



**INSTITUTO POLITÉCNICO
NACIONAL**



ESCUELA SUPERIOR DE INGENIERÍA MECÁNICA Y ELÉCTRICA

SECCIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

DOCTORADO EN COMUNICACIONES Y ELECTRÓNICA

**DESARROLLO DE MÉTODOS Y ALGORITMOS DE RECONSTRUCCIÓN
DE OBJETOS TRIDIMENSIONALES USANDO VIDEO A COLOR CON
IMPLEMENTACIÓN EN HARDWARE ESPECÍFICO**

T E S I S

**QUE PARA OBTENER EL TÍTULO DE DOCTOR EN
COMUNICACIONES Y ELECTRÓNICA**

PRESENTA:

M. EN C. EDUARDO RAMOS DIAZ

ASESOR:

DR. VOLODYMYR PONOMARYOV

MÉXICO D.F. JUNIO DEL 2011



M. en C. Eduardo Ramos Diaz

**DESARROLLO DE
MÉTODOS Y
ALGORITMOS DE
RECONSTRUCCIÓN DE
OBJETOS
TRIDIMENSIONALES
USANDO VIDEO A
COLOR CON
IMPLEMENTACIÓN EN
HARDWARE
ESPECÍFICO**



INSTITUTO POLITÉCNICO NACIONAL
SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

SIP-14

ACTA DE REVISIÓN DE TESIS

En la Ciudad de México siendo las 12:00 horas del día 17 del mes de junio del 2010 se reunieron los miembros de la Comisión Revisora de la Tesis, designada por el Colegio de Profesores de Estudios de Posgrado e Investigación de SEPI-ESIME-CULHUACAN para examinar la tesis titulada:

"DESARROLLO DE MÉTODOS Y ALGORITMOS DE RECONSTRUCCIÓN DE OBJETOS
TRIDIMENSIONALES USANDO VIDEO A COLOR CON IMPLEMENTACIÓN EN HARDWARE
ESPECÍFICO"

Presentada por el alumno:

<u>RAMOS</u>	<u>DÍAZ</u>	<u>EDUARDO</u>							
Apellido paterno	Apellido materno	Nombre(s)							
Con registro: <table border="1" style="display: inline-table;"><tr><td>B</td><td>0</td><td>7</td><td>1</td><td>8</td><td>5</td><td>1</td></tr></table>			B	0	7	1	8	5	1
B	0	7	1	8	5	1			

aspirante de:

DOCTORADO EN COMUNICACIONES Y ELECTRONICA

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

LA COMISIÓN REVISORA

Director de tesis

Dr. Volodymyr Ponomaryov

Dr. Héctor Manuel Pérez Meana

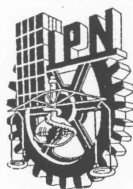
Dr. Gonzalo Isaac Duchén Sánchez

Dr. Oleksiy Pogrebnyak Boleslavovich

Dr. Mariko Nakano Miyatake

PRESIDENTE DEL COLEGIO DE PROFESORES

Dr. Gonzalo Isaac Duchén Sánchez



INSTITUTO POLITÉCNICO NACIONAL
SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

CARTA CESIÓN DE DERECHOS

En la Ciudad de México el día 21 del mes Junio del año 2011, el (la) que suscribe M. en C. Eduardo Ramos Díaz alumno (a) del Programa de Doctorado en Comunicaciones y Electrónica con número de registro B071851, adscrito a SEPI ESIME CULHUACAN, manifiesta que es autor (a) intelectual del presente trabajo de Tesis bajo la dirección de Dr. Volodymyr Ponomaryov y cede los derechos del trabajo intitulado DESARROLLO DE MÉTODOS Y ALGORITMOS DE RECONSTRUCCIÓN DE OBJETOS TRIDIMENSIONALES USANDO VIDEO A COLOR CON IMPLEMENTACIÓN EN HARDWARE ESPECÍFICO, al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

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

M. en C. Eduardo Ramos Díaz

Nombre y firma

DEDICO ESTE TRABAJO A

A mi familia, amigos, y cada una
de las personas que vivió
conmigo éste grandioso reto.

Al Dr. Volodymyr Ponomaryov,
por ser un extraordinario
asesor.

A mis compañeros de la SEPI por
darme su apoyo.

Al Instituto Politécnico Nacional y
al Consejo Nacional de Ciencia
y Tecnología por el apoyo
brindado en la realización de
este trabajo.

Resumen

DESARROLLO DE MÉTODOS Y ALGORITMOS DE RECONSTRUCCIÓN DE OBJETOS TRIDIMENSIONALES USANDO VIDEO A COLOR CON IMPLEMENTACIÓN EN HARDWARE ESPECÍFICO

En la actualidad, un variado número de aplicaciones en el área del procesamiento de imágenes y de video se han propuesto. En dichas aplicaciones, la extracción de características ha tomado relevante importancia en dichos procesos. Una de las áreas emergentes del procesamiento de video es la generación de contenido con percepción tridimensional. En la primera parte de esta tesis se realiza una revisión bibliográfica de las diferentes técnicas utilizadas en el área de la generación de contenidos con percepción tridimensional y se analizan diferentes alternativas para la generación de los mismos. Se presentan las diferentes aplicaciones usadas en la visualización de videos tridimensionales, así como sus características comerciales. En la segunda parte del trabajo de tesis, se presentan diferentes algoritmos para la obtención de mapas de disparidad, los cuales serán usados en el futuro como información útil para generar videos tridimensionales, además, se analizan dichos algoritmos en base a la calidad de sus resultados y el costo computacional. En la tercer parte de la tesis se presenta el algoritmo propuesto, el cual está basado en la transformada wavelet multinivel con funciones atómicas y la síntesis de anáglifos, como método de visualización. En la cuarta parte de la tesis se presentan las diferentes imágenes y videos tanto reales como sintéticos usados, además se presentan las medidas objetivas seleccionadas. Se incluyen las imágenes y videos resultantes, lo anterior para poder realizar una evaluación subjetiva, tomando en cuenta, pérdida de color, fantasmao y otros artefactos. Finalmente, se presenta la implementación de algunos métodos existentes y el método propuesto en hardware específico.

Abstract

Actually, several applications in signal and video processing have been proposed. Feature extraction are important in those applications. One of the emerging areas in video processing is the generation of content with tridimensional perception. In the first part of this work, a brief review of different techniques used in the generation of tridimensional perception content is presented and different algorithms in the same topic are analyzed. We present different approaches used in the visualization of tridimensional videos and its features. In the second part, we present different algorithms to the disparity map computation, which will be used as important information in the 3D video sequences generation, additionally, some algorithms based on the quality results and computational cost are analyzed. In the third part of the work, we introduce the proposed algorithm which is based in the wavelet based on atomic functions with the multi level strategy and the anaglyph synthesis as a visualization method. In the fourth part, we can find the images and video sequences in synthetic and real form and the objective assessments. We include the resulting images and videos in order to perform a subjective evaluation, taking into account the ghosting, loss of color and other artifacts. Finally, the implementation of different algorithms are presented, including the proposed algorithm in specific hardware.

Índice general

Resumen	XI
Índice general	1
Índice de figuras	5
Índice de tablas	7
1. Estado del Arte	9
1.1. Introducción	9
1.2. Planteamiento del problema	11
1.3. Solución propuesta	11
1.4. Objetivos	12
1.5. Justificación	12
1.6. Aportaciones	12
1.7. Cuerpo de trabajo de tesis	13
2. Visualización de imágenes y videos en 3D	15
2.1. Introducción	15
2.2. Algoritmo Photoshop	17
2.3. Algoritmo Dubois	17
2.4. Algoritmo Yaroslavsky, Ideses y Fishbain	19
2.5. Conclusiones del capítulo	23
3. Estimación de la disparidad en imágenes y videos a color	24
3.1. Introducción	24
3.2. Estimación de mapas de profundidad a partir de flujo óptico	25
3.2.0.1. Estimación de mapas de disparidad basado en la técnica de Horn and Schunk	26
3.2.0.2. Estimación de mapas de disparidad basado en la técnica de Lucas and Kanade	26
3.3. Estimación de mapas de disparidad basados en warping	27
3.4. Estimación de mapas de disparidad usando semejanza estéreo basada en regiones	29

3.5. Estimación de mapas de disparidad usando semejanza entre píxeles	31
3.6. Estimación de mapas de disparidad usando funciones wavelet	32
3.7. Propuesta para la estimación de mapas de disparidad	36
3.8. Conclusiones del capítulo	40
4. Simulación de las técnicas existentes y el método propuesto	43
4.1. Imágenes y videos utilizados	43
4.2. Criterios objetivos de valoración de resultados	47
4.3. Resultados obtenidos en las simulaciones	48
4.3.1. Resultados obtenidos en la estimación del mapa de disparidad	48
4.3.2. Resultados cuantitativos para los mapas de disparidad	57
4.3.3. Resultados obtenidos en la síntesis de anáglifos	58
4.3.3.1. Anáglifos resultantes usando imágenes sintéticas a color	58
4.3.3.2. Anáglifos resultantes usando video secuencias a color	63
4.3.4. Tiempos de procesamiento en simulación	63
4.4. Conclusiones del capítulo	71
5. Implementación del algoritmo propuesto en hardware específico	73
5.1. Características del EVM DM642	73
5.1.1. Mapa de Memoria	75
5.1.2. Interfase EMIF	76
5.1.3. Interfase de Memoria SDRAM	77
5.1.4. Interfase de Memoria Flash	77
5.1.5. Interfase UART	77
5.1.6. Interfase de Memoria Síncrona y Asíncrona del FPGA	77
5.1.7. Interfases Puerto/McASP de Video	78
5.1.8. Puertos Decodificadores de Video	78
5.1.9. Puerto del Codificador de Video	78
5.1.10. Funciones de Video del FPGA	78
5.1.11. Interface I ² C	79
5.2. Características del Emulador Blackhawk PCI560	79
5.3. Características del software utilizado	80
5.3.1. Utilizando <i>CCSTM</i> con <i>TargetSupportPackageTM</i>	81
5.4. Implementación del algoritmo propuesto	82
5.5. Tiempos de procesamiento	84
5.6. Conclusiones del capítulo	85
6. Conclusiones generales y trabajo a futuro	87
6.1. Conclusiones generales	87
6.2. Publicaciones con aportaciones científicas	88

6.3. Trabajo a futuro	89
Bibliografía	91
A. Publicaciones	95

Índice de figuras

1.1. Ejemplos de sistemas de visión estereoscópica: a)anáglifos, b)lentes polarizados, c) gafas obturadoras y d)display montado en la cabeza (HMD).	11
2.1. Cubo de colores en el espacio RGB.	16
2.2. Formación más simple de un anáglifo.	17
2.3. Para construir anáglifos es suficiente procesar la imagen izquierda de un estéreo par, remuestreando el canal rojo de acuerdo con el mapa de disparidad.	20
3.1. Proceso realizado por el algoritmo de warping.	29
3.2. Proceso realizado por el algoritmo semejanza estéreo.	31
3.3. Banco de filtros wavelet de Mallat.	35
3.4. Etapas del algoritmo propuesto.	37
3.5. Etapas para la obtención de la disparidad final.	39
3.6. Proceso de remuestreo. La primer imagen muestra la imagen de referencia, la segunda muestra la imagen sensada y la tercer imagen muestra el encimamiento de las dos primeras, en donde la intensidad en A en la imagen de referencia es estimada de las intensidades de una ventana de vecinos cercanos a A en la imagen sensada.	40
4.1. Imágenes a color sintéticas utilizadas: a)Aloe, b) Venus, c) Lampshade1, d)Wood1, e)Bowling1, f)Reindeer1	44
4.2. Videos a color sintéticos y reales utilizados: a)Coastguard, b) Flowers, c) Foreman, d) VideoTest1, e)VideoTest2	45
4.3. Ground truth de las imágenes a color sintéticas utilizadas: a)Aloe, b) Venus, c) Lampshade1, d)Wood1, e)Bowling1, f)Reindeer1	46
4.4. Mapas de disparidad obtenidos mediante la técnica de Lucas and Kanade para: a)Aloe, b)Wood1, c) Bowling1	49
4.5. Mapas de disparidad obtenidos mediante la técnica de warping para: a)Aloe, b) Wood1, c) Bowling1	50

4.6. Mapas de disparidad obtenidos mediante la técnica de semejanza estéreo para: a)Aloe, b) Wood1, c) Bowling1	51
4.7. Mapas de disparidad obtenidos mediante la técnica de semejanza estéreo para: a)Aloe, b) Wood1, c) Bowling1	52
4.8. Mapas de disparidad obtenidos mediante la técnica de wavelets con <i>Coiflets2</i> para: a)Aloe, b) Wood1, c) Bowling1	53
4.9. Mapas de disparidad obtenidos mediante la técnica de multi-wavelets con <i>Biortogonal6.8</i> para: a)Aloe, b) Wood1, c) Bowling1	54
4.10. Mapas de disparidad obtenidos mediante el algoritmo propuesto con la familia de funciones atómicas correspondiente a π_6 para: a)Aloe, b) Wood1, c) Bowling1	55
4.11. Mapas de disparidad obtenidos mediante el algoritmo propuesto con 4 niveles de descomposición y la función π_6 para: a)Aloe, b) Wood1, c) Bowling1	56
4.12. Anaglifos obtenidos usando Lucas y Kanade para: a)Aloe, b) Wood1, c) Bowling1	59
4.13. Anaglifos obtenidos usando warping para: a)Aloe, b) Wood1, c) Bowling1	60
4.14. Anaglifos obtenidos usando semejanza estéreo para: a)Aloe, b) Wood1, c) Bowling1	61
4.15. Anaglifos obtenidos usando M-WAF con π_6 para: a)Aloe, b) Wood1, c) Bowling1, d) Venus, e) Lampshade, f) Reindeer	62
4.16. Anaglifos obtenidos usando Lucas y Kanade para: a)Foreman, b)Flowers, c) VideoTest1	65
4.17. Anaglifos obtenidos usando Lucas y Kanade para: a)Foreman, b)Flowers, c) VideoTest1	66
4.18. Anaglifos obtenidos usando semejanza estéreo para: a)Foreman, b)Flowers	67
4.19. Anaglifos obtenidos usando la wavelet de la familia <i>Biortogonal68</i> para: a)Foreman, b)Flowers, c) VideoTest1	68
4.20. Anaglifos obtenidos usando la wavelet basada en funciones atómicas de la familia up_1 para: a)Foreman, b)Flowers, c)VideoTest1	69
4.21. Anaglifos obtenidos usando M-WAF con π_6 para: a)Coastguard, b)Flowers, c) VideoTest1, d) VideoTest2	70
5.1. Arquitectura del DM642.	74
5.2. Diagrama a bloques del EVM DM642.	76
5.3. Mapa de memoria del EVM DM642.	76
5.4. Emulador Blackhawk PCI560	79
5.5. Algoritmo propuesto desarrollado en Simulink	83

Índice de tablas

4.1. Medidas cuantitativas aplicadas a los mapas de disparidad obtenidos por diferentes métodos a diferentes imágenes.	57
4.2. Tiempos de procesamiento obtenidos por diferentes métodos a diferentes imágenes.	64
5.1. Tiempos de procesamiento obtenidos por diferentes métodos a diferentes imágenes.	85

Capítulo 1

Estado del Arte

En este capítulo se presenta una breve introducción al campo de la visualización de imágenes y videos con percepción en tres dimensiones, además, se establece el planteamiento del problema, solución propuesta, objetivos y justificación.

1.1. Introducción

El cerebro humano interpreta la realidad a partir de las imágenes que le proporcionan los dos ojos. Estas imágenes presentan pequeñas diferencias entre sí y son debidas a la separación de los ojos. La disparidad ó paralelaje entre éstas imágenes es uno de los elementos utilizados por el cerebro para percibir la profundidad [1]. Todas las especies animales que precisan de una percepción muy exacta de la profundidad poseen visión binocular estereoscópica, pues el ángulo de visión de ambos ojos se solapa. La visualización de imágenes estereoscópicas data del siglo XIX, con la aparición y desarrollo de la fotografía, la cual conduce al desarrollo de la cámara y visores estereoscópicos. En dichos visores, las fotos que constituyen el par estereográfico son montadas una junto a otra. El visor consta de unas lentes que facilitan el acomodo de la vista y un vastidor que garantiza que cada ojo perciba únicamente la imagen que le corresponde. La idea es la siguiente: captar o crear dos imágenes de un mismo entorno a partir de dos puntos de vista diferentes, para luego conseguir mediante distintos sistemas, que cada ojo perciba únicamente la imagen que le sea asignada. La adición de los sistemas computacionales abre nuevas posibilidades a los sistemas de éste tipo.

En la actualidad, el procesamiento de imágenes y video han tomado un gran auge debido a su incursión en la creación de contenidos con precepción en tres dimensiones. Por otro lado, se han realizado muchos esfuerzos en el desarrollo de dispositivos estereoscópicos de proyección [2], pero los costos y sus limitaciones en el campo de la visión todavía son grandes,

lo cual ha limitado su aceptación en el mercado. Tomando en cuenta dichas limitaciones, el camino seguido es la implementación de algoritmos eficientes para poder crear videos con percepción en 3D. Para lo anterior, existen diversas técnicas para extraer información útil y además que pueda ser fácilmente implementada en diversos tipos de hardware. En el campo de la visualización, existen algoritmos para construir dichos videos con percepción en 3D pero gozan de desperfectos como fantasmaeo y pérdida de color. Los sistemas de visualización estereoscópica mas importantes se describen a continuación [3].

- Visión paralela. El observador debe centrar su atención a un punto en el infinito, manteniendo los ejes de visión paralelos, se utilizan lentes para acomodar cada imagen a su ojo respectivo.
- Visión cruzada. Las líneas de visión de ambos ojos se cruzan, fijándose en un punto medio más cercano.
- Anáglifos. Las imágenes que forman el par estéreo son superpuestas, previamente coloreadas usando tonos complementarios (rojo-azul). El observador debe usar lentes con filtros de los colores correspondientes.
- Polarización. Las imágenes son proyectadas mediante grandes equipos con filtros polarizadores, girados 90 grados uno respecto a otro. La pantalla debe conservar la polarización de la luz reflejada. El observador debe utilizar unas gafas con lentes polarizados de la misma forma que los proyectores.
- Obturación (shutter glasses). Las dos imágenes son mostradas por un monitor de forma alternada. El observador debe utilizar lentes de cristal líquido que se oscurecen alternadamente de forma sincronizada por el monitor.
- Monitor lenticular. El monitor proyecta una imagen tridimensional sin necesidad de que el observador use ningún otro dispositivo. Sobre la pantalla se adiciona un filtro lenticular el cual ofrece a cada ojo la imagen deseada.

La figura 1.1 muestra algunos ejemplos de sistemas de visualización estereoscópica.

El parámetro de la profundidad es útil en la extracción de la estructura de múltiples vistas de una escena e involucra la determinación de la forma 3D de las superficies visibles en la escena estática de imágenes adquiridas por dos o tres cámaras (secuencias estéreo) ó de una cámara con múltiples posiciones. Con ésta descripción en 3D, podemos crear modelos de terrenos u otros ambientes naturales para usarlos en navegación de robots, simuladores de vuelo, realidad virtual, interacciones hombre-computadora, estéreo microscopía y más.

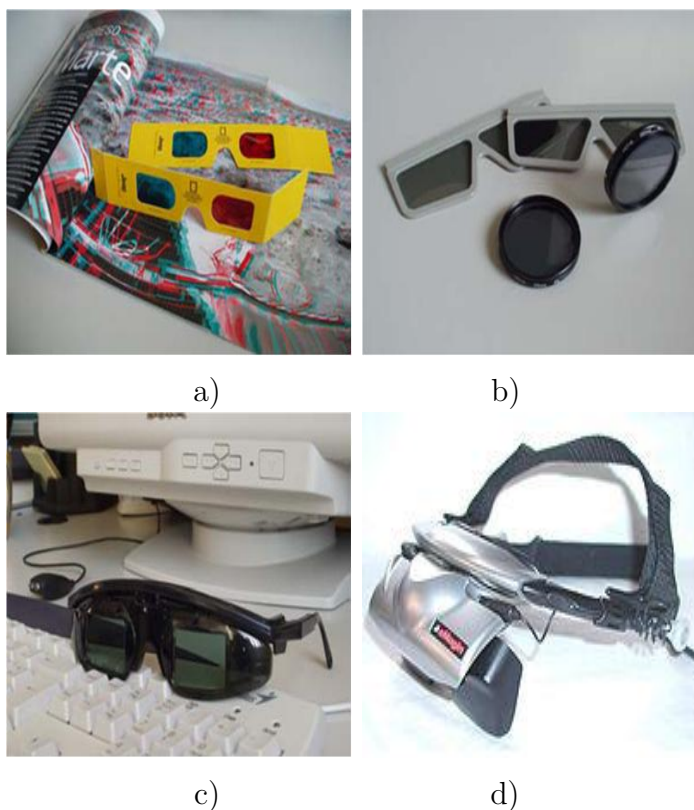


Figura 1.1: Ejemplos de sistemas de visión estereoscópica: a)anáglifos, b)lentes polarizados, c) gafas obturadoras y d)display montado en la cabeza (HMD).

1.2. Planteamiento del problema

Si bien existen algoritmos que realizan la síntesis de anáglifos, es posible utilizar estas técnicas para construir video secuencias con percepción en 3D, basándose en video secuencias en 2D?

1.3. Solución propuesta

En la literatura, existen numerosos métodos para obtener imágenes en tres dimensiones, pero no son aplicados a la obtención de videos en 3D debido a su alto costo computacional. En este trabajo de tesis, se propone investigar, diseñar e implementar nuevos algoritmos para la construcción de videos con percepción en tres dimensiones y su implementación en hardware específico.

1.4. Objetivos

GENERAL

- Analizar y diseñar técnicas para la construcción de imágenes y videos con percepción en tres dimensiones con base en información proveniente de imágenes y videos en dos dimensiones.

PARTICULARES

1. Analizar los métodos y las técnicas para la construcción de anáglifos a color.
2. Analizar los métodos y las técnicas de estimación de movimiento y profundidad en imágenes y videos a color.
3. Diseñar algoritmos novedosos para la obtención de imágenes y videos con percepción en 3D.
4. Implementar los algoritmos diseñados en hardware específico para su posible ejecución en tiempo real.

1.5. Justificación

En la actualidad no existen algoritmos que uso extensivo de la información de movimiento y profundidad en la creación de videos con percepción en 3D. Las técnicas aplicadas limitan su uso a imágenes a color ó al uso de compresores MPEG para el tratamiento del video. En este trabajo se propone la creación de videos con percepción en 3D basados en información de profundidad, sin importar el formato origen del video.

1.6. Aportaciones

Dentro de las aportaciones que éste trabajo de tesis proporciona podemos mencionar los siguientes:

- Desarrollo de algoritmos eficientes en la generación de videos con contenido con percepción tridimensional.
- Implementación de los algoritmos desarrollados justificando su eficiencia.
- Desarrollo de una metodología para la implementación de los algoritmos propuestos en hardware específico.

- Desarrollo de algoritmos basados en las novedosas funciones atómicas en el campo del procesamiento de videos con contenido con percepción tridimensional.
- Publicación de artículos en revistas indizadas.

1.7. Cuerpo de trabajo de tesis

La organización de este trabajo se estructura en seis capítulos y está conformada de la siguiente manera:

- El **Capítulo 1** habla acerca de la introducción al tema propósito de investigación.
- El **Capítulo 2** trata acerca de las técnicas existentes para la observación de imágenes con percepción en tres dimensiones, además de una técnica aplicada a generación de videos en 3D.
- El **Capítulo 3** aborda el tema de la estimación de mapas de disparidad en imágenes a color, además presenta el método propuesto para obtención de anáglifos y su posterior uso en video secuencias con percepción tridimensional.
- El **Capítulo 4** presenta la simulación del método propuesto y algunas técnicas existentes, presenta también, las imágenes y videos utilizados en la simulación y los criterios objetivos utilizados como valoración de dichos métodos. Presenta también los tiempos de procesamiento obtenidos en las simulaciones realizadas.
- El **Capítulo 5** estudia de manera general la arquitectura en hardware utilizada para la implementación del algoritmo propuesto. Además presenta los tiempos de procesamiento obtenidos en la implementación física de algunos algoritmos.
- En el **Capítulo 6** se dan las conclusiones obtenidas de este trabajo proporcionando algunos trabajos a futuro para mejorar el rendimiento obtenido aquí.
- El **Apéndice 1** muestra las publicaciones en congresos y revistas nacionales e internacionales realizadas a lo largo de este trabajo de investigación.

Capítulo 2

Visualización de imágenes y videos en 3D

En este capítulo presentamos algunas formas de obtención de imágenes con percepción tridimensional. Inicialmente se estudian las bases de la percepción tridimensional para después analizar algunos métodos computacionales para generar dichas imágenes y videos.

2.1. Introducción

Como hemos visto en el Capítulo 1, existen diversas formas de observar imágenes con percepción tridimensional, donde la mayoría de ellas requieren de software especializado para poder realizar su tarea. Por el contrario, los anáglifos son una de los métodos más económicos para la visualización tridimensional. Este método fué propuesto por Ducos du Hauron en 1891 [4]. Los anáglifos producen un efecto visual de imágenes 3D cuando son vistas con lentes de filtrado de color. Para poder llevar a cabo la síntesis de anáglifos, es necesario, tener dos imágenes que comunmente se denominan pares estéreo. Para imágenes estéreo en escala de grises, el canal rojo del anáglifo es tomado de la imagen izquierda y el canal azul, de la imagen de la derecha. Es posible incluir el canal verde de cualquiera de las imágenes del par estéreo, lo anterior resulta una imagen en escala de grises con calidad visual buena. En un principio, los anáglifos a color pueden ser sintetizados sólo con la permutación de los canales de color de las imágenes.

Un conjunto de colores representables ó un color sólido en un desplegado usando el sistema de color RGB (rojo, verde, azul) es dado por un cubo unitario R_3 [5].

El color sólido RGB en un vector de seis dimensiones en el espacio R_6 es representado por un hipercubo unitario con 64 vértices correspondientes a las esquinas RGB del cubo

en R_3 para los ojos izquierdo y derecho. Contando en base 2, es posible ordenar los vértices del hipercubo: $[0, 0, 0, 0, 0, 0] = [\text{negro}, \text{negro}]$, $[0, 0, 0, 0, 0, 1] = [\text{negro}, \text{azul}]$ y hasta $[1, 1, 1, 1, 1, 1] = [\text{blanco}, \text{blanco}]$. Los métodos para generar anáglifos de manera electrónica, calculan un mapa de un hipercubo en R_6 a uno en R_3 . Los métodos generales son lineales, esto es, una representación de matrices de 3×6 de un mapa de R_6 a R_3 . Definiendo $v = [r_l, g_l, b_l, r_r, g_r, b_r]^T$ las cuales representan los canales de color de los ojos izquierdo y derecho, los algoritmos regularmente calculan la expresión: $[r, g, b]^T = Bv$. La Figura 2.2 muestra el cubo de colores en el espacio RGB.

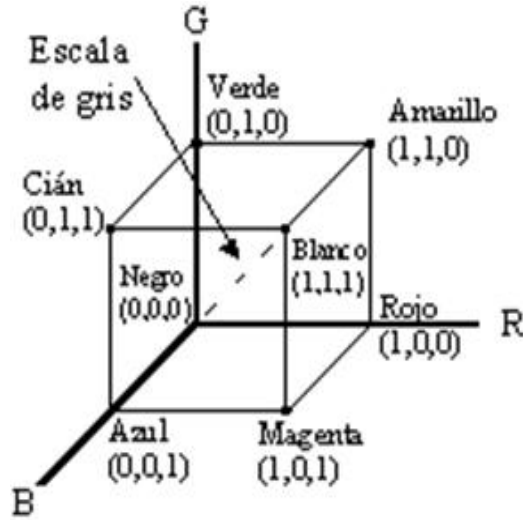


Figura 2.1: Cubo de colores en el espacio RGB.

De esta manera, los anáglifos, presentan una forma sencilla y económica de visualización de imágenes con percepción en 3D. Una posibilidad más sencilla de obtener un anáglifo a color es la simple sustitución del componente rojo de la imagen izquierda, en la imagen derecha. La figura muestra el procedimiento descrito.

Los métodos aquí presentados sólo difieren en los elementos de la matriz B usada para calcular el anáglifo. Por otro lado, si el método produce colores que no es posible representar dentro del cubo RGB, entonces se realiza un recorte de los datos. Aunado a lo anterior, es posible encontrar fusión entre regiones las cuales ocurren cuando regiones adyacentes son graficadas en el mismo anáglifo de color, efectos de fantasmao ó interferencias. Lo anterior afecta la profundidad y el detalle percibido en imágenes estéreo [6]. Para minimizar estos efectos, es posible aplicar una etapa de post-procesamiento donde, en las aplicaciones mas recientes, utilizan el desenfoque y la alineación de imágenes. A continuación presentamos algunos métodos computacionales usados para la obtención de anáglifos.

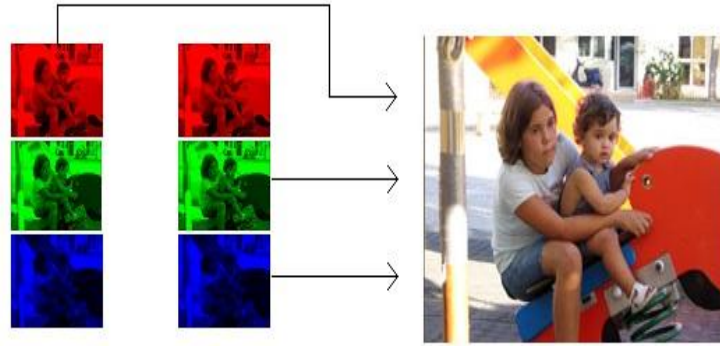


Figura 2.2: Formación más simple de un anáglifo.

2.2. Algoritmo Photoshop

En el algoritmo clásico Photoshop [7], el componente rojo de la vista del ojo izquierdo se convierte en el componente rojo del anáglifo resultante y viceversa para el componente azul y verde del ojo derecho. Lo anterior es equivalente a proyectar el punto RGB del ojo izquierdo al eje rojo del cubo RGB y el punto RGB del ojo derecho en el plano GB. Los dos vectores resultantes son sumados para clacular el color de cada píxel del anáglifo. Para este caso, la matriz B está dada por:

$$B = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.1)$$

Las regiones adyacentes no son consideradas debido a que todos los colores con el mismo canal rojo del ojo izquierdo y los mismos canales azul-verde de la vista del ojo derecho son mapeados en el mismo color.

2.3. Algoritmo Dubois

El algoritmo Dubois [8] se basa en la proyección de mínimos cuadrados en R_6 al subespacio 3D extendido a partir de columnas de la matriz particionada en 6×3 . Dicha partición se realiza al vector D del lado de la mano derecha, donde la matriz C es la matriz de conversión del espacio de color RGB-CIE y está definida por la siguiente ecuación:

$$R = A_l/A_r, \quad (2.2)$$

$$D = \begin{pmatrix} C & 0 \\ 0 & C \end{pmatrix} \quad (2.3)$$

La conversión del espacio RGB al espacio CIE requiere una transformación lineal representada por la matriz C^3 . La matriz C está definida por:

$$C = \begin{matrix} & X_R & X_G & X_B \\ \begin{matrix} Y_R \\ Y_G \\ Y_B \end{matrix} & & & \\ \begin{matrix} Z_R \\ Z_G \\ Z_B \end{matrix} & & & \end{matrix} \quad (2.4)$$

$$C = \begin{matrix} & 11.6638 & 8.3959 & 4.65843 \\ \begin{matrix} 7.10807 \\ 16.6845 \\ 2.45008 \end{matrix} & & & \\ \begin{matrix} 5.27872 \\ 3.79124 \\ 24.0604 \end{matrix} & & & \end{matrix} \quad (2.5)$$

Además, los valores que definen a las matrices A_l , A_r están dados por:

$$A_l = \begin{matrix} & 5.42327 & 0.807004 & 0.047325 \\ \begin{matrix} 2.70972 \\ 0.50201 \\ 0.0250529 \end{matrix} & & & \\ \begin{matrix} 0.00005 \\ 0.000411 \\ 0.002306 \end{matrix} & & & \end{matrix} \quad (2.6)$$

$$A_r = \begin{matrix} & 0.180431 & 1.6395 & 2.00309 \\ \begin{matrix} 0.448214 \\ 6.31551 \\ 1.3575 \end{matrix} & & & \\ \begin{matrix} 0.289201 \\ 2.3925 \\ 11.062 \end{matrix} & & & \end{matrix} \quad (2.7)$$

Los valores anteriores son obtenidos de las distribuciones espectrales de la función de transferencia de cada filtro.

En el algoritmo, la proyección es minimizada por la distancia euclidiana del vector $R[r, g, b]^T - D$, la cual es la aproximación por mínimos cuadrados. El algoritmo también hace uso del escalamiento aplicándolo a través de la matriz diagonal N . El mapa lineal de R_6 a R_3 está dado por:

$$[r, g, b]^T = N(R^T R)^{-1} R^T D \quad (2.8)$$

La matriz B resultante está conformada por los siguientes valores:

$$B = \begin{matrix} & 0.4561 & 0.050048 & 0.176381 & -0.04347 & -0.08793 & -0.001555 \\ \begin{matrix} -0.04008 \\ -0.01521 \end{matrix} & & & & & & \\ \begin{matrix} -0.03782 \\ -0.20597 \end{matrix} & & & & & & \\ \begin{matrix} -0.01575 \\ -0.00546 \end{matrix} & & & & & & \\ \begin{matrix} 0.37847 \\ -0.07215 \end{matrix} & & & & & & \\ \begin{matrix} 0.73364 \\ -0.11296 \end{matrix} & & & & & & \\ \begin{matrix} -0.01845 \\ 1.2264 \end{matrix} & & & & & & \end{matrix} \quad (2.9)$$

2.4. Algoritmo Yaroslavsky, Ideses y Fishbain

El algoritmo presentado en esta sección es utilizado para obtener videos con percepción tridimensional y además incluye una etapa de mejoramiento de los anáglifos resultantes.

La idea principal [9] para poder sintetizar anáglifos por este método es el uso de mapas de profundidad con el objetivo de obtener información útil de la imagen y usarla posteriormente para manipular los componentes de color de la imagen final y así, construir anáglifos. Este algoritmo puede ser usado tanto en imágenes como en video secuencias [10].

El mapeo de una imagen de un par estéreo en otra, puede ser obtenido por medio del registro espacio-temporal de los píxeles vecinos. En este sentido se obtiene un campo de vectores de disparidad, a este campo se le conoce comúnmente como mapa de profundidad del par estéreo y se considera como un indicador de cuántas posiciones cada píxel se mueve de una imagen a otra. El mapa de profundidad puede ser obtenido a partir de diversas técnicas, tales como técnicas correlacionales, flujo óptico, semejanza estéreo ó métodos híbridos. La mayoría de los métodos tienen la desventaja de poseer un alto costo computacional y no están desarrollados para obtener mapas de disparidad en tiempo real.

El algoritmo presente, propone como alternativa para calcular los mapas de profundidad la extracción de vectores de movimiento que están disponibles en los archivos de video de formato MPEG4. Los vectores de movimiento representan información muy útil para la estimación de la profundidad. En su forma más simple, los vectores de movimiento del eje x pueden ser usados como datos de profundidad. Lo anterior es válido para casos en donde sólo existe movimiento lateral en el eje x y no hay movimiento presente en el resto de la escena. Para otros tipos de movimiento, es necesario realizar una transformación de los mapas de vector de movimiento a mapas de profundidad. En dicha implementación, los mapas de vectores de movimiento se extraen como parte del esquema codificador MPEG4. El codificador tiene la instrucción para extraer vectores de movimiento para cada trama, con el mínimo tamaño de bloque de procesamiento.

