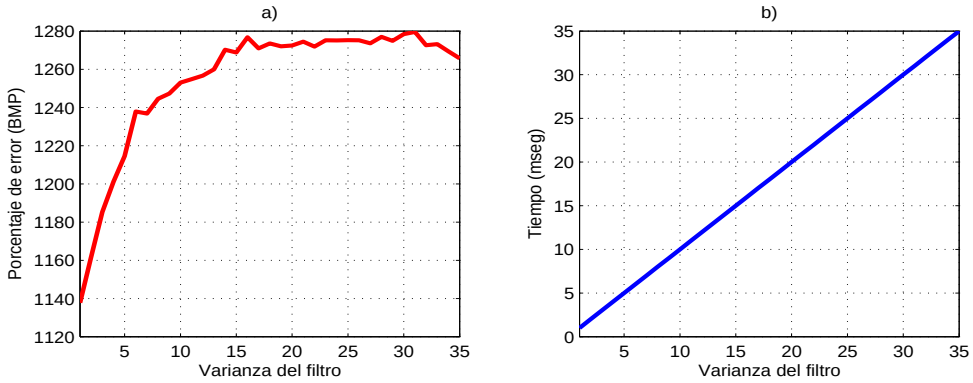


Figura 4.10. Desempeño del algoritmo propuesto considerando sub-imágenes y una etapa de filtrado para la escena Venus: (a) Porcentaje de error vs. Varianza del filtro, (b) Tiempo vs. Varianza del filtro.

4.3. Escena Conos

El par estéreo conocido como conos fue creado por Daniel Scharstein, Alexander Vandenberg Rodes y Rick Szeliski y esta compuesta por una secuencia de nueve imágenes de 450×375. Estas imágenes fueron creadas usando una técnica de iluminación estructurada y están rectificadas para mantener desplazamientos horizontales únicamente. El GT tiene una precisión de un cuarto de pixel por lo que el rango de disparidades es de 0.25 a 63.75. Debido a que nuestro algoritmo no es de precisión sub-pixel la disparidad máxima que se toma es de 60 pixeles.

El mínimo BMP que se obtuvo para esta escena es de 40.23 % con 16 candidatos y una ventana de correlación de 3×3 en un tiempo de $57.75ms$. El número promedio de candidatos utilizado por renglón para estimar el mapa de disparidad es de 1, reduciendo el espacio de búsqueda aproximadamente en un 98 %. El mapa estimado con los parámetros anteriores se muestra en la figura 4.11 junto con su respectivo GT y el par de imágenes estéreo.

Este par de imágenes es interesante y complicada por la cantidad de objetos presentes en la escena y la cantidad de planos de profundidad que presenta. A diferencia de Venus y Tsukuba donde los tamaños de ventana de correlación grandes obtienen mejores resultados que los pequeños, en la gráfica 4.12 (b) se puede apreciar que conforme aumentamos la ventana de correlación el BMP aumenta lo que tiene lógica debido a la cantidad de objetos presentes en la escena. Ventanas pequeñas proporcionan una mejor aproximación debido a que la vecindad de pixeles con la que se correlaciona es más pequeña haciendo posible una mejor estimación cuando se presentan discontinuidades en los valores de disparidad. En la figura 4.13 (b) donde se muestra el mapa estimado con una ventana de 3×3 y 27 candidatos, se puede observar como el mapa no es muy claro pero los bordes están definidos a diferencia del que se muestra en la figura 4.13 (c), a pesar de que el BMP es más bajo en el mapa estimado con la ventana de 31, esto ocurre por que al aumentar el tamaño de ventana se pierde precisión en la estimación de los bordes, pero se gana en el resto de los pixeles de la escena.

Para las gráficas de tiempo de procesamiento contra número de candidatos promedio 4.12

(c) y tamaño de ventana de correlación 4.12 (d) es lógico el aumento del tiempo conforme aumentan ambas variables, por lo mismo mencionado en la escena de Tsukuba.

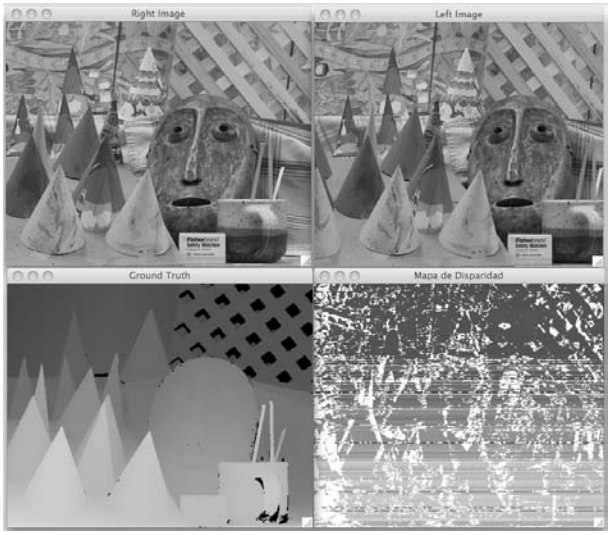


Figura 4.11. Par de imágenes estéreo, Ground Truth y mapa de disparidad con el mínimo BMP estimado con el método básico para la escena Conos.

4.3.1. Etapa de suavizado

El BMP del mapa de disparidad estimado después de la etapa de filtrado con una varianza de 7 es de 35.31 % con un tiempo de $64.94ms$, lo que reduce el error en 4.92 % respecto al mapa estimado sin el filtrado pero aumenta en $7.19ms$ el tiempo de procesamiento.

En la figura 4.14 se muestran las gráficas de la variación del tiempo de cómputo (b) y el BMP (a) respecto al aumento de los valores de varianza en el filtro gausiano.

4.3.2. Sub - imágenes con y sin etapa de suavizado

Los resultados de precisión y tiempo de cómputo para el método básico aplicando la propuesta de formar sub-imágenes se pueden observar en la tabla de la figura 4.3, donde Conos obtiene un $BMP = 58.82\,\%$ con una ventana de correlación de 17×17, 4 candidatos, un ancho de sub-imágenes de 128 pixeles con un traslape entre ellas de 60 pixeles y un tiempo de ejecución de $478.11ms$. Haciendo una comparación de estos resultados con los obtenidos a partir del método básico (ver figura 4.1) el BMP aumenta en 18.59 % y el tiempo también se incrementa en $420.36ms$ correspondiente a un aumento del 87.92 %. Volvemos a encontrar un efecto negativo en el uso de sub-imágenes

Ahora bien por último en la figura 4.16, se puede observar como el BMP no presenta un comportamiento uniforme conforme la varianza en el filtrado aumenta, a pesar de eso podemos localizar fácilmente el valor con el mínimo BMP de manera sencilla por la forma abrupta como decae el BMP para una varianza de 11. El BMP no se logra reducir pues presenta un 52.25 %, el cual es mayor al obtenido con el método básico y al obtenido

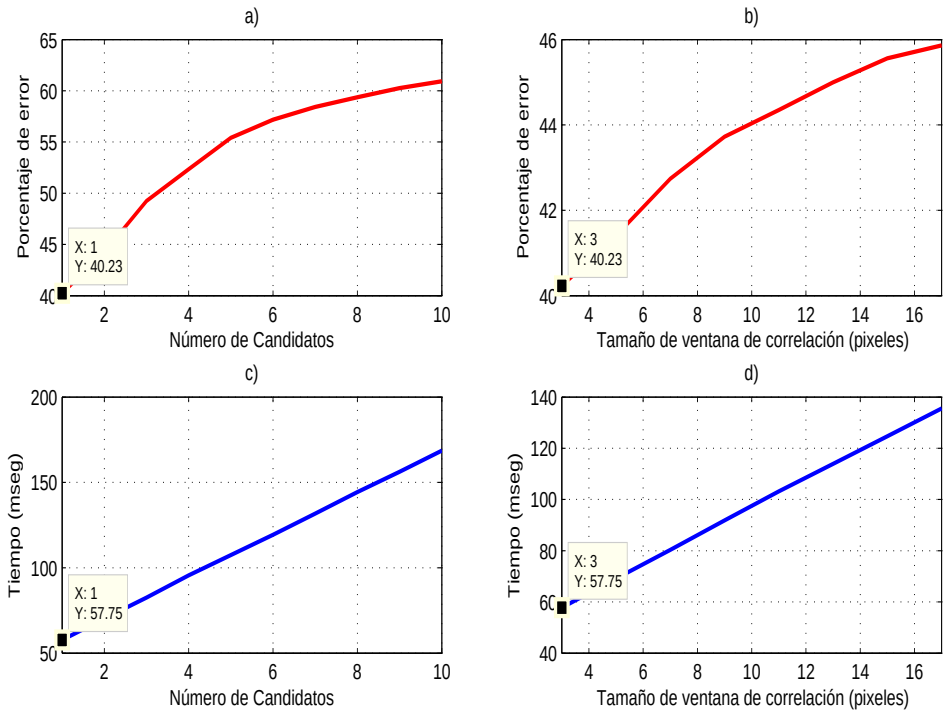


Figura 4.12. Gráficas comparativas de desempeño del algoritmo propuesto para la escena Conos: (a) Porcentaje de error vs. No. promedio de candidatos, (b) Porcentaje de error vs. Tamaño de ventana de correlación, (c) Tiempo vs. No. promedio de candidatos, (d) Tiempo vs. Tamaño de ventana de correlación.

