



Universidad Autónoma de Querétaro
Facultad de Informática
Maestría en Ciencias de la Computación

Digitalización de sólidos utilizando análisis de franjas y técnicas de desplazamiento de fase en arquitecturas ARM.

Opción de titulación
Tesis

Que como parte de los requisitos para obtener el Grado de
Maestría en Ciencias de la Computación

Presenta:

Ing. Rodrigo Escobar Díaz Guerrero

Dirigido por:

Dr. Jesús Carlos Pedraza ortega

Dr. Jesús Carlos Pedraza Ortega
Presidente


Firma

Dr. Juan Manuel Ramos Arreguín
Secretario


Firma

Dr. Saúl Tovar Arriaga
Vocal

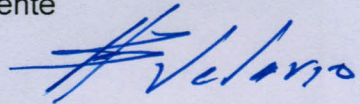

Firma

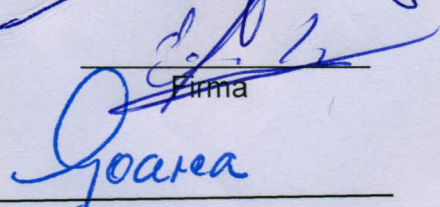
Dr. Arturo González Gutiérrez
Suplente


Firma

Dr. Efrén Gorrostieta Hurtado
Suplente


Firma


M.I.S.D Juan Salvador Hernández
Valerio
Director de la Facultad de Informática


Dra. Ma. Guadalupe Flavia Loarca
Piña
Directora de Investigación y Posgrado

RESUMEN

El presente trabajo describe cada una de las etapas para la digitalización de sólidos utilizando perfilometría por desplazamiento de fase (PSP, por sus siglas en inglés Phase-Shifting Profilometry). PSP puede describirse como el análisis de franjas sinusoidales o periódicas desplazadas en la fase proyectadas en la superficie de un objeto. Las etapas fundamentales para el análisis de franjas son el envolvimiento y el desdoblamiento de fase. El envolvimiento de fase es el proceso que determina los valores de la fase en un rango de 0 a 2π , en el cual la información de la altura del objeto se encuentra inmersa. El desdoblamiento de fase es el proceso en el que se remueve la discontinuidad de 2π , encontrando la altura del objeto. A lo largo del trabajo se describen dos técnicas de envolvimiento de fase y se analizan e implementan diversos algoritmos de desdoblamiento de fase. Por más complejo que sea un algoritmo de desdoblamiento de fase existirán errores de discontinuidad en la fase inherentes a PSP, por lo que se propuso una metodología que añade una etapa de pre y post-procesamiento que mejora la resolución en la superficie. Una de las mayores aportaciones es el desarrollo de dos algoritmos que permiten eliminar discontinuidades después del desdoblamiento de fase. Dentro de la metodología desarrollada se describe una forma de calibración utilizando una estimación geométrica a un plano de referencia. Cada una de las etapas necesarias para la digitalización de sólidos utilizando PSP fue implementada en una tarjeta de desarrollo basada en una arquitectura ARM por sus siglas en inglés Advanced RISC Machine. Las tarjetas empleadas son Raspberry Pi 2 B, BeagleBone Black, UDOO Quad y Radxa Rock Pro.

(Palabras clave: perfilometría por desplazamiento de fase (Phase-Shifting Profilometry), envolvimiento de fase (wrapped phase), desdoblamiento de fase (phase unwrapping), reconstrucción tridimensional, análisis de franjas, ARM)

SUMMARY

This work describes each of steps for digital scan using Phase-Shifting Profilometry (PSP). PSP can be described as analysis of sinusoidal or periodic fringes displaced in phase, after are projected on surface of an object. The basic steps for fringe analysis are phase wrapping and phase unwrapping. Phase wrapping is the process that determines the values of the phase in a range from 0 to 2π , which contain the height of object. Phase unwrapping is the process in which the 2π discontinuity is removed, obtained the height of object. Throughout of the paper two techniques of phase wrapping are described and various phase unwrapping algorithms are analyzed and evaluated. Regardless the complexity of one algorithm of phase unwrapping, there will be errors of discontinuities in the phase inherent to PSP, so a methodology is proposed where we add a pre and post-processing to improve resolution on the surface, one of the greatest contributions is the development of two algorithms that eliminate discontinuities after of a phase unwrapping algorithm. A calibration way using a geometric estimation to reference plane is described in the methodology developed. Each step necessary for the tridimensional scan were performed in a development board based on ARM (Advanced RISC Machine) architecture. The boards used are Raspberry Pi 2 B, BEAGLEBONE Black, UDOO, and Radxa Rock Pro Quad.

(Keywords: Phase-Shifting Profilometry, wrapped phase, phase unwrapping, three-dimensional reconstruction, analysis of fringes, ARM)

“Tu tiempo es limitado, así que no lo desperdicies viviendo la vida de alguien más. No te dejes atrapar por el dogma, que es vivir con los resultados de los pensamientos de otras personas. No dejes que el ruido de las opiniones de otros ahogue tu voz interior. Y lo más importante: ten el coraje de seguir a tu corazón e intuición. De algún modo ellos ya saben lo que realmente quieres ser. Todo lo demás es secundario”.

Steve Jobs
Co-fundador de Apple

Dedicada con todo cariño al gran erudito: Andrés Díaz Guerrero Altamirano,
Mi abuelo.

AGRADECIMIENTOS

A CONACYT por la beca otorgada por 2 años en la Maestría en Ciencias de la Computación realizada en la facultad de Informática de la Universidad Autónoma de Querétaro. También al Fondo de Proyectos Especiales de Rectoría (FOPER) por otorgar la mayoría de los materiales utilizados en la tesis.

A mi familia por apoyarme en todo momento, en especial a mi mamá que desde pequeño ha sido una inspiración en mi vida, ella es sinónimo de lucha incansable, de amor inapagable y de alegría inagotable. A mi hermano que sigue adelante con sus sueños sin importar a quien le agraden, sin importar el costo de alcanzarlos, a ese compañero de la vida que de pequeño me cuidó y de grande me escucha y aconseja.

A mis amigas de toda la vida Carolina y Mariana, por soportar todas mis pláticas ñoñas y darme aliento en los momentos más difíciles. A mi Dania, por permitirme robarle tanto tiempo.

A mis compañeros de maestría Yahvé, Alejandro, Juan, Carlos y Toño, por escucharme cuando me estancaba en algún algoritmo. A los alumnos de licenciatura que me ayudaron en la implementación, Sebastián, Said, Emiliano, Luis y Andrés.

A mis profesores, esos famosos doctores de la facultad de informática que combatían día a día la ignorancia de la facultad, al Dr. Jesús Carlos Pedraza Ortega por apoyarme en cada sueño, en cada paso académico, por ser mi tutor, asesor, mentor y amigo. Siempre un gran ejemplo de trabajo, responsabilidad, equidad, perspicacia y solidaridad. En las buenas y en las malas siempre encontré un refugio en su cubículo.

TABLA DE CONTENIDOS

1. INTRODUCCIÓN	11
1.1 OBJETIVOS	11
1.1.1 <i>Objetivo general</i>	11
1.1.2 <i>Objetivos específicos:</i>	12
1.2 JUSTIFICACIÓN.....	12
1.3 ALCANCES Y LIMITACIONES	13
1.3.1 <i>Alcances</i>	13
1.3.2 <i>Limitaciones</i>	13
1.4 ORGANIZACIÓN DE LA TESIS	14
2. MARCO TEÓRICO	15
2.1 RECONSTRUCCIÓN TRIDIMENSIONAL.	15
2.2 ANÁLISIS DE PROYECCIÓN DE FRANJAS.	17
2.3 PROCESADORES ARM.	21
2.3.1 <i>Reconstrucción 3D en sistemas embebidos</i>	24
3. METODOLOGÍA	25
3.1 PERFILOMETRÍA POR DESPLAZAMIENTO DE FASE O PSP	25
3.1.1 <i>Algoritmo three-step</i>	26
3.1.2 <i>Algoritmo four-step</i>	28
3.2 ENVOLVIMIENTO Y DESDOBLAMIENTO DE FASE	28
3.2.1 <i>Simple unwrap</i>	30
3.2.2 <i>Itoh tradicional e inverso</i>	30
3.2.3 <i>Graph Cuts</i>	30
3.2.4 <i>Algoritmo “Quality-guided”</i>	30
3.3 ADQUISICIÓN DE IMÁGENES	32
3.3.1 <i>Calibración de las imágenes</i>	33
3.4 PRE-PROCESAMIENTO DE LA IMAGEN	36
3.5 POST-PROCESAMIENTO	37

4. RESULTADOS	39
4.1 SIMULACIÓN	39
4.2 IMPLEMENTACIÓN EN UNA PC.....	44
4.3 IMPLEMENTACIÓN EN UNA TARJETA DE DESARROLLO CON PROCESADOR ARM....	51
5. CONCLUSIONES	56
6. REFERENCIAS	58
7. APÉNDICE.....	61

ÍNDICE DE TABLAS

1.Características principales de diferentes tarjetas de desarrollo con procesadores ARM, *Dólares americanos.	22
2.Especificaciones técnicas de la computadora usada en la investigación.	42
3.Tabla de resultados utilizando “three-step” con imágenes de profundidad.	43
4.Tabla de resultados utilizando “four-step” con imágenes de profundidad.	44
5.Tabla de especificaciones técnicas de las cámaras utilizadas en el proyecto de investigación.....	46
6.Especificaciones técnicas significativas del proyector Acer k132.....	47

ÍNDICE DE DIAGRAMAS

1.Diagrama en el que se muestra la diferencia entre la simulación y el procesamiento de una imagen de un objeto real.....	40
---	----

ÍNDICE DE FIGURAS

1.Técnicas de adquisición 3D (Molleda meré, 2008).....	16
2.Técnicas ópticas (Molleda meré, 2008).....	16
3.Imágenes de franjas. (a) franja de interferencia, contorno generado por el interferómetro de Michelson, (b) una imagen de rejilla deformada por una forma de la superficie de un objeto obtenido con el método de proyección de franjas (Kato, 2009).....	18
4.Triangulación entre cámara proyector y objeto.	18
5.Transformada de Fourier utilizada en la reconstrucción 3D, se obtiene la fase modulada, se hace un cambio y filtro en el espacio espectral, y por último se aplica la inversa de la transformada para demodular la fase. (Kato, 2009).	19
6.Extracción del mapa de fase utilizando transformada de Fourier 2D. (a) Patrones de franjas de entrada. (b) Resultado de la transformada de Fourier aplicada en un espectro 2D. (c) Extracción del mapa de fase resultado de aplicar la transformada inversa (Kato, 2009).	19
7.Desplazamiento de fase. (a) Pasos del método desplazamiento de fase, (b) Ejemplo de digitalización de objetos utilizando desplazamiento de fase teniendo de entrada 4 imágenes desplazadas en la fase.	21
8.Diferentes tarjetas de desarrollo con procesadores ARM, Raspberry Pi 2B, BeagleBone Black, Udo Quad y Radxa Rock Pro.	22
9.Cruce de sección de patrones de franja sinusoidales. a(-120°),b(0°),c(120°).(Huang & Zhang , 2006).	26
10.Fase continua y fase envuelta (Gdeisat et al, 2012).	29
11.Sistema completo. Objeto real, cámara, proyección de 3 franjas desplazadas en la fase y proyectadas al objeto real, dispositivo de procesamiento y despliegue de la imagen tridimensional.....	32
12.Sistema de captura. Del lado izquierdo puede observarse el proyector, del lado derecho la cámara de adquisición y en el fondo el objeto a digitalizar.	33
13.Ejemplos de sombras en las imágenes y el resultado de una buena y mala calibración.	33

14. Patrón de referencia utilizado para calibrar las imágenes tomadas con la cámara digital.....	34
15. Ejemplos de imágenes sin calibrar.....	34
16. Binarización del patrón proyectado al plano y relación de puntos que coinciden con el patrón original.	35
17. Imágenes calibradas.	35
18. Resultado de la calibración y recorte de la zona de interés.	35
19. Ejemplo de ecualización del histograma donde se aplica una función para repartir uniformemente el histograma.	36
20. Ejemplos de imágenes de profundidad generados por computadora.	39
21. Ejemplo de simulación del proceso por medio de imágenes de profundidad...	40
22. (a) Ejemplo de imagen de profundidad, (b) representación tridimensional de la imagen de profundidad, (c) Reconstrucción usando PSP, (d) Representación tridimensional del error.....	41
23. (a) Ejemplo de imagen de profundidad, (b) Reconstrucción tridimensional usando PSP sin ruido, (c) Reconstrucción tridimensional usando PSP con ruido, (d) Representación del error de imagen de profundidad con ruido.....	42
24. Imagen de profundidad con bordes abruptos y alto contraste.....	42
25. Resultados gráficos de la evaluación de las técnicas “three-step” y “four-step” evaluados con los algoritmos de desdoblamiento de fase: “Simple unwrap”, “Itoh tradicional e inverso” y “Graph Cuts”	43
26. Patrones proyectados a una figura con forma de pescado tomadas de la página oficial del asistente de profesor Chen Lujie de la “Singapore University of technology and design”.....	44
27. Aplicación de diversos métodos de desdoblamiento de fase usando “three-step” y “four-step” a un objeto real.	45
28. Diferentes ejemplos de capturas con patrones proyectados a objetos reales desplazados en la fase 0, 90°, 180° y 270°. (a) fondo de un vaso desechable encima de una caja blanca. (b) máscara. (c) casa de cerámica. (d) sol de barro.	47

29.Comparación entre el desdoblamiento de fase antes de los algoritmos propuestos ((a), (c), (e)) y después de implementar los algoritmos propuestos ((b), (d), (f)).	48
30.Imagen del objeto original y diversas vistas de la reconstrucción tridimensional de una casa de cerámica.	49
31.Imagen del objeto original y diversas vistas de la reconstrucción tridimensional de una máscara de plástico.	49
32.Imagen del objeto original y diversas vistas de la reconstrucción tridimensional de una llanta de juguete.	50
33.Imagen del objeto original y diversas vistas de la reconstrucción tridimensional de un sol de barro.	50
34.Gráfica de velocidad en diversos lenguajes realizada por la empresa “Libre Solutions Pty Ltd” donde se añaden cadena de caracteres en ciclos (cada ciclo una cadena de 256 kiB).	51
35.Gráfica de uso de memoria en diversos lenguajes realizada por la empresa “Libre Solutions Pty Ltd” donde se añaden cadena de caracteres en ciclos (cada ciclo una cadena de 256 kiB).	52
36.Generación de franjas en las tarjetas de desarrollo. Del lado izquierdo utilizando <code>OpenCV</code> y del lado derecho <code>matplotlib</code> .	52
37.Ejemplo de mapa de fase dentro de las tarjetas de desarrollo.	53
38.Comparación entre desdoblamiento Simple unwrap en una tarjeta utilizando Python (imagen de la izquierda) y desdoblamiento Simple unwrap en una PC utilizando Matlab (imagen de la derecha).	53
39.Ejemplo de mapa de calidad implementado en la tarjeta de desarrollo.	54
40.Desdoblamiento de fase utilizando “Quality-guided”.	54
41.Ejemplo de resultado en 3D.	55

1. INTRODUCCIÓN

Este trabajo presenta la implementación del método Perfilometría por Desplazamiento de Fase (PSP, por sus siglas en inglés Phase-Shifting Profilometry) en una arquitectura ARM por sus siglas en inglés Advanced RISC Machine, para ello se describen los pasos necesarios para una reconstrucción exitosa los cuales son la adquisición de imágenes (incluyendo la calibración), pre-procesamiento, mapa de fase, desdoblamiento de fase y post-procesamiento.

Se muestran y evalúan diferentes algoritmos de desdoblamiento de fase (una etapa crucial para cualquier método de análisis de franjas) cómo son: “Simple unwrap”, Itoh tradicional, Itoh inverso, “Graphs cuts” y “Quality-guided”.

Se muestra una técnica de simulación que permite la cuantificación y visualización de métodos de análisis de franjas. Posteriormente se examinan los resultados obtenidos al implementar PSP en una PC y en una tarjeta de desarrollo que contiene un procesador ARM.

1.1 Objetivos

1.1.1 Objetivo general

Implementar y evaluar un algoritmo de digitalización de sólidos basado en Perfilometría por Desplazamiento de Fase aplicado al análisis de franjas en un dispositivo con arquitectura ARM. Al implementar un algoritmo simplificado en la etapa del desdoblamiento de fase de este método (la cual tiene una mayor complejidad computacional), se pretende reducir el tiempo de procesamiento, y pueda ser ejecutado en un dispositivo basado en la arquitectura ARM.

1.1.2 Objetivos específicos:

- Hacer un estudio comparativo de las tarjetas en el mercado que cuenten con procesadores ARM, seleccionando una que cumpla las características necesarias para la digitalización de sólidos.
- Implementar un algoritmo de Perfilometría por Desplazamiento de Fase en una PC estándar y en una tarjeta equipada con un procesador de arquitectura ARM para analizar las diferencias.
- Realizar pruebas del algoritmo y rendimiento del procesador ARM en los S.O. comunes.
- Recrear imágenes tridimensionales basadas en la obtención de imágenes 2D en una tarjeta de desarrollo que contenga un procesador ARM.

1.2 Justificación

La reconstrucción 3D en las últimas décadas ha sido ampliamente investigada para el desarrollo de videojuegos, animación, películas, realidad virtual, telecirugía y múltiples disciplinas en el campo de la ingeniería. Actualmente la reconstrucción 3D se enfoca en sistemas computacionales basados en arquitecturas x86 y x64.

En este trabajo se implementaron algoritmos de reconstrucción 3D en un dispositivo con procesador ARM cuya arquitectura se ocupa en el 98% de los dispositivos móviles actuales. La arquitectura ARM es compatible con múltiples sistemas operativos como Android, Linux (Debian, Fedora, Ubuntu, Arch Linux), Chromium OS, Windows CE, Symbian, QNX, entre otros. Al implementar un sistema de reconstrucción 3D en un sistema embebido se consigue reducción de costos y energía.

Las dificultades de realizar digitalización de sólidos en dispositivos con arquitecturas ARM son las enumeradas a continuación:

a) Obtención de imágenes por medio de una cámara digital.

- Velocidad de captura.
- Tamaño de captura (almacenamiento).

b) Creación del mapa de fase.

- Obtención de la fase.
- Unión de las fases.

c) Desdoblamiento de fase.

- Implementación del algoritmo.
- Eliminación de ruido.

d) Reconstrucción de un modelo 3D.

1.3 Alcances y Limitaciones

1.3.1 Alcances

- Dar seguimiento al tema de investigación de reconstrucción 3D en el área del conocimiento de procesamiento de imágenes del cuerpo académico “Cómputo Científico y Tecnológico” basado en proyección de Franjas.
- Desarrollar un software para interpretación y manejo de imágenes obtenidas con una cámara digital dentro de un procesador con arquitectura ARM.
- Realizar pruebas pertinentes en cada una de las etapas del proyecto así como pruebas finales al concluir el trabajo.

1.3.2 Limitaciones

En el transcurso del proyecto se hizo evidente la falta de un laboratorio específico para el tratamiento de imágenes.

1.4 Organización de la tesis

El capítulo 1 muestra una breve descripción del proyecto de investigación, abarcando temas como: objetivos, justificación, alcances y limitaciones.

Posteriormente en el capítulo 2 se presenta un marco teórico necesario para el entendimiento de múltiple nomenclatura, cómo: análisis de franjas, envolvimiento y desdoblamiento de fase, reconstrucción tridimensional, ARM, etc. También se mencionan trabajos anteriores relacionados al tema expuesto.

En la metodología expuesta en el capítulo 3 se descomponen y describen cada una de las etapas necesarias para la adquisición de imágenes tridimensionales utilizando PSP, así mismo se mencionan y describen distintos algoritmos utilizados a lo largo de la tesis.

Los resultados son expuestos y examinados a detalle en el capítulo 4, abarcando la simulación y las implementaciones tanto en una computadora personal como en una tarjeta de desarrollo.

Finalmente en el capítulo 5 se concluye con las aportaciones y observaciones perceptibles a lo largo del presente trabajo.

2. MARCO TEÓRICO

2.1 Reconstrucción tridimensional.

La reconstrucción tridimensional es el proceso por el cual se obtiene la representación de un objeto tomando como entradas una o varias vistas 2D, manteniendo las características físicas reales del objeto (dimensiones, volumen y forma) (Vilá, 2009).

La adquisición de información tridimensional se divide en dos grandes ramas: Por contacto y sin contacto.

El método por contacto cómo lo indica su nombre requiere del roce con la superficie del objeto a reconstruir, la resolución del objeto está directamente relacionada con la cantidad de puntos obtenidos; una de las mayores desventajas de usar el método por contacto es la falta de precisión en objetos blandos o deformables, además se requiere una cantidad de tiempo considerable para la obtención de puntos espaciales. A su vez el método se divide en técnicas destructivas y técnicas no destructivas. Como ejemplo de su aplicación actual se encuentran las máquinas de coordenadas CMM's (Coordinate Measuring Machines).

En el método sin contacto se implementan técnicas que no requieren el contacto con alguna de sus partes, se subdivide en técnicas transmisivas y reflectivas; las transmisivas irradian al objeto con alguna clase de radiación y las reflectivas reflejan ondas de luz o sonido aplicadas al objeto (Tadeo, 2008).

En la Figura 1 puede observarse una taxonomía de las técnicas de adquisición de información tridimensional.

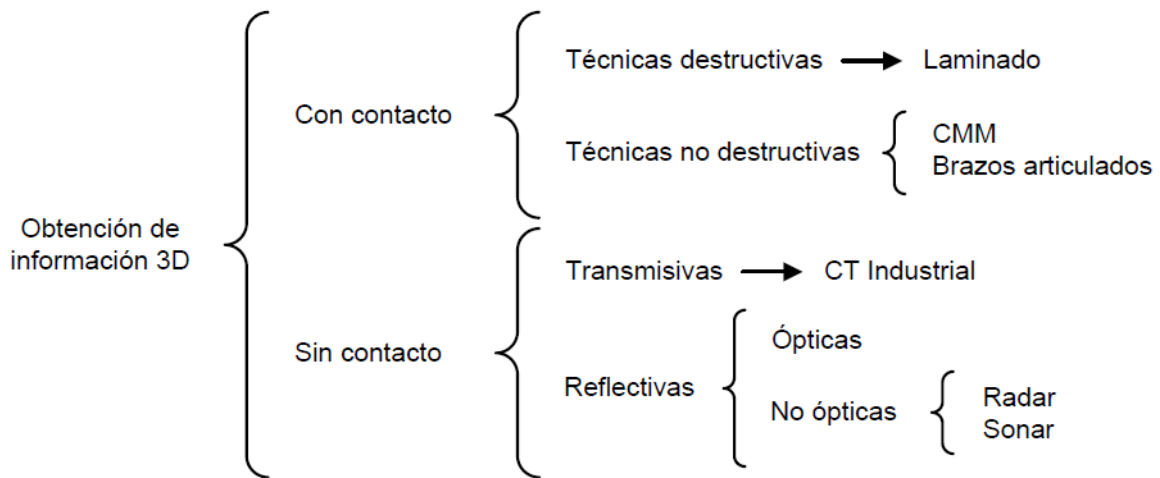


Figura 1. Técnicas de adquisición 3D (Molleda meré, 2008).