En algunos casos, los mapas de vectores de movimiento pueden ser aproximados. Dicha aproximación es posible cuando dos imágenes correspondientes a un par estéreo son tomadas en forma paralela ó cuando son tomadas con disparidades pequeñas, como en el caso de la adquisición epipolar. El algoritmo realiza la conversión de los vectores de movimiento en mapas de disparidad usando la siguiente ecuación:

$$D(i, j) = c\sqrt{MV_X(i, j)^2 + MV_Y(i, j)^2} \quad (2.10)$$

donde $D(i, j)$ es el valor de disparidad de cada píxel (i, j) y $MV_X(i, j)$ y $MV_Y(i, j)$ son los valores de los vectores de movimiento de las imágenes X y Y respectivamente, c es un parámetro de ajuste. Para obtener vectores de movimiento de cada píxel se utiliza la

interpolación del vecino más cercano, ya que no reduce significativamente la calidad del video resultante [11]. Para la formación de los anáglifos, el algoritmo se basa en el remuestreo de imágenes, en este caso la imagen es interpolada por un factor y después remuestreada dada una rejilla que es controlada por el mapa de disparidad. De esta forma es posible utilizar la disparidad encontrada entre las dos imágenes. Es necesario hacer notar que la metodología utilizada no garantiza que el estéreo par resultante sea idéntico que el estéreo par original debido a que el mapa de disparidad contiene inexactitudes y en algunas ocasiones también oclusiones. La siguiente figura muestra el proceso llevado a cabo.

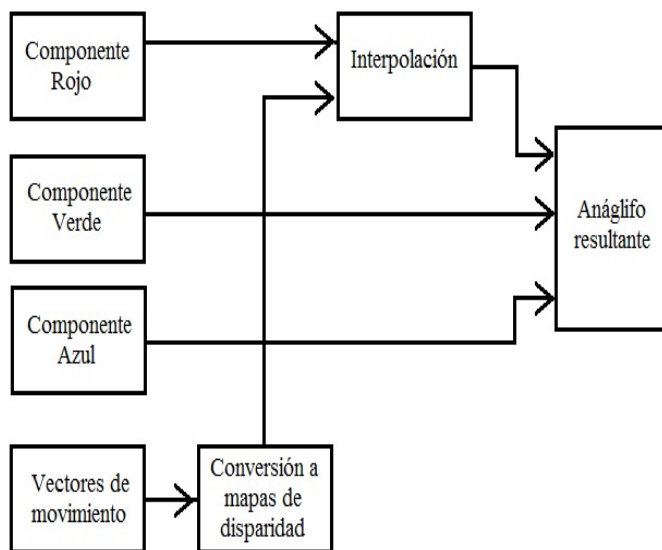


Figura 2.3: Para construir anáglifos es suficiente procesar la imagen izquierda de un estéreo par, remuestreando el canal rojo de acuerdo con el mapa de disparidad.

Mejoramiento de anáglifos

Los efectos del fantasma en los anáglifos son resultado directo de la variación de los campos de profundidad en las imágenes. En casos donde el rango dinámico de las profundidades es muy alta, los efectos de fantasma son más notorios. Es posible mejorar la calidad de los anáglifos por medio de la manipulación del mapa de disparidad. Debido a que la calidad del anáglifo aumenta en función de la disminución de las áreas no superpuestas, es necesario reducir el rango dinámico del mapa de disparidad [12]. La manipulación del rango dinámico puede ser llevada a cabo comprimiendo el rango dinámico por medio de la transformación Ley P -th. En dicha transformación, cada píxel del mapa de profundidad

debe ser sujeto a la transformación mostrada en la siguiente ecuación:

$$D_{final} = aD^P \quad (2.11)$$

donde D_{final} es el mapa final con los nuevos valores de disparidad, a es una constante de normalización que se encuentra dentro de $0 < P < 1$.

De la forma anterior, es posible retener información dentro del mapa de disparidad mientras se reducen las áreas no superpuestas en el anáglifo resultante. Otra forma de mejorar la calidad en los anáglifos es el uso del emborronamiento de color. El acomodamiento se refiere a la distancia percibida del objeto debido al desenfoque de la lente del ojo. La convergencia se refiere a la distancia percibida y que es estimada del par estéreo. El emborronamiento de componentes (convolución espacial) puede llevarse a cabo mediante muchos métodos. El más simple y rápido es la convolución de la imagen en el dominio espacial con una ventana rectangular de pesos constantes:

$$\tilde{I}(x, y) = \frac{1}{(2N_x + 1)(2N_y + 1)} \sum_{n_x=-N_x}^{N_x} \sum_{n_y=-N_y}^{N_y} [I(x - n_x, y - n_y)] \quad (2.12)$$

en donde $\tilde{I}(x, y)$ es la imagen emborronada e $I(x, y)$ es la imagen antes de la convolución. El tamaño de la ventana $(2N_x + 1)(2N_y + 1)$ varía entre 5 y 11 píxeles.

Es posible mejorar los anáglifos siguiendo los pasos que se mencionan a continuación:

- Cálculo del mapa de profundidad de las dos imágenes estéreo.
- Compresión del rango dinámico del mapa de profundidad.
- Síntesis del estéreo par con el mapa de profundidad modificado, y finalmente la creación del anáglifo.

La conversión de contenido en dos dimensiones que está disponible para poder obtener 3D es un tópico muy importante para proveedores de contenido y por lo tanto para la generación de videos en 3D en general [13]. Actualmente éstos tópicos son de gran importancia para aplicaciones 3DTV y la industria del entretenimiento como lo es cinema 3D, lo anterior es debido a que existe un gran vacío en la creación de contenidos en 3D para la nueva era de desplegados 3D y de sistemas cinema 3D. Las aplicaciones desarrolladas hasta el momento se dividen en dos categorías: métodos de conversión manual y automáticos. El costo computacional de las técnicas de conversión manuales es muy elevado, sin embargo, dicho costo computacional refleja una mejor calidad en los contenidos 3D. Por razón económicas las técnicas de conversión manuales son utilizadas mayormente para producciones cinematográficas donde la calidad de las imágenes estéreo resultantes es muy alta. El flujo

de trabajo para la conversión consiste básicamente de tres pasos: segmentación, asignación de profundidad e inpainting. En el primer paso, cada trama de la secuencia es segmentado en objetos sencillos o fragmentos dependiendo de su localización en la escena en el espacio 3D. Es necesario recordar que el contenido en 2D contiene todas las señales de profundidad del mundo real con la excepción de la convergencia y la discrepancia binocular. Así, un operador puede fácilmente seleccionar los objetos que podrían ser segmentados en una imagen o en una secuencia de imágenes.

Si la segmentación de objetos es posible, es necesario reconstruir la escena de acuerdo a la profundidad de los objetos. Este paso en la producción se llama asignación de la disparidad ó composición 3D. La forma más sencilla para generar una vista estereoscópica es el corrimiento de los objetos hacia la izquierda ó hacia la derecha, de acuerdo con la profundidad. El último paso de la conversión es el relleno de las ocusiones ó inpainting. Cuando los objetos son recorridos de acuerdo con su profundidad asignada, la información del fondo no está disponible en las discontinuidades de la profundidad. Estas regiones deben ser recuperadas a partir de las tramas en el tiempo directamente de la información espacial con el tiempo actual. Un simple proceso de rellenado puede ser conformado con una herramienta de procesamiento de imágenes para retocar las partes faltantes; éste último proceso es el mas tardado de todos. Sólo unas pocas compañías en el mundo proveen el servicio de la conversión 2D a 3D.

Los sistemas de conversión automáticos que han sido introducidos durante los últimos años pueden ser divididos como aplicaciones fuera de linea y aplicaciones en tiempo real. Los métodos para la conversión en tiempo real son basados en aproximaciones muy simples y son usados comunmente en aplicaciones de 3DTV donde las degradaciones en la percepción de profundidad esteresocópica es más tolerable. La forma más simple es, como hemos mencionado, el uso de vectores de movimiento extraídos directamente de los datos de compresión de video. Sin embargo esta técnica sólo puede recuperar cierta disparidad relativa acertadamente, si el movimiento de todos los objetos de la escena es directamente proporcional a su distancia con la camara. Otras aplicaciones sugieren la decodificación de la profundidad utilizando el color o la información de intensidad de los pixeles directamente de las imagenes de entrada[14], [15]. Mayor intensidad o mayor saturación en los valores de los pixeles indican que los objetos son más cercanos a la camara. Los sistemas de conversión 2D/3D automáticos más avanzados se especializan en condiciones de captura limitadas. Básicamente, estas aplicaciones nunca corren en tiempo real pero incrementan el tiempo de conversión y la calidad de conversión. En [16] y [17] la estimación del movimiento bidireccional se toma en cuenta para la recuperación de estructura rigida y movimiento. En [18] se presenta un sistema para la generación de super resolución estéreo y video multivistas basado en video monocular usado en la síntesis de vistas estéreo reales.

2.5. Conclusiones del capítulo

Se analizaron las bases de la percepción tridimensional en imágenes a color, la cual toma como base el cubo de colores en donde es posible, mediante combinaciones de sus ejes, la formación de todos los colores existentes.

Se analizó también la forma en cómo se pueden obtener imágenes con percepción tridimensional usando como base imágenes llamadas *pares estéreo*, las cuales forman la parte principal del método.

Se presentaron algunos métodos para construir anáglifos, dichos métodos se basan en su mayoría en una matriz de corrimiento la cual manipula el componente rojo del par estéreo. Con éstos, se obtienen anáglifos aunque por su naturaleza gozan de fantasmaeo y pérdida de color.

Se estudió un algoritmo que, a partir de una video secuencia de entrada, obtiene un video con percepción tridimensional. El algoritmo se basa en el codificador MPEG4 para obtener información útil de la video secuencia de entrada, además el algoritmo presenta una etapa de mejoramiento de anáglifos basada en compresión de rango dinámico.

Se estudió la clasificación de los algoritmos para la conversión de videos 2D en 3D, además, se mencionaron las ventajas y las desventajas de los métodos de conversión manuales y automáticos.

Con base en la información presentada, es necesario encontrar un algoritmo eficiente en calidad y con poca carga computacional para la estimación de mapas de disparidad, ya que representa una de las etapas fundamentales en la construcción de videos con percepción tridimensional. Es necesaria además una etapa de mejoramiento del mapa de disparidad estimado para minimizar los efectos del fantasmaeo y pérdida de color en el anáglifo resultante.

Capítulo 3

Estimación de la disparidad en imágenes y videos a color

En este Capítulo abordaremos las diversas técnicas y algoritmos encontrados en la literatura para la estimación de información de disparidad. Es importante mencionar que, en este estudio, seleccionamos dichos algoritmos con la finalidad de usarlos como proveedores de información útil para poder construir imágenes y video secuencias con percepción tridimensional. A lo largo de los algoritmos investigados, incluimos algoritmos basados en flujo óptico, warping, en estimación de energía y los más novedosos que incluyen la utilización de transformada wavelet. Además se presenta la propuesta desarrollada para la estimación de disparidad en mencionada aplicación.

3.1. Introducción

En la literatura, existen diversos algoritmos para el cálculo de información de profundidad en imágenes, los cuales, en este trabajo pretenden ser extendidos a aplicaciones de video. En esta sección, se hace la revisión de las técnicas más representativas tomando como criterio principal el bajo costo computacional y su rendimiento. Los algoritmos para obtener información de parámetros en imágenes se ha diversificado en los últimos años. Dentro de dichos algoritmos tenemos el cálculo de flujo óptico, la estéreo semejanza y algoritmos híbridos. El cálculo del flujo óptico toma como base un par estéreo ó una secuencia de imágenes dentro de un video para calcular el movimiento de los píxeles dentro de las mismas. El mapa de disparidad puede indicar a partir de una colección de datos, el movimiento de los píxeles dentro de una escena. Cada elemento del mapa de disparidad indica el número de posiciones en que se ha desplazado un píxel. Si bien existe un número grande de algoritmos para

obtener dicha información, la mayoría de ellos se basa en algoritmos recursivos los cuales añaden una excesiva carga computacional, lo cual, los limita en aplicaciones en tiempo real y su uso en video secuencias.

3.2. Estimación de mapas de profundidad a partir de flujo óptico

Un problema fundamental en el procesamiento de imágenes es la medición del flujo óptico. La tarea es calcular de una aproximación del campo de movimiento en 2D, una proyección en 3D de las velocidades de diversos puntos de la superficie de una imagen [19]. Una vez calculadas, las mediciones de la velocidad de la imagen pueden ser usadas en una gran cantidad de aplicaciones. A pesar de que muchos métodos se han dado a conocer, siguen faltando mediciones cuantitativas de los métodos existentes y comparaciones directas para un conjunto de entradas sencillas. Dentro de los métodos más conocidos están los diferenciales, los de semejanza basada en regiones, los basados en energía, los basados en fase, etc. Aunque los métodos mencionados anteriormente tienen diferencias sustanciales, todos ellos contienen conceptualmente tres etapas:

1. Prefiltrado ó suavizado con filtros pasabandas, su objetivo es extraer la estructura de la señal de interés y mejorar la relación de señal a ruido.
2. Extracción de las medidas básicas, así como derivadas espaciotemporales ó correlación local en las estructuras.
3. Integración de las mediciones realizadas para producir el campo de flujo en 2D.

Las técnicas diferenciales calculan la velocidad a partir de derivadas espaciotemporales de las intensidades de las imágenes filtradas. Por principio, usando derivadas de primer orden y basadas en el modelo de movimiento clásico de traslación de la imagen podemos obtener la siguiente expresión:

$$I(x, t) = I(x - Vt, 0) \quad (3.1)$$

donde $V = (u, v)^T$. Aplicando la expansión de Taylor a la ecuación anterior, y considerando que la intensidad es conservada, la ecuación de la restricción de gradiente está dado por:

$$\nabla I(x, t) \cdot V + I_t(x, t) = 0 \quad (3.2)$$

donde $I_t(x, t)$ es la derivada parcial en tiempo de $I(x, t)$, $\nabla I(x, t) = (I_x(x, t), I_y(x, t))^T$ y $\nabla I \cdot V$ denota el producto punto.

3.2.0.1. Estimación de mapas de disparidad basado en la técnica de Horn and Schunk

Horn and Schunk combinan el gradiente de restricciones dado por la ecuación de gradiente incorporando un término global de suavizado para poder restringir el cálculo del campo de velocidad [20]. El campo de velocidad está dado por: $v(x, t) = (u(x, t), v(x, t))$. Dicha restricción se obtiene minimizando la siguiente ecuación:

$$\int_D (\nabla I v + I_t)^2 + \lambda^2 (\|\nabla u\|^2 + (\|\nabla v\|)^2) \quad (3.3)$$

donde la magnitud de λ refleja la influencia del término de suavizado.

Las ecuaciones iterativas utilizadas para minimizar la ecuación 3.3 y como consecuencia obtener la velocidad de la imagen, se presentan en la siguiente ecuación:

$$u^{k+1} = u^{-k} - \frac{I_x[I_x \bar{u}^{-k} + I_y \bar{v}^{-k} + I_t]}{\alpha^2 + I_x^2 + I_y^2} \quad (3.4)$$

$$v^{k+1} = v^{-k} - \frac{I_y[I_x \bar{u}^{-k} + I_y \bar{v}^{-k} + I_t]}{\alpha^2 + I_x^2 + I_y^2} \quad (3.5)$$

donde k denota el número de iteraciones, u^0 y v^0 denotan la estimación de la velocidad inicial y $\bar{u}^k$, $\bar{v}^k$ denotan los promedios de los vecinos [22].

3.2.0.2. Estimación de mapas de disparidad basado en la técnica de Lucas and Kanade

En esta técnica se hace uso del algoritmo de cálculo de mínimos cuadrados con pesos para restringir los elementos de primer orden de la ecuación 3.1, en un modelo constante para la velocidad en cada vecino espacial dentro de una ventana de influencia por medio de la minimización de la siguiente ecuación [21]:

$$\sum_{x \in \Omega} W^2(x) [\nabla I(x, t) \cdot v + I_t(x, t)]^2 \quad (3.6)$$

donde $W(x)$ denota una función ventana que da mayor influencia a la restricción hacia el centro de los vecinos que a los de la periferia. La solución de la ecuación 3.6 está dada por:

$$A^T W^2 A \cdot v = A^T W^2 b \quad (3.7)$$

donde, para n puntos $x_i \in \Omega$ a un tiempo t , además:

$$\begin{aligned} A &= [\nabla[I(x_1)], \dots, \nabla[I(x_n)]]^T, \\ W &= \text{diag}[W(x_1), \dots, W(x_n)], \\ b &= -(I_t(x_1), \dots, I_t(x_n))^T. \end{aligned} \quad (3.8)$$

La solución de la ecuación 3.7 es $v = [A^T W^2 A]^{-1} A^T W^2 b$, la cual puede ser resuelta en forma cerrada cuando el término $A^T W^2 A$ es no singular, ya que es una matriz de 2×2 como se muestra a continuación:

$$A^T W^2 A = \begin{bmatrix} \sum W^2(x) I_x^2(x) & \sum W^2(x) I_x(x) I_y(x) \\ \sum W^2(x) I_y(x) I_x(x) & \sum W^2(x) I_y^2(x) \end{bmatrix} \quad (3.9)$$

donde todas las sumas son tomadas sobre puntos dentro de los vecinos en Ω . En esta implementación, inicialmente la secuencia de imágenes se suaviza con un filtro gaussiano espacio-temporal con una desviación estándar de 1.5 píxeles por trama. Esto ayuda a atenuar el solapamiento temporal y efectos de cuantización en la imágenes de entrada.

3.3. Estimación de mapas de disparidad basados en warping

Esta técnica [23] se basa en un modelo variacional que asume restricciones a ciertos parámetros los cuales son minimizados usando modelos grueso-fino llamados también warping, para formar la ecuación final para calcular el flujo óptico. La restricción de niveles de grises constantes usada también en flujo óptico, la extiende a la restricción de gradientes constantes, lo cual proporciona robustez a la técnica cuando las imágenes presentan cambios en los valores de grises. Para el caso de asumir valores de grises constantes, la siguiente ecuación puede ser utilizada:

$$I(x, y, t) = I(x + u, y + v, t + 1), \quad (3.10)$$

En donde $I : \Omega \subset R^3$ con R como una secuencia de imágenes rectangular y $w := (u, v, 1)^T$ es el vector de desplazamiento buscado entre una imagen en el tiempo t y otra en el tiempo $t + 1$. La versión linearizada del valor constante de gris asumida proporciona la clásica ecuación para encontrar el flujo óptico [22]: $I_x u + I_y v + I_t = 0$. La restricción de gradientes constantes se usa para permitir pequeñas variaciones en los valores de grises y ayuda a

determinar el valor del vector de desplazamiento por un criterio que es invariante bajo cambios de niveles de grises. La ecuación usada se presenta a continuación [24]:

$$\nabla I(x, y, t) = \nabla I(x + u, y + v, t + 1), \quad (3.11)$$

en donde $\nabla = (dx, dy)^T$ denota el gradiente espacial. El modelo estima el desplazamiento de un píxel localmente sin tomar ninguna interacción entre píxeles vecinos, por lo tanto es útil introducir un término que asume el suavizado en el campo de flujo. Esta restricción de suavizado puede ser aplicada en el dominio espacial y en el espacio-temporal, dependiendo si usamos un par de imágenes ó una secuencia de video en la aplicación. En el caso de desplazamientos que son mayores a un píxel por trama, el modelo aplica soluciones multi-escala [25]. Empieza solucionando una versión gruesa, una versión suavizada del problema trabajando con una secuencia de imágenes suavizada. La solución encontrada en la versión gruesa es usada como inicialización en una versión fina del problema hasta que el problema original es resuelto. Tomando en cuenta lo anterior, es posible encontrar una ecuación de energía que penalice las desviaciones del modelo. Definiendo $x := (x, y, t)^T$ y $w := (u, v, 1)^T$, las desviaciones del valor de gris constante y el valor de gradiente constante pueden ser expresadas por la siguiente ecuación:

$$E_{data}(u, v) = \int_{\Omega} (|I(x + w) - I(x)|^2 + \gamma |\lambda I(x + w) - \lambda I(x)|^2) dx, \quad (3.12)$$

donde γ representa el peso entre ambas restricciones.

Para minimizar la ecuación anterior se incluye un término de minimización dado por: $\Psi(s)^2 = \sqrt{s^2 + \epsilon^2}$, con el cual, la ecuación anterior se define como:

$$E_{data}(u, v) = \int_{\Omega} (\Psi |I(x + w) - I(x)|^2 + \gamma |\nabla I(x + w) - \nabla I(x)|^2) dx \quad (3.13)$$

El término de suavizado se incluye penalizando la variación total del campo de flujo, la ecuación usada se presenta a continuación:

$$E_{smooth}(u, v) = \int_{\Omega} \Psi (|\nabla_3 u|^2 + |\nabla_3 v|^2) dx, \quad (3.14)$$

en donde $\nabla_3 := (dx, dy, dt)^T$, el cual indica que es usado el suavizado espacio-temporal. La energía total es la suma de los pesos entre el término de datos y el término de suavizado, la siguiente ecuación muestra lo anterior [26]:

$$E(u, v) = E_{data} + \alpha E_{smooth}. \quad (3.15)$$

Para hacer uso de la aplicación con ideas multi-escala, las cuales son necesarias para aproximar mejor el óptimo global de la energía, es necesario incluir en dichas iteraciones una

estrategia de submuestreo. Se propone el uso de una pirámide de imágenes con un factor arbitrario de submuestreo dado por: $\eta \epsilon(0, 1)$.

Además, la pirámide de imágenes formada es usada empezando del nivel más pequeño y haciendo $w^k = (u^k, v^k, 1)^T$, con $w^0 = (0, 0, 1)^T$. Usando las ecuaciones de Euler-Lagrange en la ecuación anterior y las iteraciones variables w^{k+1} se obtiene la solución.

La siguiente figura muestra el diagrama de flujo de esta aplicación.

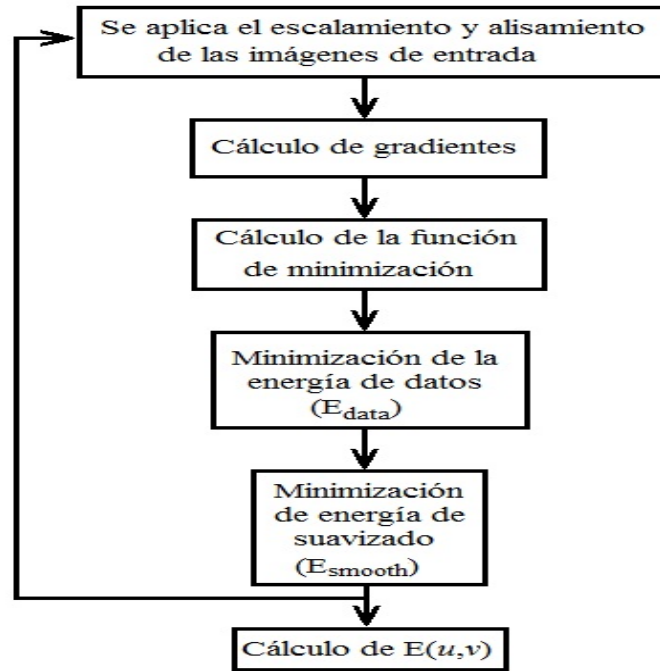


Figura 3.1: Proceso realizado por el algoritmo de warping.

3.4. Estimación de mapas de disparidad usando semejanza estéreo basada en regiones

En éste método [27], se usa la técnica de semejanza aplicada en bloques de procesamiento con varios tamaños de ventana con el objetivo de construir una matriz de error para cada valor de disparidad asignado. Nombrando a la imagen izquierda en formato RGB por $L(i, j, c)$, a la imagen derecha en el mismo formato por $R(i, j, c)$ y la energía de error por

$e(i, j, d)$, para una ventana de tamaño $m \times n$, la energía de error puede ser expresada por:

$$e(i, j, d) = \frac{1}{3 \cdot n \cdot m} \sum_{x=i}^{i+n} \sum_{y=j}^{j+m} \sum_{k=1}^3 (L(x, y + d, k) - R(x, y, k))^2, \quad (3.16)$$

donde k representa los componentes de color RGB de las imágenes y d es la disparidad. Para un determinado rango de búsqueda de disparidad, cada matriz de error $e(i, j, d)$ es aliada aplicando un filtro promedio. El filtro de promedio se encarga de remover los cambios bruscos en la matriz de energía, los cuales pueden dar como resultado un cálculo de semejanza incorrecta. Otra propiedad importante del filtro de promedio iterativo es que puede resolver las variaciones globales de la energía. Cuando se considera una tendencia global del error de la energía, el algoritmo se comporta como un algoritmo basado en regiones. Para un tamaño de ventana de $n \times m$, el filtro de promedio de $e(i, j, d)$ puede expresarse como:

$$\tilde{e}(i, j, d) = \frac{1}{n \cdot m} \sum_{x=i}^{i+n} \sum_{y=j}^{j+m} e(x, y, d). \quad (3.17)$$

Después de la aplicación iterativa del filtro de promedio sobre la energía de error de cada disparidad, se selecciona la disparidad d que presente el mínimo error en la ecuación 3.17, como la mejor estimación de disparidad para cada píxel del mapa de disparidad.

Para poder definir el mejor valor de la estimación de la energía de error del mapa de disparidad, dado un umbral de error promedio, es necesario un parámetro de fiabilidad. Dicho parámetro de fiabilidad está expresado como:

$$R_d = \frac{1}{\text{mean}E_d - ne} = \frac{1}{S_d} \left(\sum_{(i,j) \in [E_d(i,j)=ne]}^{n,m} E_d(i, j) \right)^{-1}, \quad (3.18)$$

donde S_d es el parámetro que representa el número de puntos de energía de error que no están estimados. Finalmente, la energía de error del mapa de disparidad se puede escribir como:

$$E_d(i, j) = \frac{1}{3 \cdot n \cdot m} \sum_{x=i}^{i+n} \sum_{y=j}^{j+m} \sum_{k=1}^3 (L(x, y + d(i, j), k) - R(x, y, k))^2. \quad (3.19)$$

El mapa de disparidad contiene algunas estimaciones de disparidad no fiables de algunos puntos alrededor de los límites de los objetos derivados de objetos ocultos en las imágenes. Estas disparidades son no deseadas y pueden ser detectadas a partir de valores altos en la energía del error. Para incrementar la fiabilidad del mapa de disparidad obtenido, se utiliza un mecanismo de umbralización. La ecuación siguiente muestra dicho mecanismo:

$$\tilde{d}(i, j) = \begin{cases} d(i, j) & E_d(i, j) \leq V_e \\ 0 & E_d(i, j) > V_e \end{cases} \quad (3.20)$$

$$\tilde{E}_d(i, j) = \begin{cases} E_d(i, j) & E_d(i, j) \leq V_e \\ ne & E_d(i, j) > V_e \end{cases} \quad (3.21)$$

donde $\tilde{d}(i, j)$ es la versión mejorada de $d(i, j)$ por medio del filtrado de algunas estimaciones de disparidad. El valor *ne* indica el estado de *no estimado* y $V_e = \alpha \cdot \text{mean}(E_d)$ en la cual α es un coeficiente de tolerancia para ajustar la fiabilidad del proceso de filtrado; decrementando este valor, la disparidad se mejora, disminuyéndolo, el mapa de profundidad tiende a erosionarse debido a la eliminación de más puntos de disparidad en el mapa.

La figura 3.2 muestra el proceso que realiza el algoritmo descrito.

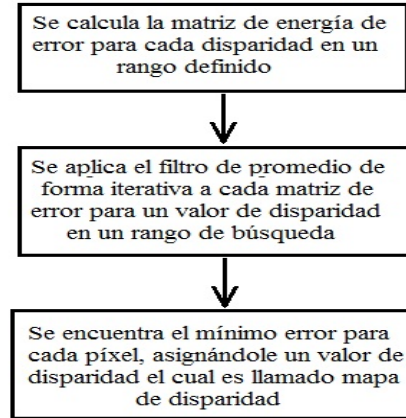


Figura 3.2: Proceso realizado por el algoritmo semejanza estéreo.

3.5. Estimación de mapas de disparidad usando semejanza entre píxeles

Los métodos basados en semejanza entre píxeles se encuentran dentro de la clasificación de los algoritmos locales, debido a que se implementan en ventanas de procesamiento de tamaños bien definidos. Dentro de dichos métodos se encuentran los métodos de Block Matching (BM) los cuales tratan de estimar la disparidad en un punto en una imagen comparando una pequeña región alrededor de dicho punto, con una serie de pequeñas regiones

extraídas de la otra imagen. Suelen usar la restricción epipolar para reducir el área de búsqueda a una dimensión. Para estimar la disparidad hacen uso de medidas de semejanza, las cuales, entre las más comunes se encuentran las siguientes:

Suma de diferencias cuadráticas (SSD)

$$SSD = \sum_{u,v} (I_L(u, v) - I_R(u + d, v))^2 \quad (3.22)$$

Suma de diferencias absolutas (SAD)

$$SAD = \sum_{u,v} |I_L(u, v) - I_R(u + d, v)|^2 \quad (3.23)$$

Correlación cruzada normalizada (NCC)

$$NCC = \frac{\sum_{u,v} (I_L(u, v) - \bar{I}_L)(I_R(u, v) - \bar{I}_R)}{\sqrt{\sum_{u,v} (I_L(u, v) - \bar{I}_L)^2 (I_R(u, v) - \bar{I}_R)^2}} \quad (3.24)$$

donde I_L e I_R son los valores de intensidad de las imágenes izquierda y derecha respectivamente, $\bar{I}_L$ e $\bar{I}_R$ son los valores de intensidad promedio en toda la ventana.

Cuando el rango de disparidades es muy grande, los algoritmos mencionados pueden tener problemas en encontrar la correspondencia correcta al aumentar el número de posibles candidatos. Generalmente este problema se presenta al tener una mayor separación entre las cámaras por lo que las características de los puntos (color, iluminación, etc.) pueden haber cambiado y la búsqueda de correspondientes, fallaría. Este mismo efecto lo encontramos en aplicaciones en donde, en lugar de usar un par estéreo, se utilizan tramas subsecuentes en secuencias de video. Las disparidades fuera del rango de búsqueda pueden ser debidas a objetos con velocidades muy altas y que se buscan dentro de la escena.

3.6. Estimación de mapas de disparidad usando funciones wavelet

La teoría de las wavelet casi no ha sido explorada en aplicaciones de visión estéreo [28]. Algunos trabajos han sido reportados en la aplicación de la teoría de wavelet abordando el problema de la correspondencia. El primer trabajo presentado en el que se usa el concepto de la teoría de wavelets para su uso en semejanza entre imágenes se realizó en [29], [30]. Dicho trabajo se basa en el cruce por cero de la transformada wavelet para buscar la correspondencia en pares de imágenes. En [29] y [30], también se explora la descomposición de señales en formas de onda lineales y distribuciones de energía de la señal en el plano del tiempo-frecuencia. En Unser [49], se aborda el concepto de multi-resolución (estrategia

fino-grueso) para el registro de patrones de imagenes usando pirámides de wavelets ortogonales. En Olive-Deubler-Boulin [31], se introduce el método de block matching usando transformadas wavelet ortogonales, además en Zhou [50] se realiza la semejanza de imagenes usando wavelets Haar ortogonales. Las wavelets Haar son unas de las primeras y mas simples wavelets y poseen propiedades básicas en términos de suavizado y orden de aproximación, pero no están bien adaptadas para la mayoría de las aplicaciones con imágenes, en especial en la estimación del problema de la correspondencia. El uso de la teoría de wavelets basado en analisis multiresolución para semejanza entre imágenes fué presentado por He-Pan [32]. En dicho trabajo, se toma la teoría de wavelet introduciendo un algoritmo de cálculo de estéreo semejanza usando wavelets complejas. También exploró diferentes propiedades de las wavelets que pueden ser usadas y adaptadas al problema de estereo semejanza. El uso de wavelets reales y complejas que pueden ser usadas en forma de pirámide de wavelets es tambien llamado árbol de filtros wavelet diádicos de Mallat (MDWFT) [33]. Magarey [34] presenta el uso de medidas de similitud y filtros de cuadratura de Gabor como algoritmos de estimación de movimiento y de semejanza de imágenes.

La teoría wavelet está basada en ecuaciones de dilación como:

$$\phi(t) = \sum_h c_h \phi(M_t - h) \quad (3.25)$$

$$\psi(t) = \sum_h w_h \phi(M_t - h) \quad (3.26)$$

donde c_h y w_h representan los coeficientes wavelet y de escalamiento, además, M representa la banda del banco de filtros [35]. Adicionalmente, la multiresolución puede ser generada no solo en el contexto escalar sino tambien en el caso vectorial, donde existe mas de una función de escala y una wavelet. Las bases de las multiwavelets son caracterizadas por funciones de escalamiento y wavelets en contraste con una función de escalamiento y una wavelet.

En el caso de las multiwavelets, las funciones de escalamiento satisfacen la ecuación de la matriz de dilatación.

$$\Theta(t) = \sum_h c_h \Theta(M_t - h), \quad (3.27)$$

de la misma manera, para las multiwavelets, la ecuación de la matriz de dilatación es de la forma:

$$\Psi(t) = \sum_h c_h \Psi(M_t - h). \quad (3.28)$$

En la ecuaciones anteriores, c_h y w_h son matrices reales de los coeficientes de los multifiltros donde Θ y Ψ pueden ser expresados en términos de funciones de escalamiento r y wavelets r como:

$$\Theta(t) = \begin{pmatrix} \theta_0(t) \\ \theta_1(t) \\ \vdots \\ \theta_{r-1}(t) \end{pmatrix} \quad (3.29)$$

y

$$\Psi(t) = \begin{pmatrix} \psi_0(t) \\ \psi_1(t) \\ \vdots \\ \psi_{r-1}(t) \end{pmatrix} \quad (3.30)$$

La transformación wavelet (WT) produce representaciones escala-espacio de la señal de entrada generando versiones escaladas en espacios de aproximaciones y detalles, teniendo la propiedad de $A_{s-1} = A_s \oplus D_s$, donde A_s y D_s representan el espacio de aproximaciones y de detalles a menor resolución.

El uso de los filtros diádicos de Mallat [33] resultan en tres componentes en el espacio de detalles, los cuales son el horizontal, vertical y diagonal. La siguiente figura muestra lo anterior, donde C y W representan el filtro pasa bajas y filtro pasa altas de acuerdo con las funciones de escalamiento y los coeficientes wavelet, respectivamente.

