



FACULTAD DE INGENIERÍA,
ARQUITECTURA Y URBANISMO

ESCUELA ACADEMICO PROFESIONAL DE INGENIERIA
DE SISTEMAS

TESIS

**RECONSTRUCCIÓN 3D A PARTIR DE IMÁGENES RAYOS X DE FRUTOS
CON SEMILLA.**

PARA OPTAR EL TITULO PROFESIONAL DE INGENIERO DE SISTEMAS.

Autor:

Bach. ARMANDO LÓPEZ YOVERA

Pimentel-Perú

2017.



**RECONSTRUCCIÓN 3D A PARTIR DE IMÁGENES DE RAYOS X DE
FRUTOS CON SEMILLA.**

Aprobación de la tesis.

López Yovera Armando

Autor

Ing. Mejia Cabrera Heber Ivan

Vocal/Asesor de Jurado

MG. Tuesta Monteza Victor Alexci

Presidente de Jurado

DR. Gutiérrez Gutiérrez, Jorge Luis

Secretario(a) de Jurado

Ing. Mejia Cabrera Heber Ivan

Asesor Especialista



DEDICATORIA

A mis padres José López Mendoza y Rosa Yovera Sandoval, quienes son parte de lo que quiero lograr.

A mis hermanos(as) Laura, Cesar, German, Orlando, Marisela., quienes comparto momentos gratos.

A mis compañeros de clase de la USS que me acompañaron durante mi formación personal y profesional.



AGRADECIMIENTO

En primer lugar a Dios por ser el centro de mi existencia, porque guía mis pasos y me regala una vida llena de paz, amor y humildad.

A mis padres José y Rosa por apoyarme de manera incondicional en todo momento, todos mis logros son suyos.

Quiero agradecer al Ing. Heber Ivan Mejia por su invaluable orientación, confianza y gran apoyo en el desarrollo de esta investigación.

Al Mg. Víctor Alexci Tuesta Monteza por su invaluable orientación, confianza y gran apoyo en el desarrollo de esta investigación.

Al Dr. Manuel Forero por su tiempo, consejos y asesoría para ser posible esta investigación.

A todos mis docentes de quienes aprendí muchos más de lo que enseñaban en sus cursos, gracias por transmitir su conocimiento y exigir siempre lo mejor de sus alumnos.



INDICE

DEDICATORIA	1
AGRADECIMIENTO	2
INDICE	3
Listado de Ilustraciones	5
Listado de Tablas	7
Listado de Gráficos	8
RESUMEN	9
ABSTRACT	12
INTRODUCCION	15
I. PROBLEMA DE INVESTIGACIÓN	16
1.1. Situación problemática	16
1.2. Formulación del problema	21
1.3. Delimitación de la investigación	21
1.4. Justificación e importancia de la investigación:	21
1.5. Limitaciones de la investigación	23
1.6. Objetivos de la investigación	23
II. Marco teórico	24
2.1. Antecedentes de la investigación	24
2.2. Estado del arte	28
2.3. Base teórica científicas	29
2.4. Definición de términos básicos	60
III. MARCO METODOLÓGICO	62
3.1. Tipo y diseño de la investigación	62
3.2. Población y muestra:	63
3.3. Hipótesis	64
3.4. Operacionalización	64
3.5. Métodos, técnicas e instrumentos de recolección de datos	65
3.6. Procedimiento para la recolección de datos	66
3.7. Plan de análisis estadístico de datos	66



3.8.	Criterios éticos.....	66
3.9.	Criterios de rigor científico.....	67
IV.	ANÁLISIS E INTERPRETACIÓN DE RESULTADOS	67
4.1.	Resultados	67
4.1.1.	Pruebas Realizadas	67
4.1.2.	Resultados de las pruebas	69
4.2.	Discusión de Resultados	92
V.	PROPUESTA DE INVESTIGACIÓN	95
5.1.	Propuesta De Investigación.....	95
VI.	CONCLUSIONES Y RECOMENDACIONES	145
6.1.	Conclusiones.....	145
6.2.	Recomendaciones.....	147
	REFERENCIAS	148
	Anexos	149



Listado de Ilustraciones

Ilustración 1 Consumo de memoria aplicando la técnica tradicional, con diferentes resoluciones de imágenes. 20

Ilustración 2 Consumo de memoria aplicando la técnica de pistas, con diferentes resoluciones de imágenes. 20

Ilustración 3 Par de imágenes estéreo. 30

Ilustración 4 Mapa de disparidad. 31

Ilustración 5 Ilustración de la geometría estéreo, lo que representa dos cámaras idénticas y paralelas con una distancia focal f y la distancia entre ellos b 32

Ilustración 6 Ilustración de restricción nucleal. En este ejemplo, el punto P que genera PL en L debe pertenecer a la línea recta definida por CL y PL , y la proyección P en R debe pertenecer a la recta eR definida como la línea epipolar. 34

Ilustración 7 Instalación estéreo con dos cámaras idénticas y paralelas. 35

Ilustración 8 disparidad Space Image. 36

Ilustración 9 (a) par de entrada estéreo; (b) (c) (d) una representación en escala de grises relacionada con las disparidades 21,26 y 10. 38

Ilustración 10 - (a) Representación DSI indica un corte en $y = 151$; (B) rebanada (x, d) para DSI $y = 151$. Fuente de las imágenes individuales. 39

Ilustración 11 - (a) Cortar en una DSI; (b) la indicación de un camino que minimiza la energía. 40

Ilustración 12 - (a) mapa de disparidad ideal; (b) mapa de disparidad método local; (C) mapa de disparidades método global de los recortes (gráfico); (D) relación entre (a) y (b) donde el blanco indica la igualdad de píxeles y los píxeles negros que indican diferentes; (e) relación de (a) y (c); Observación :. Para las imágenes que tienen bordes blancos, que incluye un contorno azul (fuera de la imagen) para facilitar la visualización. 41

Ilustración 13 - punto en cuestión y su ventana de 3×3 a la izquierda de la imagen, y la línea epipolar a la derecha de la imagen, se busca la correspondencia. 42

Ilustración 14 - Descripción de las variables a la ventana de la izquierda y la derecha. 44

Ilustración 15 - Los cálculos repetidos con el desplazamiento de la ventana de la imagen izquierda. 47

Ilustración 16 Planos que almacenan los costos de correspondencias. 49

Ilustración 17 Posiciones involucrados durante el proceso de agregación al punto (60,70). 49

Ilustración 18 Simulación de valores en los diferentes planos para la posición (40,50) 51

Ilustración 19 - Comparación de los requisitos entre las aplicaciones de visión estéreo, con excepción de la reconstrucción 3D de bajo coste. 53

Ilustración 20 Estructura de la librería OpenCV 58

Ilustración 21 La calibración de cámaras para obtener imágenes estéreo. 97

Ilustración 22 Utilización del equipo SGHealthCare. 97

Ilustración 23 Ilustración de la distancia del equipo en relación a la placa. 98

Ilustración 24 Visualización de la Pantalla de rayos x. 99

Ilustración 25 Adquisición de 2 imágenes rayos x estéreo del fruto de la palta. 99

Ilustración 26 Carga de los datos que están en el archivo Last.txt 102

Ilustración 27 Carga de 2 imágenes rayos x (648x864 píxeles) del fruto de la palta para la búsqueda por intensidad, mediante el Método Tradicional. 103

Ilustración 28 (a) Imagen con banda azul claro en el centro de una región azul oscuro; (B) representación gráfica de las intensidades de color para la línea punteada en (a); (C) Primera derivada. 104

Ilustración 29 Máscaras conocidas como operadores Sobel. 105



Ilustración 30 (a) Par de imágenes estéreo; (B) Carga de 2 imágenes rayos x del fruto de la palta procesado con el operador gradiente, con valores absolutos y limitado a 255, mediante el Método Tradicional..... 108

Ilustración 31 Disparity Space Image (DSI)..... 109

Ilustración 32 Punto en cuestión y su ventana 3x3 en la imagen izquierda, y la línea epipolar en la imagen derecha, donde se busca la correspondencia. 111

Ilustración 33 Descripción de las variables para la ventana izquierda y derecha..... 112

Ilustración 34 Los cálculos repetidos con el desplazamiento de la ventana de la imagen 115

Ilustración 35 Planos que almacenan los costos de correspondencia..... 116

Ilustración 36 Posiciones involucradas durante el proceso de agregación para el punto (60,70). 118

Ilustración 37 Agregación de evidencias usando plano temporal, mediante el método 1: Técnica Tradicional..... 120

Ilustración 38 Simulación de valores en los distintos planos para la posición (40, 50). 120

Ilustración 39 El resultado del mapa de disparidad aplicando el método 1: Técnica tradicional por búsqueda de intensidad. 125

Ilustración 40 malla inicial 3D 126

Ilustración 41 - (a) origen de coordenadas en el mapa de bits en la esquina superior izquierda; (B) origen de coordenadas en el centro..... 126

Ilustración 42 - (a) un seguimiento con tonos de gris de partida desde el blanco al negro; (b) los resultados generados con la técnica de reconstrucción 3D se describe aquí..... 127

Ilustración 43 (a) Imagen izquierda estéreo;(b) mapa de disparidades ideal utilizado; (c) presentará da geometría 3D reconstruida a partir do mapa de disparidades en modo wireframe; (d) Reconstrucción 3D en modo sólido. 128

Ilustración 44 Resultado del algoritmo de reconstrucción mediante el método 1: Técnica tradicional 131

Ilustración 45 Generación del archivo .wrl 131

Ilustración 46 Diagrama: Reconstrucción 3d mediante el Método 1: Técnica Tradicional. 132

Ilustración 47 Diagrama de la Técnica de Pistas 134

Ilustración 48 Costo de Correspondencia para el primer llenado de agregación de pistas para una ventana 3X3..... 135

Ilustración 49 Costo por correspondencia de otras líneas hasta el final de las imágenes de entrada..... 136

Ilustración 50 En la parte superior de la primera agregación, debajo de la siguiente utilizar los resultados obtenidos anteriormente..... 138

Ilustración 51 Proceso de agregación para las otras agregación de pistas..... 138

Ilustración 52 El resultado del mapa de disparidad del fruto de la palta aplicando el método 2: Técnica de pistas por búsqueda de intensidad..... 142

Ilustración 53 Resultado del algoritmo de reconstrucción mediante el método 2: Técnica de pistas 143

Ilustración 54 Diagrama: Reconstrucción 3d mediante el Método 2: Técnica de Pistas 144

Ilustración 55 Argumentos de la Interfaz Gráfica para el procesamiento del mapa de disparidad y la reconstrucción 3d del fruto de la palta con semilla, utilizando el Método Tradicional. ... 150

Ilustración 56 Reconstrucción 3d con el envase VOLT (Solido). 152

Ilustración 57 Reconstrucción 3d con el envase VOLT (malla)..... 153

Ilustración 58 Reconstrucción 3d del fruto de la palta (solido). 154

Ilustración 59 Reconstrucción 3d del fruto de la palta (malla). 155



Ilustración 60 Reconstrucción 3d del fruto de la palta (solido) con un nivel de disparidad de 160..... 156

Ilustración 61 Reconstrucción 3d del fruto de la palta (malla) con un nivel de disparidad de 160..... 157

Ilustración 62 Instalación de Directx SDK..... 157

Ilustración 63 Finalización de la Instalación de Directx SDK 158

Ilustración 64 Código Fuente de la aplicación..... 159

Listado de Tablas

Tabla 1 Tiempo que tarda cada método en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles) 70

Tabla 2 Resumen de la tabla del Tiempo que tarda cada método en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles) 71

Tabla 3 Tiempo Promedio que tarda cada método en procesar el mapa de disparidad, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles). 72

Tabla 4 Tiempo que tarda cada método en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles) 75

Tabla 5 Resumen de la tabla del Tiempo que tarda cada método en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles) 76

Tabla 6 Tiempo Promedio que tarda cada método en procesar el mapa de disparidad, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles). 77

Tabla 7 Tiempo que tarda cada método en realizar la reconstrucción 3d de imágenes rayos x del fruto con semilla (palta), mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles). 79

Tabla 8 Resumen de la tabla del tiempo que tarda cada método en realizar la reconstrucción 3d de imágenes rayos x del fruto con semilla (palta), mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles). 80

Tabla 9 Tiempo Promedio que tarda cada método en reconstruir el fruto de la palta con semilla, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles). ... 81

Tabla 10 Tiempo que tarda cada método en realizar la reconstrucción 3d de imágenes rayos x del fruto con semilla (palta), mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles). 83

Tabla 11 Resumen de la tabla del tiempo que tarda cada método en realizar la reconstrucción 3d de imágenes rayos x del fruto con semilla (palta), mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles). 84

Tabla 12 Tiempo Promedio que tarda cada método en reconstruir el fruto de la palta con semilla, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles). 85

Tabla 13 Consumo de memoria de cada método en el proceso de búsqueda de correspondencias para obtener el mapa de disparidad expresado en MB, ingresando las 2 imágenes con tamaño de 648x864 pixeles 86

Tabla 14 Tabla de las variables de entrada(x, y, d, j), que se utilizaran para el cálculo del consumo de memoria, ingresando las 2 imágenes con tamaño de 648x864 pixeles..... 87

Tabla 15 % de consumo de memoria de cada método, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles) 87

Tabla 16 Tabla de las variables de entrada(x, y, d, j), que se utilizaran para el cálculo del consumo de memoria, ingresando las 2 imágenes con tamaño de 1792x2392 pixeles..... 89



Tabla 17 % de consumo de memoria de cada método, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles) 90

Tabla 18 Precisión en centímetros cúbicos comparando el volumen del objeto real de la palta con el objeto reconstruido de la palta mediante la técnica tradicional y pistas con resolución de imagen (648x864 pixeles). 90

Tabla 19 % de Precisión de ambos métodos 91

Listado de Gráficos

Gráfico 1 Tiempo que tarda cada método en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles)..... 72

Gráfico 2 Tiempo promedio de procesamiento de cada método, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles) 72

Gráfico 3 Tiempo que tarda cada método en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles)..... 77

Gráfico 4 Tiempo promedio de procesamiento de cada método, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles) 78

Gráfico 5 Tiempo que tarda cada método en realizar la reconstrucción 3d de imágenes rayos x del fruto con semilla (palta), mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles). 80

Gráfico 6 Tiempo Promedio que tarda cada método en reconstruir el fruto de la palta con semilla, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles). ... 82

Gráfico 7 Tiempo que tarda cada método en realizar la reconstrucción 3d de imágenes rayos x del fruto con semilla (palta), mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles). 85

Gráfico 8 Tiempo Promedio que tarda cada método en reconstruir el fruto de la palta con semilla, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles).86

Gráfico 9 % de consumo de memoria de cada método, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles) 88

Gráfico 10 Consumo de memoria de cada método en el proceso de búsqueda de correspondencias para obtener el mapa de disparidad expresado en MB, ingresando las 2 imágenes con tamaño de 1792x2392 pixeles 89

Gráfico 11 % de consumo de memoria de cada método, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles) 90



RESUMEN

Este trabajo plantea como objetivo la aplicación de 2 técnicas de Visión Estéreo para la reconstrucción 3d de 2 imágenes rayos x del fruto de la palta con semilla; esta investigación surge debido a que el tiempo de congelación resulta ser mucho más complejo según la variación de forma del producto, y a pesar que existen modelos matemáticos con el que podemos apoyarnos para determinar resultados de congelamiento estos aún siguen siendo aproximaciones, ya que la forma geométrica como esfera, elipses, cilindros no son formas precisas que se puedan comparar con otros tipos de frutos como la palta con semilla, ya que esta fruta es solo una aproximación a figuras geométricas, como sabemos tiene variación de forma lo que hace que el cálculo de enfriamiento no sea el adecuado; el propósito es entonces reconstruir la forma real del fruto de la palta en tres dimensiones permitiendo de esta manera que los cálculos de enfriamiento del fruto sea el adecuado en las industrias que practican este tipo de procesos.

Para realizar esta investigación, se ha estudiado las bases teóricas y los trabajos previos relacionados a la reconstrucción 3d mediante visión estereo usando el método 1: Técnica tradicional y método 2: Técnica de pistas.

Se utilizaron 2 imágenes rayos x del fruto de la palta como muestra para la puesta a prueba de nuestros métodos implementados, los indicadores utilizados para la evaluación de cada método fueron el tiempo de procesamiento del mapa de disparidad, tiempo de reconstrucción del fruto de la palta con semilla, porcentaje de consumo de memoria en procesar el mapa de disparidad, porcentaje de precisión y error en la reconstrucción 3d del fruto de la palta con semilla.



En esta investigación se desarrollan dos métodos para la reconstrucción 3d del fruto de la palta con semilla , y se determinó que el método que procesa de manera más rápida el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles) es el Método 1 que permite preparar el mapa de disparidad para poder reconstruirlo basándose en la técnica tradicional, obteniendo un tiempo de 18.2825 segundos, en comparación con el método 2 que permite preparar el mapa de disparidad para poder reconstruirlo basándose en la técnica de pistas ,que tarda un tiempo de 19.451 segundos en procesar el mapa de disparidad, tardando 1.1685 segundos más que el método 1. El método que reconstruye de manera más rápida el fruto de la palta con semilla mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles) es el Método 1 que permite ver como el fruto está en su tercera dimensión basándose en la técnica tradicional, obteniendo un tiempo de 18.3825 segundos, en comparación con el método 2 que permite ver como el fruto está en su tercera dimensión basándose en la técnica de pistas, que tarda un tiempo de 19.551 segundos en reconstruir el fruto de la palta con semilla, tardando 1.1685 segundos más que el método 1. El método que consume menos memoria en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles) es el Método 2 que permite preparar el mapa de disparidad para poder reconstruirlo basándose en la técnica de pistas, obteniendo un porcentaje de consumo de memoria de 11.37 %, en comparación con el método 1 que permite preparar el mapa de disparidad para poder reconstruirlo basándose en la técnica tradicional ,que consume 88.63 %, consumiendo 72.26% más de memoria que el método 2.



Comparando ambos métodos mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles), según la fórmula del rango estándar, se puede notar que ambos métodos que permiten la reconstrucción 3d del fruto de la palta con semilla tienen una precisión de $\pm 5.88 \text{cm}^3$ respecto al objeto real del fruto de la palta. Esto demuestra que según la percepción del usuario en ver el objeto reconstruido con un mejor panorama, lo sustenta con este indicador. La diferencia de estos métodos es el tiempo de procesamiento, reconstrucción así como la memoria consumida.

Palabras clave: Reconstrucción 3D, Visión Estéreo, Mapa de disparidad, Costo de correspondencia, Agregación de pruebas, selección de disparidad.



ABSTRACT

This work aims at the application of 2 techniques of Vision Stereo for the 3d reconstruction of 2 x - ray images of the fruit of the avocado with seed; This research arises because the freezing time turns out to be much more complex according to the variation of form of the product, and although there are mathematical models with which we can support ourselves to determine freezing results these still remain approximations, since the geometric form Like sphere, ellipses, cylinders are not precise forms that can be compared with other types of fruits like the avocado with seed, since this fruit is only an approximation to geometric figures, as we know it has variation of form what makes that the calculation of Cooling is not adequate; The purpose is then to reconstruct the actual shape of the fruit of the avocado in three dimensions allowing in this way that the calculations of cooling of the fruit is appropriate in the industries that practice this type of processes.

To do this research, has studied the theoretical bases and the preparatory work related to the 3D reconstruction using stereo vision using the method 1: traditional technique and method 2: Technical tracks.

Two x-ray images of the avocado fruit were used as a sample for the test of our implemented methods, the indicators used for the evaluation of each method were the time of processing of the disparity map, time of reconstruction of avocado fruit With seed, percentage of memory consumption in processing the map of disparity, percentage of precision and error in the 3d reconstruction of the fruit of the avocado with seed.



In this research two methods are developed for the 3d reconstruction of the fruit of the avocado with seed, and it was determined that the method that processes the map of disparity more quickly through the input of 2 x-ray images whose dimension is (648x864 pixels) is the Method 1 that allows to prepare the disparity map to be able to reconstruct it based on the traditional technique, obtaining a time of 18.2825 seconds, compared to the method 2 that allows to prepare the map of disparity to be able to reconstruct it based on the technique of tracks, Which takes a time of 19.451 seconds to process the disparity map, taking 1.1685 seconds more than method 1. The method that more quickly reconstructs the fruit of the avocado with seed by the input of 2 x-ray images whose dimension is (648x864 pixels) is Method 1 that allows to see how the fruit is in its third dimension based on the traditional technique, obtaining a time of 18.3825 seconds, compared to the method 2 that allows to see as the fruit is in its third dimension based on The technique of tracks, which takes a time of 19,551 seconds to reconstruct the fruit of the avocado with seed, taking 1.1685 seconds more than method 1. The method that consumes less memory in processing the map of disparity by entering 2 images rays X whose dimension is (648x864 pixels) is Method 2 that allows to prepare the disparity map to be able to reconstruct it based on the technique of tracks, obtaining a percentage of memory consumption of 11.37%, compared to the method 1 that allows to prepare the Disparity map to be able to reconstruct it based on the traditional technique, which consumes 88.63%, consuming 72.26% more memory than method 2.

Comparing both methods by entering 2 x-ray images whose dimension is (648x864 pixels), according to the standard range formula, it can be noticed that



both methods that allow the 3d reconstruction of the fruit of the avocado with seed have an accuracy of $\pm 5.88 \text{ cm}^3$ with respect to the actual object of the fruit of the avocado. This shows that according to the user's perception of seeing the reconstructed object with a better picture, it supports it with this indicator. The difference of these methods is the processing time, reconstruction as well as the memory consumed.

Keywords: 3D Reconstruction, Stereo Vision.



INTRODUCCION

La reconstrucción tridimensional (3D), un área muy investigada en la visión computacional y tiene por objetivo obtener información tridimensional en ambientes reales manteniendo sus características físicas (dimensiones, volumen y forma). En la actualidad existe un gran problema con el cambio fisiológico o deterioro de frutas, para ello existen métodos de conservación como la congelación y la refrigeración que basan su cálculo en las formas geométricas de las frutas; existen modelos matemáticos con el que podemos apoyarnos para determinar resultados de congelamiento pero estos aún siguen siendo aproximaciones, ya que la forma geométrica como esfera, elipses, cilindros no son formas precisas que se puedan comparar con otros tipos de frutos como la palta con semilla que tiene forma irregular y hace que el cálculo de enfriamiento no sea el adecuado.

Es necesario encontrar una metodología adecuada para modelar la forma real de una fruta y ser visto en una computadora manteniendo sus características físicas.

El objetivo de esta investigación es generar una imagen más fiable de una fruta en este caso es la palta con semilla, por el que se propone realizar la reconstrucción tridimensional (3D) a partir de imágenes rayos x. Ahora planteamos el problema: ¿Cómo reconstruir una imagen tridimensional a partir de imágenes rayos x de la fruta de la palta con semilla?, para ello utilizaremos la técnica de visión estéreo.



Teniendo en cuenta lo expuesto, el objetivo de este trabajo fue desarrollar un método que permita modelar la geometría real de la palta a través la técnica de visión estéreo.

I. PROBLEMA DE INVESTIGACIÓN

1.1. Situación problemática

La temperatura es un factor crucial en el mantenimiento en la calidad de los alimentos. La congelación, refrigeración son métodos de conservación de alimentos, es una tecnología aplicada a frutas, hortalizas, productos cárnicos, del mar, preparados y lácteos, entre otros. Es bien conocido que este método de conservación puede afectar las características organolépticas y por consiguiente la calidad de los productos alimenticios. Es por ello que se deben de evaluar las características de los alimentos y las condiciones de los sistemas de refrigeración y congelación que se van a aplicar a alimentos. (Gómez, Cerón, & Rodríguez, 2007)

El caso del enfriamiento o congelación de un producto alimenticio sólido, el mecanismo de transferencia de calor dominante es la llamada conducción, en el cual la transferencia de calor se propaga paulatinamente debido a la agitación molecular, lo que hace que se trate de un sistema de transferencia interno lento. La necesidad de establecer sistemas de ahorro energético dentro del sector de las industrias agroalimentarias y en concreto aquellas que suponen elevados consumos, y aspectos de calidad del producto conservado, exigen no solo analizar equipamiento si no ajustar los cálculos de necesidades de calor o frío del producto a las necesidades reales. (Virsedá & Pinazo, 1998).



(Virsedá & Pinazo, 1998) Explican que, para el cálculo de transferencia de calor, existen modelos aplicables a procesos de enfriamiento sin y con cambio de fase. Dentro de los modelos sin cambio de fase hallamos a los métodos numéricos, y los métodos analíticos. Los métodos numéricos han sido ampliamente estudiados desde 1978 hasta 1995, donde se desarrollaron programas informáticos que trabajaban con el método de los factores de respuesta, para la aplicación en productos agroalimentarios basados en formas geométricas simples (cilindros y esferas).

En los modelos de transferencia de calor aplicables a procesos de enfriamiento con cambio de fase, el problema de enfriamiento y congelación de alimentos ha sido ampliamente estudiado y se ha planteado numerosos modelos matemáticos para la determinación de tiempos de enfriamiento o congelamiento, basados en su mayor parte en el cálculo de diferencia finita y elemento finito, aunque se ha desarrollado también métodos basados en la integración de la ecuación diferencial ordinaria para cambio de fase. Aunque el modelo físicamente más realista para la congelación de materiales biológicos sólidos es el de la transmisión de calor con propiedades térmicas variables. (Comini, Del Guidice, Lewis, & Zienkiewicz, 1974) (Virsedá & Pinazo, 1998).

Diversos investigadores propusieron sus modelos basados en factores de forma, por ejemplo (Pham, 1991), estableció un modelo para calcular tiempos de enfriamiento de productos basado en formas elipsoidales. Así mismo (Hossian, Cleland, & A.C., 1992), desarrollaron una expresión para



factor geométrico con formas elipsoidales aplicando soluciones analíticas, ampliándolo posteriormente a tres dimensiones. (Virsedá & Pinazo, 1998).

Los factores que influyen en el tiempo de enfriamiento de productos alimenticios se deben principalmente a las características del alimento y las condiciones del equipo en el cual se va a llevar a cabo el proceso. Referente al alimento, es necesario conocer su conductividad térmica, sus dimensiones y su temperatura inicial. De las condiciones del equipo, se debe de considerar el coeficiente convectivo de transferencia de calor, el medio de congelación y la temperatura a la cual se encuentra éste. (Heldman & Hartel, 1997) (Gómez-Sánchez, Cerón-Carrillo, Rodríguez-Martínez, & VázquezAguilar., 2007).

La aproximación para estimar el tiempo de enfriamiento, usa la ecuación de Planck la cual fue desarrollada para sistemas ideales que considera constantes, que dependen de las dimensiones y forma del producto a enfriar. (Heldman & Hartel, 1997) establecieron valores estándares para las constantes, para tres formas de productos más comunes como son: El plato infinito, el Cilindro Infinito y la Esfera. Se puede observar que productos con una forma más esférica, tendrán menores tiempos de enfriamiento que los productos con una forma cilíndrica; y los productos con forma cilíndrica tendrán valores de tiempo de enfriamiento más bajos que los productos con forma de placa. (Gómez-Sánchez, Cerón-Carrillo, Rodríguez-Martínez, & Vázquez Aguilar., 2007).

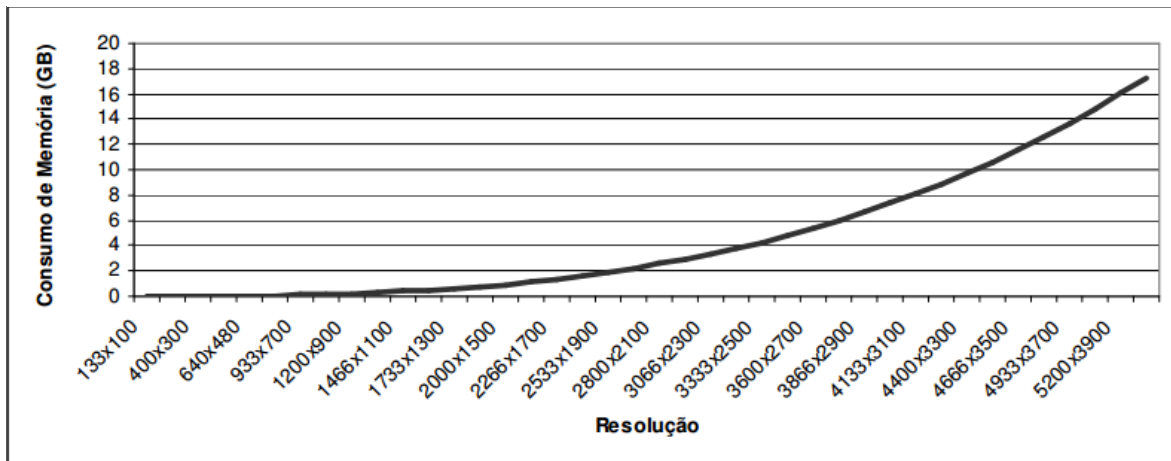


Como se puede observar, los modelos para calcular tiempos de congelamiento toman como referencia la forma del producto, sin embargo, estos cálculos son aproximaciones basadas en figuras geométricas conocidas como esferas, elipses, cilindros, por lo tanto, los resultados corresponden a aproximaciones. Se sabe que los productos agroindustriales en especial, las frutas (la palta) se aproximan a figuras geométricas conocidas, pero tienen múltiples variantes en sus formas, incluso algunas poseen pepa, por lo que un cálculo de enfriamiento no estaría siendo el adecuado.

En los últimos años el avance en las investigaciones de reconstrucción tridimensional (3d) ha ido creciendo considerablemente, sobre todo en la ciencia de la medicina humana, Arquitectura, Robótica, Automatización industrial, Reconocimiento de patrones. Su amplia aplicabilidad hace que sea una área muy investigada, importante y prometedor. Así tenemos el caso de (Marcelo & Roseli, 2007) En su tesis “SISTEMA DE RECONSTRUCCION 3D DE BAJO COSTO”, propuso aplicar visión estéreo para la obtención de información 3d, mediante la Técnica Tradicional. El funcionamiento de un algoritmo tradicional de visión estéreo se siguen los siguientes pasos: Costo de correspondencia, Agregación de evidencias, selección de disparidad, pero la forma de realizar las búsquedas de correspondencias para generar el mapa de disparidad, consume mucho memoria sobre todo en imágenes con resoluciones altas; es por eso que propuso aplicar la Técnica de Pistas que reduce el consumo de memoria sobre todo en imágenes con resoluciones altas. Una vez construido el mapa de disparidad crea una malla 3d.

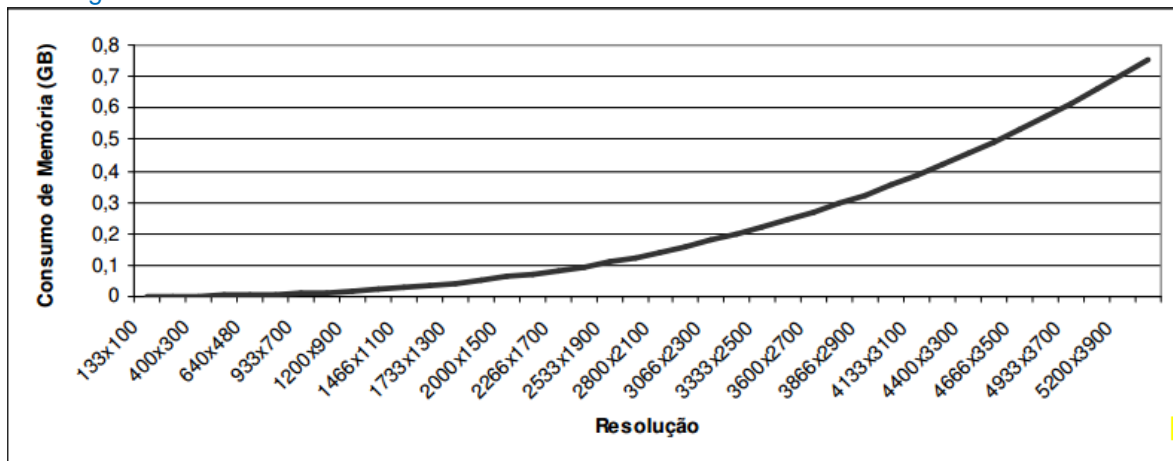


Ilustración 1 Consumo de memoria aplicando la técnica tradicional, con diferentes resoluciones de imágenes.



Fuente: (Marcelo & Roseli, 2007)

Ilustración 2 Consumo de memoria aplicando la técnica de pistas, con diferentes resoluciones de imágenes.



Fuente: (Marcelo & Roseli, 2007)

Por lo que podemos observar es que existen varios métodos para poder realizar la reconstrucción 3D, pero también vemos que han sido en su mayoría aplicadas a otros objetos más no en frutos de formas irregulares. Es por ello que el presente proyecto pretende reconstruir la forma real del producto en tres dimensiones, utilizando el procesamiento de imágenes de rayos X del fruto de la palta con semilla. El proyecto abordará, la captura de imágenes de rayos X del fruto de palta con semilla, el procesamiento del mapa de disparidad y la



reconstrucción tridimensional (3d) del fruto de la palta con semilla, aplicando el método de visión estéreo mediante su técnica tradicional y técnica de pistas por el tema de disminución del consumo de memoria en procesar el mapa de disparidad en imágenes de altas resoluciones, permitiendo de esta manera que los cálculos de enfriamiento del fruto sea el adecuado en las industrias que practican este tipo de procesos.

1.2. Formulación del problema

El problema consiste principalmente, en que los modelos para calcular tiempos de congelamiento para productos que tienen formas irregulares (palta), no son precisos.

Es por ello que se trata de responder la siguiente interrogante, ¿Cómo realizar la reconstrucción de una imagen tridimensional a partir de imágenes de rayos X de frutos con semilla?

1.3. Delimitación de la investigación

La presente investigación será desarrollada teniendo en cuenta los siguientes agregados.

- a) Se analizaran solo imágenes rayos x estéreo.
- b) Se pretende realizar la reconstrucción 3D de imágenes rayos x mediante visión estéreo aplicando la técnica tradicional y la técnica de pistas.
- c) Se tiene en cuenta la reducción de costos, mediante la disminución del consumo de memoria aplicando le técnica de pistas.

1.4. Justificación e importancia de la investigación:

La optimización de los procesos en la industria de alimentos hace uso de herramientas computacionales para modelar y simular la transferencia de calor



y/o de masa, con la finalidad de ahorrar tiempo y dinero durante las etapas de investigación de desarrollo y diseño de los productos.

Una técnica muy utilizada para simular la transferencia de calor y/o de masa es la utilización de del método analítico, específicamente el método de separación de variables, este método fue desarrollado por Josep Fourier (1888 - 1999), con la limitación de uso exclusivamente para formas regulares como (placa infinita, cilindro infinito, esfera) aunque se pueden obtener formas finitas como rectángulos y cubos mediante la intersección de dos y tres placas infinitas respectivamente y cilindros finitos, mediante la intersección de una placa infinita y un cilindro infinito.

Como sabemos algunos de los alimentos, pueden ser fácilmente asemejados a estas formas, pero existen otros a los cuales no se podría, debido a la irregularidad en sus dimensiones, es por esto, que para poder ser más exactos en la simulación de los procesos que involucren transferencia de calor, planteamos una metodología para recuperar imágenes digitalizadas en tres dimensiones a partir de imágenes de rayos X, de alimentos de forma irregular.

Estas aplicaciones son de fácil uso y por lo tanto aplicable en empresas dedicadas a optimizar sus procesos, evitando de esta manera la compra de escáneres o dispositivos costosos, los cuales dificultan la investigación rentable. La validación de la metodología nos permitirá comprobar la precisión de la metodología propuesta de obtención de imágenes digitalizadas en tres dimensiones para que de esta manera ser más precisos durante la simulación y validación de la transferencia de calor durante el calentamiento y enfriamiento de los alimentos. Siendo que la región Lambayeque, es agro-exportadora de



una variedad de frutas, y las empresas para sus procesos de congelamiento utilizan lo que la ciencia, tecnología ha puesto a su disposición hasta hoy, el presente trabajo se justifica y muestra su importancia en la medida en que se pretende contribuir con la generación de tecnología que mejore la competitividad de nuestras empresas en cuanto a exportación de alimentos.