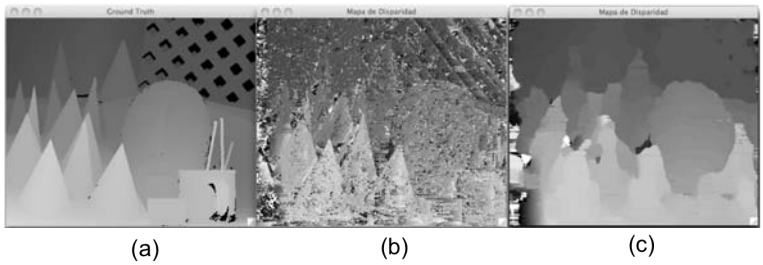


Figura 4.13. Efecto del tamaño de ventana de correlación para la escena Conos: (a) Ground Truth, (b) Ventana de tamaño 3×3 BMP= 54.81 %, (c) Ventana de tamaño 3×3 BMP= 42.32 %.

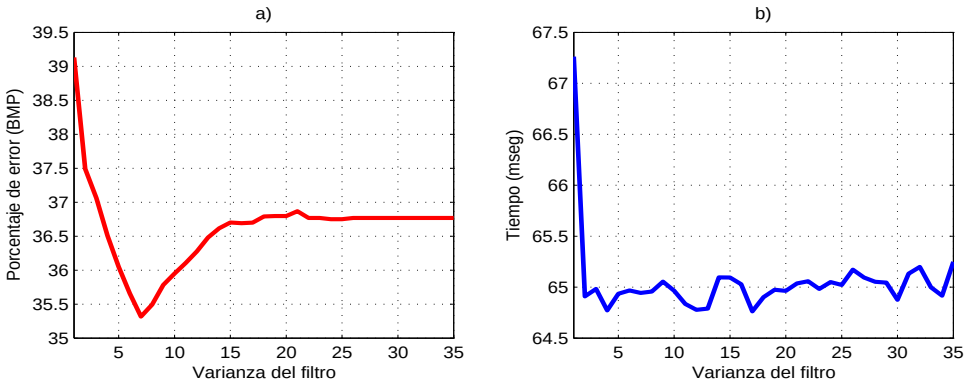


Figura 4.14. Desempeño del algoritmo propuesto con respecto de la etapa de filtrado para la escena Conos: (a) Porcentaje de error vs. Varianza del filtro, (b) Tiempo vs. Varianza del filtro.

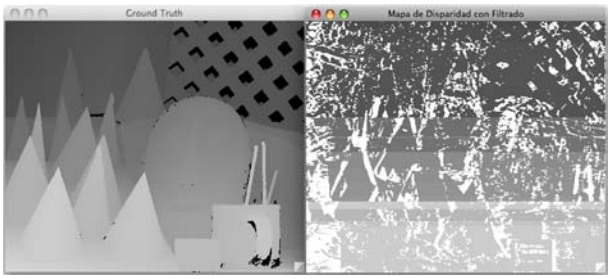


Figura 4.15. Mapa estimado con el mínimo BMP después de la etapa de filtrado para la escena Conos.

aplicando el filtrado en un 16.94 %, registra un tiempo de 596.17ms que es mayor al tiempo registrado con el método básico con la etapa de filtrado por 531.23ms.

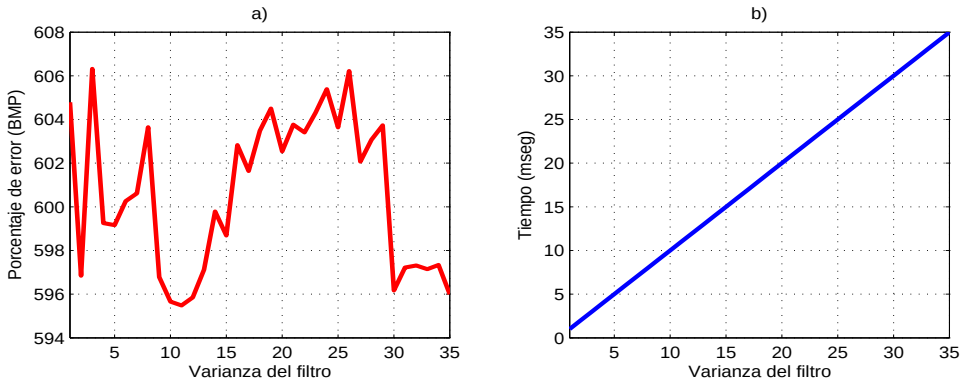


Figura 4.16. Desempeño del algoritmo propuesto considerando sub-imágenes y la etapa de filtrado para la escena Conos: (a) Porcentaje de error vs. Varianza del filtro, (b) Tiempo vs. Varianza del filtro.

4.4. Escena Teddy

Teddy al igual que conos fue creada por Daniel Scharstein, Alexander Vandenberg Rodes y Rick Szeliski y esta compuesta por una secuencia de imágenes de 450×375. Ambas secuencias fueron creadas con las mismas especificaciones. El mínimo BMP que se obtuvo para esta escena es de 38.64 % con 39 candidatos y una ventana de correlación de 15×15 en un tiempo de 1602ms. El número promedio de candidatos utilizado por renglón para estimar el mapa de disparidad es de 34, reduciendo el espacio de búsqueda aproximadamente en un 43.33 %. El mapa estimado con los parámetros anteriores se muestra en la figura 4.17 junto con su respectivo GT y el par de imágenes estéreo.

Debido a que Teddy fue creada con las mismas características que Conos, podemos ver el mismo comportamiento que Conos tuvo en nuestras gráficas de la figura 4.18.

El tiempo en que se logra estimar el mapa de disparidad es alto debido al tamaño de la imagen y el amplio rango de disparidad que tiene la escena.

4.4.1. Etapa de suavizado

En la figura 4.19 se muestran las gráficas de la variación del tiempo de procesamiento. (b) y el BMP (a) respecto al aumento de la varianza del filtro gausiano.

El BMP del mapa de disparidad estimado después de la etapa de filtrado con una varianza de 16 es de 36.65 % con un tiempo de 1787.6ms, lo que reduce el error en 1.99 % respecto al mapa estimado sin el filtrado pero aumenta en 185.6ms el tiempo.

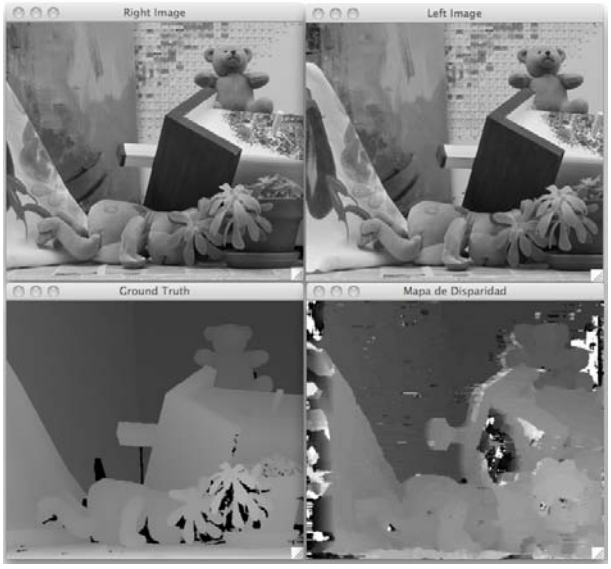


Figura 4.17. Par de imágenes estéreo, Ground Truth y mapa de disparidad con el mínimo BMP estimado con el método básico para la escena Teddy.

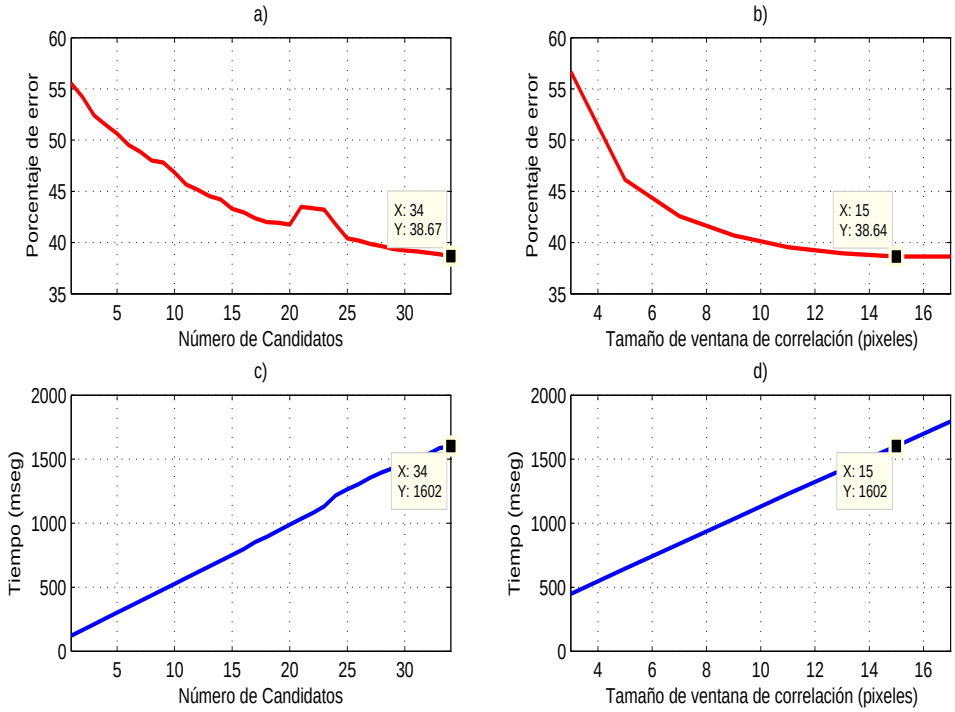


Figura 4.18. Gráficas comparativas de desempeño del algoritmo propuesto para la escena Teddy: (a) Porcentaje de error vs. No. promedio de candidatos, (b) Porcentaje de error vs. Tamaño de ventana de correlación, (c) Tiempo vs. No. promedio de candidatos, (d) Tiempo vs. Tamaño de ventana de correlación.