En Batthi [28], se propone el uso de los filtros diádicos de Mallat con el módulo de la transformada wavelet. El módulo de la transformada wavelet, como representación general de vector, puede ser expresado como:

$$WTM_{s,k} = W_{s,k} \angle \Theta_{W_{s,k}}, \quad (3.31)$$

donde $W_{s,k}$ es:

$$W_{s,k} = \sqrt{|D_{h,s,k}|^2 + |D_{v,s,k}|^2}, \quad (3.32)$$

en la cual, $D_{h,s,k}$ y $D_{v,s,k}$ son el k -ésimo componente de detalles horizontal y vertical a la escala s . Además, $\Theta_{W_{s,k}}$ puede ser expresado como:

$$\Theta_{W_{s,k}} = \begin{cases} \alpha(s, k) & \text{if } D_{h,s,k} > 0 \\ \pi - \alpha(s, k) & \text{if } D_{h,s,k} < 0 \end{cases} \quad (3.33)$$

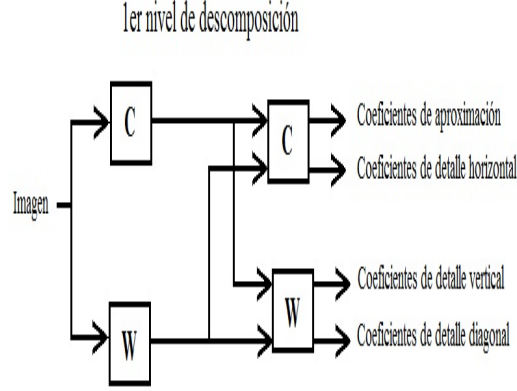


Figura 3.3: Banco de filtros wavelet de Mallat.

donde

$$\alpha(s, k) = \tan^{-1}\left(\frac{D_{v,s,k}}{D_{h,s,k}}\right) \quad (3.34)$$

Antes de incluir el bloque de medición de similitud, se realiza la normalización de escala wavelet (WSN), lo cual ayuda a minimizar el efecto de variación de iluminación que pudiera existir en las respectivas vistas:

$$NW_{s,k} = \frac{D_{d,c,s,k}}{|A_{s,k}|}, \text{ donde } \forall dc \in \{h, v, d\} \quad (3.35)$$

Después del cálculo del módulo máximo de la transformada wavelet, la correlación es usada como algoritmo de similitud y con ello se obtiene el mapa de disparidad inicial. La aplicación multi ventana se usa para mejorar el rendimiento de la correlación basada en similitud. El indicador de correlación multiventana, es definido por:

$$NC(s, k) = \overline{NC_{s,k,W_0}} + \sum_{j=1}^{n_W} \frac{n_W}{2} \quad (3.36)$$

donde NC_{s,k,W_0} representa la correlación normalizada con respecto a la ventana central, NC_{s,k,W_j} define la correlación normalizada con respecto a la j -ésima ventana con n_W como número de ventanas deslizantes. Finalmente, el algoritmo usa una medida de pesos probabilísticos basada en selección de candidatos en espacios de búsqueda bien definidos, además se seleccionan mediante una bandera a los candidatos sin ambigüedades. En cada

nivel de procesamiento se realiza una interpolación de escala y la disparidad final se obtiene de acuerdo a la siguiente ecuación:

$$d_F = d_L + 2d_c \quad (3.37)$$

donde d_L es la disparidad local obtenida en cada nivel de la pirámide. Este proceso se repite hasta subir todos los niveles de procesamiento.

3.7. Propuesta para la estimación de mapas de disparidad

En este trabajo de tesis, se propone un algoritmo eficiente para construir video secuencias con percepción tridimensional basados en video secuencias en dos dimensiones adquiridas por cámaras de video convencionales. El algoritmo hace uso de las wavelets basadas en funciones atómicas (WAF) para estimar el mapa de disparidad de imágenes subsecuentes en una video secuencia. Dichas WAFs son implementadas en la estrategia fino-gruesa (multi wavelet) para potenciar la eficiencia de los resultados. Hace uso de la técnica de anáglifo para desplegar imágenes con percepción en tres dimensiones, en desplegados convencionales. La principal diferencia de éste algoritmo con otros presentados aquí, es que no requiere operaciones computacionales intensas para poder obtener buena percepción tridimensional y puede generar videos sin la necesidad de hacer uso de hardware adicional.

El uso de las WAFs es debido a que presentan mejores resultados que las wavelets de las familias clásicas en aplicaciones como: reconocimeinto en imágenes médicas, reconocimiento de hablantes, procesamiento de imágenes, y otros [36], [37].

El algoritmo propuesto contiene las siguientes etapas: descomposición de la video secuencia en 2D, separación de componentes en formato RGB de cada imagen, cálculo del mapa de disparidad usando multi-WAF, mejoramiento del mapa de disparidad por medio de la compresion de su rango dinámico, síntesis del anáglifo usando la interpolación del vecino más cercano (NNI), y finalmente la reconstrucción del video en 3D y su visualización. La siguiente imagen muestra el algoritmo propuesto.

Para obtener la estimación del mapa de disparidad, es implementada la estimación de correspondencia estéreo basada en la técnica de multiwavelets y consiste en dos etapas: La implemetación de las WAF y el cálculo del módulo de la transformada wavelet (WTM).

Las wavelets basadas en función atómicas usadas en este trabajo corresponden a las familias madre up , fup_n y π_n . La definición de las funciones atómicas está conectada con el siguiente problema matemático: el aislamiento de una función que tiene derivadas con un máximo y mínimo similar a las de la función inicial. Para resolver el problema anterior, se

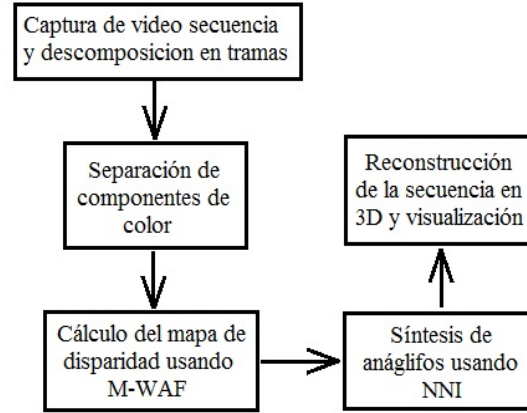


Figura 3.4: Etapas del algoritmo propuesto.

requiere una solución diferenciable infinita de ecuaciones diferenciales con argumento y con corrimiento. En algunos trabajos, se ha demostrado que las funciones atómicas son compactas, y como los polinomios, son universales en términos de sus propiedades de aproximación [38], [39].

La función atómica (AF) más simple e importante es generada por una serie de convoluciones de impulsos rectangulares que son fáciles de analizar usando la transformada de Fourier. Basados en convoluciones $N - a - 1$ de $(N + 1)$ impulsos rectangulares idénticos, la spline compacta $\Theta_N(x)$ puede ser definida como:

$$\Theta_N(x) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} e^{jux} \left(\frac{\sin(\frac{u}{2})}{\frac{u}{2}} \right)^{N+1} du. \quad (3.38)$$

La función $up(x)$ puede ser presentada por la transformada de Fourier de convoluciones infinitas de impulsos rectangulares con duración variable 2^{-k} , como en la siguiente ecuación:

$$up(x) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} e^{jux} \prod_{k=1}^{\infty} \left(\frac{\sin(u \cdot 2^{-k})}{u \cdot 2^{-k}} \right) du \quad (3.39)$$

La función atómica $fup_N(x)$ es definida por la convolución de $\Theta_{N-1}(x)$ y $up(x)$ en el intervalo $[\frac{-(N+2)}{2}, \frac{(N+2)}{2}]$. Esto es, $fup_N(x)$ puede ser escrita de la siguiente forma:

$$fup_N(x) = \int_{-\infty}^{+\infty} e^{jux} \left(\frac{\sin(\frac{u}{2})}{\frac{u}{2}} \right)^N \prod_{k=1}^{\infty} \left(\frac{\sin(u \cdot 2^{-k})}{u \cdot 2^{-k}} \right) du, \quad (3.40)$$

con $fup_0(x) \equiv up(x)$. La generalización de la AF $up(x)$ se puede escribir como:

$$up_m(x) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} e^{jux} \prod_{k=1}^{\infty} \frac{\sin^2(\frac{mu}{(2m)^k})}{\frac{mu}{(2m)^k} m \sin(\frac{u}{(2m)^k})} du \quad (3.41)$$

con $up_1(x) = up(x)$.

La función $\pi_m(x)$ puede ser representada por la transformada inversa de Fourier $\pi_m(t) = \int_{-\infty}^{+\infty} e^{jux} F_m(t)$; basados en dicha representación para la función $F_m(t)$ tenemos:

$$F_m(t) = \prod_{k=1}^m \frac{\sin(2m-1)t + \sum_{v=2}^m (-1)^v}{\sin(2m-2v+1)t(3m-2)t}. \quad (3.42)$$

Cada AF con norma unitaria posee la siguiente estructura:

$$\psi_{\theta}^p(x) = \frac{\exp(j\pi) h_{\theta}^p(x)}{\|h_{\theta}^p(x)\|}, \quad (3.43)$$

donde la función $h_{\theta}p(x)$ en la ecuación anterior, se puede caracterizar como:

$$h_{\theta}^p(x) = \frac{1}{2^p} \sum_{k=0}^{\frac{p-1}{2}} C_p^k \left(\theta \left(x + \frac{p-2k}{2} \right) + \theta \left(x - \frac{p-2k}{2} \right) \right), \text{pimpares}, \quad (3.44)$$

$$h_{\theta}^p(x) = \frac{1}{2^p} \sum_{k=0}^{\frac{p-1}{2}} C_p^k \left(\theta \left(x + \frac{p-2k}{2} \right) + \theta \left(x - \frac{p-2k}{2} \right) \right) + C_p^{\frac{p}{2}} \theta(x), \text{ppares}. \quad (3.45)$$

Existen diferentes AF presentadas en la literatura tales como up , fup_n , up_n , Ξ_n , π_n .

Una vez establecidas las funciones atómicas, el algoritmo propuesto usa la técnica multiwavelet en forma de banco de filtros para los tres componentes de detalles. Retomando la definición del módulo de la transformada wavelet usado por Bhatti [28], podemos encontrar la definición de la normalización de WTM como:

$$NW_s = \frac{\sqrt{|D_{hs}|^2 + |D_{vs}|^2 |D_{ds}|^2}}{|A_s|} \quad (3.46)$$

donde W_s es el módulo wavelet para los diferentes niveles de descomposición s , D_{hs} , D_{vs} , D_{ds} son los detalles horizontales, verticales y diagonales en cada nivel s , A_s es el componente de aproximación.

Una vez que WTM_s es calculado por cada par de tramas, el mapa de disparidad perteneciente a cada nivel de descomposición es formado por medio de la correlación cruzada normalizada:

$$NCC_s(x, y) = \sum_{i, j \in W} \frac{WTM_l(i, j) \cdot WTM_r(x + i, y + j)}{\sqrt{\sum_{i, j \in W} WTM_l^2(i, j) \cdot \sum_{i, j \in W} WTM_r^2(x + i, y + j)}} \quad (3.47)$$

donde WTM_l y WTM_r representan la transformada wavelet de la imagen izquierda y derecha, respectivamente, en cada nivel de descomposición s .

La disparidad final de cada nivel de descomposición se calcula por medio de la interpolación del vecino más cercano (NNI). Para esta aplicación, proponemos cuatro niveles de descomposición. La siguiente figura muestra el algoritmo propuesto para obtener el mapa de disparidad final.

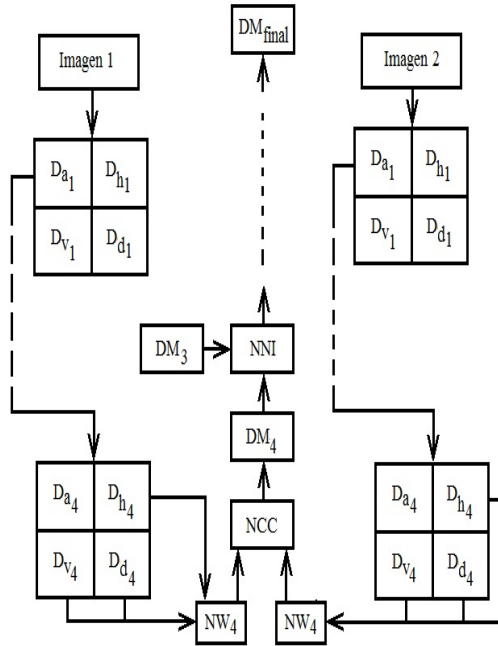


Figura 3.5: Etapas para la obtención de la disparidad final.

El parámetro de la profundidad representa información útil en el caso de la navegación robótica, la reconstrucción de imágenes médicas, en imágenes multidimensionales, imágenes de radar, tele presencia, entrenamiento de la percepción espacial, entre otras. Como establecimos en el Capítulo 1, los métodos clásicos usados en la construcción de anáglifos pueden producir efectos de fantasma y pérdidas de color. Una forma de reducir éstos desperfectos es el uso de la compresión del rango dinámico del mapa de disparidad. La compresión del rango dinámico permite la retención de la información de profundidad [12], lo cual reduce los efectos de fantasma en áreas donde no existe solapamiento. El algoritmo propuesto incluye también una etapa de mejoramiento de anáglifos y ésta se basa en la aplicación de la ecuación 2.11.

En la etapa final, la síntesis de anáglifos es realizada usando el mapa de disparidad mejorado. Para generar el anáglifo, se toman tramas subsecuentes y se interpolan con una rejilla de control conformada por el mapa de disparidad. Durante las simulaciones, se implementaron la interpolación bilineal, sinc y del vecino más cercano con el objetivo de obtener la mejor percepción tridimensional. La interpolación del vecino más cercano (NNI) mostró un mejor rendimiento visual en la obtención de los anáglifos.

Para el caso de la interpolación del vecino mas cercano; definiendo $X = f(x, y)$ y $Y = g(x, y)$ como coordenadas de puntos en la imagen de referencia y las cooredenadas de esos puntos correspondientes en la imagen sensada. Dados las coordenadas del punto (x, y) en la imagen de referencia, las definiciones presentadas anteriormente determinan las coordenadas (X, Y) del punto correspondiente en la imagen sensada. Tomando en cuenta la intensidad en (X, Y) en la imagen sensada y almacenando (x, y) en una nueva imagen, la imagen sensada es remuestreada punto a punto a la geometría de la imagen de referencia [40]. Usando la *NNI* dadas las coordenadas (X, Y) en un punto, los vecinos en una ventana recatangular dada por los píxeles (u, v) , $(u, v+1)$, $(u+1, v)$ y $(u+1, v+1)$ contienen el punto (X, Y) , A lo largo de los cuatro píxeles, el más cercano a las coordenadas (X, Y) es tomado como intensidad en (X, Y) . La siguiente figura muestra lo anterior.

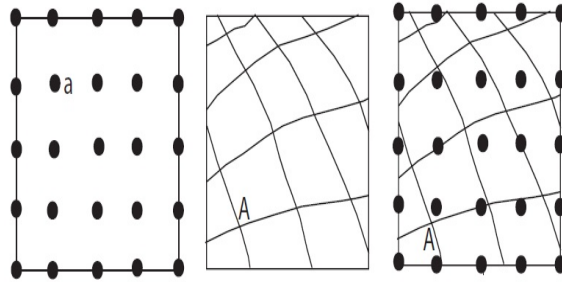


Figura 3.6: Proceso de remuestreo. La primer imagen muestra la imagen de referencia, la segunda muestra la imagen sensada y la tercer imagen muestra el encimamiento de las dos primeras, en donde la intensidad en A en la imagen de referencia es estimada de las intensidades de una ventana de vecinos cercanos a A en la imagen sensada.

3.8. Conclusiones del capítulo

Se investigaron diferentes algoritmos y técnicas para extraer información de disparidad en un par estéreo. Existen una gran cantidad de ellos y se basan en diferentes características en las imágenes, van desde el cálculo de vectores de movimiento en escenas, algoritmos

híbridos que hacen uso de las estrategias fino-grueso para estimar de mejor forma el mapa de disparidad, algoritmos basados en minimización de energía sobre regiones, algoritmos basados en semejanza entre píxeles y los más novedosos que son los algoritmos que usan transformadas wavelet con la finalidad de mejorar la estimación del mapa de disparidad.

En el caso de los algoritmos que calculan la disparidad haciendo uso de vectores de movimiento, encontramos que éstos algoritmos tienen problemas cuando las imágenes de entrada tienen cambios de intensidad bruscos, tienen oclusiones y se mezclan diferentes velocidades en objetos que forman la escena.

El algoritmo híbrido presentado aquí, se basa en una pirámide gaussiana para encontrar versiones gruesas del problema a resolver, siendo la version mas gruesa, la inicialización de la version más fina. Lo anterior conlleva a un costo computacional elevado, que merma el comportamiento del algoritmo en aplicaciones en donde la velocidad de procesamiento es importante.

Cuando se estudió el algoritmo de semejanza estéreo en regiones, pudimos observar que el algortimo conlleva una serie de pasos los cuales están enfocados a el mejoramiento en la estimación del mapa de disparidad, esto tambien merma en la velocidad de procesamiento del mismo.

Se estudió la forma en cómo aplicar las transformadas wavelet clásicas en la obtención del mapa de disparidad y se presentó un algoritmo el cual utiliza las multi-wavelets y las medidas de semejanza para mejorar la estimación del mapa de disparidad. Pudimos ver que las wavelets tienen poco uso en problemas de estimación de parámetros en imágenes.

Se presentó el algoritmo propuesto, el cual está basado en la técnica de multi-wavelet y se complementa con el uso de las funciones atómicas, las cuales han sido probadas en diferentes aplicaciones en el procesamiento de imágenes y video y reportadas en la literatura. El algoritmo también hace uso de medidas de similitud y la generación de anáglifos para poder obtener videos en 3D.

Capítulo 4

Simulación de las técnicas existentes y el método propuesto

4.1. Imágenes y videos utilizados

Para la realización de las simulaciones se utilizaron imágenes sintéticas y que son ampliamente citadas en la literatura debido a sus diferentes características. Las imágenes de prueba pueden ser obtenidas ingresando en la siguiente dirección electrónica <http://vision.middlebury.edu/stereo/dataset/> [43]. Las imágenes usadas son: *Aloe*, *Venus*, *Lampshade1*, *Wood1*, *Bowling1* y *Reindeer*; dichas imágenes están en formato *PNG* con un tamaño de 480×720 píxeles.

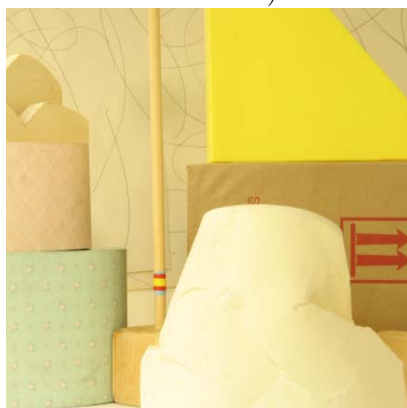
También se usaron video secuencias de prueba ampliamente citadas en la literatura. Los videos de prueba pueden ser obtenidos en la siguiente página electrónica: <http://trace.eas.asu.edu/yuv/index.html> [44]. Las secuencias de video usadas son: *Coastguard*, *Flowers* y *Foreman*; están en formato *AVI* y se usaron 300 tramas con un tamaño de 480×720 píxeles. Adicionalmente, se grabaron dos video secuencias (*VideoTest1* y *VideoTest2*) en ambientes reales para probar el algoritmo propuesto en condiciones normales de operación. Las secuencias de prueba reales contienen 200 secuencias con un tamaño de 480×720 píxeles.



a)



b)



c)



d)



e)



f)

Figura 4.1: Imágenes a color sintéticas utilizadas: a) Aloe, b) Venus, c) Lampshade1, d) Wood1, e) Bowling1, f) Reindeer1

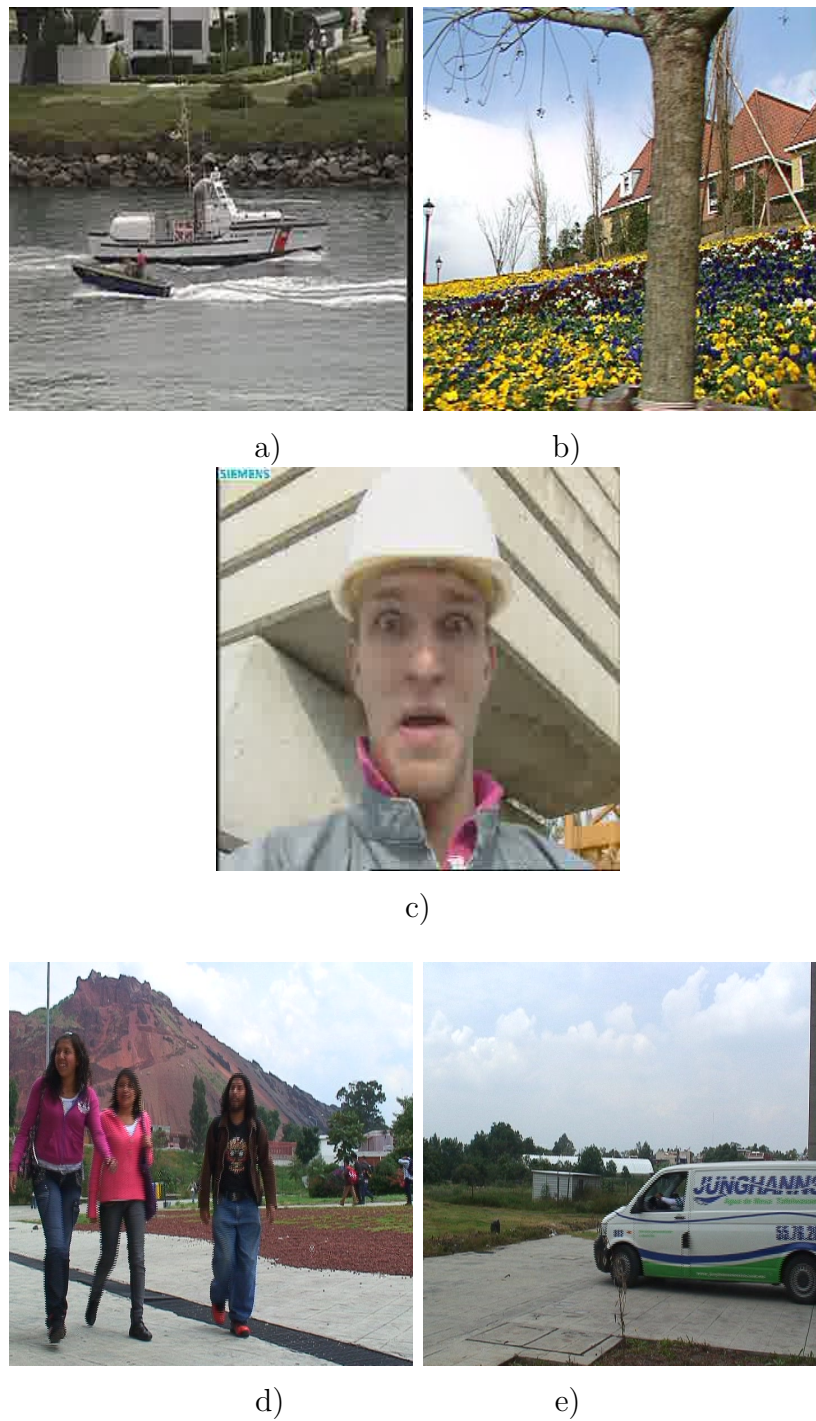


Figura 4.2: Videos a color sintéticos y reales utilizados: a)Coastguard, b) Flowers, c) Foreman, d) VideoTest1, e)VideoTest2

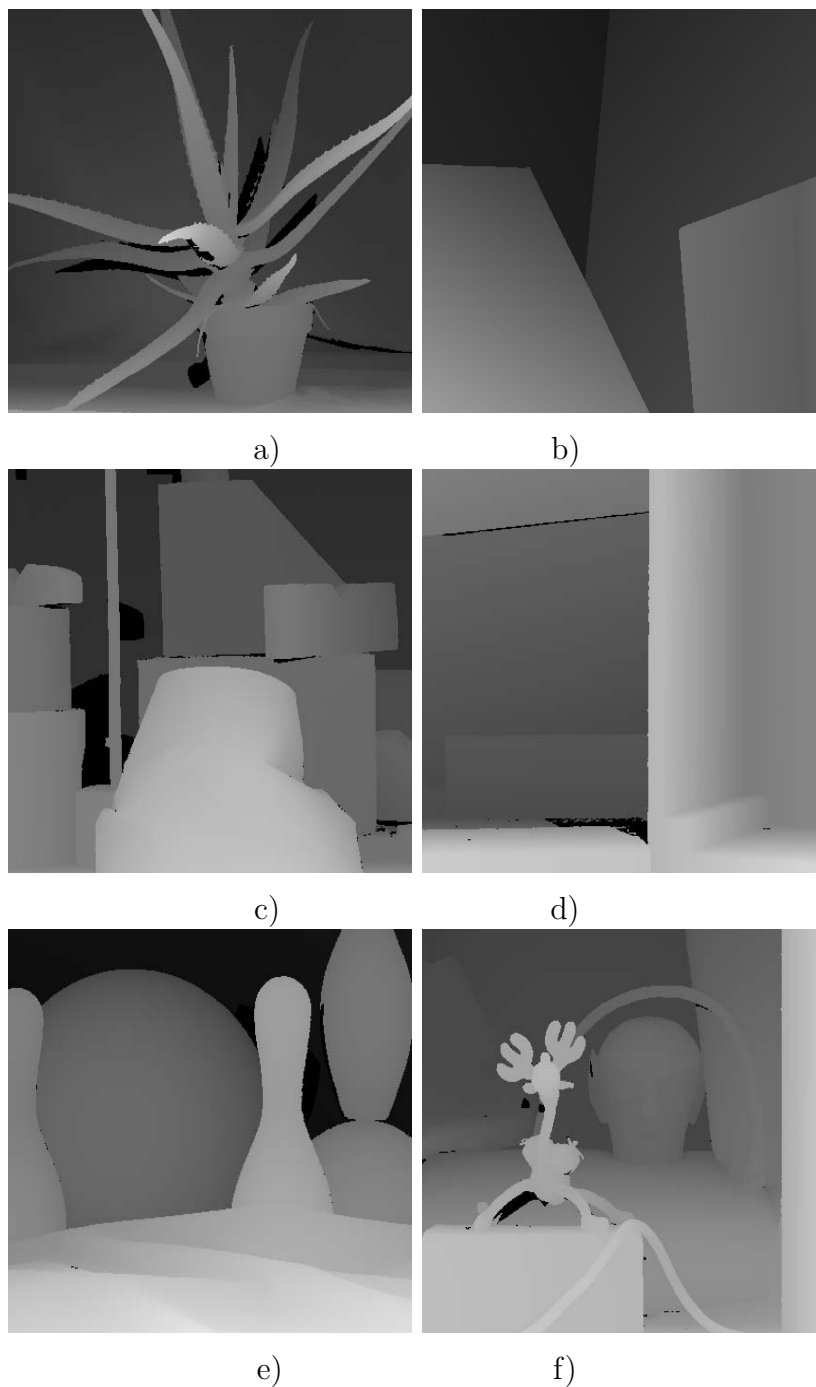


Figura 4.3: Ground truth de las imágenes a color sintéticas utilizadas: a)Aloe, b) Venus, c) Lampshade1, d)Wood1, e)Bowling1, f)Reindeer1

Recordando el algoritmo propuesto y que se presenta en el capítulo 3, en el caso del

bloque correspondiente al cálculo del mapa de disparidad, se implementaron las siguientes técnicas: Lucas y Kanade, warping, semejanza estéreo basadas en regiones, wavelets clásicas y la propuesta descrita en éste trabajo. Dichas implementaciones corresponden a poder comparar el rendimiento del algoritmo dadas diferentes técnicas para calcular el mapa de disparidad. Antes de presentar los resultados obtenidos, presentaremos los *ground truth*, dichas imágenes están compuestas de los elementos que respresentan las disparidades de cada una de las imágenes sintéticas. El *ground truth* nos sirve como criterio visual cualitativo de cada uno de los algoritmos implementados. La figura 4.3 muestra los *ground truth* de cada una de las imágenes sintéticas.

4.2. Criterios objetivos de valoración de resultados

En esta sección se presentan los criterios objetivos de valoración de resultados utilizados. Dichos criterios objetivos son: Porcentaje de disparidades mal seleccionados *QBD* y la medida de similitud en imágenes (*SSIM*).

Para el caso del porcentaje de disparidades mal seleccionadas, la ecuación utilizada se presenta a continuación:

$$QBD = \frac{1}{N} \sum_{x,y} |d_E(x,y) - d_G(x,y)|^2 \quad (4.1)$$

en donde N representa el número total de pixeles de la imagen de entrada, d_E y d_G representan la disparidad estimada y la disparidad original ó ground truth, respectivamante.

Para el caso de la medida de similitud en imágenes *SSIM*, la ecuación utilizada para obtener este criterio se define como:

$$SSIM(x,y) = [l(x,y)] \cdot [c(x,y)] \cdot [s(x,y)], \quad (4.2)$$

en donde los parámetro l , c y s se pueden definir mediante las siguientes ecuaciones:

$$l(x,y) = \frac{2\mu_X(x,y)\mu_Y(x,y) + C_1}{\mu_X^2(x,y) + \mu_Y^2(x,y) + C_1}, \quad (4.3)$$

$$c(x,y) = \frac{2\sigma_X(x,y)\sigma_Y(x,y) + C_2}{\sigma_X^2(x,y) + \sigma_Y^2(x,y) + C_2}, \quad (4.4)$$

$$s(x,y) = \frac{\sigma_{XY}(x,y) + C_3}{\sigma_X(x,y) + \sigma_Y(x,y) + C_3}. \quad (4.5)$$

en las cuales, X representa la imagen estimada, Y es el ground truth, μ y σ son la media y la desviación estándar de las imágenes X y Y respectivamente; en nuestra aplicación $C_1 = C_2 = C_3 = 1$.

En este caso, $l(x, y)$ es la medida de similitud correspondiente a la luminancia, $c(x, y)$ está dedicado a medir el contraste y finalmente, $s(x, y)$ la similitud en las estructuras que componen a el par de imágenes a medir.

El uso de la medida SSIM en este trabajo está definido a partir de las siguientes premisas [41]:

- La fidelidad de la señal es independiente a la relación espacial ó temporal entre las muestras de la señal original. En otras palabras, si la señal original y la señal distorsionada son aleatoriamente ordenadas, entonces el error cuadrático medio (MSE) ó cualquier medida basada en l_p (Como lo son la relación señal a ruido pico PSNR y otras) deberá ser la misma.
- La fidelidad de la señal es independiente a cualquier relación entre la señal original y la señal de error. Dada una señal de error, el MSE se mantiene sin cambios, sin importar qué señal original ha sido sumada.
- La fidelidad de la señal es independiente a los cambios de signo.
- Todos las muestras de la señal son de la misma manera importantes en la finalidad de la señal.

Desafortunadamente ninguno de los puntos se cumple en el contexto de la medición de la percepción visual de la fidelidad de la imagen. Dado lo anterior, la métrica SSIM está orientada a tomar en cuenta dentro de las características a medir, la luminancia, el contraste y la estructura [42].

4.3. Resultados obtenidos en las simulaciones

4.3.1. Resultados obtenidos en la estimación del mapa de disparidad

En esta sección, presentaremos los resultados visuales obtenidos para el mapa de disparidad usando diferentes algoritmos y diferentes imágenes sintéticas.

En la implementación del algoritmo de Lucas y Kanade se utilizó un tamaño de ventana de procesamiento de 5×5 y con $k = 5$. Los resultados obtenidos se presentan en la figura 4.4. Para cada una de las imágenes, se ha aplicado el mejoramiento del mapa de disparidad con los siguientes parámetros: $a = P = 0.5$.

Así mismo, para el algoritmo de warping, $\alpha = 30$, $\beta = 80$, $k = 50$ y se implementó para 5 niveles en la pirámide de procesamiento. Los resultados obtenidos se muestran en la

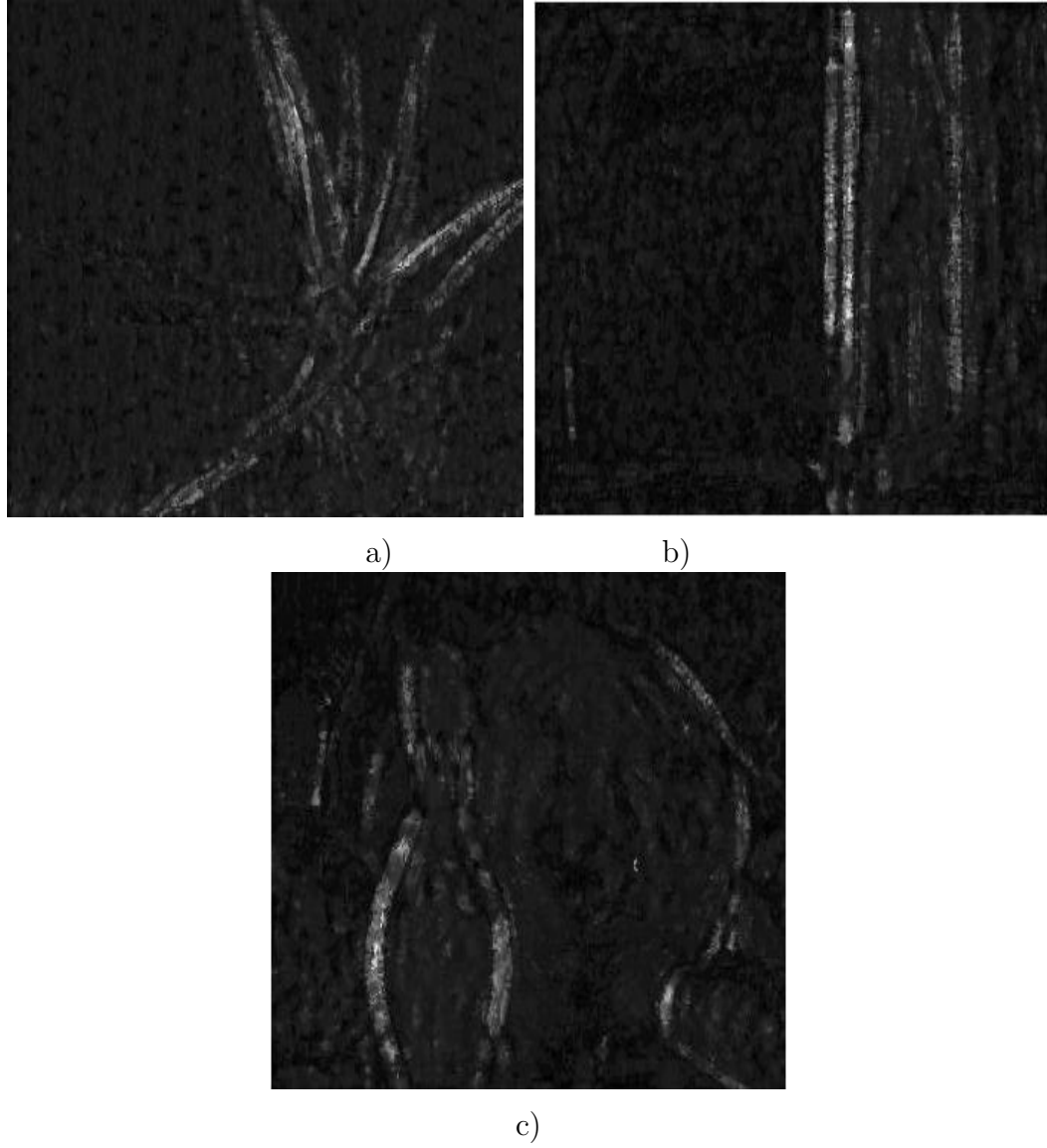


Figura 4.4: Mapas de disparidad obtenidos mediante la técnica de Lucas and Kanade para: a) Aloe, b) Wood1, c) Bowling1

figura 4.5. Para cada una de las imágenes, se ha aplicado el mejoramiento de el mapa de disparidad con los siguientes parámetros: $a = P = 0.5$.

Los parámetros usados para la implementación del algoritmo de semejanza estéreo son $\alpha = 1$, con un umbral de 20, foco de 30. Los resultados obtenidos se muestran en la figura 4.6. Para cada una de las imágenes, se ha aplicado el mejoramiento del mapa de disparidad con los siguientes parámetros: $a = P = 0.5$.