1.5. Limitaciones de la investigación

- a) Para el proceso del mapa de disparidad se ha utilizado el Método 1:
La técnica tradicional que consume mucha memoria en imágenes de resoluciones altas, es por eso que se utilizó el Método 2: la técnica de pistas para disminuir el consumo de memoria.
- b) Para realizar la reconstrucción 3d de imágenes rayos x de fruta con semilla (palta), resulta complicado ya que existe investigaciones de reconstrucción pero en la parte de huesos y otros objetos.

1.6. Objetivos de la investigación

Objetivo General

Aplicar la técnica de visión estéreo para la reconstrucción 3d de imágenes rayos X de frutos con semilla.

Objetivos Específicos

- a) Desarrollar estrategia de adquisición de imágenes de rayos X para frutos con semilla.
- b) Aplicar las técnicas de visión estéreo que contiene los siguientes pasos: Lectura de imágenes, carga de imágenes, búsqueda de correspondencias, generación de la malla 3d.
- c) Evaluar el método propuesto.



II. Marco teórico

2.1. Antecedentes de la investigación

(Ubieto, 2009), en su labor de investigación “Reconstrucción 3D de modelos utilizando técnicas de Visión artificial”, El objetivo consiste en la reconstrucción de objetos basándose en el análisis de imágenes adquiridas desde un par de cámaras dispuestas en forma paralela (cámaras estéreo).

A partir de ese par de imágenes se deberá generar una malla en tres dimensiones, formada por puntos en tres dimensiones y rectas que unan los puntos. Se ha desarrollado un algoritmo que a partir de imágenes, llegue a crear un modelo tridimensional de un objeto. Para conseguir dicho objetivo, en primer lugar estudió las diferentes técnicas desarrolladas para la reconstrucción en 3D, luego analizó los principales detectores de puntos de interés (esquinas y bordes), implantando algunos de ellos como los detectores Harris, KLT, SUSAN y CSS, para la detección de esquinas.

Por último se investigó sobre la segmentación en imágenes, ya que éste es un paso esencial en todo proceso de reconstrucción. Una vez estudiado e implementado todos los detectores de puntos de interés y algoritmos de segmentación, llevó a cabo la programación de un algoritmo que le permitió crear un modelo tridimensional.

Así, el algoritmo final realiza una primera etapa de segmentación donde separa el objeto del fondo. De esta manera, se reduce la zona de trabajo. Después se procede a ejecutar un detector de esquinas, para identificar los puntos de interés del objeto. Por último se lleva a cabo la etapa de reconstrucción en 3D, realizando una correspondencia de puntos entre las dos imágenes para



finalmente obtener sus puntos tridimensionales por un proceso de triangulación.

Para comprobar la eficacia del algoritmo se realizaron varias pruebas con varias imágenes tomadas por una cámara estéreo.

Los mejores resultados obtenidos con el algoritmo son con imágenes con objetos de geometrías rectas. Al tratarse de una reconstrucción estereoscópica basada en esquinas, los resultados obtenidos con objetos redondos no son los óptimos, puesto que estos objetos apenas contienen esquinas.

Finalmente haciendo un estudio comparativo de los distintos métodos. El detector que mejores resultados obtiene es el detector Canny, ya que detecta todos los bordes de los objetos. El resto de los detectores, basados en el cálculo del gradiente no son tan eficaces como el detector de Canny.

Según (Lamecker, 2006), en su trabajo de investigación “Atlas -based 3D-Shape Reconstruction from X-Ray Images”, desarrollado en Berlín, abordan el problema de cómo reconstruir a priori la geometría en 3D a partir de objetos de 2D. Proponen, un modelo estadístico, y un modelo que optimiza una medida de similitud, para evaluar la diferencia entre las proyecciones de la forma y las imágenes de rayos X. Basándose en observaciones de carácter teórico y experimental para medir la distancia entre las siluetas de los objetos en las proyecciones.

Los experimentos fueron realizados por una imagen de rayos x (coro-nal: CO) y dos imágenes de rayos x (CO plus sagital: Co-SA).

Se realizaron 3 experimentos diferentes: Prueba Leave-All-In Test (LI), Leave-OneOut Test (LO), Surface-Optimization (SO). El método fue validado basado en los datos de rayos X (CT) de pelvis con conocidos con calibración de la



cámara. Los valores promedio de los errores de medida, incluso para una sola imagen de proyección fueron hasta unos pocos milímetros dentro de un intervalo donde se puede esperar obtener una estimación suficiente de la geometría 3D.

Según Hosseinian & Arefi (2015), en su trabajo de investigación “3d Reconstruction From Multi-View Medical X-Ray Images – Review And Evaluation Of Existing Methods”, desarrollado en la Escuela de Topografía y Geoespacial Ingeniería de la Universidad de Teherán – Irán, analizaron los métodos existentes para hallar el adecuado para la reconstrucción 3d de imágenes de rayos-X.

Los métodos analizados fueron: métodos basados en puntos (Técnicas de puntos basado en Stereo-Corresponding y Técnicas de puntos basado en Non Stereo Corresponding), métodos basados en contornos, métodos basados en estadísticas de formas, métodos paramétricos, métodos híbridos. Se describe las ventajas y desventajas de los métodos. Se determinó por ejemplo el Método basado en punto dependen en gran medida de la pericia del operador. Es difícil determinar con exactitud y coincidencia de puntos de vista múltiples radiografías. Por lo tanto, los métodos basados en puntos no pueden garantizar la reproducibilidad.

Según (Camacho, 2013), en su tesis “Sistema de Reconstrucción 3D Multicámara”, donde el objetivo es desarrollar un sistema de visión 3D mediante múltiples cámaras y metrología óptica. Se realizó un sistema de reconstrucción de objetos en 3D, el cual fue utilizado para obtener la superficie de una horma de calzado. Se implementó la técnica de línea de luz láser, obteniendo la topografía mediante aproximación Gaussiana. La Técnica de



metrología láser implementada también puede ser usada a nivel industrial, como herramienta de inspección e ingeniería en procesos poco críticos, sin embargo para su uso en procesos en serie se vería limitada por el tiempo de procesamiento.

Se está presentando un sistema sencillo, eficiente y confiable que se encuentran en el mercado dentro del ámbito del sector calzado y biomédico. Es un método no invasivo que presenta componentes con una gran tolerancia a diversas condiciones del medio.

Según (López, 2014), en su tesis “Análisis de cuerpos no rígidos usando algoritmos de difusión para la detección de simetría en modelos 3D”, donde el objetivo es el desarrollo de un algoritmo para encontrar key components y detectar simetría en cuerpos no rígidos. Se presentaron 3 nuevos algoritmos. El primer algoritmo propuesto lo denominamos , Minimal área interest point ,la cual permite detectar key points o puntos de interés mediante un algoritmo eficiente, es decir de complejidad computacional lineal respecto al número de faces de la triangulación del modelo. El segundo algoritmo que propusimos, permite detectar key components basado en procesos de difusión y fue comparado con el método propuesto por Ivan Sipiran, quien recalco el término de key components para referirse a regiones del modelo que representan altas protusiones en la estructura. Últimamente, el tercer algoritmo propuesto, permite detectar la simetría mediante procesos de difusión. En nuestros experimentos, encontramos que la detección de simetría funciona correctamente para el caso de modelos con simetría extrínseca, tanto para la distancia geodésica, como con la distancia por difusión. Sin embargo, nuestro algoritmo, utilizando distancia geodésica falla al encontrar la simetría en



cuerpos no rígidos, es decir con simetría intrínseca. Se propuso nuevos algoritmos para detectar simetría en cuerpos no rígidos, el cual implicó tener en cuenta varios aspectos como, la cantidad de puntos que forman la malla del objeto, la cantidad de triángulos y las técnicas utilizadas para encontrar simetría en el objeto.

2.2. Estado del arte

(Marcelo & Roseli, 2007), en su Investigación “SISTEMA DE RECONSTRUCCION 3D DE BAJO COSTO”, propuso aplicar visión estéreo para la obtención de información 3d, mediante la Técnica Tradicional. El funcionamiento de un algoritmo tradicional de visión estéreo se siguen los siguientes pasos: Costo de correspondencia, Agregación de evidencias, selección de disparidad, pero la forma de realizar las búsquedas de correspondencias para generar el mapa de disparidad, consume mucha memoria sobre todo en imágenes con resoluciones altas; es por eso que propuso aplicar la Técnica de Pistas que reduce el consumo de memoria sobre todo en imágenes con resoluciones altas. Una vez construido el mapa de disparidad crea una malla 3d.

Baka et al (2011), en su trabajo de investigación “2d–3d shape reconstruction of the distal femur from stereo X-ray imaging”, utilizaron la técnica de Modelo de Forma Estadística (SSM). La técnica consiste en la estimación de la posición y forma de reconstrucción de superficies Óseas 3D a partir de 2 o más imágenes de rayos X calibradas.

(Zhenga, 2009), en su trabajo de investigación “A 2d/3d correspondence building method for reconstruction of a patient-specific 3D bone surface model using point distribution models and calibrated X-ray images”, utilizaron la



técnica de Modelo de Distribución de puntos (MDP). La técnica consiste en la construcción 3D de un modelo de superficie del hueso a partir de un número limitado de imágenes rayos X calibradas. En la investigación sostienen que la técnica del Modelo de Distribución de punto (MDP) es una tarea difícil, sobre todo cuando se quiere construir un modelo de un hueso.

2.3. Base teórica científicas

Se presentan las bases teóricas que fueron empleadas a lo largo de la investigación.

2.3.1. Introducción

La reconstrucción 3D tiene por objetivo restablecer la información tridimensional en un ambiente real o virtual a partir de imágenes de prueba.

Los sistemas basados en fotos de forma natural ya trabajan con información de dos dimensiones (x, y), que carece de la profundidad (z). Para obtener la profundidad, la clave está en encontrar la correspondencia entre dos o más imágenes. En los enfoques que utilizan cámaras estéreo, la estrategia se basa el funcionamiento del sistema visual humano, para obtener información tridimensional a partir de dos imágenes, llamada visión binocular (estereoscópica). Cada ojo proyecta en la retina una imagen que es transmitida al cerebro, donde se obtiene la información en tres dimensiones de la disparidad entre las imágenes.

2.3.2. Vision estéreo

La visión por ordenador hace uso del conocimiento de la Inteligencia Artificial (RUSSELL & NORVIG, 2004) y realiza el procesamiento de imágenes cuyo objetivo es extraer información para el reconocimiento de características. Tiene varias ramas, como la visión estéreo, que actúa específicamente sobre la



situación de la extracción de información de profundidad a partir de dos imágenes entrada con desplazamiento lateral.

Según (SCHARSTEIN D. , 1999) describe Visión estéreo, como la escena de inferencias geométricas a partir de dos o más imágenes digitales posicionadas desde posiciones ligeramente diferentes. Esto es realizado sin ningún esfuerzo por el sistema visual humano, lo que se traduce gran parte de la información tridimensional de las diferencias entre los puntos de vista de cada ojo.

Para el caso de dos imágenes, se obtiene la información de profundidad mediante la determinación de la disparidad entre los puntos correspondientes en ambas imágenes.

Ilustración 3 Par de imágenes estéreo.



Fuente: (SCHARSTEIN & SZELISKI, A taxonomy and evaluation of dense twoframe stereo correspondence algorithms., April-June 2002)

Cuando se trabaja con más de dos imágenes, el cálculo de disparidad se hace en pares, es decir el proceso se lleva a cabo sucesivamente entre cada par de imágenes.

Comúnmente se presenta el resultado del mapa de disparidad con tonos claros en escala de gris. Es una representación de dos dimensiones, un escenario tridimensional, donde la profundidad es presentada en forma de tonos de gris.

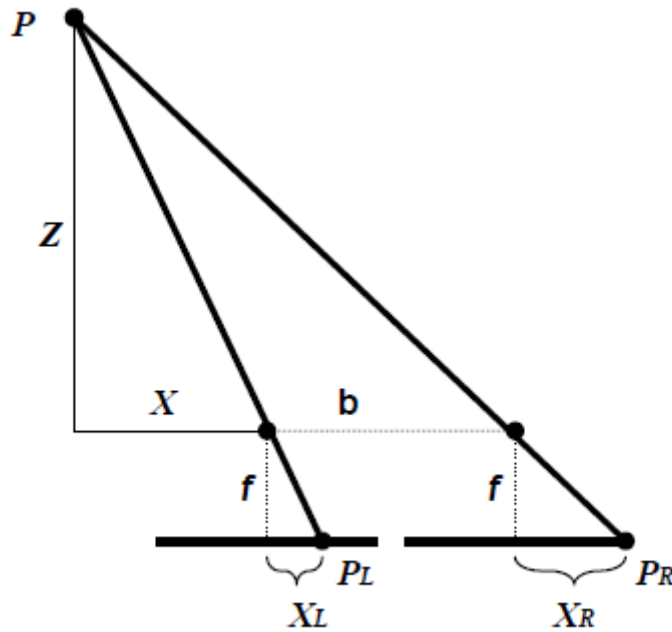
Ilustración 4 Mapa de disparidad.



2.3.3. Geometría

Sea un punto P y sus dos imágenes: PL es la imagen izquierda y PR es la imagen de la derecha. Sea f la distancia focal y b la distancia entre las dos cámaras. Teniendo en cuenta que el punto P tiene una profundidad Z , y un desplazamiento lateral X (en relación con la cámara izquierda) y PL se encuentra en la coordenada XL y PR en la coordenada XR .

Ilustración 5 Ilustración de la geometría estéreo, lo que representa dos cámaras idénticas y paralelas con una distancia focal f y la distancia entre ellos b



Fuente: (SCHARSTEIN D. , 1999)

Llegamos a la conclusión por semejanza de triángulos:

$$\frac{X_L}{f} = \frac{X}{Z}$$

e

$$\frac{X_R}{f} = \frac{X + b}{Z}$$

Para la disparidad d , se tiene:

$$d = X_R - X_L = \frac{f * b}{Z}$$



2.3.4. Correspondencia

Como se puede concluir que en un par de imágenes, un punto PL corresponde al mismo punto P, que tiene un punto PR? Resolver este problema de correspondencia es la parte difícil de la visión estéreo. La dificultades incluyen ambigüedades debido a superficies con textura y / o regiones repetitivas con intensidad uniforme, la incertidumbre causada por ruido.

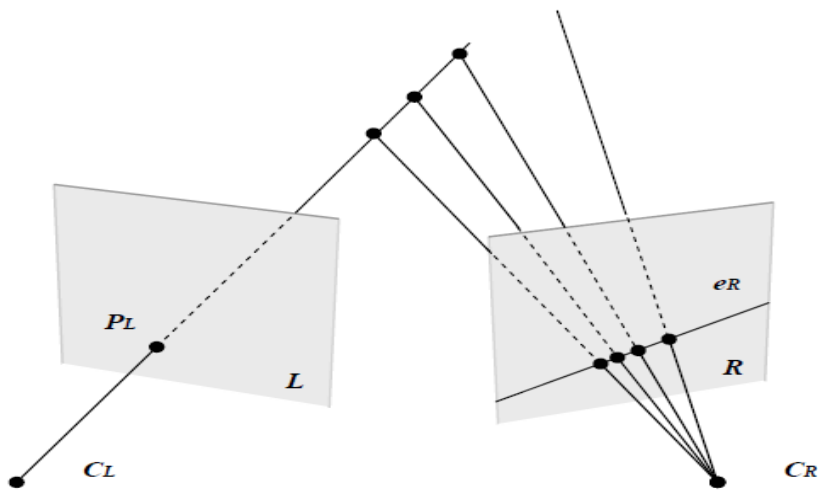
Básicamente hay dos formas de buscar correspondencia. Uno, que señala con mayor certeza, es la de los algoritmos estéreo basado en la característica feature-based, en la función donde la demanda correspondencia se limita a partes tales como los bordes y las esquinas, generando de esta forma un mapa de disparidad escaso que debe ser interpolado si hay la necesidad de información de la profundidad de los puntos no tratados. La otra forma de búsqueda trata de la imagen por regiones (o áreas) que contiene suficiente información para proporcionar la correspondencia más ambigua. Esta forma es conocida como área-based, que tiene la ventaja de producir un mapa de disparidad completa, pero con posibles errores, ya que los puntos distantes de los bordes y las esquinas tienen un menor grado de certeza correspondencia.

2.3.5. Restricción nuclear(Epipolar Constraint)

Con el reto de encontrar puntos correspondientes entre dos imágenes, una pregunta surge: ¿dónde buscar esta correspondencia? Suponiendo dos cámaras cuyos centros de proyección son CL y CR y sus planos de imagen L y R. Considerando que PL es la proyección de un punto P en L, este punto P debe estar alineado por CL y PL, entonces su proyección en R deben pertenecer a eR, en esta línea se conoce como línea de nuclear.



Ilustración 6 Ilustración de restricción nuclear. En este ejemplo, el punto P que genera PL en L debe pertenecer a la línea recta definida por CL y PL , y la proyección P en R debe pertenecer a la recta eR definida como la línea epipolar.



Fuente: (SCHARSTEIN D. , 1999)

Una situación particular de geometría epipolar simple resulta de la utilización de dos cámaras idénticas y paralelas, donde planos de imagen coinciden y los ejes X son paralelos a la línea que conecta sus centros de proyección. En este caso, líneas correspondientes epipolares son horizontal y tienen la misma coordenada y . Esta es la situación ilustrada en la ilustración 5.

La geometría epipolar es la relación entre los puntos de una imagen y su correspondiente línea epipolar en la otra imagen. Puede ser fácilmente calculado si se conocen los valores de ambas cámaras (la posición de los centros de proyección CL y CR y planos de imagen L y R) en algún sistema de coordenadas globales. Esto se conoce como configuración estéreo totalmente calibrados (calibración completa), que es extremadamente rígida y se requiere una gran cantidad de precisión en la construcción del aparato en donde están las cámaras.



Ilustración 7 Instalación estéreo con dos cámaras idénticas y paralelas



Fuente: (SCHARSTEIN D. , 1999)

En la práctica, la restricción epipolar asegura que un cierto punto a la izquierda de la imagen se encuentra en la misma línea a la derecha de la imagen, cambiando únicamente la columna, reduciendo el problema a un 1D búsqueda. Es ampliamente utilizado en los algoritmos de visión estéreo (SCHARSTEIN & SZELISKI, 2001) , porque a pesar de tener un aparato y un montaje más sofisticado permite centrarse en los esfuerzos de investigación y desarrollo para mejorar la calidad o el rendimiento de los algoritmos.

La calibración completa es extremadamente rígida, tanto la demanda de cámaras precisas, como el ajuste y la elección de los parámetros extrínsecos (disposición física: posición y orientación) e intrínsecos (características ópticas



y la geometría interna de la cámara). Esto causa complejidad para los sistemas de reconstrucción 3D.

Es importante resaltar que, con el aumento de resolución, las configuraciones físicas y ópticas de las cámaras se hacen aún más delicada, lo que requiere una mayor precisión, y en consecuencia resulta un equipo más caro.

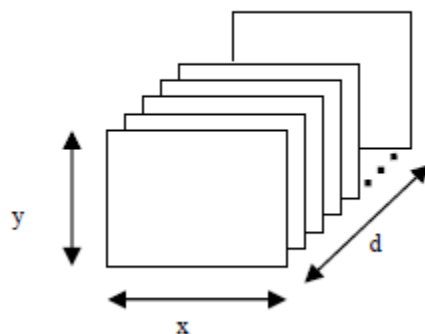
No utilizar una restricción epipolar requiere un algoritmo de búsqueda de correspondencia en una región de dos dimensiones (2D) y ya no en una línea (1D), que se encarga de la ejecución del algoritmo. Dependiendo de la aplicación del algoritmo, esto puede ser un factor crítico.

2.3.6. Enfoques tradicionales de visión estéreo

En general, los algoritmos de visión estéreo se pueden dividir en dos grupos principales: globales y locales. Un método, conocida como semi-Global, según los siguientes autores (HIRSCHMÜLLER H. , 2005), (HIRSCHMÜLLER H. , 2008) , muestra algunas de las características de ambos métodos.

Antes de presentar los métodos globales y locales, es necesario introducir el concepto de Image Space disparidad (DSI), según (BOBICK & INTILLE, 1999), (SUN, 2002), lo definen como un volumen (x, y, d) donde x es el ancho de la imagen de entrada, y la altura y d es el número de niveles de disparidad.

Ilustración 8 disparidad Space Image.



Cada plano se corresponde con la diferencia cuadrada (o absoluta) píxel por píxel entre la imagen izquierda y la imagen de la derecha, siendo que la imagen de la derecha se desplaza por cero (0) a d. Para cada cambio de imagen de la derecha, el resultado se almacena en un plano diferente, a continuación, cada plano representa una disparidad.

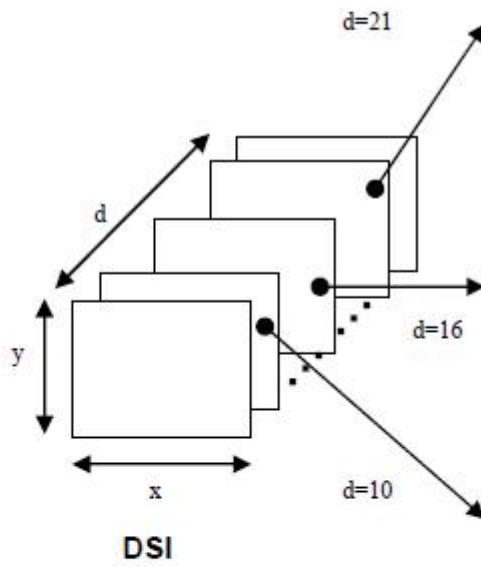
La ilustración 9 ilustra una representación visual en escala de grises contenidos en 3 planos DSI. Nota en la Ilustración 9d o fondo más oscuro (menos diferencia cuadrada) indica que hubo una mejor correspondencia para una disparidad de 10. En la Figura 9c, se observa la región oscura en una profundidad intermedia con una disparidad de 16. En la Figura 9b, la región oscura está por delante (cerca de objetos) y la disparidad es 21 (superior a la disparidad en la Figura 9c $d = 16$ y la Figura 9d $D = 10$), demostrando una relación: cuanto más próximo al objeto esté mayor es la falta de correspondencia.



Ilustración 9 (a) par de entrada estéreo; (b) (c) (d) una representación en escala de grises relacionada con las disparidades 21,26 y 10.



(a)



(b)



(c)



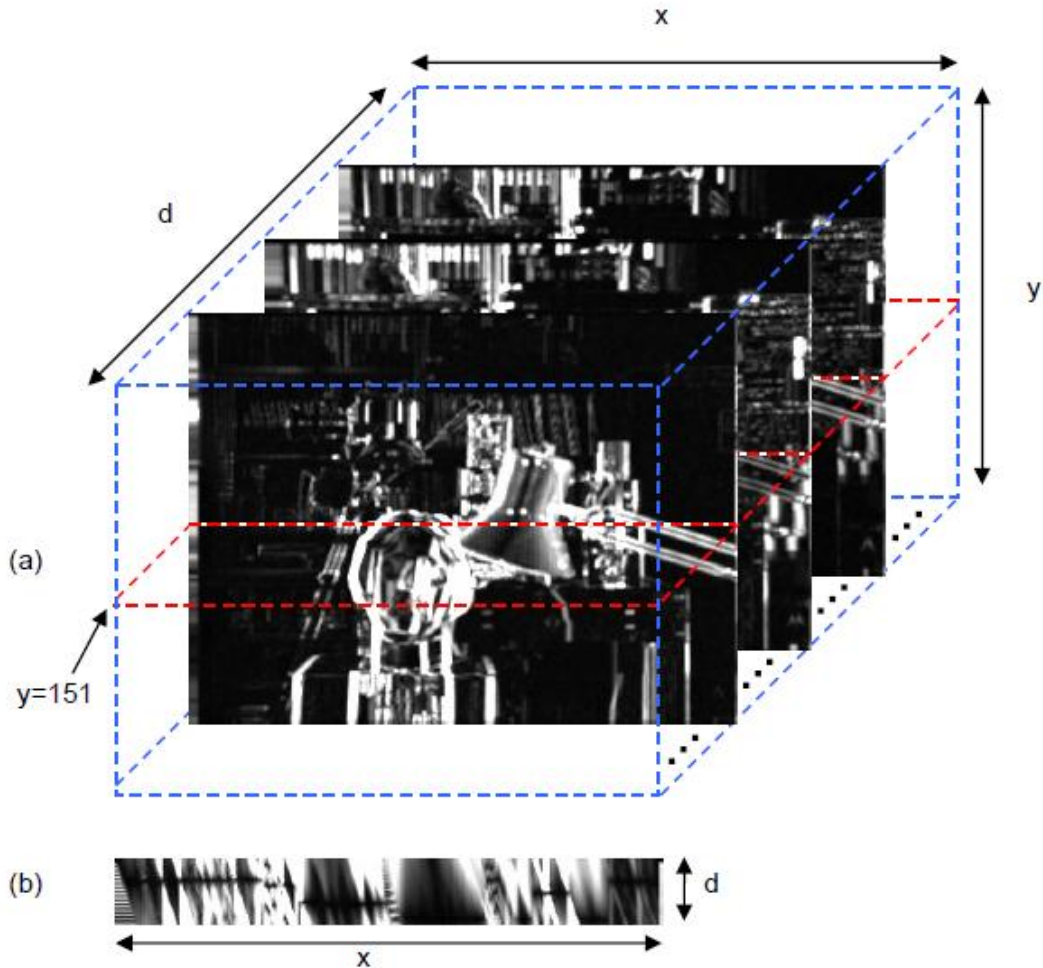
(d)

Fuente: (SCHARSTEIN & SZELISKI, April-June 2002)



Hacer un corte en el DSI en la posición mostrada en la Ilustración 10a, y ver desde arriba (plano), se obtiene la imagen de la Ilustración 10b.

Ilustración 10 - (a) Representación DSI indica un corte en $y = 151$; (B) rebanada (x, d) para DSI $y = 151$. Fuente de las imágenes individuales.

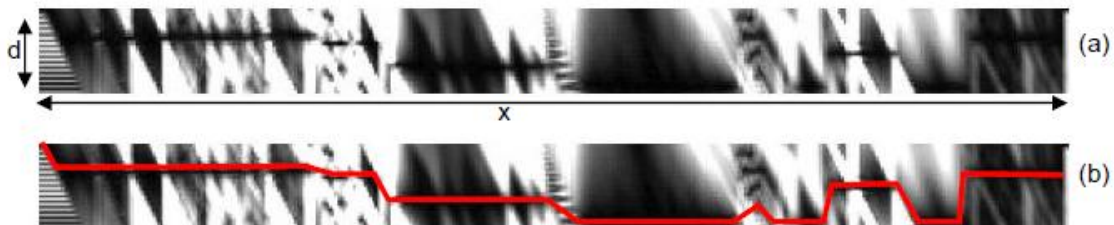


Fuente: (SCHARSTEIN & SZELISKI, 2001)

Métodos globales buscan encontrar una disparidad que disminuya la energía es decir, rastrear la izquierda de la trayectoria a la derecha en el plano (x, d)

(para un determinado DSI) que atraviesa las regiones más oscuras, generando un perfil como se muestra en la ilustración 11b.

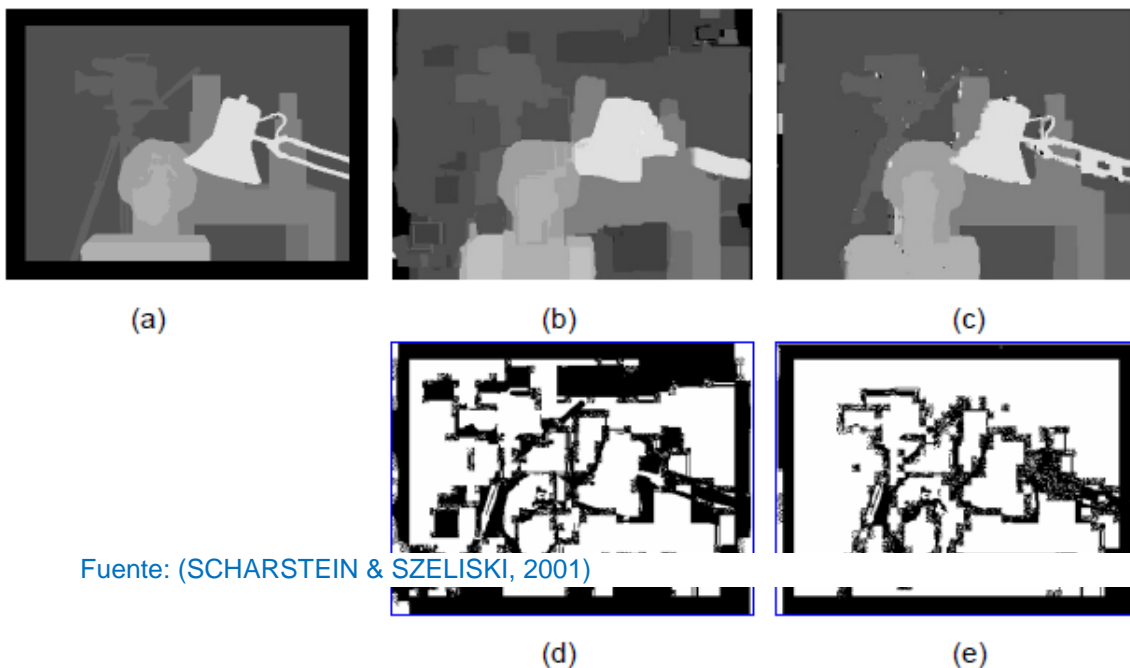
Ilustración 11 - (a) Cortar en una DSI; (b) la indicación de un camino que minimiza la energía.



Fuente: (SCHARSTEIN & SZELISKI, 2001)

Según (BOYKOV, VEKSLER, & ZABIH, 1999.) y (VEKSLER, 1999) , manifiestan que los métodos globales en general producen mapas de disparidad más precisos y se compararon con métodos locales, pero el costo de esta calidad es el tiempo de procesamiento. Se obtuvo el mapa de disparidades Ilustración 12b utilizando un método local tradicional y fue procesado en 1,1s. Se obtuvo el mapa de disparidades de la figura 12c en un método global en un tiempo de 662s. La figura 12a muestra un mapa de disparidad ideal. La precisión del método se ilustra en la Ilustración 12d (método local) y la Ilustración 12e (método global), donde el color blanco indica píxeles igual al mapa de disparidad óptimo.

Ilustración 12 - (a) mapa de disparidad ideal; (b) mapa de disparidad método local; (C) mapa de disparidades método global de los recortes (gráfico); (D) relación entre (a) y (b) donde el blanco indica la igualdad de píxeles y los píxeles negros que indican diferentes; (e) relación de (a) y (c); Observación :. Para las imágenes que tienen bordes blancos, que incluye un contorno azul (fuera de la imagen) para facilitar la visualización.



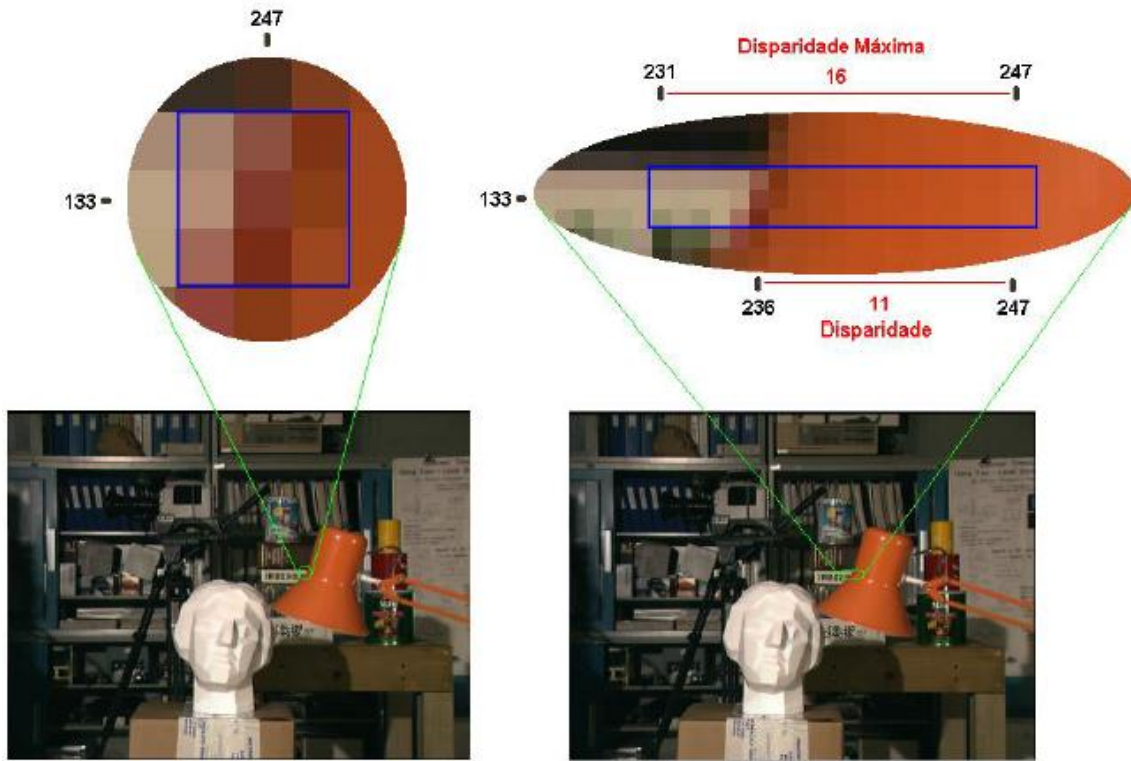
Fuente: (SCHARSTEIN & SZELISKI, 2001)

Según (HANNAH, 1974) Manifiesta que los métodos locales fueron los primeros en ser desarrollado. Desde un punto de la imagen de la izquierda se procura la correspondiente de la imagen derecha. Una forma de medir esta correspondencia es restar los valores de color (R: rojo, V: verde, B: azul) de un

punto de la imagen de la izquierda, de un punto de la imagen de la derecha, y luego realizar la suma de los módulos. Este proceso se realiza sucesivamente por las posibles disparidades, siendo que resultado más bajo se considera la mejor correspondencia. Comparar un punto a otro sería un enfoque muy simplista, que resultaría muchos errores, sin embargo una forma de evitar estos errores es realizar una comparación no sólo en un punto, sino en una región vecina al punto, algo así como una ventana de 3x3 (o 5x5 o incluso 7x7) centrada sobre el punto en cuestión. La Ilustración 13 muestra un ejemplo que considera el uso de restricción epipolar. Tenga en cuenta que el punto de búsqueda para (247,133) de la imagen izquierda comienza exactamente en esta posición en la imagen de la derecha y mueve a la izquierda a partir de $x = 247$ y van hasta $x = 231$.

Ilustración 13 - punto en cuestión y su ventana de 3x3 a la izquierda de la imagen, y la línea epipolar a la derecha de la imagen, se busca la correspondencia.





Fuente: (SCHARSTEIN & SZELISKI, April-June 2002)

Restar los colores de cada píxel de la ventana (en el ejemplo, 3x3), sumar cada módulo de salida, y luego resumir el resultado para cada píxel, este resultado se guarda y el proceso se realiza en la línea epipolar en la de lado derecho. Entre los resultados almacenados, que presenta el valor más bajo se considera la mejor correspondencia. Este enfoque se conoce como suma de las diferencias absolutas o SAD (suma de las diferencias absolutas).