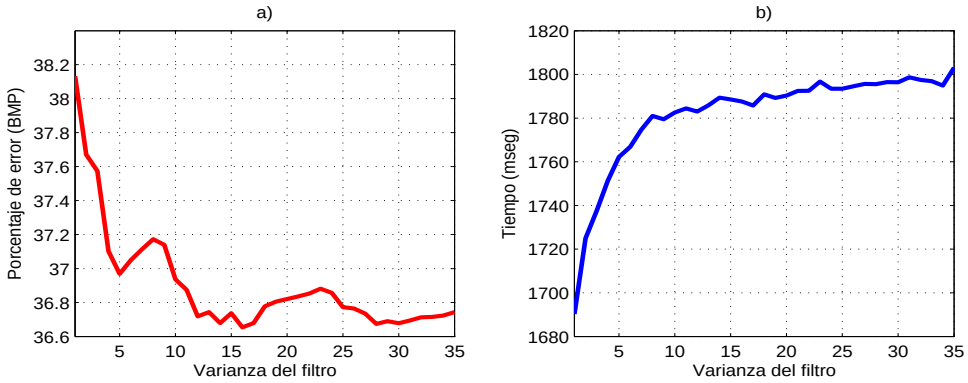


Figura 4.19. Desempeño del algoritmo propuesto con respecto de la etapa de filtrado para la escena Teddy: (a) Porcentaje de error vs. Varianza del filtro, (b) Tiempo vs. Varianza del filtro.

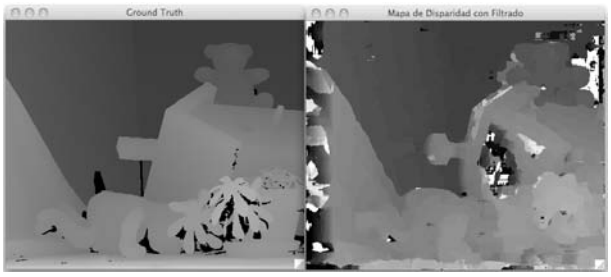


Figura 4.20. Mapa estimado con el mínimo BMP después de la etapa de filtrado para la escena Teddy.

4.4.2. Sub - imágenes con y sin etapa de suavizado

Los resultados de precisión y tiempo para el método básico aplicando la propuesta de formar sub-imágenes se pueden observar en la tabla de la figura 4.3, donde Teddy obtiene un $BMP = 55.26\,\%$ con una ventana de correlación de 17×17, 34 candidatos, un ancho de sub-imágenes de 128 pixeles con un traslape entre ellas de 60 pixeles y un tiempo de ejecución de $3466ms$. Haciendo una comparación de estos resultados con los obtenidos del método básico (ver figura 4.1) volvemos a obtener un aumento del BMP de $16.62\,\%$ y un aumento en el tiempo de ejecución de $1864ms$ correspondiente a $53.77\,\%$.

Ahora bien por último en la figura 4.21 se puede observar como el BMP aumenta conforme el valor de la varianza del filtrado también aumenta de 1a 25 unidades, mientras que este mismo valor decae para varianzas entre 26 a 35 unidades. No es necesario realizar más pruebas esperando que el BMP disminuya aún más puesto que la gráfica de tiempo vs. el valor de varianza tiene un crecimiento lineal. El BMP no se logra reducir pues presenta un aumento de $16.92\,\%$ respecto al obtenido con el método básico aplicando el filtrado, registra un tiempo de $3595ms$ que es mayor al tiempo registrado con el método básico con la etapa de filtrado por $1807.4ms$.

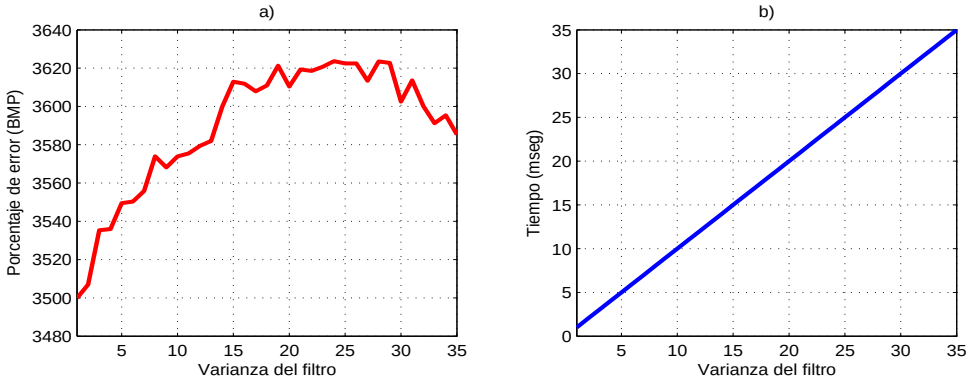


Figura 4.21. Desempeño del algoritmo propuesto considerando sub-imágenes y una etapa de filtrado para la escena Teddy: (a) Porcentaje de error vs. Varianza del filtro, (b) Tiempo vs. Varianza del filtro.

La tabla 4.1 muestra los parámetros que nos permitieron alcanzar los mejores valores de precisión para los cuatro pares estéreo. Para imágenes con desplazamientos enteros se alcanzo un BMP de $11.2\,\%$ mientras que para imágenes con desplazamientos fraccionarios se obtuvo un BMP de $39\,\%$. Los tiempos de ejecución varían con respecto a los parámetros de entrada.

Sin embargo, para obtener un balance entre exactitud y velocidad se encontró la intersección entre las gráficas del tiempo y el BMP, de tal manera se obtiene el valor del tamaño de ventana de correlación y el número de candidatos. Dichas gráficas se muestran en las figuras 4.22, 4.23, 4.24, 4.25 y los valores obtenidos se muestran en la tabla 4.2.

Los resultados para sub-imágenes se muestran en la tabla 4.3. Se puede notar que esta variante del método no alcanza los valores de BMP que el método básico además de incurrir en una penalidad superior al $50\,\%$ en tiempo de ejecución.

Método básico - Parámetros óptimos

Escena	Tamaño de ventana de correlación (pixeles)	Número de candidatos	Número de candidatos promedio	Tiempo promedio (mseg)	BMP
Tsukuba	15 x 15	15	10	315.94	11.19
Venus	17 x 17	16	14	694.16	11.33
Conos	3 x 3	1	1	57.75	40.23
Teddy	15 x 15	39	34	1602	38.64

Combinación promedio óptima para el algoritmo con el método básico:

17 x 17	15	——	685.33	28.28

Tabla 4.1. Tabla de parámetros óptimos del método básico.

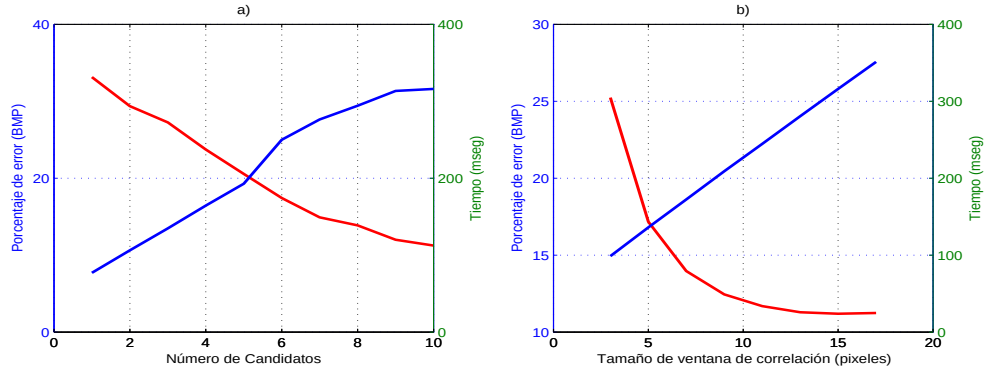


Figura 4.22. Balance entre exactitud y velocidad para la escena Tsukuba: (a) Balance para obtener el número de candidatos, (b) Balance para obtener el tamaño de ventana de correlación.