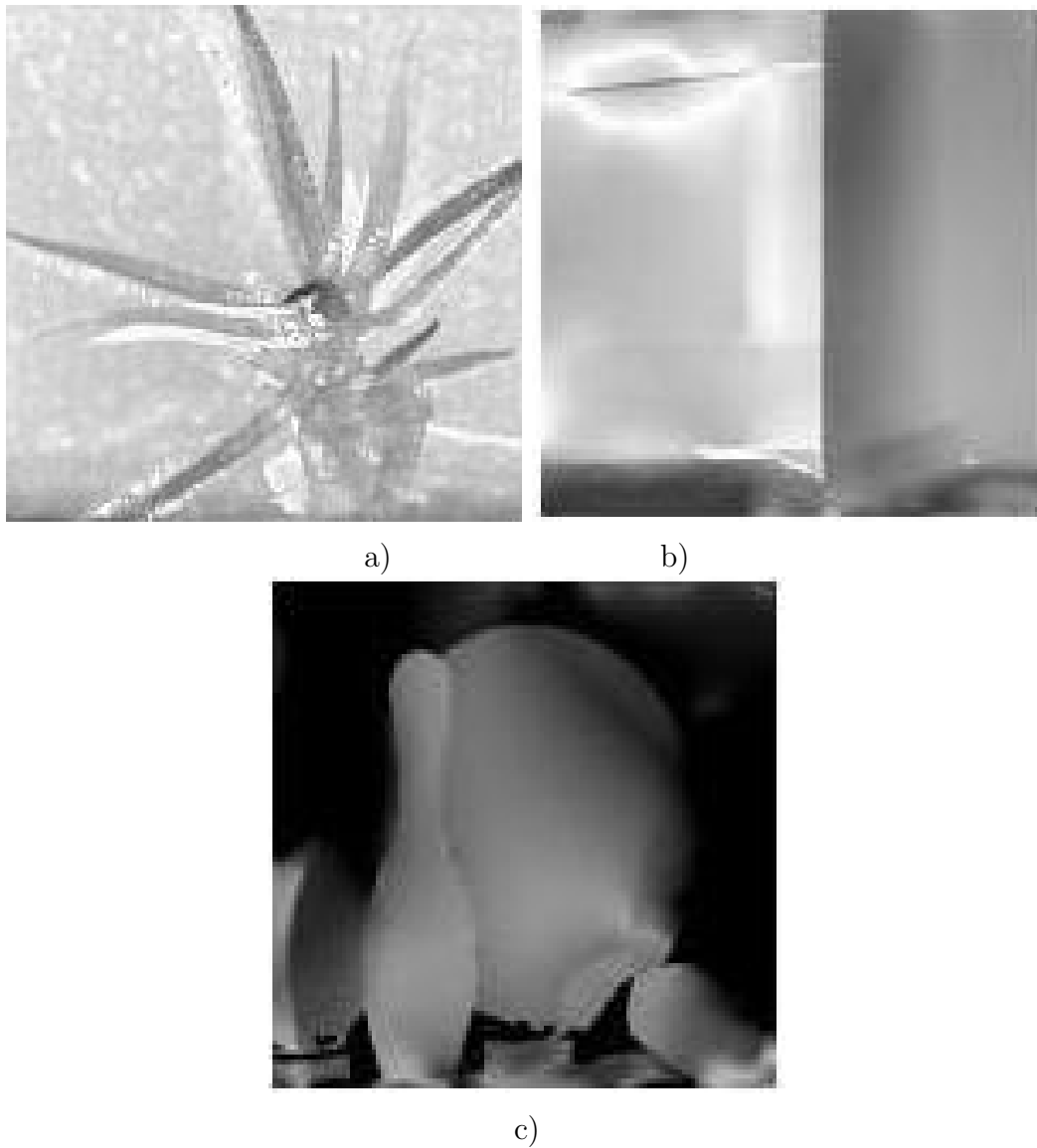


Figura 4.5: Mapas de disparidad obtenidos mediante la técnica de warping para: a) Aloe, b) Wood1, c) Bowling1

Para el caso del algoritmo implementado para semejanza entre píxeles, se ha seleccionado el basado en la suma cuadrática de distancias (SSD). La ventana de procesamiento es de 9×9 en un rango de búsqueda de 20 píxeles con una disparidad máxima de 14 píxeles. Los resultados obtenidos se muestran en la figura 4.7. Para cada una de las imágenes, se ha aplicado el mejoramiento de el mapa de disparidad con los siguientes parámetros: $a = 0.2, P = 0.5$.

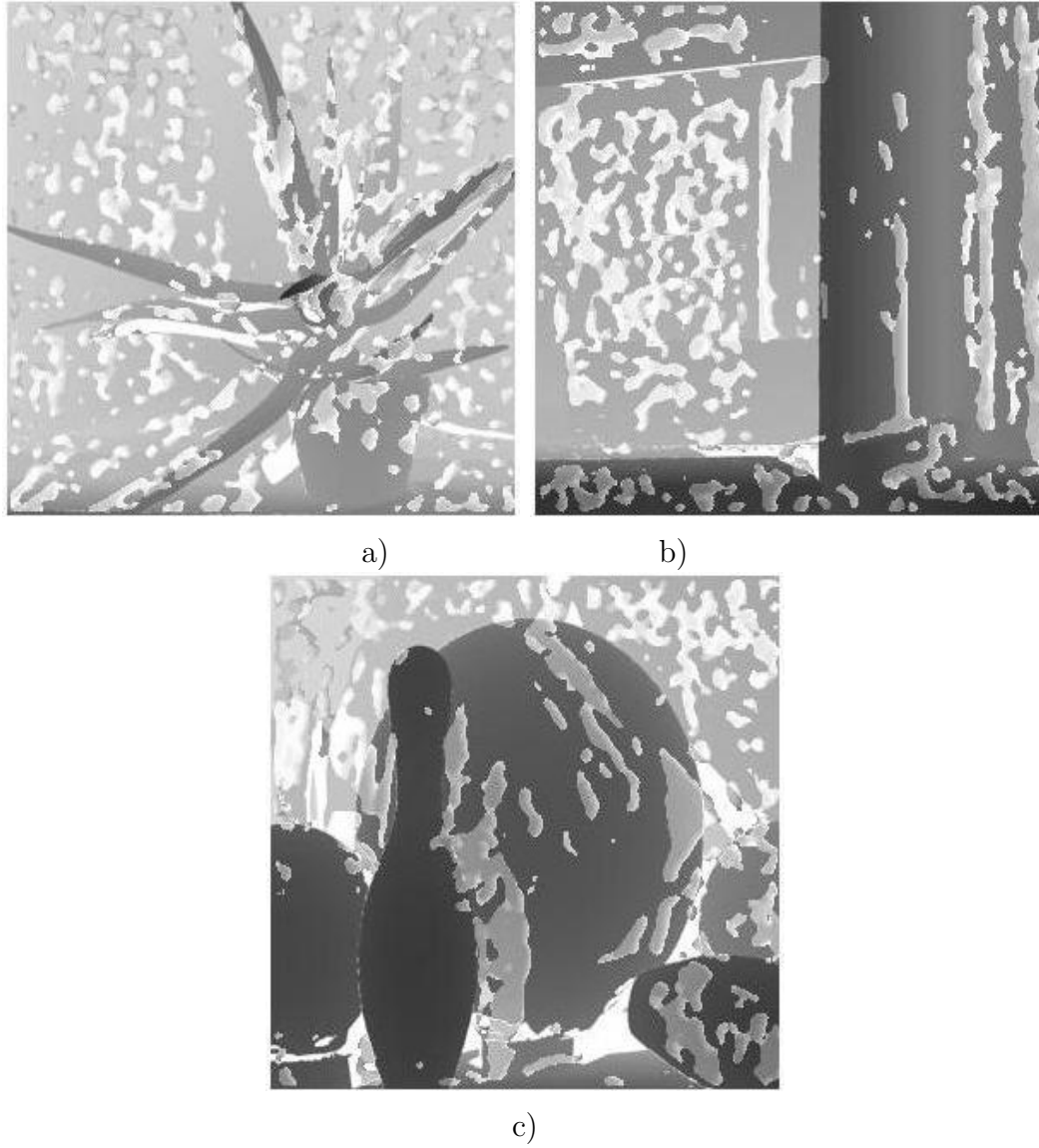


Figura 4.6: Mapas de disparidad obtenidos mediante la técnica de semejanza estéreo para: a) Aloe, b) Wood1, c) Bowling1

El siguiente algoritmo presentado es el correspondiente a la wavelets con familias clásicas, en donde se probaron diferentes familias como Daubechies, Coiflets y Harr. Sólo se presentan los resultados visuales obtenidos para la correspondiente a *Coiflets2*. Para encontrar el mapa de disparidad final, se usa la correlación cruzada normalizada con una ventana de procesamiento de 5×5 y un rango de búsqueda de 20 píxeles. Los resultados obtenidos se muestran en la figura 4.8. Para cada una de las imágenes, se ha aplicado el mejoramiento

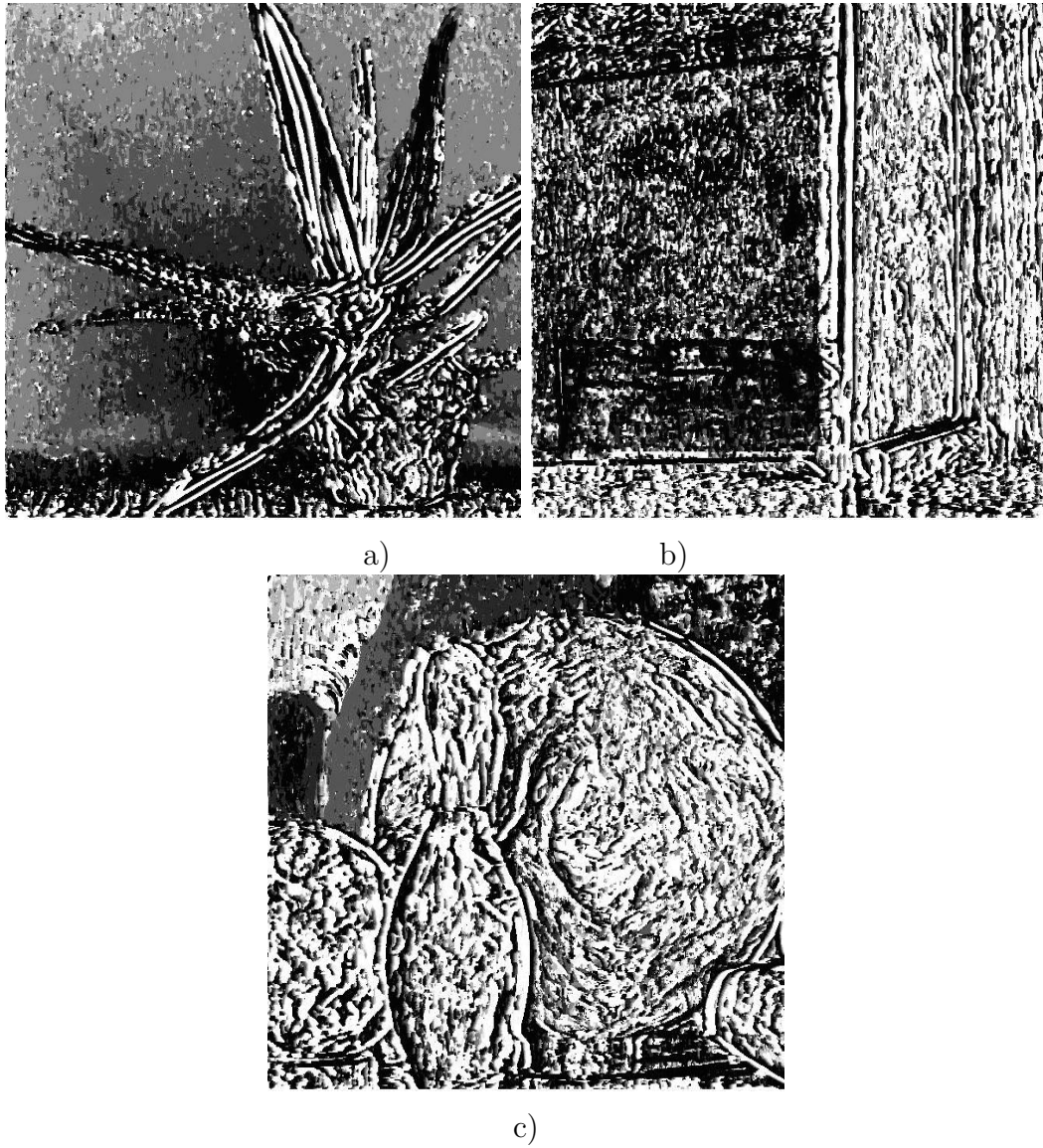


Figura 4.7: Mapas de disparidad obtenidos mediante la técnica de semejanza estéreo para: a) Aloe, b) Wood1, c) Bowling1

del mapa de disparidad con los siguientes parámetros: $a = P = 0.5$.

Además el algoritmo usado con familia de wavelets clásicas se modificó para poder usarlo con estrategia fino-grueso, con 4 niveles de descomposición. Se implementaron las simulaciones con las familias antes mencionadas. Sólo se presentan los resultados visuales obtenidos para la correspondiente a *Biortogonal*6.8. Para encontrar el mapa de disparidad final, se usa la correlación cruzada normalizada con una ventana de procesamiento de 5×5

y un rango de búsqueda de 20 píxeles para cada uno de los niveles de descomposición. Para cada una de las imágenes, se ha aplicado el mejoramiento del mapa de disparidad con los siguientes parámetros: $a = P = 0.5$. Los resultados obtenidos se muestran en la figura 4.9.

Finalmente, presentamos los resultados para los mapas de disparidad usando el algoritmo propuesto. En dicho algoritmo, primero se realizaron pruebas con un nivel de descomposición usando las familias de wavelets basadas en funciones atómicas correspondientes a up , fup_m

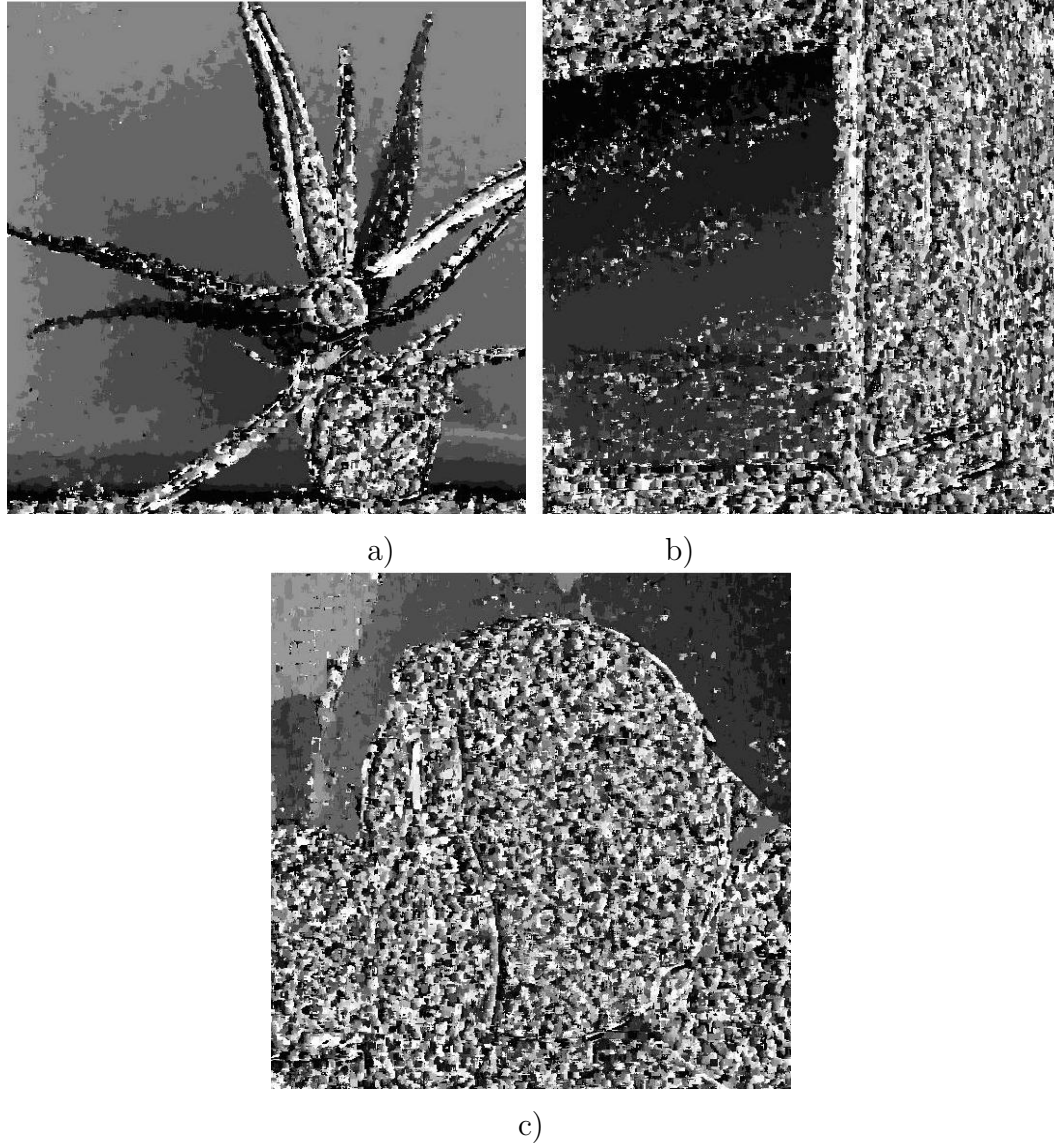


Figura 4.8: Mapas de disparidad obtenidos mediante la técnica de wavelets con *Coiflets2* para: a) Aloe, b) Wood1, c) Bowling1

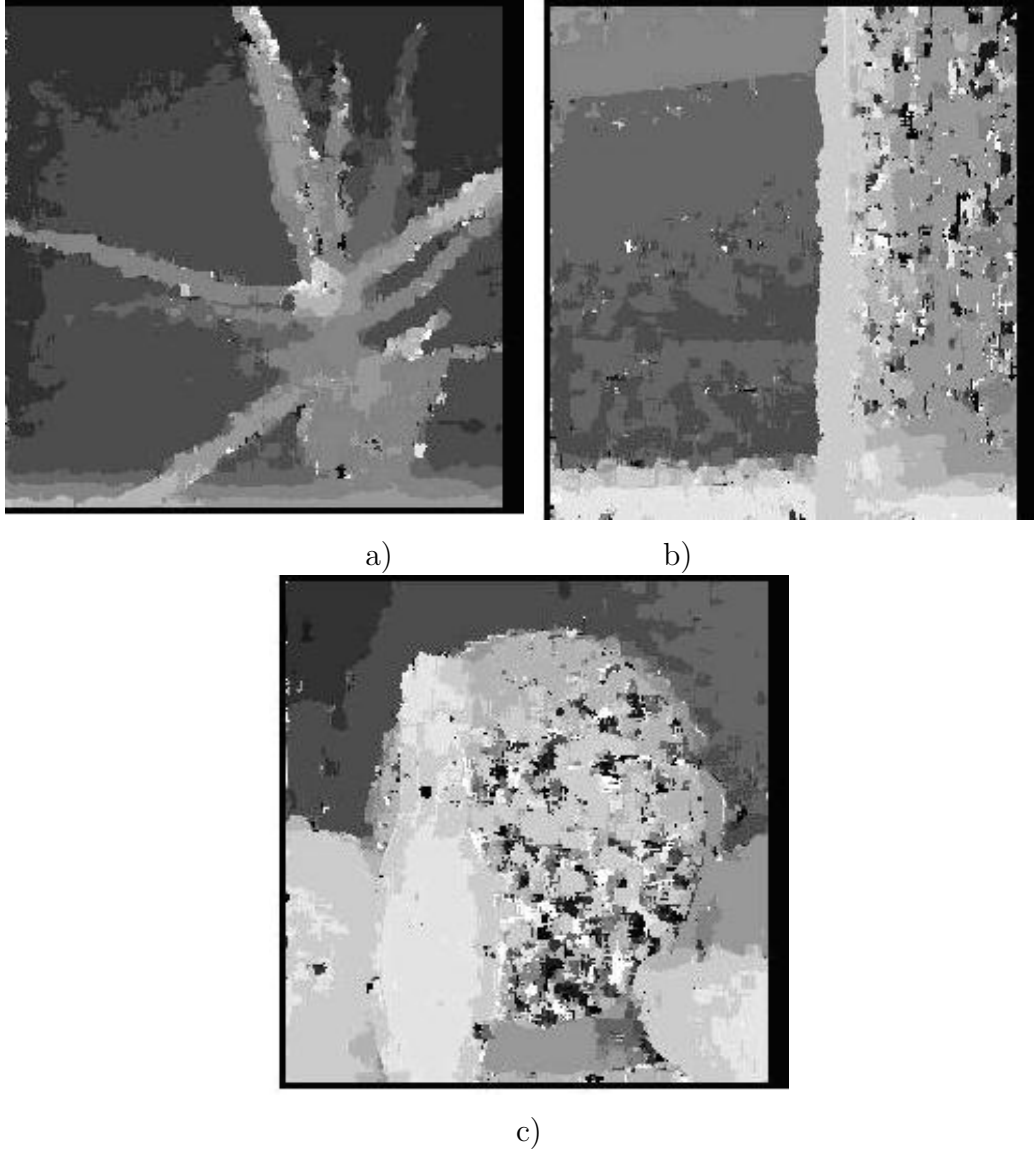


Figura 4.9: Mapas de disparidad obtenidos mediante la técnica de multi-wavelets con *Biortogonal6.8* para: a) Aloe, b) Wood1, c) Bowling1

y pi_m . Para encontrar el mapa de disparidad final, se usa la correlación cruzada normalizada con una ventana de procesamiento de 5×5 y un rango de búsqueda de 20 píxeles para cada uno de los niveles de descomposición. Para cada una de las imágenes, se ha aplicado el mejoramiento del mapa de disparidad con los siguientes parámetros: $a = P = 0.5$. Los resultados obtenidos, en el caso de la implementación con un solo nivel de descomposición se muestran en la figura 4.10. Para el caso de la aplicación con 4 niveles de descomposición

la figura 4.11 presenta los resultados. Sólo se presentan los resultados visuales obtenidos para la correspondiente a π_6 .

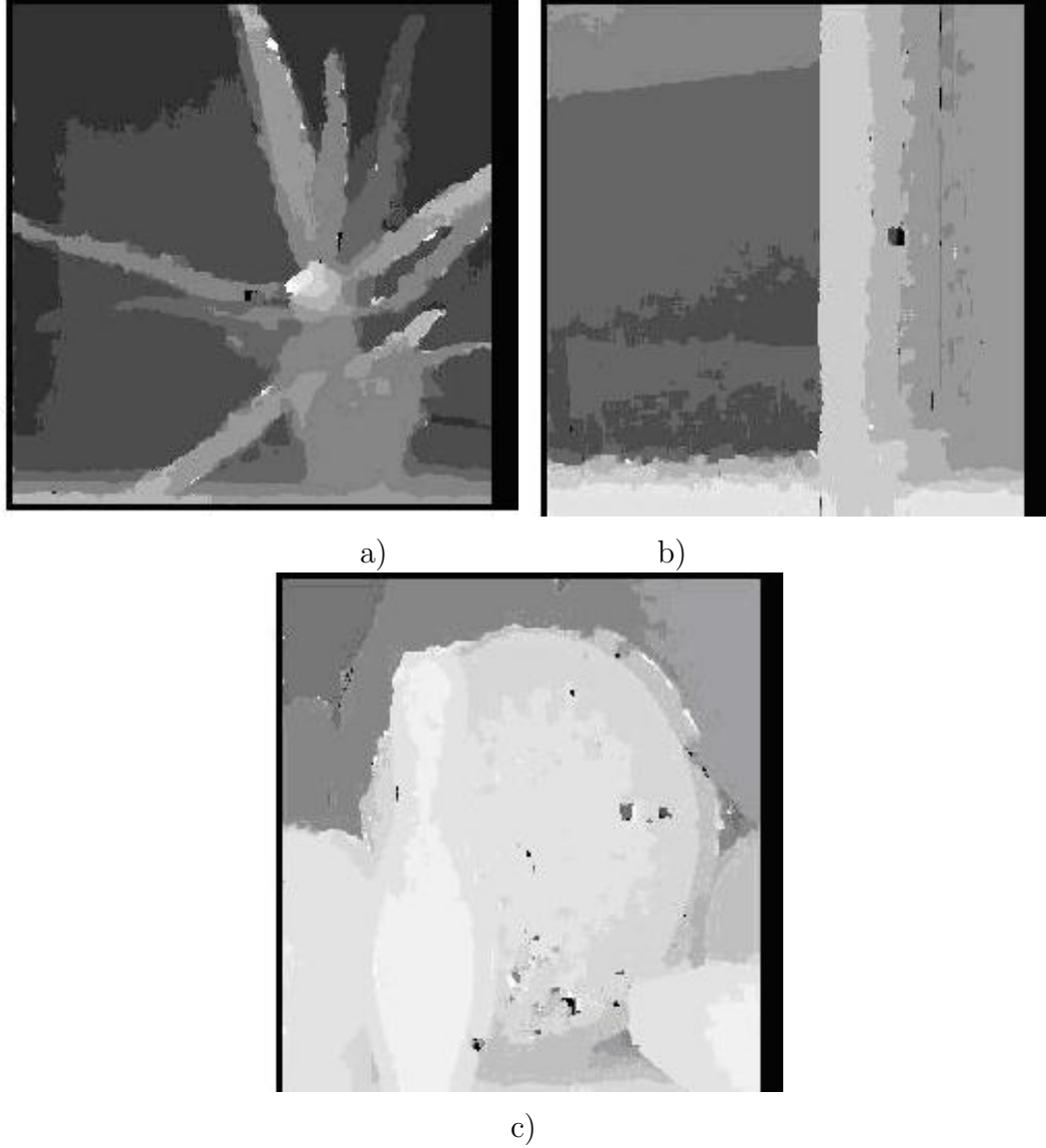


Figura 4.10: Mapas de disparidad obtenidos mediante el algoritmo propuesto con la familia de funciones atómicas correspondiente a π_6 para: a) Aloe, b) Wood1, c) Bowling1

Realizando un comparativo de forma cualitativa entre los mapas de disparidad obtenidos por los métodos presentados y el ground truth de cada una de las imágenes, podemos decir que de manera visual los mapas de disparidad más parecidos a sus respectivos ground truth

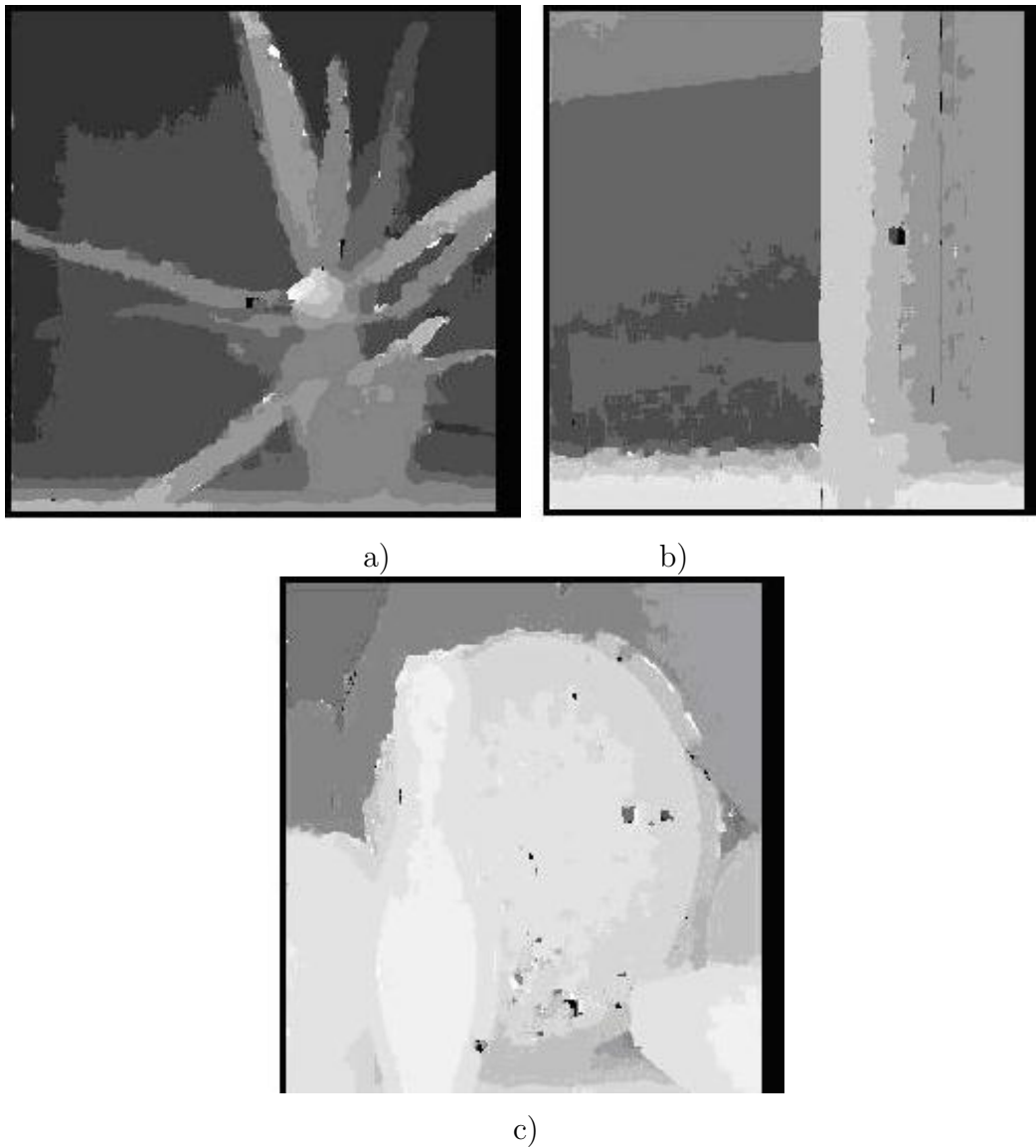


Figura 4.11: Mapas de disparidad obtenidos mediante el algoritmo propuesto con 4 niveles de descomposición y la función π_6 para: a) Aloe, b) Wood1, c) Bowling1

son los obtenidos por el método propuesto, es decir, usando 4 niveles de descomposición y la aplicación de la transformada wavelet basada en la función atómica de la familia π , en este caso π_6 . Así mismo, podemos decir que los peores resultados cualitativos son para la aplicación usando flujo óptico con las restricciones de Lucas y Kanade.

4.3.2. Resultados cuantitativos para los mapas de disparidad

Una vez obtenidos los mapas de disparidad por las diferentes técnicas, se implementaron las medidas cuantitativas presentadas en el inicio de éste capítulo. Las medidas de Porcentaje de disparidades mal seleccionados (*QBD*) y la medida de similitud en imágenes (*SSIM*) se presentan en la siguiente tabla.

Tabla 4.1: Medidas cuantitativas aplicadas a los mapas de disparidad obtenidos por diferentes métodos a diferentes imágenes.

Imagen	LyK	SSD	RB	<i>WBio</i>	<i>WCoi f</i>	<i>WHaar</i>	<i>WAF</i> π_6	<i>MWAF</i> π_6
Aloe								
SSIM	0.3983	0.6166	0.3017	0.9267	0.5826	0.5776	0.9232	0.9232
QBP	0.1121	0.4722	0.9190	0.0297	0.4517	0.4420	0.0130	0.0111
Venus								
SSIM	0.1991	0.4320	0.2145	0.5979	0.4530	0.4472	0.4604	0.6947
QBP	0.3084	0.1428	0.2013	0.1694	0.5014	0.5010	0.1930	0.1091
<i>Lampshade1</i>								
SSIM	0.0861	0.6320	0.3124	0.7061	0.7061	0.7081	0.6897	0.7619
QBP	0.2430	0.2800	0.3410	0.2072	0.2071	0.2071	0.2017	0.1426
<i>Wood1</i>								
SSIM	0.1089	0.7142	0.7051	0.9367	0.7096	0.7072	0.9448	0.9448
QBP	0.1316	0.2376	0.2100	0.1258	0.2400	0.2402	0.1180	0.0919
<i>Bowling1</i>								
SSIM	0.1118	0.6925	0.7081	0.8828	0.6690	0.6672	0.9084	0.9084
QBP	0.1720	0.1885	0.0645	0.0555	0.2010	0.2011	0.0119	0.0165
Reindeer								
SSIM	0.1557	0.7460	0.7143	0.7393	0.7321	0.7308	0.6819	0.7001
QBP	0.3910	0.1250	0.2810	0.1418	0.1565	0.1570	0.1513	0.1680

Tomando en cuenta que un valor cercano a uno en el parámetro SSIM se considera mejor, es decir, la similitud entre las imágenes es alta y por otro lado un valor cercano a cero en el parámetro QBP indica que el número de disparidades mal seleccionadas es muy bajo, es necesario realizar un análisis a la tabla 4.1. Como podemos observar en la tabla anterior, los valores presentados en la aplicación basada en multiwavlets con funciones atómicas es la que obtiene los más altos indicadores en el caso del parámetro SSIM, en forma general.

Para el parámetro QBP, los valores presentados en la aplicación basada en multiwavelets con funciones atómicas es la que obtiene los más bajos indicadores para todas las imágenes presentadas. Además podemos observar que la aplicación presentada con wavelets basadas en funciones atómicas obtiene los segundos mejores resultados en general y por último, es necesario mencionar que la implementación de la estimación del mapa de disparidad con la técnica de Lucas y Kanade presenta los peores resultados tanto para el parámetro SSIM como para el QBP.

4.3.3. Resultados obtenidos en la síntesis de anáglifos

Siguiendo con la presentación de los resultados, presentamos los anáglifos sintetizados por los diferentes métodos implementados. La primera parte presenta los anáglifos sintetizados usando las imágenes sintéticas, la segunda parte, presenta los anáglifos sintetizados para los videos sintéticos y los videos reales.

4.3.3.1. Anáglifos resultantes usando imágenes sintéticas a color

La figura 4.12 muestra los resultados obtenidos para la técnica de Lucas y Kanade.

Para la técnica de warping, la figura 4.13 presenta los resultados visuales.

Los anáglifos resultantes usando el algoritmo basado en la estimación de la semejanza estéreo se presentan en la figura 4.14.

En la figura 4.15 presentamos los resultados obtenidos cuando se implementó el algoritmo de multi-wavelets basadas en funciones atómicas.

Realizando un comparativo de forma cualitativa entre los diferentes anáglifos obtenidos, podemos decir que de manera visual los anáglifos con mejor percepción en 3D son los obtenidos por el método propuesto. Así mismo, podemos decir que los peores resultados cualitativos son para la aplicación usando flujo óptico con las restricciones de Lucas y Kanade, ya que presentan poca percepción en 3D y también presentan efectos de fantasma.