Siendo E la matriz (fijo) con los valores de la ventana izquierda de colores y la matriz D (móvil) con los valores de los colores de la ventana derecha. En la Ilustración 12 se muestra la distribución de estas variables

Ilustración 14 - Descripción de las variables a la ventana de la izquierda y la derecha.

E1	E2	E3
E4	E5	E6
E7	E8	E9

D1	D2	D3
D4	D5	D6
D7	D8	D9

Tiene la secuencia de los siguientes cálculos:

$$DISP1 = | E1.rojo - D1.rojo | + | E1.verde - D1.verde | + | E1.azul - D1.azul |$$

$$DISP2 = | E2.rojo - D2.rojo | + | E2.verde - D2.verde | + | E2.azul - D2.azul |$$

$$DISP3 = | E3.rojo - D3.rojo | + | E3.verde - D3.verde | + | E3.azul - D3.azul |$$

$$DISP4 = | E4.rojo - D4.rojo | + | E4.verde - D4.verde | + | E4.azul - D4.azul |$$

$$DISP5 = | E5.rojo - D5.rojo | + | E5.verde - D5.verde | + | E5.azul - D5.azul |$$

$$DISP6 = | E6.rojo - D6.rojo | + | E6.verde - D6.verde | + | E6.azul - D6.azul |$$

$$DISP7 = | E7.rojo - D7.rojo | + | E7.verde - D7.verde | + | E7.azul - D7.azul |$$

$$DISP8 = | E8.rojo - D8.rojo | + | E8.verde - D8.verde | + | E8.azul - D8.azul |$$

$$DISP9 = | E9.rojo - D9.rojo | + | E9.verde - D9.verde | + | E9.azul - D9.azul |$$

$$Disp. = Disp1 + Disp2 + Disp3 + Disp4 + Disp5 + Disp6 + Disp7 + Disp8 + Disp9$$

En es el píxel en la imagen izquierda, **Dn** píxel de la imagen de la derecha, la disparidad **Disp_n** para ese píxel y, por último, que la ventana **Disp** disparidad 3x3 en cuestión. El **Disp** se calcula para varias posiciones en la imagen de la derecha, estas posiciones son conocidas como los niveles de disparidad, siendo la última posición conocida como la disparidad máxima, es decir, el máximo desplazamiento horizontal de un objeto puede tener.

Otra forma de hacer la búsqueda de correspondencia es restar los colores de los píxeles de cada ventana (por ejemplo, 3x3), añadir el resultado al cuadrado



para cada uno (en lugar de la magnitud) y luego utilizar el mismo procedimiento descrito anteriormente. Este enfoque se conoce como la suma de los cuadrados de las diferencias o SSD (suma de las diferencias al cuadrado). Se observa en varias publicaciones que se incrementó el uso del SSD en lugar de SAD, pero los algoritmos de visión estéreo que trabajan en tiempo real así como lo manifiesta (HIRSCHMÜLLER, INNOCENT, & GARIBALDI, 2002), haciendo referencia a las investigaciones de (KANADE T. e., 1996) , (KIMURA, 1999), utilizan al SAD, sobre todo puesto que el cuadrado utilizado en el SDD es más caro que la elección de un resultado positivo por el SAD. Además, los resultados generados por el SAD permiten el uso de las variables más pequeñas.

Tiene la secuencia de los siguientes cálculos:

$$\text{Disp1}=(E1.\text{rojo}-D1.\text{rojo})^2+(E1.\text{verde}-D1.\text{verde})^2+(E1.\text{azul}-D1.\text{azul})^2$$

$$\text{Disp2}=(E2.\text{rojo}-D2.\text{rojo})^2+(E2.\text{verde}-D2.\text{verde})^2+(E2.\text{azul}-D2.\text{azul})^2$$

$$\text{Disp3}=(E3.\text{rojo}-D3.\text{rojo})^2+(E3.\text{verde}-D3.\text{verde})^2+(E3.\text{azul}-D3.\text{azul})^2$$

$$\text{Disp4}=(E4.\text{rojo}-D4.\text{rojo})^2+(E4.\text{verde}-D4.\text{verde})^2+(E4.\text{azul}-D4.\text{azul})^2$$

$$\text{Disp5}=(E5.\text{rojo}-D5.\text{rojo})^2+(E5.\text{verde}-D5.\text{verde})^2+(E5.\text{azul}-D5.\text{azul})^2$$

$$\text{Disp6}=(E6.\text{rojo}-D6.\text{rojo})^2+(E6.\text{verde}-D6.\text{verde})^2+(E6.\text{azul}-D6.\text{azul})^2$$

$$\text{Disp7}=(E7.\text{rojo}-D7.\text{rojo})^2+(E7.\text{verde}-D7.\text{verde})^2+(E7.\text{azul}-D7.\text{azul})^2$$

$$\text{Disp8}=(E8.\text{rojo}-D8.\text{rojo})^2+(E8.\text{verde}-D8.\text{verde})^2+(E8.\text{azul}-D8.\text{azul})^2$$

$$\text{Disp9}=(E9.\text{rojo}-D9.\text{rojo})^2+(E9.\text{verde}-D9.\text{verde})^2+(E9.\text{azul}-D9.\text{azul})^2$$



$Disp = Disp1 + Disp2 + Disp3 + Disp4 + Disp5 + Disp6 + Disp7 + Disp8 + Disp9$

Siendo **En** píxel de la izquierda de la imagen, **Dn** píxel de la imagen de la derecha, la disparidad **Disp_n** para ese píxel y **Disp** la disparidad de la ventana de 3x3 en cuestión, una vez encontrado la mejor correspondencia, se mide la disparidad. Por ejemplo, si el punto de que se trate en la imagen de la izquierda se encuentra en las coordenadas (22,30) y su correspondiente imagen de la derecha es de (12,30) significa que la disparidad es de 10 píxeles.

Para la creación de un mapa de disparidad, este proceso se realiza en toda la imagen. Una vez obtenidas las disparidades, simplemente asociar a cada uno a escala de grises, siendo que el máximo tendrá un tono más claro (cerca de objeto) y el mínimo tendrá más oscuro (objeto distante).

2.3.6.1. Uso de métodos DSI

Actualmente, DSI se utiliza en ambos métodos globales como en locales. En estos, la DSI es un aspecto importante en la eliminación de cálculos repetitivos, garantizando un excelente rendimiento, pero la desventaja es el alto consumo de memoria.

Se puede codificar un método local con algunos bucles “**for**” sin almacenar los resultados intermedios, pero muchos cálculos se repetirán muchas veces. Supongamos que el punto de análisis (22,30) en la imagen de la izquierda y la que trabaja con una ventana de 3x3. Por lo tanto, se llevarán a cabo los



cálculos de correspondencia (SAD o SSD o cualquier otros) serán calculados nueve veces el número máximo de las diferencias esperadas en la imagen de la derecha, es decir, si el valor de disparidad máxima esperada es de 10 píxeles, se hará 90 veces estos cálculos . Para pasar al siguiente píxel de la imagen de la izquierda (23,30), los 90 cálculos se repiten, y por lo menos 60 se habían calculado directamente tal como lo muestra la Ilustración 15.

Ilustración 15 - Los cálculos repetidos con el desplazamiento de la ventana de la imagen izquierda.

12	25	41	18	93	87	34	40	23	26	54	40	78	62	12	25	41	18	93	87	34	40	23	26	54	40	78	62
36	53	7	29	74	15	29	49	15	4	27	49	71	2	36	53	7	29	74	15	29	49	15	4	27	49	71	2
38	69	89	40	23	26	54	25	83	33	58	25	38	41	38	69	89	40	23	26	54	25	83	33	58	25	38	41
45	78	62	49	15	4	27	1	28	56	37	1	29	74	45	78	62	49	15	4	27	1	28	56	37	1	29	74
37	71	2	25	83	33	58	74	15	29	62	49	15	62	37	71	2	25	83	33	58	74	15	29	62	49	15	62
21	38	41	1	28	56	37	45	78	62	49	15	4	27	21	38	41	1	28	56	37	45	78	62	49	15	4	27
3	57	68	17	6	93	81	37	71	2	25	83	33	58	3	57	68	17	6	93	81	37	71	2	25	83	33	58
Imagen Izquierda							Imagen Derecha							Imagen Izquierda							Imagen Derecha						

$$\begin{aligned}
 &(7 - 4)^2 + (29 - 27)^2 + (74 - 49)^2 + (89 - 33)^2 + (40 - 58)^2 + (23 - 25)^2 \\
 &\quad + (62 - 56)^2 + (49 - 37)^2 + (15 - 1)^2 = DO \\
 &(29 - 27)^2 + (74 - 49)^2 + (15 - 71)^2 + (40 - 58)^2 + (23 - 25)^2 + (26 - 38)^2 \\
 &\quad + (49 - 37)^2 + (15 - 1)^2 + (4 - 29)^2 = DO
 \end{aligned}$$



La forma adoptada para mejorar este proceso es almacenar resultados parciales de cálculos y volver a empaquetar en diferentes etapas de buscar correspondencias. En consecuencia, hay una mayor utilización de la memoria, pero con una ganancia muy significativa en el rendimiento en tiempo de ejecución.

Según (SCHARSTEIN D. , 1999), (SCHARSTEIN & SZELISKI, 2001) la secuencia de procesamiento es la siguiente.

- i. Costo de correspondencia - cálculo de la diferencia cuadrada (u otra), es en esta etapa que la DSI se crea y carga.
- ii. Agregación de pruebas - la suma de todas las diferencias para una ventana dada.
- iii. Selección de disparidades - elija el mejor resultado.

Para ilustrar este enfoque con un ejemplo, si se considera la siguiente situación en la que la distancia máxima (d) es de 10 píxeles, y el par de imágenes en cuestión tiene una resolución $x = 320$ - $y = 200$ cada uno.

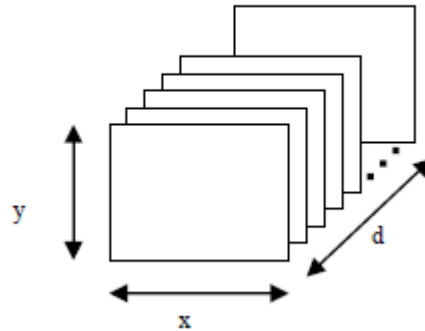
a) Costo de Correspondencia

Se creó la DSI (disparidad espacial de imagen) con 10 planos con 320x200 posiciones, creando un espacio x, y, d (Ilustración 16) la cual se lleva a cabo el proceso de cálculo de la correspondencia (SAD, SSD o cualquier otro) para cada píxel (sin añadir a la ventana) para cada disparidad, es decir, compara el píxel (60,50) de la imagen de la izquierda para el píxel (60,50) de la imagen de la derecha y percibir el resultado en la posición (60,50) del Plano 0 (equivalente a disparidad 0). Luego se Compara el píxel (60,50) desde la izquierda de la imagen con el píxel (59,50) a la derecha de la imagen y almacena el resultado en la posición (60,50), en el Plano 1 la disparidad es 1. Este paso se produce



en ambos métodos locales como globales. Este proceso se conoce como "matching cost".

Ilustración 16 Planos que almacenan los costos de correspondencias.



b) Agregación de pruebas

A continuación, se asignan otros 10 planos con 320x200 posiciones en las que se registran los resultados de las sumas de las ventanas. Es la adición (suponiendo una ventana de 3x3 cuadrados) de los nueve resultados de cada plano de costos de correspondencias en la posición central de esta nueva ventana. Por ejemplo, a un plano de costos de correspondencia, la posición (60,70) se suman los resultados se encuentran en las posiciones mostradas en la Ilustración 17.

Ilustración 17 Posiciones involucrados durante el proceso de agregación al punto (60,70)

(59,70)	(60,70)	(61,70)
(59,71)	(60,71)	(61,71)



Se almacena el resultado de esta suma en la posición (60,70) del plano 3, recién asignado.

Este proceso se conoce como la agregación de las pruebas (en esta descripción, se trabajó con ventana cuadrada).

Dentro del bucle "**for**", lo que hace esta exploración, es habitual llevar a cabo la suma de tres columnas separadas. En este ejemplo, las columnas 59, 60 y 61; cuando el bucle "**for**" avanza de posición (60,70) a la posición (61,70), se resta de la columna 59 y se añade a la columna 62. Esta técnica se usa ampliamente con el fin de mejorar el rendimiento y se conoce como **box filtering** (MCDONNELL, 1981).

Fue desarrollado originalmente para su uso en filtros de imagen, de dónde surgió el nombre más adelante llegó a ser usado en la visión estereoscópica en el proceso de agregación.

En lugar de asignar más de 10 planos, sólo se puede asignar un plano que almacena temporalmente el resultado de la agregación. Cuando termina el proceso de llenado todo el plano, se sobrescribe el plano de costos de correspondencia.

Según (KANADE & OKUTOMI, 1994) manifiesta que la ventana cuadrada tradicional, existen otros enfoques en la forma de adaptar el tamaño de la ventana a la región de que se trate. Este enfoque se conoce como ventana



adaptativa. Así como el autor (ARNOLD, 1983) manifiesta otra forma de trabajo la cual se le conoce como ventana desplazable, que se mueve la ventana, manteniéndolo anclado en diferentes puntos de la ventana principal.

c) Selección de disparidades

Por último, se creó otro plano con 320x200. Se almacena, en cada posición, el valor de la disparidad que obtiene la mejor correspondencia, es decir, el valor más bajo. Por ejemplo, hasta el punto (40,50) presenta en los planos evidencia de la agregación de los resultados de la Ilustración 18, el resultado más bajo se encuentra en el plano 6, lo que significa que la mejor correspondencia fue encontrado con disparidad de 6 píxeles.

Ilustración 18 Simulación de valores en los diferentes planos para la posición (40,50)

Plano	0	1	2	3	4	5	6	7	8	9
Valor	101	232	184	609	900	143	7	240	78	93

Así, se almacena en la posición (40,50) el valor 6. El proceso se lleva a cabo en 320x200 puntos y se conoce como disparidades de selección. A partir de este plano, se crea una imagen en tonos de escalas de grises, en los que, por ejemplo, con una separación máxima de 10 píxeles del 0 sería (0, 0,0) (negro en RGB) y 9 serían (255, 255,255) (blancas RGB) y otros valores de niveles intermedios uniformemente distribuidos.

Elija el número más bajo de la mejor correspondencia que se conoce como "el ganador se lleva todo" (WTA-WINNER –TAKE-ALL). La desventaja de este enfoque es que no garantiza la unicidad de la correspondencia, es decir, un punto de la imagen derecha se puede considerar como una buena correspondencia para varios puntos a la izquierda.



Eventualmente, si es necesario un par de imágenes de la preparación antes de que estos procesamientos, es decir, una etapa de pre-procesamiento. Si se desea obtener una mayor resolución en el mapa de disparidad incluye a una etapa posterior a la selección de las disparidades que son píxeles interpolados entre los ya existentes.

Según (SCHARSTEIN D. , 1999) y (SCHARSTEIN & SZELISKI, 2001) el algoritmo de visión estéreo comprende.

- a. Pre-procesamiento (opcional)
- b. Costo correspondencia
- c. agregación de pruebas.
- d. La selección, las disparidades
- e. Estimación de la disparidad de sub-píxeles (opcional).

Según (KANADE T. e., 1996), (KIMURA, 1999) y (HIRSCHMÜLLER, INNOCENT, & GARIBALDI, 2002) manifiestan que los métodos locales pueden ser codificados de forma eficiente lograr incluso operar en tiempo real. Para lograr estos resultados se utilizan algunas optimizaciones, entre los que cabe mencionar es la técnica de box filtering. (MCDONNELL, 1981)

2.3.7. Comparación de requisitos

Ahora que se observaron los detalles sobre la visión estéreo, puede analizar sus aplicaciones y requisitos, en particular, este trabajo: Reconstrucción 3D de imágenes rayos x estéreo de bajo costo. La Ilustración 19 muestra una comparación entre los requisitos de las diferentes aplicaciones.



Ilustración 19 - Comparación de los requisitos entre las aplicaciones de visión estéreo, con excepción de la reconstrucción 3D de bajo costo

	Cartografia	Navegação	Reconhecimento	Controle Baseado em Visão	Síntese Visual	Reconstrução 3D	Reconstrução 3D de baixo custo
Entrada							
Limitada	Sim (a)	Não	Não	Sim (b)	Não	Não	Não
Sempre com Textura	Sim	Sim (c)	Não	Não	Não	Não	Não
Possui Oclusão	Não (d)	Sim	Sim	Sim	Sim	Sim	Sim
Requisitos							
Saída Densa	Sim	Não	Não (e)	Não	Sim	Sim	Sim
Trata Oclusão	Não	Não	Sim	Não	Sim	Sim	Não
Calibração Completa	Sim	Não (f)	Sim	Sim	Não (g)	Sim	Não
Precisão na Profundidade	Sim	Não	Sim	Sim	Não	Sim	Não (h)
Geometria correta	Sim	Sim	Sim	Sim	Não	Sim	Não
Tempo Real	Não	Sim	Não	Sim	Não	Não	Não

Fuente: (SCHARSTEIN D. , 1999)

- (a) Las imágenes aéreas.
- (b) Por lo general, en un entorno de laboratorio.
- (c) En ambiente cerrado hay regiones sin textura.
- (d) A excepción de la oclusión causada por puentes, viaductos y edificios altos.
- (e) depende del enfoque.
- (f), excepto para permitir la proyección de la dirección de la dirección de una imagen.
- (g) Excepto cuando se hizo la estimación de los puntos de referencia visuales.
- (h) Depende de la precisión en la obtención de imágenes estéreo.

A continuación, los detalles de cada elemento, así como la explicación de respuesta para la reconstrucción 3D de bajo costo:



ENTRADA:

Limitada - significa que las imágenes de entrada tienen características conocidas.

(N) - El objetivo es trabajar con imágenes Lambertianas en cualquier entorno.

Siempre con textura - por lo general significa que la aplicación de fotos son ricos en detalles (en general, imágenes de la naturaleza tienen esta característica).

(N) - No se puede esperar que todos los objetos tienen textura, siempre existe la posibilidad de montar objetos con superficies homogéneas.

Tiene la oclusión - oclusión se produce siempre en imágenes de objetos cercanos.

(Sí) - ambientes libres tiene siempre la oclusión, por otro lado, el uso de la vista estéreo con sólo dos imágenes no trata la oclusión.

REQUISITOS:

Salida densa - requiere que el mapa de disparidad es denso, es decir, obtienen con técnicas basadas en el área (mientras que las aplicaciones que no tienen este requisito pueden utilizar técnicas basadas en la función).

(Sí) - Como resultado de la aplicación es una malla 3D del medio ambiente fotografiado, no hay lugar para mostrar sólo algunas de las características de la imagen.

Esta oclusión - la aplicación requiere que la oclusión es causada por paralaje resuelto.

(No) - Para el tratamiento de la oclusión, un enfoque con más de un par estéreo sería necesaria según (KANADE T. e., 1996) , (KIMURA, 1999) , describen



que no produce una solución de bajo costo, por lo que no es objetivo de este trabajo.

Calibración completa - requiere calibración rígida de las cámaras.

(N) - Calibración completa presenta muy fuertemente al sistema, ya que el propósito de este trabajo es la flexibilidad, la calibración completa no se ha adoptado.

Exactitud de profundidad - requisito de que la profundidad de los datos sobre el mapa de disparidades son exactos.

(No) - Al no utilizar calibración completa y requerir equipo preciso, el nivel de precisión puede variar dependiendo de la precisión de la obtención de proceso de imágenes estéreo.

Geometría correcta - equivalente a la precisión en profundidad, pero considera todas las coordenadas.

(No) - Debido a la utilización de cámaras convencionales y calibración flexibles, no garantiza que la geometría es completamente correcto

Tiempo real - procesamiento de mapa de disparidad en tiempo real sólo es necesario en la navegación y el control basado en la visión (control visual).



2.3.8. Trabajo relacionado

Hay varios estudios que se relacionan con la visión estéreo en reconstrucción 3D. De hecho, el producto se utiliza normalmente para las técnicas de visión estéreo, el mapa de disparidad es una representación tridimensional del medio ambiente, y la profundidad se muestra en forma de tonos de gris.

Una obra que merece la atención fue el libro desarrollado por (SCHARSTEIN D. , 1999) mediante visión estéreo (desarrollado a partir de su tesis doctoral) y aborda el uso de la visión estereoscópica en la síntesis visual.

Una síntesis Visual busca producir una imagen sintética (artificial) en un determinado ángulo de otras imágenes reales en diferentes ángulos. Las técnicas utilizadas comúnmente buscan hacer un giro ("warping") de las imágenes reales produciendo una nueva, simulando el ángulo deseado. La propuesta de Daniel Scharstein era mejorar la síntesis visual, con la información tridimensional mapa de disparidad obtenido producido por técnicas de visión estéreo, lo que produce imágenes sintéticas más realistas.

El trabajo de Daniel Scharstein está relacionado con este trabajo porque es una muy profunda investigación en visión estéreo con el fin de relacionarlo con otra área, en su caso, la síntesis visual. Esta investigación fue inspirador para la preparación de este trabajo, ya que ambos se centran en alternativas para modificar las técnicas tradicionales y no sólo se utilicen.



2.3.9. Herramientas utilizadas para la reconstrucción.

Opencv-Open Computer Vision

OpenCV es una biblioteca libre de visión artificial originalmente desarrollada por Intel. Desde que apareció su primera versión alfa en el mes de enero de 1999, se ha utilizado en infinidad de aplicaciones. Desde sistemas de seguridad con detección de movimiento, hasta aplicativos de control de procesos donde se requiere reconocimiento de objetos. Esto se debe a que su publicación se da bajo licencia BSD, que permite que sea usada libremente para propósitos comerciales y de investigación con las condiciones en ella expresadas.

Open CV es multiplataforma, existiendo versiones para GNU/Linux, Mac OS X y Windows. Contiene más de 500 funciones que abarcan una gran gama de áreas en el proceso de visión, como reconocimiento de objetos (reconocimiento facial), calibración de cámaras, visión estéreo y visión robótica.

El proyecto pretende proporcionar un entorno de desarrollo fácil de utilizar y altamente eficiente. Esto se ha logrado, realizando su programación en código C y C++ optimizados, aprovechando además las capacidades que proveen los procesadores multi núcleo. OpenCV puede además utilizar el sistema de primitivas de rendimiento integradas de Intel, un conjunto de rutinas de bajo nivel específicas para procesadores Intel.

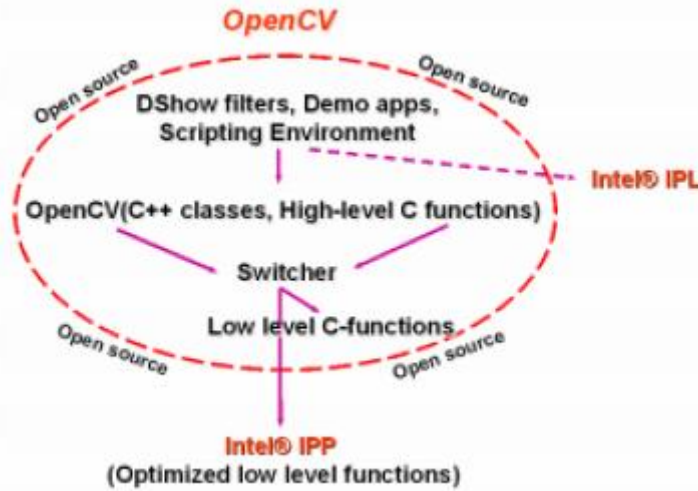
Estructura y características de la Librería OpenCV

La librería OpenCV está dirigida fundamentalmente a la visión por computador en tiempo real. Entre sus muchas áreas de aplicación destacarían: interacción



hombre-máquina (HCI4); segmentación y reconocimiento de objetos; reconocimiento de gestos; seguimiento del movimiento; estructura del movimiento (SFM5); y robots móviles

Ilustración 20 Estructura de la librería OpenCV



Fuente: www.opencv.org

Open CV proporciona filtros Microsoft® DirectShow para realizar tareas tales como: calibración de la cámara, seguidores de objetos (Kalman tracker y ConDensation tracker), etc.

La librería OpenCV proporciona una gran diversidad de entornos y todas estas herramientas de alto nivel hacen uso de un paquete de clases C++ y funciones C de alto nivel que utilizan a su vez funciones muy eficientes escritas en C. Concretamente, el conjunto de funciones suministradas por la librería OpenCV se agrupan en los siguientes bloques:

- a. Estructuras y operaciones básicas: matrices, grafos, árboles, etc.
- b. Procesamiento y análisis de imágenes: filtros, momentos, histogramas, etc.
- c. Análisis estructural: geometría, procesamiento del contorno, etc.
- d. Análisis del movimiento y seguimiento de objetos: plantillas de movimiento, seguidores (i.e. Lucas-Kanade), flujo óptico, etc.



- e. Reconocimiento de objetos: objetos propios (eigen objects), modelos HMM, etc.
- f. Calibración de la cámara
- g. Reconstrucción tridimensional (funcionalidad experimental): detección de objetos, seguimiento de objetos tridimensionales, etc.

Lenguaje de Programación C++

C++ es un lenguaje de programación diseñado a mediados de los años 1980 por Bjarne Stroustrup. La intención de su creación fue el extender al lenguaje de programación C mecanismos que permiten la manipulación de objetos. En ese sentido, desde el punto de vista de los lenguajes orientados a objetos, el C++ es un lenguaje híbrido.

Posteriormente se añadieron facilidades de programación genérica, que se sumaron a los paradigmas de programación estructurada y programación orientada a objetos. Por esto se suele decir que el C++ es un lenguaje de programación multiparadigma.

Actualmente existe un estándar, denominado ISO C++, al que se han adherido la mayoría de los fabricantes de compiladores más modernos. Existen también algunos intérpretes, tales como ROOT.

Una particularidad del C++ es la posibilidad de redefinir los operadores, y de poder crear nuevos tipos que se comporten como tipos fundamentales.

2.3.10. Conclusión

En este capítulo, se presentaron los detalles de las principales técnicas de visión estereo .Se presentaron métodos globales y locales, así como también la comprensión del funcionamiento de los métodos locales, como también las herramientas a emplear en esta investigación.



2.4. Definición de términos básicos

Visión Estéreo: Es la escena de inferencias geométricas a partir de dos o más imágenes digitales posicionadas desde posiciones ligeramente diferentes. Esto es realizado sin ningún esfuerzo por el sistema visual humano, lo que se traduce gran parte de la información tridimensional de las diferencias entre los puntos de vista de cada ojo.

Mapa de disparidad: Es una manera de presentar la información tridimensional, donde se tiene la disposición espacial (x, y) y la información de la disparidades (profundidad inversa) presenta en forma de tonos de gris.

Image Space disparidad (DSI): Es el volumen (x, y, d) donde x es el ancho de la imagen de entrada, y la altura y y d es el número de niveles de disparidad.

Costo de correspondencia: Cálculo de la diferencia cuadrada (u otra), es en esta etapa que la DSI se crea y carga.

Agregación de pruebas: Es la suma de todas las diferencias para una ventana dada.

Selección de disparidades: Es la elección del mejor resultado en base a la agregación de pruebas realizadas.

Formas regular: Son figuras como las geométricas: cuadrado, círculo, triángulo, rectángulo, etc. Siempre son de la misma forma y tienen denominación.



Formas irregulares: Son las que utilizando líneas y arcos concretos no poseen regla alguna. Suelen ser unión de dos o más formas geométricas regulares.

Imagen 3D: Resultado final del proceso de renderizado de un modelo en 3D. Una imagen en 3D, en definitiva, es una imagen en dos dimensiones que simula las tres dimensiones.

Renderización: es el proceso de generar una imagen (imagen en 3D o una animación en 3D) a partir de un modelo, usando una aplicación de computadora.

Imagen: Es una representación visual, que manifiesta la apariencia visual de un objeto real o imaginario.

Técnica: es un procedimiento o conjunto de reglas, normas o protocolos que tiene como objetivo obtener un resultado determinado, ya sea en el campo de las ciencias, de la tecnología, del arte, del deporte, de la educación, de la investigación, o en cualquier otra actividad.

Algoritmo: Método que describe la Solución de un problema computacional, mediante una serie de pasos precisos, ordenados y finitos.

Método: Procedimiento utilizado para llegar a un fin.

Procesamiento de imágenes: Tiene como objetivo mejorar el aspecto de las imágenes y hacer más evidentes en ellas ciertos detalles que se desean hacer notar. La imagen puede haber sido generada de muchas maneras.



Modelado tridimensional: Es un trabajo de arte gráfico, es el proceso de desarrollar una representación matemática de cualquier objeto tridimensional, envuelve por completo la superficie tridimensional.

Rayos-X: Radiación electromagnética que atraviesa cuerpos opacos a la luz ordinaria, tiene mayor uso en imágenes médicas.

III. MARCO METODOLÓGICO

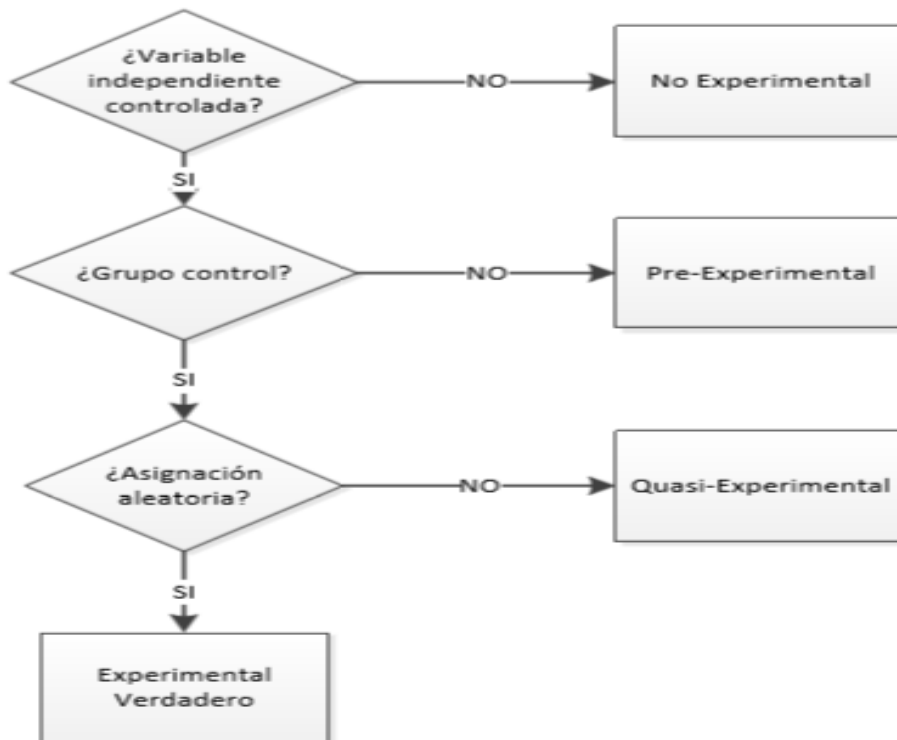
3.1. Tipo y diseño de la investigación

3.1.1. Tipo de investigación

Tecnológica Aplicada, porque tiene como objetivo crear nueva tecnología a partir de conocimientos adquiridos a través de la investigación estratégica, aplicada; su principal objetivo está basado en resolver problemas prácticos, con un margen de generalización limitado. (Tam, Vera, & Oliveros, 2008).

3.1.2. Diseño de la investigación:

Cuasi experimenta, por el motivo que no se utiliza ningún tipo de selección aleatoria o proceso de pre-selección, para escoger los grupos en las que se prueba una variable.



Proceso de pre-selección de variables.

Fuente (Tam, Vera, & Oliveros, 2008)

3.2. Población y muestra:

3.2.1. Población

La presente investigación se realizará con una población de imágenes rayos x de palta.

3.2.2. Muestra

(Baka, Kaptein, Giphart, & Niessen, 2011) En su investigación proponen un método de estimación de pose y forma la reconstrucción en 3D de las superficies óseas de dos (o más) calibrado imágenes de rayos X.

Tomando como referencia dicha investigación mi muestra inicial seria 2 imágenes rayos x calibradas de la fruta palta.



3.3. Hipótesis

Si se aplica la técnica de Vision Estero, se realizará la reconstrucción 3D de imágenes rayos X de fruto con semilla (palta).

3.4. Operacionalización

Variable Dependiente	Dimensiones	Indicadores	Fórmula
Reconstrucción 3D de imágenes rayos x de fruto con semilla	<ol style="list-style-type: none"> 1. Medición del método tradicional 2. Medición de método PISTAS. 	Tiempo de procesamiento de cada método para obtener el mapa de disparidad.	$TP = \text{Tiempo inicio} - \text{Tiempo final.}$



		Tiempo de reconstrucción de cada método.	TR=Tiempo inicio - Tiempo final.
		% de consumo de memoria de cada método para procesar el mapa de disparidad.	Tradicional=16xy + 4xyd
			PISTAS=12xy+4xd+4x(1+j)d
		Precisión final de la reconstrucción	P=Valor máximo-Valor mínimo.

3.5. Métodos, técnicas e instrumentos de recolección de datos

Sampieri (2010), La recaudación de datos “implica un plan detallado de procedimientos que conduzcan a reunir los datos que permitan comprobar el grado de validez de la hipótesis”. “Los datos se recolectan de la muestra seleccionada, la cual contiene, teóricamente, las mismas características que se desean investigar en la población de interés”. Desprendiendo: para demostrar el grado de validez de mi hipótesis requiero datos procesados a partir de la población, puesto que la población son las imágenes.



Para la recolección de datos se aplicará la siguiente técnica: Visión Estéreo la cual consiste en los siguientes pasos:

- a. Pre-procesamiento(opcional)
- b. Costo de correspondencia
- c. Agregación de evidencias
- d. Selección de disparidad
- e. Calibración rígida
- f. Precisión de color
- g. Generación de la malla 3d
- h. Rendimiento.

3.6. Procedimiento para la recolección de datos

Técnica de Visión Estéreo.

- a) Carga de imágenes: se carga 2 imágenes rayos x.
- b) Mapa de disparidad: Es una representación tridimensional.
- c) Reconstrucción 3D: se reconstruye a partir del mapa de disparidad.

3.7. Plan de análisis estadístico de datos

La evaluación del método, se realizara el cálculo mediante la siguiente fórmula del rango estándar.

Precisión

$P = \text{Valor máximo} - \text{Valor mínimo}$

3.8. Criterios éticos

Los criterios éticos cruciales para mi proyecto de investigación son:

Justificación al fin: Que toda decisión que he tomado persiga una meta, con el fin de buscar un resultado que se consigue con la ejecución de mi decisión tomada, la meta tiene que ser legítima.

La reflexión Ética

Tenemos que tener en cuenta la decisión entre varias alternativas en un proceso de decisión estratégica en una comunidad de personas.

Desde la reflexión filosófico-moral o ética se fundamentan y legitiman escalas de valores que no pueden ser ignorados en ningún ámbito de acción humana, se realizó aporte crítico en nuestro tema de investigación.

Confidencialidad: asegurar la identidad de las fuentes como también de los autores a quien hago referencia.

3.9. Criterios de rigor científico

Fiabilidad: Se asegura que los resultados representan algo verdadero y las respuestas que dan los participantes son independientes de las circunstancias de la investigación.

Consistencia: Se busca realizar una investigación con información verídica.

IV. ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

4.1. Resultados

4.1.1. Pruebas Realizadas

En este apartado se muestran las pruebas realizadas a los 2 métodos propuestos basados en la técnica de visión estéreo para la reconstrucción 3d de imágenes rayos x de fruto con semilla (palta).

Se realizaron las pruebas de los métodos propuestos en el capítulo 5.



A. Método 1: **Técnica Tradicional**

B. Método 2: **Técnica de pistas**

Aplicación desarrollada

En este proyecto se desarrolló una aplicación en el lenguaje de programación C++, y la librería OpenCv, además se trabajó con la aplicación DirectX para, para poder implementar los métodos propuestos en el capítulo 5.