Figura 2. Técnicas ópticas (Molleda meré, 2008).

Dentro de las diversas técnicas ópticas el presente trabajo se encuentra en técnicas activas (aquellas en las cuales se regula la fuente de iluminación) específicamente luz estructurada. En la Figura 2 pueden observarse las distintas técnicas ópticas y su subdivisión.

El análisis de franjas es un grupo especial de técnicas de luz estructurada que utiliza patrones sinusoidales (también conocidos como patrones de franjas). Estos patrones tienen intensidades que varían continuamente punto a punto de una manera conocida, lo que permite obtener información píxel a píxel.

Jun-ichi Kato (2009) menciona que el análisis de franjas es el proceso de extraer los valores cuantitativos de patrones para propósitos metrológicos.

2.2 Análisis de proyección de franjas.

Dentro de las técnicas ópticas se encuentra la proyección de franjas que se implementa de la siguiente manera: en primera instancia son proyectados patrones definidos a un objeto, la forma del objeto crea distorsiones en el patrón, después las distorsiones se capturan con una cámara digital desde otro ángulo y finalmente las imágenes adquiridas son procesadas con alguna técnica de distorsión.

Existen múltiples patrones que pueden ser utilizados como: líneas verticales, líneas horizontales, líneas diagonales, cuadros, rombos entre otras. Existen dos tipos de análisis de franjas: un caso general se muestra en la Figura 3. la Figura 3(a) es una imagen que se interpreta como curvas de nivel por medio de un interferómetro, y en la Figura 3(b) se encuentra un patrón de franjas de alta frecuencia (convencionalmente llamado “*carrier fringes*”) que se proyecta a cierta dirección que lleva a su deformación espacial. Para cada tipo, existen técnicas de análisis adecuadas. El presente trabajo utiliza “*carrier fringes*” para visualizar la deformación en la superficie del objeto.

Para obtener la profundidad del objeto en la proyección de franjas se realiza una triangulación entre el ángulo del proyector, la cámara y el objeto a digitalizar (Merner et al, 2013), tal como se observa en la Figura 4.

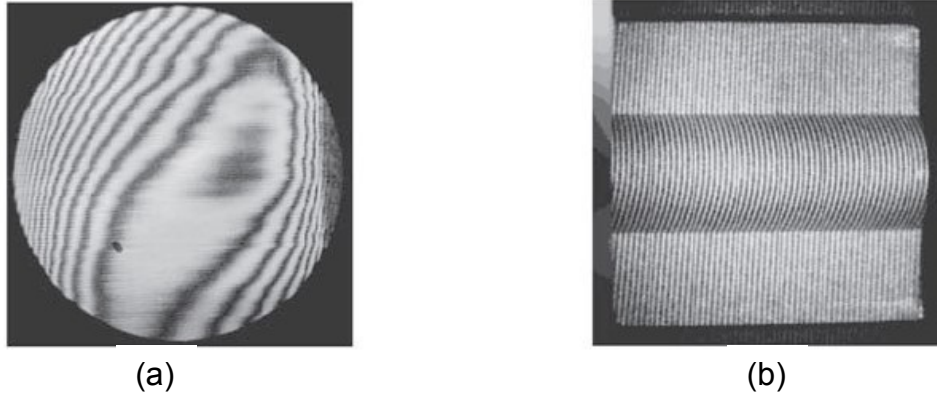


Figura 3. Imágenes de franjas. (a) franja de interferencia, contorno generado por el interferómetro de Michelson, (b) una imagen de rejilla deformada por una forma de la superficie de un objeto obtenido con el método de proyección de franjas (Kato, 2009).

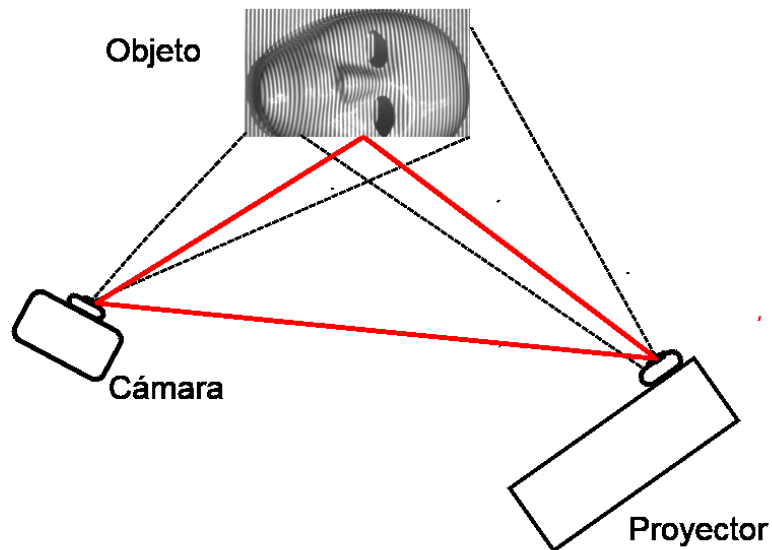


Figura 4. Triangulación entre cámara proyector y objeto.

En la reconstrucción tridimensional existen múltiples algoritmos como son: Perfilometría por Transformada de Fourier (FTP), Perfilometría por Transformada Wavelet (WTP), Perfilometría por Filtrado Espacial (SFP) o Perfilometría por Desplazamiento de Fase (PSP). Cada una de las técnicas aporta diferentes cualidades y desventajas con respecto a las otras.

Cuando se tiene solo una imagen de entrada, comúnmente se utiliza la transformada de Fourier como una modulación de fase espacial de las franjas portadoras con una frecuencia fundamental f_0 . Diversos algoritmos de demodulación de fase se pueden aplicar a esta franja portadora. La forma principal es el método FTP que extrae sólo el término de fase modulada a través del espectro filtrado, utilizando una o dos dimensiones FTP y su inversa. En las Figuras 5 y 6 se puede observar la técnica usada con una y dos dimensiones.

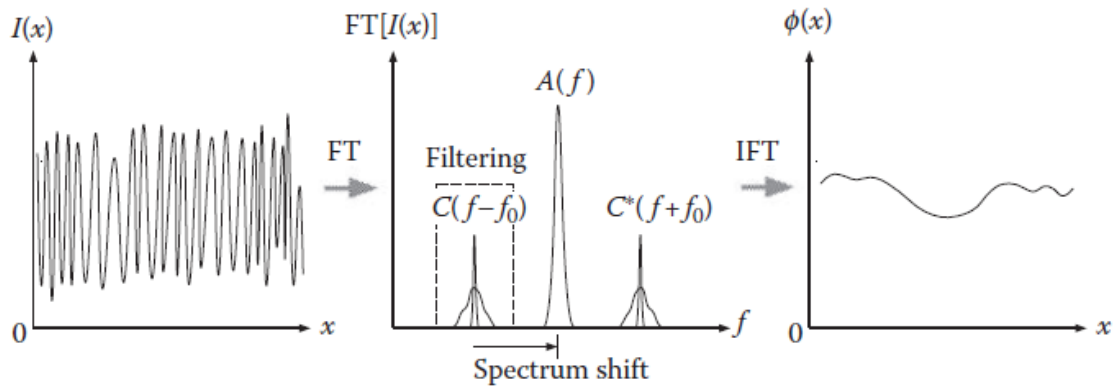


Figura 5. Transformada de Fourier utilizada en la reconstrucción 3D, se obtiene la fase modulada, se hace un cambio y filtro en el espacio espectral, y por último se aplica la inversa de la transformada para demodular la fase. (Kato, 2009).

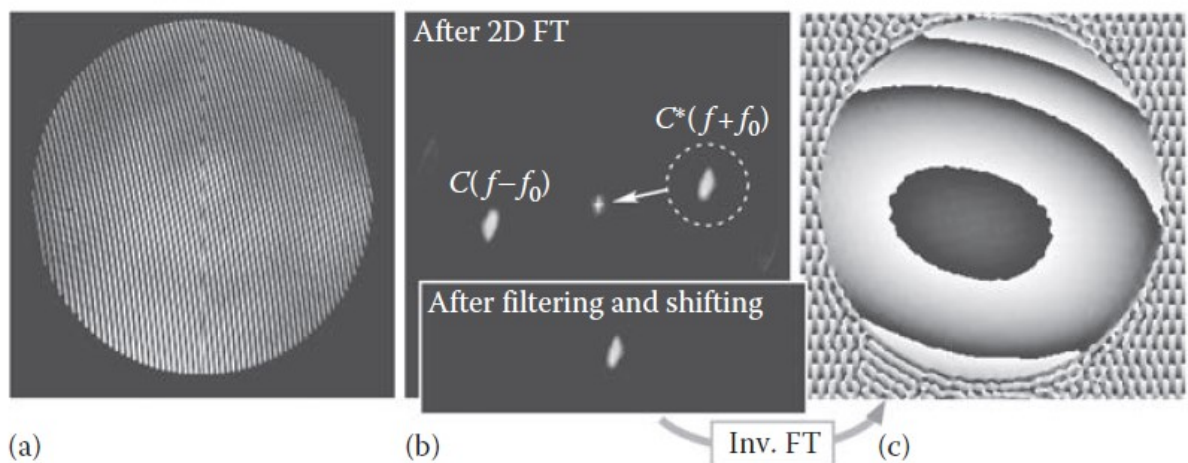


Figura 6. Extracción del mapa de fase utilizando transformada de Fourier 2D. (a) Patrones de franjas de entrada. (b) Resultado de la transformada de Fourier aplicada en un espectro 2D. (c) Extracción del mapa de fase resultado de aplicar la transformada inversa (Kato, 2009).

Perfilometría por Desplazamiento de Fase (Phase-Shifting Profilometry) o PSP es una técnica que emplea 3 o más patrones aplicados a la imagen desplazados en la fase. En la Figura 7 se pueden apreciar la estructura básica y un ejemplo del método. Las etapas del PSP se explican a continuación:

- **Adquisición de imágenes**

La adquisición de imágenes es la primera etapa del PSP y se encarga de calibrar y capturar las imágenes de los patrones que serán proyectados a la superficie del objeto.

- **Pre-procesamiento**

Las imágenes capturadas son mejoradas por uno o varios métodos. Por ejemplo; la ecualización y estiramiento de histogramas.

- **Mapa de fase**

Un mapa de fase contiene el perfil (conjunto de características propias del objeto) incluyendo el ángulo.

- **Desdoblamiento de fase**

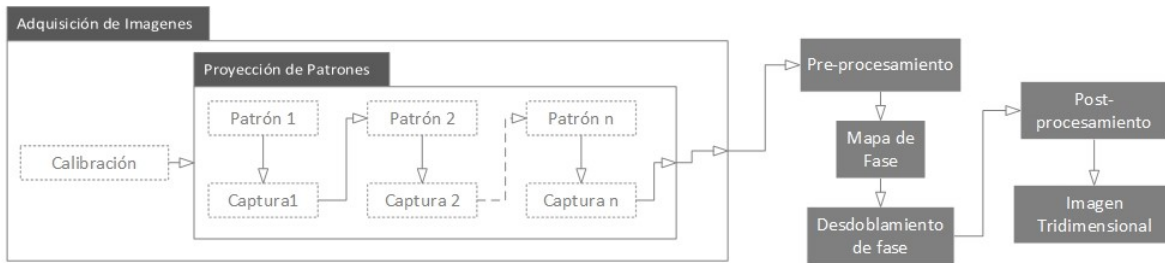
Proceso en el que se remueve la discontinuidad de 2π inherente en el mapa de fase.

- **Post-procesamiento**

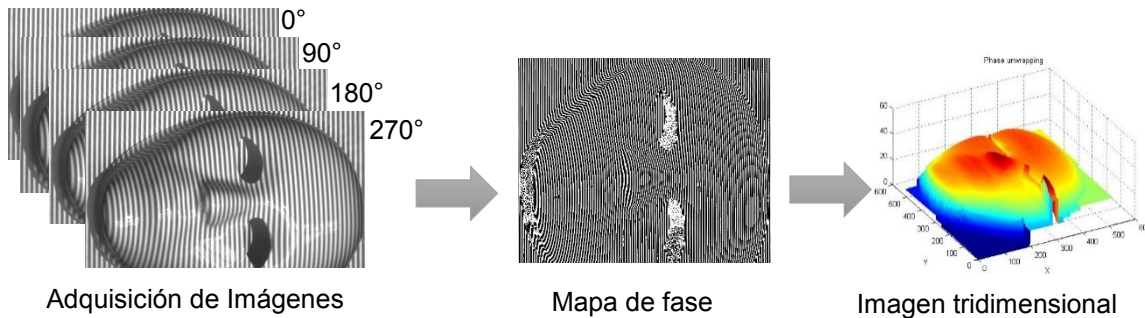
Técnicas de procesamiento de imágenes enfocadas a la mejora del objeto tridimensional una vez desdoblada la fase.

- **Imagen tridimensional**

Imagen que contiene 3 dimensiones.



(a)



(b)

Figura 7. Desplazamiento de fase. (a) Pasos del método desplazamiento de fase, (b) Ejemplo de digitalización de objetos utilizando desplazamiento de fase teniendo de entrada 4 imágenes desplazadas en la fase.

2.3 Procesadores ARM.

ARM (Advanced RISC Machine) es una arquitectura RISC (Reduced Instruction Set Computer) de 32 bits (hoy en día, también de 64 bits) desarrollada por ARM holdings, especialmente diseñada para embebidos de bajo costo y bajo consumo de energía. ARM ha sido licenciado por distintas compañías para su fabricación en las que destacan Apple Inc., Atmel, Broadcom, Freescale, Intel (a través de DEC), LG, Microsoft, NEC, Nintendo, Nokia, Nvidia, Sony, Samsung, Sharp y Texas Instruments.

Actualmente existen múltiples plataformas para el desarrollo de esta tecnología. En la Figura 8 se observan diferentes tarjetas de desarrollo con procesadores ARM y en la Tabla 1 sus características.

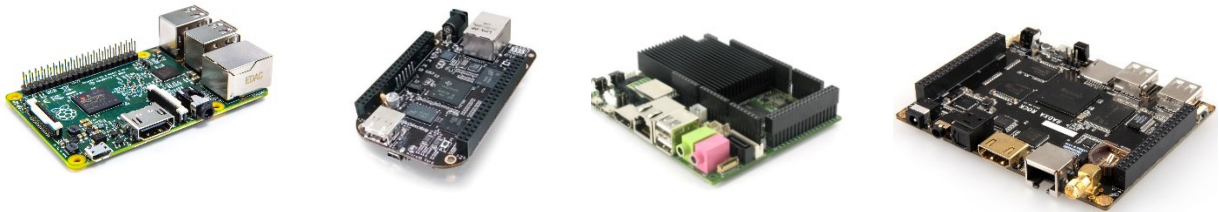


Figura 8. Diferentes tarjetas de desarrollo con procesadores ARM, Raspberry Pi 2B, BeagleBone Black, Udooc Quad y Radxa Rock Pro.

	Raspberry Pi 2 B	BeagleBone Black	UDOO Quad	Radxa Rock Pro
<i>Procesador</i>	A 900MHz quad-core ARM Cortex-A7 CPU	Sitara AM3359AZCZ100, a 1GH	Freescale i.MX 6 ARM Cortex-A9 Quad core 1GHz	ARM Cortex-A9 quad core 1.6Ghz
<i>RAM</i>	1GB RAM	512 MB	1GB	2GB DDR3 800Mhz
<i>USB</i>	4 x USB 2.0	1USB 2.0	2 USB	2 USB 2.0
<i>Vídeo</i>	Full HDMI	HDMI D type interface	HDMI and LVDS + Touch	HDMI 1.4 up to 1080p 60hz
<i>GPU</i>	VideoCore IV	SGX530 3d, 20M Polygons/S	GPU Vivante GC 2000 + Vivante GC 355 + Vivante GC 320	Mali400-mp4 533Mhz, OpenGL ES 2.0
<i>Precio*</i>	\$35.00	\$55.00	\$135.00	\$99.00

Tabla 1. Características principales de diferentes tarjetas de desarrollo con procesadores ARM, *Dólares americanos.

Debido a la amplia demanda diferentes compañías han creado tarjetas de desarrollo entre las que destacan Texas Instruments (Beagle) y Freescale (UDOO, Raspberry Pi) la comparación entre sus principales tarjetas puede observarse en la

Tabla 1. La tarjeta UDOO cuenta con mayores capacidades comenzando por los 4 núcleos, 1Gb de RAM y un mejor GPU. Actualmente los procesadores de gamma destacables son: Snapdragon serie 800 (de la compañía Qualcomm) que soportan 64-Bits y vídeo UHD, Samsung Exynos 5 Octa cuenta con 8 hilos simultáneos y el Tegra 4 de Nvidia con 5 núcleos.

Existen 2 grandes grupos de procesadores ARM los clásicos y los Cortex; al principio los procesadores clásicos eran nombrados ARM {x}{etiquetas} donde x es el número de versión de núcleo, posteriormente ARM {x}{y}{z}{etiquetas} donde “y” y “z” corresponden a la información de cache/MMU/MPU y tamaño de cache respectivamente por ejemplo el procesador ARM1156 utiliza la versión 11 del núcleo, el número 5 indica que contiene caché, buffer de escritura para la unidad de protección de memoria (MPU), y corrección de errores en memoria, por último el número 6 indica una estrecha memoria acoplada (tightly coupled Memory). La etiqueta nos informa del tipo de tecnología que soporta y puede ser D, E, F, I, J, K, T y –S; por ejemplo la J indica que soporta tecnología Jazelle (tecnología de aceleración de Java).

Por otra parte los procesadores Cortex se dividen en tres familias: Cortex–A, Cortex-R y Cortex-M. La familia Cortex-A es la familia de las computadoras, son diseñados por completo como computadoras funcionales, tiene la capacidad de ejecutar directamente sistemas operativos complejos, es usado en teléfonos móviles, tabletas y laptops. La familia Cortex-R es la familia que reacciona más rápido, trabajan en tiempo real, son menos poderosos que los Cortex-A pero reaccionan más rápido a estímulos externos, esta familia se usa principalmente en dispositivos médicos, sistemas automovilísticos y dispositivos controlables de bajo nivel. La familia Cortex-M es la familia de ultra baja potencia, opera con menor rendimiento que los procesadores Cortex-A y Cortex-R pero puede ejecutarse por encima de los 100 MHz. Es incorporado en microcontroladores con múltiples líneas de entrada y salida. Se utiliza en robótica y pequeños productos electrónicos de consumo (Langbridge, 2013).

2.3.1 Reconstrucción 3D en sistemas embebidos.

Existen diversas aplicaciones que requieren un escaneo tridimensional de bajo costo, preciso y de rápida solución, por ello diversos autores han investigado formas de integrar un escaneo tridimensional en un sistema embebido. Giryes et al. (2008) implementan un escáner tridimensional usando como base un DSP (Digital Signal Processor) auxiliado por un procesador ARM, su aplicación está enfocada sistemas biométricos.

Hadjitheophanous et al. (2010) proponen una reconstrucción tridimensional usando cámaras estereoscópicas cuyo algoritmo es implementado en un FPGA, teniendo avances en el campo de profundidad. Igualmente Wang et al. (2008) utilizan un FPGA para realizar reconstrucción tridimensional, en este caso para visualizar el entorno de un robot móvil, se utilizan técnicas de línea laser para recrear los entornos.

3. METODOLOGÍA

El presente trabajo se dividió en tres grandes etapas: la simulación, la implementación de los algoritmos en una PC y la implementación en un dispositivo con procesador ARM. En cada una de las etapas se llevaron a cabo los pasos descritos en la Figura 7(a) exceptuando la simulación en la que no se requiere de un pre- procesamiento. En la presente sección se detalla cada uno de los pasos que se requiere para obtener una imagen tridimensional utilizando la técnica PSP.

3.1 Perfilometría por desplazamiento de fase o PSP

Como se mencionó en el capítulo anterior la Perfilometría por Desplazamiento de Fase (PSP, por sus siglas en ingles Phase-Shifting Profilometry) es una técnica de análisis de franjas que emplea 3 o más patrones aplicados a la imagen, desplazados en la fase para obtener información tridimensional de la superficie de un objeto.

Los algoritmos de PSP son ampliamente utilizados en mediciones ópticas debido a sus numerosas ventajas, “point – by- point measurement” (lo que permite la resolución espacial a nivel de píxel-cámara), menor sensibilidad a las variaciones de reflectividad de la superficie, facilitando la medición de objetos muy complejos con fuertes variaciones de textura, y menor sensibilidad a la luz ambiental (Peng et al, 2005; Zhang et al, 2010; Ekstrand et al, 2013).

Existen diversos métodos para el desplazamiento de fase en los que destacan; “three-step”, “four-step” y “double three-step”. Estos métodos difieren en la cantidad de imágenes de entrada que se tendrán, el “three-step” tendrá 3 imágenes, el “four-step” tendrá 4 y así consecutivamente. Para obtener la información de un objeto es necesario encontrar la fase, la fase contendrá la información necesaria para su reconstrucción tridimensional.

En general, el desplazamiento de fase de “N-step” con fases iguales se puede describir cómo:

$$I_n(x, y) = I'(x, y) + I''(x, y)\cos(\varphi + 2\pi n/N) \quad (1)$$

Donde $I'(x, y)$ es la intensidad promedio, $I''(x, y)$ es la modulación de la intensidad, $\varphi(x, y)$ es la fase que hay que resolver para $n = 1, 2, 3, \dots, N$. resolviendo esta ecuación se obtiene:

$$\varphi(x, y) = \tan^{-1} \frac{\sum_{n=1}^N I_n(x, y) \sin(\frac{2\pi n}{N})}{\sum_{n=1}^N I_n(x, y) \cos(\frac{2\pi n}{N})} \quad (2)$$

3.1.1 Algoritmo three-step

El algoritmo “three-step” contiene el menor número de imágenes a procesar (Li et al, 2013). En la práctica los ángulos de desplazamiento suelen ser de 120° , 0° y -120° (Guzhov et al, 2012), cómo se observa en la Figura 9.