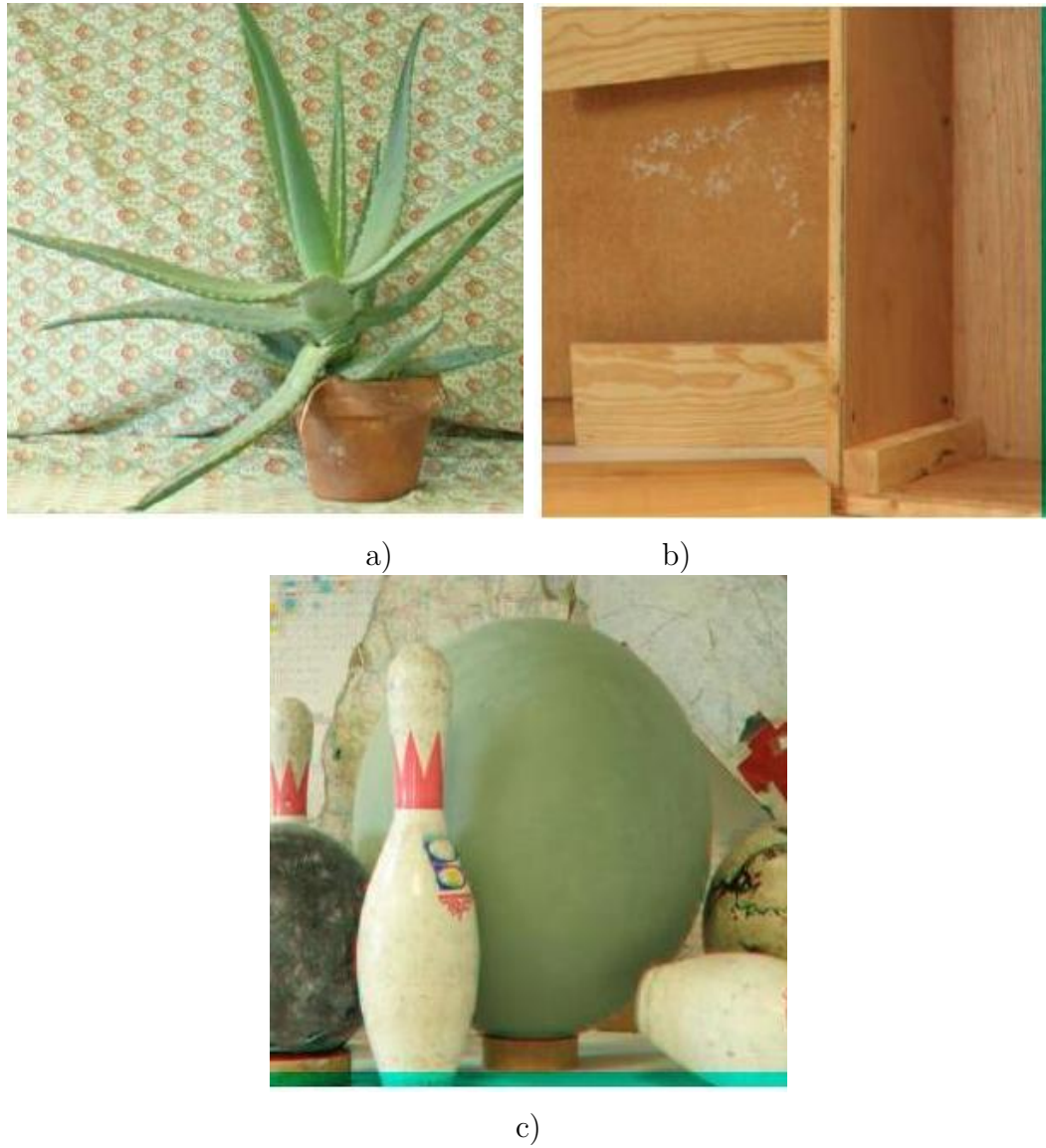


Figura 4.12: Anaglifos obtenidos usando Lucas y Kanade para: a)Aloe, b) Wood1, c) Bowling1



a)



b)



c)

Figura 4.13: Anaglifos obtenidos usando warping para: a) Aloe, b) Wood1, c) Bowling1



a)



b)



c)

Figura 4.14: Anaglifos obtenidos usando semejanza estéreo para: a) Aloe, b) Wood1, c) Bowling1

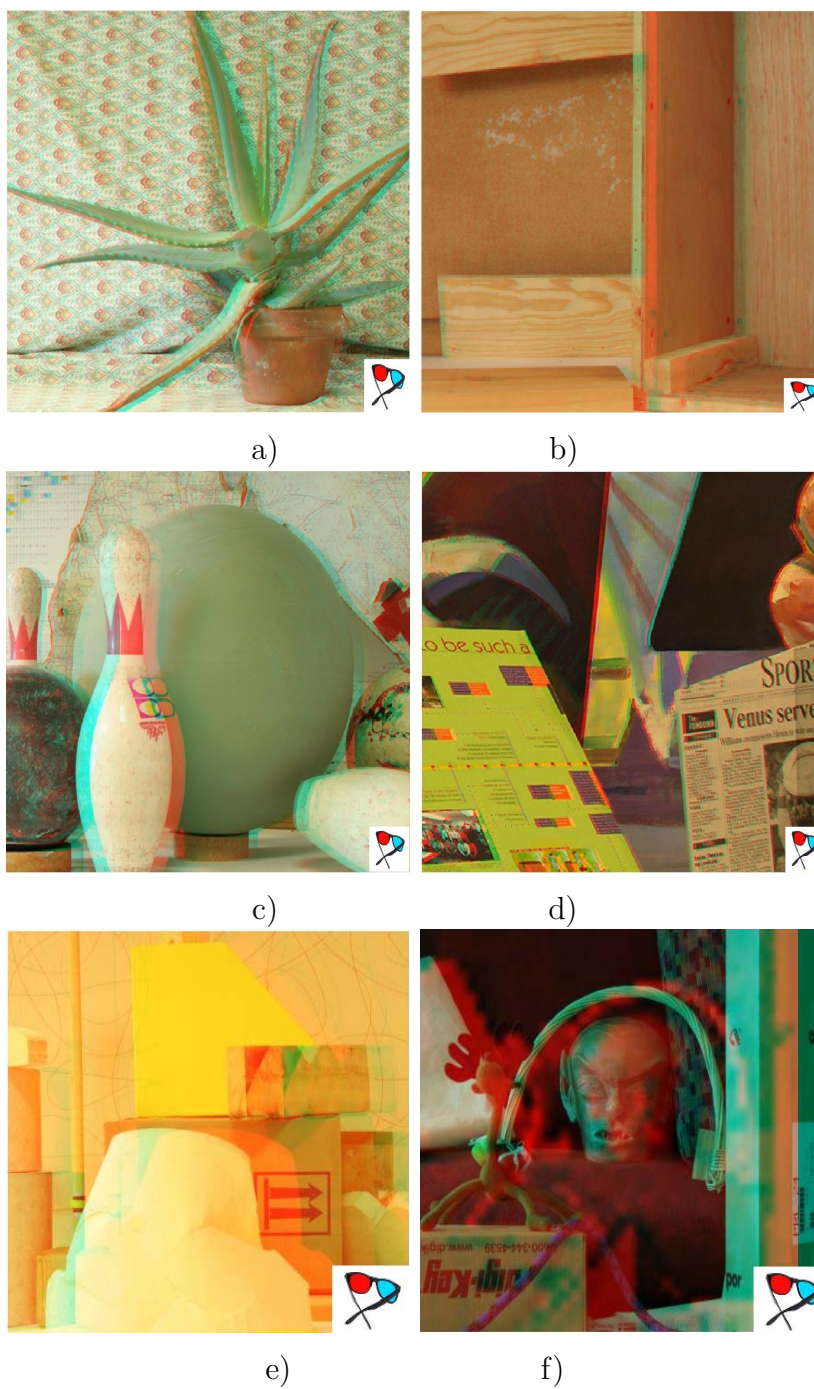


Figura 4.15: Anaglifos obtenidos usando M-WAF con π_6 para: a) Aloe, b) Wood1, c) Bowling1, d) Venus, e) Lampshade, f) Reindeer

4.3.3.2. Anáglifos resultantes usando video secuencias a color

Esta sección presenta los anáglifos obtenidos cuando se usan secuencias de video como entrada de los algoritmos. Se han realizado pruebas tanto en las video secuencias sintéticas y las video secuencias reales.

Para la implementación usando el mapa de disparidad obtenido mediante Lucas y Kanade, los anáglifos obtenidos se presentan en la figura 4.16.

La figura 4.17 presenta los resultados usando la semejanza entre píxeles.

Para el caso del algoritmo implementando semejanza estéreo, la figura 4.18 presenta una trama de dos video secuencias usadas.

Se realizaron además simulaciones utilizando las wavelets de las familias clásicas. Como resultado de diversas simulaciones con dichas familias, la wavelet seleccionada es *Biortogonal*6.8 ya que fue con ésta wavelet que se observaron mejores resultados. La figura 4.19 muestra los anáglifos obtenidos.

Como comparativo, también se realizaron diversas simulaciones pero ahora con wavelets basadas en funciones atómicas, la función atómica seleccionada fué up_1 ya que representa la wavelet más sencilla de todas las familias. La figura 4.20 muestra los resultados obtenidos.

En la figura 4.21 se presentan resultados para los videos mencionados en el caso de implementar el algoritmo propuesto.

Realizando un comparativo visual entre los distintos anáglifos presentados, podemos observar que en el caso de los presentados usando Lucas y Kanade, los anáglifos tienen poca percepción tridimensional y presentan desperfectos tales como fantasmao y pérdida de color, para el caso de la técnica usando warping, los anáglifos presentan poca percepción tridimensional aunque no presentan fantasmao. Visualmente, los anáglifos con mejor percepción tridimensional y con menor presencia de desperfectos son los presentados por el método propuesto.

4.3.4. Tiempos de procesamiento en simulación

Tomando en consideración que un algoritmo eficiente para la obtención de imágenes y videos con percepción tridimensional se considera cuando el algoritmo tiene un bajo costo computacional y buena percepción de profundidad, en esta sección presentamos los tiempos de simulación de dichas implementaciones. Los algoritmos fueron corridos en una pc con un procesador *PhenomTM X3* a 2.10 GHz con 2GB en RAM, el software utilizado fué *MatlabTM 2008b*.

La tabla 4.2 muestra los tiempos de procesamiento de cada uno de los algoritmos implementados, es necesario mencionar que los tiempos de procesamiento fueron medidos desde la lectura de las diversas imágenes utilizadas hasta la visualización de los anáglifos en pantalla.

Tabla 4.2: Tiempos de procesamiento obtenidos por diferentes métodos a diferentes imágenes.

Imagen	LyK	Warping	SM	WClasicas	WAF	MWAF
Aloe	7.06s	5.31s	11.24s	6.16s	6.19s	6.77s
Venus	5.55s	4.42s	11.09s	6.20s	6.19s	6.70s
<i>Bowling1</i>	7.00s	4.07s	11.22s	6.15s	6.21s	6.75s
<i>Lampshade1</i>	6.97s	6.23s	11.19s	6.22s	6.20s	6.75s
<i>Wood1</i>	7.09s	4.18s	11.16s	6.18s	6.24s	6.74s
Reindeer	6.98s	4.28s	11.15s	6.18s	6.22s	6.76s

Observando la tabla anterior podemos decir que los mejores tiempos de procesamiento son los correspondientes a la aplicación de warping. Haciendo un balance entre los tiempos de procesamiento de las multi-wavelets basadas en funciones atómicas, dichos tiempos estan en promedio abajo de las aplicaciones correspondientes a Lucas y Kanade, estéreo semejanza y para wavelets ya sea clásicas ó basadas en funciones atómicas. Es importante remarcar que tanto en las medidas SSIM, QBP, visualización de mapas de disparidad y percepción en 3D, la propuesta presentada en este trabajo, supera a los algoritmos presentados.



Figura 4.16: Anaglifos obtenidos usando Lucas y Kanade para: a)Foreman, b)Flowers, c) VideoTest1



Figura 4.17: Anaglifos obtenidos usando Lucas y Kanade para: a)Foreman, b)Flowers, c) VideoTest1

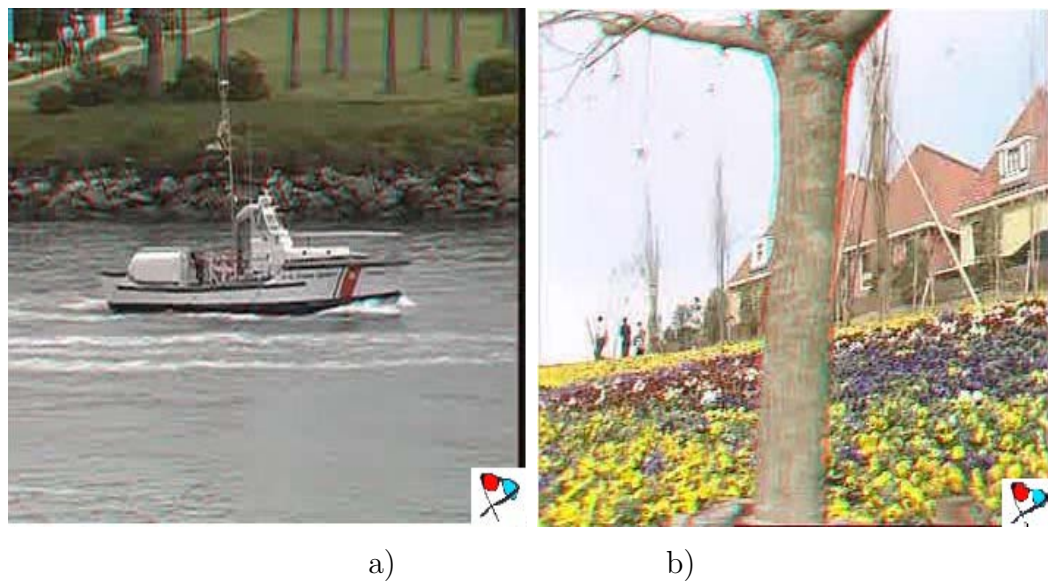


Figura 4.18: Anaglifos obtenidos usando semejanza estéreo para: a)Foreman, b)Flowers



Figura 4.19: Anáglifos obtenidos usando la wavelet de la familia *Biortogonal68* para:
a)Foreman, b)Flowers, c) VideoTest1



Figura 4.20: Anaglifos obtenidos usando la wavelet basada en funciones atómicas de la familia up_1 para: a)Foreman, b)Flowers, c)VideoTest1



Figura 4.21: Anaglifos obtenidos usando M-WAF con π_6 para: a) Coastguard, b) Flowers, c) VideoTest1, d) VideoTest2

4.4. Conclusiones del capítulo

Se investigaron y seleccionaron las imágenes y videos que se presentan en la literatura con la finalidad de simular los algoritmos seleccionados. Es importante decir que las imágenes sintéticas fueron seleccionada debido a que presentan el ground truth correspondientes a los mapas de disparidad originales. Además se presentaron las medidas cuantitativas seleccionadas para medir el rendimiento de los algoritmos seleccionados. La medida SSIM representa una nueva forma de medir diferentes tipos de imágenes y está basado en la estructura, el contraste y la luminosidad de las imágenes a medir.

Se presentan los mapas de profundidad para cada uno de los algoritmos e imágenes seleccionadas y además, basados en los resultados cualitativos y cuantitativos, se determinó que el algoritmo propuesto para la extracción de mapas de profundidad presenta mayor robustez a diferentes tipos de imágenes. Así mismo, debido a que los mapas de profundidad son de mejor calidad, los anáglifos resultantes gozan de mejor percepción tridimensional que los obtenidos con las técnicas investigadas.

En el caso de las video secuencias, podemos observar que los anáglifos obtenidos con el método propuesto también gozan de mejor percepción tridimensional y con menores desperfectos. Los anáglifos resultantes cuando se aplican video secuencias reales como información de entrada, justifican el uso del algoritmo propuesto en aplicaciones reales.

Se presentan los tiempos de procesamiento para los algoritmos investigados y se justifica que el algoritmo propuesto es candidato para posibles aplicaciones en tiempo real, implementándolas en hardware específico.

Capítulo 5

Implementación del algoritmo propuesto en hardware específico

En este capítulo, se abordan las características físicas del hardware seleccionado para la implementación del algoritmo propuesto, además se establece la metodología requerida para el desarrollo de la aplicación, finalmente se presentan los tiempos obtenidos en el hardware seleccionado.

5.1. Características del EVM DM642

El componente más importante del EVM DM642 es un Procesador de Medios Digitales, el cuál está basado en la línea de DSP's (Digital Signal Processors) de TI (Texas Instruments) C64xx. El DM642 se caracteriza por tener un conjunto de periféricos integrados en el chip, el cuál incluye 3 puertos de interfase de video, un controlador bus I²C, un puerto de audio serial multicanal, una EMIF de 64-bit, un controlador MAC ethernet 10/100 y una interfase PCI. El EVM DM642 añade todos los componentes externos necesarios para convertir al DM642 en un sistema de trabajo[45], [46]. Algunas características de la tarjeta incluyen:

- * Un DSP TMS320DM642 que trabaja a 720 MHz
- * 32 Mb de SDRAM
- * 4 Mb de memoria Flash lineal
- * 2 decodificadores de video
- * 1 codificador de video

- * Implementación del FPGA para despliegue en pantalla
- * UART doble con drivers RS-232
- * Un codec estéreo AIC23
- * Tarjeta Ethernet 10/100
- * EEPROM I²C de 32 Kb
- * 8 LEDs programables
- * Diversos formatos de entradas y salidas de video
- * Soporte para rangos de datos en HDTV
- * Conectores de expansión para tarjetas hermanas
- * Emulación JTAG a través de una interface de emulador externo

La Figura 5.1 muestra una fotografía del DSP, donde podemos observar, de manera general la arquitectura del mismo.

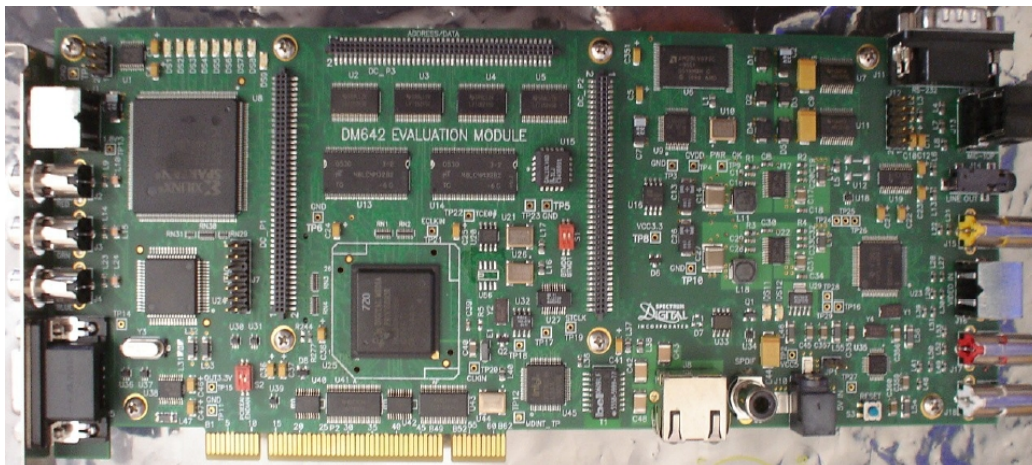


Figura 5.1: Arquitectura del DM642.

El EVM (Evaluation Module) sirve como una referencia en el diseño de hardware para el DSP TMS320DM642. La interface del DSP en el EVM DM642 para los periféricos es a través de un bus EMIF de 64 bits o por uno de los tres puertos de video de 8/16 bits. El SDRAM, Flash, FPGA y UART están conectado a uno de los buses. El bus EMIF también se conecta a la expansión de la tarjeta hermana. La interface de los codificadores de video y

decodificadores están conectados con los puertos de video y los conectores de expansión. El estándar en el EVM son dos decodificadores y un codificador. Las funciones de despliegue en pantalla se implementan en un FPGA externo el cuál reside entre el puerto de salida de video y el decodificador de video.

El codec en la tarjeta *AIC23B* permite al DSP transmitir y recibir señales de audio analógico. El bus I²C se usa para la interface de control del codec y el bus McASP se usa para los datos. La interface análoga se hace a través de tres jacks de audio que corresponden a la entrada del micrófono, la línea de entrada y la línea de salida. El codec puede seleccionar a el micrófono o a la línea de entrada como entrada activa. La salida analógica se lleva al conector de la línea de salida (con una ganancia fijada). El bus McASP se puede re-direccionar a los conectores de expansión mediante software.

Se usa un arreglo de compuertas programable llamado FPGA para implementar la lógica que enlaza a los componentes de la tarjeta. El FPGA tiene también un registro de interfase de usuario basado en software que permite al usuario configurar la tarjeta permitiendo escribir y leer en dichos registros.

El EVM incluye 8 LEDs los cuales pueden ser usados para proporcionar al usuario una retroalimentación interactiva. Estos LEDs se accesan leyendo y escribiendo en los registros del FPGA.

El Code Composer Studio (CCS) se comunica con el EVM a través de un emulador externo vía conectores JTAG externos de 14 ó 60 pines. El EVM trabaja directamente con el CCS. La Figura 5.2, muestra un diagrama a bloques de la arquitectura del EVM DM642, donde se muestran los buses y componentes incluidos en el sistema de desarrollo.

Una descripción general de los espacios de memoria que posee este DSP se describen a continuación.

5.1.1. Mapa de Memoria

La familia de DSPs C64xx tiene un gran espacio de direcciones en byte direccionables. El código del programa y los datos se pueden colocar en cualquier lugar del espacio de direcciones unificado. Las direcciones son siempre de 32 bits.

El mapa de memoria muestra en el lado izquierdo el espacio de direcciones de un procesador genérico DM642, con detalles específicos del uso de cada región en el lado derecho (Figura 5.3). Por default, la memoria interna se coloca al comienzo del espacio de direcciones. Las porciones de memoria pueden ser remapeadas en software en el caché *L2* en vez de la RAM fija.

La Interface de Memoria Externa (“EMIF” External Memory Interface) tiene 4 regiones direccionables separadas llamadas “espacios habilitados del chip” (CE0-CE3). La SDRAM

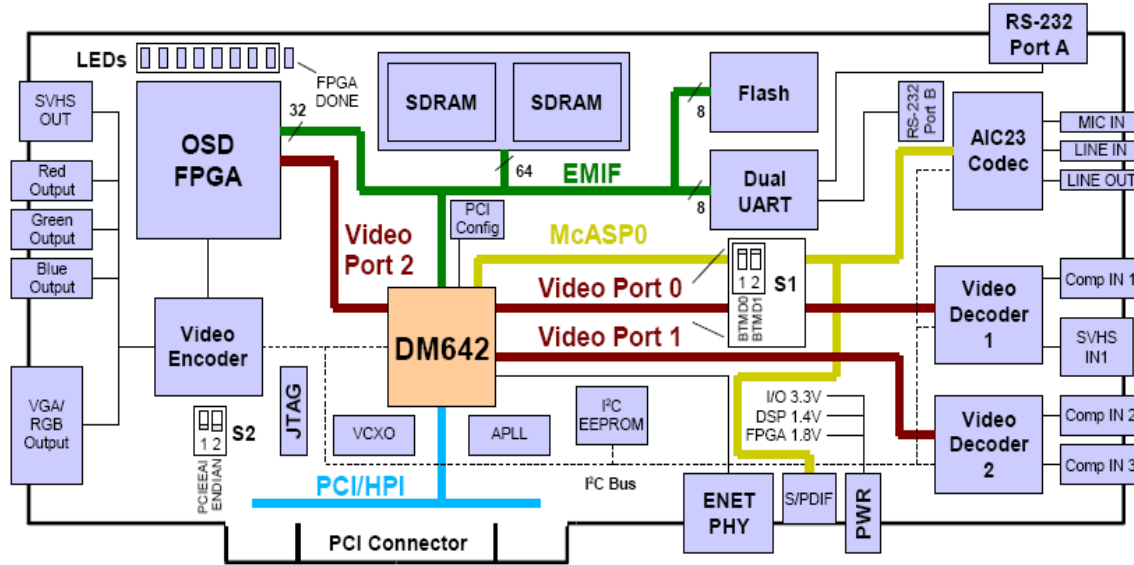


Figura 5.2: Diagrama a bloques del EVM DM642.

ocupa el CE0, mientras que la Flash, UART y el FPGA se mapean en CE1. Las tarjetas hermanas usan CE2 y CE3. El espacio CE3 se configura para una operación síncrona para funciones de despliegue en pantalla y los otros registros síncronos son implementados en el FPGA externo.

Address	Generic DM642 Address Space	DM642 EVM
0x00000000	Internal Memory/Cache	Internal Memory/Cache
0x00040000	Reserved Space or Peripheral Registers	Reserved or Peripheral
0x80000000	EMIF CE0	SDRAM
0x90000000	EMIF CE1	Flash
0xA0000000	EMIF CE2	UART/FPGA Regs
0xB0000000	EMIF CE3	Daughter Card
		FPGA Sync Regs
		Daughter Card

Figura 5.3: Mapa de memoria del EVM DM642.

5.1.2. Interfase EMIF

El DM642 incorpora una interfase de memoria externa de 64 bits. Cuatro “chip enables (CEs)” permiten dividir hasta por cuatro el espacio de direcciones y permitir accesos sín-

cronos y asíncronos de 8, 16, 32 y 64 bits. El EVM DM642 usa los “chip enables” CE0, CE1 y CE3. CE0 se rutea para el bus SDRAM de 64 bits. El CE1 se usa para funciones de 8 bits en Flash, UART y el FPGA. CE3 se selecciona para funciones síncronas. El CE2 y CE3 se rutean hacia la interface de los conectores de la tarjeta hermana.

5.1.3. Interfase de Memoria SDRAM

El EVM DM642 tiene una interfase de 64 bits para el bus SDRAM en el espacio CE0. Estos 32 megabytes de espacio SDRAM se usan para almacenamiento del programa, datos y video. El bus usa un dispositivo externo PLL para operar el SDRAM a 133 megahertz para una ejecución óptima. El refresco para el SDRAM se maneja de manera automática por el DM642.

5.1.4. Interfase de Memoria Flash

El DM642 tiene 4 Megabytes de memoria Flash mapeadas dentro de la porción más baja del espacio CE1. La memoria Flash se usa principalmente para la carga de boteo (boot) y el almacenaje de la información de configuración del FPGA. El espacio CE1 está configurado con una amplitud de 8 bits en el DM642 y la memoria Flash tiene una amplitud de 8 bits. El espacio de direcciones de memoria disponibles en el espacio CE1 es más pequeño que el tamaño de la Flash, por tanto, el FPGA se usa para crear 3 páginas de líneas de direcciones extendidas. Estas líneas de direcciones extendidas son direccionables vía el Registro Base de la Flash del FPGA.

5.1.5. Interfase UART

El UART doble (TLC16C752) es una memoria mapeada dentro de la mitad superior del CE1 del DM642 junto con los registros asíncronos del FPGA. Cada UART, *A* y *B*, ocupan 8 locaciones. El CE1 está configurado para un acceso de 8 bits.

5.1.6. Interfase de Memoria Síncrona y Asíncrona del FPGA

El FPGA implementa registros síncronos en el espacio CE3. Estos registros se usan primeramente para funciones de despliegue en pantalla y algunas funciones fijas del EVM. El FPGA tiene 10 registros de memoria asíncrona los cuales residen en la porción superior del espacio CE1.

5.1.7. Interfases Puerto/McASP de Video

El DM642 tiene tres puertos de video. Estos puertos se pueden subdividir para permitir funciones opcionales tales como un McASP o SPDIF en los puertos 0 y 1. El puerto de video 0 y el puerto de video 1 son usados como puertos de captura y el puerto de video 2 se usa como un puerto de despliegue.

5.1.8. Puertos Decodificadores de Video

En el EVM DM642 los puertos subdivididos de video 0 y 1, se usan como entradas ó puertos de captura (1 y 2). Estos puertos hacen interfase con los decodificadores de video *TVP5416* y el *TVP5150A*. Los puertos de video corren a través de switches CBT, así, pueden ser deshabilitados de manera selectiva para el uso de tarjetas hermanas. La otra mitad de los puertos se usan para una interfase en el McASP. El puerto de captura 1 hace interfase con las fuentes de video vía un jack RCA J15 para video y un conector J16 de cuatro pines de un mini-din de S-video. La entrada es una fuente de video compuesto tales como un reproductor DVD o una cámara de video. Los decodificadores son programables vía el bus I²C del DM642 y pueden hacer interfase con la mayoría de los estándares de video tales como el NTSC, PAL y SECAM, programando de manera apropiada los registros internos en el decodificador.

5.1.9. Puerto del Codificador de Video

El puerto de video 2 del DM642 se usa para manejar el codificador de video. Éste es ruteado a través del U8 del FPGA para implementar funciones avanzadas tales como el despliegue en pantalla, pero el modo por default es pasar el video directamente al decodificador de video SAA7105 de Phillips. El codificador puede manejar cualquiera de los siguientes espacios, el RGB, video componente de HD, video compuesto NTSC/PAL ó S-video dependiendo de como sean programados los registros internos del SAA7105. El SAA7105 está configurado para programar los registros internos vía el bus I²C del DM642. El codificador hace interface a video compuesto o unidades de despliegue RGB. El EVM DM642 soporta salida de TV de alta definición pero requiere de algunos cambios en el filtro.

5.1.10. Funciones de Video del FPGA

El EVM DM642 usa un FPGA de la serie *XilinxTM XC2S300E* para implementar funciones de video mejoradas. En el modo default el FPGA pasa el video del puerto 2 de video al codificador Phillips. Para HDTV, el FPGA proporciona un reloj mejorado y, para funciones de despliegue en pantalla, el FPGA tiene FIFOs para mezclar los datos del puerto

de video 2 con los datos desde estos FIFOs internos. Los FIFOs en el FPGA se accesan vía el EMIF del DM642 en modo síncrono vía el espacio CE3.

5.1.11. Interface I²C

El bus I²C en el DM642 es ideal para la interfase con los registros de control de muchos dispositivos. En el EVM DM642 el bus I²C se usa para configurar el codificador de video, los video decodificadores y el codec estéreo. Una ROM I²C también tiene su interface vía el bus serial.

5.2. Características del Emulador Blackhawk PCI560

Este emulador pertenece a la tercera generación de una amplia gama que utiliza un emulador clase XDS560, este emulador soporta las capacidades avanzadas de los DSP's de Texas Instruments. Entre sus características incluye un TCK arriba de los $50MHz$ e incluye un cable multi-voltaje de 1.6m de longitud el cual es desmontable y tiene un montaje adecuado para usarse desde 0.5 volts hasta 5 volts para soportar voltaje bajo de los dispositivos TI. Además es compatible con todas las versiones de CCS. El emulados Blackhawk PCI560 JTAG suporta plataformas como TMS320, TMS470 y familias *OMAPTM* además de la nueva plataforma DaVinci.

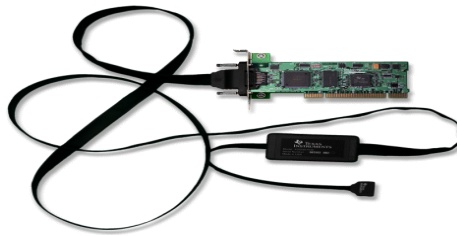


Figura 5.4: Emulador Blackhawk PCI560

Características

- Soporte *RTDXTM* de alta velocidad con una transferencia de datos arriba de $2MB/sec$.
- 1.6m de cable micro coaxial flexible de alta velocidad con cTI (Interfaz de tarjeta compacta de 20-pines)
- Sensor automático de bajo voltaje que soporta abajo de $0.5v$

- Capacidad de no interrupciones en tiempo real, con el AET.
- Más de 100 veces más rápido que el emulador XDS510.

Requerimientos del sistema

- PC Windows 2000/XP
- *CodeComposerStudioIDETM* v3.3 o anterior.
- Slot PCI libre en la PC.
- Disco CD-ROM para la instalación de los controladores.
- Tarjeta compatible con JTAG.

5.3. Características del software utilizado

El trabajo realizado en este trabajo de tesis se llevo acabo en *MATLABTM*, el cual es un lenguaje de alto nivel con un entorno interactivo que permite realizar tareas computacionalmente intensivas más rápido que en los tradicionales lenguajes de programación como: C, C++ y Fortran [47].

Características

- Lenguaje de alto nivel para técnicas de computación.
- Entorno de desarrollo para la gestión de código, archivos y datos.
- Herramientas interactivas para la exploración iterativa, el diseño, y la resolución de problemas, funciones matemáticas para álgebra lineal, estadísticas, análisis de Fourier y filtrado, optimización, e integración numérica, entre otros
- 2-D y 3-D para visualizar gráficos de las funciones de datos
- Herramientas para la construcción de interfases gráficas de usuario personalizadas
- Funciones para la integración de algoritmos basados en *MATLABTM* con aplicaciones externas, e idiomas, tales como C, C + +, Fortran, Java, COM y Microsoft Excel

MATLABTM cuenta con paquetes de soporte, uno de ellos el *TheTargetSupportPackageTMTC6software* se utiliza para trabajar con software en implementaciones de tiempo real a lenguaje C desde

un modelo de Simulink. Se puede compilar, vincular, descargar, y ejecutar código en diversas tarjetas de la familia C6000 de Texas Instruments. Este soporte de tarjeta es ideal para crear prototipos rápidos y desarrollar sistemas embebidos.

Simulink es una herramienta de modelado, simulación y diseño, está basado en plataformas GUI que son entornos basados en objetos, los cuales facilitan el diseño, así los diseños se realizan basados en modelos. Tiene integración con *MATLABTM* y además cuenta con una extensa librería de bloques predefinidos.

Existen herramientas fundamentales tales como:

- El bloque *Video and Image Processing*), que nos permite modelar, simular, implementar y verificar en Tiempo-Real sistemas con contenido de imágenes y video.
- Incluye diversos algoritmos predeterminados como son: conversiones, filtrado, analisis y mejora de video e imágenes, transformaciones geométricas, fuentes, etc.
- Es ideal para la implementación de sistemas embebidos.
- Enlace con *CodeComposerStudioTM*.
- Permite la co-simulación, la automatización de proyectos, depuración y verificación para los DSP's C2000, C5000 y C6000.
- Bloque de comunicación con TI *C6000TM*.
- soporta perifericos y tarjetas de evaluación.

Simulink es muy util en el procesamiento de imágenes y video porque tiene una libreria de algoritmos extensa, se puede modelar en punto flotante, genera el código automáticamente y se realizan fácilmente los prototipos.

5.3.1. Utilizando *CCSTM* con *TargetSupportPackageTM*

Texas Instruments (TI) mercadea un conjunto completo de herramientas de software para utilizar cuando se desarrollen aplicaciones para su hardware C6000. Estas herramientas trabajan en conjunto: Target Support Package TC6 software usando Code Composer Studio (CCS) Integrated Development Environment (IDE) con the Real-Time Workshop software y el bloque C6000lib de Simulink. La ejecución de código genera desde el trabajo en Tiempo-Real en una tarjeta particular que requiere un código específico que contenga dispositivos de I/O y servicios de rutinas de interrupciones (ISR).

5.4. Implementación del algoritmo propuesto

Para la implementación del algoritmo propuesto, es necesario reescribir el código generado en MATLAB a su equivalente en Simulink. Una vez teniendo el proyecto en Simulink, se hacen uso de las herramientas descritas con anterioridad para poder establecer la comunicación entre MATLAB y la tarjeta EVM DM642 [48]. De ésta forma, MATLAB transfiere el código desarrollado a CCS, el emulador realiza la conexión entre la PC y la tarjeta y finalmente el código se baja a la misma y empieza a correr el programa resultante. Es importante incluir los módulos de captura y desplegado de información en el programa principal desarrollado en Simulink. La figura siguiente muestra el programa desarrollado.