Equipo para la puesta a prueba de los métodos

Las características más importantes de la computadora donde se realizaron las pruebas, se detallan a continuación:

Sistema Operativo: Windows 8.1 Pro 64 Bits.

Memoria Ram: 4 Gb.

Microprocesador: Intel(R) Core (TM) i3-2350 CPU @ 2.30GHz

Indicadores

Todos los métodos propuestos fueron probados para poder medir y comparar los siguientes indicadores:

a) Tiempo que tarda cada método en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles).



- b) Tiempo que tarda cada método en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles).
- c) Tiempo que tarda cada método en realizar la reconstrucción 3d de imágenes rayos x del fruto con semilla (palta), mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles).
- d) Tiempo que tarda cada método en realizar la reconstrucción 3d de imágenes rayos x del fruto con semilla (palta), mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles).
- e) Consumo de memoria de cada método en procesar el mapa de disparidad, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles).
- f) Consumo de memoria de cada método en procesar el mapa de disparidad, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles).
- g) Precisión final de la reconstrucción del fruto de la palta con semilla.

4.1.2. Resultados de las pruebas

- a) Tiempo que tarda cada método en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles)**

Este indicador es el resultado de tiempo que tarda cada método en procesar el mapa de disparidad. Para calcular el tiempo que tarda en procesar el mapa de



disparidad, el algoritmo obtiene primero la hora de inicio (hora actual del sistema) en milisegundos y al final de procesarse el algoritmo capturamos la hora fin (hora actual del sistema), para después restar y obtener el tiempo en milisegundos que demora cada método, mostrando al final el tiempo convertido a segundos.

Tiempo Procesamiento=Tiempo Final – Tiempo Inicial

A continuación se presenta los resultados del tiempo que tarda en procesar cada método el mapa de disparidad mediante el ingreso de 2 imágenes rayos x que se ha establecido como muestra.

A continuación se presenta los resultados del tiempo que tarda en procesar el mapa de disparidad teniendo la dimensión de la imagen así como los siguientes parámetros (Disparidad máxima, variación máxima de línea, tipo de búsqueda, resolución de malla).

Tabla 1 Tiempo que tarda cada método en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles)

PRUEBA	TECNICA	DIMENSION (pixeles)	DISPARIDAD MAXIMA	Δ MAXIMA DE LINEA	TAMAÑO DE VENTANA	TIPO DE BUSQUEDA	RESOLUCION DE MALLA	TIEMPO DE PROCESAMIENTO (DISPARIDAD (s))	Δ de Tiempo(s)	Tiempo de Reconstrucción(s)
1	TRADICIONAL	648x864	50	3	21	Intensidad	64 64	9.991	-1.575	10.091
2	PISTAS	648x864	50	3	21	Intensidad	64 64	11.566		11.666
3	TRADICIONAL	648x864	60	3	21	Intensidad	64 64	13.611	0.126	13.711
4	PISTAS	648x864	60	3	21	Intensidad	64 64	13.485		13.585
5	TRADICIONAL	648x864	160	3	21	Intensidad	64 64	21.502	-2.421	21.602



6	PISTAS	648x 864	160	3	21	Intensidad	64 64	23.923		24.023
7	TRADICIONAL	648x 864	150	3	21	Intensidad	64 64	28.026	-0.804	28.126
8	PISTAS	648x 864	150	3	21	Intensidad	64 64	28.83		28.93

Fuente: Elaboración propia

Mostramos a continuación una tabla resumida teniendo en cuenta el método, tamaño de la imagen, parámetros y el tiempo de procesamiento por cada prueba realizada.

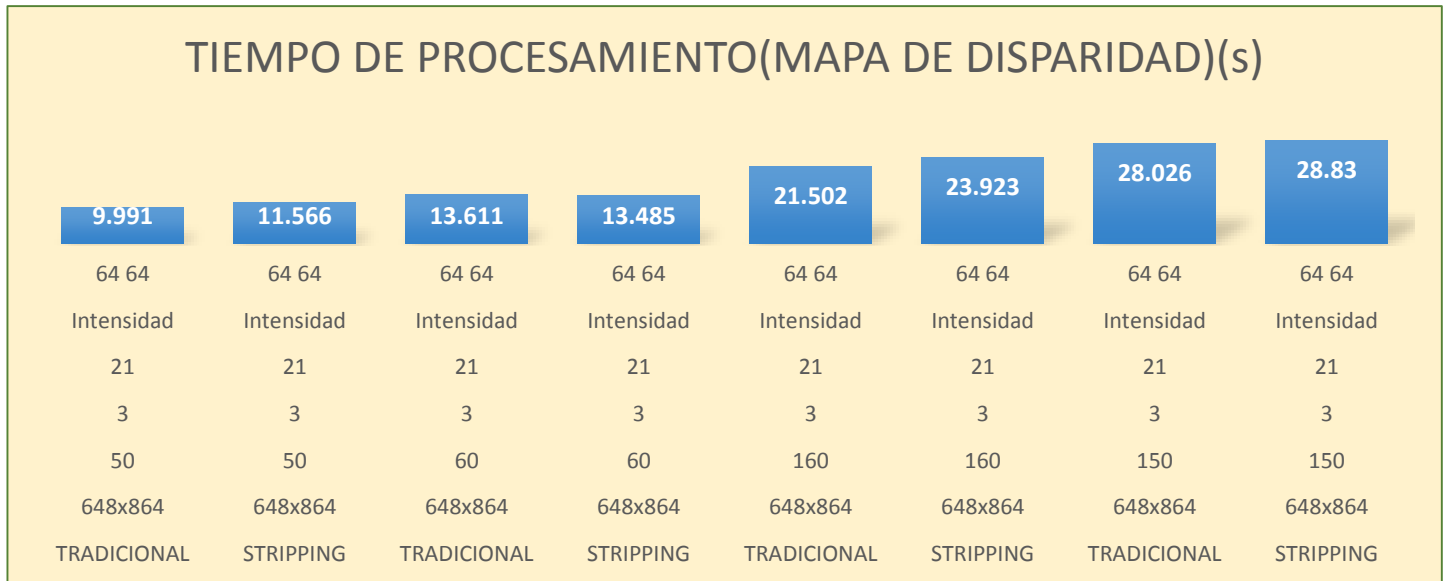
Tabla 2 Resumen de la tabla del Tiempo que tarda cada método en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles)

TECNICA	DIMENSION (pixeles)	DISPARIDAD MAXIMA	Δ MAXIMA DE LINEA	TAMAÑO DE VENTANA	TIPO DE BUSQUEDA	RESOLUCION DE MALLA	TIEMPO DE PROCESAMIENTO(MAPA DE DISPARIDAD)(s)
TRADICIONAL	648x864	50	3	21	Intensidad	64 64	9.991
PISTAS	648x864	50	3	21	Intensidad	64 64	11.566
TRADICIONAL	648x864	60	3	21	Intensidad	64 64	13.611
PISTAS	648x864	60	3	21	Intensidad	64 64	13.485
TRADICIONAL	648x864	160	3	21	Intensidad	64 64	21.502
PISTAS	648x864	160	3	21	Intensidad	64 64	23.923
TRADICIONAL	648x864	150	3	21	Intensidad	64 64	28.026
PISTAS	648x864	150	3	21	Intensidad	64 64	28.83

Fuente: Elaboración propia



Gráfico 1 Tiempo que tarda cada método en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles)



Fuente: Elaboracion propia

Mostramos a continuación el tiempo promedio que demora cada método para poder procesar el mapa de disparidad.

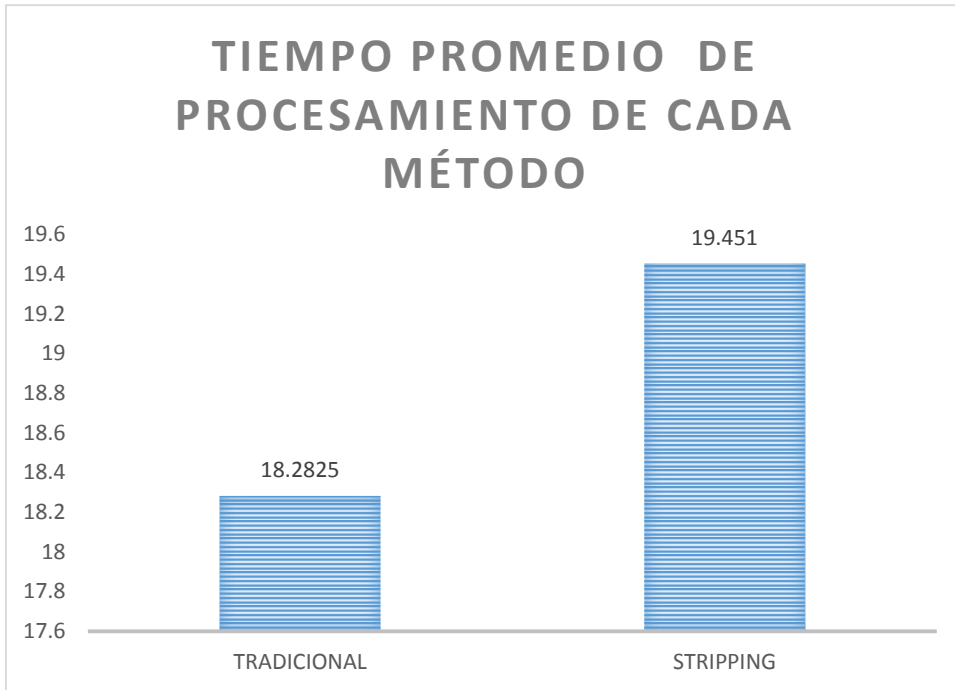
Tabla 3 Tiempo Promedio que tarda cada método en procesar el mapa de disparidad, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles).

Tiempo Promedio de procesamiento de Cada Método	
TECNICA	TIEMPO DE PROCESAMIENTO(MAPA DE DISPARIDAD)(s)
TRADICIONAL	18.2825
PISTAS	19.451

Fuente: Elaboración propia

Gráfico 2 Tiempo promedio de procesamiento de cada método, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles)





Fuente: Elaboracion propia.

Obteniendo como resultado que el método que procesa de manera más rápida el mapa de disparidad es el Método 1 que permite preparar el mapa de disparidad para poder reconstruirlo basándose en la Técnica Tradicional, la diferencia de tiempo es de 1.1685 segundos en comparación con el Método 2 que permite procesar el mapa de disparidad utilizando la Técnica de Pistas.



b) Tiempo que tarda cada método en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles)

Este indicador es el resultado de tiempo que tarda cada método en procesar el mapa de disparidad. Para calcular el tiempo que tarda en procesar el mapa de disparidad, el algoritmo obtiene primero la hora de inicio (hora actual del sistema) en milisegundos y al final de procesarse el algoritmo capturamos la hora fin (hora actual del sistema), para después restar y obtener el tiempo en milisegundos que demora cada método, mostrando al final el tiempo convertido a segundos.

Tiempo Procesamiento=Tiempo Final – Tiempo Inicial

A continuación se presenta los resultados del tiempo que tarda en procesar cada método el mapa de disparidad mediante el ingreso de 2 imágenes rayos x que se ha establecido como muestra.

A continuación se presenta los resultados del tiempo que tarda en procesar el mapa de disparidad teniendo la dimensión de la imagen así como los siguientes parámetros (Disparidad máxima, variación máxima de línea, tipo de búsqueda, resolución de malla).



Tabla 4 Tiempo que tarda cada método en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles)

PRUEBA	TECNICA	DIMENSION N (pixeles)	DISPARIDAD MAXIMA	Δ MAXIMA DE LINEA	TAMAÑO DE VENTANA	TIPO DE BUSQUEDA	RESOLUCION DE MALLA	TIEMPO DE PROCESAMIENTO (DISPARIDAD (s))	Δ de Tiempo(s)	Tiempo de Reconstrucción(s)
1	TRADICIONAL	1792x2392	50	3	21	Intensidad	64 64	56.529	14.737	56.629
2	PISTAS	1792x2392	50	3	21	Intensidad	64 64	41.792		41.829
3	TRADICIONAL	1792x2392	60	3	21	Intensidad	64 64	64.487	2.53	64.587
4	PISTAS	1792x2392	60	3	21	Intensidad	64 64	61.957		62.057
5	TRADICIONAL	1792x2392	160	3	21	Intensidad	64 64	168.456	-2.421	168.556
6	PISTAS	1792x2392	160	3	21	Intensidad	64 64	170.877		170.977
7	TRADICIONAL	1792x2392	150	3	21	Intensidad	64 64	127.776	-0.804	127.876
8	PISTAS	1792x2392	150	3	21	Intensidad	64 64	128.58		128.68

Fuente: Elaboración propia



Mostramos a continuación una tabla resumida teniendo en cuenta el método, tamaño de la imagen, parámetros y el tiempo de procesamiento por cada prueba realizada.

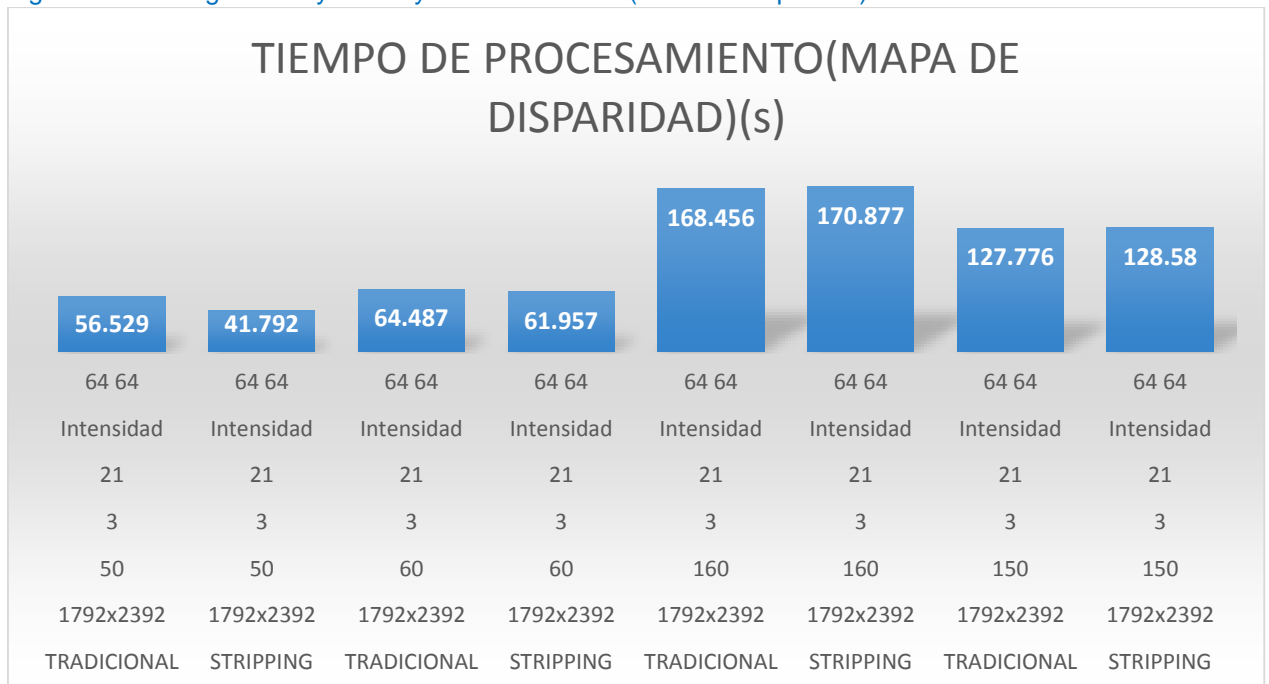
Tabla 5 Resumen de la tabla del Tiempo que tarda cada método en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles)

TECNICA	DIMENSION (pixeles)	DISPARIDAD MAXIMA	Δ MAXIMA DE LINEA	TAMAÑO DE VENTANA	TIPO DE BUSQUEDA	RESOLUCION DE MALLA	TIEMPO DE PROCESAMIENTO(MAPA DE DISPARIDAD)(s)
TRADICIONAL	1792x2392	50	3	21	Intensidad	64 64	56.529
PISTAS	1792x2392	50	3	21	Intensidad	64 64	41.792
TRADICIONAL	1792x2392	60	3	21	Intensidad	64 64	64.487
PISTAS	1792x2392	60	3	21	Intensidad	64 64	61.957
TRADICIONAL	1792x2392	160	3	21	Intensidad	64 64	168.456
PISTAS	1792x2392	160	3	21	Intensidad	64 64	170.877
TRADICIONAL	1792x2392	150	3	21	Intensidad	64 64	127.776
PISTAS	1792x2392	150	3	21	Intensidad	64 64	128.58

Fuente: Elaboración propia



Gráfico 3 Tiempo que tarda cada método en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles)



Fuente: Elaboracion propia

Mostramos a continuación el tiempo promedio que demora cada método para poder procesar el mapa de disparidad.

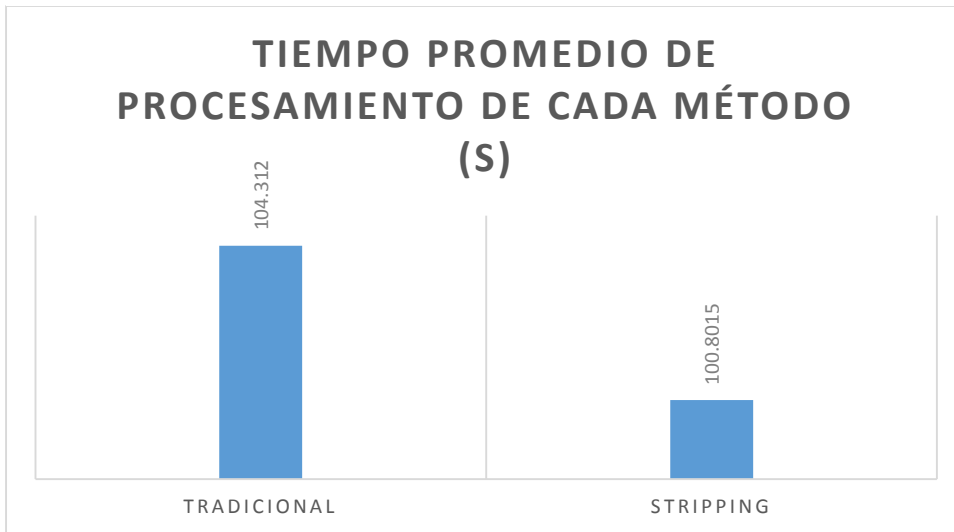
Tabla 6 Tiempo Promedio que tarda cada método en procesar el mapa de disparidad, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles).

Tiempo promedio de procesamiento de Cada Método	
TECNICA	TIEMPO DE PROCESAMIENTO(MAPA DE DISPARIDAD)(s)
TRADICIONAL	104.312
PISTAS	100.8015

Fuente: Elaboración propia



Gráfico 4 Tiempo promedio de procesamiento de cada método, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles)



Fuente: Elaboracion propia

Obteniendo como resultado que el método que procesa de manera más rápida el mapa de disparidad es el Método 2 que permite preparar el mapa de disparidad para poder reconstruirlo basándose en la Técnica de Pistas, la diferencia de tiempo es de 3.5105 segundos en comparación con el Método 1 que permite procesar el mapa de disparidad utilizando la Técnica Tradicional.

c) Tiempo que tarda cada método en realizar la reconstrucción 3d de imágenes rayos x del fruto con semilla (palta), mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles).

Este indicador es el resultado de tiempo que tarda cada método en realizar la reconstrucción 3d de las 2 imágenes procesadas. Para calcular el tiempo que tarda en reconstruir el fruto de la palta con semilla, el algoritmo obtiene primero la hora de inicio (hora actual del sistema) en milisegundos y al final de reconstruir el algoritmo capturamos la hora fin (hora actual del sistema), para



después restar y obtener el tiempo en milisegundos que demora cada método, mostrando al final el tiempo convertido a segundos.

Tiempo de reconstrucción =Tiempo Final – Tiempo Inicial

A continuación se presenta los resultados del tiempo que tarda en reconstruir cada método el fruto de la palta con semilla mediante el ingreso de 2 imágenes rayos x que se ha establecido como muestra.

A continuación se presenta los resultados del tiempo que tarda en reconstruir el fruto de la palta con semilla, teniendo la dimensión de la imagen así como los siguientes parámetros (Disparidad máxima, variación máxima de línea, tipo de búsqueda, resolución de malla).

Tabla 7 Tiempo que tarda cada método en realizar la reconstrucción 3d de imágenes rayos x del fruto con semilla (palta), mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles).

PRUEBA	TECNICA	DIMENSION (pixeles)	DISPARIDAD MAXIMA	Δ MAXIMA DE LINEA	TAMAÑO DE VENTANA	TIPO DE BUSQUEDA	RESOLUCION DE MALLA	TIEMPO DE PROCESAMIENTO (DISPARIDAD (s))	Δ de Tiempo(s)	Tiempo de Reconstrucción(s)
1	TRADICIONAL	648x864	50	3	21	Intensidad	64 64	9.991	-1.575	10.091
2	PISTAS	648x864	50	3	21	Intensidad	64 64	11.566		11.666
3	TRADICIONAL	648x864	60	3	21	Intensidad	64 64	13.611	0.126	13.711
4	PISTAS	648x864	60	3	21	Intensidad	64 64	13.485		13.585
5	TRADICIONAL	648x864	160	3	21	Intensidad	64 64	21.502	-2.421	21.602
6	PISTAS	648x864	160	3	21	Intensidad	64 64	23.923		24.023
7	TRADICIONAL	648x864	150	3	21	Intensidad	64 64	28.026	-0.804	28.126
8	PISTAS	648x864	150	3	21	Intensidad	64 64	28.83		28.93

Fuente: Elaboracion propia.



Mostramos a continuación una tabla resumida teniendo en cuenta el método, tamaño de la imagen, parámetros y el tiempo de reconstrucción por cada prueba realizada.

TECNICA	DIMENSION (pixeles)	DISPARIDAD MAXIMA	Δ MAXIMA DE LINEA	TAMAÑO DE VENTANA	TIPO DE BUSQUEDA	RESOLUCION DE MALLA	Tiempo de Reconstrucción(s)
TRADICIONAL	648x864	50	3	21	Intensidad	64 64	10.091
PISTAS	648x864	50	3	21	Intensidad	64 64	11.666
TRADICIONAL	648x864	60	3	21	Intensidad	64 64	13.711
PISTAS	648x864	60	3	21	Intensidad	64 64	13.585
TRADICIONAL	648x864	160	3	21	Intensidad	64 64	21.602
PISTAS	648x864	160	3	21	Intensidad	64 64	24.023
TRADICIONAL	648x864	150	3	21	Intensidad	64 64	28.126
PISTAS	648x864	150	3	21	Intensidad	64 64	28.93

Fuente: Elaboracion propia

Tabla 8 Resumen de la tabla del tiempo que tarda cada método en realizar la reconstrucción 3d de imágenes rayos x del fruto con semilla (palta), mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles).

Gráfico 5 Tiempo que tarda cada método en realizar la reconstrucción 3d de imágenes rayos x del fruto con semilla (palta), mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles).



Tiempo promedio de reconstrucción del futo de la palta con semilla(s)	
TECNICA	Tiempo de Reconstrucción del fruto de la palta con semilla (S)
TRADICIONAL	18.3825
PISTAS	19.551

Fuente: Elaboracion propia

Fuente: Elaboracion propia

Mostramos a continuación el tiempo promedio que demora cada método para poder reconstruir el fruto de la palta con semilla.

Tabla 9 Tiempo Promedio que tarda cada método en reconstruir el fruto de la palta con semilla, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles).

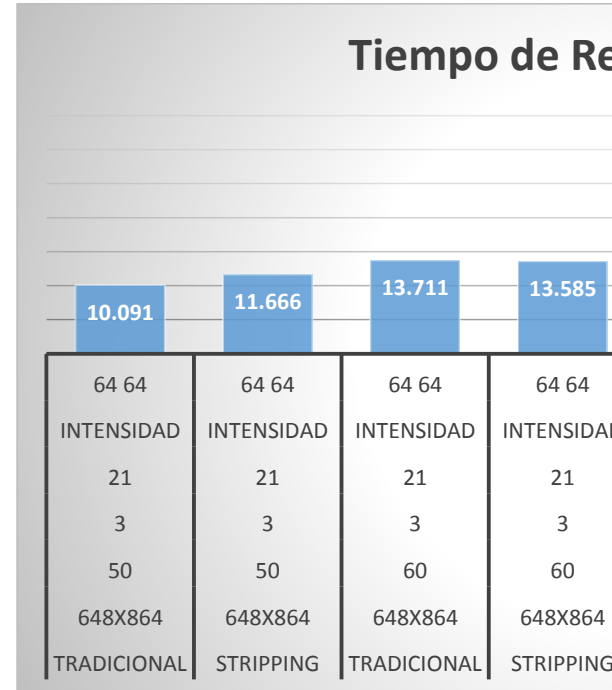
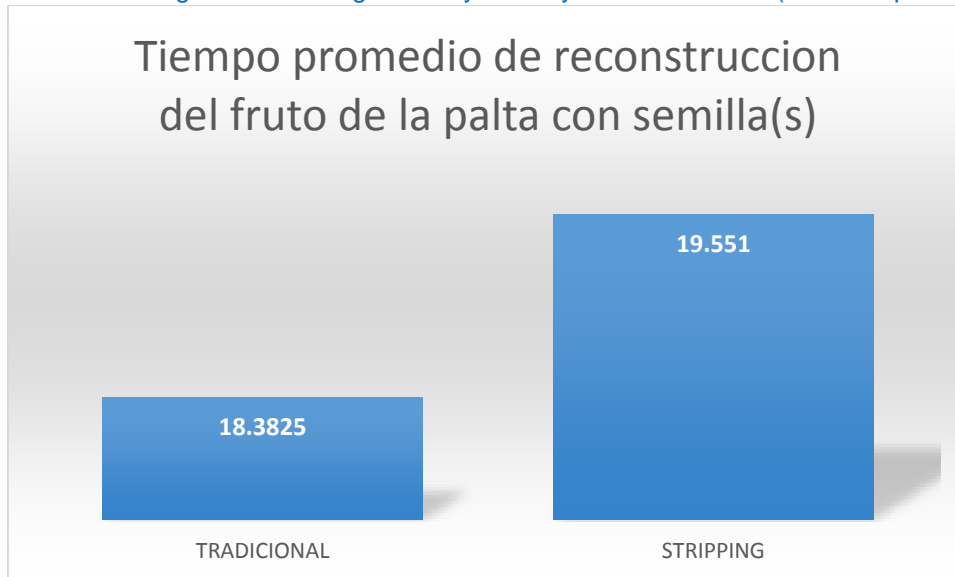


Gráfico 6 Tiempo Promedio que tarda cada método en reconstruir el fruto de la palta con semilla, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles).



Fuente: Elaboracion propia

Obteniendo como resultado que el método que reconstruye de manera más rápida el fruto de la palta con semilla es el Método 1 que permite ver como el fruto está en su tercera dimensión basándose en la Técnica Tradicional, la diferencia de tiempo es de 1.1685 segundos en comparación con el Método 2 que permite ver como el fruto está en su tercera dimensión utilizando la Técnica de Pistas.

d) Tiempo que tarda cada método en realizar la reconstrucción 3d de imágenes rayos x del fruto con semilla (palta), mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles).

Este indicador es el resultado de tiempo que tarda cada método en realizar la reconstrucción 3d de las 2 imágenes procesadas. Para calcular el tiempo que



tarda en reconstruir el fruto de la palta con semilla, el algoritmo obtiene primero la hora de inicio (hora actual del sistema) en milisegundos y al final de reconstruir el algoritmo capturamos la hora fin (hora actual del sistema), para después restar y obtener el tiempo en milisegundos que demora cada método, mostrando al final el tiempo convertido a segundos.

Tiempo de reconstrucción =Tiempo Final – Tiempo Inicial

A continuación se presenta los resultados del tiempo que tarda en reconstruir cada método el fruto de la palta con semilla mediante el ingreso de 2 imágenes rayos x que se ha establecido como muestra.

A continuación se presenta los resultados del tiempo que tarda en reconstruir el fruto de la palta con semilla, teniendo la dimensión de la imagen así como los siguientes parámetros (Disparidad máxima, variación máxima de línea, tipo de búsqueda, resolución de malla).

Tabla 10 Tiempo que tarda cada método en realizar la reconstrucción 3d de imágenes rayos x del fruto con semilla (palta), mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles).

PRUEBA	TECNICA	DIMENSION (pixeles)	DISPARIDAD MAXIMA	Δ MAXIMA DE LINEA	TAMAÑO DE VENTANA	TIPO DE BUSQUEDA	RESOLUCION DE MALLA	TIEMPO DE PROCESAMIENTO (DISPARIDAD (s))	Δ de Tiempo(s)	Tiempo de Reconstrucción(s)
1	TRADICIONAL	1792x2392	50	3	21	Intensidad	64 64	56.529	14.737	56.629
2	PISTAS	1792x2392	50	3	21	Intensidad	64 64	41.792		41.829
3	TRADICIONAL	1792x	60	3	21	Intensidad	64 64	64.487	2.53	64.587



		2392								
4	PISTAS	1792x 2392	60	3	21	Intensidad	64 64	61.957		62.057
5	TRADICIONAL	1792x 2392	160	3	21	Intensidad	64 64	168.456	-2.421	168.556
6	PISTAS	1792x 2392	160	3	21	Intensidad	64 64	170.877		170.977
7	TRADICIONAL	1792x 2392	150	3	21	Intensidad	64 64	127.776	-0.804	127.876
8	PISTAS	1792x 2392	150	3	21	Intensidad	64 64	128.58		128.68

TECNICA	DIMENSION (pixeles)	DISPARIDAD MAXIMA	Δ MAXIMA DE LINEA	TAMAÑO DE VENTANA	TIPO DE BUSQUEDA	RESOLUCION DE MALLA	Tiempo de Reconstrucción(s)
---------	---------------------	-------------------	-------------------	-------------------	------------------	---------------------	-----------------------------

Fuente: Elaboracion propia.

Mostramos a continuación una tabla resumida teniendo en cuenta el método, tamaño de la imagen, parámetros y el tiempo de reconstrucción por cada prueba realizada.

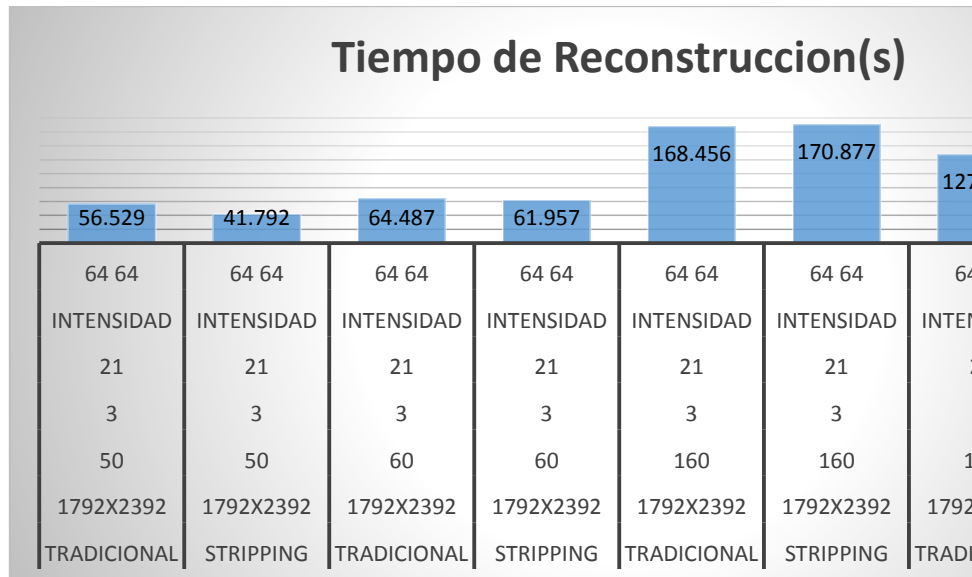
Tabla 11 Resumen de la tabla del tiempo que tarda cada método en realizar la reconstrucción 3d de imágenes rayos x del fruto con semilla (palta), mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles).



TRADICIONAL	1792x2392	50	3	21	Intensidad	64 64	56.529
PISTAS	1792x2392	50	3	21	Intensidad	64 64	41.792
TRADICIONAL	1792x2392	60	3	21	Intensidad	64 64	64.487
PISTAS	1792x2392	60	3	21	Intensidad	64 64	61.957
TRADICIONAL	1792x2392	160	3	21	Intensidad	64 64	168.456
PISTAS	1792x2392	160	3	21	Intensidad	64 64	170.877
TRADICIONAL	1792x2392	150	3	21	Intensidad	64 64	127.776
PISTAS	1792x2392	150	3	21	Intensidad	64 64	128.58

Gráfico 7 Tiempo que tarda cada método en realizar la reconstrucción 3d de imágenes rayos x del fruto con semilla (palta), mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles).

Fuente: Elaboracion propia



Fuente: Elaboracion propia

Mostramos a continuación el tiempo promedio que demora cada método para poder reconstruir el fruto de la palta con semilla.

Tabla 12 Tiempo Promedio que tarda cada método en reconstruir el fruto de la palta con semilla, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles).

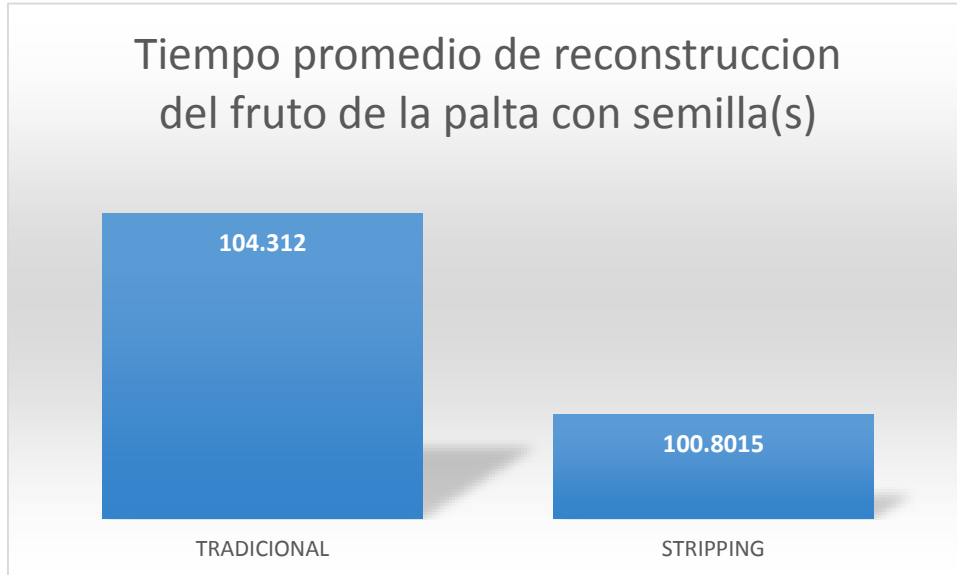
Tiempo promedio de reconstrucción del fruto de la palta con semilla(s)	
TECNICA	Tiempo de Reconstrucción del fruto de la palta con semilla (S)
TRADICIONAL	104.312



PISTAS	100.8015
--------	----------

Fuente: Elaboracion propia

Gráfico 8 Tiempo Promedio que tarda cada método en reconstruir el fruto de la palta con semilla, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles).



Fuente: Elaboracion propia

Obteniendo como resultado que el método que reconstruye de manera más rápida el fruto de la palta con semilla es el Método 2 que permite ver como el fruto está en su tercera dimensión basándose en la Técnica de Pistas, la diferencia de tiempo es de 3.5105 segundos en comparación con el Método 1 que permite ver como el fruto está en su tercera dimensión utilizando la Técnica Tradicional.

e) Consumo de memoria de cada método en procesar el mapa de disparidad, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles).