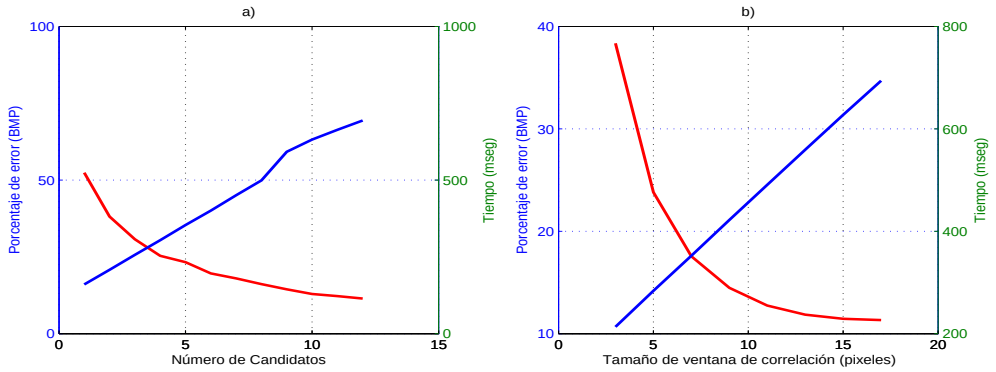


Figura 4.23. Balance entre exactitud y velocidad para la escena Venus: (a) Balance para obtener el número de candidatos, (b) Balance para obtener el tamaño de ventana de correlación.

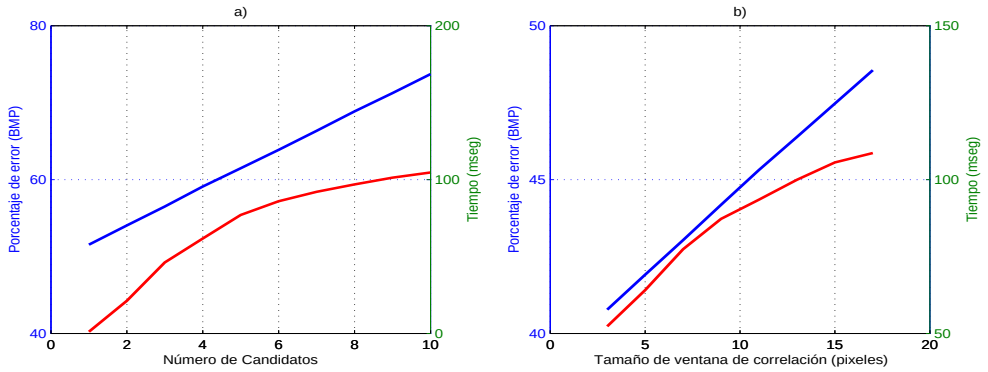


Figura 4.24. Balance entre exactitud y velocidad para la escena Conos: (a) Balance para obtener el número de candidatos, (b) Balance para obtener el tamaño de ventana de correlación.

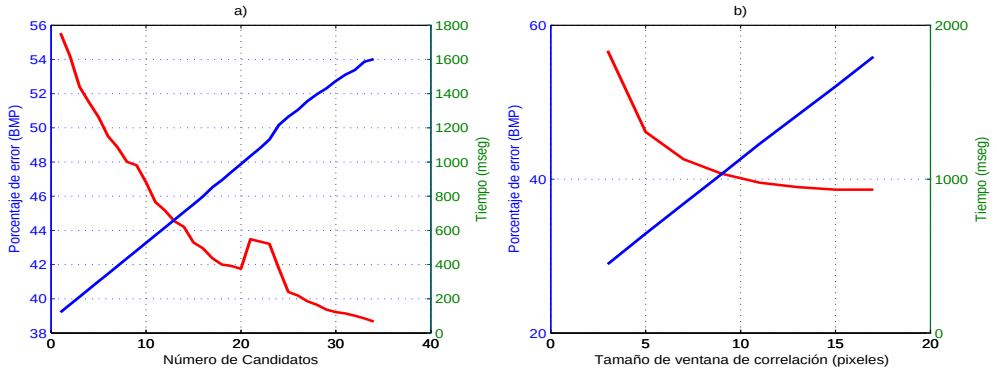


Figura 4.25. Balance entre exactitud y velocidad para la escena Teddy: (a) Balance para obtener el número de candidatos, (b) Balance para obtener el tamaño de ventana de correlación.

Método básico - Balance exactitud-tiempo

Escena	Tamaño de ventana de correlación (pixeles)	Número de candidatos	Tiempo promedio (mseg)	BMP
Tsukuba	11 x 11	5	216.78	15.21
Venus	15 x 15	4	151.82	20.42
Conos	11 x 11	1	57.75	40.23
Teddy	19 x 19	39	822.81	43.93

Tabla 4.2. Tabla de parámetros óptimos para mantener el balance entre exactitud y velocidad.

Método básico con sub-imágenes - Parámetros óptimos

Escena	**Tamaño de ventana de correlación (pixeles)**	**Número de candidatos**	**Ancho de sub-imagen (pixeles)**	**Traslape entre sub-imágenes (pixeles)**	**Tiempo promedio (mseg)**	**BMP**
Tsukuba	17 x 15	13	16	64	374.10	13.19
Venus	17 x 17	16	64	40	1075	13.43
Conos	17 x 17	4	128	60	478.11	58.82
Teddy	17 x 17	34	128	60	3466	55.26

Combinación promedio óptima para el algoritmo con el método básico:

	15 x 15	50	128	120	1343	37.74

Tabla 4.3. Tabla de parámetros óptimos del método básico con sub- imágenes.

Métodos en tiempo real

Método	**Tamaño de imagen**	**Velocidad (fps)**	**Plataforma**	**Disparidad máxima**	
Propuesto	384×288	3.17	Intel Core 2 Duo 2.53GHz (secuencial)	16	
SMP [32]	320×240	31.25	Intel Pentium III, 800 MHz (paralelo)	32	
BM [32]	320×240	33.68	Intel Pentium III (paralelo)	32	
Point Grey [25]	320×240	20	Pentium IV 1.4 GHz (paralelo)	32	
Triclops [25]	320×240	13	Pentium IV 1.4 GHz (paralelo)	32	
SRI SVS [58]	320×240	30	Pentium III 700 MHz (paralelo)	32	

Tabla 4.4. Tabla comparativa de velocidades entre métodos del estado del arte.

Capítulo 5

Conclusiones

En este trabajo de tesis se analizó un algoritmo estéreo para la estimación de disparidad a partir de dos vistas, basado en la correlación por fase que puede ser clasificado como un método local de correspondencia por bloques. El algoritmo implementado muestra buenos resultados en escenas que contienen objetos planos o que no presentan muchos planos de profundidad. Se comprobó la reducción del espacio de búsqueda con un porcentaje por arriba del 37 % en todas las imágenes de prueba, además para aquellas escenas donde existen muchos objetos con volumen y un rango grande de disparidad (ej. 60 pixeles) el algoritmo muestra muy buenos resultados en velocidad, pero un porcentaje de error alto en comparación a las escenas con pocos objetos u objetos planos.

La etapa de filtrado disminuye la velocidad del algoritmo pero aumenta la precisión. El porcentaje de penalidad con respecto al tiempo es de 20 % mientras que en promedio el filtrado mejora el BMP alrededor de 3 %. Debido a la alta penalidad en tiempo y pequeña mejora en precisión el filtrado no es una buena opción.

El algoritmo empleando sub-imagenes pierde exactitud. El BMP aumenta alrededor de 5 % al 12 % mientras que el tiempo de ejecución aumenta entre el 20 % y el 200 % por lo cual no es una buena aproximación al decremento del BMP.

El método analizado no presenta buenos resultados en cuanto a precisión como se ve reflejado en el BMP para los pares estéreo analizados. Uno de los mayores problemas detectados es el hecho de que los desplazamientos son fraccionarios, los cuales deberán ser abordados para alcanzar precisiones que se puedan comparar con los últimos algoritmos presentados en la literatura. Una forma de incluir estimación fraccionaria es por medio de la interpolación de la imágenes.

El método no alcanza las velocidades de procesamiento esperadas para un algoritmos en tiempo real. Para ser considerada una implementación en tiempo real debe alcanzar los 15 cuadros por segundo o equivalentemente un promedio de 66 ms por cuadro. Nuestra implementación alcanzo en promedio $658ms$. Sin embargo, la implementación presentada es secuencial. Lo cual no explota el potencial del procesador. Una implementación en paralelo podría disminuir este tiempo alrededor de un 40 % o mas dependiendo del grado de paralelismos empleado. Las implementaciones del estado del arte emplean Graphic Processor Units (GPU) o Field Programmable Gate Arrays (FPGA). Este tipo de implementaciones

evitan al sistema operativo el cual contribuye con el 30 % del tiempo de ejecución reportado. Por lo cual como parte del trabajo futuro se podría realizar la implementación en paralelo utilizando GPUs o FPGA.

5.1. Trabajo a futuro

5.1.1. Método básico a nivel sub-pixel

Para reducir el espacio de búsqueda el algoritmo propuesto localiza los máximos de la función POC entre el mismo renglón de dos imágens f y g. Estos máximos representan los posibles valores de disparidad, que corresponden a los desplazamientos de cada pixel del renglón comparado. Hasta ahora se ha supuesto que estos desplazamientos son valores enteros, pero cabe la posibilidad de que existan desplazamientos con valores fraccionarios, situación que es frecuente en aplicaciónes de visión estéreo. Supongamos que se calcula la disparidad con 1/5 de precisión de pixel entre un par de imágenes estéreo, podríamos entonces calcular la distancia a la que se encuentra algún objetivo con una precisión 5 veces mayor, comparada con un sistema de visión estéreo de resolución a nivel pixel. De hecho, al aumentar la precisión en el cálculo de disparidades, se obtiene un aumento en la calidad de los mapas de disparidad resultantes, lo que es la principal motivación para implementar el algoritmo propuesto a nivel sub-pixel.