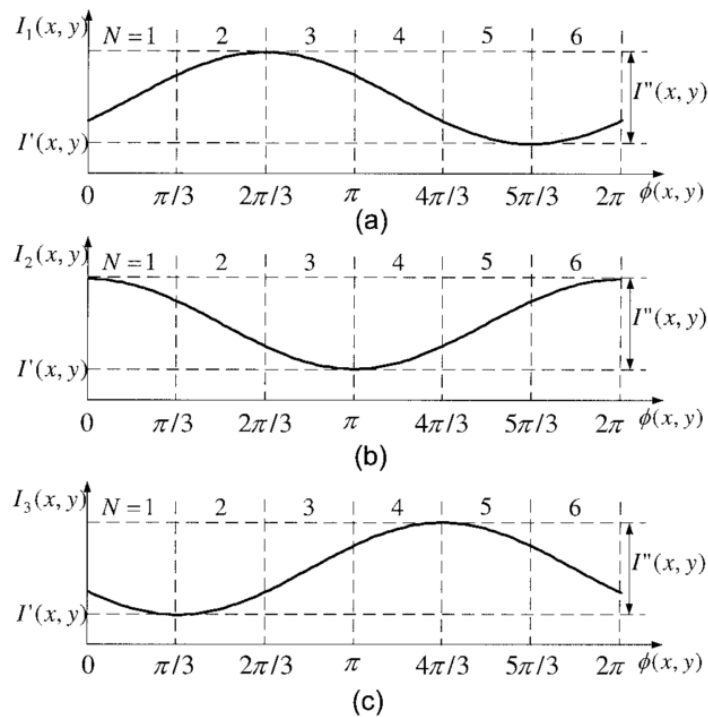


Figura 9. Cruce de sección de patrones de franja sinusoidales. (a)- 120° , (b) 0° y (c) 120° . (Huang & Zhang, 2006).

PSP three-step se puede describir de la siguiente manera:

$$I_1(x, y) = I'(x, y) + I''(x, y) \cos[\phi(x, y)] \quad (3)$$

$$I_2(x, y) = I'(x, y) + I''(x, y) \cos[\phi(x, y) - \alpha] \quad (4)$$

$$I_3(x, y) = I'(x, y) + I''(x, y) \cos[\phi(x, y) + \alpha] \quad (5)$$

$$\phi(x, y) = \tan^{-1} \left(\sqrt{3} \frac{I_3(x, y) - I_2(x, y)}{2I_1(x, y) - I_2(x, y) - I_3(x, y)} \right) \quad (6)$$

Donde $I'(x, y)$ es la intensidad promedio, $I''(x, y)$ es la modulación de intensidad, $\phi(x, y)$ es la fase a resolver. En nuestro caso consideramos $\alpha = 2\pi/3$.

3.1.1.1 Algoritmo de Song Zhang

El Algoritmo de Song Zhang es una variante del algoritmo three-step que reduce 4 veces la velocidad de procesamiento comparado con el método tradicional.

El algoritmo elimina el arco tangente en el desdoblamiento de fase obteniendo la fase de la siguiente forma:

$$\phi'(x, y) = \frac{\pi}{3} \left[2 \times \text{round} \left(\frac{N-1}{2} \right) + (-1)^{N-1} r(x, y) \right] \quad (7)$$

Donde N es la sección en la que se dividen las franjas sinusoidales (en donde no hay un cruce). Como puede observarse en la Figura 9 utilizando un método de 3 pasos y ángulos de 0° , 120° y -120° existen 6 diferentes Secciones (Zhang & Huang, 2006).

$r(x, y)$ es la razón de intensidad proporcionada por los valores mínimos, medios y máximos de la intensidad de cada región de la siguiente manera:

$$r(x, y) = \frac{I_{med}(x, y) - I_{min}(x, y)}{I_{max}(x, y) - I_{min}(x, y)} \quad (8)$$

Aunque el método de Song Zhang reduce considerablemente el tiempo de procesamiento se crea un error inherente en el método que suele ser corregido por medio de una LUT (“Look Up Table”).

3.1.2 Algoritmo four-step

El algoritmo four-step es muy similar al three-step con la diferencia de que se usan 4 patrones de imágenes desplazados en fase en lugar de tres.

PSP four-step se puede describir de la siguiente manera:

$$I_1(x, y) = I'(x, y) + I''(x, y) \cos[\phi(x, y) + 3\alpha] \quad (9)$$

$$I_2(x, y) = I'(x, y) - I''(x, y) \sin[\phi(x, y) + 2\alpha] \quad (10)$$

$$I_3(x, y) = I'(x, y) - I''(x, y) \cos[\phi(x, y) + \alpha] \quad (11)$$

$$I_4(x, y) = I'(x, y) + I''(x, y) \sin[\phi(x, y)] \quad (12)$$

$$\phi(x, y) = \tan^{-1} \left(\frac{I_4(x, y) - I_2(x, y)}{I_1(x, y) - I_3(x, y)} \right) \quad (13)$$

Donde $I'(x, y)$ es la intensidad promedio, $I''(x, y)$ es la modulación de intensidad, $\phi(x, y)$ es la fase a resolver. En nuestro caso consideramos $\alpha = \pi/2$.

3.2 Envolvimiento y desdoblamiento de fase

Existen 2 etapas fundamentales para el tratamiento de la fase, el envolvimiento de fase y el desdoblamiento de fase. El envolvimiento de fase es el proceso que determina los valores de la fase en un rango de 0 a 2π . El desdoblamiento de fase es el proceso en el que se remueve la discontinuidad de 2π para generar un mapa de fase suavizado (Huang & Zhang, 2006).

Cómo se muestra en la Figura 10 la fase obtenida por lo general supera los valores de amplitud de $-\pi$ a π por lo cual se envuelve la fase y así se obtienen valores de $-\pi$ a π .

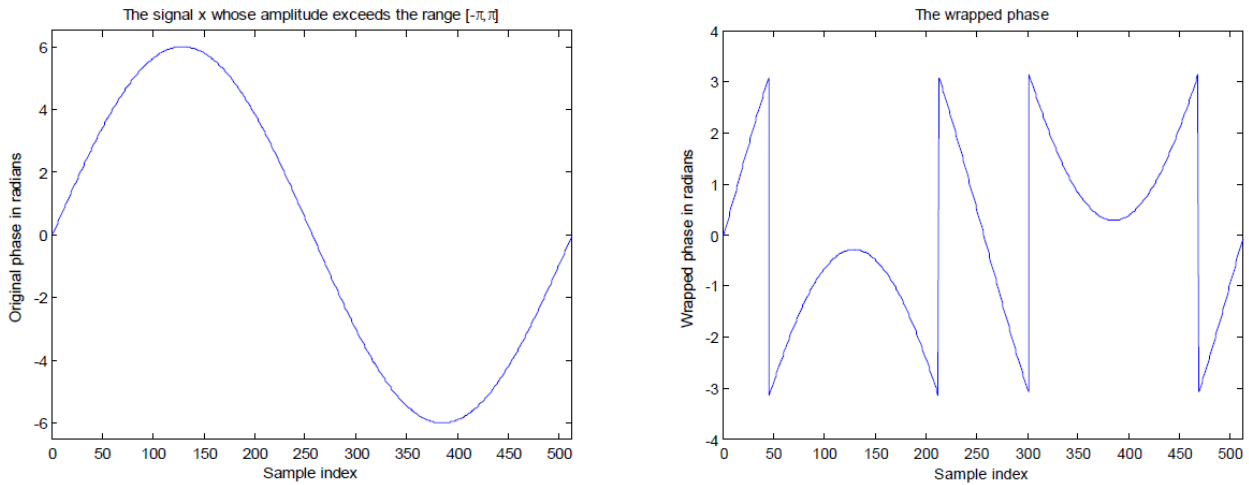


Figura 10. Fase continua y fase envuelta (Gdeisat et al, 2012).

Se puede expresar matemáticamente el envoltamiento de fase como:

$$x_W(n) = W[x(n)] \quad (14)$$

Donde $x(n)$ es la fase continua original, $W []$ es el la operación de fase envuelta y $x_W(n)$ es la fase envuelta.

La función de fase $\varphi(x, y)$ presenta discontinuidades con saltos de 2π para variaciones mayores de 2π . Estas discontinuidades se pueden corregir sumando o restando 2π dependiendo de si el salto de fase va de π a $-\pi$ o viceversa. Existen algoritmos digitales (phase-unwrapping) que se emplean para corregir las discontinuidades de fase. Los algoritmos de desdoblamiento de fase aplicados al análisis de franjas se pueden categorizar en 2 grandes grupos. Los temporales (temporal phase unwrapping) y los espaciales (spatial phase unwrapping). Los métodos temporales son generalmente eficaces y robustos, pero requieren “multi-frames” de fase envuelta a lo largo de la dimensión del tiempo; los principales algoritmos de desdoblamiento de fase temporal son “dynamic unwrapping method”, “multifrequency method” y “heterodyne method” por otra parte los métodos espaciales tienen menos restricciones, pero es un reto hacer frente a las regiones disjuntas y discontinuidades inmersas en la fase; los principales algoritmos de esta

rama son: “Goldstein’s method”, “quality-guided”, “Flynn’s method” y “minimum Lp-norm method”. (Zhao, et al, 2011).

En el presente trabajo se estudiaron los principales algoritmos de desdoblamiento de fase, los cuales son descritos a continuación.

3.2.1 Simple unwrap

El “Simple unwrap” (desdoblamiento simple) corrige los ángulos de fase en un vector añadiendo múltiplos de $\pm 2\pi$ cuando existen saltos absolutos entre elementos consecutivos (Matlab, 2015).

3.2.2 Itoh tradicional e inverso

El algoritmo de Itoh tradicional desenvuelve las filas una por una para verificar la discontinuidad entre los valores de la fila, seguido por un desenrollado en las columnas (igual una por una). El algoritmo inverso primero desenvuelve las columnas y después las filas (Gdeisat et al, 2012).

3.2.3 Graph Cuts

El algoritmo implementado para el desdoblamiento de la fase aplicando “Graph Cuts” (cortes gráficos) utiliza cálculos “max-flow/min-cut” descritos por José M. Bioucas-Dia y Gonçalo Valadão en 2007.

3.2.4 Algoritmo “Quality-guided”

El algoritmo “Quality-guided” se compone de 2 etapas fundamentales: crear el mapa de calidad y la estrategia de guía. El mapa de calidad es una imagen basada en el mapa de fase y le asigna pesos a cada píxel según su posibilidad de desdoblarse correctamente, por otra parte la estrategia de guía es el método a implementar para decidir el orden en el que cada píxel se desdoblará.

Para representar la calidad de un punto se utiliza la siguiente ecuación:

$$\gamma(x, y) = I''(x, y)/I'(x, y) \quad (15)$$

Siendo 1 el punto con mejor calidad. En el caso particular de “three-step” la ecuación que determina el primer mapa de calidad donde se elimina el fondo comúnmente es la siguiente:

$$\gamma(x, y) = \frac{\sqrt{3(I_1 - I_3)^2 + (2I_2 - I_1 - I_3)^2}}{I_1 + I_2 + I_3} \quad (16)$$

Otro mapa de calidad con el cual se ponderaran los valores de la imagen y ayudará para determinar la trayectoria del desdoblamiento de fase se obtiene al aplicar la siguiente ecuación:

$$Q(i, j) = \max\{\Delta_{i,j}^x, \Delta_{i,j}^y\} \text{ y } Q(i, j) \in [0,1] \quad (17)$$

Donde $\Delta_{i,j}^x$ y $\Delta_{i,j}^y$ son los máximos valores de las derivadas parciales en la fase en la dirección de “x” y “y” respectivamente. (zhang et al, 2007).

El algoritmo más sencillo para implementar una estrategia guiada una vez que se obtuvo el mapa de calidad es la siguiente:

Paso 1: Encontrar el píxel base con la calidad más alta dentro del mapa de calidad y agregarlo a una lista “adjoin”.

Paso 2: Encontrar el píxel con la calidad más alta de la lista “adjoin”, desenvolverlo, y sacarlo de la lista “adjoin”. Agregar a la lista “adjoin” los vecinos no procesados.

Paso 3: Repita el paso 2 hasta que la lista “adjoin” está vacía. (Zhao, et al, 2011).

Cabe aclarar que después de años de investigación de “quality- guided” existen gran cantidad de formas para crear el mapa de calidad y la estrategia guiada.

Una vez obtenido el mapa de fase, se elimina el ruido y se obtiene la imagen tridimensional, el sistema completo puede observarse en la Figura 11.

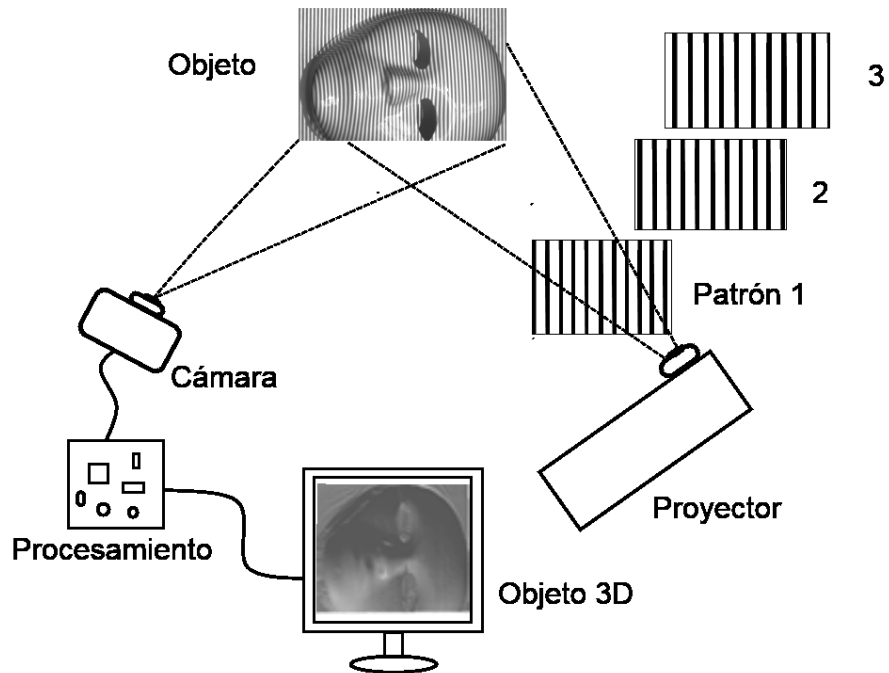


Figura 11. Sistema completo. Objeto real, cámara, proyección de 3 franjas desplazadas en la fase y proyectadas al objeto real, dispositivo de procesamiento y despliegue de la imagen tridimensional.

3.3 Adquisición de imágenes

Como se mencionó anteriormente el método PSP requiere de la adquisición de imágenes (de un objeto con patrones proyectados en él) a partir una cámara digital. Para poder tener mayor fidelidad, las imágenes deben estar alineadas a la cámara; debido a que el ángulo de captura y de proyección es diferente, una calibración es necesaria. El sistema de captura implementado en este trabajo puede observarse en la Figura 12.



Figura 12. Sistema de captura. Del lado izquierdo puede observarse el proyector, del lado derecho la cámara de adquisición y en el fondo el objeto a digitalizar.

3.3.1 Calibración de las imágenes

Si las imágenes capturadas no son calibradas o existen sombras en la imagen se obtienen errores en el mapa de fase tal como se observa en la Figura 13. Para corregir los errores se diseñó un sistema de calibración que alinea las franjas en un plano que utiliza un patrón de referencia, el patrón puede observarse en la figura 14.

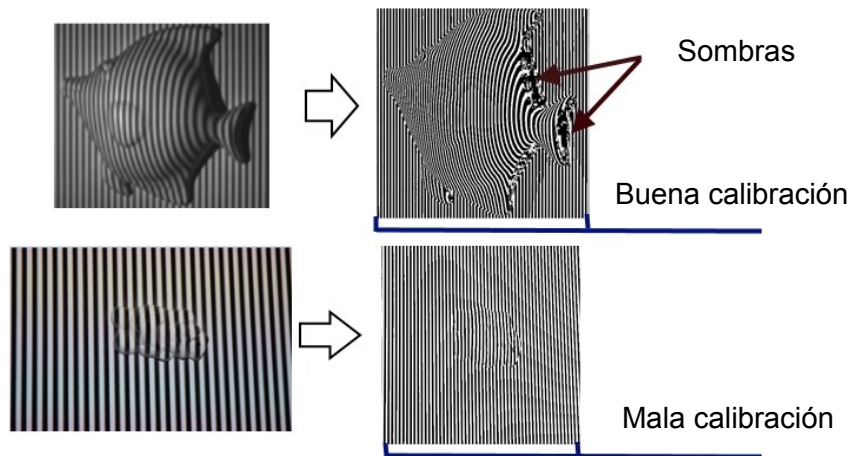


Figura 13. Ejemplos de sombras en las imágenes y el resultado de una buena y mala calibración.

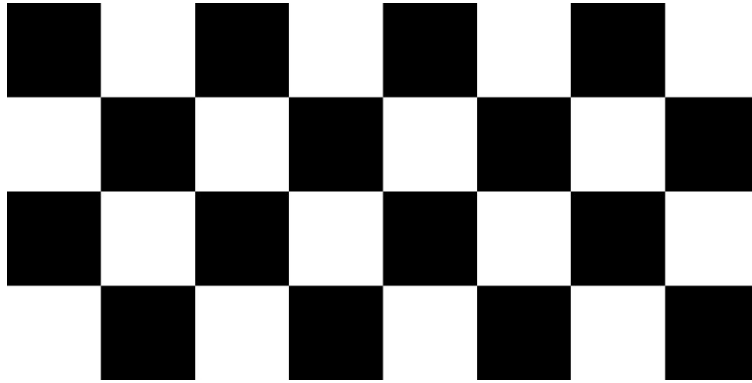


Figura 14. Patrón de referencia utilizado para calibrar las imágenes tomadas con la cámara digital.

Primero se capturan las imágenes sin calibrar incluyendo el patrón de referencia proyectado al plano en donde se pondrán los objetos a digitalizar, un ejemplo se observa en la Figura 15. Después se binariza el patrón proyectado y se encuentran los puntos de coincidencia con el patrón original como se observa en la Figura 16. Con estos puntos se crea una estimación geométrica del plano y la diferencia con el plano original, después se realiza una transformación geométrica del plano sin calibrar al plano geométrico y se aplica la misma transformación al resto de las imágenes tal como se observa en la Figura 17. Por último se recortan las imágenes en la zona de interés el resultado se observa en la Figura 18.

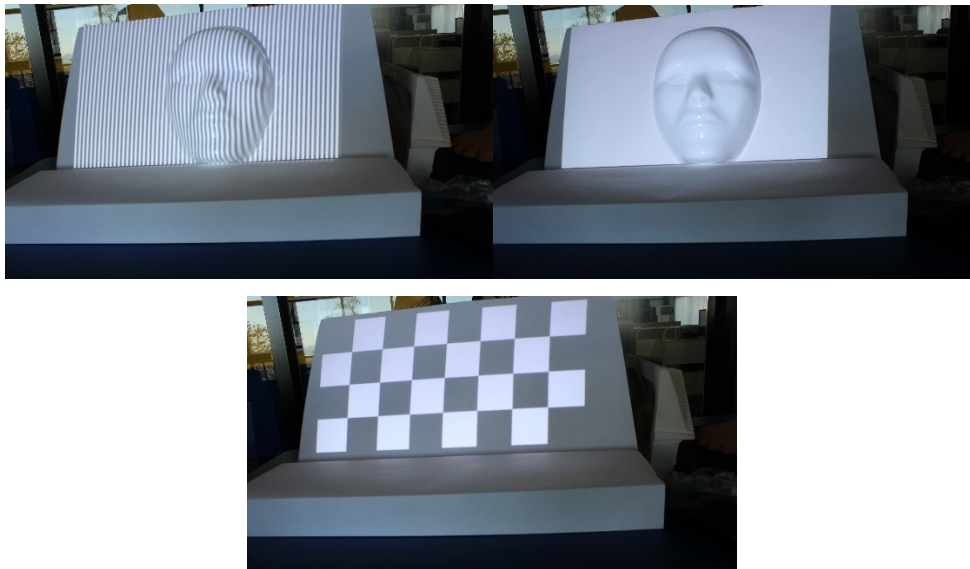


Figura 15. Ejemplos de imágenes sin calibrar.

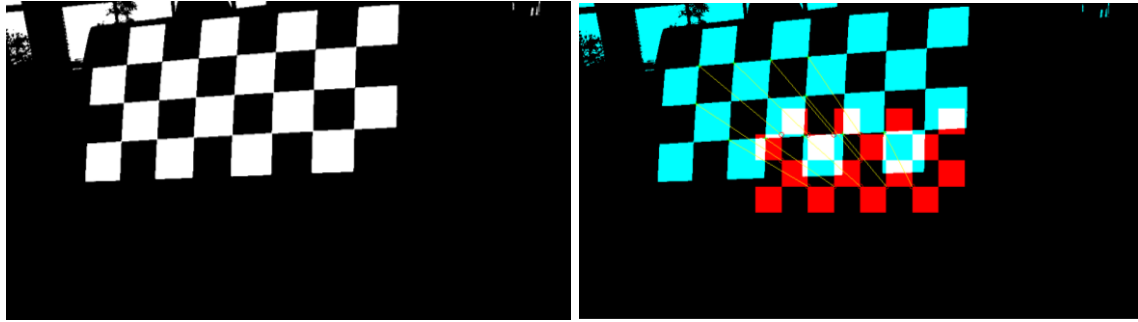


Figura 16. Binarización del patrón proyectado al plano y relación de puntos que coinciden con el patrón original.