Tabla 13 Consumo de memoria de cada método en el proceso de búsqueda de correspondencias para obtener el mapa de disparidad expresado en MB, ingresando las 2 imágenes con tamaño de 648x864 pixeles

Consumo de memoria en el proceso de búsqueda de correspondencias para obtener el mapa de disparidad expresado en MB, ingresando las 2 imágenes con tamaño de 648x864 pixeles

Consumo de memoria en el proceso de búsqueda de correspondencias para obtener el mapa de disparidad



Proceso	METODO TRADICIONAL	PISTAS	Método Tradicional(MB)	PISTAS(MB)
Lectura de imágenes	8xy	8xy	4478976	4478976
Costo de correspondencia	4xyd	4x(1+j)d	58047528.96	1808610.509
Agregación de evidencia	4xy	4xd	2239488	67184.64
Mapa de disparidad	4xy	4xy	2239488	2239488
Consumo Total	16xy + 4xyd	12xy+4xd+4x(1+j)d	67005480.96	8594259.149

Fuente: Elaboracion propia

Mostramos a continuación los valores establecidos de las variables de entrada(x, y, d, j), que se utilizaran para el cálculo del consumo de memoria.

Tabla 14 Tabla de las variables de entrada(x, y, d, j), que se utilizaran para el cálculo del consumo de memoria, ingresando las 2 imágenes con tamaño de 648x864 pixeles

Resolución Imagen Palta	
x	648
y	864
d(disparidad)=(4%x)	25.92
J(altura de ventana) =(3%y)	25.92

Fuente: Elaboracion propia

Mostramos a continuación el % de consumo de memoria de cada método, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles)

Los valores establecidos de las variables de entrada(x, y, d, j), que se utilizaron para el cálculo del consumo de memoria.

Tabla 15 % de consumo de memoria de cada método, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles)

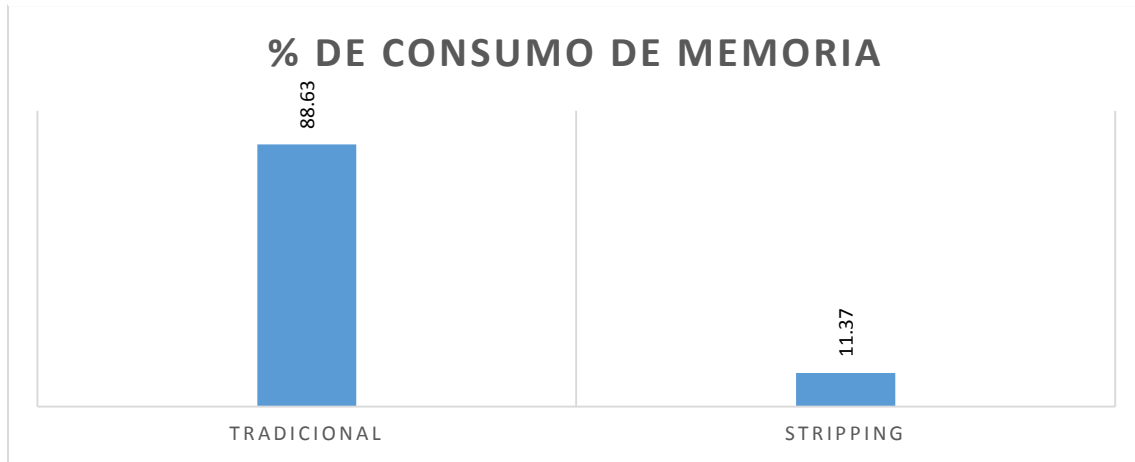
% de Consumo de memoria	
METODO	%
Tradicional	88.63
PISTAS	11.37



Total	100.00
-------	--------

Fuente: Elaboracion propia

Gráfico 9 % de consumo de memoria de cada método, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles)



Fuente: Elaboracion propia

Obteniendo como resultado que el método que consume menos memoria es el método 2 que procesa el mapa de disparidad, basándose en la Técnica de Pistas , el porcentaje de consumo de memoria es 11,37 % reduciendo un 72,26 % del Método 1 que consume el 88,63 % de memoria, basándose en la Técnica Tradicional.

f) Consumo de memoria de cada método en procesar el mapa de disparidad, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles).



Gráfico 10 Consumo de memoria de cada método en el proceso de búsqueda de correspondencias para obtener el mapa de disparidad expresado en MB, ingresando las 2 imágenes con tamaño de 1792x2392 pixeles

Consumo de memoria en el proceso de búsqueda de correspondencias para obtener el mapa de disparidad expresado en MB, ingresando las 2 imágenes con tamaño de 1792x2392 pixeles

Consumo de memoria en el proceso de búsqueda de correspondencias para obtener el mapa de disparidad				
Proceso	METODO TRADICIONAL	PISTAS	Método Tradicional(MB)	PISTAS(MB)
Lectura de imágenes	$8xy$	$8xy$	34291712	34291712
Costo de correspondencia	$4xyd$	$4x(1+j)d$	1229014958	37384250.98
Agregación de evidencia	$4xy$	$4xd$	17145856	513802.24
Mapa de disparidad	$4xy$	$4xy$	17145856	17145856
Consumo Total	$16xy + 4xyd$	$12xy+4xd+4x(1+j)d$	1297598382	89335621.22

Mostramos a continuación los valores establecidos de las variables de entrada(x, y, d, j), que se utilizaran para el cálculo del consumo de memoria.

Tabla 16 Tabla de las variables de entrada(x, y, d, j), que se utilizaran para el cálculo del consumo de memoria, ingresando las 2 imágenes con tamaño de 1792x2392 pixeles

Resolución Imagen Palta	
X	1792
Y	2392
$d=(4\%x)$	71.68
$j=(3\%y)$	71.76

Fuente: Elaboracion propia

Mostramos a continuación el % de consumo de memoria de cada método, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392pixeles). Los valores establecidos de las variables de entrada(x, y, d, j), que se utilizaron para el cálculo del consumo de memoria.

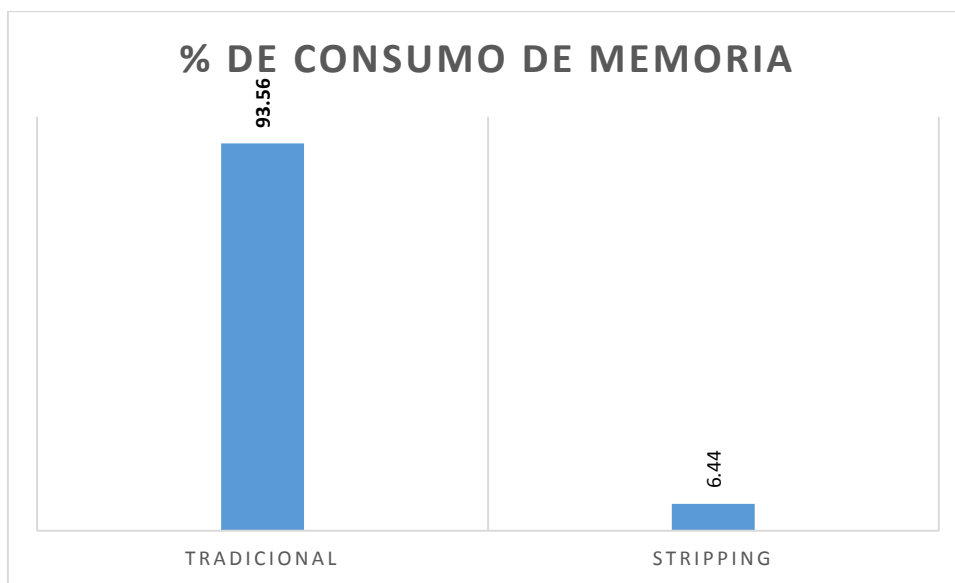


Tabla 17 % de consumo de memoria de cada método, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles)

% de Consumo de memoria	
METODO	%
Tradicional	93.56
PISTAS	6.44
Total	100.00

Fuente: Elaboracion propia

Gráfico 11 % de consumo de memoria de cada método, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles)



Fuente: Elaboracion propia

Obteniendo como resultado que el método que consume menos memoria es el método 2 que procesa el mapa de disparidad, basándose en la Técnica de Pistas , el porcentaje de consumo de memoria es 6,44 % reduciendo un 87,12 % del Método 1 que consume el 93,56 % de memoria, basándose en la Técnica Tradicional.

g) Precisión final de la reconstrucción, mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles).

Tabla 18 Precisión en centímetros cúbicos comparando el volumen del objeto real de la palta con el objeto reconstruido de la palta mediante la técnica tradicional y pistas con resolución de imagen (648x864 pixeles).



Tradicional-Pistas:Precision en la Reconstrucción del fruto(648x864)		
Descripción	Medida(pulgadas cubicas)	Medida(centímetros cúbicos)
Volumen Palta Reconstruido	21.00	344.12
Volumen Palta Real		350

Fuente: Elaboración propia

Tabla 19 % de Precisión de ambos métodos

Método	Precisión(centímetros cúbicos)+ -
Tradicional/Pistas	5.88

Fuente: Elaboración propia

Según la fórmula del rango estándar

http://www.batesville.k12.in.us/physics/apphynet/Measurement/estimating_precision.htm para un cálculo de precisión se toma el valor máximo con el valor

mínimo y se realiza la diferencia para calcular la precisión. El volumen de la palta real se ha realizado mediante el principio de Arquímedes

<http://www.sc.ehu.es/sbweb/fisica/fluidos/estatica/arquimedes/arquimedes.htm>

que afirma que: Un cuerpo total o parcialmente sumergido en un fluido en reposo, recibe un empuje de abajo hacia arriba igual al peso del volumen del fluido que desaloja. El volumen de la palta real fue de 350 mililitros esto lo convertimos a centímetros cúbicos sabiendo que 1 mililitro es igual a 1 centímetro cúbico y tenemos 350 centímetros cúbicos tal como lo muestra la tabla anterior. Por otra parte para aplicar la fórmula del rango estándar tengo que tener el volumen del objeto reconstruido mediante la aplicación, en donde me genera un archivo .vrmf la cual esto lo importado en un software 3DS Max que me permite visualizar el volumen.

En este caso el volumen me arroja como resultado 21 pulgadas cúbicas pero esto convertido a centímetros cúbicos me arroja 344,12 centímetros cúbicos.

Ya teniendo el volumen de ambos objetos en una unidad de medida aplico la fórmula del rango estándar que mide la precisión:



Precisión = Valor máximo – Valor mínimo

Precisión=350cm³ – 344cm³

Precisión=±5.88cm³

Al aplicar esta fórmula tengo como resultado + - 5.88 centímetros cúbicos la cual quiere decir que es una escala que solo es precisa dentro de un rango de +- 5.88 centímetros cúbicos del fruto real.

4.2. Discusión de Resultados

4.2.1. El método que procesa de manera más rápida el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles) es el Método 1 que permite preparar el mapa de disparidad para poder



reconstruirlo basándose en la Técnica Tradicional, obteniendo un tiempo de 18.2825 segundos, en comparación con el método 2 que permite preparar el mapa de disparidad para poder reconstruirlo basándose en la Técnica de Pistas, que tarda un tiempo de 19.451 segundos en procesar el mapa de disparidad, tardando 1.1685 segundos más que el método 1.

4.2.2. El método que procesa de manera más rápida el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles) es el Método 2 que permite preparar el mapa de disparidad para poder reconstruirlo basándose en la Técnica de Pistas, obteniendo un tiempo de 100.8015 segundos, en comparación con el método 1 que permite preparar el mapa de disparidad para poder reconstruirlo basándose en la Técnica Tradicional, que tarda un tiempo de 104.312 segundos en procesar el mapa de disparidad, tardando 3.5105 segundos más que el método 2.

4.2.3. El método que reconstruye de manera más rápida el fruto de la palta con semilla mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles) es el Método 1 que permite ver como el fruto está en su tercera dimensión basándose en la Técnica Tradicional, obteniendo un tiempo



de 18.3825 segundos, en comparación con el método 2 que permite ver como el fruto está en su tercera dimensión basándose en la Técnica de Pistas, que tarda un tiempo de 19.551 segundos en reconstruir el fruto de la palta con semilla, tardando 1.1685 segundos más que el método 1.

4.2.4. El método que reconstruye de manera más rápida el fruto de la palta con semilla mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles) es el Método 2 que permite ver como el fruto está en su tercera dimensión basándose en la Técnica de Pistas, obteniendo un tiempo de 100.815 segundos, en comparación con el método 1 que permite ver como el fruto está en su tercera dimensión basándose en la Técnica Tradicional, que tarda un tiempo de 104.312 segundos en reconstruir el fruto de la palta con semilla, tardando 3.5105 segundos más que el método 2.

4.2.5. El método que consume menos memoria en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles) es el Método 2 que permite preparar el mapa de disparidad para poder reconstruirlo basándose en la Técnica de Pistas, obteniendo un porcentaje de consumo de memoria de 11.37 %, en comparación con el método 1 que permite preparar el mapa de disparidad para poder reconstruirlo basándose en la Técnica tradicional, que consume 88.63 %, consumiendo 72.26% más de memoria que el método 2.

4.2.6. El método que consume menos memoria en procesar el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles) es el Método 2 que permite preparar el mapa de disparidad para poder reconstruirlo basándose en la Técnica de Pistas,



obteniendo un porcentaje de consumo de memoria de 6.44 %, en comparación con el método 1 que permite preparar el mapa de disparidad para poder reconstruirlo basándose en la Técnica Tradicional ,que consume 93.56 %, consumiendo 87.12% más de memoria que el método 2.

4.2.7. Según la fórmula del rango estándar he podido calcular la precisión. he tomado como referencia el volumen del objeto real de la palta y el volumen del objeto reconstruido mediante la aplicación de visión estéreo. La precisión está expresando en centímetros cúbicos y me arroja como resultado que el objeto reconstruido mediante la aplicación tiene una precisión de +- 5.88 centímetros cúbicos con respecto al objeto real. Esto demuestra que según la percepción del usuario en ver el objeto reconstruido con un mejor panorama, lo sustenta con este indicador.

V. PROPUESTA DE INVESTIGACIÓN

5.1. Propuesta De Investigación



Se ha implementado dos métodos de visión estéreo para realizar la reconstrucción 3d de imágenes rayos, cumpliendo de esta forma con el objetivo principal de esta investigación.

Para la implementación del método propuesto a continuación, se ha desarrollado una aplicación en el lenguaje de programación C++ utilizando Visual C++ y la librería OpenCv.

Se logró el objetivo general que es la reconstrucción 3d de imágenes rayos x mediante visión estéreo teniendo en cuenta los siguientes objetivos específicos.

5.1.1. Desarrollar estrategia de adquisición de imágenes de rayos X para frutos con semilla.

Esta estrategia se ha logrado realizar teniendo como referencia las investigaciones de los siguientes autores.

La investigación realizada por (Hosseinian & Arefi, 2015), “3d Reconstruction From Multi-View Medical X-Ray Images – Review And Evaluation Of Existing Methods” (Reconstrucción 3D a partir de imágenes Rayos X Multivista – Revisión y valuación de los métodos existentes), recomienda lo siguiente.

- a) Sacar al menos 2 imágenes rayos x.
- b) Las imágenes pueden ser capturados por los dispositivos radiográficos estándar comerciales.

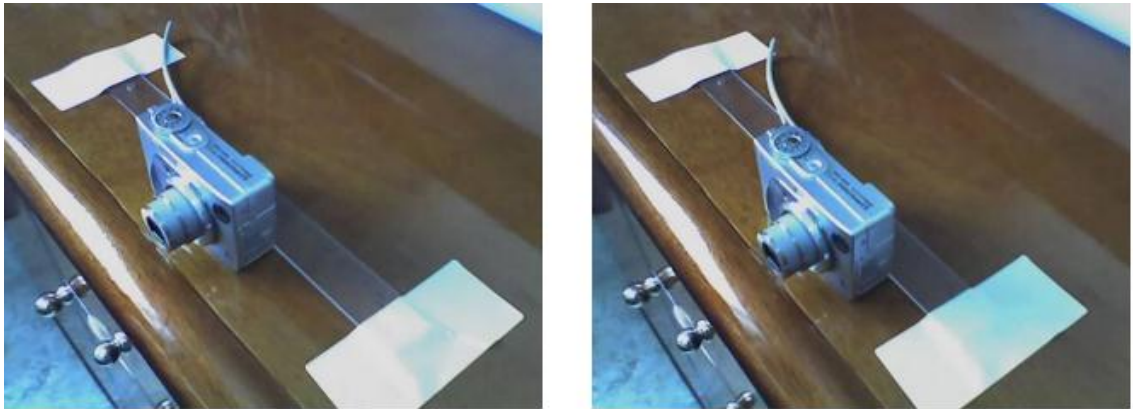


La investigación realizada por (Marcelo J. , 2008), en su trabajo de investigación “Reconstrução Tridimensional De Baixo Custo A Partir De Par De Imagens Estéreo” (reconstrucción tridimensional de bajo costo

De pares de imágenes estéreo), recomienda lo siguiente.

- a) Dos fotografías fueron tomadas con una cámara digital normal, Sony® Cyber-shot DSC-W1 de 5,1 megapíxeles en un ambiente no controlado.
- b) La cámara es apoyada sobre una mesa.
- c) El desplazamiento es de aproximadamente 60 mm.

Ilustración 21 La calibración de cámaras para obtener imágenes estéreo.



Fuente: (Marcelo J. , 2008)

Teniendo en cuenta las investigaciones explicadas anteriormente se adquirió 2 imágenes rayos x estéreo del fruto de la palta tipo dura.

- a) Equipo SGHealthCare.

Ilustración 22 Utilización del equipo SGHealthCare.

b) Distancia con 1 metro en relación a la placa.



Fuente: Elaboración propia.

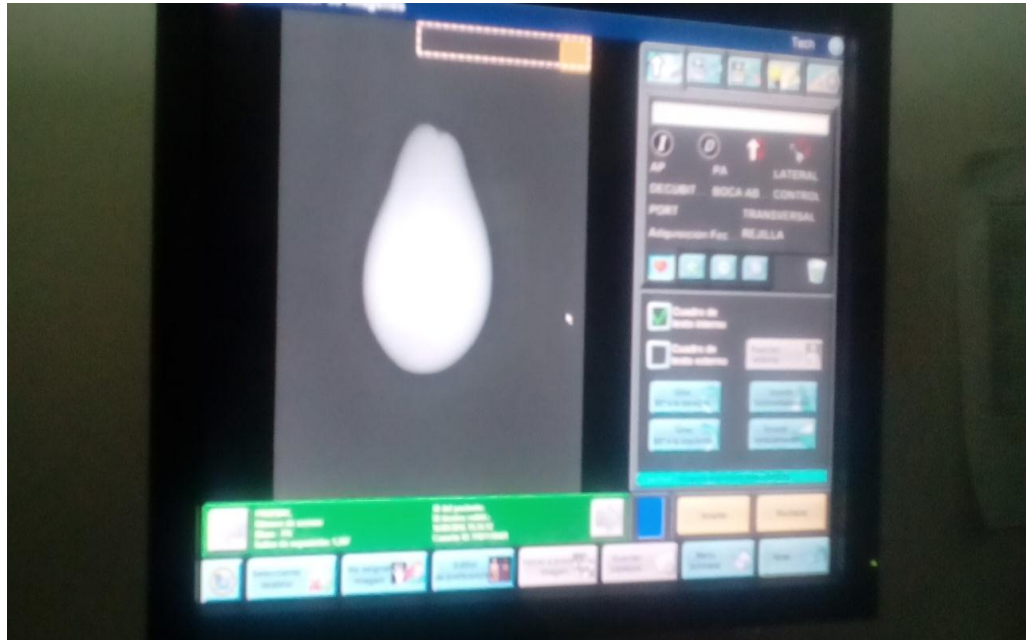
Ilustración 23 Ilustración de la distancia del equipo en relación a la placa.



Fuente: Elaboración propia.

- c) 52 KVP (Kilovoltio).
- d) 0,8 MAS (miliamperio/segundo).
- e) El desplazamiento es de aproximadamente 60 mm.

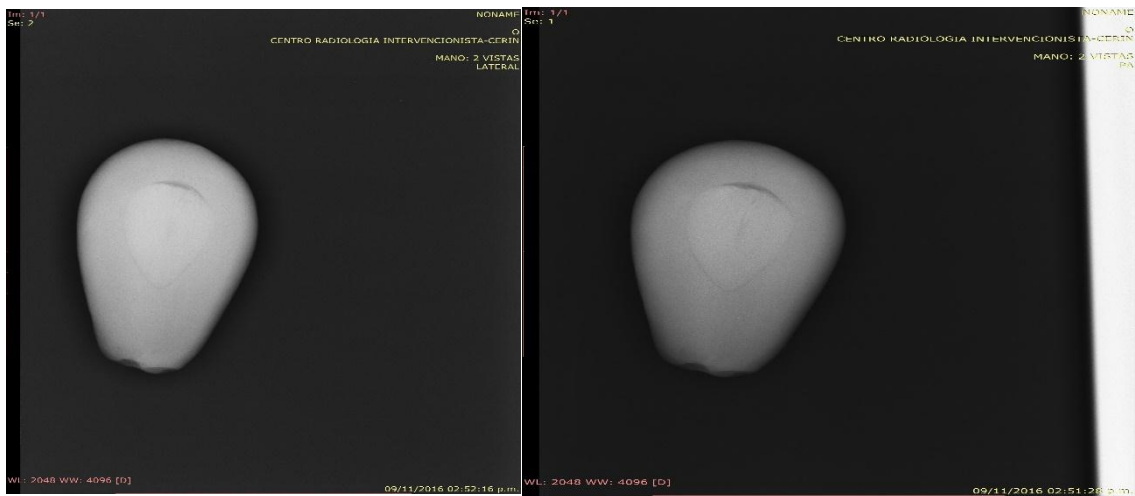
f) Tam
 año de la [Ilustración 24 Visualización de la Pantalla de rayos x.](#)
 pantalla es de 18 x 24.



Fuente: Elaboración propia.

g) Par de imágenes rayos x estéreo obtenida.
 Las imágenes adquiridas tienen una dimensión de 1792x2392 pixeles.

[Ilustración 25 Adquisición de 2 imágenes rayos x estéreo del fruto de la palta.](#)



5.1.2. Método 1: Técnica Tradicional

Se ha implementado el método de Visión Estéreo Tradicional para la reconstrucción 3d de imágenes rayos x.

Este método consiste en la lectura y carga de las imágenes rayos luego de la carga de estas se procede a la búsqueda de correspondencias de estas imágenes cargadas, la cual consiste en realizar el costo de correspondencia que realiza los cálculos matemáticos como el SSD(Suma de las diferencias al

Fuente: Elaboración propia.
 cuadrado) O SAD(Suma de diferencia absoluta) para poder cargar el DSI(Disparidad espacial de la imagen),que viene hacer el volumen (x,y,d) ,la cual x es el ancho de la imagen , y es la altura de la imagen y d es los niveles de disparidad de la imagen. Una vez que se calcula el costo de correspondencia se realiza la Agregación de evidencia la cual consiste en sumar los resultados de las disparidades calculados en el costo de correspondencia, posteriormente sigue la línea de selección de disparidad la cual busca el resultado más bajo la cual consiste en que se obtiene una mejor disparidad y así obtener el mapa de disparidad que es una representación tridimensional y luego se genera la malla en 3d.

A continuación explicaremos de manera detallada este método.

Lectura de Imágenes

Este método inicia con la carga de 2 imágenes rayos x que se encuentra almacenado en una carpeta llamada **Reléase** en el archivo **Last.txt**, establecemos el tiempo que inicia el proceso de este método para después determinar el tiempo que demora en procesar el mapa de disparidad y luego



obtener información del mapa de disparidad para reconstruir en tercera dimensión, y comparar ese tiempo con el otro método propuesto.

Se listan los 10 primeros registros de entrada del archivo last.txt. La lectura de los datos recibe como parámetros el ingreso de la imagen izquierda, imagen derecha, disparidad máxima, variación máxima de línea, tamaño de la ventana, el tipo de búsqueda que puede ser por intensidad o gradiente, y la resolución de la malla. (**anexo1**).

Para realizar la lectura y carga de las 2 imágenes rayos x del fruto de la palta con semilla presentamos el código en el lenguaje de programación C++ de la siguiente forma.

```
//Lista de los 10 primeros ítems, ordenados por los más recientes y sin repetir ítems
//Guarda los argumentos en el archivo de texto last.txt. , solo si es nuevo, en caso ya se haya
//utilizado, se convierte en el primer elemento de la lista.
int numberOfRecents=0;
int i;
bool already=false;
int itemRepetido=-1;
lastArguments last[10];
numberOfRecents=load10Recent("last.txt",&last[0]);
for(i=0;((i<numberOfRecents)&&(already==false));i++){
    if((wcscmp(GleftFile,last[i].leftFile)==0)&&(wcscmp(GrightFile,last[i].rightFile)==0)&&
        (GmaxDisp==last[i].maxDisp)&&(GmaxLineVar==last[i].maxLineVar)&&
        (GwindowSize==last[i].windowSize)&&
        (GintensityOrGradient==last[i].intensityOrGradient)&&
        (VERTS_PER_EDGE_WIDTH==last[i].VertsPerEdgeWidth)&&
        (VERTS_PER_EDGE_HEIGHT==last[i].VertsPerEdgeHeight))
        {already=true; itemRepetido=i; } }
//Luego se carga en un combobox los registros que se han realizado en el archivo Last.txt
WCHAR text[512];
//creamos una variable de tipo LPCTSTR para almacenar caracteres.
LPCTSTR pszFormat1 = TEXT("%s %s %u %u %u %c %u %u\n");
StringCbPrintfW(text,(512*sizeof(WCHAR)),pszFormat1,GleftFile,GrightFile,GmaxDisp,GmaxLine
Var,GwindowSize,GintensityOrGradient,VERTS_PER_EDGE_WIDTH,VERTS_PER_EDGE_HEIGHT);
logW("last.txt",text);
if(numberOfRecents==10)
    for(i=0;i<numberOfRecents-1;i++)
    {
        if(i!=itemRepetido)
        {
```



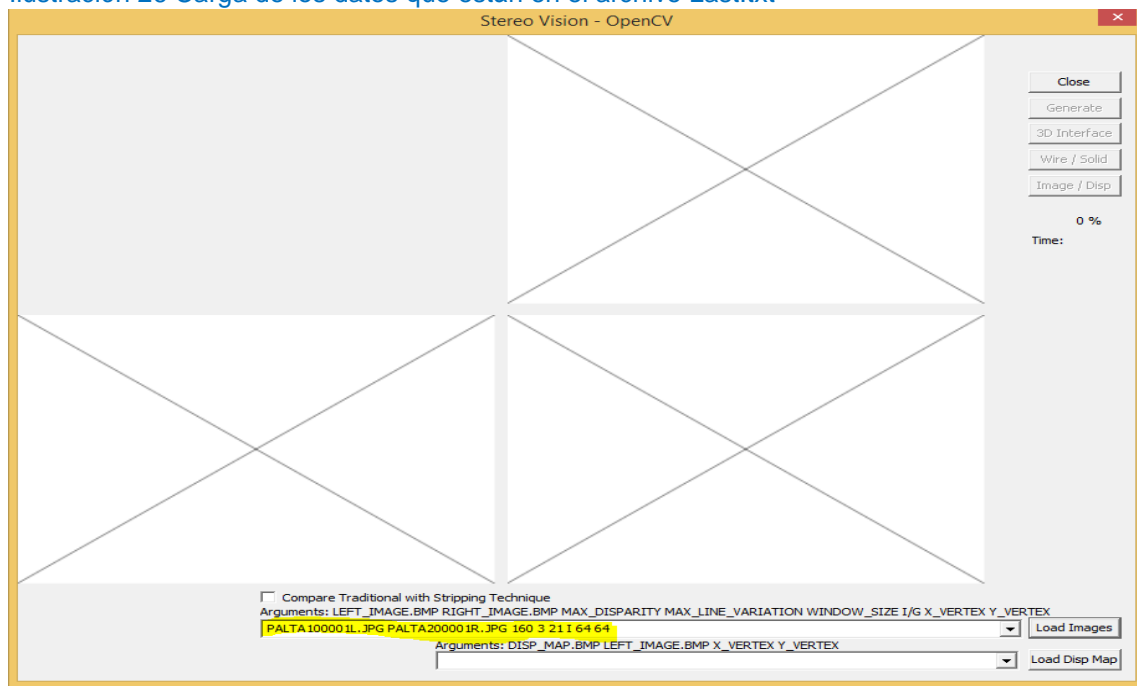

```

StringCbPrintFW(text,(512*sizeof(WCHAR)),pszFormat1,last[i].leftFile,last[i].rightFile,last[i].maxDisp,last[i].maxLineVar,last[i].windowSize,last[i].intensityOrGradient,last[i].VertsPerEdgeWidth,last[i].VertsPerEdgeHeight);
log("last.txt",text);}
}
else
for(i=0;i<numberOfRecents;i++)
{
if(i!=itemRepetido)
{
StringCbPrintFW(text,(512*sizeof(WCHAR)),pszFormat1,last[i].leftFile,last[i].rightFile,last[i].maxDisp,last[i].maxLineVar,last[i].windowSize,last[i].intensityOrGradient,last[i].VertsPerEdgeWidth,last[i].VertsPerEdgeHeight);
log("last.txt",text);
}
}
return true;

```

A continuación se muestra la lectura de los datos registrados con sus respectivos parámetros mencionados anteriormente en la ventana para su respectivo procesamiento y reconstrucción.

Ilustración 26 Carga de los datos que están en el archivo Last.txt



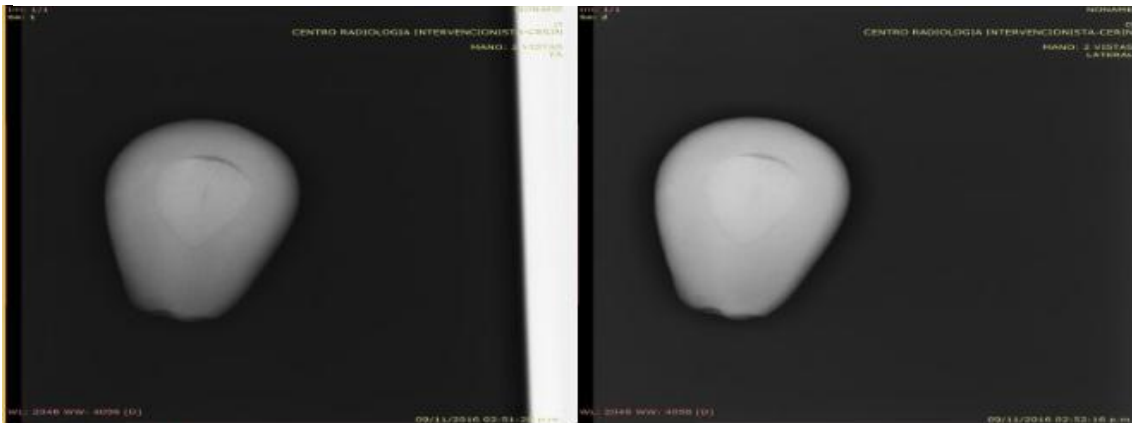
Fuente: Elaboración propia



Carga de Imágenes

Posteriormente de la lectura se procede a la carga de las imágenes según el tipo de búsqueda ya sea por intensidad o por gradiente. La búsqueda por intensidad se realiza con la imagen original sin ningún cambio, en cambio la búsqueda por gradiente las imágenes son procesadas mediante el operador gradiente.

Ilustración 27 Carga de 2 imágenes rayos x (648x864 pixeles) del fruto de la palta para la búsqueda por intensidad, mediante el Método Tradicional.



Fuente: Elaboración propia

-Por intensidad:

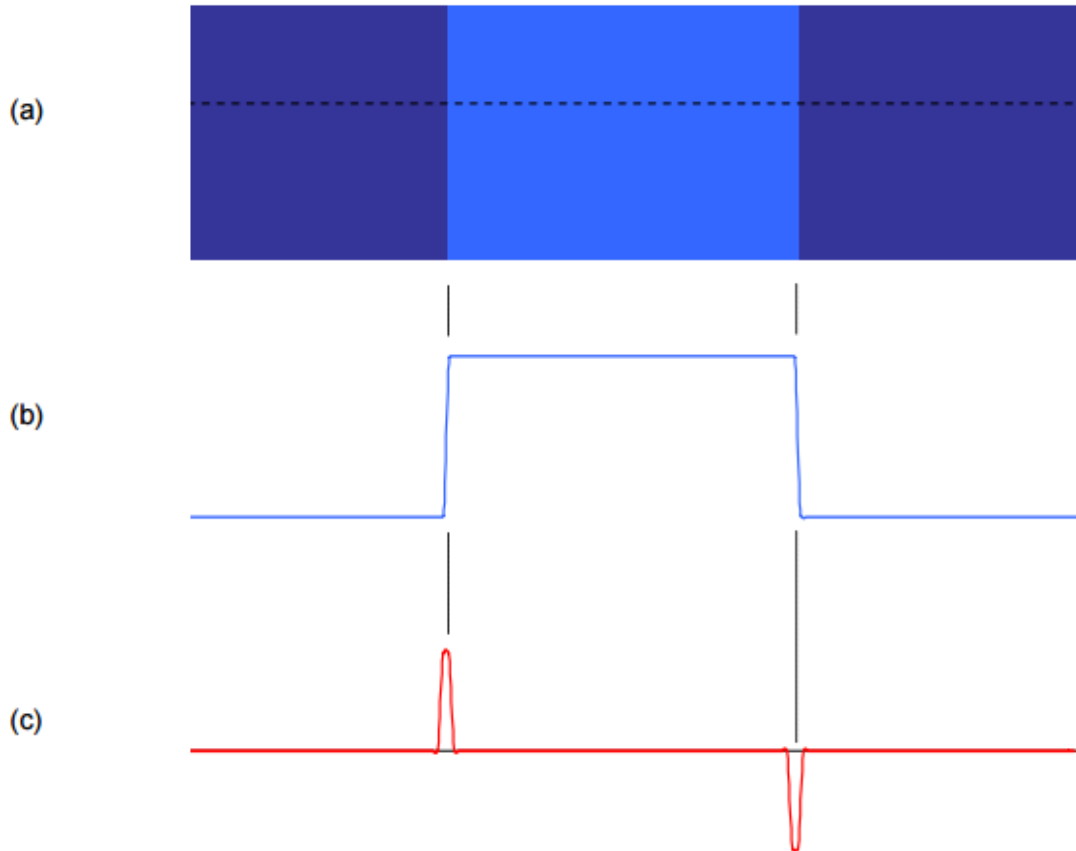
-Por gradiente:

Una manera conocida de anular la sensibilidad a las variaciones de intensidad de color es realizar un pre procesamiento de imágenes de entrada y, específicamente, la detección de bordes utilizando el operador de gradiente que según (SCHARSTEIN D. , 1999) resulta ser muy eficiente.



El operador de gradiente representa la primera derivada de una señal tal como lo muestra la ilustración 28.

Ilustración 28 (a) Imagen con banda azul claro en el centro de una región azul oscuro; (B) representación gráfica de las intensidades de color para la línea punteada en (a); (C) Primera derivada.



Para realizar esta operación en una imagen se usan máscaras conocidas como operadores Sobel (ilustración 29). El operador Sobel es utilizado en procesamiento de imágenes, especialmente en algoritmos de detección de bordes. Técnicamente es un operador diferencial discreto que calcula una aproximación al gradiente de la función de intensidad de una imagen. Para cada punto de la imagen a procesar, el resultado del operador Sobel es tanto el



vector gradiente correspondiente como la norma de éste vector. (Gonzales & Words, 2000).

Matemáticamente, el operador utiliza dos kernels de 3x3 elementos para aplicar convolución a la imagen original para calcular aproximaciones a las derivadas, un kernel para los cambios horizontales y otro para las verticales. Si definimos **A** como la imagen original, el resultado, que son las dos imágenes G_x y G_y que representan para cada punto las aproximaciones horizontal y vertical de las derivadas de intensidades, es calculado como:

Ilustración 29 Máscaras conocidas como operadores Sobel.

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * A \quad \text{y} \quad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} * A$$

En cada punto de la imagen, los resultados de las aproximaciones de los gradientes horizontal y vertical pueden ser combinados para obtener la magnitud del gradiente, mediante:

$$G = \sqrt{G_x^2 + G_y^2}$$

Con esta información, podemos calcular también la dirección del gradiente:

$$\Theta = \arctan\left(\frac{G_y}{G_x}\right)$$

donde, por ejemplo, Θ es 0 para bordes verticales con puntos más oscuros al lado izquierdo.

Se adoptó la estrategia de sólo utilizar el módulo de gradiente. Además, también hay un límite superior para los valores de operaciones en las imágenes que es 255, se adoptó un corte en valores mayores que 255.



A continuación se muestra el método propuesto para procesar la imagen aplicando el operador gradiente mediante Sobel.

```

void carregaImgBufferWithSobel(IMGpixel *pointImg, Bitmap *bitmap)
//Creando fila x columna (3x3).
    int knlV[][3]= {
                                1, 0, -1,
                                2, 0, -2,
                                1, 0, -1
                                };
    int knlH[][3]= {
                                -1, -2, -1,
                                0, 0, 0,
                                1, 2, 1
                                };

    IMGpixel *pointLoclmg;
    pointLoclmg=pointImg;

    Color z1,z2,z3;
    Color z4,z5,z6;
    Color z7,z8,z9;
    ARGB black = 0x00000000;

    BYTE alfa=255;
    BYTE red,green,blue;
    int resRed,resGreen,resBlue;
    int resRedH,resGreenH,resBlueH;
    int resRedV,resGreenV,resBlueV;
    //color resultante
    int x,y;
    for(y=0;y<(int)bitmap->GetHeight();y++)
        for(x=0;x<(int)bitmap->GetWidth();x++,pointLoclmg++)
        {
            if(((x-1)<0) || ((y-1)<0)) z1.SetValue(black);
            else bitmap->GetPixel(x-1,y-1,&z1);

            if((y-1)<0) z2.SetValue(black);
            else bitmap->GetPixel(x,y-1,&z2);

            if(((x+1)>(int)bitmap->GetWidth()) || ((y-1)<0)) z3.SetValue(black);
            else bitmap->GetPixel(x+1,y-1,&z3);

            if((x-1)<0) z4.SetValue(black);
            else bitmap->GetPixel(x-1,y,&z4);

            bitmap->GetPixel(x,y,&z5);
        }
    
```



```
if((x+1)>(int)bitmap->GetWidth()) z6.SetValue(black);
else bitmap->GetPixel(x+1,y,&z6);
```

```
if(((x-1)<0) || ((y+1)>(int)bitmap->GetHeight())) z7.SetValue(black);
else bitmap->GetPixel(x-1,y+1,&z7);
```

```
if((y+1)>(int)bitmap->GetHeight()) z8.SetValue(black);
else bitmap->GetPixel(x,y+1,&z8);
```

```
if(((x+1)>(int)bitmap->GetWidth()) || ((y+1)>(int)bitmap->GetHeight())) z9.SetValue(black);
else bitmap->GetPixel(x+1,y+1,&z9);
```

//Tratamiento para el color rojo

```
resRedH=(z1.GetR()*knIH[0][0])+(z2.GetR()*knIH[0][1])+(z3.GetR()*knIH[0][2])+
(z4.GetR()*knIH[1][0])+(z5.GetR()*knIH[1][1])+(z6.GetR()*knIH[1][2])+
(z7.GetR()*knIH[2][0])+(z8.GetR()*knIH[2][1])+(z9.GetR()*knIH[2][2]);
```

```
if(resRedH<0) resRedH=-resRedH;
if(resRedH>255) resRedH=255;
```

```
resRed=resRedH+resRedV;
```

```
if(resRed>255) resRed=255; //Usted no necesita protección para lo negativo porque siempre
//habrá números positivos.
red=(BYTE) resRed;
```

//Tratamiento para color verde

```
resGreenH=(z1.GetG()*knIH[0][0])+(z2.GetG()*knIH[0][1])+(z3.GetG()*knIH[0][2])+
(z4.GetG()*knIH[1][0])+(z5.GetG()*knIH[1][1])+(z6.GetG()*knIH[1][2])+
(z7.GetG()*knIH[2][0])+(z8.GetG()*knIH[2][1])+(z9.GetG()*knIH[2][2]);
```

```
if(resGreenH<0) resGreenH=-resGreenH;
if(resGreenH>255) resGreenH=255;
```

```
resGreen=resGreenH+resGreenV;
```

```
if(resGreen>255) resGreen=255; //Usted no necesita protección para lo negativo porque
siempre //habrá números positivos.
green=(BYTE) resGreen;
```

//Tratamiento para color azul

```
resBlueH=(z1.GetB()*knIH[0][0])+(z2.GetB()*knIH[0][1])+(z3.GetB()*knIH[0][2])+
(z4.GetB()*knIH[1][0])+(z5.GetB()*knIH[1][1])+(z6.GetB()*knIH[1][2])+
(z7.GetB()*knIH[2][0])+(z8.GetB()*knIH[2][1])+(z9.GetB()*knIH[2][2]);
```

```
if(resBlueH<0) resBlueH=-resBlueH;
if(resBlueH>255) resBlueH=255;
```

```
resBlue=resBlueH+resBlueV;
```

```
if(resBlue>255) resBlue=255; //Usted no necesita protección para lo negativo porque siempre
//habrá números positivos.
```



```
blue=(BYTE) resBlue;
```

```
//Establecer la combinación de colores después de Sobel
```

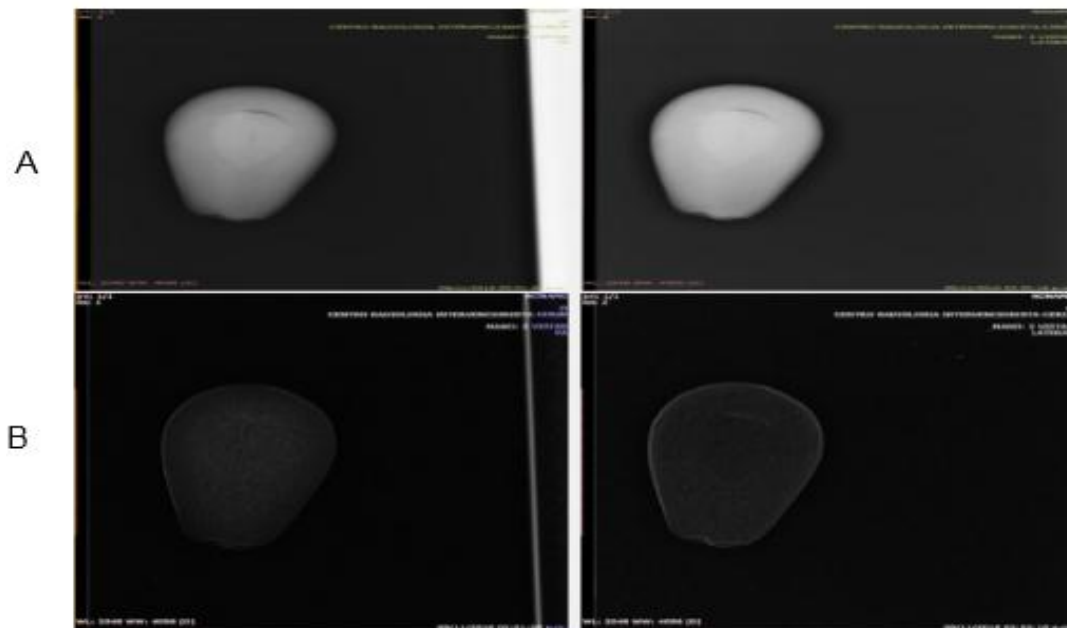
```
pointLoclmg->a=alfa;
pointLoclmg->r=red;
pointLoclmg->g=green;
pointLoclmg->b=blue;}
```

```
//Luefo de establecer la combinación con Sobel ,esto se aplica a la imagen original
```

```
pointLoclmg=pointlmg;
for(y=0;y<(int)bitmap->GetHeight();y++)
    for(x=0;x<(int)bitmap->GetWidth();x++,pointLoclmg++)
        {
            bitmap->SetPixel(x,y,Color(255,pointLoclmg->r,pointLoclmg->g,pointLoclmg->b));//Recordar que alfa debe ser 255.
        }
}
```

A continuación se muestra el resultado del método propuesto para procesar la imagen aplicando el operador gradiente mediante Sobel. El resultado visual de esta operación se muestra en la Ilustración 30(b).

Ilustración 30 (a) Par de imágenes estéreo; (B) Carga de 2 imágenes rayos x del fruto de la palta procesado con el operador gradiente, con valores absolutos y limitado a 255, mediante el Método Tradicional.



Fuente: Elaboración propia.



Búsqueda de Correspondencias

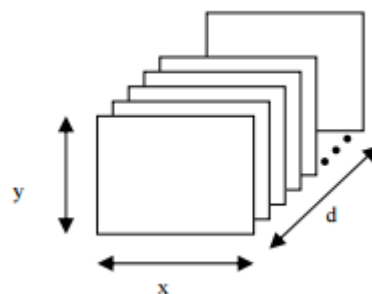
La reconstrucción 3D tiene como objetivo recuperar la información tridimensional en entornos reales. Hasta ahora lo que hemos realizado es la carga de las imágenes rayos x que puede ser por intensidad o por gradiente.

Las imágenes obtenidas son bidimensionales(x, y), con la profundidad (z) que falta.

Para obtener la profundidad, la clave es encontrar las coincidencias entre 2 imágenes para calcular la disparidad y el resultado de este es el mapa de disparidad en escala de grisis. El color claro representan objetos cercanos y los tonos oscuros representan objetos lejanos.

Antes de hacer énfasis en la búsqueda de correspondencia es esencial conocer la definición de Disparity Space Image (DSI), (BOBICK & INTILLE, 1999), que es un volumen (x, y, d), donde x es el ancho de la imagen de entrada, y es la altura y d es la cantidad de niveles de disparidad.

Ilustración 31 Disparity Space Image (DSI)



Es crucial resaltar que cada plano corresponde a la diferencia cuadrática (o absoluta), píxel a píxel entre la imagen izquierda y la derecha, desplazándose la imagen derecha de 0 (cero) a d. Para cada desplazamiento de la imagen derecha, el resultado se almacena en un plano diferente, por lo que cada plano representa una disparidad.

Formas de Medir la Correspondencia

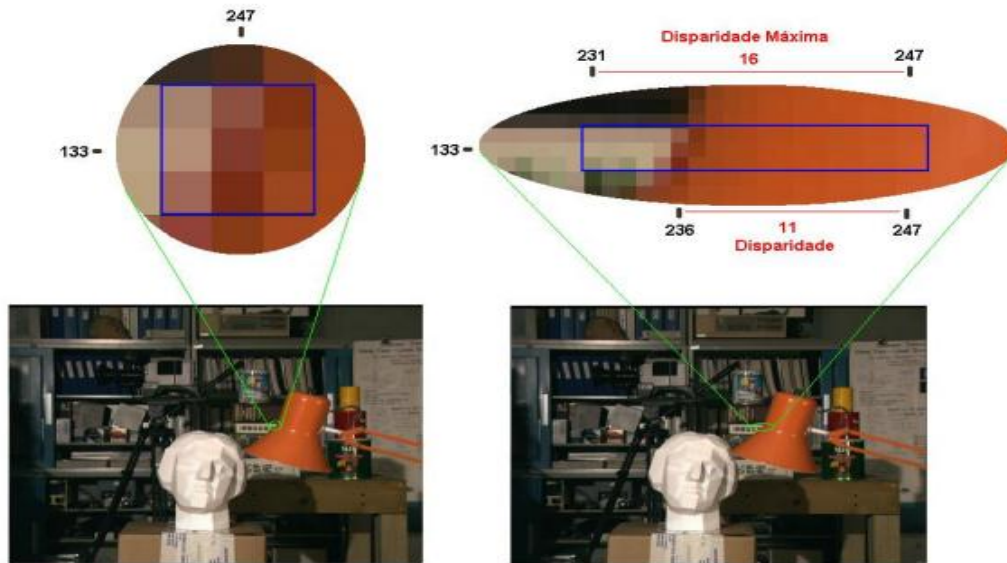
a- SAD (Suma De Las Diferencias Absolutas)

Una forma de medir la correspondencia según (HANNAH, 1974), es restar los valores de color (R: rojo, G: verde, B: azul) de un punto en la imagen izquierda de un punto en la imagen derecha y luego hacer la suma de los módulos. Este proceso se realiza sucesivamente para posibles disparidades, y el que tiene el resultado más bajo se considera la mejor correspondencia.

Comparando solo un punto a otro sería un enfoque muy simplista, lo que daría lugar a muchos errores, sin embargo, una forma de evitar estos errores es comparar no sólo un punto, sino una región cerca del punto, algo así como una ventana 3x3 5x5 o Incluso 7x7) centrado en el punto en cuestión. Una clara forma de explicación se muestra en la ilustración 32, que muestra un ejemplo que considera el uso de la restricción epipolar. Observe que la búsqueda del punto (247,133) de la imagen de la izquierda comienza exactamente en esta posición en la imagen derecha y se desplaza a la izquierda de $x = 247$ a $x = 231$.



Ilustración 32 Punto en cuestión y su ventana 3x3 en la imagen izquierda, y la línea epipolar en la imagen derecha, donde se busca la correspondencia.



Fuente: (SCHARSTEIN & SZELISKI, A Taxonomy and Evaluation of Dense TwoFrame Stereo Correspondence Algorithms., 2001)

Se realiza la diferencia de los colores de cada píxel de la ventana (en el ejemplo de la ilustración 32 es, 3x3), luego se agrega el módulo de cada resultado para que después se agregue el resultado a cada pixel y este sea almacenado; luego el proceso continua en la línea epipolar de la imagen derecha. Entre los resultados almacenados, el que presenta el valor más bajo se considerará la mejor correspondencia. Este enfoque es conocido como la suma de diferencias absolutas (SAD).

Siendo E la matriz (fija) con los valores de los colores de la ventana izquierda y D la matriz (móvil) con los valores de los colores de la ventana derecha. La distribución de estas variables se muestra en la ilustración 33.



Ilustración 33 Descripción de las variables para la ventana izquierda y derecha.

E1	E2	E3
E4	E5	E6
E7	E8	E9

D1	D2	D3
D4	D5	D6
D7	D8	D9

Para medir la disparidad se realiza los siguientes cálculos.

$$DISP1 = | E1.rojo - D1.rojo | + | E1.verde - D1.verde | + | E1.azul - D1.azul |$$

$$DISP2 = | E2.rojo - D2.rojo | + | E2.verde - D2.verde | + | E2.azul - D2.azul |$$

$$DISP3 = | E3.rojo - D3.rojo | + | E3.verde - D3.verde | + | E3.azul - D3.azul |$$

$$DISP4 = | E4.rojo - D4.rojo | + | E4.verde - D4.verde | + | E4.azul - D4.azul |$$

$$DISP5 = | E5.rojo - D5.rojo | + | E5.verde - D5.verde | + | E5.azul - D5.azul |$$

$$DISP6 = | E6.rojo - D6.rojo | + | E6.verde - D6.verde | + | E6.azul - D6.azul |$$

$$DISP7 = | E7.rojo - D7.rojo | + | E7.verde - D7.verde | + | E7.azul - D7.azul |$$

$$DISP8 = | E8.rojo - D8.rojo | + | E8.verde - D8.verde | + | E8.azul - D8.azul |$$

$$DISP9 = | E9.rojo - D9.rojo | + | E9.verde - D9.verde | + | E9.azul - D9.azul |$$

$$Disp. = Disp1 + Disp2 + Disp3 + Disp4 + Disp5 + Disp6 + Disp7 + Disp8 + Disp9$$

En es el píxel en la imagen izquierda, **Dn** píxel de la imagen de la derecha, **Disp_n** la disparidad para ese píxel y, finalmente, **Disp** como la disparidad de la ventana 3x3 en cuestión. El Disp se calcula para varias posiciones en la imagen de la derecha, estas posiciones son conocidas como los niveles de disparidad, siendo la última posición conocida como la disparidad máxima, es decir, el máximo desplazamiento horizontal que un objeto puede tener.



b- SSD (Suma De Las Diferencias Cuadradas)

Otra forma de medir la correspondencia es restar los colores de cada píxel de la ventana (en el ejemplo, 3x3), añadir el cuadrado de cada resultado (en lugar del módulo) y luego usar el mismo proceso descrito anteriormente. Este enfoque se conoce como suma de diferencia de cuadrados o SSD (suma de las diferencias cuadradas)

Para medir la disparidad se realiza los siguientes cálculos.

$$DISP1 = (E1.rojo - D1.rojo)^2 + (E1.verde - D1.verde)^2 + (E1.azul - D1.azul)^2$$

$$DISP2 = (E2.rojo - D2.rojo)^2 + (E2.verde - D2.verde)^2 + (E2.azul - D2.azul)^2$$

$$DISP3 = (E3.rojo - D3.rojo)^2 + (E3.verde - D3.verde)^2 + (E3.azul - D3.azul)^2$$

$$DISP4 = (E4.rojo - D4.rojo)^2 + (E4.verde - D4.verde)^2 + (E4.azul - D4.azul)^2$$

$$DISP5 = (E5.rojo - D5.rojo)^2 + (E5.verde - D5.verde)^2 + (E5.azul - D5.azul)^2$$

$$DISP6 = (E6.rojo - D6.rojo)^2 + (E6.verde - D6.verde)^2 + (E6.azul - D6.azul)^2$$

$$DISP7 = (E7.rojo - D7.rojo)^2 + (E7.verde - D7.verde)^2 + (E7.azul - D7.azul)^2$$

$$DISP8 = (E8.rojo - D8.rojo)^2 + (E8.verde - D8.verde)^2 + (E8.azul - D8.azul)^2$$

$$DISP9 = (E9.rojo - D9.rojo)^2 + (E9.verde - D9.verde)^2 + (E9.azul - D9.azul)^2$$

$$DISP = DISP1 + DISP2 + DISP3 + DISP4 + DISP5 + DISP6 + DISP7 + DISP8 + DISP9$$

En es el píxel de la imagen de la izquierda, **Dn** es el píxel en la imagen derecha, **Disp_n** la disparidad para ese píxel y **Disp** la disparidad de la ventana 3x3 en cuestión, una vez que se encuentra la mejor coincidencia, la disparidad se mide. Por ejemplo, si el punto de la imagen izquierda está en las coordenadas (22.30) y su correspondiente en la imagen de la derecha está en (12.30) significa que la disparidad es de 10 píxeles.



Para crear un mapa de disparidades, este proceso se realiza a través de la imagen. Una vez que se obtienen las disparidades, basta con asociar tonos de gris a cada uno, siendo que el máximo tendrá un tono más claro (objeto cercano) y el mínimo tendrá un tono más oscuro (objeto distante).

El DSI tiene un aspecto importante en la eliminación de los cálculos repetitivos, garantizando un excelente rendimiento, pero la desventaja es el alto consumo de memoria.

Se Puede codificar un método local con algunos bucles "**for**" sin almacenar resultados intermedios, pero varios cálculos se repetirán muchas veces. Supongamos que el punto de análisis del punto (22, 30) en la imagen izquierda y que funciona con una ventana 3x3. Por lo tanto, los cálculos de correspondencia (SAD o SSD o cualquier otro) se realizarán nueve veces el número máximo de disparidades esperadas en la imagen derecha, es decir, si el valor de disparidad máximo esperado es 10 píxeles, estos cálculos se realizarán 90 veces. Al pasar al siguiente píxel de la imagen izquierda (23,30), se repiten los 90 cálculos, y al menos 60 ya se han calculado directamente tal como lo muestra la ilustración 34.



Ilustración 34 Los cálculos repetidos con el desplazamiento de la ventana de la imagen

12	25	41	18	93	87	34	40	23	26	54	40	78	62	12	25	41	18	93	87	34	40	23	26	54	40	78	62
36	53	7	29	74	15	29	49	15	4	27	49	71	2	36	53	7	29	74	15	29	49	15	4	27	49	71	2
38	69	89	40	23	26	54	25	83	33	58	25	38	41	38	69	89	40	23	26	54	25	83	33	58	25	38	41
45	78	62	49	15	4	27	1	28	56	37	1	29	74	45	78	62	49	15	4	27	1	28	56	37	1	29	74
37	71	2	25	83	33	58	74	15	29	62	49	15	62	37	71	2	25	83	33	58	74	15	29	62	49	15	62
21	38	41	1	28	56	37	45	78	62	49	15	4	27	21	38	41	1	28	56	37	45	78	62	49	15	4	27
3	57	68	17	6	93	81	37	71	2	25	83	33	58	3	57	68	17	6	93	81	37	71	2	25	83	33	58
Imagen Izquierda							Imagen Derecha							Imagen Izquierda							Imagen Derecha						

$$(7 - 4)^2 + (29 - 27)^2 + (74 - 49)^2 + (89 - 33)^2 + (40 - 58)^2 + (23 - 25)^2 + (62 - 56)^2 + (49 - 37)^2 + (15 - 1)^2 = DO$$

$$(29 - 27)^2 + (74 - 49)^2 + (15 - 71)^2 + (40 - 58)^2 + (23 - 25)^2 + (26 - 38)^2 + (49 - 37)^2 + (15 - 1)^2 + (4 - 29)^2 = DO$$

Pues la forma de perfeccionar este proceso es almacenar resultados parciales de los cálculos y reutilizarlos en las diversas etapas de la búsqueda de correspondencia. En consecuencia, hay un mayor uso de la memoria, pero con una ganancia bastante significativa en el rendimiento en tiempo de ejecución.

Según (SCHARSTEIN D. , 1999) , (SCHARSTEIN & SZELISKI, A Taxonomy and Evaluation of Dense TwoFrame Stereo Correspondence Algorithms., 2001) la secuencia de procesamiento es la siguiente.

- a- Costo de correspondencia:** cálculo de la diferencia cuadrática (u otra), es en esta etapa que se crea y carga la DSI;
- b- Agregación de pruebas:** suma de todas las diferencias para una ventana determinada;
- c- Selección de disparidades:** elegir el mejor resultado.



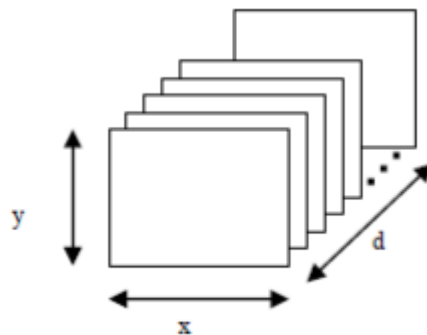
Nota: Para ilustrar este enfoque en un ejemplo de uso, consideramos una situación en la que el valor de disparidad máximo (d) es de 10 píxeles y el par de imágenes en cuestión tiene una resolución $x = 320$ e $y = 200$ cada uno.

La búsqueda de correspondencia comprende los siguientes pasos:

a) Costo de Correspondencia

El DSI (Disparity Space Image) se crea con 10 planos con 320×200 posiciones, creando un espacio x, y, d (Ilustración 35) y se realiza el proceso de cálculo de correspondencia (SAD, SSD o cualquier otro) para cada pixel (sin añadir a la ventana) para cada disparidad, es decir, compara el pixel (60.50) de la imagen izquierda con el pixel (60.50) de la imagen de la derecha y almacenar el resultado en la posición (60.50) de la imagen del Plano 0 (equivalente a disparidad 0). A continuación, se compara el pixel (60.50) de la imagen izquierda con el pixel (59.50) de la imagen derecha y el resultado se almacena en la posición (60.50) del plano 1 (igual a la disparidad 1), y así sucesivamente. Este paso se produce en ambos métodos locales y globales. Este proceso se conoce como "costo de correspondencia (matching cost)".

Ilustración 35 Planos que almacenan los costos de correspondencia.



Los valores de costo de correspondencias sobre todos los píxeles y todas las disparidades forman la imagen de espacio de disparidad inicial $M_0(x, y, d)$.

El DSI inicial puede incorporar fácilmente información de más de dos imágenes simplemente resumiendo los valores de coste para cada imagen correspondiente m , ya que el DSI está asociado con una imagen de referencia fija. Ver ecuación 1:

Las coordenadas (x, y) del espacio de disparidad se toman para coincidir con las coordenadas de píxeles de una imagen de referencia elegida de nuestro conjunto de datos de entrada. La correspondencia entre un píxel (x, y) en la imagen de referencia r y un píxel (x', y') en la imagen correspondiente m es entonces dado por.

$$x' = x + (r - m)d(x, y), \quad y' = y. \quad (1)$$

Una vez que el espacio de disparidad ha sido especificado, se realizara su respectiva carga del DSI. En general, DSI es cualquier imagen o función definida sobre una versión continua del espacio de disparidad (x, y, d) . En la práctica, el DSI suele representar la confianza o probabilidad de logaritmos (es decir, el coste) de una coincidencia particular implicada por $d(x, y)$.



b- Agregación de pruebas: Luego de crear y cargar el DSI (Disparidad espacial de imagen) ya sea por la fórmula de la Diferencia cuadrática, absoluta (u otra), se suma todas las diferencias para una ventana determinada.

$$DISP = DISP1 + DISP2 + DISP3 + DISP4 + DISP5 + DISP6 + DISP7 + DISP8 + DISP9$$

Siguiendo el ejemplo descrito en el costo de correspondencia (matching cost), se asignan otros 10 planos con 320x200 posiciones donde se registrarán los resultados de las sumas de las ventanas. Esta es la suma (considerando una ventana cuadrada de 3x3) de los nueve resultados de cada costo de Correspondencia en la posición central de la ventana de este nuevo plano. Por ejemplo, para un plano de costo de correspondencia, en la posición (60,70) agregamos los resultados que están en las posiciones indicadas en la ilustración 36.

Ilustración 36 Posiciones involucradas durante el proceso de agregación para el punto (60,70).

(59,69)	(60,69)	(61,69)
(59,70)	(60,70)	(61,70)
(59,71)	(60,71)	(61,71)

El resultado de esta suma se almacena en la posición (60, 70) del plano 3, recién asignado. La agregación de evidencia bidimensional se ha implementado utilizando ventanas cuadradas.



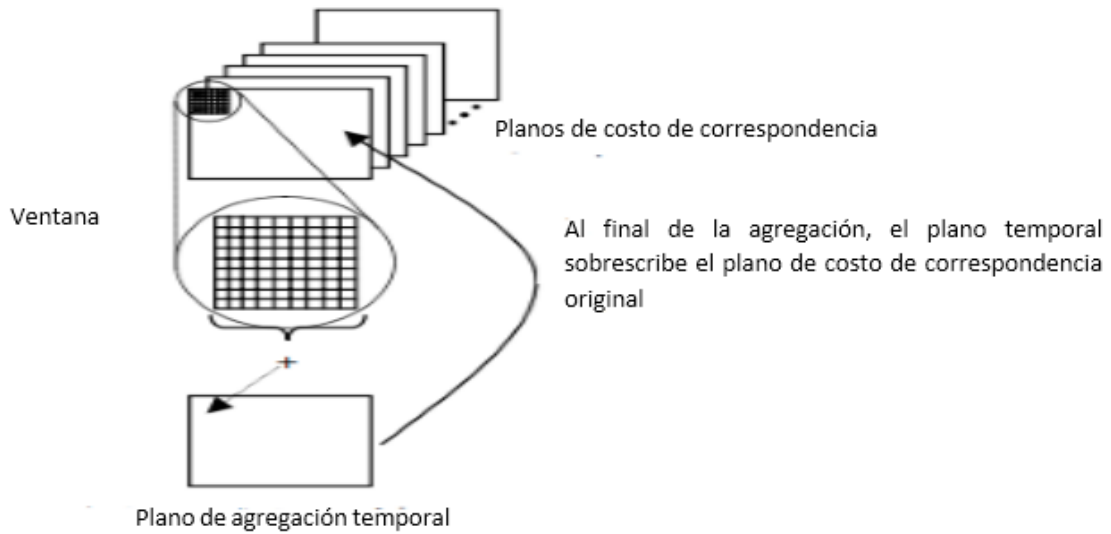
Todo lo explicado en este apartado se conoce como agregación de evidencia (en esta descripción, trabajamos con ventana cuadrada).

Dentro del bucle "for", donde se realiza el recorrido, es habitual realizar la suma en 3 columnas separadas. En este ejemplo, las columnas son 59, 60 y 61; Cuando el bucle "for" va de la posición 60,70 a la posición 61,70 se resta de la columna 59 y se añade a la columna 62. Esta técnica se utiliza ampliamente para mejorar el rendimiento y se conoce como filtro de caja (MCDONNELL, 1981). Originalmente desarrollado para su uso en filtros de imagen, de donde el nombre apareció, más tarde, se utilizó en la visión estéreo en el proceso de agregación.

En lugar de asignar más de 10 planos, sólo puede asignar un plano que almacene temporalmente el resultado de la agregación de pruebas. Cuando termina el proceso de llenando del plano, se sobrescribe el plano del costo de correspondencia original. Tal como se muestra en la ilustración 37.



Ilustración 37 Agregación de evidencias usando plano temporal, mediante el método 1: Técnica Tradicional.



c- Selección de disparidades: Elegir el mejor resultado. Finalmente, se crea otro plano con 320x200. En ella se almacena, en cada posición, el valor de la disparidad que obtuvo la mejor correspondencia, es decir, el valor más bajo. Por ejemplo, si para el punto (40.50) tenemos en los planos de agregación de evidencia los resultados de la ilustración 38, el resultado más bajo se encuentra en el plano 6, lo que significa que la mejor correspondencia se encontró con una disparidad de 6 píxeles.

Ilustración 38 Simulación de valores en los distintos planos para la posición (40, 50).

Plano	0	1	2	3	4	5	6	7	8	9
Valor	101	232	184	609	900	143	7	240	78	93

Entonces, el valor (40.50) se almacena en el valor 6. Este proceso se realiza para los 320x200 puntos y se conoce como selección de disparidades. A partir de este plano, se crea una imagen en escalas de escala de grises, para



obtener el mapa de disparidad que viene hacer la representación tridimensional de la profundidad, donde para el ejemplo con una disparidad máxima de 10 píxeles el 0 sería (0, 0,0) (negro en RGB) y 9 sería (255, 255,255) (blancas RGB) y otros valores de niveles intermedios uniformemente distribuidos.

Elegir el número más bajo como la mejor correspondencia se conoce como "el ganador se lleva todo" (WTA-Winner-takes-all). La desventaja de este enfoque es que no garantiza la unicidad de la correspondencia, es decir, un punto de la imagen de la derecha se puede considerar una buena correspondencia para varios puntos a la izquierda.

Todo lo expuesto con respecto a la búsqueda de correspondencias (Costo de correspondencia, Agregación de pruebas y Selección de disparidades), para obtener el mapa de disparidad, se ha implementado en el lenguaje de programación C++ de la siguiente forma.

//creamos el método de costo de correspondencia (matchingCost), mediante el método 1:

//Técnica Tradicional

```
void matchingCost(HDC hdc,unsigned int width,unsigned int height,unsigned int range)
```

```
{
```

```
    void *voidSquaredDiff;
    unsigned long layerProdImgSize,yProdWidth,increment;
    unsigned int x,y,layer;
    unsigned long ImgSize;
    float tempL,tempR,tempDiff,Sdiff;
    IMGpixel *pointLoclmgL,*pointLoclmgR;
    float *pointLocSquaredDiff;
```

```
    if(range==0) range=1;// El rango = 0 no es válido.
```

```
    ImgSize=width*height;
```

```
    voidSquaredDiff= new float[ImgSize*range];
```

```
    pointLocSquaredDiff=(float *)voidSquaredDiff;
```

```
    pointLoclmgR=(IMGpixel *)voidlmgR;
```

```
    startT=clock();//Sólo cuenta la ejecución del código de búsqueda de coincidencias.
```

```
    for(layer=0;layer<range;layer++)
```

```
    {
```

```
        layerProdImgSize=layer*ImgSize; //Calcula una sola vez para cada capa.
```

```
        pointLoclmgL=(IMGpixel *)voidlmgL;
```

```

for(y=0;y<height;y++)
{
yProdWidth=y*width;// Calcular una sola vez para cada y
for(x=0;x<width;x++,pointLocImgL++)
{
if(x>=(range-1)) // Si el rango es 10 la capa va de 0 a 9, por lo que el mínimo que
puedo //aceptar de la capa es 9 que es menor que uno, por lo que la sustracción de x
no sube //una línea (para y = 0 existe un riesgo Para acceder a la memoria no
asignada)
{
increment=yProdWidth+(x-layer); // Calcular una sola vez para cada x
tempL=(float)pointLocImgL->r; // Yo uso puntero puro aquí, por ser más rápido
tempR=(float)pointLocImgR[increment].r;
tempDiff=tempL-tempR;
Sdiff=tempDiff*tempDiff; //SSD
tempL=(float)pointLocImgL->g;
tempR=(float)pointLocImgR[increment].g;
tempDiff=tempL-tempR;
Sdiff=(tempDiff*tempDiff)+Sdiff;//SSD
tempL=(float)pointLocImgL->b;
tempR=(float)pointLocImgR[increment].b;
tempDiff=tempL-tempR;
Sdiff=(tempDiff*tempDiff)+Sdiff; //SSD
pointLocSquaredDiff[layerProdImgSize+yProdWidth+x]=Sdiff;
}
else pointLocSquaredDiff[layerProdImgSize+yProdWidth+x]=0;
}
}
}
// Llama al método de Agregación de pruebas
aggregation(width,height,range>windowSizeForCompare,&pointLocSquaredDiff[0]);
// llama al método Selección de disparidad
selection(hdc,width,height,range,&pointLocSquaredDiff[0]);
delete(voidSquaredDiff);
}

// Método de Agregación de pruebas
void aggregation(unsigned long width,unsigned long height,unsigned int range,unsigned long
Kernel,float *squaredDiff)
{
    unsigned int layer;
    unsigned long layerProdImgSize;
    unsigned long ImgSize;

    ImgSize=width*height;

    for(layer=0;layer<range;layer++)
    {
        layerProdImgSize=layer*ImgSize;// Calcula una sola vez para cada capa.
    }
}

```




```

        aggregationOneLayer(width,height,range,Kernel,&squaredDiff[layerProdImgSize]);
    }
}

//Método de Selección de disparidad-WTA (Winner Take-All)
void selection(HDC hdc,unsigned long width,unsigned long height,unsigned int range,float
*squaredDiff)
{
    //WCHAR text[250];
    unsigned long yProdWidth;
    unsigned int x,y,layer;
    unsigned int disp;
    float SSDtemp;//SSD Temporal
    //Diferencia entre la técnica Tradicional y la de pistas (parte x)
    //x (dispMap *pointLocDispMap)
    unsigned long *pointLocDispMap;
    unsigned int part,color;
    unsigned long ImgSize;

    ImgSize=width*height;
    part=255/range;
    //Diferencia entre la técnica Tradicional y la de pistas(dispMap *pointLocDispMap)(parte x)
    //x voidDispMapLegacy= new dispMap[width*height]; //Se elimina sólo al final del programa.
    //x pointLocDispMap=(dispMap *)voidDispMapLegacy;

    voidDispMapLegacy= new unsigned long[width*height]; //Se elimina sólo al final del programa.
    pointLocDispMap=(unsigned long *)voidDispMapLegacy;

    for(y=0;y<height;y++)
    {
        yProdWidth=y*width; // Calcular una sola vez para cada y
        for(x=0;x<width;x++)
        {
            disp=0;// Supongamos que la capa 0 tiene la suma más baja de SSD.
            SSDtemp=squaredDiff[yProdWidth+x];
            for(layer=1;layer<range;layer++)// Como el rango tiene que ser al menos 1, la
            //capa puede comenzar desde 1, ya que tengo asumido que 0 es la menor disparidad.
            {
                if(SSDtemp>squaredDiff[(layer*ImgSize)+yProdWidth+x])
                {
                    disp=layer;
                    SSDtemp=squaredDiff[(layer*ImgSize)+yProdWidth+x];
                }
            }
            color=part*disp;
            pointLocDispMap[yProdWidth+x]=color;//disp;
            // se usa color y no puntero pointLocDispMap [] porque para acceder a su contenido
            //es necesario multiplicar, por lo que es más eficiente usar una variable ya como valor de
            color.
        }
    }
}

```



```

        }//for(x=0

    }//for(y=0)

finishT=clock(); //Sólo cuenta la ejecución del código, eliminando el paso para guardar el
archivo.
durationT=finishT-startT;

```

```

Bitmap DispMap(GleftFile);//left;//i
unsigned int xs,ys;
unsigned long ysProdWidth;
pointLocDispMap=(unsigned long *)voidDispMapLegacy;

for (ys=0; ys<height;ys++)
{
    ysProdWidth=ys*width;
    for (xs=0; xs<width;xs++)
    {
        color=pointLocDispMap[(ys*width)+xs];
        DispMap.SetPixel(xs, ys, Color(255,color,color,color)); // Recuerde que el alfa DEBE ser
        // 255.
    }
}

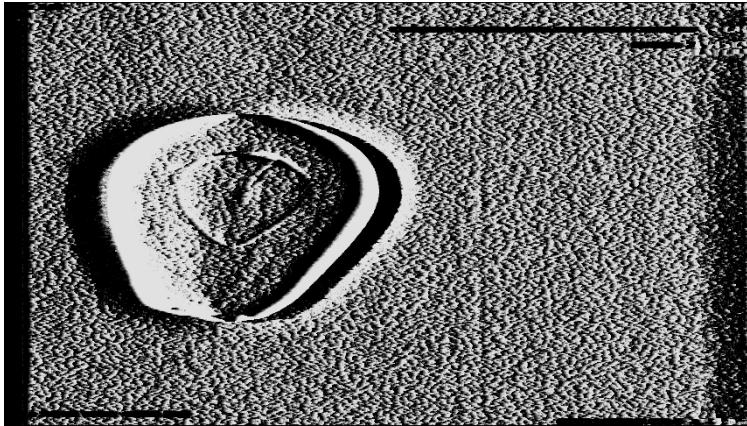
CLSID clsid;
GetEncoderClsid(L"image/bmp", &clsid);
Graphics graphics(hdc);
DispMap.Save(disparityMap, &clsid, NULL);//se guarda el mapa de disparidad
}

```

Al final tenemos el mapa de disparidad (En pantalla como en mapa de bits) aplicando el método 1: Técnica Tradicional que se almacena con una extensión .bmp en escala de grises.



Ilustración 39 El resultado del mapa de disparidad aplicando el método 1: Técnica tradicional por búsqueda de intensidad.



Fuente: Elaboración propia

Después de obtener el mapa de disparidad (base para generar la reconstrucción 3d) se genera una malla 3d.

Generación de la malla 3d

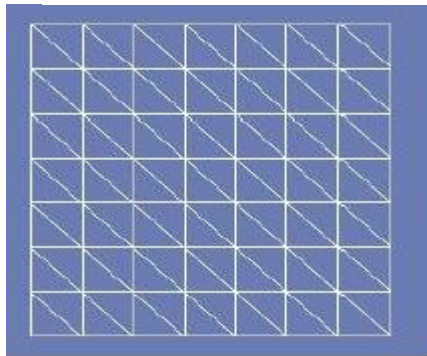
El mapa de disparidad es una manera de presentar la información tridimensional, donde se tiene la disposición espacial (x, y) y la información de la disparidades (profundidad inversa) presenta en forma de tonos de gris.

El objetivo aquí es crear una malla tridimensional abierta con la textura aplicada en ella, que permite tanto la vista del medio ambiente, como la generación de los archivos con esta información.

Para generar la superficie 3D abierta, la base es el mapa de disparidad. La enfoque adoptado aquí, era para construir una malla de vértices uniformes como se muestra en la ilustración 40.



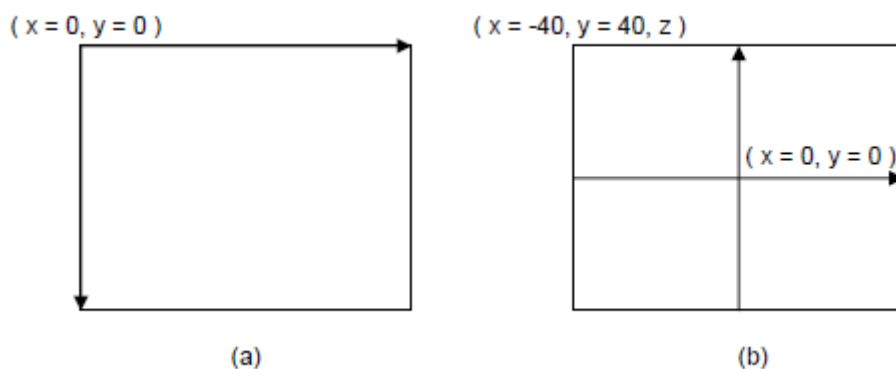
Ilustración 40 malla inicial 3D



Fuente: (Marcelo J. , 2008)

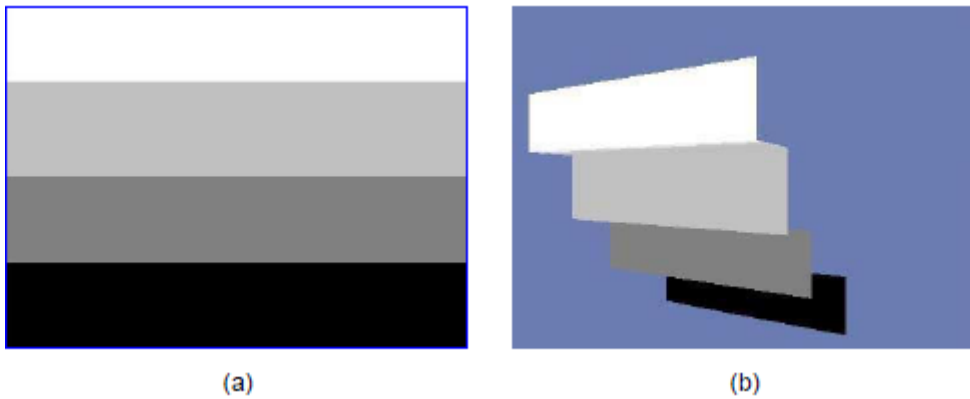
Se crea una relación directa entre los vértices de esta malla con los píxeles del mapa de disparidad. Por ejemplo, para una amplia disparidad de 80 de ancho y 80 de altura y una malla de 8 x 8 vértices superpuestos en el mapa de bits, hay un vértice cada $80/8 = 10$ píxeles horizontalmente y un vértice cada $80/8 = 10$ píxeles verticales. Al igual que en el mapa de bits del origen de coordenadas es la esquina superior izquierda y malla 3D el origen está en el centro, se adoptó la posición $(0, 0)$ del mapa de bits es equivalente a $(-40, 40, z)$ ejemplo de malla (la Figura 41).

Ilustración 41 - (a) origen de coordenadas en el mapa de bits en la esquina superior izquierda; (B) origen de coordenadas en el centro.



Para cada vértice de la malla se obtiene el color del píxel similar a su posición, su valor se resta de 255 generando el valor de z. Para cada tono de gris hay un nivel de profundidad, utilizando una imagen con tonos de escala de grises como en la ilustración 42a, demostrando el posicionamiento de cada pista de la escala de una profundidad produciendo un efecto visual de la ilustración 42b.

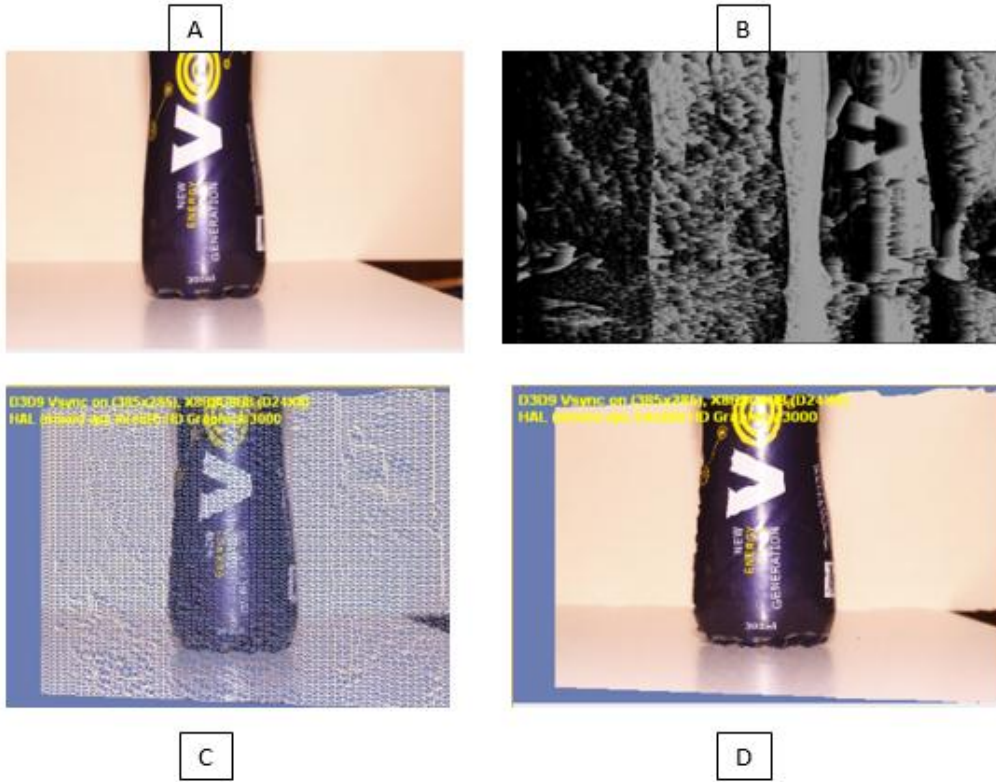
Ilustración 42 - (a) un seguimiento con tonos de gris de partida desde el blanco al negro; (b) los resultados generados con la técnica de reconstrucción 3D se describe aquí.



Fuente: (Marcelo J. , 2008)

Dentro del mismo proceso de definir los vértices también se lleva a cabo el mapeo de texturas. Así mismo se origina a partir de un mapa de bits con dimensiones iguales al mapa de disparidades, siguieron la misma distribución de píxeles en la malla 3D. En esta investigación se realizó prueba con diferentes objetos así como la palta. En el ejemplo de la ilustración 43 se percibe el resultado en su tercera dimensión de la bebida VOLT, en forma sólida y en malla.

Ilustración 43 (a) Imagen izquierda estéreo;(b) mapa de disparidades ideal utilizado; (c) presentará da geometría 3D reconstruida a partir do mapa de disparidades en modo wireframe; (d) Reconstrucción 3D en modo sólido.



Fuente: Elaboración propia



Todo lo expuesto anteriormente se ha implementado en el lenguaje de Programación C++ de la siguiente forma, en donde lo genera un archivo con extensión .vrml que tiene como referencia los parámetros de mapa de disparidad y el tipo de resolución de la malla 3d.

//La generación del modelado 3d se realiza con la siguiente función, mediante el método //tradicional.

```
int InitD3D9(HINSTANCE hInstance) {
    posicaoInicial();

    #if defined(DEBUG) | defined(_DEBUG)
        _CrtSetDbgFlag(_CRTDBG_ALLOC_MEM_DF |
            _CRTDBG_LEAK_CHECK_DF );
    #endif

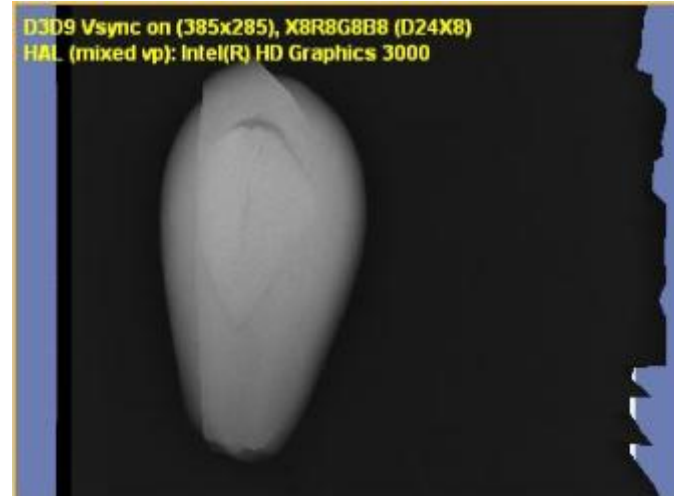
    DXUTSetCallbackD3D9DeviceAcceptable( IsDeviceAcceptable );
    DXUTSetCallbackD3D9DeviceCreated( OnCreateDevice );
    DXUTSetCallbackD3D9DeviceReset( OnResetDevice );
    DXUTSetCallbackD3D9DeviceLost( OnLostDevice );
    DXUTSetCallbackD3D9DeviceDestroyed( OnDestroyDevice );
    DXUTSetCallbackMsgProc( MsgProc );
    DXUTSetCallbackKeyboard( KeyboardProc );
    DXUTSetCallbackMouse( MouseProc );
    DXUTSetCallbackD3D9FrameRender( OnFrameRender );
    DXUTSetCallbackFrameMove( OnFrameMove );
    DXUTSetCallbackDeviceChanging( ModifyDeviceSettings );
    DXUTSetCursorSettings( true, true );
```




```
InitApp();  
  
DXUTInit( true, true );  
  
DXUTCreateWindow( L"Stereo Vision");  
  
HWND d3dHWND = DXUTGetHWNDDeviceWindowed();  
  
d3dHWNDsave=d3dHWND;  
  
LONG style = GetWindowLong( d3dHWND, GWL_STYLE );  
  
style &= ~(WS_OVERLAPPEDWINDOW);  
  
SetWindowLong( d3dHWND, GWL_STYLE, style );  
  
LONG styleEx = GetWindowLong( d3dHWND, GWL_EXSTYLE );  
  
styleEx |= WS_EX_ACCEPTFILES;  
  
SetWindowLong( d3dHWND, GWL_EXSTYLE, styleEx );  
  
SetFocus( d3dHWND );  
  
DXUTCreateDevice( true, width, height );  
  
DXUTMainLoop();  
  
return DXUTGetExitCode();  
  
}
```

Como resultado de la método 1: Técnica tradicional obtenemos el fruto de la palta en su tercera dimensión.

Ilustración 44 Resultado del algoritmo de reconstrucción mediante el método 1: Técnica tradicional

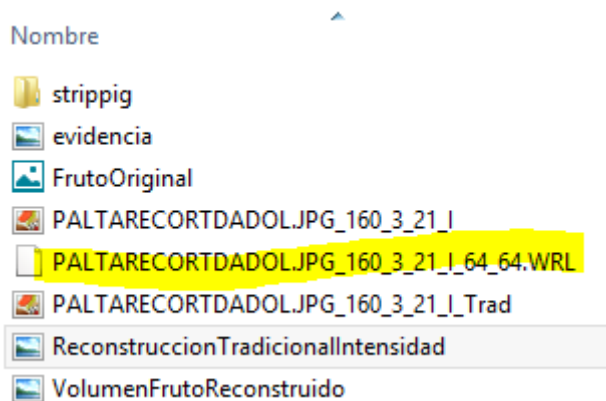


Fuente: Elaboración propia

```
//convertir el archivo 3d en vrmf
PCTSTR pszFormat = TEXT("%s_%s_%d_%d.WRL");
StringCbPrintFW(GVRMLfile,(256*sizeof(WCHAR)),pszFormat,GdisparityMap,GleftFile,VERTS_PER_EDGE_WIDTH,VERTS_PER_EDGE_HEIGHT);
//Este archivo se exporta en con el siguiente método
void exportVRML(WCHAR *filename,D3DNMVERTEX* pNMVertex,WORD* pIndices)
//y se guarda con esta función
exportVRML(GVRMLfile,pNMVertexSave,pIndicesSave);
```

Se presenta el archivo .wrl que se ha logrado mediante la aplicación del método exportVRML

Ilustración 45 Generación del archivo .wrl



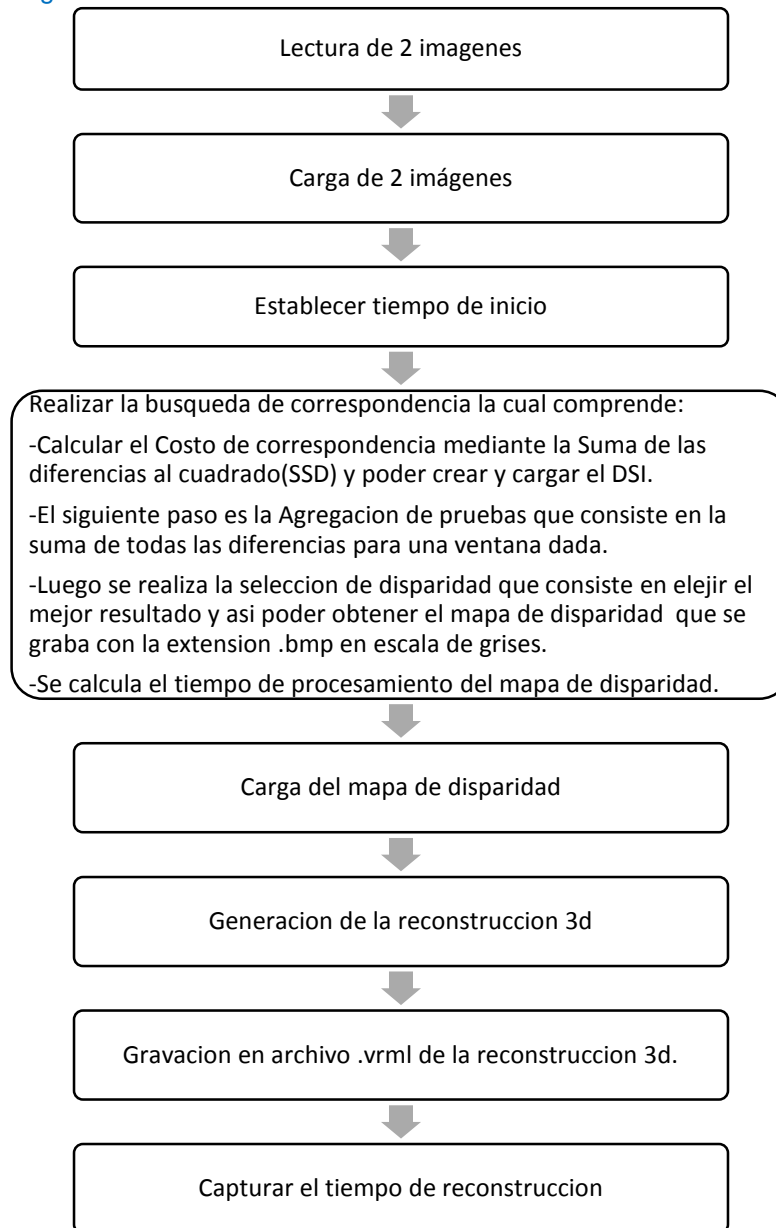
Fuente: Elaboración propia



Como se puede observar este es el resultado final de la explicación del método 1: Técnica Tradicional, la cual consiste en cargar las imágenes, luego realizar la búsqueda de correspondencias para poder lograr el mapa de disparidad y a partir del mapa de disparidad realizar la reconstrucción 3d de imágenes rayos x del fruto de palta y esa reconstrucción guardarlo en un archivo .vrml.

Reconstrucción 3d mediante el método 1: Técnica tradicional.

Ilustración 46 Diagrama: Reconstrucción 3d mediante el Método 1: Técnica Tradicional.



Fuente: Elaboración propia



5.1.3. Método 2: Técnica de Pistas

Se ha implementado el método de Visión Estéreo mediante la Técnica de pistas para la reconstrucción 3d de imágenes rayos x.

A continuación explicaremos de manera detallada este método.

La lectura y carga de imágenes se repite el mismo procedimiento que el método 1: Técnica Tradicional.

Con la técnica de pistas, las imágenes de entrada ya cargadas se procesan en grupos y el proceso de búsqueda de correspondencias sólo se produce en el rango, deben ser lo más pequeños posibles, manteniendo los resultados intermedios para su reutilización durante el proceso.

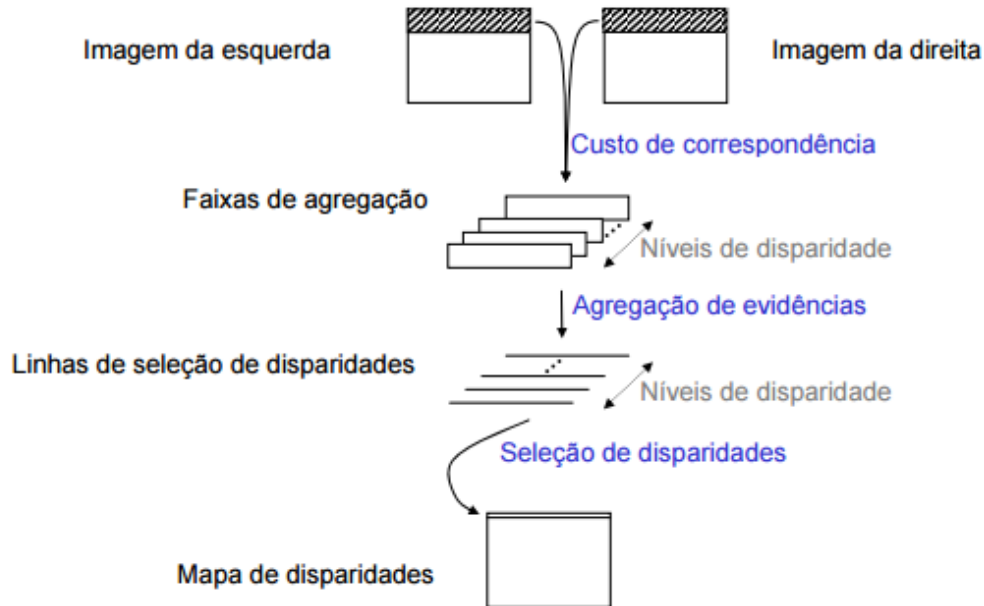
La altura más baja del rango es la altura de la ventana más 1, porque durante el proceso de agregación es necesario disponer de los resultados de costo de correspondencias y una altura menor no permitiría esto. Otra conclusión es que el rango debe tener el mismo ancho que la imagen de entrada porque los anchos más pequeños eliminan los datos útiles para su reutilización durante el desplazamiento lateral de la ventana.

El proceso de costo de correspondencia sólo maneja las bandas de las imágenes de entrada rellenando las bandas de agregación (bloque pequeño del DSI) con la altura de la ventana (una línea debajo estará libre en la primera ronda).

Estos se trabajaron en el proceso de agregación de pruebas que producen las líneas de selección Disparity, que se abordan en el proceso de selección de la disparidad al producir una línea en el mapa de disparidad (ilustración 47).



Ilustración 47 Diagrama de la Técnica de Pistas



Fuente: (Marcelo & Roseli, 2007)

Después de la primera ronda, el costo de correspondencia se calcula en una sola línea (inmediatamente inferior) de las imágenes de entrada, el resultado se almacena en esa línea que queda en los grupos de agregación de pistas. El proceso de agregación de pruebas ahora ignora la primera fila de agregación de pistas, produciendo líneas de selección de disparidades y normalmente siguiendo el proceso de selección de disparidades. Después de la segunda ronda, los datos de la primera fila de los grupos de agregación de pistas se suprimen, y en su lugar, los datos de costo de Correspondencias se colocan inmediatamente debajo de las imágenes de entrada, y así sucesivamente, hasta el final de la imagen. Este proceso recicla los datos que ya se habían



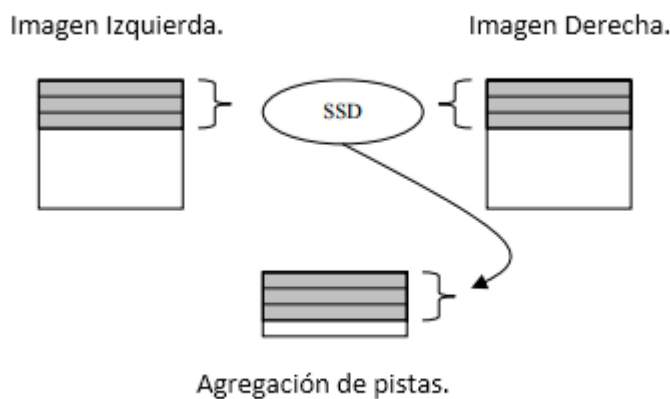
calculado en rondas anteriores, sólo elimina lo que es innecesario y añade nuevos.

Con el fin de facilitar la comprensión en el análisis de las etapas del costo de la correspondencia, la agregación de pruebas y la selección de disparidad. Se utilizan como ejemplo los datos de un par de imágenes de entrada con 320x200 con disparidad máxima de 10 y con ventana de 3x3.

a- Costo de Correspondencia

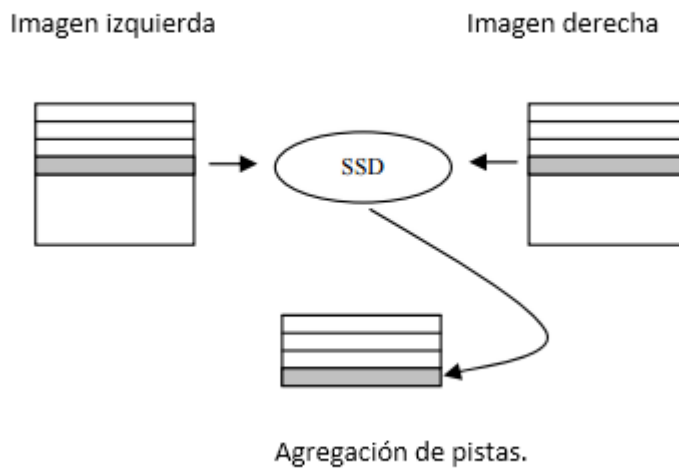
Para la primera ronda, se asignan 10 agregaciones de pistas de 320 de ancho y 4 de alto y se realiza el cálculo de correspondencia (SSD) para cada píxel de las líneas 0, 1 y 2 (ventana 3x3). Es decir, el píxel (45, 1) de la imagen izquierda se compara con el píxel (45.1) de la imagen derecha y el resultado se almacena en la posición (45.1) de la agregación de pista 0 (igual a Disparidad 0). A continuación, se compara el píxel (45.1) de la imagen izquierda con el píxel (44.1) de la imagen derecha y el resultado se almacena en la posición (45.1) de la agregación de pista 1 (equivalente a Disparidad 1), y así sucesivamente tal como lo muestra la ilustración 48.

Ilustración 48 Costo de Correspondencia para el primer llenado de agregación de pistas para una ventana 3X3.



En la segunda ronda se usan las mismas 10 agregaciones de pistas, pero ahora se realiza el cálculo del costo de correspondencia de hacer coincidir sólo una línea por debajo de las imágenes de entrada y el resultado se almacena en la línea que queda de la primera ronda de la agregación tal como lo muestra la ilustración 49.

Ilustración 49 Costo por correspondencia de otras líneas hasta el final de las imágenes de entrada.



En la tercera ronda, la línea 1 de las agregaciones de pistas se sustituye por el cálculo del costo de correspondencias de la siguiente fila por debajo de las imágenes de entrada. El proceso continúa sustituyendo así la agregación innecesaria de la línea y añadiendo nuevas filas recién calculadas.



b- Agregación de pruebas

Para el proceso de agregación se asignan 10 líneas con 320 píxeles cada una, estas líneas se llaman **líneas de selección de disparidad**. En estas líneas se almacenan los resultados de las sumas de los valores contenidos en la **agregación de pistas**, recordando que las **pistas** tienen la altura de la ventana más 1.

En la primera ronda, sólo deben considerarse las 3 primeras líneas. Siguiendo el mismo principio de filtrado de caja, cada desplazamiento de ventana hacia la derecha aprovecha el resultado de la agregación anterior añadiendo una columna a la derecha y restando una columna a la izquierda tal como lo muestra la ilustración 50.

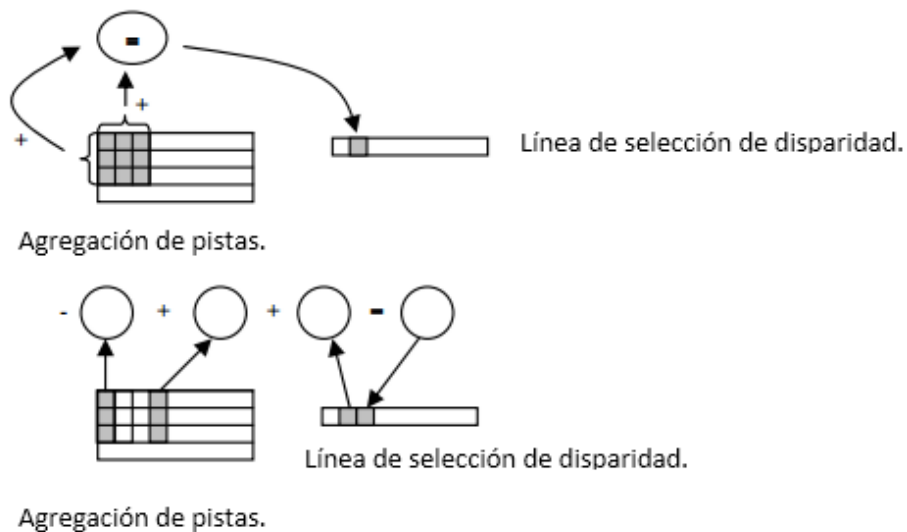
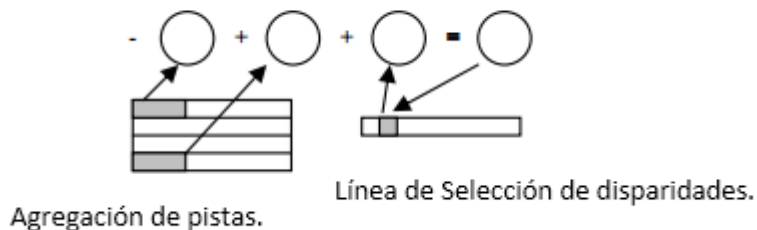


Ilustración 50 En la parte superior de la primera agregación, debajo de la siguiente utilizar los resultados obtenidos anteriormente.

En la segunda ronda, la técnica cambia. A partir de aquí, se toma el resultado de la **línea de selección de disparidades** de la ronda anterior, se resta la línea de ventana anterior y se añade la nueva línea de ventana (ilustración 51). Y así, continúa hasta el final de la exploración en las imágenes de entrada.

Ilustración 51 Proceso de agregación para las otras agregación de pistas.



c- Selección de disparidades

Finalmente, se crea otro plan con 320x200, donde el valor de la disparidad que tiene la mejor correspondencia se almacena línea por línea. La selección de la mejor correspondencia se produce en la comparación de la misma posición de las líneas de selección de disparidades, por ejemplo, para la posición 70 entre todas las líneas, la línea 4 obtuvo el resultado más bajo (WTA). Así, en el mapa de disparidades, la sombra gris que corresponde a la disparidad 4 (donde y representa el punto del exploración principal) se almacena en la posición (70, y).

Todo lo expuesto con respecto a la búsqueda de correspondencias mediante el método 2: Técnica de pistas (Costo de correspondencia, Agregación de



pruebas y Selección de disparidades), para obtener el mapa de disparidad, se ha implementado en el lenguaje de programación C++ de la siguiente forma.

//creamos el método de búsqueda de correspondencia mediante el método 2: //Técnica de //pistas.

```
void FastCorrespSearchCV(unsigned int width,unsigned int height,unsigned int
maxDisp,unsigned int maxLineVar,int windowSize) {
maxDisparity=maxDisp;
unsigned long n;
int yVar;
if(windowSize%2==0) {log(L"windowSize invalido, número par.\n");return;}
if(windowSize<3) {log(L"windowSize invalido, menor que 3.\n");return;}
if(maxDisp<1) {log(L"maxDisp invalido, menor que 1.\n");return;}
if(maxLineVar<1) {log(L"maxLineVar invalido, menor que 1.\n");return;}
if(maxLineVar%2==0){log(L"maxLineVar invalido, número par.\n");return;}
n=(windowSize-1)/2;
yVar=(maxLineVar-1)/2;
```

// Asignación de memoria para el intervalo de agregación

```
unsigned long stripSize,numberOfStrips,aggregationStripSize;
void *voidAggregationStrip;
unsigned long *pointLocAggregationStrip;
if(maxDisp<=0) maxDisp=1;
if(maxLineVar<=0) maxLineVar=1;
stripSize=width*(windowSize+1);
numberOfStrips=maxDisp*maxLineVar;
aggregationStripSize=stripSize*numberOfStrips;
voidAggregationStrip= new unsigned long[aggregationStripSize];
pointLocAggregationStrip=(unsigned long *)voidAggregationStrip;
```

// Asignación de memoria para las líneas de selección de disparidad

```
unsigned long dispSelectLineSize;
void *voidDispSelectLine;
unsigned long *pointLocDispSelectLine;
dispSelectLineSize=width*numberOfStrips;//Debe tener una fila para cada pista
voidDispSelectLine= new unsigned long[dispSelectLineSize];
pointLocDispSelectLine=(unsigned long *)voidDispSelectLine;
// Asignación de memoria para el Mapa de Disparidades
unsigned long *pointLocDispMap;
voidDispMap= new unsigned long[width*height];
pointLocDispMap=(unsigned long *)voidDispMap;
WCHAR text[10];
LPCTSTR pszFormat = TEXT("%0.0f %%");
//Generar el mapa de disparidad
unsigned int x,yFull,searchRegionX,searchRegionY;
int y;
```



```

unsigned
longstepStrip,stepLineStrip,stepImageL,stepImageR,stepImageUnderStripIn,stepImageUnderStripOut,stepLine,increment;
unsigned long yFullProdWidth;
long jumpLineIn,jumpLineOut;
startST=clock();//Cuenta sólo la ejecución del código de búsqueda correspondencia.
for(yFull=0;yFull<height;yFull++) {
yFullProdWidth=yFull*width;
StringCbPrintFW(text,(10*sizeof(WCHAR)),pszFormat,((float)yFull/(height))*100 );
SetDlgItemText(hDialog, IDC_STATIC_PROGRESS,text);
long tempL,tempR,tempDiff,Sdiff;
IMGpixel *pointLoclmgL,*pointLoclmgR;
if((yFull>=n)&&(yFull<height-n)) { if(yFull==n) {

```

//Matching Cost

```

pointLoclmgR=(IMGpixel *)voidlmgR;
for(searchRegionY=0;searchRegionY<maxLineVar;searchRegionY++)
for(stepStrip=0,searchRegionX=0;searchRegionX<maxDisp;searchRegionX++) {
stepStrip=(searchRegionY*maxDisp*stripSize)+searchRegionX*stripSize;
pointLoclmgL=(IMGpixel *)voidlmgL;
for(y=0;y<windowSize;y++){stepLineStrip=y*width;
if( (y>=yVar)&&(y<=((int)height-(1+yVar))) ) {
stepImageR=(y+searchRegionY-yVar)*width;
for(x=0;x<width;x++,pointLoclmgL++) {
if(x>=(maxDisp-1)) {
increment=stepImageR+(x-searchRegionX);
tempL=pointLoclmgL->r;
tempR=pointLoclmgR[increment].r;
tempDiff=tempL-tempR;
Sdiff=tempDiff*tempDiff;
tempL=pointLoclmgL->g;
tempR=pointLoclmgR[increment].g;
tempDiff=tempL-tempR;
Sdiff=(tempDiff*tempDiff)+Sdiff;
tempL=pointLoclmgL->b;
tempR=pointLoclmgR[increment].b;
tempDiff=tempL-tempR;
Sdiff=(tempDiff*tempDiff)+Sdiff;
pointLocAggregationStrip[stepStrip+stepLineStrip+x]=Sdiff; } else
pointLocAggregationStrip[stepStrip+stepLineStrip+x]=195075;
} } else { for(x=0;x<width;x++)
pointLocAggregationStrip[stepStrip+stepLineStrip+x]=195075; } } }

```

//Agregación de evidencia

```

for (searchRegionY=0;searchRegionY<maxLineVar;searchRegionY++)
for (stepLine=0,searchRegionX=0;searchRegionX<maxDisp;searchRegionX++)
{ stepLine=(searchRegionY*maxDisp*width)+(searchRegionX*width);
stepStrip=(searchRegionY*maxDisp*stripSize)+(searchRegionX*stripSize);
if((yFull>=n)&&(yFull<height-n)) }
jumpLineOut=0;
jumpLineIn=windowSize; }

```



//Disparidad de selección

```

unsigned long bestResult;
unsigned long disp;
unsigned int part,color;
part=255/maxDisp;
for(x=0;x<width;x++) { disp=0;
bestResult=pointLocDispSelectLine[x];
for(searchRegionY=0;searchRegionY<maxLineVar;searchRegionY++)
for(stepLine=0,searchRegionX=0;searchRegionX<maxDisp;searchRegionX++)
{ stepLine=(searchRegionY*maxDisp*width)+searchRegionX*width;
if((yFull>=n)&&(yFull<height-n))
if(bestResult>pointLocDispSelectLine[stepLine+x])
{disp=searchRegionX;
bestResult=pointLocDispSelectLine[stepLine+x];} }
color=part*disp;
pointLocDispMap[(yFullProdWidth)+x]=color;
} }

```

finishST=clock();

durationST=finishST-startST;

//Luego de a ver realizado la selección de disparidad se obtiene el mapa de disparidad y se guarda //en escala de grises.