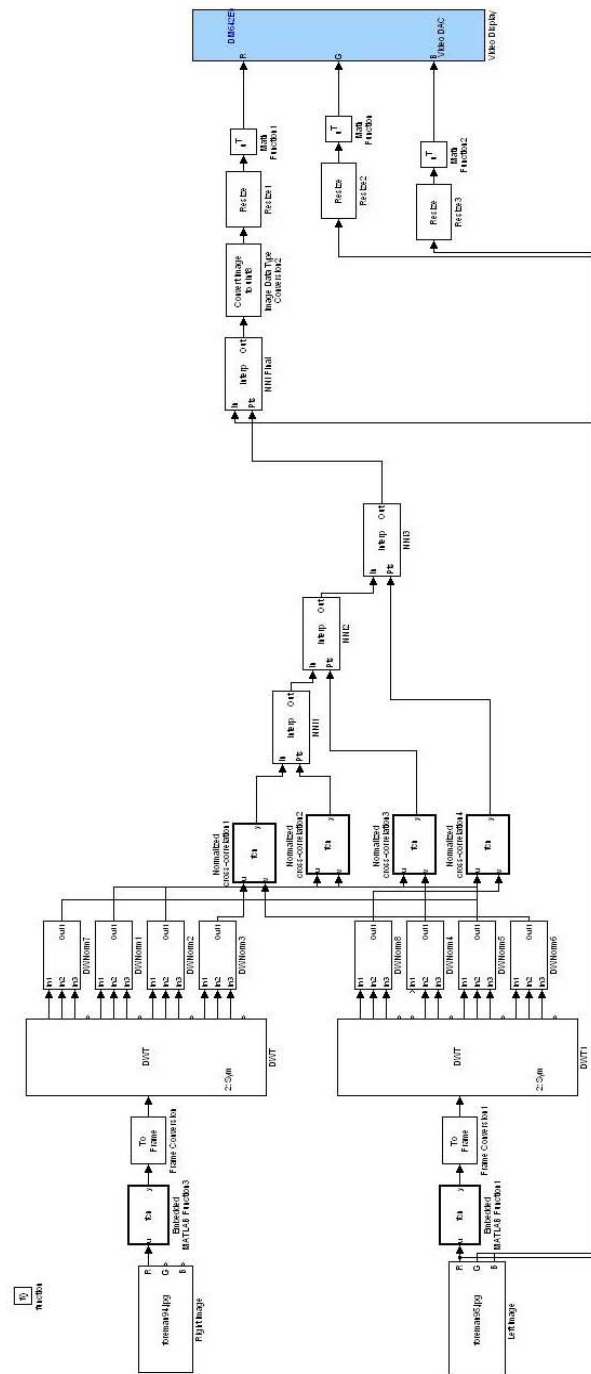


Figura 5.5: Algoritmo propuesto desarrollado en Simulink

Como podemos observar, existe un bloque llamado *VideoDisplay*, el cual se encarga de realizar la conexión entre la tarjeta utilizada y un display de uso común para poder observar el anáglifo resultante. Es importante mencionar, que dentro de cada una de las subfunciones existe un desarrollo de bloques (por ejemplo, *DWNormalized* contiene la normalización en cada nivel de descomposición) ó existe una función S (por ejemplo, en *NormalizedCross-correlation* contiene un programa desarrollado en Matlab) que realiza diversos procesos dentro del algoritmo en general. La figura presentada anteriormente implica todo el algoritmo desarrollado en Simulink, pero, para poder llevarlos hacia el DSP se realizaron varios pasos, los cuales nos dieron la configuración final de la aplicación. Fué necesaria la separación del proceso en tres partes iguales y con la misma carga computacional cada uno de ellos. Se decidió la separación en tres partes iguales debido a que la memoria disponible en la tarjeta DSP no era suficiente. Es así que la presente implementación se realizó en tres tarjetas EVM DM642 conectadas en serie en donde la primera de ellas, contiene el par estéreo y dos niveles de descomposición de la wavelet basada en funciones atómicas. El segundo DSP contiene dos niveles de descomposición y la interpolación entre los niveles para obtener el mapa de disparidad final. El tercer DSP contiene las etapas de síntesis y mejoramiento de anáglifos, así como la interfase de conexión al monitor.

5.5. Tiempos de procesamiento

Una vez que el proceso ha sido implementado, se realizaron las mediciones correspondientes al tiempo de procesamiento del algoritmo propuesto y algunos algoritmos eficientes en la generación de anáglifos que funcionan como comparativo en esta etapa. Para medir los valores del tiempo de procesamiento, se hizo uso del módulo de estadísticas del CCS, en donde, se configuró la tarea correspondiente al proceso en prioridad 1. Se procesaron dos tamaños de imágenes, la primera fué con tamaño 480×720 píxeles y la segunda con tamaño 240×360 píxeles. Los tiempos obtenidos están dados en $(Tiempo/Trama, s)$ y se presentan en la tabla siguiente.

Con los tiempos obtenidos, podemos encontrar que el algoritmo propuesto es capaz de procesar arriba de 12 tramas por segundo cuando la secuencia de entrada es de tamaño 480×720 píxeles, así mismo, tiene la posibilidad de procesar arriba de 20 tramas por segundo cuando la secuencia de entrada es de tamaño 240×60 píxeles. Los tiempos de procesamiento fueron encontrados sumando los tiempos de procesamiento de cada DSP y desde que se adquiere la imagen hasta que se despliega el anáglifo.

Tabla 5.1: Tiempos de procesamiento obtenidos por diferentes métodos a diferentes imágenes.

Algoritmo	Imagen de 480×720	Imagen de 240×360
Wavelet clásicas ($Coi f_2$, $Db6.8$, $Haar$)	0.071	0.031
Wavelet AF (up , fup_4 , π_6)	0.071	0.031
M-WClasicas ($Coi f_2$, $Db6.8$, $Haar$)	0.080	0.048
M-WAF (up , fup_4 , π_6)	0.081	0.048

5.6. Conclusiones del capítulo

Con base en las características presentadas se seleccionó el hardware específico para la implementación del algoritmo propuesto. Se investigaron las diversas configuraciones e interfaces útiles en la programación de dispositivos físicos y su interconexión con el mundo exterior. Se realizó la conversión del código de Matlab a Simulink para simplificar la creación de un proyecto y su aplicación en hardware específico. Se extrajeron los tiempos de procesamiento para diferentes tamaño de imágenes a color de entrada de los mejores algoritmos de obtención de análogos, teniendo en uno de los casos el procesamiento casi real de la información.

Capítulo 6

Conclusiones generales y trabajo a futuro

6.1. Conclusiones generales

A lo largo de éste trabajo de tesis, se revisaron algunas técnicas para construir anáglifos, la cual representa una de las formas de representar imágenes estereoscópicas. Las técnicas clásicas utilizan una matriz de corrimiento para manipular la imagen, defasando los componentes de color para cada caso. Uno de los problemas más comunes en dichas técnicas es la presencia del fantasma y la pérdida de color. Los algoritmos estudiados fueron extendidos a su uso con videos sintéticos y reales tomando en cuenta tramas subsecuentes para obtener información de disparidad, ampliando su uso de sólo pares estéreo a procesamiento de video a color. Otra de las técnicas estudiadas y que es usada en video secuencias, es la presentada por Yaroslavsky, en la cual se usa información de movimiento proveniente del compresor MPEG con la finalidad de calcular mapas de disparidad que posteriormente será usada para sintetizar anáglifos. Con el mapa de disparidad construye una malla de remuestreo para manipular el color rojo del par estéreo y finalmente formar el anáglifo. Adicionalmente presenta un método para minimizar el fantasma en los anáglifos resultantes el cual está basado en la compresión del rango dinámico de los mapas de disparidad. Se realizó una revisión a las técnicas de obtención de mapas de disparidad basados en funciones wavelets clásicas las cuales tienen pocas y muy recientes aplicaciones en visión estéreo.

Derivado de la revisión de dichas técnicas, se propuso un algoritmo el cual está basado en funciones atómicas. Las funciones atómicas se proponen debido a que han demostrado mejor rendimiento en el área del procesamiento de imágenes y vídeo en comparación con las wavelets de familias clásicas; adicionalmente se propone el uso de estrategias fino-gruesas

en conjunto con las wavelets basadas en funciones atómicas para mejorar los resultados obtenidos.

Como pudimos constatar en base a los diferentes resultados visuales y a las medidas objetivas propuestas, el algoritmo presenta un mejor rendimiento en la generación de información con percepción tridimensional que los algoritmos estudiados y además pueden operar sin la utilización de hardware adicional en ninguna de sus etapas de procesamiento.

Se estudió la configuración de diversos equipos para la implementación del algoritmo propuesto y se seleccionó el EVM DM642 debido a su amplio uso y construcción dedicada al procesamiento de imágenes y video. Se implementaron los algoritmos con mejores resultados cualitativos y cuantitativos en hardware específico para probar su eficiencia computacional, dando como resultado un procesamiento casi real de imágenes y videos de diferente naturaleza.

Se demostró la eficiencia del algoritmo propuesto debido a diversas pruebas realizadas tanto en imágenes y videos sintéticos y reales.

6.2. Publicaciones con aportaciones científicas

Aportaciones científicas en la parte del algoritmo propuesto:

- Eduardo Ramos Diaz and Volodymyr Ponomaryov, "Reconstruction on 3D video from 2D real life sequences", *Revista de Ingeniería de la Universidad de Antioquia*, vol. 56, pp. 111-121, 2010. (JCR-ISI)
- E. Ramos, V. Ponomaryov, "Generation of 3D video sequences employing multi wavelets functions", *Optics Express*, The Optical Society, 2011. Aceptado con modificaciones (JCR-ISI)
- E.Ramos, V. Ponomaryov, L. Niño de Rivera, "Optimization of one algorithm for 3D video sequence visualization", *EURASIP Journal on Advances in Signal Processing*, Hindawi Publishing Corporation, 2011. En revisión. (JCR-ISI)
- Edaurdo Ramos and Volodymyr Ponomaryov, "3D color conversion using depth map estimation", *Telecommunications and Radio Engineering*, *Begell House Editors*, vol. 69, issue 5, 2010. (SCOPUS)
- Eduardo Ramos and Volodymyr Ponomaryov, "Generation of 3D video sequences employing depth map information", *Int. Trans. on Computer Science and Engin.*, vol. 61, no.1, 2010. (INSPEC)

Aportaciones científicas en la parte de la implementación en DSP:

- Eduardo Ramos and Volodymyr Ponomaryov, "Reconstruction of three dimensional objects using video sequences" *2010 Int. Kharkov Symp. on Physics and Eng. of Microwaves, Millimeter and Submill. Waves, MSMW 2010*, 2010 (Proceedings ISI)
- E. Ramos, M. Cruz Irisson, L. Nino de Rivera, V. Ponomaryov, "3D color video conversion from 2D video sequence using stereo matching technique", *52nd IEEE International Midwest Symposium on Circuits and Systems*, pp.739-742, 2009. (Proceedings ISI)
- E. Ramos, V. Ponomaryov, L. Niño de Rivera, "Optimization of one algorithm for 3D video sequence visualization", *EURASIP Journal on Advances in Signal Processing*, Hindawi Publishing Corporation, 2011. En revisión. (JCR-ISI)
- Eduardo Ramos and Volodymyr Ponomaryov, "Generation of 3D video sequences employing depth map information", *Int. Trans. on Computer Science and Engin.*, vol. 61, no.1, 2010. (INSPEC)
- V. Ponomaryov, E. Ramos, "3D video sequence reconstruction algorithms implemented on a DSP", *Proc. of Real-Time Image and Video Processing 2011, SPIE7871*, vol. 7871, 2011. (Proceedings ISI).

6.3. Trabajo a futuro

Como trabajo a futuro se proponen los siguientes puntos:

- Mejoramiento del algoritmo propuesto realizando diversas pruebas con imágenes con diferente naturaleza.
- Explorar la posibilidad de mejorar el tiempo de procesamiento del algoritmo propuesto ya sea con la inclusión de DSPt's con mejores características o con alguna configuración diferente presentada en este trabajo de tesis.
- Se propone el estudio de una medida de percepción tridimensional la cual pueda medir el rendimiento de los algoritmos orientados a la percepción tridimensional y no sólo a los mapas de profundidad.
- Desarrollar una aplicación en software especializado que mejore la distribución de la carga computacional en el algoritmo propuesto.
- Realizar algún algoritmo híbrido que tome como base el algoritmo propuesto y que incluya otras técnicas complementarias en el cálculo de mapas de disparidad.

Bibliografía

- [1] P. Howard, B. J. Rogers, *Binocular vision and stereopsis*, Oxford University Press, 1995.
- [2] B. Blundell, A. Schwarz, *Volumetric three dimensional display systems*, Wiley-IEEE Press, 2000.
- [3] L. Lipton, “Stereovision formats for video and computer graphics”, *StereoGraphics Corporation*, 1997.
- [4] H. Gernsheim, A. Gernsheim, *The history of photography from the camera obscura to the beginning of the modern era*, McGraw-Hill, 1969.
- [5] W. Sanders, D. McAllister, “Producing anaglyphs from synthetic images”, *Proceedings of SPIE 5006*, pp. 348-358, 2003.
- [6] D. F. McAllister, P. D. Hebbar “Color cuantization aspects in stereopsis”, *Proceedings of Stereoscopic Displays and Applications II, SPIE 1457*, pp. 233-241, 1991.
- [7] A. Woods, J. Merrit, “Stereoscopic display application issues”, *EIO2, SPIE*, 2002.
- [8] E. Dubois, “A projection method to generate anaglyph stereo images” *Proc. IEEE Int. Conf. Acoustics Speech Signal Processing*, pp. 1661-1664, 2001.
- [9] I. Ideses, L. Yaroslavsky, B. Fishbain, “3D from compressed 2D video”, *Proc. of Stereoscopic Displays and Virtual Reality Systems, SPIE6490*, 2007.
- [10] I. Ideses, L. Yaroslavsky, “A method to generating 3D video from a single video stream”, *Proc. of Vision, Modelling and Visualization*, Germany, November 2002.
- [11] L. P. Yaroslavsky, J. Campos, M. Espinola, I. Ideses, “Redundancy os stereoscopic images: Experimental evaluation”, *Optics Express*, vol. 13, pp. 10895-10907, 2005.

- [12] I. Ideses, L. Yaroslavsky, "New methods to produce high high quality color anaglyphs for 3D visualization", *Lecture Notes in Computer Science*, vol 3212/2004, pp.273-280, 2004.
- [13] A. Smolic, P. Kauff, S. Knorr, A. Hornung, M. Kunter, M. Muller, M. Lang, "Three dimensional video postproduction and processing", *Proc. of the IEEE*, vol. 99, no. 4, pp. 607-625, 2011.
- [14] D. Kim, D. Min, K. Sohn, ".A stereoscopic video generation method using stereoscopic display characterization and motion analysis", *IEEE Trans. Broadcast*, vol. 54, no. 2, pp. 188-197, 2008.
- [15] W. J. Tam, C. A. Vasquez, F. Speranza, "Three dimensional TV: A novel method to generating surrogate depth maps using color information", *Proc. SPIE Stereoscopic Displays and Applications XX*, vol. 7237, 2009.
- [16] S. Diplaris, N. Grammalidis, D. Tzovaras, M. G. Strintzis, "Generation of stereoscopic image sequences using structure and rigid motion estimation by extended Kalman filters", *Proc. IEEE Int. Conf. Multimedia Expo*, no.2, pp. 233-236, 2002.
- [17] K. Moustakas, D. Tzovaras, M. .G. Stintzis, "Stereoscopic video generation based on efficient layered structure and motion estimation form a monoscopic image sequence", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 8, pp. 1065-1073, 2005.
- [18] S. Knorr, M. Kunter, T. Sikora, "Stereoscopic 3D from 2D video with super resolution capability", *Signal Process, image Commun.* vol. 23, no.9, pp. 665-676, 2008.
- [19] D. Fleet, *Measurement of image velocity*, Kluwer Academic Publishers, 2001.
- [20] S. S. Beauchemin, J. L. Barron, "The computation of optical flow", *ACM Computing Surveys*, vol. 2, no.3, 1995.
- [21] J. L. Barron, S. S. Beauchemin, D. J. Fleet, "Performance of optical flow techniques", *IJCV*, vol. 12, 1994.
- [22] B. Horn, B. Shunck, "Determininig optical flow", *AI*, vol. 17, pp. 187-204, 1981.
- [23] T. Brox, A. Bruhn, N. Papenberg, J. Weickert, "High accuracy optical flow estimation based on a theory for warping", *Proc. 8th European Conference on Computer Vision Springer LNCS*, vol. 4, pp. 25-36, 2004.
- [24] S. Uras, F. Girosi, A. Verri, V. Torre, ".A computational approach to motion perception", *Biological Cybernetics*, vol. 60, pp. 79-87, 1988.

- [25] E. Mémin, P. Pérez, "Hierarchical estimation and segmentation of dense motion fields", *International Journal of Computer Vision*, vol. 26, no. 2, pp. 129-155, 2002.
- [26] I. Rudin, S. Osher, E. Fatemi, "Non linear total variation based noise removal algorithms", *Physica D*, vol. 60, pp. 259-268, 1992.
- [27] B. Alagoz, "Obtaining depth maps from color images by region based stereo matching algorithms", *OncuBilim Algorithm and System Labs*, vol. 8, no. 4, 2008.
- [28] A. Bhatti, S. Nahavandi, *Stereo Vision*, I-Tech, 2008.
- [29] S. Mallat, Z. Zhang, "Matching pursuits with time-frequency dictionaries", *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp 3397-3415, 1993.
- [30] S. Mallat, "Zero crossings of a wavelet transform", *IEEE Transactions on Information Theory*, vol. 37, pp 1019-1033, 1991.
- [31] J. C. Olive, J. Deubler, C. Boulin, "Automatic registration of images by a wavelet based multiresolution approach" *SPIE 2569*, pp. 234-244, 1994.
- [32] H. P. Pan, "General stereo matching using symmetric complex wavelets" *Wavelet Applications in Signal and Image Processing*, 1996.
- [33] S. Mallat, *A wavelet tour of signal processing*, Academic Press, 1999.
- [34] J. Magarey, A. Dick, "Multiresolution stereo image matching using complex wavelets", *Proc 14th Int. Conf. on Pattern Recognition*, vol. 1, pp. 4-7, 1998.
- [35] A. Bhatti, H. Ozkarmanli, "M-band multiwavelets from spline super functions with approximation order", *Conf. on Acoustics Speech and Signal Processing*, vol 4, pp. 4169-4172, 2002.
- [36] C. Juarez, V. Ponomaryov, J. Sanchez, V. Kravchenko, "Wavelets based on atomic function used in detection and classification of masses in mammography", *Lecture notes in artificial intelligence, LNAI 5317*, pp.295-304, 2008.
- [37] F. Gomeztagle, V. Kravchenko, V. Ponomaryov, "Super-resolution algorithms based on atomic wavelet functions in real-time processing of video sequences", *Proc. SPIE*, vol. 772403, 2010.
- [38] V. Kravchenko, H. Meana, V. Ponomaryov, *Adaptive digital processing of multidimensional signals with applications*, FizMatLit Edit., Moscow, 2009.

- [39] Y. Gulayev, V. Kravchenko, V. Pustoviot, ".^A new class of WA-Systems of Kravchenko-Rvachev Functions", *Doklady Mathematics*, vol. 75, no. 2, pp. 325-332, 2007.
- [40] A. Goshtasby, *2D and 3D image registration*, Wiley Publishers, 2005.
- [41] Z. Wang, A. Bovik, "Mean Squared Error: leve it or leave it?", *IEEE Signal Processing Magazine*, 2009.
- [42] W. Malpica, A. Bovik, Range image quality assessment by structural similarity", *IEEE Int. Conf. on Acoustics, Speech and Signal Proc.*, pp. 1149-1152, 2009.
- [43] "Página de la Universidad de Middlebury Stereo Vision", Septiembre 2009, <http://vision.middlebury.edu/stereo/data>.
- [44] "Página de la Universidad de Arizona State", Septiembre 2009, <http://trace.eas.asu.edu/yuv/index.html>.
- [45] TMS320DM642 Evaluation module with TVP Video Decoders, *Technical Reference 507345-0001*, Rev. B, 2004.
- [46] Texas Instruments, *Manual Reference SPRS200*, 2002.
- [47] S. Qureshi, *Embedded image processing in Matlab*, Labcyte Springer, 2005.
- [48] E. S. Gophi, *Algorithm collections for digital signal processing applications using Matlab*, Springer, 2007.
- [49] M. Unser, A. Aldroubi, ".^A multiresolution image registration procedure using spline pyramids", *SPIE Mathematical Imaging*, vol.2034, pp.160-170, 1993.
- [50] X. Zhou, E. Dorrer, ".^AAutomatic image-matching algorithm based on wavelet decomposition", *IAPRS*, vol. 30, no. 1, pp. 951-960, 1994.

Apéndice A

Publicaciones

Aportaciones científicas en la parte del algoritmo propuesto:

- Eduardo Ramos Diaz and Volodymyr Ponomaryov, "Reconstruction on 3D video from 2D real life sequences", *Revista de Ingeniería de la Universidad de Antioquia*, vol. 56, pp. 111-121, 2010. (JCR-ISI)
- E. Ramos, V. Ponomaryov, "Generation of 3D video sequences employing multi wavelets functions", *Optics Express*, The Optical Society, 2011. Aceptado con modificaciones (JCR-ISI)
- E.Ramos, V. Ponomaryov, L. Niño de Rivera, "Optimization of one algorithm for 3D video sequence visualization", *EURASIP Journal on Advances in Signal Processing*, Hindawi Publishing Corporation, 2011. En revisión. (JCR-ISI)
- Edaurdo Ramos and Volodymyr Ponomaryov, "3D color conversion using depth map estimation", *Telecommunications and Radio Engineering*, *Begell House Editors*, vol. 69, issue 5, 2010. (SCOPUS)
- Eduardo Ramos and Volodymyr Ponomaryov, "Generation of 3D video sequences employing depth map information", *Int. Trans. on Computer Science and Engin.*, vol. 61, no.1, 2010. (INSPEC)
- Eduardo Ramos and Volodymyr Ponomaryov, "Reconstruction of three dimensional objects using video sequences" *2010 Int. Kharkov Symp. on Physics and Eng. of Microwaves, Millimeter and Submill. Waves, MSMW 2010*, 2010 (Proceedings ISI)
- E. Ramos, M. Cruz Irisson, L. Nino de Rivera, V. Ponomaryov, "3D color video conversion from 2D video sequence using stereo matching technique", *52nd IEEE Interna-*

tional Midwest Symposium on Circuits and Systems, pp.739-742, 2009. (Proceedings ISI)

- V. Ponomaryov, E. Ramos, "3D video sequence reconstruction algorithms implemented on a DSP", *Proc. of Real-Time Image and Video Processing 2011, SPIE7871*, vol. 7871, 2011. (Proceedings ISI).

.....Rev. Fac. Ing. Univ. Antioquia N.º 56 pp. 111-121. Diciembre, 2010

Reconstruction of 3d video from 2d real-life sequences

Reconstrucción de video 3d desde secuencias reales en 2d

Eduardo Ramos Diaz, Volodymyr Ponomaryov*

National Polytechnic Institute of Mexico, E.S.I.M.E.-Culhuacan, Av. Santa Ana N.º 1000, Col. San Francisco Culhuacán, C.P.04430, Mexico City, Mexico

(Recibido el 27 de junio de 2009. Aceptado el 18 de mayo de 2010)

Abstract

In this paper, a novel method that permits to generate 3D video sequences using 2D real-life sequences is proposed. Reconstruction of 3D video sequence is realized using depth map computation and anaglyph synthesis. The depth map is formed employing the stereo matching technique based on global error energy minimization with smoothing functions. The anaglyph construction is implemented using the red component alignment interpolating the previously formed depth map. Additionally, the depth map transformation is realized in order to reduce the dynamic range of the disparity values, minimizing ghosting and enhancing color preservation. Several real-life color video sequences that contain different types of motions, such as translational, rotational, zoom and combination of previous ones are used demonstrating good visual performance of the proposed 3D video sequence reconstruction.

----- *Keywords:* Video sequence, anaglyph, depth map, dynamic range

Resumen

En este artículo, se propone un método novedoso que permite generar secuencias de video en 3D usando secuencias de video reales en 2D. La reconstrucción de la secuencia de video en 3D se realiza usando el cálculo del mapa de profundidad y la síntesis de anaglifos. El mapa de profundidad es formado usando la técnica de correspondencia estéreo basada en la minimización de la energía de error global a partir de funciones de suavizado. La construcción del anaglifo es realizada usando la alineación del componente de color interpolándolo con el mapa de profundidad previamente formado. Adicionalmente, se emplea la transformación del mapa de profundidad para

* Autor de correspondencia: teléfono: 52 + 55 + 565 620 58, fax: + 52 + 55 + 565 620 58, correo electrónico: eramos@ieee.org. (E. Ramos)

reducir el rango dinámico de los valores de disparidad, minimizando el efecto fantasma mejorando la preservación de color. Se usaron numerosas secuencias de video a color reales que contienen diferentes tipos de movimientos como traslacional, rotacional, acercamiento, y la combinación de los anteriores, demostrando buen funcionamiento visual de la reconstrucción de secuencias de video en 3D propuesta.

----- **Palabras clave:** Secuencia de video, Anaglifo, Mapa de profundidad, Rango dinámico

Introduction

In recent years, we are seeing great advances in the development of different techniques to reconstruct 3D video sequences from 2D ones. Significant advances have also occurred in the development of stereoscopic displays. Such efforts enable the 3D viewing and multi-view autostereoscopic displays that show more than two view points for a given scene [1, 2].

While the advances above mentioned are more evident, the problem of content generation still lingers. Additionally, stereoscopic displays are still expensive in comparison to conventional displays.

One possible solution to this problem is conversion of 2D to 3D video using optical flow, depth map, or other techniques to extract image information that should be used later to reconstruct the 3D video sequence.

Anaglyph is considered the simplest way to obtain 3D perception and have less computational cost in comparison with other methods, such as Photoshop, and the least squares algorithm [3].

In order to construct anaglyphs, the simplest method relies on simple time delay between frames and adjustment of left-right images [4].

Usually, to obtain anaglyphs, the left view in blue (or green) color is superimposed on the same image with the right view in red one. When viewed through spectacles of corresponding colors but reversed, the three-dimensional effect can be perceived.

Anaglyph based 3D projection is considered as the cheapest stereoscopic projection technique, because it can be implemented at television sets or computer screens with no special hardware different than inexpensive colored glasses. In this case, it can be used a color anaglyph based on the Photoshop for red-cyan glasses presented, for example as in [3].

Recently proposed in [3] approach can be applied that employs the adjacent frames from video sequence in order to compute depth maps. The resulting depth map is used to generate synthetic views. These depth maps can be also employed to reconstruct 3D images, because they contain the information on the 3D shape of the scene.

Other simple algorithm to construct anaglyphs is presented in [5], where the anaglyph is realized in the depth map acquisition from MPEG-4 protocol.

There are several methods to compute depth maps from a stereo pair or via adjustment of the video frames, among them, the optical flow algorithms, matching algorithms, etc. The major drawback of all these methods is that they require intensive computational operations.

In papers [6, 7], the differential techniques Lucas and Kanade, and Horn and Shunck to compute optical flow are presented, where it has been implemented a weighted least squares fit of local first order constraints to a constant model for motion map in each small spatial neighborhood.

In papers [8, 9], the energy-based method on the output energy of velocity-tuned filters is

applied, where the local energy is extracted using twelve Gabor-energy filters of several spatial scales, tuned to different spatial orientations and different temporal frequencies. Calculating a dense depth map is basically consisted of finding the correspondence between a pixel location in one frame and its position in the other one. This mapping of an image to other one can be obtained by registering a spatial neighborhood, surrounding each a pixel in an image to another one.

Stereovision technique is one of the methods yielding depth information of the scene. It uses the stereo image pairs from two cameras or a video sequence [10] to produce disparity maps that can be easily turned into depth maps.

Most disparity estimation algorithms that exist are pixel-based ones [10-12], in which the disparities are estimated pixel-by-pixel, and that is one a common feature for various otherwise quite different disparity estimation algorithms. The main problem associated with pixel-based approaches is that they cannot effectively handle un-textured surfaces.

In work [13], the mean shift segmentation algorithm is used to segment the images into different regions, then, the depth map is calculated.

Region based disparity estimation uses the stereo image pairs from two cameras to produce disparity maps that can be easily turned into depth maps. Reliability of depth maps or optical flow computation and computational cost of algorithm is key issue for implementing robust applications.

In this work, a novel algorithm to reconstruct 3D video sequences from 2D information is presented.

The paper is organized as follows. The depth map computation, the anaglyph generation and improvement, and data used are presented in section 1. In section 2, we expose the simulation results for depth map, anaglyph synthesis, anaglyph enhancement, and a brief discussion of the results. Finally, the conclusion of this work is presented in section 3.

Methodology

In order to obtain the 3D video, the following steps are should be employed: depth map computation, anaglyph construction, anaglyph enhancement and finally, the 3D video construction from anaglyph obtained. Below each a step is described in details.

Depth map computation

Using the classic definition of stereo projection, the depth can be calculated as follows:

$$Z(i, j) = f \cdot \frac{T}{d(i, j)} \quad (1)$$

Where T is the distance between camera lens, f is the focal length, and $d(i, j)$ is the disparity map between two images.

We use the method proposed in paper [14], which is based on the Global Error Energy Minimization by Smoothing Functions.

To calculate error energy matrix for every disparity value, the block matching technique has been implemented using different window sizes.

Let denote left image as $L(i, j, c)$ and right one as $R(i, j, c)$, each one in RGB format, so, the error energy can be written as:

$$e(i, j, d) = \frac{1}{3 \cdot n \cdot m} \sum_{x=i}^{i+n} \sum_{y=j}^{j+m} \sum_{k=1}^3 (L(x, y + d, k) - R(x, y, k))^2 \quad (2)$$

Where d is the disparity, c represents color components in an image, m and n are the rows and columns of an image.

For a predetermined range of disparity search range, an averaging filter

$$\bar{e}(i, j, d) = \frac{1}{n \cdot m} \sum_{x=i}^{i+n} \sum_{y=j}^{j+m} e(x, y, d) \quad (3)$$

is applied several times to smooth the error in the energy matrix of disparity removing very sharp

changes in energy, which possibly belongs to incorrect matching.

We select the disparity that has the minimum error energy as the most reliable disparity estimation of the disparity map. So, the error energy of disparity maps can be written as:

$$E_d(i, j) = \frac{1}{3 \cdot n \cdot m} \sum_{x=i}^{i+n} \sum_{y=j}^{j+m} \sum_{k=1}^3 (L(x, y + d(i, j), k) - R(x, y, k))^2 \quad (4)$$

and the reliability of the obtained disparity map is calculated as follows:

$$R_d = \frac{1}{S_d} \left(\sum_{(i,j) - E_d(i,j) = ne}^{n,m} E_d(i, j) \right) \quad (5)$$

where S_d is the number of points in error energy, which are no-estimated (ne).

Disparity map contains some unreliable disparity estimation for some points, so, to avoid this, a threshold should be applied in such a form:

$$\tilde{d}(i, j) = \begin{cases} d(i, j), & E_d(i, j) \leq V_e \\ 0, & E_d(i, j) > V_e \end{cases} \quad (6)$$

$$\tilde{E}_d(i, j) = \begin{cases} E_d(i, j), & E_d(i, j) \leq V_e \\ ne & E_d(i, j) > V_e \end{cases}$$

Here, we use V_e as an error energy threshold, that limits the disparity map, and is found as: $V_e = \alpha \cdot \text{Mean}(E_d)$, where α is the tolerance coefficient to adjust the filtering process reliability, and $0 < \alpha < 1$.

Anaglyph generation and enhancement

Once the disparity map is computed [15], we can construct the depth map as follows:

$$\text{Depth Map}(i, j) = c \sqrt{MV_x^2(i, j) + MV_y^2(i, j)} \quad (7)$$

where c is used as a scaled parameter to fits the maximal disparity over the frames; $MV(i, j)_x^2$ and $MV(i, j)_y^2$ are the X and Y motion vectors values for each a pixel.

Classical methods, such as Photoshop, least squares, etc. which are used in anaglyph construction, always produce ghosting effects and color losses. Therefore, dynamic range reduction of the depth map values could be employed to minimize mentioned drawbacks. Using the P-th law transformation for dynamic range compression [16], the original depth map D is changed as follows:

$$D_{new} = a \cdot D^P \quad (8)$$

where D_{new} is the new depth map pixel value, $0 < a < 1$ is a normalizing constant, and $0 < P < 1$. The dynamic range compression permits retaining the depth ordering information within the depth map, while reducing ghost effects in the non-overlapping areas in the anaglyph.

Anaglyph synthesis requires generating a new stereo pair from one of the images and the modified map, so, the initial stereo pair image has to be re-sampled in a grid dictated by the depth map. This process should be realized via an interpolation. Some interpolation methods, such as bilinear, sp-line and nearest neighbor are tested experimentally, but in exposed experiments we only use the nearest neighbor interpolation as a promising one according to results obtained.

The nearest neighbor interpolation uses the neighborhoods defined by pixels (u, v) , $(u, v+1)$, $(u+1, v)$, $(u+1, v+1)$ around the point (X, Y) . Among the four pixels, the one closest to (X, Y) is determined, and its intensity is used as intensity for (X, Y) point. A flow chart of design the 3D video sequence proposed here is exposed in figure 1. Firstly, the video sequence decomposition, in order to obtain frames, is arranged into pairs; secondly, each a color component should be separated. Then, it is necessary to compute depth map and realize the dynamic range manipulation in order to reduce

ghosting effects; therefore, the nearest neighbor interpolation for the compressed depth map and red component in each a pair of the frames is required. The interpolation method provides the color preservation and low computational cost in the video processing. Finally, the reconstructed 3D video is formed. We apply all process presented below to each an image pair through a video sequence.

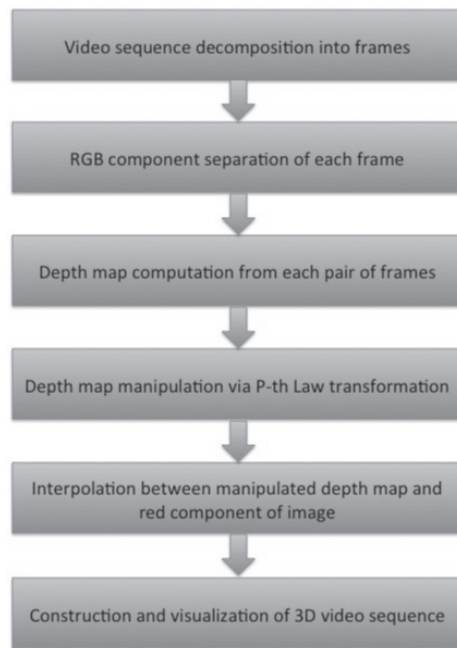


Figure 1 Flow chart to obtain 3D video sequence

Data used

In the simulation experiments, different real-life color video sequences, such as *Coastguard*, *Flowers*, *Foreman*, *Salesman* and *Alley* are employed in the Avi format. *Coastguard* color video sequence (300 frames, frame size 352 x

288 pixels) presents a particular boat movement from left to right side in the image; the *Flowers* sequence (300 frames, 352 x 288 pixels) exposes the right to left camera movement; in *Foreman* sequence (300 frames, 352 x 288 pixels), one can see a talking man; the *Salesman* sequence (48 frames, 352 x 288 pixels) shows the man that, holding a little box in his hands, realizes some movements; and finally, the *Alley* sequence (50 frames, 336 x 272 pixels) presents a movement as if we are walking in the alley. In figure 2, we present the images extracted from video sequences that used as stereo pair further.

Results and discussion

In this section, the obtained experimental results are presented: the depth map calculation using the proposed method; anaglyph results, which are obtained for the constructed anaglyphs with nearest neighbor interpolation method; and finally, the depth map manipulated via dynamic range is used to the anaglyph reconstruction. Simulation data were acquired varying the parameters in the depth map calculation in order to realize the better perception in 3D. The video sequence was separated in JPEG images, allowing the usage of the frames as stereo pairs.