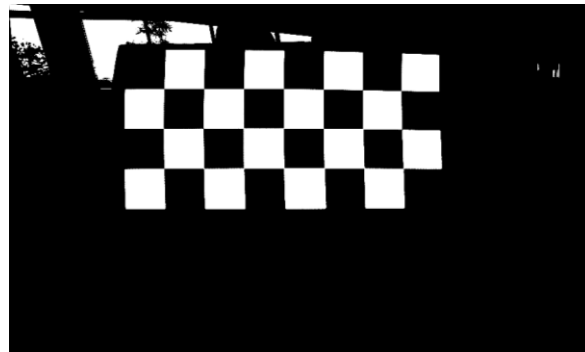
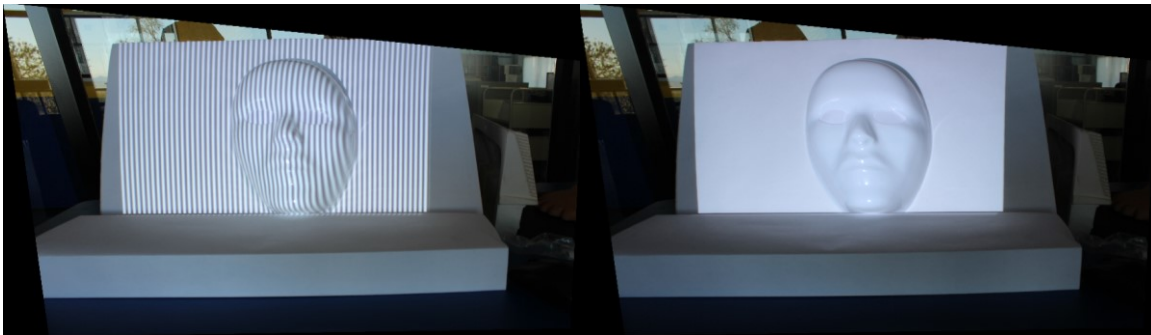


Figura 17. Imágenes calibradas.



Figura 18. Resultado de la calibración y recorte de la zona de interés.

3.4 Pre-procesamiento de la imagen

Para mejorar el rendimiento del algoritmo se realizó un pre-procesamiento a cada imagen con un conjunto de técnicas diferentes que faciliten la adquisición de información tridimensional. Lo primero que se realiza es una conversión a escala de grises reduciendo el procesamiento a un sólo canal, dado que se tienen valores enteros en la imagen de entrada se prosigue a normalizar la imagen en flotantes de 0.0 a 1.0 para tener mayor precisión en las operaciones.

Después de normalizar la imagen se realiza una ecualización del histograma (representación gráfica de una distribución de frecuencias) con la finalidad de reducir el brillo provocado por el proyector y repartir uniformemente el rango de grises. Un ejemplo de ecualización del histograma de observa en la Figura 19.

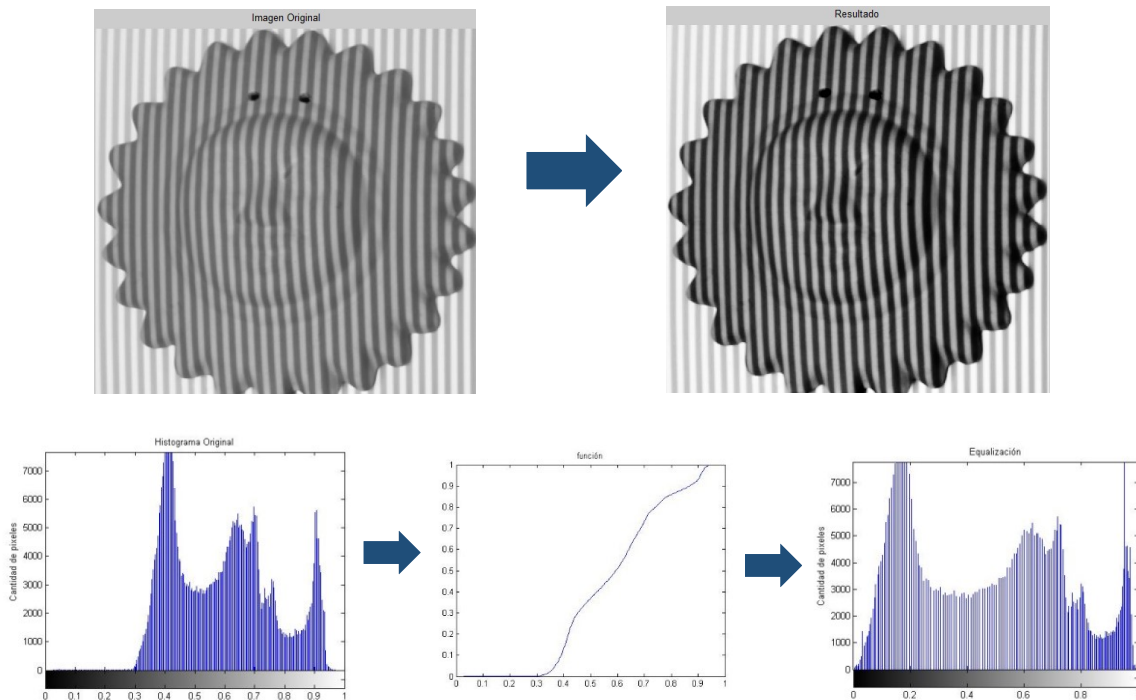


Figura 19. Ejemplo de ecualización del histograma donde se aplica una función para repartir uniformemente el histograma.

Por último si se desea resaltar aún más las franjas se puede hacer un cambio de gamma con la función:

$$gamma = 1 * (a/1)^{(1/vd)} \quad (18)$$

Donde vd es el valor deseado, si el valor es igual a 1 la imagen será la original, mientras el valor sea mayor a uno la imagen se aclarará y mientras el valor se aproxime a cero la imagen se oscurecerá.

3.5 Post-procesamiento

Debido a que el desdoblamiento de fase es un problema NP no existe una solución exacta que funcione en todos los casos u objetos, en muchas ocasiones persiste una discontinuidad en la fase debido a diversas causas como malas condiciones de luz, mala alineación en las franjas, efecto de sombras en las imágenes etc. Por ello un post-procesamiento es requerido para la mejora de las imágenes tridimensionales. Para solucionar el problema de la discontinuidad persistente en la imagen diferente a 2π fueron propuestos e implementados 2 algoritmos. El primer algoritmo encuentra las secciones donde ocurren las discontinuidades a través de una línea de referencia, si una discontinuidad es encontrada se almacena en una matriz (con la posición y los valores de los píxeles entre los cuales ocurre la discontinuidad), el segundo algoritmo corrige el error de discontinuidad sumando o restando valores en la sección de la línea de referencia donde se encontró la discontinuidad y propagando esa compensación al resto de la imagen. Diversos autores proponen soluciones enfocadas a modificar el desdoblamiento de fase, un ejemplo claro es la propuesta en el 2013 por Ke Chen Jiangtao Xi y Yanguang Yu que modifican el algoritmo tradicional de quality guided para eliminar estas discontinuidades. La descripción de ambos algoritmos puede observarse a continuación.

Algoritmo 1: Encuentra las secciones

Entrada w_i = imagen de la fase desenvuelta ($M \times N$)

```
1: // Se define la línea de referencia
2: línea =  $w_i$  (número de fila que atraviese más secciones discontinuas, :)
3: número de secciones = 0
4: for  $i = 2 \rightarrow N$ 
5:     píxel actual = línea ( $i$ )
6:     píxel previo = línea ( $i-1$ )
7:     discontinuidad = píxel actual - píxel previo
8:     if discontinuidad > umbral
9:         número de secciones ++
10: // valores en los bordes son almacenados
11:     valor de sección (número de secciones, 1) =  $i$ 
12:     valor de sección (número de secciones, 2) =  $i - 1$ 
13:     valor de sección (número de secciones, 3) = discontinuidad
14:     end if
15: end for
16: valor de sección (número de secciones + 1, 2) =  $N$ ;
```

Algoritmo 2: Eliminar la discontinuidad

Entrada w_i = imagen de la fase desenvuelta ($M \times N$), *línea* = w_i ($M/2$, :), *valor de sección*(:,1) = *píxel previo a la discontinuidad*, *valor de sección* (:,2) = *siguiente píxel a la discontinuidad*, *valor de sección* (:,3) = *valor de la discontinuidad*

```
1:  $e$  = discontinuidad a reparar = 0
2: if número de secciones > 0
3:     for  $j = 1 \rightarrow$  número de secciones
4:          $v1$  = valor de sección ( $j, 1$ )
5:          $v2$  = valor de sección ( $j, 2$ )
6:          $e$  = valor de sección ( $j, 3$ )
7:         for  $l = 1 \rightarrow M$ 
8:             // linC = línea a corregir
9:             linC =  $w_i$  ( $l$ , :)
10:            for  $k = v1 \rightarrow v2$ 
11:                linC( $k$ ) = linC( $k$ ) -  $e$ 
12:                 $w_i$  ( $l$ , :) = linC
13:            end for
14:        end for
15:    end for
16: end if
```

4. RESULTADOS

4.1 Simulación

Para evaluar y calcular el error del proceso de digitalización usando PSP se utilizaron imágenes de profundidad generadas por computadora. Las imágenes de profundidad o intensidad son imágenes a escala de grises que a través de una graduación en el valor del píxel otorgan datos tridimensionales. Mientras más cercano se encuentre el objeto más clara será el valor del píxel y mientras más alejado se encuentre más oscuro estará el píxel. Varios ejemplos se ilustran en la Figura 20.

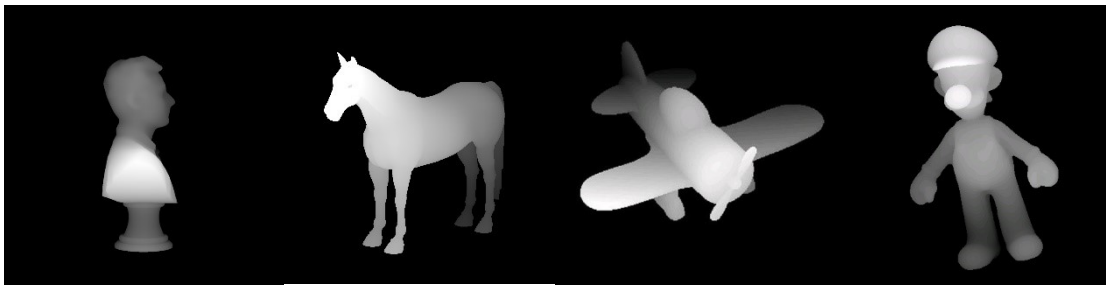


Figura 20. Ejemplos de imágenes de profundidad generados por computadora.

La imagen de profundidad pasa por el mismo proceso que una imagen de un objeto real sin la etapa de pre-procesamiento tal y como se observa en el Diagrama1 donde se aprecia el tratamiento de la imagen en la simulación y con objetos reales. Un ejemplo de tratamiento de imágenes de profundidad puede observarse en la Figura 21 donde a una imagen de profundidad de un cubo se le proyectan 3 patrones de franjas diferentes y se obtiene la reconstrucción tridimensional del cubo.

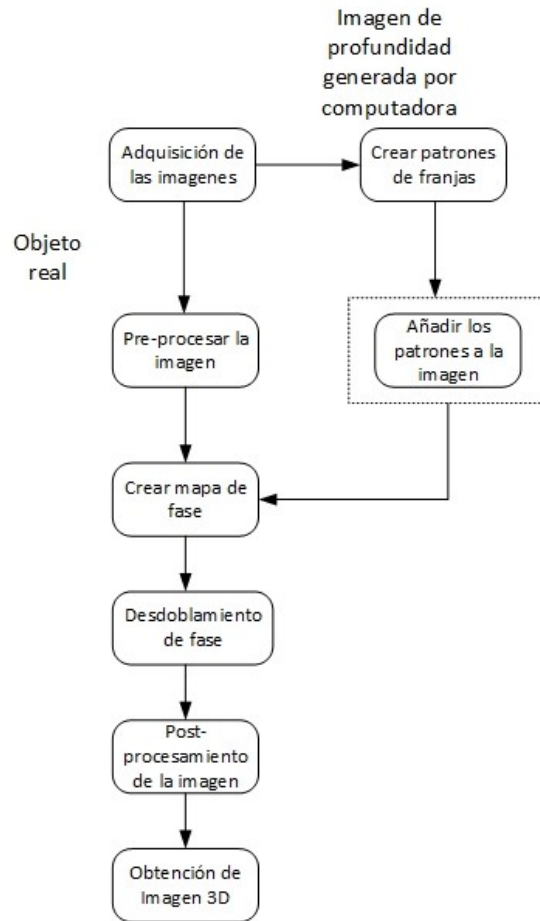


Diagrama 1. Diagrama en el que se muestra la diferencia entre la simulación y el procesamiento de una imagen de un objeto real.

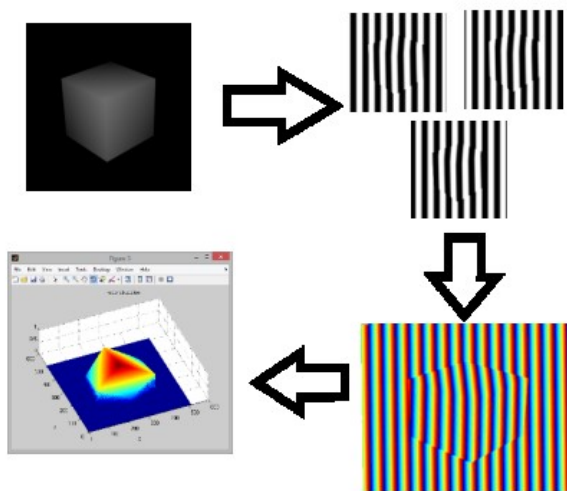


Figura 21. Ejemplo de simulación del proceso por medio de imágenes de profundidad.

Una imagen de profundidad fácilmente puede ser representada en un espacio con 3 dimensiones (se tienen las coordenadas “x” y “y” de la imagen y los valores de “z” están en el rango de 0-1 siendo 1 el color blanco y 0 el color negro) esto permite calcular el error del método propuesto, para ello se utiliza la diferencia (píxel por píxel) entre la imagen reconstruida y la representación tridimensional de la imagen de profundidad. En la Figura 22 se observa un ejemplo de imagen de profundidad (a), su representación tridimensional (b), la reconstrucción usando PSP (c) y la representación tridimensional del error (d).

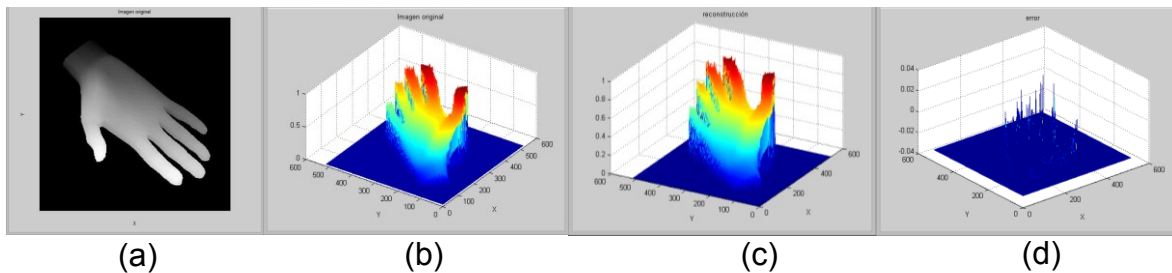


Figura 22. (a)Ejemplo de imagen de profundidad, (b) representación tridimensional de la imagen de profundidad, (c) Reconstrucción usando PSP, (d) Representación tridimensional del error.

Más de 30 imágenes de profundidad fueron evaluadas usando la técnica “three-step” y “four-step” (descritas en el capítulo 3) para evaluar el desempeño y la diferencia entre ambas, al principio de la investigación se usaron imágenes con superficies suaves pero la diferencia entre los métodos no era apreciable (pues son imágenes ideales), por lo que se decidió añadirles ruido cómo se muestra en la figura 23 pero aún la diferencia no era relevante, al final se utilizaron imágenes con bordes abruptos y alto contraste tal como se muestra en la Figura 24.

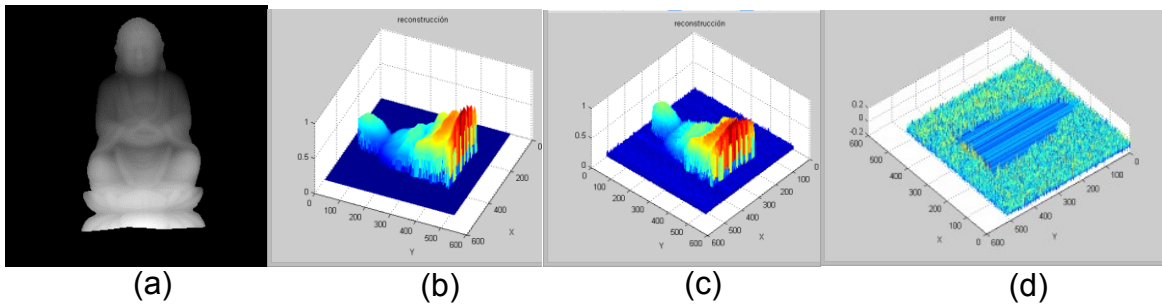


Figura 23. (a) Ejemplo de imagen de profundidad, (b) Reconstrucción tridimensional usando PSP sin ruido, (c) Reconstrucción tridimensional usando PSP con ruido, (d) Representación del error de imagen de profundidad con ruido.



Figura 24. Imagen de profundidad con bordes abruptos y alto contraste.

La computadora en la que se ejecutó la simulación es una Lenovo Y510P con las propiedades descritas en la tabla 2 y los programas fueron implementados en Matlab R2013b.

Componentes	Especificaciones
Procesador	Intel® Core™ i7-4700MQ 2.4GHz
Memoria RAM	DDR3 de 8 GB
Sistema Operativo	Windows 8.1 64 bits

Tabla 2. Especificaciones técnicas de la computadora usada en la investigación.

Los resultados gráficos de la evaluación de las técnicas “three-step” y “four-step” evaluados con los algoritmos de desdoblamiento de fase: “Simple unwrap”, “Itoh tradicional e inverso” y “Graph Cuts” pueden observarse en la Figura 25. Los resultados cuantitativos puede observarse en la tabla 3 y 4 donde se evaluó el error máximo, el error mínimo, el error promedio y el tiempo de ejecución.

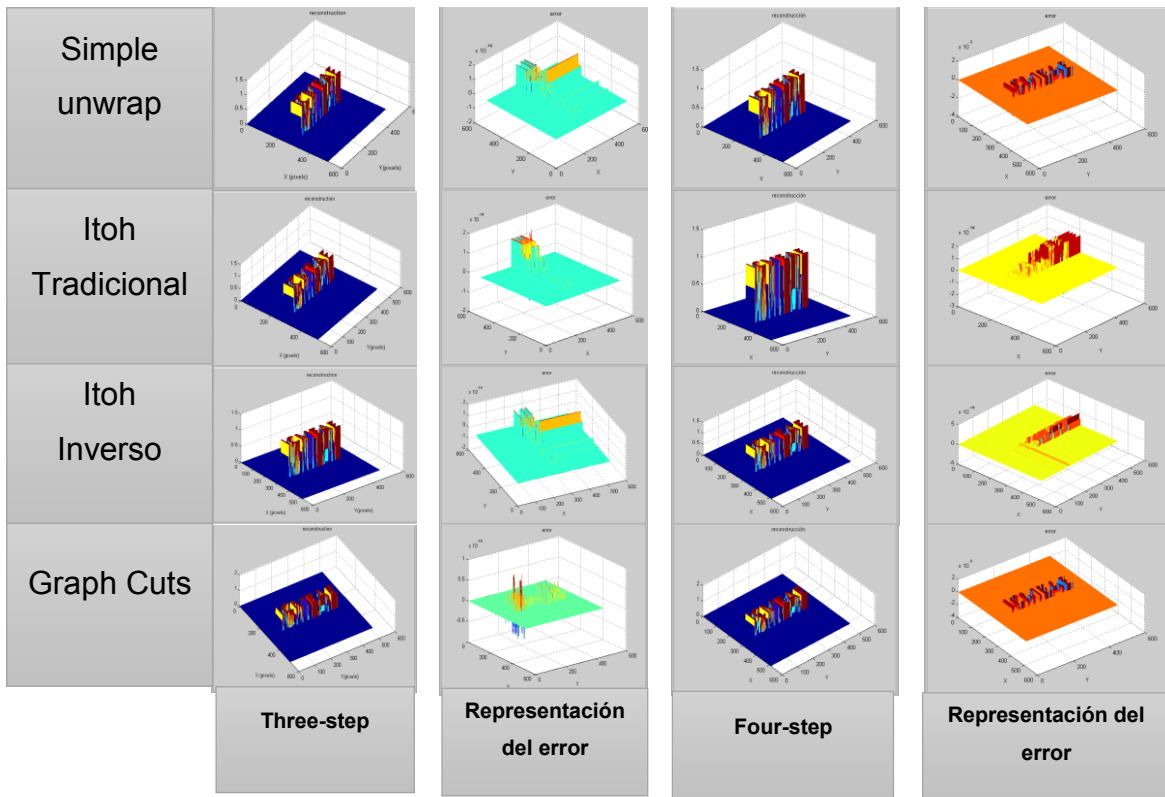


Figura 25. Resultados gráficos de la evaluación de las técnicas “three-step” y “four-step” evaluados con los algoritmos de desdoblamiento de fase: “Simple unwrap”, “Itoh tradicional e inverso” y “Graph Cuts”

Algoritmo de desdoblamiento	Error máximo con 16 franjas (%)	Error mínimo con 16 franjas (%)	Error promedio con 16 franjas (%)	Tiempo de ejecución
<i>Simple unwrap</i>	1.4210×10^{-12}	1.7652×10^{-12}	1.2957×10^{-13}	0.090288 s
<i>Itoh Tradicional</i>	8.4376×10^{-13}	7.5217×10^{-13}	3.5376×10^{-15}	0.134108 s
<i>Itoh Inverso</i>	1.4210×10^{-12}	1.4210×10^{-12}	1.2670×10^{-13}	0.138749 s
<i>Graph Cuts</i>	1.076991×10^{-12}	7.1054×10^{-13}	1.9556×10^{-16}	31.49989 s

Tabla 3. Tabla de resultados utilizando “three-step” con imágenes de profundidad.

Algoritmo de desdoblamiento	Error máximo con 16 franjas (%)	Error mínimo con 16 franjas (%)	Error promedio con 16 franjas (%)	Tiempo de ejecución
<i>Simple unwrap</i>	1.4210×10^{-12}	1.4210×10^{-12}	1.2670×10^{-13}	0.138749 s
<i>Itoh Tradicional</i>	0.0817	0.2545	1.4349×10^{-04}	0.108028 s
<i>Itoh Inverso</i>	0.0817	0.2545	1.4349×10^{-04}	0.138513 s
<i>Graph Cuts</i>	0.0817	0.25456	1.4349×10^{-04}	23.0078 s

Tabla 4. Tabla de resultados utilizando “four-step” con imágenes de profundidad.