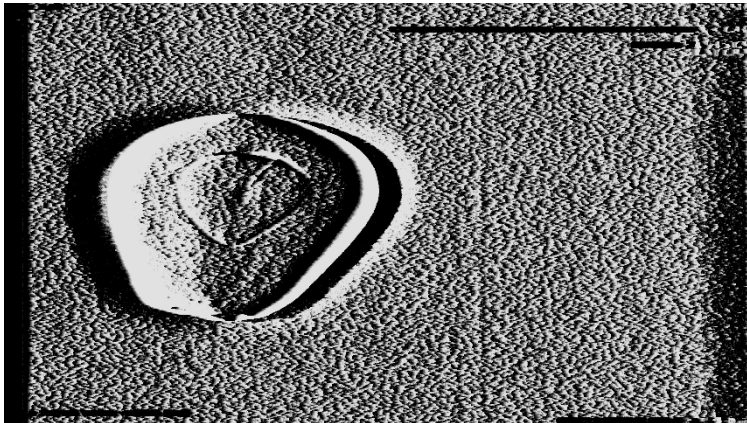
```

Bitmap DispMap(GleftFile);
unsigned int xs,ys;
unsigned long ysProdWidth;
pointLocDispMap=(unsigned long *)voidDispMapLegacy;
for (ys=0; ys<height;ys++){
ysProdWidth=ys*width;
for (xs=0; xs<width;xs++){
color=pointLocDispMap[(ys*width)+xs];
DispMap.SetPixel(xs, ys, Color(255,color,color,color));}}
CLSID clsid;
GetEncoderClsid(L"image/bmp", &clsid);
Graphics graphics(hdc);
DispMap.Save(disparityMap, &clsid, NULL);

```

Al final tenemos el mapa de disparidad (En pantalla como en mapa de bits) aplicando el método 2: Técnica de pistas que se almacena con una extensión .bmp en escala de grises.

Ilustración 52 El resultado del mapa de disparidad del fruto de la palta aplicando el método 2: Técnica de pistas por búsqueda de intensidad.



Fuente: Elaboración propia.

Después de obtener el mapa de disparidad se genera una malla 3d.

Generación de la malla 3d

Se sigue el mismo proceso que se desarrolló en el método tradicional para la generación de la malla 3d.

Resultado del algoritmo de reconstrucción mediante la Técnica de Pistas.

Ilustración 53 Resultado del algoritmo de reconstrucción mediante el método 2: Técnica de pistas



Fuente: Elaboración propia

Reconstrucción 3d mediante el Método 2: Técnica de Pistas

Ilustración 54 Diagrama: Reconstrucción 3d mediante el Método 2: Técnica de Pistas



{

Fuente: Elaboración propia

VI. CONCLUSIONES Y RECOMENDACIONES

6.1. Conclusiones



6.1.1. El Método 1: Técnica tradicional que procesa el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles) es eficiente en el aspecto del tiempo de procesamiento en relación a la técnica de Pistas.

6.1.2. El Método 2: Técnica de Pistas que procesa el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles) es eficiente en el aspecto del tiempo de procesamiento en relación al Método 1: Técnica tradicional.

6.1.3. El método 1: Técnica tradicional que reconstruye el fruto de la palta con semilla mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles) es eficiente en el aspecto del tiempo de reconstrucción en relación al Método 2: Técnica de Pistas.

6.1.4. El Método 2: Técnica de Pistas que reconstruye el fruto de la palta con semilla mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392 pixeles) es eficiente en el aspecto del tiempo de reconstrucción en relación al Método 1: Técnica Tradicional.

6.1.5. El Método 2: Técnica de Pistas que procesa el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (648x864 pixeles) es la que consume menos memoria en el tiempo de procesamiento en relación al Método 1: Técnica Tradicional.

6.1.6. El Método 2: Técnica de Pistas que procesa el mapa de disparidad mediante el ingreso de 2 imágenes rayos x cuya dimensión es (1792x2392



pixeles) es la que consume menos memoria en el tiempo de procesamiento en relación al método 1: Técnica Tradicional.

6.1.7. Ambos métodos son cruciales para la reconstrucción 3d de imágenes rayos x de fruto con palta. La diferencia de estos métodos son el tiempo de procesamiento, el consumo de memoria de acuerdo a la resolución de la imagen y los parámetros de entrada en la búsqueda de correspondencia. La precisión está expresando en centímetros cúbicos y me arroja como resultado que el objeto reconstruido mediante la aplicación tiene una precisión de +- 5.88 centímetros cúbicos con respecto al objeto real. Esto demuestra que según la percepción del usuario en ver el objeto reconstruido con un mejor panorama, lo sustenta con este indicador.

6.1.8. La técnica de visión estéreo requiere una alta precisión en la toma de las imágenes, ya que es vital para una buena reconstrucción.

6.2. Recomendaciones

Se plantea a continuación algunas recomendaciones, con el fin de extender los objetivos de la investigación.

6.2.1. Mejorar la adquisición en las imágenes rayos x.



REFERENCIAS

- ARNOLD, R. D. (1983). Automated Stereo Perception. *Technical Report AIM-351, Artificial Intelligence Laboratory, Stanford University.*
- Baka, N., Kaptein, B. d., Giphart, J., & Niessen, J. &. (2011). 2D–3D shape reconstruction of the distal femur from stereo X-ray imaging using statistical shape models.
- Bates, T. (2000). *Strategies for College and University Leaders* . San Fransico: McGraw-Hill.
- BOBICK, A. F., & INTILLE, S. S. (1999). Large occlusion stereo. *International Journal of*, 181–200.
- BOYKOV, Y., VEKSLER, O., & ZABIH, R. (1999.). Fast approximate energy minimization via graph cuts. *In Seventh International Conference on Computer Vision (ICCV'99)*, 377–384.
- C.Archimedes, R. B. (Julio-2004). *law sets a good energy-minimization example*. Physics Education.
- Camacho, L. (2013). Sistema de reconstrucción 3D multicámara. *Centro de investigaciones en óptica A. C.*
- Comini, G., Del Guidice, S., Lewis, R., & Zienkiewicz, O. (1974). Finite element solution of non-linear heat conduction problems with special reference to phase change. 613-624.
- Davis. (2001). *Athabasca University: Conversion from Traditional Distance Education to Online Courses, Programs and Services*. Retrieved from International Review of Research in Open and Distance Learning: technologic, <http://www.educared.net/>
- Gómez, A., Cerón, T., & Rodríguez, V. &. (2007). Aspectos tecnológicos de la congelación en alimentos. *Temas Selectos de Ingeniería de Alimentos*, 180-86.
- Gonzales, R., & Words, R. (2000). *Procesamiento de imagenes Digitales*. Edgard Blucher Ltda.
- HANNAH, M. J. (1974). Computer Matching of Áreas in Stereo Images. *Ph.D thesis Stanford University.*
- Heldman, D. R., & Hartel, R. W. (1997). Principles of food processing. Freezing and frozen-food storage. 113-137.
- HIRSCHMÜLLER, H. (2005). Accurate and efficient stereo processing by semiglobal matching and mutual information. 2, 807–814.
- HIRSCHMÜLLER, H. (2008). Stereo Processing by Semiglobal Matching and Mutual Information. *IEEE Transactions On Pattern Anaylsis And Machine Intelligence.*
- HIRSCHMÜLLER, H., INNOCENT, P. R., & GARIBALDI, J. M. (2002). Real-time correlation-based stereo vision with reduced border errors. *International.*
- Hosseinian, S., & Arefi, H. (2015). 3d Reconstruction From Multi-View Medical X-Ray Images – Review And Evaluation Of Existing Methods. *The International Archives of the Photogrammetry. Remote Sensing and Spatial Information Sciences, Volume XL-1/W5*, 319-326.
- Hosseinian, S., & Arefi, H. (2015). 3d Reconstruction From Multi-View Medical X-Ray Images – Review And Evaluation Of Existing Methods. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 319-326.



- Hossian, M., Cleland, D., & A.C., C. (1992). Prediction of freezing and thawing times for foods of regular multi-dimensional shape by using an analytically derived geometric factor.
- KANADE, T. e. (1996). . A stereo machine for video-rate dense depth mapping and its new applications. *In IEEE Computer Society Conference on Computer.*
- KANADE, T., & OKUTOMI, M. (1994). A Stereo Matching Algorithm with an Adaptive Windows : Theory and Experiment. *IEEE Transactions On Pattern Analysis And Machine Intelligence.*
- KIMURA, R. e. (1999). A convolver-based real-time stereo machine (SAZAN). *In IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 457–463.*
- Lamecker, H. W.-C. (2006). Atlas-based 3D-Shape Reconstruction from X-Ray Images. *Zuse Institute Berlin - Visualization and Data Analysis Takustr 7,, 1-4.*
- López, C. (2014). Análisis de cuerpos no rígidos usando algoritmos de difusión.
- Marcelo, J. (2008). *RECONSTRUÇÃO TRIDIMENSIONAL DE BAIXO CUSTO.* Sao Paulo.
- Marcelo, J., & Roseli, L. (2007). SISTEMA DE RECONSTRUÇÃO 3D DE BAIXO. *GRAPHICA.*
- MCDONNELL, M. J. (1981). Box-filtering techniques. *Computer Graphics and Image, 65-70.*
- Pham, Q. (1991). Shape factor for the freezing time ellipses and ellipsoids. *J.Food Engng.*
- RUSSELL, S. J., & NORVIG, P. (2004). Inteligência Artificial. (Second Edition).
- SCHARSTEIN, D. (1999). View Synthesis Using Stereo Vision. *Springer.*
- SCHARSTEIN, D., & SZELISKI, R. (April-June 2002). A taxonomy and evaluation of dense twoframe stereo correspondence algorithms. *International Journal of Computer.*
- SCHARSTEIN, D., & SZELISKI, R. (2001). A Taxonomy and Evaluation of Dense TwoFrame Stereo Correspondence Algorithms. *Technical Report, MSR-TR-.*
- SUN, C. (2002). Fast Stereo Matching Using Rectangular Subregioning and 3D Maximum-Surface Techniques. *International Journal of Computer Vision, 99-117.*
- Ubieto, K. V. (2009). Reconstrucción 3D de modelos utilizando técnicas de Visión artificial.
- VEKSLER, O. (1999). Efficient Graph-based Energy Minimization Methods in. *Ph.D. thesis, Cornell University.*
- Virsedá, P., & Pinazo, J. (1998). Métodos de transferencia de calor en la refrigeración y congelación de alimentos. *Alimentación, Equipos y Tecnología, 161-167.*
- Zhenga, G. G. (2009). A 2D/3D correspondence building method for reconstruction of a.

Anexos

Anexo 1

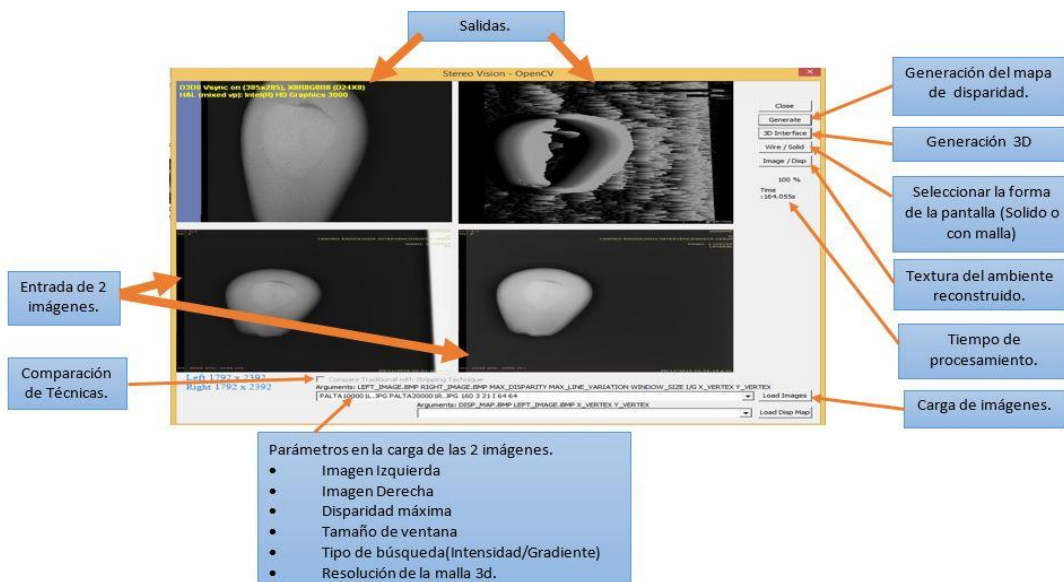


Método I: Reconstrucción 3d del fruto de la palta con semilla aplicando el Método Tradicional.

A continuación se muestra la interfaz que permite procesar el mapa de disparidad y la reconstrucción 3d del fruto de la palta con semilla, utilizando el método Tradicional.

1. Primero se realiza la lectura de 2 imágenes rayos x del fruto de la palta con semilla.
2. Cargar las imágenes rayos x del fruto de la palta con semilla.
3. Generar el mapa de disparidad.
4. Generar la reconstrucción 3d del fruto de la palta con semilla.
5. Seleccionar la forma de la reconstrucción (solido / con malla).

Ilustración 55 Argumentos de la Interfaz Gráfica para el procesamiento del mapa de disparidad y la reconstrucción 3d del fruto de la palta con semilla, utilizando el Método Tradicional.



Anexo 2: Prueba con cámara bajo las siguientes características:

Cannon 60 SD.

Lente 50 mm.

Apertura F11

Fuente: [Elaboración propia.](#)

Velocidad 1:60 segundos

Reconstrucción tridimensional (3d) del envase volt tomada con flash, mediante el ingreso de 2 imágenes cuyas dimensiones son 5184x3456 pixeles.



Ilustración 56 Reconstrucción 3d con el envase VOLT (Solido).

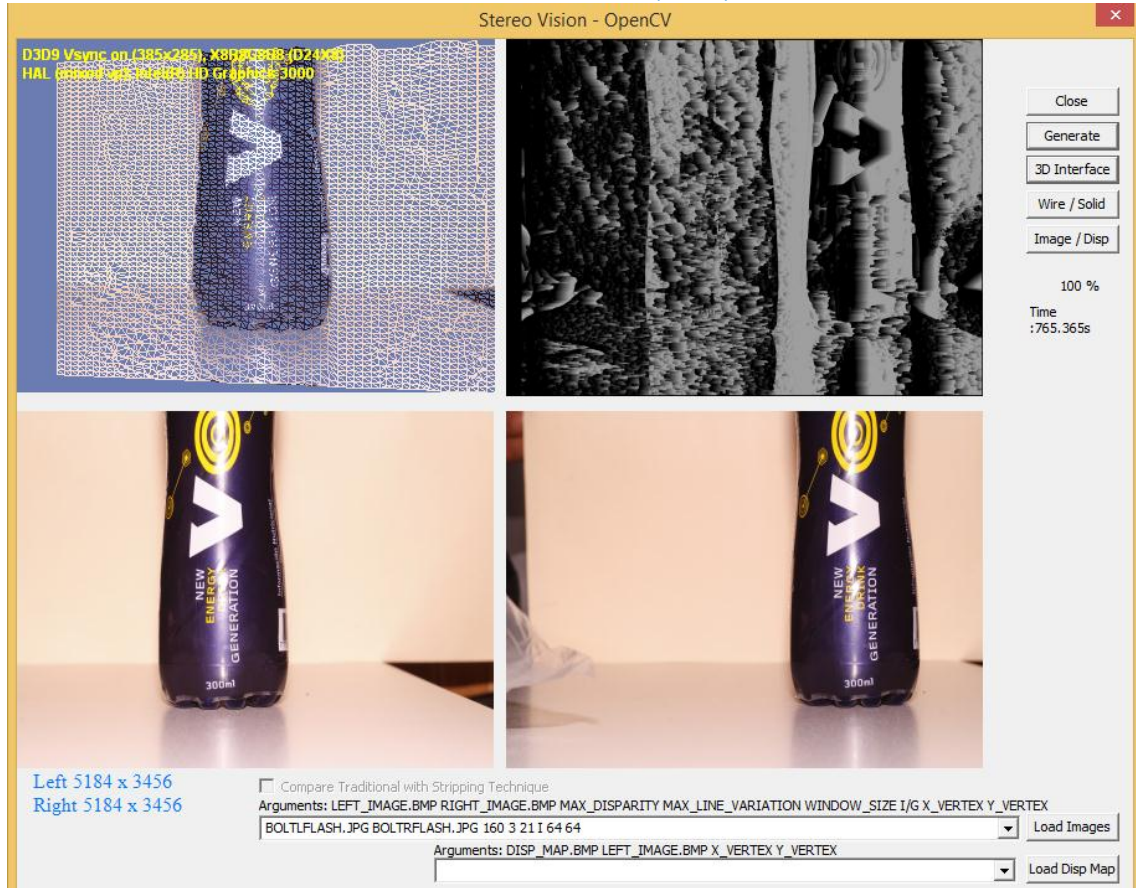


Fuente: Elaboración propia

Reconstrucción tridimensional (3d) del envase volt tomada con flash, mediante el ingreso de 2 imágenes cuyas dimensiones son 5184x3456 pixeles.



Ilustración 57 Reconstrucción 3d con el envase VOLT (malla).

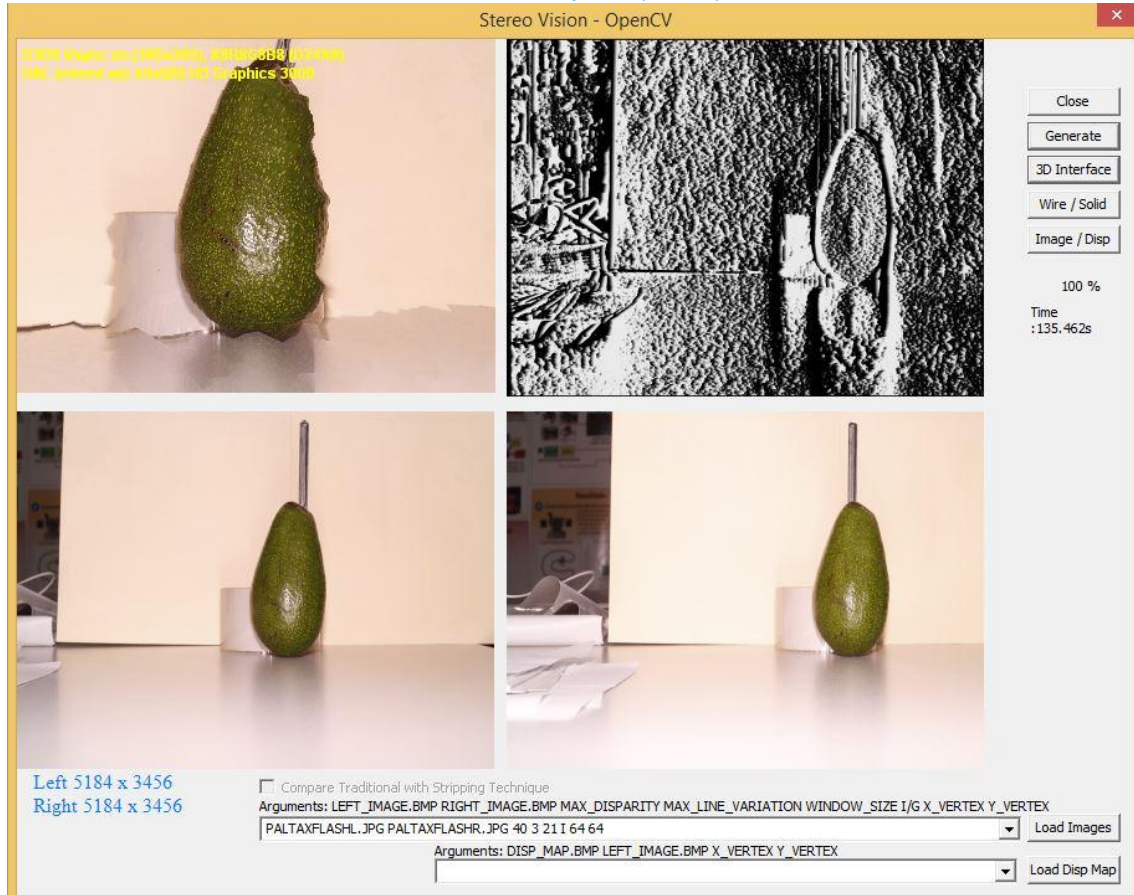


Fuente: Elaboración propia

Anexo 3: Reconstrucción tridimensional (3d) del fruto de la palta, tomada con flash, mediante el ingreso de 2 imágenes cuyas dimensiones son 5184x3456 pixeles con 40 de nivel de disparidad.



Ilustración 58 Reconstrucción 3d del fruto de la palta (solido).

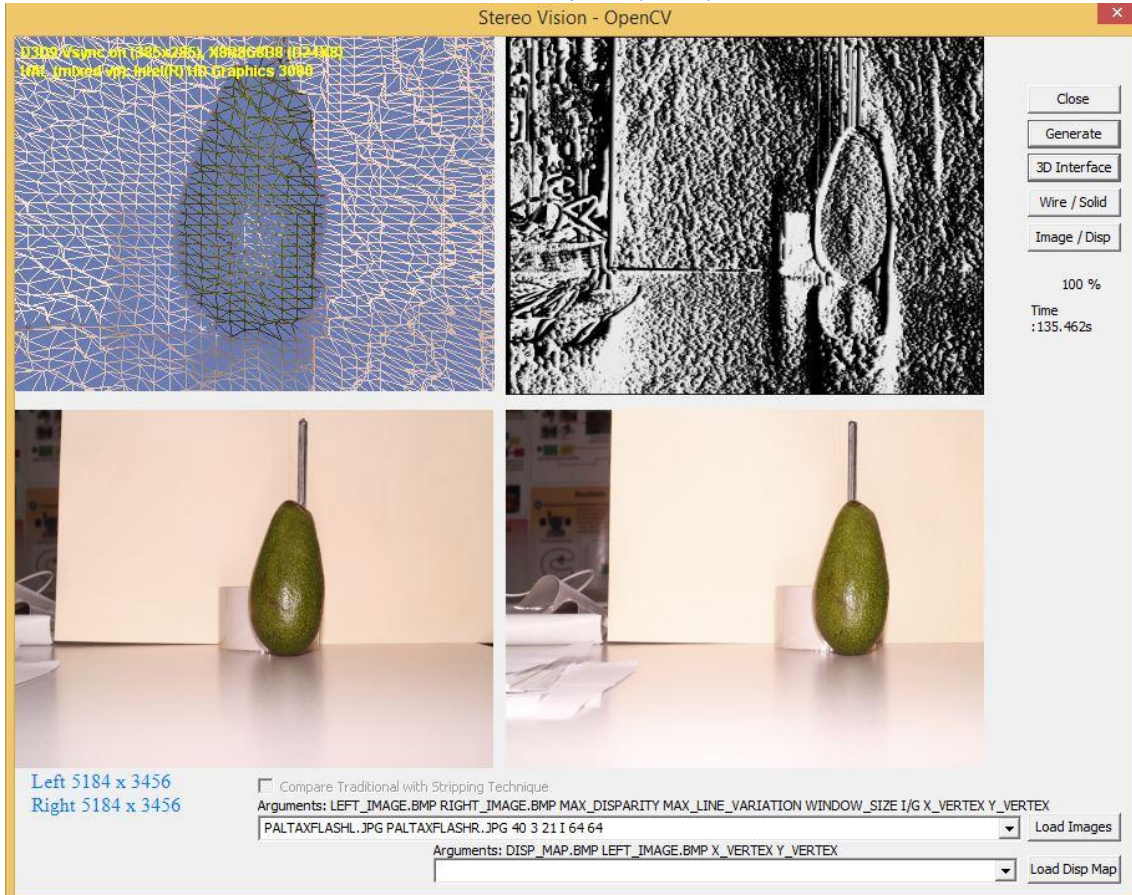


Fuente: Elaboración propia

Reconstrucción tridimensional (3d) del fruto de la palta, tomada con flash, mediante el ingreso de 2 imágenes cuyas dimensiones son 5184x3456 pixeles con 40 de nivel de disparidad.



Ilustración 59 Reconstrucción 3d del fruto de la palta (malla).

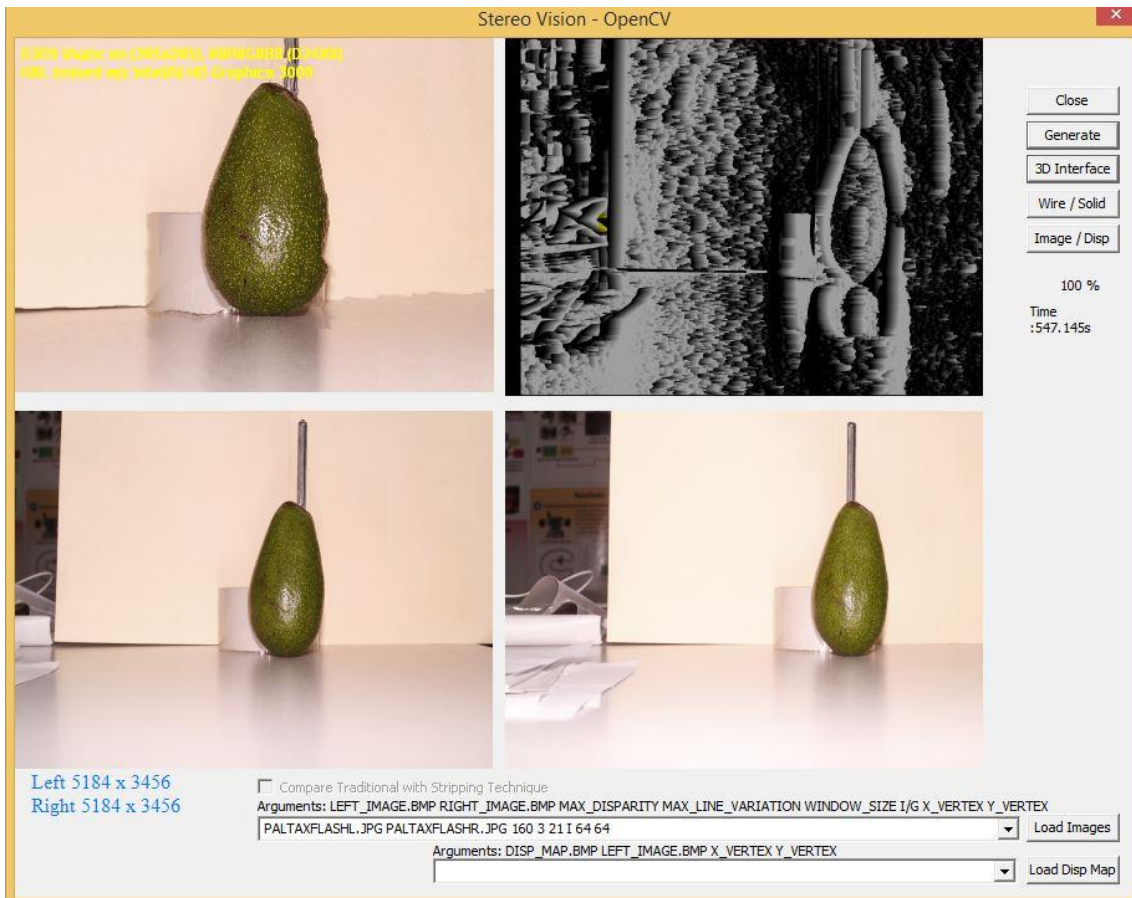


Fuente: Elaboración propia

Anexo 4: Reconstrucción tridimensional (3d) del fruto de la palta, tomada con flash, mediante el ingreso de 2 imágenes cuyas dimensiones son 5184x3456 pixeles con 160 de nivel de disparidad.



Ilustración 60 Reconstrucción 3d del fruto de la palta (solido) con un nivel de disparidad de 160.

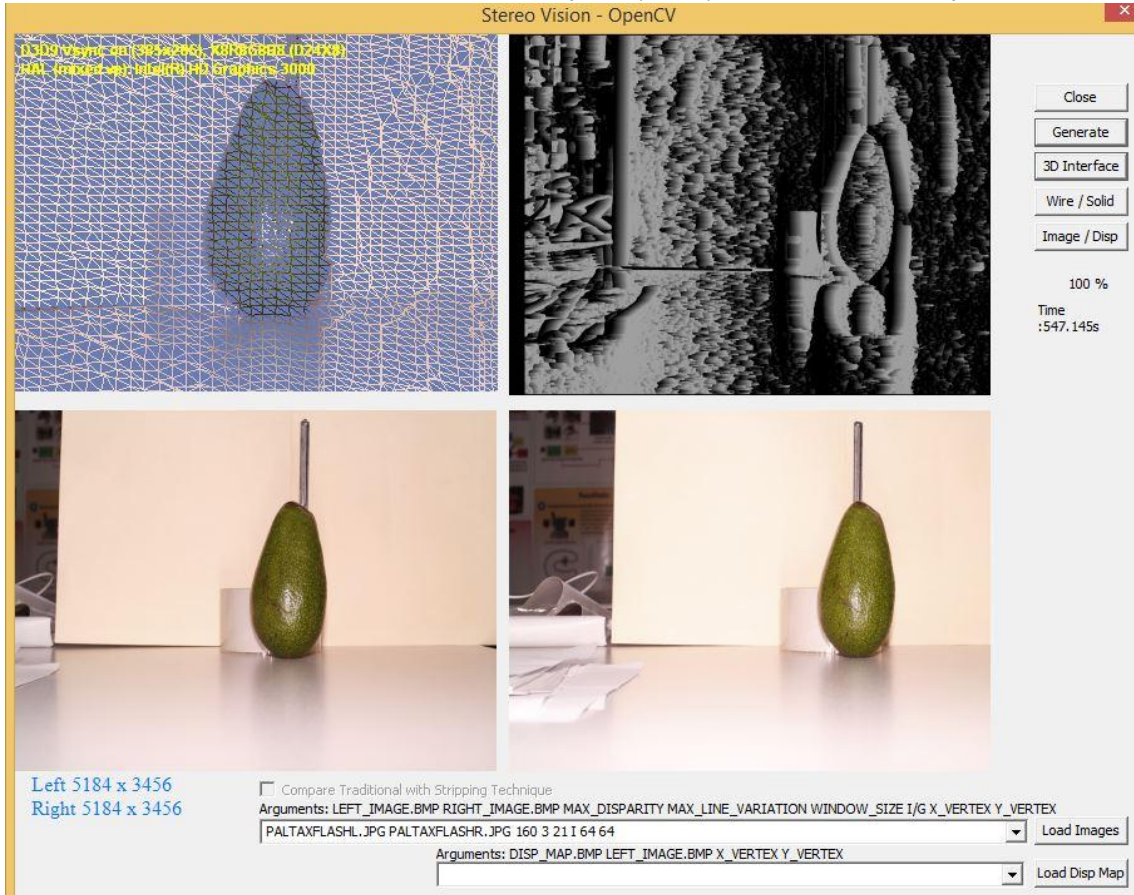


Fuente: Elaboración propia

Reconstrucción tridimensional (3d) del fruto de la palta, tomada con flash, mediante el ingreso de 2 imágenes cuyas dimensiones son 5184x3456 pixeles con 160 de nivel de disparidad.



Ilustración 61 Reconstrucción 3d del fruto de la palta (malla) con un nivel de disparidad de 160.

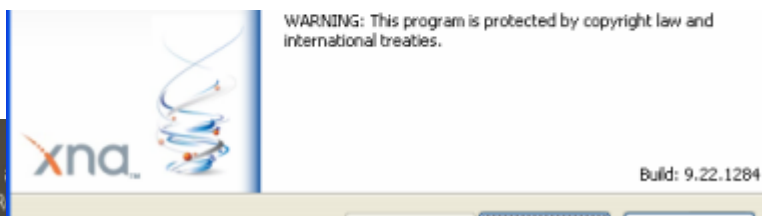


Fuente: Elaboración propia



Ilustración 62 Instalación de Directx SDK

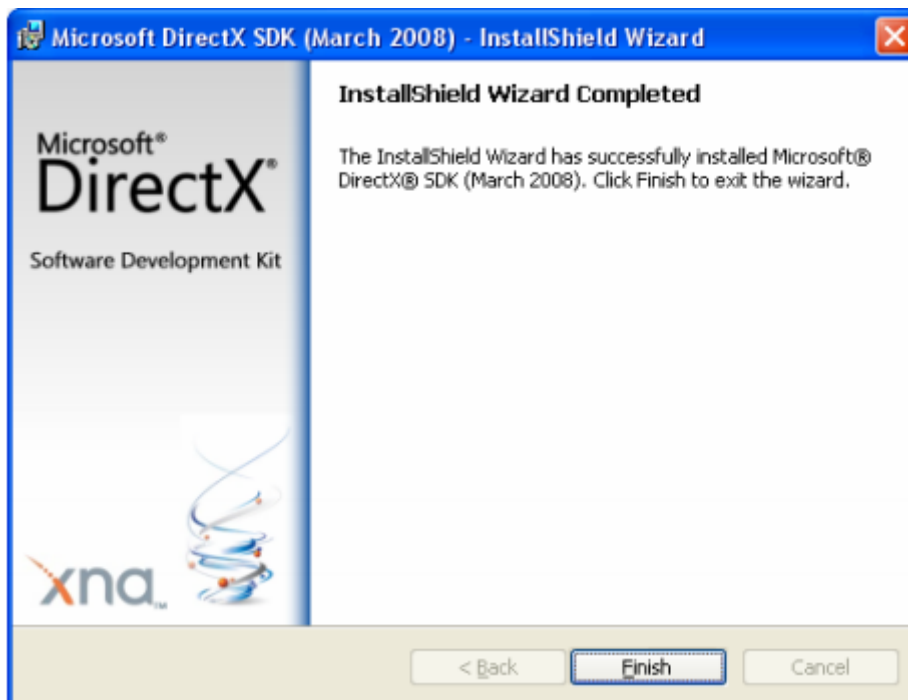
Anexo 5: Instalación de Directx SDK



Fuente: Elaboración propia



Ilustración 63 Finalización de la Instalación de Directx SDK



Fuente: Elaboración propia

Anexo 7: Código Fuente



Ilustración 64 Código Fuente de la aplicación