Depth map

The video sequence is separated into frames and disparity maps are computed for each a frame using the corresponding stereo pair. Then, the corresponding depth map for each a frame from all video sequence is computed using MatlabTM R2009a software with an AMDTM Phenom X3 processor (64 bits) in a PC.

Figure 3 exposes the depth map image, where white pixels represent pixels with movement and black pixels represent pixels without movement into a frame. In all the cases, the obtained depth map of the stereo pair is sufficient for 3D perception purpose.

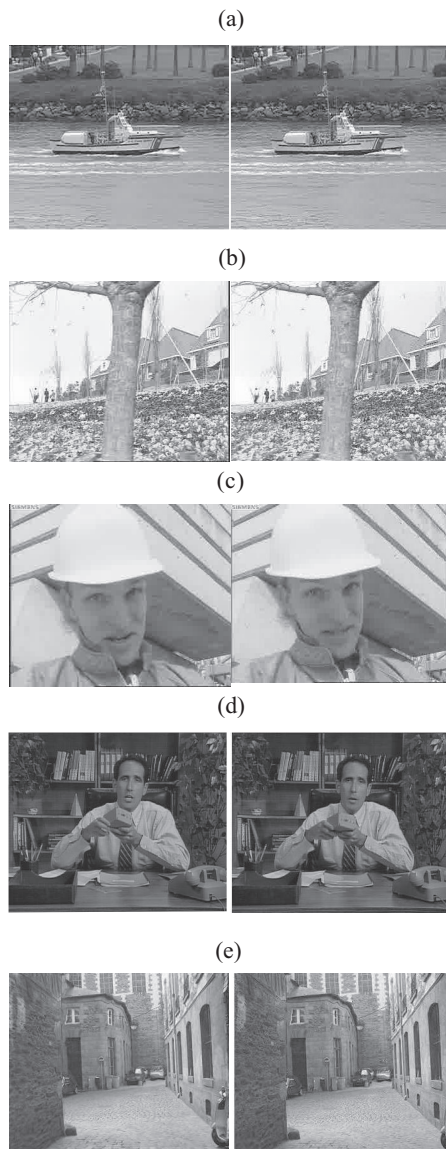


Figure 2 Video frames (neighboring frames used as the stereo pairs) for the following sequences: a) *Coastguard*, b) *Flowers*, c) *Foreman*, d) *Salesman* and e) *Alley*

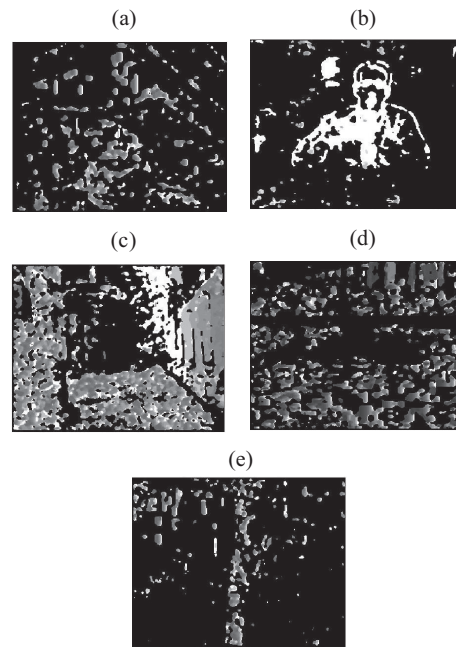


Figure 3 Resulting depth map for the following sequences: a) *Coastguard*, b) *Flowers*, c) *Foreman*, d) *Salesman* and e) *Alley*

Anaglyph synthesis

Once depth map is computed, anaglyph synthesis can be realized by the interpolation of the red component of the right image using the depth map. At the interpolation stage, nearest neighbor method has been employed because it is capable to produce satisfactory results in the visual observation.

Figure 4 presents the anaglyph synthesized for each a video sequence. In left images, the anaglyph construction is exposed using nearest neighbor Interpolation that does not significantly reduce the quality of the resulting 3D video in comparison with other interpolation algorithms [17, 18]. In the right images, the resulting anaglyph with depth map compression is shown, when the P-th law transformation is applied to depth map in order to compress the disparity values [19]. Values

Reconstruction of 3d video from 2d real-life sequences

$P=0.5$ and $a=0.5$ were selected for all the frames in the investigated video sequences. As one can see, the application of the compression improves

the anaglyph and allows better 3D perception, reducing ghosting in the final anaglyph. The selected parameter values are shown in table 1.

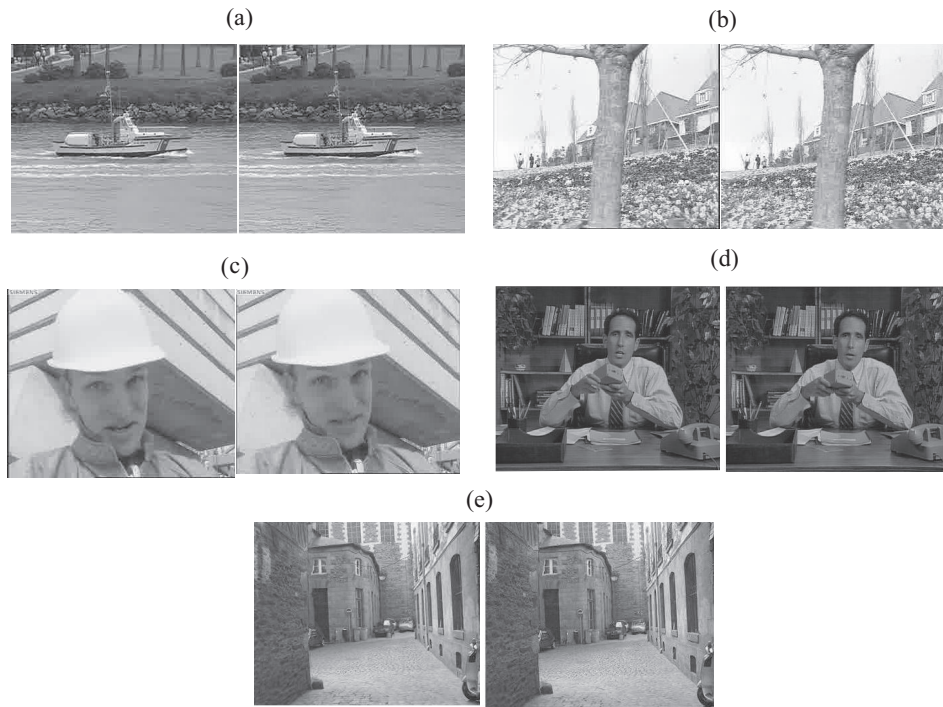


Figure 4 Synthesized Anaglyph without depth map manipulated (left images), and synthesized anaglyph with depth map adjusted using P-th law compression (right images) for: a) *Coastguard*, b) *Flowers*, c) *Foreman*, d) *Salesman* and e) *Alley*.

Table 1 Selected parameter values for each a video sequence and calculating time

Sequence	Alpha	Disparity threshold	Disparity map	Disparity map filtered	Focal lenght	Calculating time (sec)
Coastguard	1	2.2	0.0057	0.0022	40	26.09
Flowers	0.9	2.0	0.0036	0.00081	40	21.89
Foreman	0.9	2.0	0.030	0.0042	40	14.80
Salesman	1	5.0	0.051	0.013	40	14.72
Alley	1	0.6	0.003	0.0011	60	13.11

Calculating time is obtained since the video capture even the anaglyph visualization in a standard monitor.

In order to compare the proposed technique with a commonly used one in reconstruction of 3D video sequences, we implement the Lucas & Kanade classic differential technique to compute optical flow [8], constructing the depth map according to equation (7).

The figure 5 presents the depth map results using Lucas and Kanade technique justifying

the possibility to construct the similar depth map employing the Lucas and Kanade method, where one can see its drawback, missing of some details, which lead to worse results in 3D view using anaglyphs. Depth map computation via region based stereo matching calculation contains more important motion elements that are not presented in Lucas and Kanade algorithm, so, this justifies that the proposed algorithm allows better 3D perception without significant loss information extracted from stereo pair.

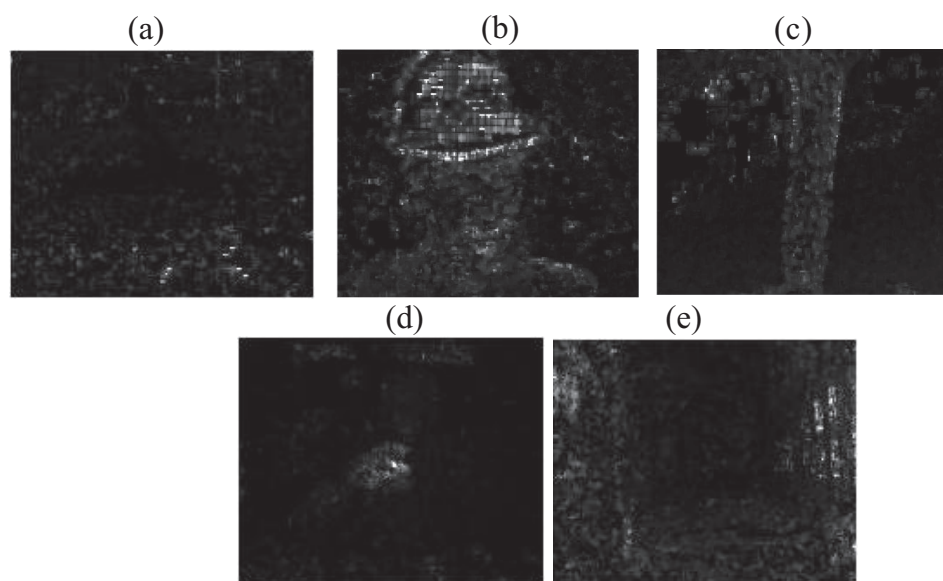


Figure 5 Depth Map computation using Lucas & Kanade technique for: a) *Coastguard*, b) *Flowers*, c) *Foreman*, d) *Salesman* and e) *Alley*

Additionally, in order to verify that the 3D perception is worth in the case of the Lucas and Kanade depth map computation, we used it to synthesize the corresponding anaglyph. The figure 6 shows the obtained results.

Observing the figure 6 with spectacles, the ghosting effects and low 3D perception are easy observed in

That can be explained as a consequence of poor details' reconstruction in the depth map formed by optical flow technique.

In order to compare the proposed method to the anaglyph reconstruction, we implemented the original Photoshop (PS) algorithm presented in [20] exposing the results in figure 7.

Reconstruction of 3d video from 2d real-life sequences

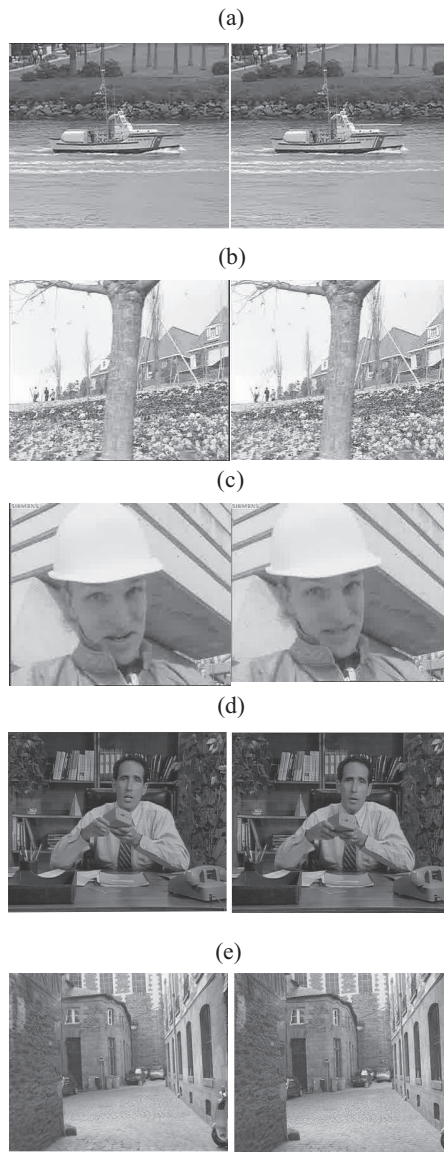


Figure 6 Anaglyph synthesized in case of use the Lucas & Kanade technique for video sequences: a) *Coastguard*, b) *Flowers*, c) *Foreman*, d) *Salesman* and e) *Alley*

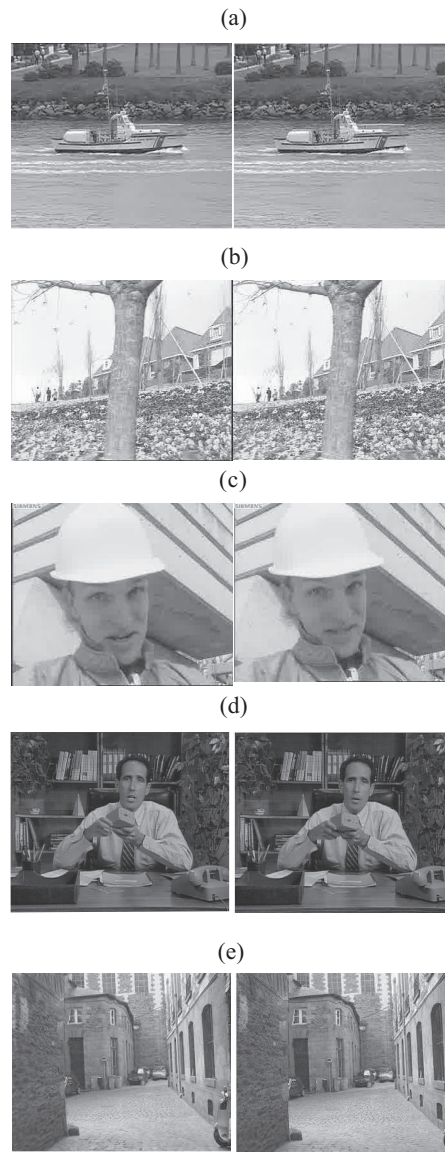


Figure 7 Anaglyph synthesized using the original Photoshop algorithm for video sequences: a) *Coastguard*, b) *Flowers*, c) *Foreman*, d) *Salesman* and e) *Alley*

It can be concluded analyzing this figure that PS algorithm for anaglyph construction allows sufficiently acceptable 3D perception in the video sequences, but more ghosting effects can be seen here in comparison with the proposed method results (see Figure 4). Additionally, we found that the time processing values for Photoshop algorithm are much more than in the case of the proposed algorithm usage.

Conclusions

In this paper, a novel method in the design of the high quality anaglyphs from color video sequences is introduced. The proposed method presents more precise motion information permitting better 3D perception. Anaglyphs are used as visualization technique viewing 3D information due to its simpler and cheaper way to display 3D information. Region based stereo matching calculation provides more reliable depth map in comparison with commonly used algorithms to obtain depth map like as Lucas & Kanade implementation.

Depth map interpolation for red component data allows the anaglyph construction and depth map compression via dynamic range computation permitting to realize better 3D perception due reduction the ghosting in anaglyphs, as it has been shown in the presented images. The proposed method has shown its efficiency in construction 3D color video sequences from 2D ones that are justified in color video sequences with different types of movements.

Acknowledgements

The authors thank National Polytechnic Institute of Mexico and CONACYT (grant 81599) for their support.

References

1. B. Blundell, A. Schwartz. *Volumetric three dimensional display systems*. Ed. Wiley. New York. Vol. 5. 2000. pp. 196-200.
2. M. Halle. "Autostereoscopic displays and computer graphics". *Comput Graph*. Vol. 31. 1997. pp. 58-62.
3. E. Dubois. "A projection method to generate anaglyph stereo". *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 3. 2001. pp. 1661-1664.
4. I. Ideses, L. Yaroslavsky. "A method for generating 3D video from a single video stream". *Proc of the Vision, Modeling and Visualization*. Ed. Aka Gmbh. Erlangen. Germany. 2002. pp. 435-438.
5. I. Ideses, L. Yaroslavsky. "New Methods to produce high quality color anaglyphs for 3-D visualization". *Lecture Notes in Computer Science*. Vol. 3212. 2004. pp. 273-380.
6. S. S. Beauchemin, J. L. Barron. "The computation of optical flow". *ACM Computing surveys*. Vol. 27. 1995. pp. 436-466.
7. J. L. Barron, D. J. Fleet, S. S. Beauchemin. "Performance of optical flow techniques". *International Journal of Computer Vision*. Vol. 12. 1994. pp. 43-77.
8. D. J. Heeger. "Model for the extraction of image flow". *Journal of the Optical Society of America*. Vol. 4. 1987. pp. 1455-1471.
9. D. J. Heeger. "Optical flow using spatiotemporal filters". *International Journal Computer Vision*. Vol. 1. 1988. pp. 279-302.
10. X. Huang, E. Dubois. "3D reconstruction based on a hybrid disparity estimation algorithm". *IEEE International Conference on Image Processing*. Vol. 8. 2006. pp. 1025-1028.
11. C. Zitnick, T. Kanade. "A cooperative algorithm for stereo matching and occlusion detection". *Robotics Institute Technical Reports*. No. CMU.RI-TR-99-35. Carnegie Mellon University. 1999.
12. H. H. Barker, T. O. Binford. "Depth from edge and intensity based stereo". *Proc. of the 7th International Joint Conference on Artificial Intelligence*. Vancouver. 1981. pp. 631-636.
13. D. Comaniciu, P. Meer. "Mean-shift: a robust approach toward feature space analysis". *IEEE Transactions Pattern Anal Machine Intelligence*. Vol. 24. 2002. pp. 603-619.
14. B. B. Alagoz. "Obtaining depth maps from color images by region based stereo matching algorithms". *OncuBilim Algorithm and Systems Labs*. Vol. 08. 2008. Art.4. pp. 1-12.

..... Reconstruction of 3d video from 2d real-life sequences

15. I. Ideses, L. Yaroslavsky, B. Fishbain. "Real time 2D to 3D video conversion". *J. Real Time Image Proc.* Vol. 2. 2007. pp. 3-9.
16. I. Ideses, L. Yaroslavsky. "3 methods to improve quality of color anaglyph". *J. Optics A. Pure, Applied Optics.* Vol. 7. 2005. pp. 755-762.
17. I. Ideses, L. Yaroslavsky, B. Fishbain, R. Vistuch. "3D compressed from 2D video". *Stereoscopic displays and virtual reality systems XIV in Proc. of SPIE & IS&T Electronic Imaging.* Vol. 6490. 2007. pp. 64901C.
18. L. Yaroslavsky, J. Campos, M. Espinola, I. Ideses. "Redundancy of stereoscopic images: Experimental evaluation". *Optics Express.* Vol. 13. 2005. pp. 0895-10907.
19. L. Yaroslavsky, *Holography and digital image processing*, Ed. Kluwer Academic Publishers. Boston. 2004. pp. 600.
20. W. Sanders, D. McAllister. *Producing anaglyphs form synthetic images*. Conf. 5005: A stereoscopic displays and applications XIV. Proc. SPIE/IS&T. Vol. 5006. 2003. pp. 348-358.

Telecommunications and Radio Engineering, 69(5):459-471 (2010)

3D COLOR VIDEO CONVERSION USING DEPTH MAP ESTIMATION

E. Ramos & V. Ponomaryov

National Polytechnic Institute of Mexico, Mexico-city, Mexico

*Address all correspondence to V. Ponomaryov E-mail: kvf@pochta.ru

Color video algorithm, which permits to convert 2D video sequence to 3D one, is presented. Optical flow approach based on differential technique and warping theory is used, obtaining the disparity estimation from two frames in a color video sequence, and finally, the estimation of the depth map. Formed depth map is employed in construction of 3D video sequence by means of an anaglyph employment. The depth map manipulation via P-th law compression is used permitting to avoid ghosting in anaglyph synthesis. The results of the proposed 3D reconstruction method are compared with the same obtained by Photoshop algorithm demonstrating better performance of the proposal. Numerous simulation experiments with several color video sequences have confirmed good visual performance of novel framework in reconstruction of 3D sequences.

KEY WORDS: *stereovision, anaglyph, depth map, image warping technique*

1. INTRODUCTION

The stereoscopic projection devices have received many attentions permitting their development in recent years [1,2]. By now, they are still expensive and have significant limitations in vision; so, this justifies their marginal acceptance in the market.

Anaglyph based on 3D projection presents one promising solution in developing the stereoscopic projection devices. This device is considerate as the cheapest stereoscopic projection because it can be implemented on standard monitors without significant modifications, using only simple lens for 3D viewing.

Simple algorithm to reconstruct anaglyphs is presented in [3], where the procedure employed is based on the depth map acquisition from MPEG-4 format.

Stereovision technique is one of the methods yielding the depth map of information scene. It uses stereo image pairs from two cameras or frames from a video sequence [4].

There are several methods to compute depth maps from a stereo pair or via adjustment of the video frames, such as optical flow algorithms, matching algorithms, stereo correspondence, etc. [5-7].

Reliable depth maps are important in different areas, such as, stereovision in robotic applications where the 3D robot position in scenery should be found [8].

The mayor drawback of all these methods is that they require intensive computational operations [3].

Actually, there exist several methods to estimate the flow field using the neighbouring sequence frames. For example, in [6,7,9], the differential techniques for computing the velocity from spatial-temporal derivatives of image intensity for filtered image are applied.

In [9], the matching algorithm is employed, in order to maximize a normalized cross correlation used as similarity measure that depends on a distance. Here, the velocity is defined as the shift that yields the best fit between image regions at different times.

In papers [11,12], the energy-based method is applied, where the local filter energy is extracted using twelve Gabor filters of several spatial scales, tuned to different spatial orientations and temporal frequencies.

Recently [13,14], two methods that compute a correspondence measure between pixels in an image stereo pair are presented. In the first method, the stereo correspondence is computed using matching technique, and, in the second one, the wavelet and multi-wavelet analysis are used computing a stereo correspondence. Unfortunately, these techniques still have high computational cost, so, it is difficult to implement them in real time video processing applications.

Calculation of the dense depth map is consisted of finding the correspondence between a pixel location in one frame and its position in other one. This mapping of an image to other one can be realized by registering a spatial neighbourhood, surrounding each a pixel for the image that corresponds to other one.

In this work, the implementation of the depth map computation employing the optical flow techniques, Lucas and Kanade algorithm and Warping is considered. Once the optical flow is being formed, it is used to calculate the corresponding depth map. Additionally, the compression of depth map should be employed in order to avoid ghosting, and finally, the anaglyph and video 3D sequence are constructed.

The paper is organized as follows. Optical flow computation, the depth map conversion and improvement, the anaglyph generation and data used are presented in Section 2. In Section 3, the simulation results for depth map and reconstructed anaglyphs are exposed. Finally, the conclusion is presented in Section 4.

2. THEORY

In the proposed algorithm, the first step consists of the optical flow computation employing the differential and warping techniques. Next, the depth map using the optical flow is calculated, then, dynamic range compression of depth map is realized, and finally, the nearest neighbor interpolation is used to construct an anaglyph. So, this final procedure permits 3D viewing of a video sequence employing simple lens.

2.1 Optical Flow Computation

As it was mentioned, there exist several algorithms to compute optical flow. Here, we investigate three algorithms, first one is a classical algorithm based on the Lucas and Kanade method, the second one is the Horn and Schunck algorithm, and the third algorithm consists of the optical flow estimation based on a theory of warping with additional constraints to obtain the better quality of the flow field.

2.2 Optical Flow Based on the Lucas and Kanade Algorithm

A common solution in the computing the optical flow is to assume that the total spatial and temporal derivatives of the image brightness remain constant [6], then, the velocity vector can be found as: $v = (u, w) = (dx/dt, dy/dt)$.

The first order approximation of the presented assumption gives the following:

$$\frac{\partial E}{\partial t}u + \frac{\partial E}{\partial y}w + \frac{\partial E}{\partial t} = 0, \quad (1)$$

where the spatial and temporal derivatives should be estimated between two images or two frames in a video sequence.

According to [8], the weighted least squares are fitted using local first order constraints to a constant model, that can be implemented for v in each a small spatial neighborhood β minimizing the following functional:

$$\sum_{x \in \beta} W^2(x) [\nabla I(x, t) \cdot v + I_t(x, t)]^2, \quad (2)$$

where $W(x)$ is a window function that gives more influence to constraints at the centre of the neighborhood than those at the periphery, $I_t(x, t)$ denotes the partial temporal derivative, and $\nabla I(x, t) = (I_x(x, t), I_y(x, t))^T$.

The solution of the equation (2) is written as:

$$v = [A^T W^2 A]^{-1} A^T W^2 b, \quad (3)$$

where

$$A = [\nabla I(x_1), \dots, \nabla I(x_n)]^T, \quad W = \text{diag}[W(x_1), \dots, W(x_n)],$$

$$b = -(I_t(x_1), \dots, I_t(x_n))^T.$$

2.3 Optical Flow Estimation Based on the Horn and Schunck Algorithm

Horn and Schunck [7] combined the gradient constant with a global smoothness term to constrain the estimated velocity field $v(x, t) = (u(x, t), (v(x, t)))$, minimizing the following equation:

$$\int_D (\nabla I \cdot v + I_t)^2 + \lambda^2 (\|\nabla u\|_2^2 + \|\nabla v\|_2^2) dx, \quad (4)$$

where λ is the influence of the smoothness term.

Iterative equations are used to minimize the equation (4) obtaining the image velocity:

$$u^{k+1} = u^k - \frac{I_x [I_x \bar{u}^k + I_y \bar{v}^k + I_t]}{\alpha^2 + I_x^2 + I_y^2}, \quad (5)$$

$$v^{k+1} = v^k - \frac{I_y [I_x \bar{u}^k + I_y \bar{v}^k + I_t]}{\alpha^2 + I_x^2 + I_y^2}, \quad (6)$$

where k denotes the iteration number, u^0 and v^0 denote the initial velocity estimates, finally $\bar{u}^k$ and $\bar{v}^k$ are the neighborhood averages of the velocity terms.

2.4 Optical Flow Based on the Warping Theory

The optical flow estimation based on a theory of warping has been proposed in [15], and it combines three assumptions.

The first assumption is brightness constancy [7], in which the model uses the original non-linearized grey value; the second one is the gradient constancy assumption [16] allowing small variations in the grey value obtained from a color image to determine the displacement vector under invariant grey values. Final assumption is the smoothness demand. This constraint can be either applied in the spatial domain where the model estimates locally the displacement of a pixel without taking into account any interaction between the neighboring pixels.

Additionally, if the displacements in the images are larger than one pixel of a frame, usually the multi-scale approach is applied. In order to find the global minimum in a minimization algorithm, it can be useful to apply multi-scale solutions [10]: starts with solving a coarse solution of the problem by working on the smoothed image sequence.

The coarse solution is taken as initialization step to find the refined solution of the optical flow estimation in iterative form. In the coarse solution, the main goal is to derive energy functional that penalizes deviations from these model assumptions.

Let denote the variables $x = (x, y, t)^T$ and $w = (u, v, 1)^T$. The global deviations from the grey value constancy and the gradient constancy assumptions are measured by energy [17], and can be presented as:

$$E_{data}(u, v) = \int_{\Omega} \Psi \left((I(x+w) - I(x))^2 + \gamma (\nabla I(x+w) - \nabla I(x))^2 \right) dx, \quad (4)$$

where γ is a weight between two assumptions and $\Psi(s^2) = \sqrt{s^2 + \varepsilon^2}$ is a concave function, which results in L_1 minimization.

A smoothness term [18] that describes the model assumption for a piecewise smooth flow field is written as:

$$E_{smooth}(u, v) = \int_{\Omega} \Psi \left(|\nabla_3 u|^2 + |\nabla_3 v|^2 \right) dx. \quad (5)$$

The total energy is the weighted sum of two terms: $E_{total}(u, v) = E_{data} + \alpha E_{smooth}$, where a regularization parameter $\alpha > 0$.

2.5 Depth Map Conversion and Manipulation

To compute the depth map, we use the following implementations: optical flow based on Lucas & Kanade and warping techniques, constructing the depth map as follows [3]:

$$DepthMap = k \sqrt{MV(i, j)_x^2 + MV(i, j)_y^2}, \quad (6)$$

where $MV(i, j)_x^2$ and $MV(i, j)_y^2$ are the movement vectors and k is the tolerance coefficient (near to 1) and is selected in order to avoid erode depth map due of eliminating more disparity points in the map.

Computed anaglyph usually presents ghosting artifacts because of variations of depth fields in an image. As a result of such artifacts, 3D perception of the images and video sequence can be lost. The analysis shows that ghosting artifact is more perceptible when dynamic range of the depth is high, so, a depth map manipulation to reduce the dynamic range of 3D information should be performed in order to avoid this drawback and improve the quality of the anaglyphs with a high depth value. The *Pth-law* transformation [3] is a standard technique that is used to manipulate the dynamic range compression, and it can be written in such a form:

$$\tilde{D}(i, j) = aD^p(i, j), \quad (7)$$

where $\tilde{D}(i, j)$ is the modified depth map, D is the original depth map, and a is a normalization value.

2.6 Color Anaglyph Generation

Anaglyph synthesis consists of the following: the red channel in one image is replaced by the red channel of second neighboring image of stereo pair [19], or neighboring frame from the video sequence. Anaglyphs produce a visual effect of three-dimensional perception when the images are viewed by color filter lens that are usually constructed with red and blue filter, where the red filter is usually placed in left position in the spectacles. In order to form the anaglyph, different methods that rely on image re-sampling are used. Nearest neighbor interpolation is usually employed, for example as in the MPEG format.

This type of interpolation does not significantly reduce the quality of the resulting anaglyph and does not require extra computation [20].

The nearest neighbor interpolation [21] uses the neighborhoods defined by pixels $(u, v), (u, v+1), (u+1, v), (u+1, v+1)$ in vicinity of a point (X, Y) . Analyzing four pixels, the one closest to (X, Y) is determined, and its intensity is used as the intensity for (X, Y) point.

There exist some basic techniques to construct anaglyphs: Photoshop, the midpoint and Dubois algorithms [22]. In this paper, we compare the proposed method that constructs 3D video sequences with classical Photoshop algorithm implemented here, too. The red channel of the left eye view becomes the red channel of the anaglyph and vice versa for blue and green channels of the right eye; so, this is equivalent to projection of the left eye RGB point to the red axis of the RGB cube.

The anaglyph is constructed using the following equation

$$\begin{bmatrix} r, g, b \end{bmatrix}^T = Bv, \quad (9)$$

where r, g, b are the color components of the resulting image, $v = \begin{bmatrix} r_l, g_l, b_l, r_r, g_r, b_r \end{bmatrix}^T$, where r_l, g_l, b_l are the components of the left and r_r, g_r, b_r are the components of the right images, finally, matrix

$$B = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

The resulting vector presented in (9) is added to compute the color components of a pixel in the resulting anaglyph.

2.7 Data

In simulation experiments, different color video sequences were used: *Ball1*, *Ball2* as the synthetic sequences, and *Chair*, *Salesman* and *Bus* as the real-life sequences. All these sequences were employed in Avi format. The *Ball1* and *Ball2* color video synthetic sequences contain 150 frames with 320x240 pixels in each one. In first sequence, a bouncing ball is coming from back to front into the sequence. In *Ball2* sequence, a bouncing ball is coming from back to front, and the camera has a rotational movement from left to right.

The *Chair* color video sequence (42 frames, 320 x 240 pixels) presents a particular chair movement from left to right side in the image. In the *Salesman* sequence (48 frames, 352 x 288 pixels), the man, who realizes some movements, holding a little box in his hands, is exposed.

The *Bus* sequence (50 frames, 486 x 355 pixels) presents a bus with movement from right to left frame side. Camera realizes the rotational movement, so this sequence has complex movements, that why, it is more difficult to reconstruct 3D video in this case.

3. SIMULATION RESULTS

Numerous simulation results were obtained varying the parameters during the depth map reconstruction in order to select the better perception in 3D. All video sequences are separated in JPEG images, so, the adjacent frames were used as stereo pairs.

The simulation results are exposed in form of reconstructed depth maps, and finally, presenting the anaglyphs obtained via applications of the different interpolation methods.

3.1 Depth Map Computation

Tables 1-3 expose the optimal values of the parameters: window size, time needed to process, and k , α , γ , etc. used in implementation of the algorithms permitting the better depth map quality in 3D reconstruction of video sequences. One can see that values of processing time needed are less when the algorithm of the warping theory is used.

Figures 1-3 expose the depth map for analyzed techniques. In the depth map, the white pixels represent the movement of the pixels and black pixels show the pixels without any movement in the frames.

Observing details presented in the depth maps shown in Fig. 1, one can see that the Lucas & Kanade algorithm presents the depth maps with worst quality. Better quality, as in Fig. 3, for depth map guarantees better 3D perception in the reconstructed video sequence.