Anteriormente se mencionó que una de las principales causas por la que no se pudo obtener mejores resultados en la estimación de disparidad se debe a que la mayoría de la escenas de prueba presentan desplazamientos fraccionarios. En la figura 5.1 se compara un renglón del ground truth con el mismo renglón del mapa de disparidad estimado utilizando el método básico a nivel pixel para dos escenas (Tsukuba y Venus). Observe como los valores de la estimación del renglón de Tsukuba (rojo figura 5.1(a)) no resultan tan distantes de los valores reales (azul 5.1(a)). Lo anterior debido en parte a que los valores de disparidad reales de la escena son valores enteros. Caso contrario al de la escena Venus donde los desplazamientos reales son valores fraccionarios. En la figura 5.1(b) se puede observar como la estimación (azul) no es tan precisa para ese renglón respecto al ground truth (rojo), debido a que es imposible estimar valores fraccionarios. Planteado el problema se decidió implementar el método básico a nivel sub-pixel.

Las etapas extras y modificaciones a las realizadas en el método básico hechas para obtener la estimación de disparidad con precisión sub-pixel son:

1. **Interpolación**: Se tiene la necesidad de aumentar la resolución de las imágenes de entrada f y g de tamaño $nc \times nr$ de tal forma que un sobremuestreo de ellas debe realizarse, esto se logra calculando las intensidades para pixeles intermedios por medio de algún método de interpolación. En principio duplicando cada columna de una imagen podemos aumentar al doble en dirección horizontal el tamaño de esta, de igual manera se puede aumentar al doble la imagen en dirección vertical duplicando cada renglón. Se utilizaron tres métodos diferentes, interpolación bilineal, vecino más próximo y bicúbica. La interpolación se realizó en dirección horizontal de modo que el factor de sobremuestreo L determina el nuevo tamaño de las imágenes f y g. Esto

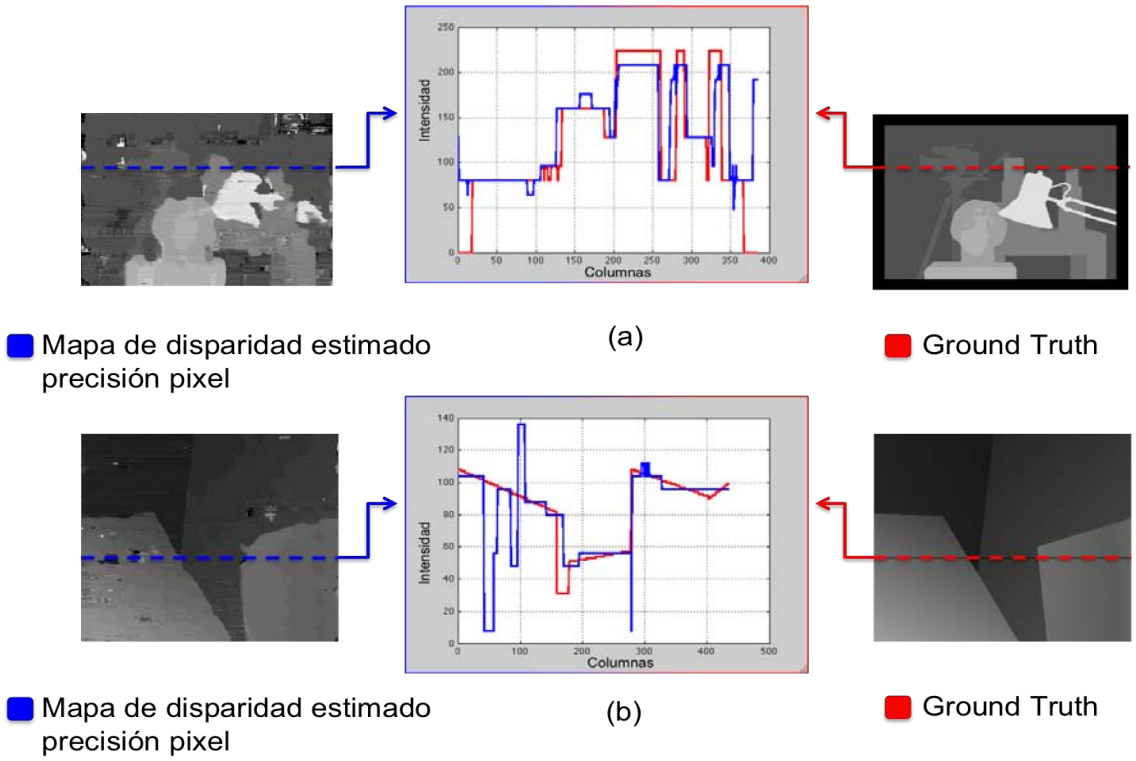


Figura 5.1. Renglón 150 de la escena Tsukuba y renglón 227 de la escena Venus: (a) Ground Truth (rojo) vs. Mapa estimado con precisión de un pixel (azul), (b) Ground Truth (rojo) vs. Mapa estimado con precisión de un pixel (azul)

es $f(x,y) = f(x*L,y)$ y $g(x,y) = g(x*L,y)$ donde $x = 0,1,2,..nc$ y $y = 0,1,2,..nr$.

2. **Correlación por Fase de las imágenes interpoladas**: Esta etapa no tiene modificación alguna respecto a la expuesta en el capítulo de Metodología a excepción de la longitud de las señales involucradas aumentada por el factor de sobremuestreo L. Esto es: Dado un par de imágenes estéreo $f(Lx,y)$ y $g(Lx,y)$, con $0 \leq x < Lnc$ y $0 \leq y < nr$, que representan la imagen izquierda y derecha, respectivamente. Se obtiene la correlación por fase $r_i(Lx)$ entre los renglones $f_i(Lx) = f(Lx,y)$ y $g_i(Lx) = g(Lx,y)$ con $0 < x < Lnc$ y y fija en i. Esto es:

   - Para i desde 0 hasta nr:

$$r_i(Lx) = \frac{1}{LN} \sum_{k=0}^{LN-1} R_i(Lk) e^{-\frac{j2\pi xLk}{N}} \tag{5.1}$$

$$R_i(Lx) = \frac{F_i(Lk)G_i^*(Lk)}{|F_i(Lk)G_i^*(Lk)|} \tag{5.2}$$

   donde $R_i(Lk)$ es el espectro cruzado normalizado entre los renglones $f_i(Lx)$ y $g_i(Lx)$, $F_i(Lk)$ es la transformada discreta de Fourier del i-ésimo renglón y $G_i(Lk)^*$ es el complejo conjugado de la transformada discreta de Fourier, del renglón correspondiente $g_i(Lx)$.

3. **Búsqueda de candidatos.**: Para cada $r_i(Lx)$ obtenido de la etapa anterior se forma un conjunto de nk candidatos, donde cada candidato es la posición de alguno de los máximos m más significativos de la función POC $r_i(Lx)$. Particularmente se escogen aquellos valores positivos de m y en las posiciones d_n con valor menor o igual al valor máximo de disparidad LD. Nótese que debido a estas restricciones es posible encontrar solo n número de candidatos y no nk como se desea. Esto es:

   - Para y desde 0 hasta nr y mientras n sea menor o igual a nk:

     Se encuentra el máximo m más significativo de $r_i(Lx)$ y se localiza su posición d_n. Si $d_n <= LD$ y $m > 0$ entonces se almacena d_n en el conjunto $candidatos[n]$, se vuelve cero su altura ($m = 0$) y se incrementa n.

4. **Elección del mejor candidato.**: Para cada pixel (x,y), se encuentra el candidato d_n/L con el menor error dado como:

$$d(x,y) = \underset{d_n/L=d_1/L,\ldots,d_n/L}{\arg\min} \{\Phi(x,y,d_n/L)\} \tag{5.3}$$

   donde n es el número de candidatos encontrados para el renglón y, $d(x,y)$ es el mapa de disparidad y $\Phi(x,y,d_n/L)$ es una función que mide la diferencia entre una vecindad de tamaño $2*wsad \times 2*wsad$ de $f(x,y)$ y una vecindad similar de $g(x-d_n/L,y)$. Note que la búsqueda se realiza solamente hasta n candidatos y no a través de todo el rango de disparidad. Esta minimización puede realizarse de la siguiente forma:

   I. Para cada $n = 1,\ldots,m$

      - Si $\Phi(x,y,d_n/L) < Error_{mnimo}(x,y)$,
        entonces $d(x,y) = d_n/L$ y $Error_{mnimo}(x,y) = \Phi(x,y,d_n/L)$.

Para elegir el valor de disparidad para cada pixel y formar el mapa disparidad $d(x, y)$ se utiliza una técnica de costo agregado, para hacer más eficiente el cálculo del error mínimo entre ventanas de tamaño $2 * wsad + 1$, donde la idea en general es calcular el error por columnas, de tal forma que no se necesite volver a calcular para cada pixel el error de toda la ventana completa.

Para medir los resultados obtenidos con el método básico a nivel sub-pixel es necesario introducir una nueva medida de error debido a que el BMP es un buen indicador de si un algoritmo funciona en general o no de manera aceptable pero no es buena medida de error para cálculos más finos como lo es la estimación a nivel sub-pixel. La media que se utilizara es el error cuadrático medio (medido en valores de disparidad) entre el mapa de disparidad estimado $d_{AP}(x, y)$ y el ground truth $d_{GT}(x, y)$ que esta definido como:

$$RMS = \left(\frac{1}{N} \sum_{(x,y)} |d_{AP}(x, y) - d_{GT}(x, y)|^2 \right)^{1/2} \tag{5.4}$$

donde N es el número total de pixeles.

En la figura 5.2 se muestra para un renglón de la escena Venus el histograma normalizado del ground truth (rojo), la función de correlación por fase (azul continua) y los posibles valores de disparidad estimados (azul punteada) para cada pixel del renglón, para precisiones de 1 pixel (a), 1/2 de pixel (b) y 1/4 de pixel (c). De igual forma en la figura 5.3 se muestra el histograma normalizado, la función de correlación por fase y los candidatos para el mismo renglón de la escena Venus pero con precisiones de 1/6 de pixel (a), 1/8 de pixel (b) y 1/10 de pixel (c). Para estimar la disparidad con las distintas precisiones se realizó el sobremuestreo del par de imágenes estéreo utilizando interpolación bilineal, un tamaño de ventana de correlación de 19×19 y 15 candidatos. Se puede observar de las distintas gráficas descritas anteriormente como conforme se aumenta la precisión de estimación resultan más candidatos que corresponden a los valores de disparidad más frecuentes dentro del renglón, lo que resulta en una reducción del error. Por ejemplo observemos la figura 5.2 (a) donde la precisión es de 1 pixel, note como solo 6 candidatos de los 14 que se obtuvieron son los que ayudan a la estimación de la disparidad. Mientras que en la figura 5.2 (c) son 17 candidatos de 48 que se obtuvieron los que ayudan a la estimación. De esta observación se puede concluir que el método a nivel sub-pixel logra obtener más candidatos con valores correspondientes a los valores más recurrentes de disparidad de los pixeles del renglón, lo que resultara en una mejor estimación del mapa de disparidad. Por otro lado aquellos candidatos que no corresponden con los valores más recurrentes y que por lo tanto no ayudan a mejorar la estimación de disparidad son mayoría, lo que lleva a pensar que una nueva técnica de selección de candidatos es necesaria.

Los resultados del método básico a nivel sub-pixel con distintos métodos de interpolación se muestran en las tablas 5.1 (bilineal), 5.2 (vecino más próximo) y 5.3 (bicúbica), donde se resalta en gris la precisión que resulta con el error más pequeño. El método básico a nivel sub-pixel reduce a poco más del 75 % el error obtenido con el método básico a nivel pixel, lo que hace prometedora esta línea de estudio. La interpolación más adecuada según los resultados presentados de RMS es la bicúbica.

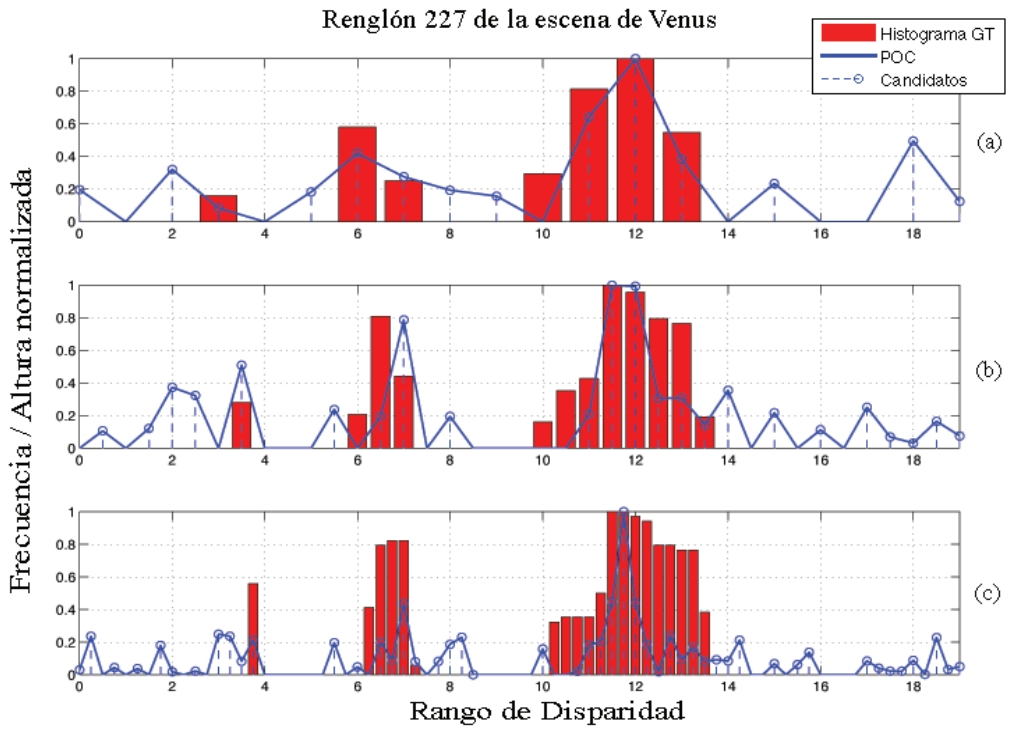


Figura 5.2. Histograma del renglón 227 de la escena de Venus: (a) Precisión de 1 pixel, (b) Precisión de 1/2 de pixel, (c) Precisión de 1/4 de pixel.

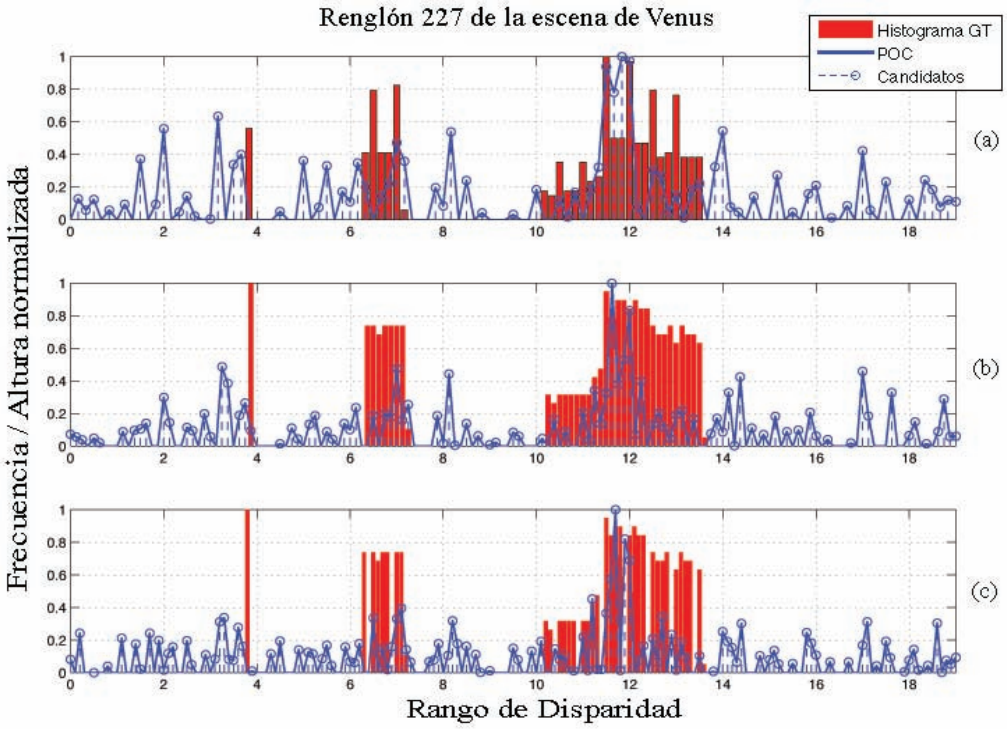


Figura 5.3. Histograma del renglón 227 de la escena de Venus: (a) Precisión de 1/6 pixel, (b) Precisión de 1/8 de pixel, (c) Precisión de 1/10 de pixel.

Finalmente en la figura 5.4 se compara un renglón del ground truth con el mismo renglón del mapa de disparidad estimado utilizando el método básico a nivel sub-pixel para dos escenas (Tsukuba y Venus). Observe como los valores de la estimación del renglón de Tsukuba (azul 5.4(a)) no resultan tan distantes de los valores reales (rojo figura 5.4(a)) y tiene mejoras mínimas respecto a la estimación a nivel pixel (negro 5.4(a)). Lo anterior debido en parte a que los valores de disparidad reales de la escena son valores enteros, las mejoras con el método a nivel sub-pixel no se deben a un mayor número de candidatos con valores iguales a los reales si no más bien al aumento en la probabilidad de acertar al valor real. Para el caso de la escena Venus donde los desplazamientos reales son valores fraccionarios. En la figura 5.4(b) se puede observar como la estimación (azul) no es tan precisa pero también presenta mejoras mínimas respecto a la estimación a nivel pixel (negro) para ese renglón respecto al ground truth (rojo), debido a que son necesarios más valores fraccionarios. Por lo que se concluye que el método-básico a nivel sub-pixel si representa mejoras en la precisión pero es necesario un análisis más profundo de las implicaciones alrededor de la propuesta.

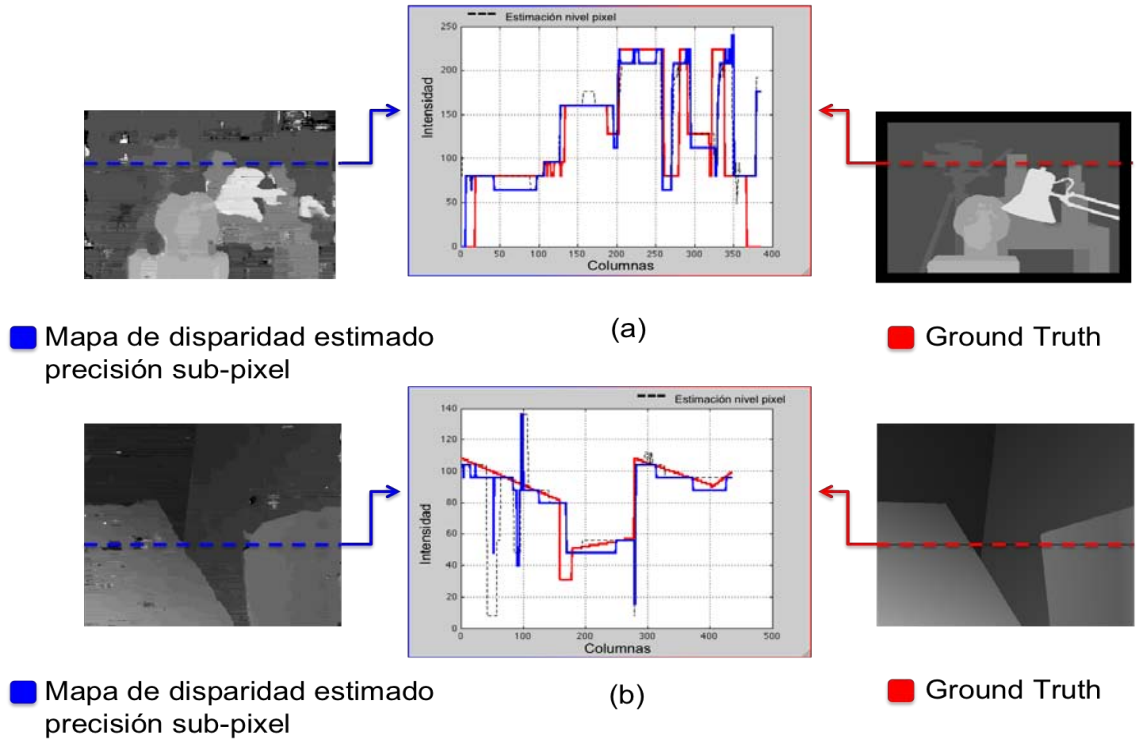


Figura 5.4. Renglón 150 de la escena Tsukuba y renglón 227 de la escena Venus: (a) Ground Truth Tsukuba (rojo) vs. Mapa estimado con precisión de 1/10 de pixel e interpolación bilineal (azul), (b) Ground Truth (rojo) vs. Mapa estimado con precisión de 1/4 de pixel utilizando interpolación bicúbica (azul)

Método básico con precisión sub-pixel

Escena	**Precisión**	**Número de candidatos**	**Tiempo (ms)**	**RMS**
Tsukuba	1	15	315.94	27.1593
	1/2	30	650.336	1.70655
	1/4	60	1259.24	1.68167
	1/6	90	1879.99	1.66988
	1/8	120	2515.26	1.65597
	1/10	150	3132.88	1.65284

Escena	**Precisión**	**Número de candidatos**	**Tiempo (ms)**	**RMS**
Venus	1	15	694.16	12.2088
	1/2	30	1549.85	1.36553
	1/4	60	2998.32	1.33709
	1/6	90	4487.72	1.34965
	1/8	120	6008.07	1.3464
	1/10	150	7528.5	1.34193

Escena	**Precisión**	**Número de candidatos**	**Tiempo (ms)**	**RMS**
Conos	1	1	57.75	50.6854
	1/2	2	122.068	10.8063
	1/4	4	237.362	10.0857
	1/6	6	356.42	9.99786
	1/8	8	470.994	9.90113
	1/10	10	596.39	9.96401

Escena	**Precisión**	**Número de candidatos**	**Tiempo (ms)**	**RMS**
Teddy	1	39	1602	21.4866
	1/2	78	3955.71	5.31769
	1/4	156	7870.47	5.27225
	1/6	234	11907.3	5.27409
	1/8	312	15940.8	5.24011
	1/10	390	19963.3	5.25227

Tabla 5.1. Tabla de resultados con diferentes precisiones de pixel utilizando los parámetros óptimos del método básico y una interpolación bilineal.

Método básico con precisión sub-pixel

Escena	Precisión	Número de candidatos	Tiempo (ms)	RMS
Tsukuba	1	15	315.94	27.1593
	1/2	30	841.205	1.83458
	1/4	60	1621.67	1.83949
	1/6	90	2715.69	1.84521
	1/8	120	2945.34	1.84926
	1/10	150	3600.05	1.85167

Escena	Precisión	Número de candidatos	Tiempo (ms)	RMS
Venus	1	15	694.16	12.2088
	1/2	30	1802.88	1.49985
	1/4	60	3365.24	1.4936
	1/6	90	5447.89	1.50668
	1/8	120	6661.1	1.50432
	1/10	150	9209.12	1.50225

Escena	Precisión	Número de candidatos	Tiempo (ms)	RMS
Conos	1	1	57.75	50.6854
	1/2	2	118.783	10.8473
	1/4	4	231.067	9.63122
	1/6	6	348.036	9.39469
	1/8	8	458.704	9.6844
	1/10	10	582.267	9.85992

Escena	Precisión	Número de candidatos	Tiempo (ms)	RMS
Teddy	1	39	1602	21.4866
	1/2	78	4512.72	5.38338
	1/4	156	8975.09	5.2941
	1/6	234	13863.8	5.30289
	1/8	312	17834.6	5.29159
	1/10	390	22632.9	5.29788

Tabla 5.2. Tabla de resultados con diferentes precisiones de pixel utilizando los parámetros óptimos del método básico y una interpolación conocida como vecino más próximo (Nearest Neighbor).

Método básico con precisión sub-pixel

Escena	**Precisión**	**Número de candidatos**	**Tiempo (ms)**	**RMS**
Tsukuba	1	15	315.94	27.1593
	1/2	30	650.207	1.80855
	1/4	60	1257.51	1.76952
	1/6	90	1880.22	1.77115
	1/8	120	2506.44	1.77344
	1/10	150	3141.31	1.76434

Escena	**Precisión**	**Número de candidatos**	**Tiempo (ms)**	**RMS**
Venus	1	15	694.16	12.2088
	1/2	30	1548.15	1.34042
	1/4	60	2994.38	1.32765
	1/6	90	4508.57	1.3272
	1/8	120	6012.13	1.29062
	1/10	150	7562.03	1.30424

Escena	**Precisión**	**Número de candidatos**	**Tiempo (ms)**	**RMS**
Conos	1	1	57.75	50.6854
	1/2	2	127.716	11.159
	1/4	4	250.5	10.2797
	1/6	6	374.374	10.1654
	1/8	8	495.494	10.0674
	1/10	10	629.39	10.1427

Escena	**Precisión**	**Número de candidatos**	**Tiempo (ms)**	**RMS**
Teddy	1	39	1602	21.4866
	1/2	78	3950.38	5.29737
	1/4	156	7845.2	5.22528
	1/6	234	11914.1	5.2126
	1/8	312	15959.4	5.1884
	1/10	390	20002.7	5.19125

Tabla 5.3. Tabla de resultados con diferentes precisiones de pixel utilizando los parámetros óptimos del método básico y una interpolación bicúbica.

Referencias

[1] H. Gross, F. Blechinger y B. Achtner. *Handbook of Optical Systems*. Vol.1, Fundamentals of Technical Optics, Herbert Gross, Abril 2005.

[2] R.F. Lecumberry. *Cálculo de disparidad y segmentación de Objetos en secuencias de video*. Tesis de Maestría en Ingeniería Eléctrica. Montevideo, Uruguay, Agosto 2005.

[3] Emanuele Trucco y Alessandro Verri, *Introductory Techniques for 3D Computer Vision*. Prentice Hall,1998.

[4] Institute of Software Technology and Interactive Systems, Vienna University of Technology. *Evaluation and Design of Energy Functions for Global Stereo Matching*.`http://www.ims.tuwien.ac.at/research_detail.php?ims_id=stereo_matching`.

[5] R. Depaoli, D. Diaz, L. Fernandez y R. Stockly. *Un estudio sobre calibración de cámaras digitales en visión computacional y reconstrucción 3D*. Congreso de Microelectrónica Aplicada. Matanza, Argentina, 2010.

[6] M. Sonka, V. Hlavac y R. Boyle.*Image processing, analysis, and machine vision*. Pacific Grove, 2nd edn., Albany Bonn, PWS Publishing, 1999.

[7] S. Florczyk. *Robot Vision: Video-based Indoor Exploration with Autonomous and Mobile Robots*. WILEY-VCH Verlag GmbH and Co. KGaA, Weinheim,2005.

[8] O. D. Faugeras. *Three-dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, Cambridge (Massachusetts), London, 1993.

[9] M.A. Torres Torriti. *Reconstrucción confiable de superficies usando rango de disparidad adaptivo*. Tesis para obtener el grado de Magister en Ciencias de la Ingeniería. Santiago, Chile. 1998.

[10] E. Krotkov, M. Herbert y R. Simmons. *Stereo Perception and Dead Reckoning for a Prototype Lunar Rover*. The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, U.S.A., Autonomous Robots, 2, 313-331 (1995).

[11] J.Lotti y G. Giraudon. *Correlation Algorithm with Adaptive Window for Aerial Image in Stereo Vision*. Inst Nat. de Recherche en inf. et Autom., Shophia-Antipolis. International Conference on Pattern Recognition, 1994.

[12] G.P. Stein. *Lens Distortion Calibration Using Point Correspondences*. A.I. Memo 1595. Massachusetts Institute of Technology, Artificial Intellingence Laboratory. Noviembre,

1996.

[13] D.V. Papadimitriou y T.J. Dennis. *Epipolar Line Estimation and Rectification for Stereo Image Matching Pairs*. IEEE Transactions on Pattern Analysis and Machine Intelligence, (5)4: 672-676, Abril, 1996.

[14] M. Pollefeys, R. Koch y L. Van Gool. *A simple and efficient rectification method for general motion*. IEEE International Conference on Computer Vision, (1): 496-501, 1999.

[15] B. K. P. Horn. *Robot Vision*. The MIT Press, Cambridge, Massachusetts, U.S.A., 1986.

[16] S. Cochrane y G. Medioni. *3-D Surface Description from Binocular Stereo*. IEEE Transactions on Pattern Analysis and Machine Intelligence, (14)10:981-944, Octubre, 1992.

[17] O. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. The MIT Press. Cambridge, Massachusetts, U.S.A., 1993.

[18] R.C. Gonzalez y R.E. Woods. *Digital Image Processing* 2ed., Prentice Hall, 2002.

[19] S.B. Goldberg, M.W. Maimone y L. Matthies. *Stereo vision and rover navegation software for planetary exploration*. IEEE Aerospace Conference, 2002.

[20] `http://www.intergraph.com/`

[21] `http://www.zeiss.de/`

[22] J. J. Aguilar, F. Torres y M. A. Lope. *Stereo vision for 3D measurement: accuracy analysis, calibration and industrial applications*. Measurement, Vol.18, No.4, 1996.

[23] T. M. Peters, C. J. Henri, P. Munger, A. M. Takahashi, A. C. Evans, B. Davey y A. Olivier. *Integration of stereoscopic DSA and 3D MRI for image-guided neurosurgery*. Computerized Medical Imaging and Graphics, Vol.18, No.4, 1994.

[24] Koichi Ito, Hiroshi Nakajima, Koji Kobayashi, Takafumi Aoki y Tatsu Higuchi. *A Finguerprint Matching Algorithm Using Phase-Only Correlation*. IEICE Trans. Fundamentals, Vol.E87-A, No.3, Marzo 2004.

[25] Myron Z. Brown, Darius Burschka y Gregory D. Hager. *Advances in computational stereo*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(8):993-1008, August 2003.

[26] Daniel Scharstein y Richard Szeliski. *A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms*. International Journal of Computer Vision, pp. 7-42, April-June 2002.

[27] R. Hartley y A. Zisserman. *Multiple View Geometry in Computer Vision*, Cambridge, UK, Cambridge University Press,2000

[28] O. Faugeras, Quang-Tuan Luong y T. Papadopoulo. *The Geometry of Multiple Images: The Laws That Govern the Formation of Multiple Images of a Scene and Some of Their Applications*, MIT Press,2001

[29] I. Ernst y H. Heiko. *Mutual Information Based Semi-Global Stereo Matching on the GPU*. Lecture Notes in Computer Science, 5358:228–239,2008.

[30] Alba A. y Arce-Santana E., *Phase-Correlation Guided Search for Realtime Stereo Vision*. P. Wiederhold and R.P. Barneva (Eds.): IWCIA 2009, LNCS 5852, pp. 212–223, 2009.

[31] E. Arce y J. L. Marroquin. *High-precision stereo disparity estimation using HMMF models*. Image and Vision Computing, 25:623–636, 2007.

[32] Luigui Di Stefano, Massimiliano Marchionni, Stefano Mattoccia, *A fast area-based stereo matching algorithm* Science Direct, Image and Vision Computing 22(2004) 983-1005.

[33] H. Hirschm¨uller, P. R. Innocent, y J. Garibaldi. *Real-time correlationbased stereo vision with reduced border errors*. Int. J. Comput. Vision,47(1-3):229–246, 2002.

[34] B. Bartczak, D. Jung, y R. Koch. *Real-time neighborhood based disparity estimation incorporating temporal evidence*. Lecture Notes in Computer Science, 5096:153–162, 2008.

[35] J. L. Marroquin, S. Mitter, y T. Poggio. *Probabilistic solution of illposed problems in computational vision*. J. Am. Stat. Assoc., 82:76–89,1987.

[36] M. Gong y Y.-H. Yang. *Near real-time reliable stereo matching using programmable graphics hardware*. CVPR 2005.

[37] J. Salmen, M. Schlipsing, J. Edelbrunner, S. Hegemann, y S. Lueke. *Real-time stereo vision: making more out of dynamic programming*. CAIP 2009.

[38] S. Kosov, T. Thormählen, y H.-P. Seidel. *Accurate real-time disparity estimation with variational methods*. ISVC 2009.

[39] W. Yu, T. Chen, F. Franchetti, y J. Hoe. *High performance stereo vision designed for massively data parallel platforms*. IEEE TCSVT 2010.

[40] R. Yang y M. Pollefeys. *Multi-resolution real-time stereo on commodity graphics hardware*. Proc. of CVPR, 2003.

[41] F. Essannouni, R. O. Haj Thami, A. Salam y D. Aboutajdine. *An efficient fast full search block matching algorithm using FFT algorithms*. In-ternational Journal of Computer Science and Network Security, 6(3B):130–133, 2006.

[42] K. Takita, T. Aoki, Y. Sasaki, T. Higuchi y K. Kobayashi. *Highaccuracy subpixel image registration based on phase-only correlation*. IEICE Trans. Fundamentals, vol.E86-A, no.8, pp.1925–1934, Aug. 2003.

[43] K. Takita, T. Aoki, Y. Sasaki, T. Higuchi y K. Kobayashi. *A Fingerprint Matching Algorithm Using Phase-Only Correlation*. IEICE Trans. Fundamentals, vol.E87-A, no.3, March. 2004.

[44] K. Takita, T. Aoki, M.A. Muquit y T. Higuchi. *A Sub-Pixel Correspondence Search Technique for Computer Vision Applications*. IEICE Trans. Fundamentals, vol.E87-A,

no.8, Aug. 2004.

[45] T. Shibahara, T. Aoki, H. Nakajima y K. Kobayashi. *A high-accuracy stereo correspondence technique using 1D band-limited phase-only correlation.* IEICE Electronics Express, vol.5, no.4, Nov. 2007.

[46] A.V. Oppenheim y R.W. Schafer. *Discrete-Time Signal Processing.* Prentice Hall, Second Edition,1999.

[47] E. De Castro y C. Morandi, *Registration of Translated and Rotated Images Using Finite Fourier Transforms.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 9(5):700–703, 1987.

[48] Y. Keller, Y. Shkolnisky y A. Averbuch, *The angular difference function and its application to image registration.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(6):969–976, 2005.

[49] B. S. Reddy y B. N. Chatterji, *An FFT-Based Technique for Translation, Rotation, and Scale-Invariant Image Registration*, IEEE Transactions on Image Processing, 5(8):1266–1271, 1996.

[50] M. Gong, Ruigang Yang y Liang Wang. *A Performance Study on Different Cost Aggregation Approaches Used in Real-Time Stereo Matching.* International Journal of computer vision, Volume 75, Number 2, 283-296,2007

[51] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. Black y R. Szeliski. *Middlebury stereo vision page. http://vision.middlebury.edu/stereo/*, 2007.

[52] F. Essannouni, R. Oulad Haj Thami, Ahmed Salam y Driss Aboutajdine, *An efficient fast full search block matching algorithm using FFT algorithms*, International Journal of Computer Science and Network Security, VOL.6 No.3B, March 2006.

[53] Librería OpenCV (Open Source Computer Vision) `http://opencv.willowgarage.com/wiki/`

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

[55] D. Scharstein y R. Szeliski. *High-accuracy stereo depth maps using structured light.* In IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2003), volume 1, pages 195-202, Madison, WI, June 2003.

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

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

[58] K. Konolige, *Small Vision Systems: Hardware and Implementation*, Proc. Eighth Int'l

Symp. Robotics Research, 1997.