En la tabla 3 se muestran mejores resultados en tiempo de ejecución y más precisión que en la tabla 4, en la tabla 4 los resultados del error se estabilizan sin importar el algoritmo de desdoblamiento de fase.

La simulación nos permite cuantificar la certeza de los métodos implementados y nos muestra un panorama de cuáles serán los resultados utilizando imágenes de objetos reales.

4.2 Implementación en una PC

Una vez realizada una simulación utilizando imágenes de profundidad y asegurándonos que los algoritmos ejercían buenos resultados en imágenes ideales se procedió a implementar el algoritmo en una PC con imágenes de objetos reales. Al principio de la investigación no se tenía una correcta calibración así que se evaluó con imágenes obtenidas de la página oficial del asistente de profesor Chen Lujie de la “Singapore University of technology and design” las cuales se observan en la Figura 26.

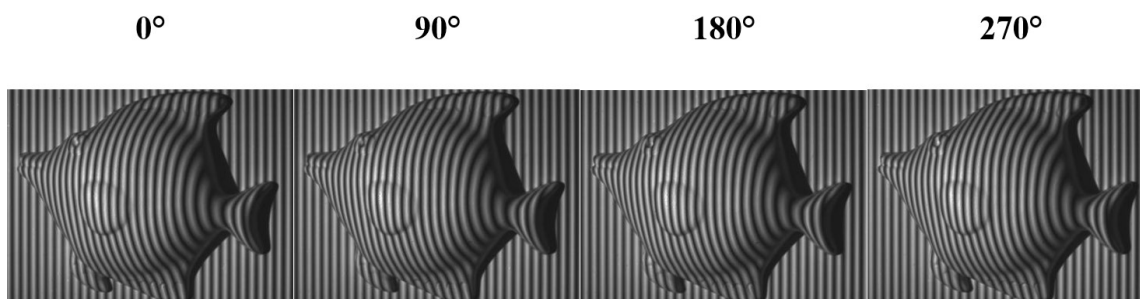


Figura 26. Patrones proyectados a una figura con forma de pescado tomadas de la página oficial del asistente de profesor Chen Lujie de la “Singapore University of technology and design”

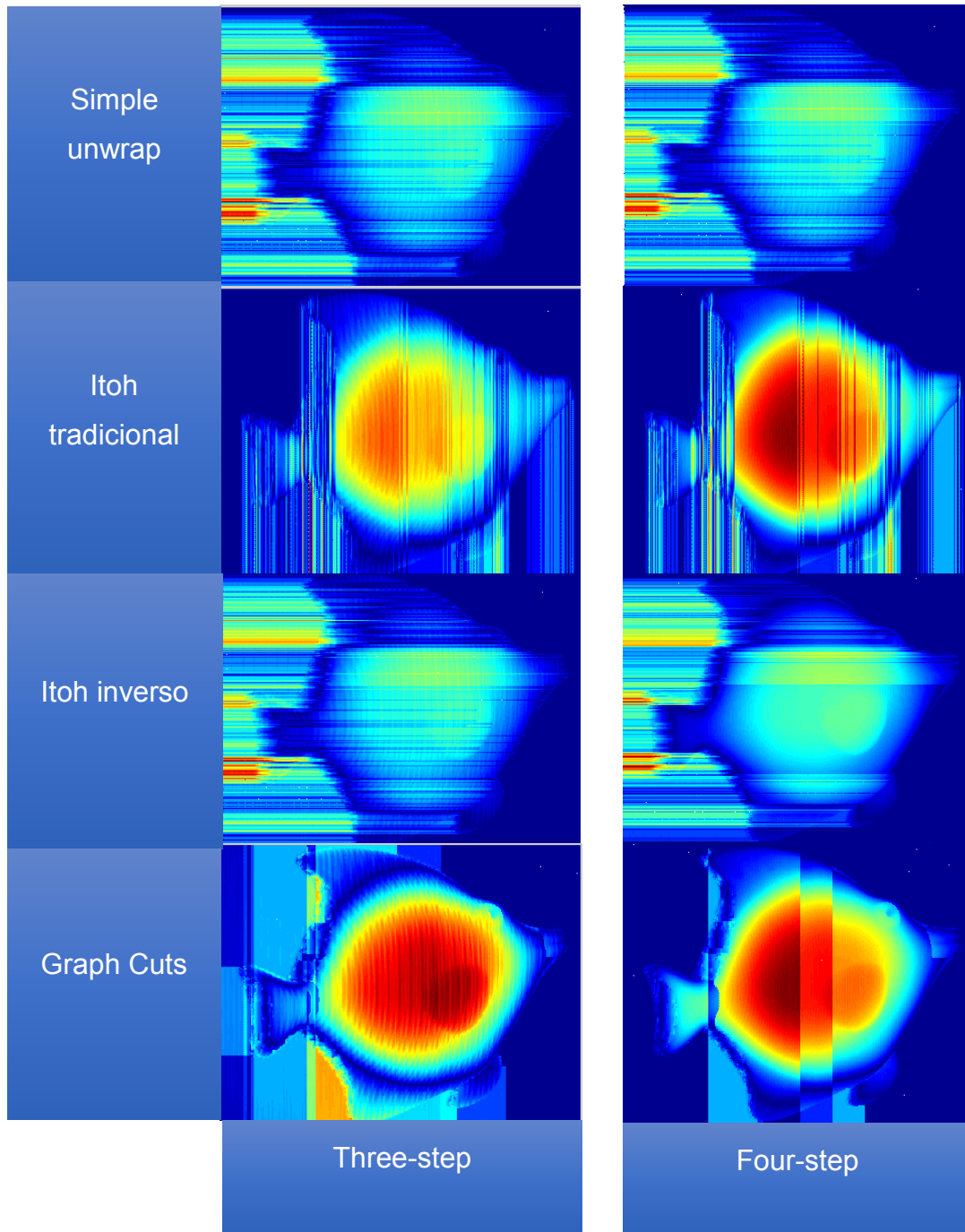


Figura 27. Aplicación de diversos métodos de desdoblamiento de fase usando “three-step” y “four-step” a un objeto real.

A las imágenes de Chen Lujie se les aplicó los mismos algoritmos de desdoblamiento de fase que a la simulación para corroborar los datos obtenidos y los resultados fueron similares “Graph Cuts” es el mejor algoritmo de los evaluados, pero genera mayor tiempo de ejecución. El método “three-step” genera ruido en la superficie del objeto y el método “four-step” genera discontinuidades diferentes a 2π . Se optó por utilizar el método “four-step” en futuras reconstrucciones pues es más sencillo eliminar las discontinuidades diferentes a 2π que el ruido en la superficie del objeto, el método “four-step” tiene mayor precisión y aunque el tiempo de ejecución es mayor se prefirió la precisión.

Una vez generada la calibración tal cómo se describe en la sección 3.3.1 “Calibración de las imágenes” se tomaron objetos de prueba para su calibración, de diferentes formas, materiales y colores algunos ejemplos pueden observarse en la figura 28. Los objetos son: (a) un fondo de vaso desechable encima de una caja blanca, (b) una máscara, (c) una casa de cerámica y (d) un sol de barro. Las figuras (a) y (b) fueron tomadas por una cámara CMOS monocromática y las imágenes (c) y (d) con una cámara CANON EOS REBEL T3i las características de cada cámara pueden observarse en la tabla 5. El proyector usado para las pruebas es un acer k132 y sus especificaciones técnicas (sólo las significativas para el análisis de franjas) se observan en la tabla 6.

Cámara	Resolución	Lente focal	sensor	Profundidad
<i>DAVID USB CMOS Monochrome Camera</i>	1.2 Megapíxeles (1280 x 960)	12mm	CMOS	24 bits
<i>CANON EOS Rebel T3i</i>	17.90 Megapíxeles (5,184 x 3,456)	18-55mm	CMOS	24 bits

Tabla 5. Tabla de especificaciones técnicas de las cámaras utilizadas en el proyecto de investigación.

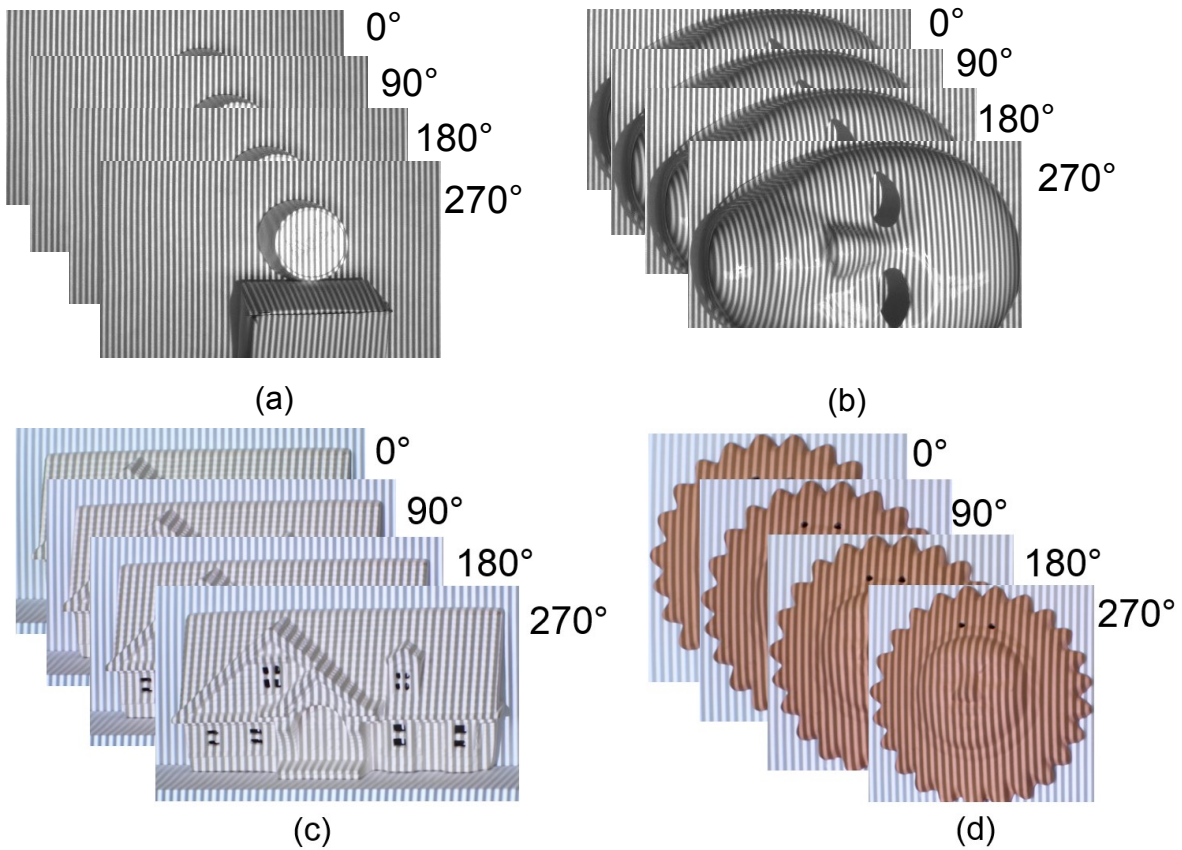


Figura 28. Diferentes ejemplos de capturas con patrones proyectados a objetos reales desplazados en la fase 0, 90°, 180° y 270°. (a) fondo de un vaso desechable encima de una caja blanca. (b) máscara. (c) casa de cerámica. (d) sol de barro.

Resolución	Distancia de Proyección (Mín.)	Distancia de Proyección (Máx.)	Tecnología de proyección
WXGA (1280 x 800)	600 mm	3.20 m	DLP 3D

Tabla 6. Especificaciones técnicas significativas del proyector Acer k132.

Para corregir los brincos o discontinuidades en la reconstrucción se implementaron los algoritmos descritos en la sección “3.5 Post-procesamiento” dando como resultado una mejora en la superficie del objeto cómo se observa en la

figura 29 donde puede observarse la comparación entre el desdoblamiento de fase utilizando “Graph Cuts” antes y después de nuestros algoritmos propuestos, también se muestra la línea de referencia (de forma transversal en la fila 96) donde los brincos o discontinuidades son más notables también se muestra el mallado (mesh) y superficie (surface) del objeto.

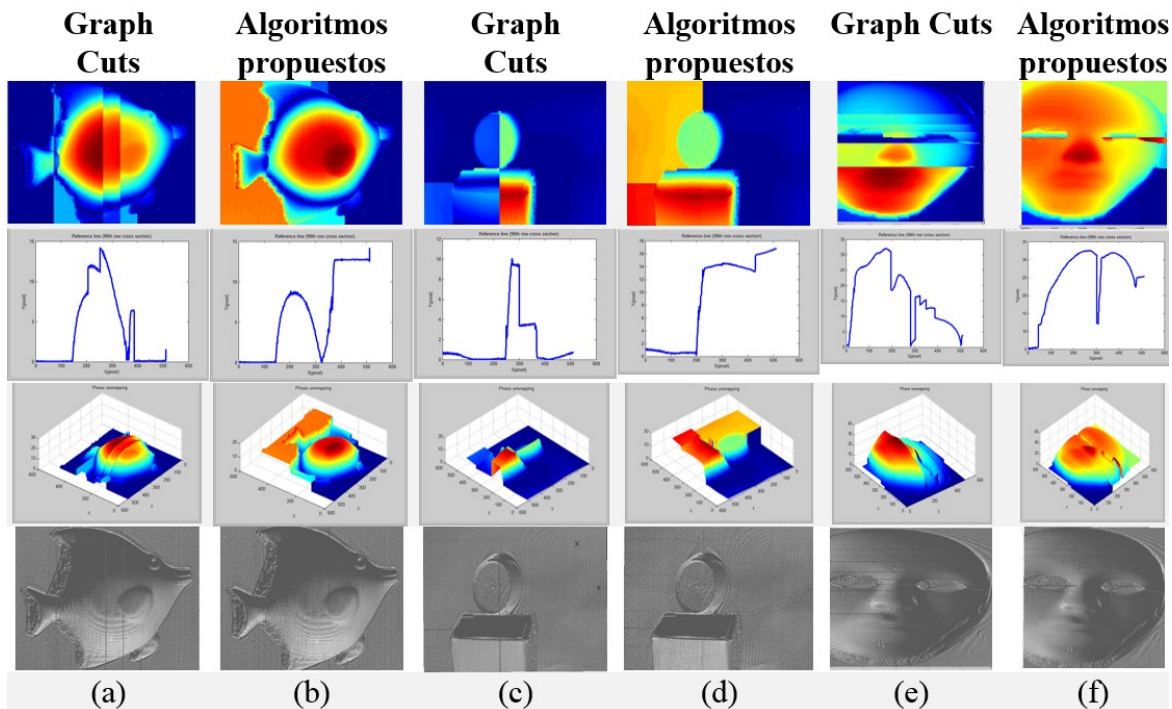


Figura 29. Comparación entre el desdoblamiento de fase antes de los algoritmos propuestos ((a), (c), (e)) y después de implementar los algoritmos propuestos ((b), (d), (f)).

Algunos ejemplos de reconstrucciones tridimensionales utilizando nuestro método pueden verse en las Figuras 30-33, en base a estos experimentos puede mencionarse que el método PSP genera mejor detalle y menor ruido en los objetos claros, también mientras más franjas incidían en el objeto se tendrán mejores resultados (en todos los objetos se proyectó un patrón de 128, pero debido al tamaños de los mismos el número de franjas que inciden en cada uno es diferente).

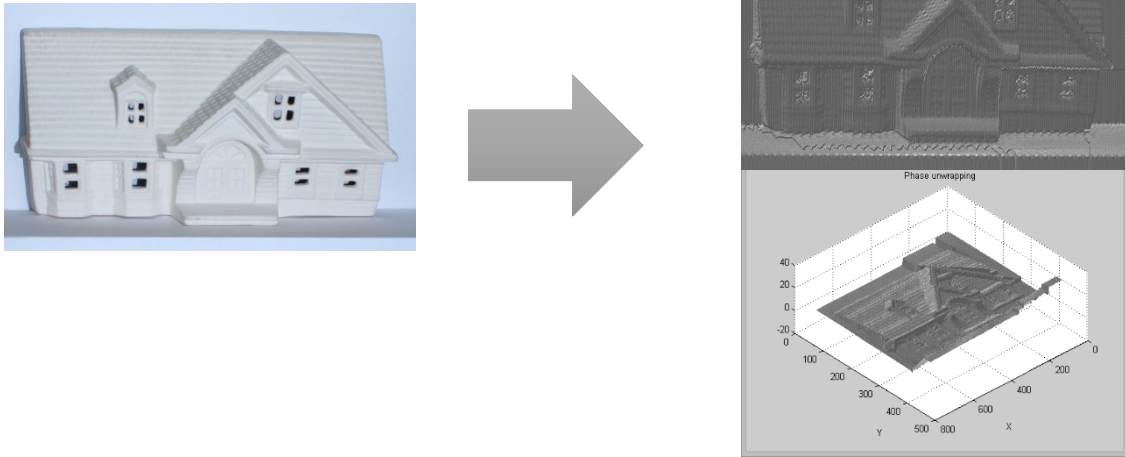


Figura 30. Imagen del objeto original y diversas vistas de la reconstrucción tridimensional de una casa de cerámica.

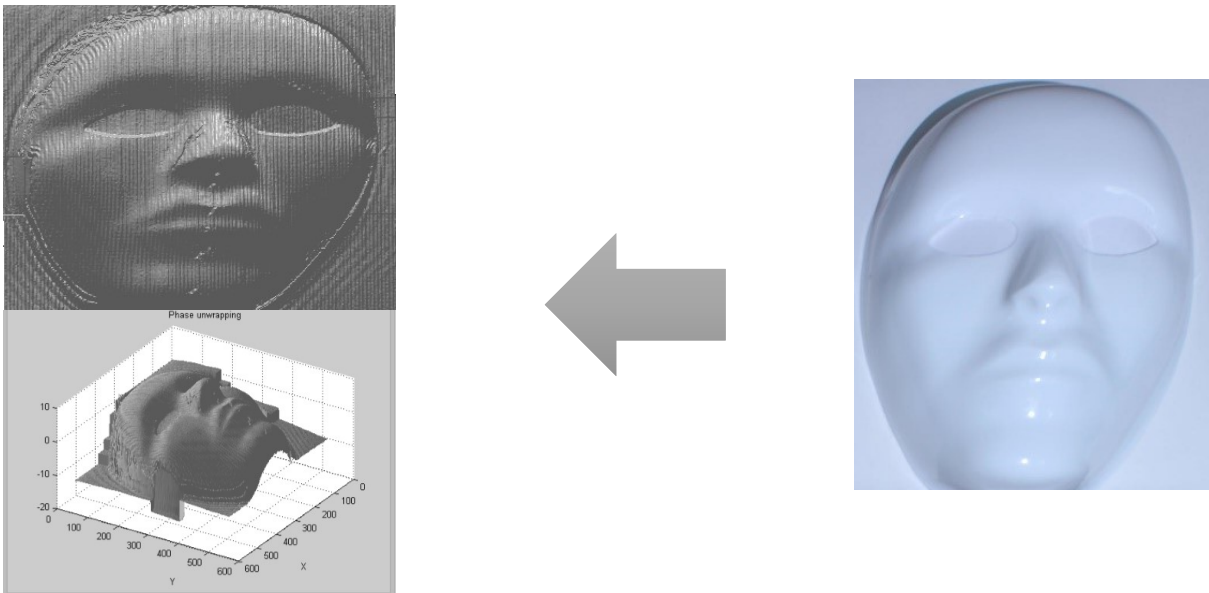


Figura 31. Imagen del objeto original y diversas vistas de la reconstrucción tridimensional de una máscara de plástico.

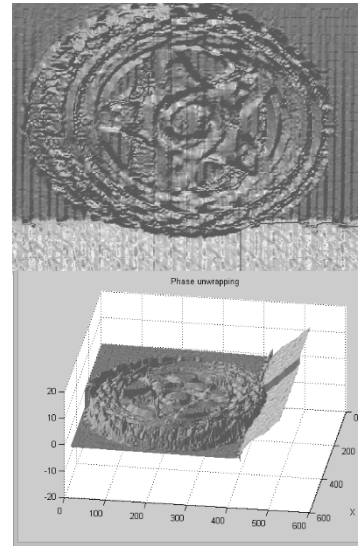
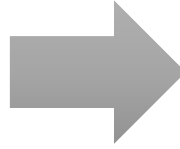


Figura 32. Imagen del objeto original y diversas vistas de la reconstrucción tridimensional de una llanta de juguete.

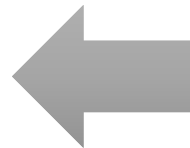
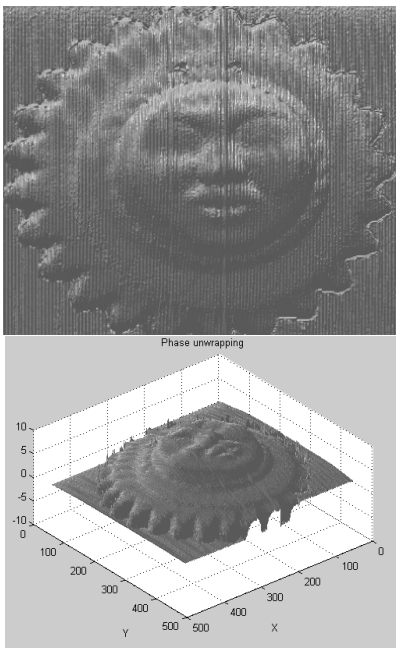


Figura 33. Imagen del objeto original y diversas vistas de la reconstrucción tridimensional de un sol de barro.

4.3 Implementación en una tarjeta de desarrollo con procesador ARM.

Una vez realizada la simulación y la implementación en una PC se procedió a la implementación de los algoritmos requeridos para la reconstrucción tridimensional usando PSP en una tarjeta de desarrollo con procesador ARM. Las diferentes tarjetas evaluadas para este proyecto se pueden observar en la sección “2.3 Procesadores ARM” específicamente en la Figura 8 se observan los diferentes modelos y en la Tabla 1 sus especificaciones técnicas. A todas las tarjetas se les instalaron las mismas bibliotecas, el mismo IDE y el mismo intérprete. El lenguaje de programación que fue elegido para llevar a cabo la programación en las tarjetas fue Python 2.7, por la facilidad de la sintaxis (en promedio, lo que en 6 líneas de código se necesitan en C, en Python sólo es necesario escribir una) y el tiempo del proyecto. La compañía de consultoría en GNU/Linux y free software llamada “Libre Solutions Pty Ltd” realizó un estudio comparativo en el año 2012 en la cual Python obtenía buenos resultados en uso de memoria y tiempos de ejecución. Las gráficas de velocidad y de uso de memoria se muestran en las Figuras 34 y 35 respectivamente sin modificación de nuestra parte. Por lo que existe una buena relación rendimiento – tiempo de desarrollo.

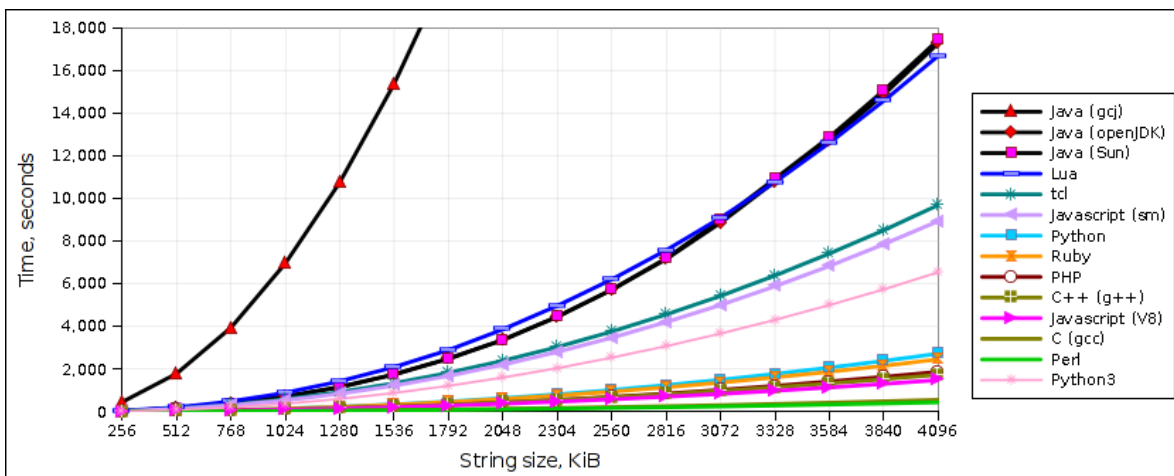


Figura 34. Gráfica de velocidad en diversos lenguajes realizada por la empresa “Libre Solutions Pty Ltd” donde se añaden cadena de caracteres en ciclos (cada ciclo una cadena de 256 kiB).

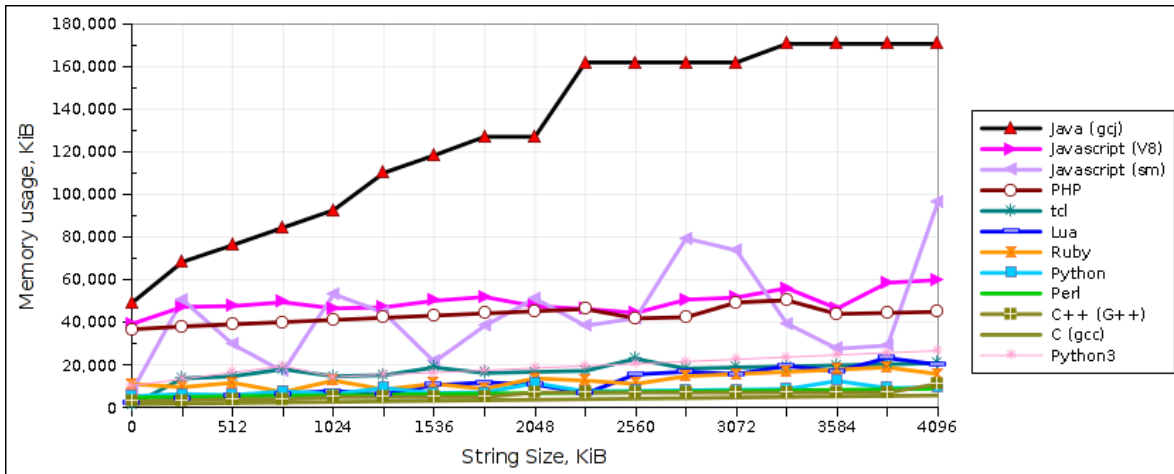


Figura 35. Gráfica de uso de memoria en diversos lenguajes realizada por la empresa “Libre Solutions Pty Ltd” donde se añaden cadena de caracteres en ciclos (cada ciclo una cadena de 256 KiB).

Lo primero que se implementó en las tarjetas fue la creación de franjas en el dispositivo, se tuvieron 2 resultados distintos pues se probó con 2 bibliotecas diferentes la generación de imágenes, la primera fue por medio de `matplotlib` y la segunda con `OpenCV`, el resultado puede observarse en la Figura 36; debido al alto contraste que genera `OpenCV` se usó en los experimentos requeridos.

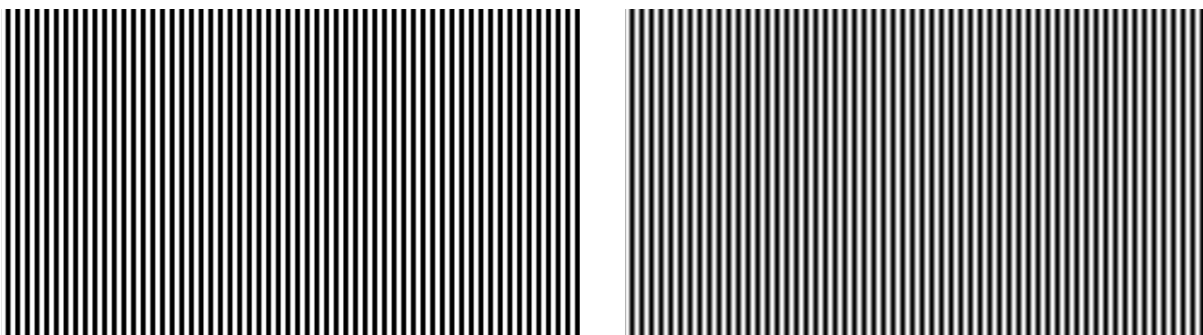


Figura 36. Generación de franjas en las tarjetas de desarrollo. Del lado izquierdo utilizando `OpenCV` y del lado derecho `matplotlib`.

Después de generar las franjas se obtuvo un mapa de fase utilizando la ecuación (13) un ejemplo puede observarse en la Figura 37.



Figura 37. Ejemplo de mapa de fase dentro de las tarjetas de desarrollo.

Una vez obtenido el mapa de fase se procedió a implementar el algoritmo Simple unwrap. Teniendo resultados similares al proceso utilizado en la PC. Una comparación entre ambas imágenes se puede observar en la Figura 38.

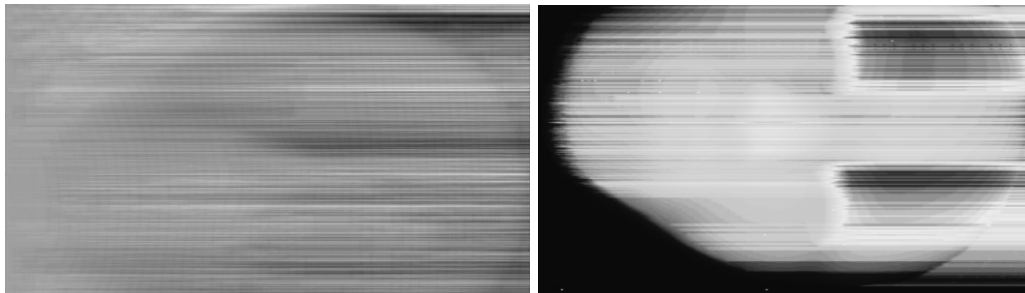


Figura 38. Comparación entre desdoblamiento Simple unwrap en una tarjeta utilizando Python (imagen de la izquierda) y desdoblamiento Simple unwrap en una PC utilizando Matlab (imagen de la derecha).

Para este punto se sabía que el mejor resultado era el otorgado por “Graph cuts” (dentro de los algoritmos evaluados), pero es un algoritmo complejo de implementar. En una PC con 8 núcleos le lleva más de medio minuto procesar el algoritmo, así que se decidió implementar por primera vez “Quality-guided” (descrito en la sección 3.2.4 Algoritmo “Quality-guided”) el primer reto fue determinar el mapa de calidad utilizando la ecuación (17). El resultado se observa en la figura 39 mientras mayor calidad cuente el píxel más oscuro será.

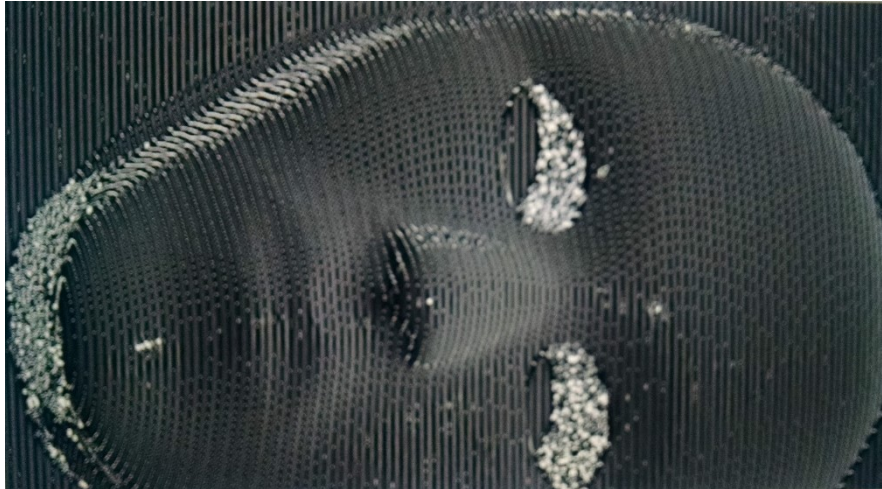


Figura 39. Ejemplo de mapa de calidad implementado en la tarjeta de desarrollo.

Después de obtener el mapa de calidad se desdobló la fase teniendo como resultado la imagen de la Figura 40.

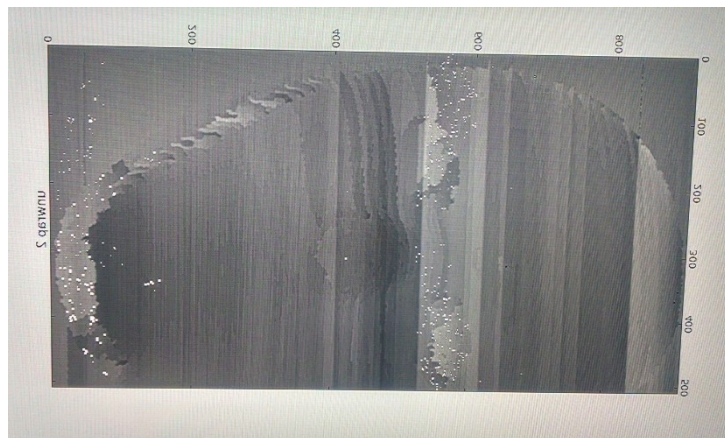


Figura 40. Desdoblamiento de fase utilizando "Quality-guided".

Se probó con diversas bibliotecas lo forma de graficar el resultado en una malla o superficie tridimensional pero el rendimiento de la tarjeta se decrementó considerablemente por lo que se decidió graficarlo por medio de `matplotlib`. Un resultado puede observarse en la Figura 41.

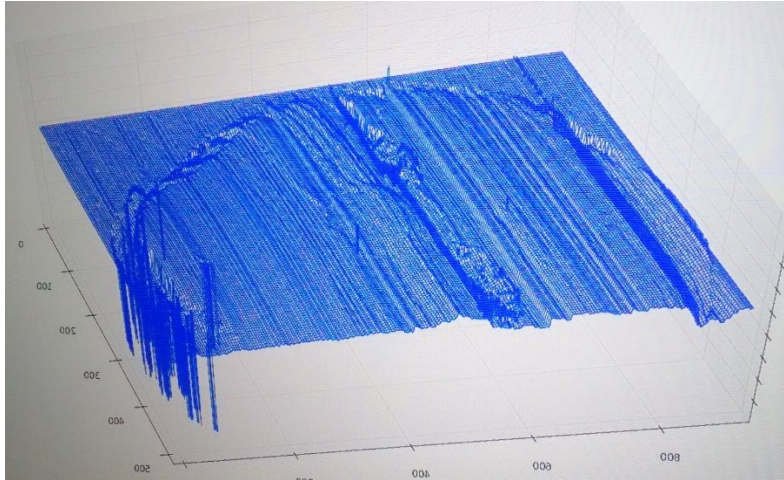


Figura 41. Ejemplo de resultado en 3D.

5. CONCLUSIONES

Existen diversas técnicas para obtener el perfil de un objeto, cada técnica es una amplia ventana de investigación, un ejemplo es el presente trabajo que se desenvuelve a través de técnicas sin contacto, las ópticas, luz estructurada y el análisis de franjas para desembocar en la descripción de un método ampliamente usado, Perfilometría por Desplazamiento de Fase (Phase-Shifting Profilometry) o PSP que utiliza 3 o más patrones sinusoidales proyectados a un objeto, la superficie del objeto distorsiona el patrón, estas escenas son capturadas utilizando cada patrón y por medio de técnicas de procesamiento de imágenes se genera una reconstrucción tridimensional.

Existen 2 etapas fundamentales en el análisis de franjas el envolvimiento de fase y el desdoblamiento de fase, el primero se encarga de obtener el ángulo del objeto envolviendo la señal en un rango de 2π , por lo que un algoritmo de desdoblamiento de fase es requerido para eliminar la discontinuidad de 2π . Durante el trabajo de investigación se evaluaron 4 algoritmos diferentes (los más comunes) que permiten desdoblar la fase. Los algoritmos que demostraron tener mejores resultados son: "Graph cuts" y "Quality-guided"; ambos utilizan rutas complejas por las cuales se irá desdoblado la fase píxel a píxel.

El presente trabajo describe cada una de las etapas necesarias para la reconstrucción tridimensional utilizando PSP, las cuales son: Adquisición de imágenes (incluyendo la calibración), pre-procesamiento, mapa de fase, desdoblamiento de fase y post-procesamiento.

En la etapa de post-procesamiento se propusieron 2 algoritmos de sencilla implementación que permiten la eliminación de discontinuidades diferentes a 2π , lo que significa que existe una contribución relevante en el campo de análisis de franjas pues para corregir este error debía ser modificado un algoritmo de desdoblamiento de fase en particular y nuestros algoritmos funcionan indiferentemente del algoritmo que se implemente.

La tesis muestra los resultados de implementar una simulación, esto permite evaluar diferentes algoritmos de análisis de franjas y nos otorga una manera de cuantificar el error inherente en cada método.

Los resultados nos muestran que PSP es sensible al color de los objetos (mientras más claro sea el objeto a digitalizar se obtendrán mejores resultados), mientras más franjas incidan en el objeto mayor será la fidelidad de la reconstrucción. La técnica “four-step” (empela 4 patrones distintos) otorga una buena relación entre tiempo de ejecución y resolución.

Se implementó el método PSP en 2 distintas arquitecturas(x64 y ARM), y 2 lenguajes de programación distintos (Matlab y Python). En la sección de resultados se demostró que PSP puede ser ejecutado por una tarjeta de desarrollo con procesador ARM.

El trabajo permite la continuación del proyecto incrementando la resolución de objetos tridimensionales en tarjetas de desarrollo con procesadores ARM, la inclusión de rotación de los objetos con el fin de obtener cada ángulo del objeto, la mejora en el tiempo de ejecución de los algoritmos de desdoblamiento de fase. La futura inclusión de una interfaz que agrupe todos los procesos en un solo sistema desde la proyección y calibración de las imágenes hasta la visualización de imágenes tridimensionales.

6. REFERENCIAS

Bioucas-Dias, J. M., & Valadao, G. (2007). Phase unwrapping via graph cuts. *Image Processing, IEEE Transactions on*, 16(3), 698-709.

Singapore University of technology and design, ChenLujie 02 de diciembre de 2015, Sitio web: <http://www.sutd.edu.sg/ChenLujie.aspx>

Ekstrand, L., Wang Y., Karpinsky N., and Zhang S. (2013). Superfast 3D Profilometry with Digital Fringe Projection and Phase-Shifting Techniques, *Handbook of 3D Machine Vision: Optical Metrology and Imaging (Vol. 16)*. CRC Press (pp. 233-252).

Gdeisat, M., & Lilley, F. (2012). One-Dimensional Phase Unwrapping Problem. 16 de noviembre de 2015, de Liverpool John Moores University Sitio web: http://www.ljmu.ac.uk/GERI/CEORG_Docs/OneDimensionalPhaseUnwrapping_Final.pdf

Gdeisat, M., & Lilley, F. (2012). Two-Dimensional Phase Unwrapping Problem. 16 de noviembre de 2015, de Liverpool John Moores University.

Giryas, R., Bronstein, A. M., Moshe, Y., & Bronstein, M. M. (2008). Embedded System for 3D Shape Reconstruction. In *Proc. of the 3rd European DSP Education and Research Symposium* (pp. 265-272).

Guzhov, V. I., Ilinykh, S. P., Haydukov, D. S., & Kuznetzov, R. A. (2012). Decoding algorithm for interference patterns in phase shifting interferometry without a priori shift knowledge. In *Strategic Technology (IFOST), 2012 7th International Forum on* (pp. 1-4). IEEE.

Hadjitheophanous, S., Ttofis, C., Georghiades, A. S., & Theocharides, T. (2010). Towards hardware stereoscopic 3D reconstruction a real-time FPGA computation of the disparity map. In *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2010* (pp. 1743-1748). IEEE.

Huang, P. S., & Zhang, S. (2006). Fast three-step phase-shifting algorithm. *Applied optics*, 45(21), 5086-5091.

Kato, J. I. (2009). Fringe Analysis. *Handbook of Optical Metrology: Principles and Applications*, 1, 499.

Langbridge, J. A. (2013). *Professional embedded ARM development*. John Wiley & Sons.

Li, B., Wang Y., Dai, J., Lohry, W., Zhang, S. (2013). Some recent advances on superfast 3D shape measurement with digital binary defocusing techniques. *Optics and Lasers in Engineering*.

Libre Solutions Pty Ltd 07 de diciembre de 2015, Sitio web: <http://raid6.com.au/~onlyjob/posts/arena/>

Matlab 16 de noviembre de 2015, Sitio web: <http://www.mathworks.com/>

Merner, L., Wang, Y., Zhang, S. (2013). Accurate calibration for 3D shape measurement system using a binary defocusing technique. *Optics and Lasers in Engineering*.

Molleda, J. (2008). Técnicas de visión por computador para la reconstrucción en tiempo real de la forma 3D de los productos laminados. Tesis doctoral, Universidad de Oviedo.

Peng, T., Gupta, S. K., & Lau, K. (2005). Algorithms for constructing 3-D point clouds using multiple digital fringe projection patterns. *J Comput Aided Design Appl*, 2, (pp.737-746).

Tadeo R. (2008). Reconstrucción de objetos tridimensionales a partir de información bidimensional. Tesis de Maestría del Instituto Politécnico Nacional.

Vilá K., Arranz A., Alvar, M. Sánchez, A. (2009). Reconstrucción 3D de Modelos Utilizando Técnicas de Visión Artificial (Proyecto de Fin de Carrera) universidad Pontificia Comila, Escuela Técnica Superior de Ingeniería (ICAI).

Zhang, S., Li, X., & Yau, S. T. (2007). Multilevel quality-guided phase unwrapping algorithm for real-time three-dimensional shape reconstruction. *Applied optics*, 46(1), 50-57.

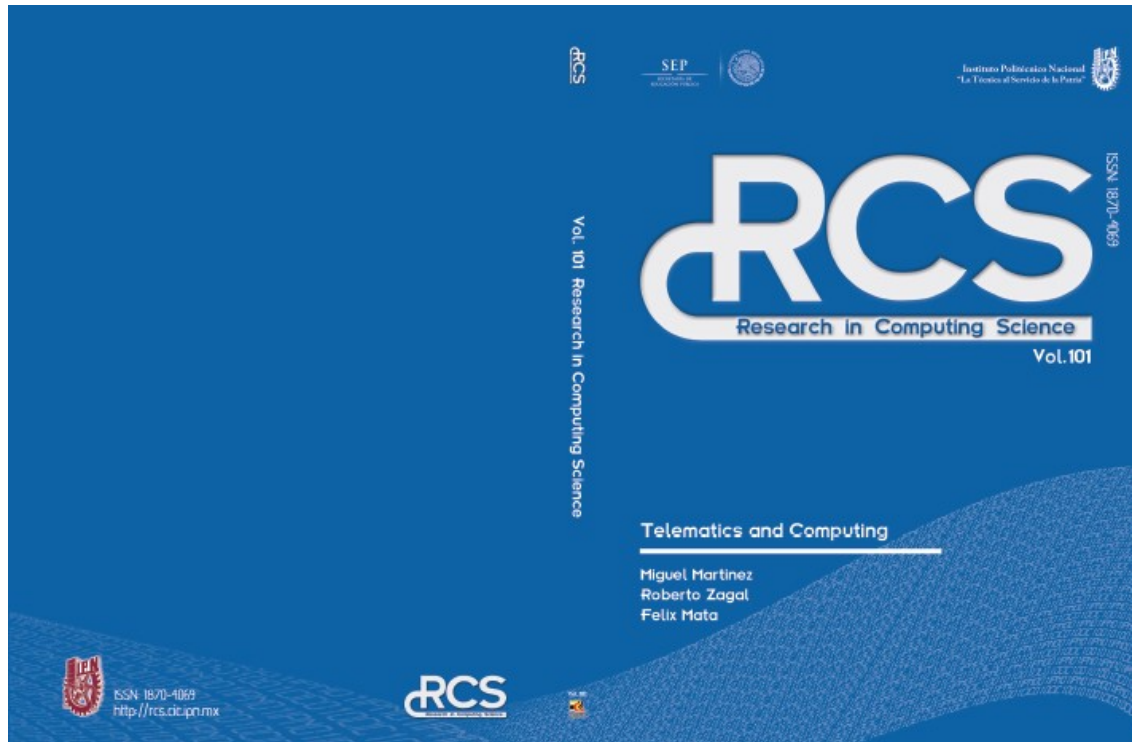
Zhang, S., & Huang, P. S. (2006). High-resolution, real-time three-dimensional shape measurement. *Optical Engineering*, 45(12), 123601-123601.

Zhang, S., Van Der Weide, D., & Oliver, J. (2010). Superfast phase-shifting method for 3-D shape measurement. *Optics express*, 18(9), 9684-9689.

Zhao, M., Huang, L., Zhang, Q., Su, X., Asundi, A., & Kemao, Q. (2011). Quality-guided phase unwrapping technique: comparison of quality maps and guiding strategies. *Applied optics*, 50(33), 6214-6224.

7. APÉNDICE

A continuación se muestran los distintos reconocimientos y artículos obtenidos a partir del trabajo presentado en esta tesis.



Novel approach to eliminate discontinuities in phase unwrapping applied to Phase-Shifting Profilometry

Rodrigo Escobar¹, Juan Carlos Moya, Juan Manuel Ramos, Efrén Gorrostieta and
Jesus Carlos Pedraza

Facultad de Informatica, Universidad Autonoma de Queretaro, Queretaro, Mexico
Corresponding author¹: rodediazg@gmail.com

Abstract. In order to obtain the 3D information using fringe analysis, phase maps are required and must be recovered from the wrapped phase, so that phase unwrapping is a critical step in the optical measurement by fringe projection. The phase wrapping is the process that determines the values of phase in a range from 0 to 2π and the phase unwrapping is a process in which discontinuity of 2π is removed. Although the phase unwrapping removed of discontinuity of 2π , by several factors in some cases there are errors different to 2π , we propose a new approach to solve this problem by adding a post-processing step, which consist in search the number of sections that contains discontinuities and after these discontinuities are eliminated, the compensation of the values for each section is carried out. The wrapped phase map is extracted from the deformed fringe patterns by the use of Phase-Shifting Profilometry (PSP) technique and particularly using four-step algorithm. Our experimental results show that if the proposed approach is applied after the phase unwrapping step, then obtains better results than the obtained by only using a the phase unwrapping step. Here, the phase unwrapping algorithms; "Itoh Traditional", "Itoh inverse", "Graph Cuts" and "Simple unwrap" are applied to the PSP.

Keywords: Phase wrapping; Phase unwrapping; Phase-Shifting Profilometry; depth image; optical measurement.

1 Introduction

One of the best techniques used in the optical measurement is the fringe projection technique, which utilizes a group of defined fringe pattern (sinusoidal or periodic) projected from a digital projector to the object surface, therefore the object's shape creates distortions in pattern. Then the distortions are captured with a digital camera from another angle, and finally the image is processed to obtain the 3D information of surface [1] [2] [3].

Between fringe projection techniques, Fourier Transform Profilometry (FTP), Wavelet Transform Profilometry (WTP) and Phase-Shifting Profilometry (PSP) are the most used methods, each of them present diverse advantages and disadvantages with respect to other. For instance, when only one image is processed commonly the Fourier

Transform Profilometry is applied, which works as a spatial phase modulation of the pattern carrier with a fundamental frequency f_0 . Different phase demodulation algorithms can be applied to this carrier pattern. FTP extracts only the term of modulated phase through filtering spectrum, using one or two dimensions FTP and its inverse [1].

When is used more than one image pattern, the PSP algorithms are used due to its numerous advantages, point - by-point measurement (which allows the resolution level pixel-camera), less sensitive to variations in reflectivity surface (facilitating the measurement of complex objects with strong variations in texture) and less sensitivity to ambient light [1] [2][4][5]. In this research we only focus on the PSP technique.

There are numerous methods of phase-shifting in which stands out: "three-step", "four-step" and "double three-step". These methods differ in the number of input images, like the "three-step" approach will use 3 images, "four-step" will use 4 and so on.

To get the height of an object is necessary to obtain the information immersed into the phase; usually these information is wrapped into the phase. The phase modulation is the result of the deformed pattern, projected on the surface and the fringe pattern can be described by phase maps. By analyzing the phase maps, the 3D shape can be recovered [3]. If more images are employed the quality of result will improve, but the runtime will increase. We consider that a good relationship runtime-quality is given by the 4-step method, therefore is the method that we select.

Two basic steps to the phase treatment are necessary, which are: the phase wrapping and the phase unwrapping. The phase wrapping is the process that determines the values of phase in a range from 0 to 2π . The phase unwrapping is a process in which discontinuity of 2π is removed to generate a phase map [6] [7] [8].

The phase wrapping can be expressed mathematically as:

$$x_w(n) = W[x(n)] \quad (1)$$

Where $x(n)$ is the original continuous phase, $W[\]$ is the phase wrapping operation and $x_w(n)$ is the phase wrapping [9].

Knowing that the phase unwrapping is an important part of any optical measurement numerous algorithms have been proposed to create better results in either response time or better resolution in measurements. For example "Itoh traditional", "Itoh inverse", "Graph cuts", "Quality guided", etc. [10].

2 Methodology

The PSP four-step used can be described as follows:

$$I_1(x, y) = I'(x, y) + I''(x, y) \cos[\phi(x, y) + 3\alpha] \quad (2)$$

$$I_2(x, y) = I'(x, y) - I''(x, y) \sin[\phi(x, y) + 2\alpha] \quad (3)$$

$$I_3(x, y) = I'(x, y) - I''(x, y) \cos[\phi(x, y) + \alpha] \quad (4)$$

$$I_4(x, y) = I'(x, y) + I''(x, y) \sin[\phi(x, y)] \quad (5)$$

Where $I'(x, y)$ is the average intensity, $I''(x, y)$ is the intensity modulation, $\phi(x, y)$ is the phase to resolve. In our case we consider $\alpha = \pi/2$.

$$\phi(x, y) = \tan^{-1} \left(\frac{I_1(x, y) - I_2(x, y)}{I_1(x, y) - I_3(x, y)} \right) \quad (6)$$

As shown in Fig.1(a), a set of images with the different fringe pattern projected are acquired, then a pre-processing stage is applied to eliminate the angle distortion of the camera view, an image calibration by using an estimate geometric transform is performed. Later an equalization of histogram for each image is carried out. To obtain the phase map we use the Equation 6. In the next step the phase unwrapping is treated with different algorithms ("Itoh Traditional", "Itoh inverse", "Graph Cuts" and "Simple unwrap"). Finally to the best phase unwrapping algorithm is applied by using the new approach proposed in algorithms 1 and 2, as a post-processing stage.

Before to apply the proposed method to real objects, a simulation of the process by using depth images in grayscale is carried out to verify all equations without errors of calibration.

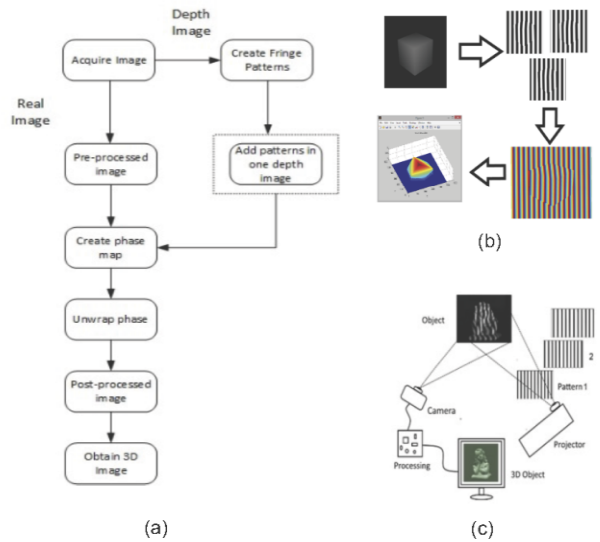


Fig. 1. (a) Methodology. (b) An example using depth images. (c) System architecture.

The complete process of PSP is simulated as if it were real objects (using depth image). In Fig.1(c) the real system architecture can be observed. Here, the sinusoidal

defined patterns are projected to an object, then the image is captured, after angle pattern will be shifting (usually 120° in case of three step and 90° in four step algorithm), then re-capture the image and so until get 3 or more images, the images being processed by using Equation 6 to obtain the phase map (contains the information of the height of the object), finally tridimensional object is obtained by the phase unwrapping. Fig. 1(b) show one example using depth images where projected patterns are created artificially and is given the same treatment that used in images of real objects (except the calibration of the distortion angle).

To calculate error, we used the difference between the reconstructed image (pixel by pixel) and the three-dimensional representation of the depth image. In the depth image each pixel has a value of 0 (black) to 255 (white) which may be represented by a value from 0 to 1 (depth) to obtain the reconstructed image as in Fig. 2.

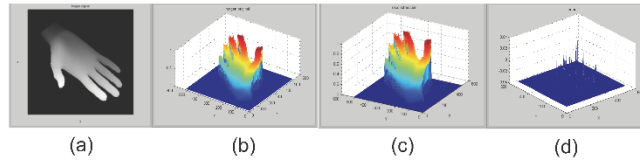


Fig. 2. (a) Example of depth image, (b) Three-dimensional representation of the depth image, (c) Reconstructed image using PSP, (d) Three-dimensional representation of the error.

2.1 Algorithms of phase unwrapping

Since there are many algorithms of phase unwrapping and the main authors differ which is more suitable, the most used were analyzed:

- **Simple unwrap.** This algorithm simply corrects the phase angles in a vector adding multiples of $\pm 2\pi$ when absolute jumps between consecutive elements [11].
- **Itoh traditional.** The algorithm of Itoh involves unwrapping the rows one by one to verify discontinuity between the values of the row, followed by an unwrapping in columns (same as one by one) [10].
- **Itoh inverse.** The algorithm of Itoh Inverse involves unwrapping the rows one by one to verify discontinuity between the values of the columns, followed by an unwrapping in row (same as one by one) [10].
- **Graph cuts.** The algorithm implemented for Phase Unwrapping via Graph Cuts use max-flow/min-cut calculations described by José M. Bioucas-Dia and Gonçalo Valadao in 2007 [12].

Fig. 3 shows the results evaluating the most common algorithms of phase unwrapping with three different test objects (a fish, a box with a half disposable cup and one mask), Fig. 3(a) shows the result of applying a simple unwrap algorithm, Fig. 3(b) shows the result of applying the Itoh traditional algorithm, Fig. 3(c) shows the result of applying the Itoh inverse algorithm and Fig. 3(d) shows the result of applying the Graph

cuts algorithm. At the end of the test the best results are given by the Graph cuts algorithm. Although the phase unwrapping removes the discontinuity of 2π , in some cases there are errors different to 2π creating sections with jumps in the image, therefore we propose a post-processing stage to solve this problem. At first instance by finding the sections where jumps occurred through a reference line and are evaluated pixel by pixel to check the discontinuity, if one discontinuity is founded, its value is store in a matrix (with the pixel position and value where occurred the discontinuity) then all values of discontinuity are eliminated for each section compensating the value with one sum. The methodology is completely described in the algorithms 1 and 2.

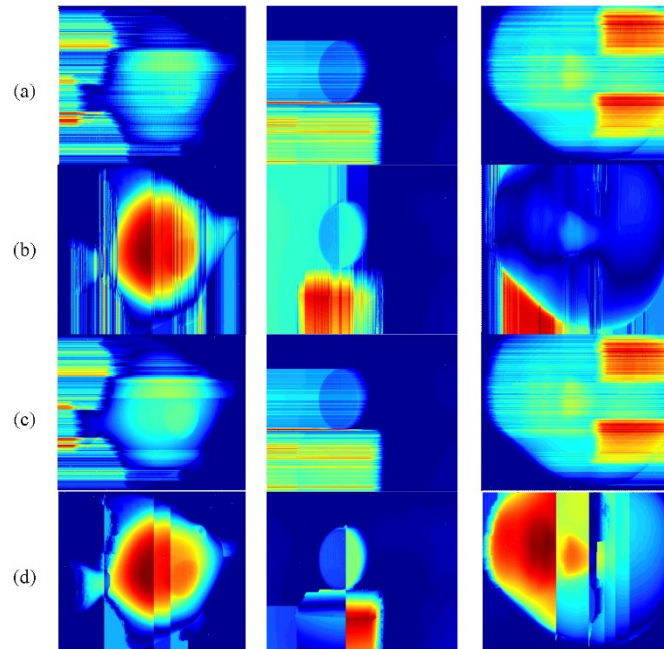


Fig. 3. The results evaluating the most common algorithms of phase unwrapping with three different test objects (a fish, a box with a half disposable cup and one mask), (a) Simple unwrap, (b) Itoh traditional, (c) Itoh inverse, (d) Graph cuts.

Algorithm 1: Find sections

```
input wi =unwrapping image (M x N)
1: // Define reference line
2: line = wi (number of row cross section , :)
3: number of sections = 0
4: for i = 2→N
5:     current pixel = line (i)
6:     previous pixel = line(i-1)
7:     discontinuity = current pixel – previous pixel
8:     if discontinuity > threshold
9:         number of sections ++
10: // border values are stored
11:     Section Value(number of sections, 1) = i
12:     Section Value(number of sections, 2) = i - 1
13:     Section Value(number of sections, 3) = discontinuity
14:     end if
15: end for
16: Section Value(number of sections +1,2)= N;
```

Algorithm 2: Delete discontinuity

```
input wi =wrapping image (M x N), line = wi ( M/2 , :), Section Value(:,1) =
previous pixel to discontinuity, Section Value(:,2) = next pixel to discontinuity,
Section Value(:,3) = value of discontinuity
1: e = discontinuity to fix = 0
2: if number of sections > 0
3:     for j = 1 → number of sections
4:         v1 = Section Value(j,1)
5:         v2 = Section Value(j,2)
6:         e = Section Value(j,3)
7:         for l = 1 →M
8: // linC = line to correct
9:             linC = wi ( l , :)
10:            for k = v1→v2
11:                linC(k)=linC(k) -e
12:                wi ( l , :) = linC
13:            end for
14:        end for
15:    end for
16: end if
```

Finally the image with discontinuity is corrected but to correct the background it is necessary to apply a mask created from raw data.

3 Experimental results

A set of four images which contain a real object were considered [13] and are shown on Fig. 4 where four different shifting angles of the fringe pattern are projected to a fish. Another set of images were taken and are shown on Fig. 5. These images are ideal to observe a discontinuity error different to 2π such as shadow effect inherent in projection, on-uniform reflectance but a good calibration of projection. In the phase map is evident this aspects as Fig. 6 that show the difference between a good and bad calibration as well as the shadow effect.

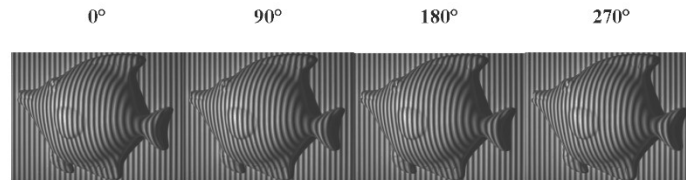


Fig. 4. Patterns projected to a fish taken of data raw of Chen Lujie [13].

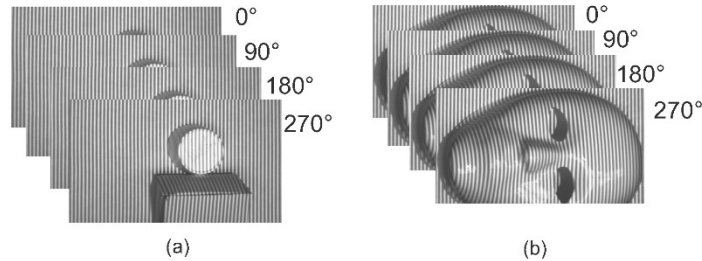


Fig. 5. Images taken for us, (a) Patterns projected to a box with a half disposable cup, (b) Patterns projected to a mask.

The Computer used to carry out the experiments is a Lenovo Y510p with 8GB of RAM, Intel® Core™ i7-4700MQ 2.4GHz and Windows 8.1 64 bits. The algorithms were implemented in Matlab R2013b. In the experiment system, the sinusoidal fringe patterns was generated by Accr K11+ with native resolution of 858 X 600 (SVGA) and 200 lumens and capture by DAVID-CAM-3-M with 1280x960 of resolution.

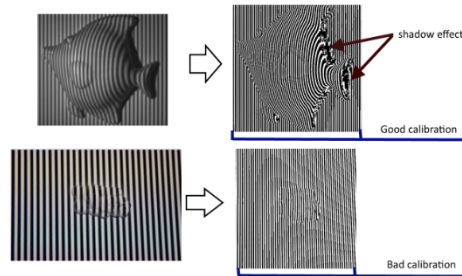


Fig. 6. Examples of shadow effect, good and bad calibration of projection.

The different test objects are a fish, a box with a half disposable cup and one mask. Fig.7 shows the comparison between the phase unwrapping before and after the algorithms proposed were applied, also shows the reference line (96th row cross section) where the jumps are more noticeable, moreover it shows the 3D reconstruction as a mesh and a surface. It can be observed that the most of the discontinuity errors were eliminated with the algorithms proposed. Only persist the error in the background of each image but not in surface in the test object.

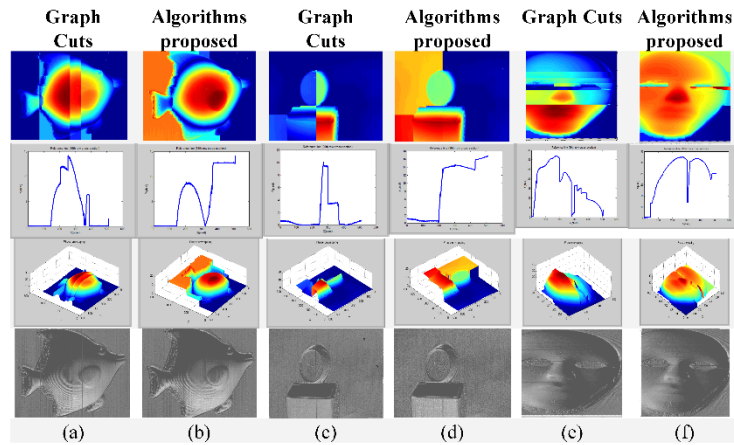


Fig. 7. Comparison between the images before of the algorithms proposed ((a), (c), (e)) and after the algorithms proposed ((b), (d), (f)).

4 Conclusions and Future Work

This research described as use the Phase-Shifting Profilometry to obtain tridimensional information (particularly four-step method) through each stage; pre-processing (where the image will be adequate for the future processing), phase wrapping (stage where extracted the phase information using PSP), phase unwrapping (process in which discontinuity of 2π is removed to generate a phase map) and post-processing (stage where we propose two algorithms to eliminate the discontinuity after of phase unwrapping).

Moreover, a novel approach can be described as the elimination of the discontinuity different to 2π (applied to PSP) caused by bad calibration, noise in the image and shadow effect. This discontinuity is achieved by finding and correcting each section of discontinuity. The new approach are divided in two algorithms after of the phase unwrapping, the first algorithm finds all the sections where there are jumps inside the image and the second algorithm compensates the discontinuity with only one sum. The proposed algorithms are simple to implement and show more accurate results in the surface of the objects than previous works presented, in which only a simple unwrapping algorithm is used. Finally in the new approach only persist the errors in the background but eliminates the errors in the surface.

As future work it is proposed to implement the algorithms in an embedded system based on the ARM processor and improve post-processing adding a mask to remove background noise. Also correct the error in the background through a mask from raw images or using a quality-guided method.

Acknowledgments. Thank to “Consejo Nacional de Ciencia y Tecnología” (CONACYT) for their support and sponsorship right through the National Scholarship number also we would like to thank Universidad Autonoma de Queretaro and “Fondo de Proyectos Especiales de Rectoría” (FOPER UAQ-2015), for the financial support in this work.

References

1. Ekstrand, L., Wang Y., Karpinsky N., and Zhang S. (2013). Superfast 3D Profilometry with Digital Fringe Projection and Phase-Shifting Techniques, Handbook of 3D Machine Vision: Optical Metrology and Imaging (Vol. 16). CRC Press (pp. 233-252).
2. Kato, J. I. (2009). Fringe Analysis. Handbook of Optical Metrology: Principles and Applications, 1, 499.
3. Chen, K., Xi, J., & Yu, Y. (2013). Quality-guided spatial phase unwrapping algorithm for fast three-dimensional measurement. Optics Communications, 294, 139-147.
4. Zhang, S. (2013). Handbook of 3D Machine Vision: Optical Metrology and Imaging (Vol. 16). CRC Press.

Mario Eduardo Rivero, Laura Ivoone Garay Jiménez and Sergio Manuel Martínez Chávez

5. Cong, P., Xiong, Z., Zhang, Y., Zhao, S., & Wu, F. (2015). Accurate dynamic 3D sensing with Fourier-assisted phase shifting. *Selected Topics in Signal Processing, IEEE Journal of*, 9(3), 396-408.
6. Huang, P. S., & Zhang, S. (2006). Fast three-step phase-shifting algorithm. *Applied optics*, 45(21), 5086-509.
7. Zhang, S., & Huang, P. S. (2006). High-resolution, real-time three-dimensional shape measurement. *Optical Engineering*, 45(12), 123601-123601.
8. Zhang, S., Van Der Weide, D., & Oliver, J. (2010). Superfast phase-shifting method for 3-D shape measurement. *Optics express*, 18(9), 9684-9689.
9. Gdeisat, M., & Lilley, F. (2012). One-Dimensional Phase Unwrapping Problem. 08/20/15, de Liverpool John Moores University.
10. Gdeisat, M., & Lilley, F. (2012). Two-Dimensional Phase Unwrapping Problem. 08/20/15, de Liverpool John Moores University.
11. Matlab 08/20/15, Site web: <http://www.mathworks.com/>
12. Bioucas-Dias, J. M., & Valadao, G. (2007). Phase unwrapping via graph cuts. *Image Processing, IEEE Transactions on*, 16(3), 698-709.
13. Singapore University of technology and design (ChenLujie) 08/20/15 <http://www.sutd.edu.sg/ChenLujie.aspx>



VISI Series México, S.A. de C.V.

Otorga el presente

Diploma

A: Ing. Rodrigo Escobar Díaz Guerrero

Por su participación en el :

CURSO DE ENTRENAMIENTO DE DIGITALIZACIÓN 3D VISI Series

Con duración de 36 horas. Impartido en _____

UNIVERSIDAD AUTÓNOMA DE QUERÉTARO

Ing. Elías Hernández Moreno
Instructor

Ing. Raúl Moreno Espinosa
Director General

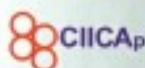
ICMEAE

**INTERNATIONAL CONFERENCE
ON MECHATRONICS, ELECTRONICS
AND AUTOMOTIVE ENGINEERING**



2015

Cuernavaca, Morelos, México.



Comparative analysis of phase unwrapping in PSP using depth images

Rodrigo Escobar Díaz Guerrero, Juan Carlos Moya Morales, Juan Manuel Ramos Arreguin, José Emilio Vargas Soto, Jesús Carlos Pedraza Ortega
 Universidad Autónoma de Querétaro, Facultad de Informática
 Querétaro, México
 rodediazg@gmail.com

Abstract— The main techniques in the optical measurement are Fourier Transform Profilometry (FTP), Wavelet Transform Profilometry (WTP) and Phase-Shifting Profilometry (PSP). Among them, PSP algorithms are the most widely used in optical measurements due to its numerous advantages as are point-by-point measurement less sensitive to variations in reflectivity surface (which makes it easier the measurement of complex objects with strong variations in texture), and less sensitivity to ambient light. This paper describes a novel approach to check the error in optical measurements, specifically PSP using firstly a simulation with depth images with all stages: pre-processing, phase wrapping and phase unwrapping. Finally, the results are checked with real objects. This research uses three-step and four-step method to establish a comparison. Moreover, the most common phase unwrapping algorithms are analyzed (The most important part to obtain tridimensional information of an object) applied to PSP.

Keywords— Phase wrapping; Phase unwrapping; Phase-Shifting profilometry; depth image; optical measurement.

1. INTRODUCTION

One of the best techniques used in the optical measurement is the fringe projection, this technique utilizes a defined fringe pattern that is projected to an object, therefore the object's shape creates distortions in the pattern, and then the distortions are captured with a digital camera from another angle and processed to obtain tridimensional information of an object.

There are many techniques from fringe projection the most common are Fourier Transform Profilometry (FTP), Wavelet Transform Profilometry (WTP) and Phase-Shifting Profilometry (PSP), each of the techniques provides diverse advantages and disadvantages with respect to the other [1][2][3]. In this research we only focus on the PSP technique. For instance, when only one image is processed commonly is used the Fourier transform which works as a spatial phase modulation of the pattern carrier with a fundamental frequency f_0 . Different phase demodulation algorithms can be applied to this carrier pattern. FTP extracts only the term of modulated phase through filtering spectrum, using one or two dimensions FTP and its inverse [1].

The PSP algorithms are widely used in optical measurements due to its numerous advantages, point-by-point measurement (which allows the resolution level pixel-camera), less sensitive to variations in reflectivity surface (facilitating the measurement of complex objects with strong variations in texture) and less sensitivity to ambient light [1] [3]. In this research only focus on the PSP technique.

The main methods of phase-shifting are: "three-step", "four-step" and "double three-step". These methods differ in number of images input, "three-step" will use 3 images, "four-step" will use 4 and so on. For obtain information of an object is necessary to find the phase; the phase contain the information of the optical measurement. While more images are employed will be better the quality of result, but the runtime will be worse.

Two basic stages to the phase treatment are necessary, which are: the phase wrapping and the phase unwrapping. The phase wrapping is the process that determines the values of phase in a range from 0 to 2π . The phase unwrapping is a process in which discontinuity of 2π is removed to generate a phase map [2] [4].

The phase wrapping can be expressed mathematically as:

$$x_w(n) = W[x(n)] \quad (1)$$

Where $x(n)$ is the original continuous phase, $W[\]$ is the phase wrapping operation and $x_w(n)$ is the phase wrapping [5].

Knowing that the phase unwrapping is an important part of any optical measurement numerous algorithms have been proposed to create better results in either response time or better resolution in measurements. For example "Itoh Traditional", "Itoh inverse", "Graph Cuts", "Quality guided", etc. [6].

II. PRINCIPLE OF THE METHOD

The PSP three-step used can be described as follows:

$$I_1(x, y) = I'(x, y) + I''(x, y) \cos[\phi(x, y)] \quad (2)$$

$$I_2(x, y) = I'(x, y) + I''(x, y) \cos[\phi(x, y) - \alpha] \quad (3)$$

$$I_3(x, y) = I'(x, y) + I''(x, y) \cos[\phi(x, y) + \alpha] \quad (4)$$

Where $I'(x, y)$ is the average intensity, $I''(x, y)$ is the intensity modulation, $\phi(x, y)$ is the phase to resolve [1][4][7][8]. In our case we consider $\alpha=2\pi/3$.

$$\phi(x, y) = \tan^{-1} \left(\sqrt{3} \frac{I_3(x, y) - I_2(x, y)}{2I_1(x, y) - I_2(x, y) - I_3(x, y)} \right) \quad (5)$$

The PSP four-step used can be described as follows:

$$I_1(x, y) = I'(x, y) + I''(x, y) \cos[\phi(x, y) + 3\alpha] \quad (6)$$

$$I_2(x, y) = I'(x, y) - I''(x, y) \sin[\phi(x, y) + 2\alpha] \quad (7)$$

$$I_3(x, y) = I'(x, y) - I''(x, y) \cos[\phi(x, y) + \alpha] \quad (8)$$

$$I_4(x, y) = I'(x, y) + I''(x, y) \sin[\phi(x, y)] \quad (9)$$

Where $I'(x, y)$ is the average intensity, $I''(x, y)$ is the intensity modulation, $\phi(x, y)$ is the phase to resolve. In our case we consider $\alpha = \pi/2$.

$$\phi(x, y) = \tan^{-1} \left(\frac{I_2(x, y) - I_3(x, y)}{I_1(x, y) - I_4(x, y)} \right) \quad (10)$$

To evaluate the process are used depth images, such images are in grayscale, while the object is closer then will have a pixel value higher and if object is far the pixel value is lower.

The complete process of phase-shifting profilometry is simulated as if it were real objects (using depth image) as shown in Figure 1.a. In Figure 1.a shown the images in which are projected defined patterns to an object, the image is captured, after angle pattern will be shifting (usually 120° in case of three-step and 90° in four-step algorithm), then re-capture the image and so until get 3 or more images, the images being processed by using (5) and (10) to obtain the phase map (contains the information of the height of the object), finally dimensional object is obtained by the phase unwrapping. Figure 1.b show one example using depth images where projected patterns are created artificially and is given the same pattern that used in images of real objects.

To calculate error is used the difference between the reconstructed image (pixel by pixel) and the three-dimensional representation of the depth image. In the depth image each pixel has a value of 0 (black) to 255 (white) which may be represented by a value from 0 to 1 (depth) to obtain the reconstructed image as in Figure 2.

More of 30 depth images were evaluated. At the beginning of research was used depth images with many smooth surfaces as Figure 3.a where the process was successful but the differences were not significant (for evaluation) because they are ideal images, therefore noise was added as Figure 3.f and yet the differences are not appreciably between techniques so was close to use simple images with abrupt borders as Figure 3.h.

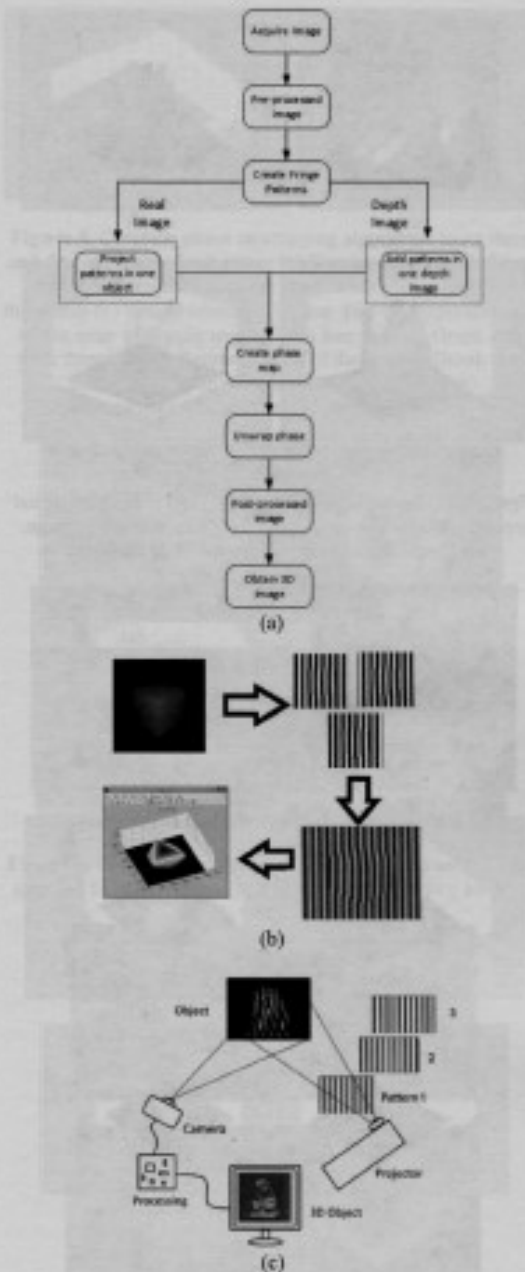


Figure 1. (a) Methodology. (b) An example using depth images. (c) An example using real images.

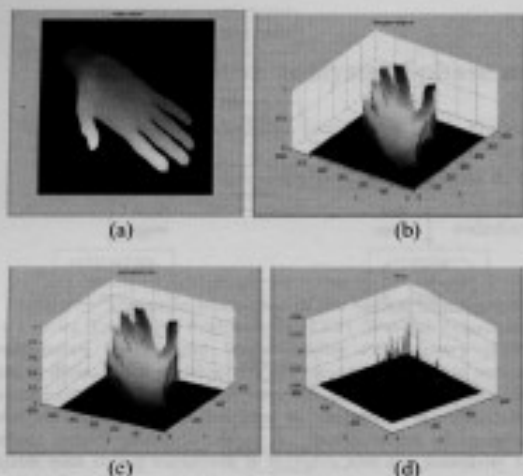


Figure 2. (a) Example of depth image, (b) Three-dimensional representation of the depth image, (c) Reconstructed image, (d) Three-dimensional representation of the error.

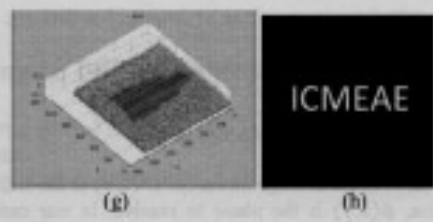
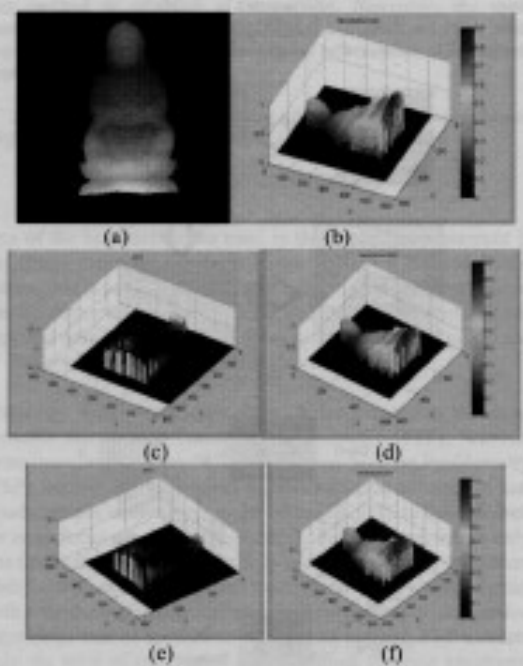


Figure 3. (a) Example of one depth image, (b) Reconstruction using three-step algorithm, (c) Error of reconstruction using three-step algorithm, (d) Reconstruction using four-step algorithm, (e) Error of reconstruction using four-step algorithm, (f) Reconstruction using three-step algorithm with noise, (g) Error of reconstruction using three-step algorithm, (h) Depth image used in this paper.

Each of phase unwrapping algorithms was evaluated with the two techniques used (three-step and four-step) and was calculated the maximum, minimum and average error (only with depth images) and runtime. The real images in this paper were taken of data raw of Chen Lujie [10] as Figure 4 because its features are ideal to observe a discontinuity error such as shadow effect inherent in projection, on-uniform reflectance but a good calibration.

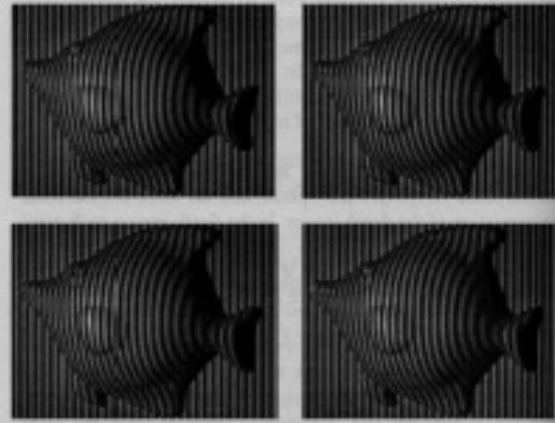


Figure 4. Image raw of Chen Lujie [10].

III. RESULTS

The Computer used to experiments is a Lenovo Y510p with 8GB of RAM, Intel® Core™ i7-4700MQ 2.4GHz and Windows 8.1 64 bits. The software implemented was made with Matlab R2013b.

A. Algorithms of phase unwrapping

Since there are many algorithms of phase unwrapping and the main authors differs which is more suitable, were analyzed the most common:

The 2D algorithm can only deal with one-dimensional phase unwrapping, which is not suitable for the 2D phase unwrapping. The 3D algorithm can deal with two-dimensional phase unwrapping, which is more suitable for the 3D phase unwrapping. The 3D algorithm can deal with two-dimensional phase unwrapping, which is more suitable for the 3D phase unwrapping. The 3D algorithm can deal with two-dimensional phase unwrapping, which is more suitable for the 3D phase unwrapping.

- **Simple unwrap:** This algorithm simply corrects the phase angles in a vector adding multiples of $\pm 2\pi$ when absolute jumps between consecutive elements [9].
- **Itoh Traditional:** The algorithm of Itoh involves unwrapping the rows one by one to verify discontinuity between the values of the row, followed by an unwrapping in columns (same one by one) [6].
- **Itoh inverse:** The algorithm of Itoh Inverse involves unwrapping the rows one by one to verify discontinuity between the values of the columns, followed by an unwrapping in row (same one by one) [6].
- **Graph Cuts:** The algorithm implemented for Phase Unwrapping via Graph Cuts use max-flow/min-cut calculations described by José M. Bioucas-Dia and Gonçalo Valadão in 2007 [11].

Figure 5 show a comparison between the Simple unwrap and graph Cuts each one with its respective representation of error using three and four-step with depth images, then in the Figures 6 and 7 shown the differences of all phase unwrapping algorithms to evaluate in this research using a real object.

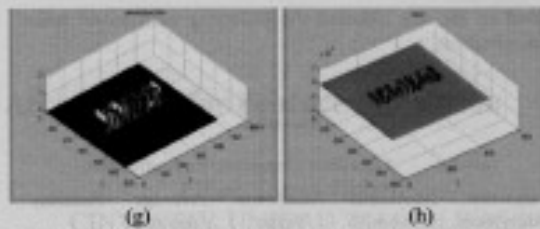
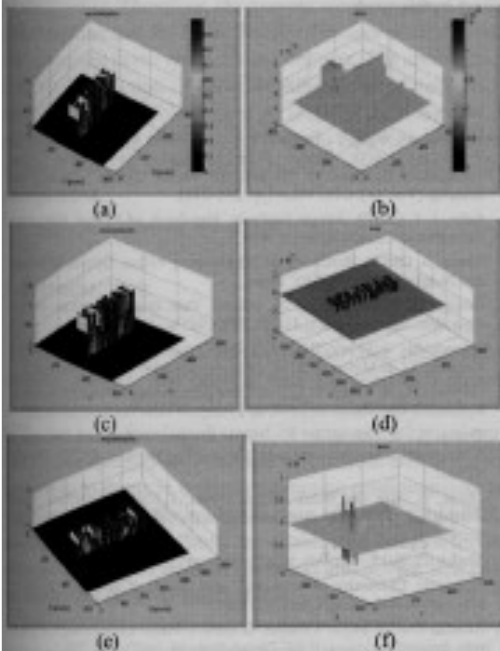


Figure 5. Common phase unwrapping algorithms using three and four-step with depth image (a) Simple unwrap with three-step (b) Representation of the error of Simple unwrap with three-step (c) Simple unwrap with four-step (d) Representation of the error of Simple unwrap with four-step (e) Graph cuts with three-step (f) Representation of the error of Graph cuts with three-step (g) Graph cuts with four-step (h) Representation of the error of Graph cuts with four-step

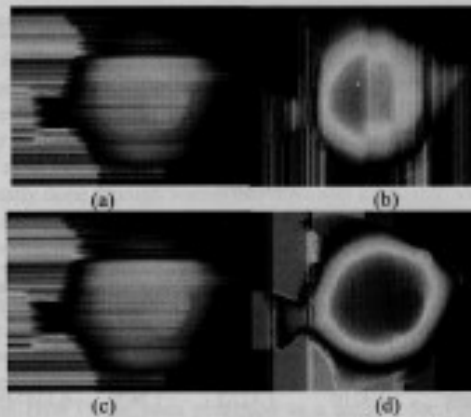


Figure 6. Common phase unwrapping algorithms using three-step, (a) Simple unwrap, (b) Itoh traditional, (c) Itoh inverse, (d) Graph cuts.

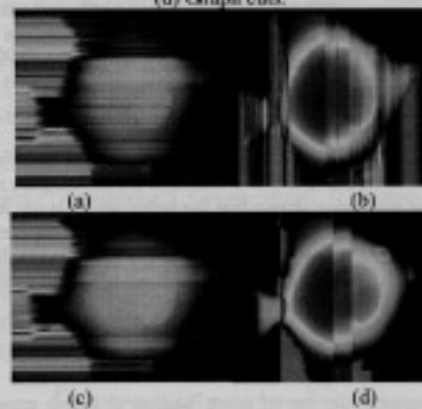


Figure 7. Common phase unwrapping algorithms using four-step, (a) Simple unwrap, (b) Itoh traditional, (c) Itoh inverse, (d) Graph cuts.

Based on the data obtained the following comparison tables were obtained:

Three-step

TABLE I. TABLE RESULT THREE-STEP WITH DEPTH IMAGE.

Unwrap Algorithm	Maximum error with 16 Fringes (%)	Minimum Error with 16 Fringes (%)	Average error With 16 Fringes (%)	Runtime
Simple unwrap	1.4210 $\times 10^{-12}$	1.7652 $\times 10^{-12}$	1.2957 $\times 10^{-12}$	0.090288 s
hoh Traditional	8.4876 $\times 10^{-12}$	7.5217 $\times 10^{-12}$	3.5376 $\times 10^{-12}$	0.134108 s
hoh Inverse	1.4210 $\times 10^{-12}$	1.4210 $\times 10^{-12}$	1.2670 $\times 10^{-12}$	0.138749 s
Graph Cut	1.076991 $\times 10^{-12}$	7.1054 $\times 10^{-12}$	1.9556 $\times 10^{-12}$	31.409889 s

Four-step

TABLE II. TABLE RESULT FOUR-STEP WITH DEPTH IMAGE.

Unwrap Algorithm	Maximum error with 16 Fringes (%)	Minimum Error with 16 Fringes (%)	Average error With 16 Fringes (%)	Runtime
Simple unwrap	1.4210 $\times 10^{-12}$	1.4210 $\times 10^{-12}$	1.2670 $\times 10^{-12}$	0.138749 s
hoh Traditional	0.0817	0.2545	1.4349 $\times 10^{-11}$	0.108028 s
hoh Inverse	0.0817	0.2545	1.4349 $\times 10^{-11}$	0.138513 s
Graph Cut	0.0817	0.25456	1.4349 $\times 10^{-11}$	23.0078 s

In the Table I. shown better results in Runtime like in different errors than In the Table II. But in the table II the results are more stable. The simulation shows that the better option is use three-step method with Graph cuts and after to evaluate the method with real images it can be inferred that the result is similar to simulation. So use a simulation is a good option to evaluate methods and algorithms applied to PSP.

IV. CONCLUSIONS AND FUTURE WORK

In this research, the use of depth images to assess errors in tridimensional reconstruction was presented, the proposed analysis is able to distinguish between methods and techniques

to choose and although it is not the same as having images of real objects, simulation method can prevent errors in implementation and evaluate the algorithm that will be applied.

Moreover it can be deduced that the three-step method is faster than the four-step method although not as stable (the error is constant) in most cases. The higher error in both techniques is in the edges especially when there is a sudden change of pixel to pixel.

As future work the reduction of the error after the phase unwrapping will be carried out by post-processing filtering or else by linear or polynomial interpolation also the implementation of a best algorithm in an embedded system is proposed.

ACKNOWLEDGMENT

Thank to "Consejo Nacional de Ciencia y Tecnología" (CONACYT) for their support and sponsorship right through the National Scholarship also we would like to thank "Universidad Autónoma de Querétaro" and "Fondo de proyectos especiales de rectoría" (FOPER UAQ-2015), for the support in this work.

REFERENCES

- [1] Ekstrand, L., Wang, Y., Karpinsky, N., and Zhang, S. (2013). Superfast 3D Profilometry with Digital Fringe Projection and Phase-Shifting Techniques, *Handbook of 3D Machine Vision: Optical Metrology and Imaging* (Vol. 16). CRC Press (pp. 233-252).
- [2] Kato, J. I. (2005). Fringe Analysis, *Handbook of Optical Metrology: Principles and Applications*, 1, 499.
- [3] Zhang, S. (2013). *Handbook of 3D Machine Vision: Optical Metrology and Imaging* (Vol. 16). CRC Press.
- [4] Huang, P. S., & Zhang, S. (2006). Fast three-step phase-shifting algorithm. *Applied optics*, 45(21), 3086-3091.
- [5] Gdeisat, M., & Lilley, F. (2012). One-Dimensional Phase Unwrapping Problem. 24 de Marzo de 2014, de Liverpool John Moores University.
- [6] Gdeisat, M., & Lilley, F. (2012). Two-Dimensional Phase Unwrapping Problem. 24 de Marzo de 2014, de Liverpool John Moores University.
- [7] Zhang, S., & Huang, P. S. (2006). High-resolution, real-time three-dimensional shape measurement. *Optical Engineering*, 45(12), 123601-123601.
- [8] Zhang, S., Van Der Weide, D., & Oliver, J. (2010). Superfast phase-shifting method for 3-D shape measurement. *Optics express*, 18(9), 9684-9689.
- [9] Matlab-03/2015. Site web: <http://www.mathworks.com/>
- [10] Singapore University of technology and design (Charulajia) 03/03/15 <http://www.sutd.edu.sg/Charulajia.npa>
- [11] Biresani-Dias, J. M., & Valério, G. (2007). Phase unwrapping via graph cuts. *Image Processing, IEEE Transactions on*, 16(3), 698-704.