466

Ramos & Ponomaryov

TABLE 1: Parameters for the Lucas & Kanade technique implementation

Sequence	Window size, pixels	k value	Spent time, s	Sequence	Iterations	α value	Spent time, s
Ball 1	2	0.5	7.80	Ball 1	100	1	5.41
Ball 2	2	0.5	9.70	Ball 2	100	1	5.07
Chair	2	0.8	7.98	Chair	100	1	5.29
Salesman	2	0.5	10.80	Salesman	100	1	5.16
Bus	2	0.8	10.09	Bus	100	1	5.34

TABLE 3: Parameters for the Theory of Warping implementation

Sequence	α	γ	Levels	k value	Spent time, s
Ball1	30	80	20	0.5	5.90
Ball2	30	80	20	0.5	6.30
Chair	30	80	20	0.5	5.80
Salesman	30	80	20	0.5	5.13
Bus	30	80	20	0.5	8.10

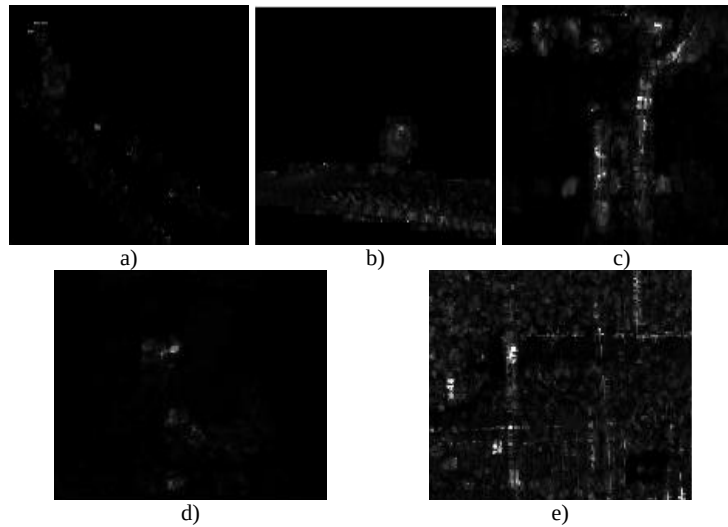


FIG. 1: Depth map obtained using Lucas and Kanade algorithm for: a) *Ball1*, b) *Ball2*, c) *Chair*, d) *Salesman*, e) *Bus*

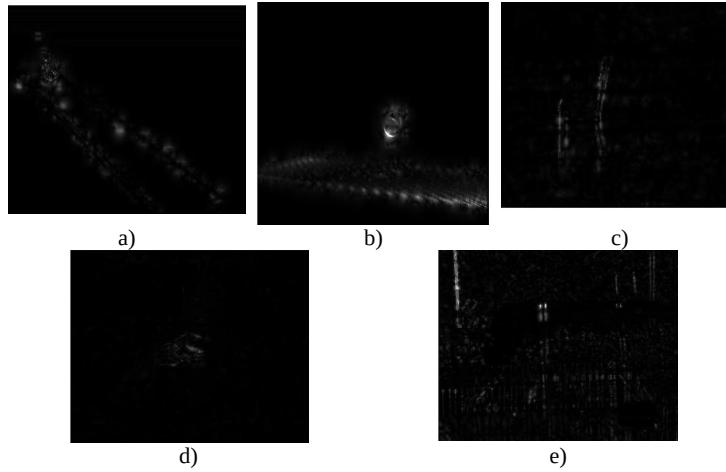


FIG. 2: Depth map obtained using Horn and Schunck algorithm for: a) *Ball1*, b) *Ball2*, c) *Chair*, d) *Salesman*, e) *Bus*

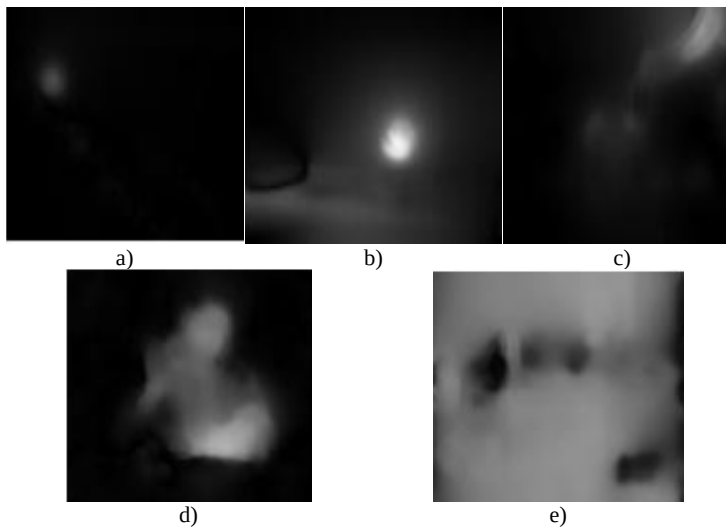


FIG. 3: Depth map obtained using Theory of Warping algorithm for: a) *Ball1*, b) *Ball2*, c) *Chair*, d) *Salesman*, e) *Bus*

468

Ramos & Ponomaryov

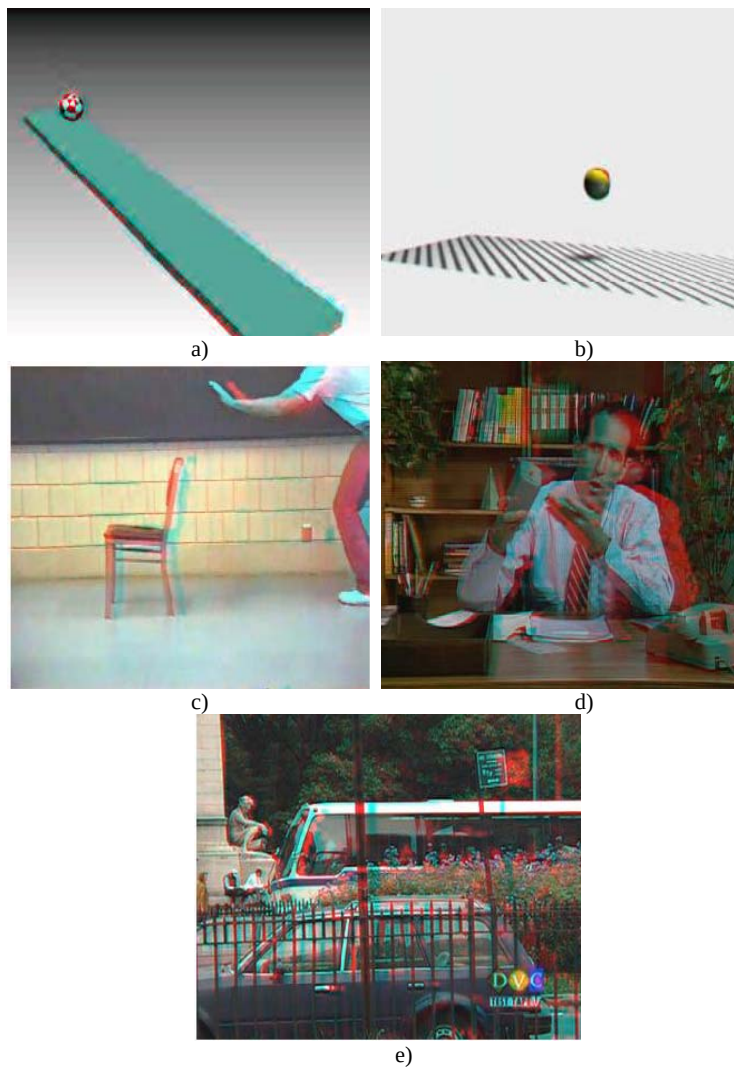


FIG. 4: Anaglyph synthesized using Theory of warping algorithm for: a) *Ball1* video sequence, b) *Ball2* video sequence, c) *Chair* video sequence, d) *Salesman* video sequence, e) *Bus* sequence

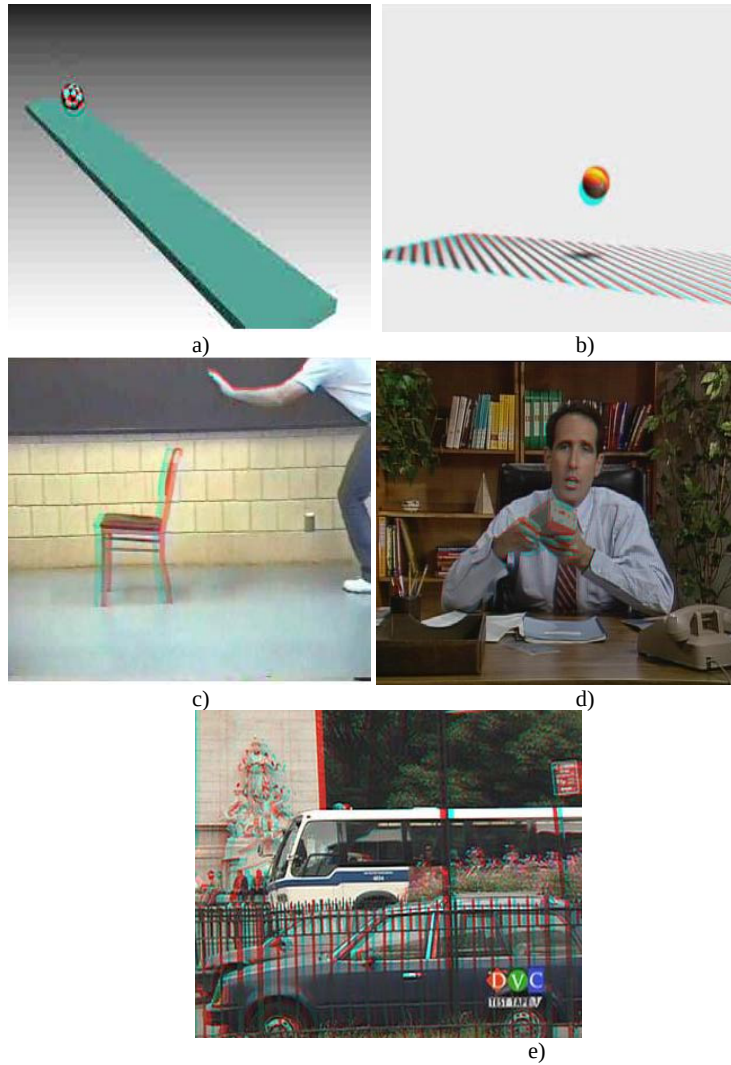


FIG. 5: Anaglyph synthesized using the Photoshop algorithm for: a) *Ball1* video sequence, b) *Ball2* video sequence, c) *Chair* video sequence, d) *Salesman* video sequence, e) *Bus* sequence

3.2 Anaglyph Construction

Two different methods for depth map computation presented below in the anaglyph construction are used. The nearest neighbor interpolation for red component that presents the better quality in the reconstructed 3D video sequence is employed, too. Additionally, the reconstruction results employing the Photoshop algorithm are presented.

The reconstruction process has been realized in all frames for analyzed sequences. In Figs. 4 and 5, we can see as an example the anaglyph result for only one frame from complete 3D sequences reconstructed in simulation experiments. Analyzing the anaglyphs, one can observe the importance to have reliable depth maps in order to avoid ghosting and color degradation. We only present the anaglyphs constructed employing the warping theory and Photoshop methods because they present better 3D perception in the anaglyph construction in comparison with the other implementations.

4. CONCLUSION

It has been investigated and implemented three methods for depth map estimation: Lucas & Kanade, Horn & Schunck and Warping techniques. The simulation results have shown that the warping method works globally better allowing the 3D perception with low computational cost.

Warping technique has better 3D perception but the computational charge increases significantly.

Synthesizing color anaglyphs is employed as the visualizing method due to its lowest cost in the stereoscopic projection techniques. Proposed method supports also 3D video conversion for video sequence with complex movements, such as Bus sequence.

Ghosting effects and loss of color quality appear present minimal values in comparison with other implementations presented in this work.

The presented method exposes better 3D perception and fewer artifacts than classical Photoshop algorithm.

In future, the fuzzy logic technique at anaglyph reconstruction stage can be used to improve the current results; also, the implementation on DSP or FPGA seems reliable because the investigated methods have low computational charge.

5. ACKNOWLEDGEMENTS

The authors would thank to the National Polytechnic Institute of Mexico and CONACYT (project 81599) for their support to realize this work.

REFERENCES

1. Halle, M., (2007), Auto-stereoscopic displays and computer graphics, *Compt. Graphics, ACM*, **31**(2):58-62.
2. Blundell, B. and Schwartz, A., (2000), *Volumetric three-dimensional display systems*, Wiley Publishers, USA.
3. Ideses, I.A., Yaroslavsky, L.P., Fishbain, B., and Vistuch, R., (2007), 3D from Compressed 2D Video, *Proc. of SPIE* 6490, pp 28-35.
4. Kravchenko, V.F., Perez-Meana, H., and Ponomaryov, V.I., (2009), *Adaptive Digital Processing of Multidimensional Signals with Applications*, FizMatLit Edit., Moscow (in Russian).
5. Ideses, I. and Yaroslavsky, L., (2002), A method for generating 3D video from a single video stream. *Vision Modeling and Visualization*, pp. 435-438.
6. Barron, J.L., Fleet, D.J., Beauchemin, S.S., and Burkitt, T., (1992), Performance of optical flow techniques. *IEEE Conf. CVPR*, pp. 236-242.
7. Horn, B.K.P., and Schunk, B.G., (1981), Determining optical flow. *AI* **17**(1-3):185-204.
8. Shkvarko Y.V., (2010), Unifying Experiment Design and Convex Regularization Techniques for Enhanced Imaging With Uncertain Remote Sensing Data-Part I: Theory, *IEEE Trans. on Geos. Rem.* DOI: 10.1109/TGRS.2009.2027695 (in press)
9. Lucas, B. and Kanade, T., (1981), An iterative image registration technique with an application to stereovision. *Proc. DARPA IU Workshop*, pp. 121-130.
10. Anandan, P., (1989), A computational framework and an algorithm for the measurement of visual motion. *Int. J. Comp. Vision*, **2**(3):283-310.
11. Heeger, D.J., (1987), Model for the extraction of image flow. *J. Opt. Soc. Am.*, **4**(8):1455-1471.
12. Heeger, D.J., (1988), Optical flow using spatiotemporal filters, *Int. J. Comp. Vision*, **1**(4):279-302.
13. Alagoz, B.B., (2008), Obtaining depth maps form color images by region based stereo matching algorithms. *OncuBilim Algorithm and Systems Labs*, **8**(4):1-12.
14. Bhatti, A. and Nahavandi, S., (2008), Stereo Correspondence estimation based on wavelets and multi-wavelets analysis. *Stereo vision*, I-Tech, Austria.
15. Brox, T., Bruhn, A., Papenberg N., and Weickert J., (2004), High accuracy optical flow estimation based on a theory for warping. *8th Proc. European on Computer Vision, Lecture Notes on Computer Science* **3024**(4):25-36.
16. Uras, S., Girosi, F., Verri, A., and Torre, V., (1988), A computational approach to motion perception, *Biological Cybernetics*, **60**(2):79-87.
17. Black, M.J. and Anandan, P., (1996), The robust estimation of multiple motions: parametric and piecewise smooth flow fields. *Computer vision and image understanding*, Academic Press, **63**(1):75-104.
18. Cohen, I., (1993), Nonlinear variational method of optical flow computation. *Int. Proc. SCIA*, pp.523-530.
19. Ideses, I.A., Yaroslavsky, L.P., and Fishbain, B., (2007), Real time 2D to 3D video conversion. *J. Real Time Image Proc.*, **2**(1):3-9.
20. Yaroslavsky, L., Campos, J., Espinola, M., and Ideses, I., (2005), Redundancy of stereoscopy images: experimental evaluation, *Opt. Express*, **13**(26):10895-10907.
21. Goshtasby, A., (2005), *2D and 3D image registration*, Wiley Publishers, USA.
22. Sanders W. and McAllister D., (2003), Producing anaglyphs from synthetic images. *Conf. 5005, A stereoscopic displays and applications XIV. Proc. SPIE*, **5006**:pp.348-358.

GESTS Int'l Trans. Computer Science and Engr., Vol.61, No.1 113

Generation of 3D Video Sequences Employing Depth Map Information

Eduardo Ramos and Volodymyr Ponomaryov

National Polytechnic Institute of Mexico, E.S.I.M.E Culhuacan, Santa Ana #1000, Col. San Francisco Culhuacan, 04430, Mexico
eramos@ieee.org, vponomar@ipn.mx

Abstract. Color video algorithm, which allows the conversion of a 2D video sequence into a 3D one, is presented. In literature, there exist several techniques to compute depth map information; however, not all techniques are used in applications with real videos. Similarity measure algorithms are employed in order to construct depth map information. Depth information is computed from each a pair of subsequent frames in the video sequence, and then it is applied in the construction of 3D video sequence by means of anaglyph employment. Additionally, the obtained anaglyphs are improved employing the depth map manipulation via P -th compression law. Finally, the video construction is realized. Numerous simulation results with several video sequences have confirmed good visual performance of current proposal. Novel framework is compared with Photoshop classical algorithm demonstrating their efficiency.

1 Introduction

Depth map information is important in several fields such as video filtering, robot navigation, video editing, scenery reconstruction, super resolution, etc. Stereo matching problem has been studied widely [1, 2], however, the computation of dense depth map information from video sequences is still a difficult task. Usually, the depth maps can be estimated employing the pixel correspondence of similar features in the stereo perspective, either via a stereo pair that is formed by neighboring frames of the video sequence. However, finding correct corresponding points produces a number of potential problems, such as occlusion, ambiguity, illuminative variations, etc. A lot of algorithms have been proposed to address some of the aforementioned issues in stereo vision; however, it is still relatively an open problem [1, 2].

Calculation of the dense depth map consists of finding the correspondence between the pixel locations in one frame with its positions in the other one. The mapping of an image to other one can be realized by registering a spatial neighborhood, surrounding each a pixel for the image that corresponds to the other one. There exist numerous techniques to solve the problem of the depth map computation: optical flow [3, 4, 5] that is based on the conversion of motion vectors into depth maps, applying the ve-

114 Generation of 3D Video Sequences Employing Depth Map Information

localities from spatial-temporal derivatives of image intensity [6]; matching algorithm that employs the cross correlation on pixel distance as a similarity measure [7]; the method, where Gabor filters in several spatial scales are tuned to different spatial orientations [8, 9].

Two recent methods in the computing of the correspondence measures between pixels in an image stereo pairs are proposed, where the first one [10] uses the stereo correspondence that is computed based on error energy of disparity maps, and the second method employs the wavelet and multi wavelet analysis [11] in order to compute stereo correspondence estimation.

Polarized lens, shutter glasses, active vision and anaglyphs are some of the systems to visualize 3D images or 3D video sequences [12]. Unfortunately, some of these devices are still expensive and its limitations in vision are great, also, some require extra hardware to implement them.

Anaglyph based on 3D projection presents a possible solution in developing the stereoscopic projection devices. It is considerate as the cheapest stereoscopic projection being implemented on standard monitors without significant modifications, using only simple lens for 3D viewing. Simple algorithm to reconstruct anaglyphs is presented in [13], where the procedure uses the color alignment based on a transformation matrix. Another algorithm applied in video sequences is presented in [14], where the anaglyph computation is realized employing the depth maps via MPEG-4 coding hardware.

The main idea of present proposal consists of the processing of an input video sequence where the depth map and anaglyph are automatically computed using only the 2D video sequence with moving scene. In order to avoid ghosting in a final 3D video sequence, an anaglyph improvement using a dynamic range compression is implemented.

2 Methodology

In the proposed framework, the first stage consists of the depth map computation. Here, the similarity measures based on: the sum of squared differences (SSD), stereo correspondence estimation using wavelets, optical flow estimation based on Lucas and Kanade algorithm[5], region based stereo matching technique [10] are used. The second stage consists of the depth map manipulation using the dynamic range compression, and the final stage presents the color anaglyph generation using the nearest neighbor interpolation.

The proposed algorithm is presented in the Figure 1.

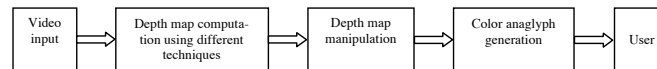


Fig. 1. Proposed algorithm to compute 3D anaglyph video sequence

2.1 Depth Map Computation

Because the depth map is the most important stage in the algorithm, some techniques are tested in order to select the most reliable one: Lucas & Kanade differential, region based stereo matching, stereo correspondence techniques based on sum of squared differences, and finally, the algorithm based on Wavelets.

2.2 Optical Flow Based on Lucas & Kanade Differential Technique

A common solution for computing the optical flow is to assume that the total spatial and temporal derivatives of the image brightness remain constant [4], then, the velocity vector can be found as: $v = (u, w) = (dx/dt, dy/dt)$.

Developing this equation in first order approximation, it is easy to write:

$$\frac{\partial E}{\partial x}u + \frac{\partial E}{\partial y}w + \frac{\partial E}{\partial t} = 0, \quad (1)$$

where the spatial and temporal derivatives should be estimated using two images or two frames of a video sequence.

According to [5], the weighted least squares are fitted employing the local first order constraints to a constant model, that can be implemented for velocity vector v in each a small spatial neighborhood β minimizing the following criterion:

$$\sum_{x \in \beta} W^2(x) [\nabla I(x, t) \cdot v + I_t(x, t)]^2, \quad (2)$$

where $W(x)$ is a window function that constrains the results, $I_t(x, t)$ denotes the partial time derivative, and $\nabla I(x, t) = (I_x(x, t), I_y(x, t))^T$.

The solution of the equation (2) can be written as:

$$v = [A^T W^2 A]^{-1} A^T W^2 b, \quad (3)$$

where

$$A = [\nabla I(x_1), \dots, \nabla I(x_n)]^T, \quad W = \text{diag}[W(x_1), \dots, W(x_n)],$$

$$b = -(I_t(x_1), \dots, I_t(x_n))^T.$$

Once the optical flow was found, the optical flow conversion to depth map can be computed in the following form [15]:

$$\text{DepthMap} = c \sqrt{MV_x^2(i, j) + MV_y^2(i, j)}, \quad (4)$$

where c is a scale parameter to fit the maximal disparity over the frames.

116 Generation of 3D Video Sequences Employing Depth Map Information

2.3 Region Based Stereo Matching Calculation

Here, we implemented the method based on *Global Error Energy Minimization by Smoothing Functions* proposed in the paper [10].

To calculate error energy matrix for every disparity value, the method of block matching technique has been implemented. Let denote left image in RGB format as $L(i, j, c)$ and the right image as $R(i, j, c)$, the error energy of disparity maps can be presented in such a form:

$$E_d(i, j) = \frac{1}{3 \cdot n \cdot m} \sum_{x=i}^{i+n} \sum_{y=j}^{j+m} \sum_{k=1}^3 \left(L(x, y + d(i, j), k) - R(x, y, k) \right)^2 \quad (5)$$

where $d(i, j)$ is the disparity and index k represents each a color component in an image.

For a predetermined range of disparity search, the averaging filter is applied several times to smooth the error energy matrix of disparity.

Disparity map contains the unreliable estimations for some points, so, to avoid this, a threshold has been applied in such a form:

$$\begin{aligned} \tilde{d}(i, j) &= \begin{cases} d(i, j) & E_d(i, j) \leq Ve \\ 0 & E_d(i, j) > Ve \end{cases} \\ \tilde{E}_d(i, j) &= \begin{cases} E_d(i, j) & \text{if, } E_d(i, j) \leq Ve \\ ne & \text{if, } E_d(i, j) > Ve \end{cases} \end{aligned} \quad (6)$$

here, $Ve = \alpha \cdot \text{Mean}(E_d)$, and α is the tolerance coefficient to adjust the filtering process.

2.4 Stereo Correspondence Based on Sum of Squared Differences

The sum of absolute differences [16] is presented as:

$$D(x, y) = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \left(f_t(i, j) - f_w(i + x, j + y) \right)^2, \quad (7)$$

where $x = 0, \dots, M - m - 1$ and $y = 0, \dots, N - n - 1$ when template is of the size $m \times n$ pixels, image is of a size $M \times N$. $f_t(i, j)$, $f_w(i + x, j + y)$ are the template and the window in the sensed image; the coordinates x and y show the position of the template in the sensed image. By changing x and y , the template is shifted through the image, and at each shift position the sum of squared differences between corresponding pixels in the template and the window can be found. The smaller D value exposes the more similarity between the template and window.

2.5 Stereo Correspondence Estimation Based on Wavelets Technique

The discrete Wavelet transform (DWT) applied in this method is usually implemented in a form of filter banks [17, 18] in three different detail space components: horizontal, vertical and diagonal and approximation. So, the wavelet transform value can be expressed as:

$$WTM_s = W_s \angle \Theta_{W_s}, \quad (8)$$

where $W_s = \sqrt{|D_{h,s}|^2 + |D_{v,s}|^2}$ is the Wavelet modulus, $D_{h,s}$, $D_{v,s}$ are the k -th horizontal and vertical detail components, and the phase $\Theta_{w,s}$ is defined as:

$$\Theta_{w,s} = \begin{cases} \alpha(s) & \text{if } D_{h,s} > 0 \\ \pi - \alpha(s) & \text{if } D_{h,s} < 0 \end{cases}, \alpha(s) = \tan^{-1} \left(\frac{D_{v,s}}{D_{h,s}} \right). \quad (9)$$

Illuminative variation in images is compensated as [18]:

$$NW_s = \frac{D_{dc,s}}{|A_s|}, \quad (10)$$

where D_{dc} are the vertical, horizontal and diagonal detail components, s represents the scale of decomposition, and A_s is the approximation scale.

Once the DWT is computed for each pair of frames, the depth map can be formed using the normalized cross-correlation:

$$NCC = \sum_{(i,j) \in W} \frac{NW_{l,s}(i,j) \cdot NW_{r,s}(x+i, y+j)}{\sqrt{\sum_{i,j \in W} NW_{l,s}^2(i,j) \cdot \sum_{i,j \in W} NW_{r,s}^2(x+i, y+j)}}. \quad (11)$$

2.6 Depth Map Manipulation

Ghosting effects can be found in the conventional methods during the anaglyphs construction. One of the promising approaches in minimizing these artifacts in the anaglyphs is the dynamic range compression of depth map [19], where every pixel of depth map is manipulated according to the following expression:

$$D_{new} = a \cdot D^P, \quad (12)$$

where D_{new} is the new depth map pixel value, a is a normalized constant and P is a compression factor.

2.7 Color Anaglyph Reconstruction

Anaglyph synthesis is a simple process, in which the red channel in one image should be replaced by the red channel of second image using a stereo pair [13], or maybe neighboring frames in the video sequence. Anaglyph produces a visual effect

118 Generation of 3D Video Sequences Employing Depth Map Information

of three-dimensional image when it is viewed by color filter lens that are usually constructed with red and blue filters.

In order to form an anaglyph, different methods that rely on image re-sampling are used. The nearest neighbor interpolation, that is the simplest and effective in applications, does not significantly reduce the quality of the resulting anaglyph and does not require extra computation [19]. So, this method has been implemented for depth map information and red channel of the left image, and the interpolation result is finally added to green and blue components of the right image. This step is realized for each a pair of the neighboring frames of the video sequence.

3 Simulation Results

In the simulation experiments, different synthetic images in *Png* format are used, such as: *Venus*, *Aloe* and *Bowling* (1280 x 1110 pixels) in order to obtain quantitative measurements. Additionally, the real-life color video sequences in the *Avi* format are employed: *Coastguard*, *Flowers* and *Foreman*. *Coastguard* color video sequence (300 frames, 352 x 288 pixels) presents a particular boat movement from left to right side in the image; the *Flowers* sequence (300 frames, 352 x 288 pixels) exposes right to left camera movement; and the *Foreman* sequence (300 frames, 352 x 288 pixels) shows a talking man.

In order to select the best depth map method in the reconstruction of the 3D video sequences, the quantity of bad disparities (*QBD*) for different synthetic images is calculated as:

$$B = \frac{1}{N} \sum_{x,y} |d_E(x,y) - d_G(x,y)|^2, \quad (13)$$

where N is the total number of pixels in the input image, d_E and d_G are the estimated and ground truth disparities, respectively.

The simulation results are presented in two parts: firstly, the depth maps results and quantity of bad pixels, and secondly, the resulting anaglyph, using the proposed method and classical Photoshop algorithm.

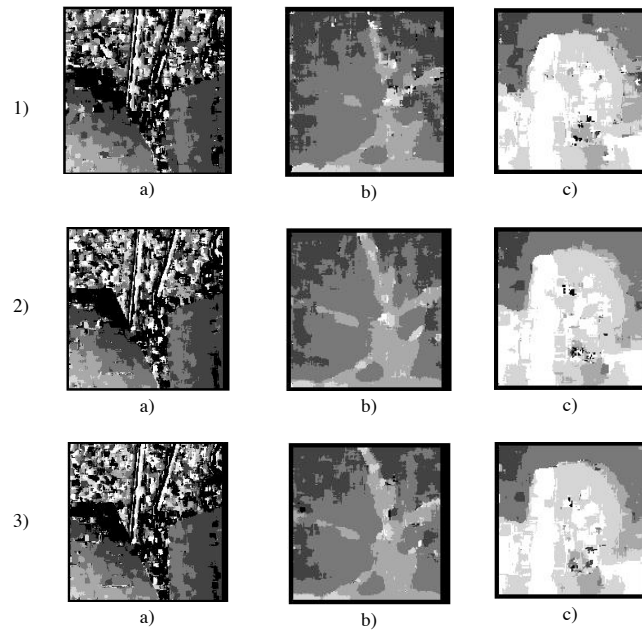
Table 1. Quantity of Bad Pixels using different wavelets in synthetic images.

Wavelet	Venus	Aloe	Bowling 1
<i>Haar</i>	0.1825	0.0399	0.0888
<i>Daubechies (Db8)</i>	0.1586	0.0319	0.0645
<i>Daubechies (Db7)</i>	0.1616	0.0314	0.0679
<i>Daubechies (Db6)</i>	0.1612	0.0320	0.0725
<i>Symlets (Sym8)</i>	0.1706	0.0299	0.0592
<i>Symlets (Sym7)</i>	0.1699	0.0289	0.0623
<i>Symlets (Sym6)</i>	0.1719	0.0295	0.0635
<i>Coiflets (Coif5)</i>	0.1650	0.0365	0.0672
<i>Coiflets (Coif4)</i>	0.1658	0.0377	0.0651

<i>Coiflets(Coif3)</i>	0.1659	0.0343	0.0590
<i>Biorthogonal(Bio6.8)</i>	0.1694	0.0297	0.0555
<i>Biorthogonal(Bio5.5)</i>	0.1714		0.0633
<i>Biorthogonal(Bio4.4)</i>	0.1710	0.0291	0.0644
Other methods			
<i>L&K</i>	0.3048		0.1720
<i>RBSM</i>	0.2013	0.4210	0.2019
<i>SSD</i>	0.1482	0.1112	0.1885

As one can see in Table 1, the wavelet Biorthogonal 6.8 presents better QBP in general comparison with other Wavelet families. In order to compare the wavelet based technique with other methods, we have implemented other methods: Lucas and Kanade, Region Based Stereo Matching, and Sum of Squared Differences.

The figure 2 shows the depth map obtained using the wavelets: Daubechies 8, Biorthogonal 6.8, Symlets 8, and Lucas & Kanade technique.



120 Generation of 3D Video Sequences Employing Depth Map Information

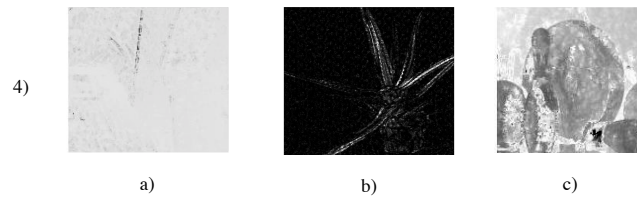


Fig. 2. Obtained depth maps using: 1) Daubechies 8, 2) Symlets 8, 3) Biorthogonal 6.8 and 4) Lucas & Kanade, in a) Venus, b) Aloe and c) Bowling images, respectively

In present implementation, based in the obtained results, we have selected Biorthogonal 6.8 as a wavelet to use in anaglyph construction. The figure 3 and figure 4 exposes the anaglyph results for different synthetic images and real videos. The figure 5 shows the obtained results using the dynamic range compression. Comparing figure 5 and figure 4 one can observe the reduction of the ghosting effects in figure 5.



a)

GESTS Int'l Trans. Computer Science and Engr., Vol.61, No.1 121

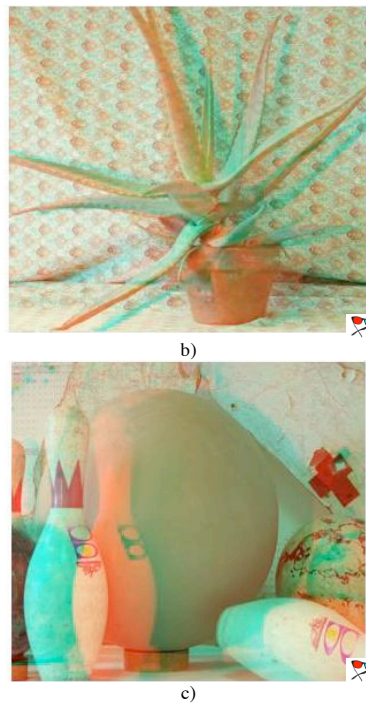


Fig. 3. Anaglyph synthesized using Biorthogonal 6.8 wavelet for: a) *Venus*, b) *Aloe* and c) *Bowling 1* images



Fig. 4. Anaglyph synthesized using Photoshop algorithm

©GESTS-Sep.2010

122 Generation of 3D Video Sequences Employing Depth Map Information

4 Conclusion

Novel framework for 3D anaglyph video sequence reconstruction based on depth maps information has been developed. The realized experiments permit to demonstrate that the color images and real color video sequences can be visualized via reconstruction of the depth maps without extra hardware equipment. Simulation experiments with different video and image formats have confirmed good 3D perception for proposed algorithm. Method based on Daubechies 8 Wavelet calculation and SSD technique expose better depth map results but in the second case, the algorithm need sufficiently more computation time.



Fig. 5. Anaglyph visualization P-th law ($P=a=0.5$) in a) *Venus*, b) *Foreman* sequence where Biorthogonal 6.8 Wavelet was used to compute the depth map

Acknowledgments. The authors would thank National Polytechnic Institute of Mexico, CONACYT (project 81599) for their support to realize this work.

References

- [1] S. B. Kang and R. Szeliski, "Extracting view dependent depth maps from a collection of images", *Intl J. Computer Vision*, Vol. 58, No. 2, pp. 139-163, 2004.
- [2] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two frames stereo correspondence algorithms", *Intl J. Computer vision*, Vol. 47, No. 1-3, pp. 7-42, 2002.
- [3] J. L. Barron, D. J. Fleet, S. S. Beauchemin and T. Burkitt, "Performance of optical flow techniques", *IEEE Conf. CVPR*, pp. 236-242, 1992.
- [4] B. K. P. Horn and B. G. Schunk, "Determining optical flow", *AI* 17, pp.185-204, 1981.
- [5] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision", *Proc. DARPA IU Workshop*, pp.121-130, 1981.
- [6] E. Ramos and V. Ponomaryov, "3D Colour Video Conversion using Depth Map Estimation", *Telecommunication and Radio Engineering*, Vol.69, No.5, pp. 459-471, Begell House Inc., 2010.

- [7] P. Anandan, "A computational framework and an algorithm for the measurement of visual motion", *Int. J. Comp. Vision*, pp. 281-310, 1989.
- [8] D. J. Heeger, "Model for the extraction of image flow", *J. Opt. Soc. Am. A4*, pp. 1455-1471, 1987.
- [9] D. J. Heeger, "Optical flow using spatiotemporal filters", *Int. J. Comp. Vision*, pp. 279-302, 1988.
- [10] B. B. Alagoz, "Obtaining depth maps from color images by region based stereo matching algorithms", *OncuBilim Algorithms and System Labs*, Vol. 8, No. 4, 2008.
- [11] A. Bhatti, S. Nahavandi, "Stereo correspondence estimation based on wavelets and multiwavelets analysis" *Stereo Vision*, Austria, 372, 2008
- [12] L. Lipton, "Stereo vision formats for video and computer graphics", *Stereo Graphics Corporation*, 1997.
- [13] E. Dubois, "A projection method to generate anaglyph stereo images", *Proceedings of the Acoustics, Speech, and Signal Processing*, IEEE, Vol. 3, pp. 1661-1664, 2001.
- [14] I. Idesses, L. Yaroslavsky, B. Fishbain, "3D from compressed video", *Stereoscopic displays and virtual reality systems*, *Proc. Of SPIE-IS&T Electronic Imaging*, vol. 6490, 2007.
- [15] I. Idesses, L. Yaroslavsky, "New methods to produce high quality colour anaglyphs for 3D visualization", *ICIAR, LNCS 3212*, pp. 273-280, 2004.
- [16] A. Goshtasby, "2D and 3D image registration", *Wiley Publishers*, USA, 2005.
- [17] V. Kravchenko, H. Meana, V. Ponomaryov, *Adaptive digital processing of multidimensional signals with applications*, FizMatLit Edit., Moscow, 2009.
- [18] A. Bhatti, S. Nahavandi, H. Zheng, "Image matching using TI multi-wavelet transform" *Proc. Digital Image Computing*, Sydney, 2003.
- [19] I. Idesses, B. Fishbain, L. Yaroslavsky, "Real time depth map manipulation for 3D visualization", *Proc. Real time image and video processing*, SPIE, vol. 7244, USA, 2009.

Biography



▲ Name: Eduardo Ramos-Diaz
 Address: National Polytechnic Institute of Mexico,
 E.S.I.M.E.
 Culhuacan, Santa Ana #1000, Col. San Francisco
 Culhuacan, 04430, Mexico
 Education & Work experience: M. SC. (2004) and actually
 he is a Ph.D. student at National Polytechnic Institute of Mex-
 ico
 (Mexico-city).
 Tel: (52)5557296000. Ext.73250
 E-mail: eramos@ieee.org

124 Generation of 3D Video Sequences Employing Depth Map Information

▲ Name: Volodymyr Ponomayov

Address: National Polytechnic Institute of Mexico,
E.S.I.M.E

Culhuacan, Santa Ana #1000, Col. San Francisco
Culhuacan, 04430, Mexico

Education & Work experience: Ph.D. (1974), Doctor of Sciences (1981), Full Professor (1984). He is working as Professor at National Polytechnic Institute of Mexico (Mexico-city). He is author more than 150 scientific papers, 300 conference papers, and also 23 patents of ex-USSR, Russia and Mexico, and several scientific books.

Tel: (52)5557296000. Ext.73250

E-mail: vponomar@ipn.mx

Other information: