

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

Facultad de Ciencias Químicas e Ingeniería

Maestría y Doctorado en Ciencias e Ingeniería



Modelado difuso para la recuperación de la fase en imágenes por proyección de franjas

TESIS

QUE PARA OBTENER EL GRADO DE

MAESTRO EN CIENCIAS

Presenta:

BLANCA LILIA SANTIAGO PABLO

Bajo la dirección de:

DR. JUAN RAMÓN CASTRO RODRÍGUEZ

Co-dirigido por:

DRA. ADRIANA NAVA VEGA

TIJUANA, BAJA CALIFORNIA, MÉXICO

ABRIL 2017

A mis padres que tanto amo y admiro, Antonio Santiago Ortiz y Herminia Pablo Hernández dedico este escrito, que gracias a su apoyo, motivación y amor he podido dar otro paso adelante.

Universidad Autónoma de Baja California
FACULTAD DE CIENCIAS QUÍMICAS E INGENIERÍA
COORDINACIÓN DE POSGRADO E INVESTIGACIÓN

FOLIO No. 204

Tijuana, B. C., a 30 de marzo de 2017

C. Blanca Lilia Santiago Pablo
Pasante de: Maestro en Ciencias
Presente

El tema de trabajo y/o tesis para su examen profesional, en la
Opción TESIS

Es propuesto, por los CC. Dres. Juan Ramón Castro Rodríguez y Adriana Nava Vega

Quienes serán los responsables de la calidad del trabajo que usted presente, referido al tema: Modelado difuso para la recuperación de la fase en imágenes por proyección de franjas.

el cual deberá usted desarrollar, de acuerdo con el siguiente orden:

- I.- INTRODUCCIÓN
- II.- MARCO TEÓRICO
- III.- DESARROLLO
- IV.- PRUEBAS Y RESULTADOS
- V.- CONCLUSIONES Y TRABAJO FUTURO
- VI.- ANEXOS

UNIVERSIDAD AUTÓNOMA
DE BAJA CALIFORNIA



FACULTAD DE CIENCIAS
QUÍMICAS E INGENIERÍA

Dr. Juan Ramón Castro Rodríguez
Director de Tesis

Dr. José Luis González Vázquez
Secretario

Dra. Adriana Nava Vega
Co-Directora de tesis

Dr. Luis Enrique Palafox Maestre
Director

Agradecimientos

Deseo agradecer principalmente al *Consejo Nacional de Ciencia y Tecnología (CONACYT)* por brindarme el recurso económico que me fue de gran utilidad para iniciar y concluir con este proceso, mi maestría. Agradezco inmensamente a mi director de tesis, el Doctor *Juan Ramón Castro Rodríguez* que me fue de gran apoyo y guía durante mi investigación, y por compartirme excelentes conocimientos. Agradezco también a mi coodirectora, la Doctora *Adriana Nava Vega*, por la confianza, paciencia y tiempo que me brindó y por los nuevos conocimientos que me otorgó basados en la parte práctica.

Agradezco a la Doctora *Olivia Mendoza Duarte* por esas tan detalladas explicaciones que me proporcionó en las clases y que me fue de grandísima ayuda; agradezco a mis compañeros, que mucho admiro, Iztel Barriba, Mauricio Sánchez, Evangelina Herrán, Gabriela Martínez, Claudia Gonzales que fueron mi apoyo y motivación para poder concluir este paso.

Índice general

1. Introducción	II
1.0.1. Planteamiento del Problema:	III
1.0.2. Hipótesis	IV
1.0.3. Objetivos	IV
1.0.4. Aplicaciones	IV
2. Marco teórico	1
2.1. Importancia de la Fase	1
2.1.1. Obtención de la Fase mediante el análisis de Patrón de Franjas de Fourier	2
2.1.2. Método de Itoh para la Desenvoltura de la Fase Unidimensional	4
2.1.3. Algoritmos locales para la desenvoltura de la fase	5
2.1.4. Algoritmos de Desenvoltura por rutas de Calidad	8
2.1.5. Mapas de calidad Bidimensional	9
2.1.6. Mapas de Calidad de Pseudo Correlación	10
2.1.7. Mapa de calidad de la Segunda Diferencia	10
2.1.8. Algoritmos de desenvoltura de fase global	11
2.1.9. Mínimos cuadrados no ponderados	12
2.1.10. Mínimos cuadrados ponderados	14
2.1.11. Norma L^p	15

2.2. Sistemas de Lógica Difusa Tipo-1	16
2.2.1. Lógica difusa como auxiliar para la desenvoltura de la fase	18
2.2.2. Sistema de Inferencia Takagi-Sugeno-Kang	18
3. Desarrollo	21
3.1. Modelo difuso para el proceso de la Desenvoltura de la fase Envuelta	21
3.2. Metodología A	22
3.2.1. Variable Lingüística	22
3.2.2. Funciones de Membresía	22
3.2.3. Reglas Difusas	24
3.3. Metodología B	24
3.3.1. Funciones de Membresía	25
3.3.2. Reglas Difusas	27
3.3.3. proceso de recuperación de la fase	27
4. Pruebas y resultados	30
4.1. Resultados Obtenidos con la Metodología A	30
4.2. Resusultados Obtenidos con la Metodología B	35
4.2.1. Aplicación	40
5. Conclusiones y trabajo futuro	43
5.0.2. Conclusiones	43
5.0.3. Trabajo Futuro	44
6. Anexos	45
6.1. ALGORITMOS	46

Índice de figuras

2.1. (a) Montaje experimental para el plano. (b) Montaje experimental para el objeto	2
2.2. Espectros de frecuencias del patrón de franjas	3
2.3. Frecuencia trasladada	4
2.4. Bordes y pixeles	9
2.5. Sistema Difuso	17
2.6. Inferencia difusa de tipo Sugeno	19
2.7. proceso de Inferencia	19
2.8. superficie generada por el modelo difuso	20
3.1. Funciones de Membresia para la variable de entrada	24
3.2. Sistema Difuso para la detección de asíntotas	26
4.1. (a) Imagen Picos,(b) Imagen Picos vista en 3D	30
4.2. (a)entrada al modelo visto como un arreglo de intensidades,(b) salida inicial del modelo	31
4.3. (a)Matriz de entrada vista como una imagen, (b) Matriz de salida vista como una imagen	31
4.4. (a)fase envuelta,(b)fase desenvuelta por renglones,(c)fase desenvuelta por renglones y columnas	32
4.5. (a)fase verdadera, (b), (c) y (d) fase obtenida vista desde diferentes ángulos	32
4.6. (a) Imagen Escalon,(b) Imagen Escalon vista en 3D	32
4.7. (a)entrada del modelo visto como imagen, (b)salida del modelo visto como imagen	33

4.8. (a)fase Envuelta, (b)fase corregida por renglones, (c) fase corregida por renglones y columnas .	33
4.9. (a) objeto de prueba, (b),(c) y (d) objeto obtenido visto desde diferentes ángulos	33
4.10. (a) Imagen Espiral,(b) Imagen Espiral vista en 3D	34
4.11. (a)entrada del modelo visto como imagen, (b)salida del modelo visto como imagen	34
4.12. (a)fase Envuelta, (b)fase corregida por renglones, (c) fase corregida por renglones y columnas .	34
4.13. (a) objeto de prueba, (b),(c) y (d) objeto obtenido visto desde diferentes ángulos	35
4.14. (a)entrada al modelo visto como un arreglo de intensidades,(b) salida inicial del modelo	35
4.15. (a)Fase corregida por renglones, (b)Diferencias de fases por columnas	35
4.16. (a)Salida del modelo, (b)fase corregida por renglones y columnas	36
4.17. (a)fase envuelta, (b)fase corregida por renglones, (c) fase corregida por renglones y columnas .	36
4.18. (a)fase envuelta, (b)fase corregida por renglones, (c) fase corregida por renglones y columnas .	36
4.19. Regresión Lineal de la fase verdadera vs la fase obtenida de la imagen picos	37
4.20. (a)Matriz de entrada al modelo difuso, (b)Matriz de salida del análisis por renglones, (c)Fase corregida por renglones	37
4.21. (a)Diferencia de Fases por columnas,(b)Salida del análisis por columnas,(c)Fase corregida por columnas y renglones	37
4.22. (a),(b),(c) y (d) objeto obtenido del modelo visto en 3D desde diferentes ángulos	38
4.23. Regresión Lineal de la fase verdadera vs la fase obtenida de la imagen escalon	38
4.24. (a)entrada del modelo visto como una imagen,(b) Salida del modelo del análisis por renglo- nes,(c)Corrección de fases por renglones	38
4.25. (a)Diferencias de fases por columnas,(b) Salida del modelo del análisis por columnas,(c)Corrección de fases por renglones y columnas	39
4.26. (a),(b)y (c) Objeto obtenido por el modelo difuso visto desde diferentes ángulos	39
4.27. Regresión Lineal de la fase verdadera vs la fase obtenida de la imagen espiral	39
4.28. Objeto de prueba: porción del hueso de un pez marlin	40

4.29. (a)y (b) Plano de referencia y Objeto con proyección de franjas cosenoidales, respectivamente	40
4.30. (a) Fase desenvuelta por renglones de Plano Referencia, (b) Fase desenvuelta por renglones y columnas	41
4.31. (a) Fase desenvuelta por renglones de imagen Hueso, (b) Fase desenvuelta por columnas de Imagen Hueso	41
4.32. Fase de la imagen de pez marlin visto como un arreglo de intensidades	42

Capítulo 1

Introducción

Durante años, los procesos naturales y biológicos han atraído la atención de los investigadores, debido a que estos sistemas tienen la capacidad de responder de forma autónoma a los cambios que el ambiente presenta. Algunos de los procesos que se han analizado son: la evolución humana, el sistema nervioso humano, psicología, física, inteligencia animal, entre otros. Dando como resultados diversos algoritmos, que si bien no simulan completamente el sistema del que se habla, simulan alguna de las partes más importantes basándose en los principios naturales.

Algunas de las técnicas que se han creado son: algoritmos genéticos, redes neuronales, aprendizaje reforzado, colonias de hormigas, etcétera. La computación inspirada en la naturaleza ha surgido, como un campo de investigación que involucra el uso de técnicas inspiradas en la biología y en fenómenos físicos naturales, con el fin de resolver problemas computacionales y del mundo real. El creciente interés que ha despertado este campo de investigación, se encuentra en el hecho de que hoy en día se presentan sistemas que tienden a ser, complejos, grandes, distribuidos y generalmente mal estructurados.

Por otro lado, la comunidad científica ha notado que las estructuras y organizaciones simples en la naturaleza, son capaces de resolver los problemas más complejos con facilidad.

Las técnicas computacionales inspiradas en la naturaleza han probado que pueden ser aplicadas ampliamente en diversas áreas, mostrando resultados favorables.

Estos algoritmos poseen diversas características como son: optimización, reconocimiento de patrones, máquinas de aprendizaje, y por esto han atraído la atención de los investigadores en el área de visión computacional. Actualmente una gran cantidad de estos métodos son aplicados para solucionar los grandes problemas que presenta la visión por computadora, uno de ellos es el análisis de imágenes reales.

La Perfilometría de Fourier [1] es una técnica empleada para determinar la fase de imágenes de objetos reales, la fase conlleva a la topología tridimensional de los objetos, los cuales pueden ser de diversos tamaños; debido a la precisión que nos proporciona el trabajo con luz, lográndose obtener detalles muy finos, del orden de la longitud de onda empleada (400 a 700 nm).

Existen diversos métodos de optimización para determinar el valor de la fase de una imagen, el mapa de fases se puede interpretar como la matriz de valores cuyos elementos contienen implícitamente la altura de cada punto del objeto. Ésta matriz junto con arreglos experimentales y optimización no lineal multivariable se logra generar una representación tridimensional del objeto. [2], [3],[4],[5]

En este trabajo de tesis, se propone construir modelos de optimización utilizando herramientas de Lógica Difusa para la obtención de la fase absoluta de imágenes sintéticas y reales.

1.0.1. Planteamiento del Problema:

Para la reconstrucción de imágenes ópticas utilizando el método de perfilometría empleando proyección de franjas, se llevan a cabo dos procesos muy importantes: la extracción de la fase y la desenvoltura de la fase conocida también como desdoblamiento de la fase. El método de Perfilometría de Fourier es un algoritmo robusto utilizado para la extracción de la fase de imágenes de objetos, pero resulta que al implementar éste algoritmo se obtienen valores acotados en un rango de $-\pi$ a π .

La finalidad de obtener los valores verdaderos de la fase en las imágenes es que mediante ellos se pueden construir las formas tridimensionales de los objetos capturados en las imágenes, por lo tanto es necesario procesar los primeros datos obtenidos de la fase. El nombre que recibe la matriz que contiene a los valores acotados de $-\pi$ a π es: *matriz de fases envueltas* y al proceso que se lleva a cabo para obtener los valores

verdaderos se le nombra *desenvoltura de la fase*.

Sin embargo, el proceso de desenvoltura de la fase es una tarea difícil de manejar al momento de querer eliminar las discontinuidades de 2π que se generan. Los algoritmos que existen para la desenvoltura de fase actualmente compiten por el costo computacional y por la calidad de la imagen obtenida por proyección de franjas.

1.0.2. Hipótesis

Al diseñar algoritmos basados en Lógica Difusa se resolverá eficazmente la tarea del procesamiento de la desenvoltura de la fase, ya que mediante las percepciones del experto se pueden considerar y estudiar los casos de una posible discontinuidad o envoltura de fase en una imagen.

1.0.3. Objetivos

Generales

Introducir sistemas difusos en los algoritmos ya desarrollados de desenvoltura de fase para mejorar el mapa de fase de la imagen, y, diseñar un modelo de optimización de acuerdo al análisis que se debe realizar, en función del tamaño del objeto y precisión a obtener.

Particulares

1.-Construir un modelo difuso que desenvuelva la fase de cualquier matriz envuelta, sin importar el tamaño de ésta.

2.-Utilizar el modelo difuso para obtener la fase absoluta de imágenes reales.

1.0.4. Aplicaciones

Las aplicaciones inmediatas de este tema de tesis se centran en la determinación de la morfología de material biológico, tales como huesos de pez espada, otocitos, prótesis, de las cuales se quiere conocer diversos parámetros, como edad, contaminantes, rutas de migración, porosidad, etc.

También tiene diversas aplicaciones en el procesamiento digital de señales como Radares de Apertura Sintética (SAR, por sus siglas en inglés), Imágenes de Resonancia Magnética (MRI, por sus siglas en inglés), Interferometría óptica, Procesamiento sísmico, proyección y difracción de Tomografía, aplicaciones médicas, física de fluídos, entre otros.[3],[6],[7],[8],[9],[10],[5],[11],[12]

Investigadores de la Universidad de Liverpool John Moores han desarrollado diversos algoritmos para el proceso de la recuperación de la fase con la finalidad de aplicarlos a distintas áreas médicas o comerciales.[13],[14]

Capítulo 2

Marco teórico

2.1. Importancia de la Fase

Como se menciona en [16] en cualquier proceso donde se utiliza un análisis de señales mediante la transformada de Fourier la fase es importante puesto que guarda información sobre el desplazamiento que recorre la señal.

Alguna de las aplicaciones son: el procesamiento digital de imágenes, el radar de apertura sintética, entre otros.

La perfilometría por proyección de franjas es una de las técnicas de medición sin contacto del perfil de algún objeto, y es una técnica no invasiva ya que no hay desgaste físico ni degradación de lo que se mide. Mediante esta técnica se logra conocer la fase del objeto capturado, la cual guarda información de la altura del objeto de interés.

La ventaja de ser una técnica sin contacto es que la pieza de prueba (pudiendo ser frágil) no se obstruirá en el proceso de medición.

La forma en que se realiza el proceso de la medición del perfil del objeto es proyectando un patrón de franjas sinusoidales sobre él mediante una rejilla de ronchi, el objeto deforma las franjas de acuerdo a la que éste mismo tiene, para después capturar la imagen con una cámara CCD (del inglés Charge-Coupled Devi-

ce) y enviarlas a la memoria de la computadora, y así iniciar con el procesamiento de la imagen.

En la figura 2.1 se muestran los esquemas del montaje experimental usando la técnica de perfilometría, considerando que se requiere contar con un plano de referencia al cual también se le aplicará el método para la obtención de la fase con la intención de que al desenvolver las fases a ambos la diferencia de ellos nos dará lugar a la fase absoluta del objeto y por lo tanto a las dimensiones de él:

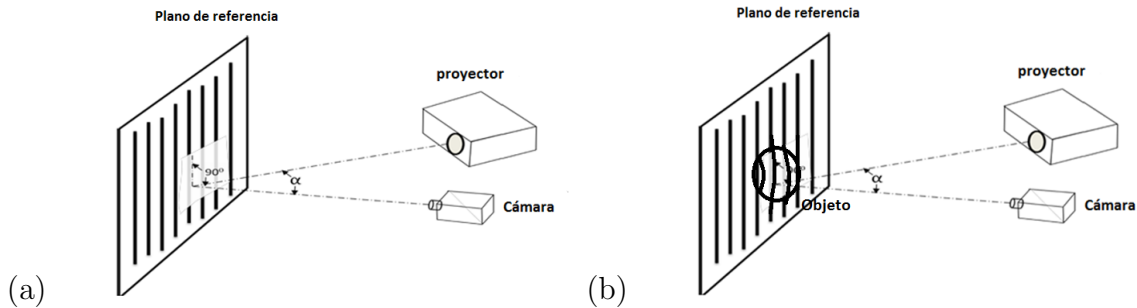


Figura 2.1: (a) Montaje experimental para el plano. (b) Montaje experimental para el objeto

2.1.1. Obtención de la Fase mediante el análisis de Patrón de Franjas de Fourier

El método de Perfilometría de Fourier consiste en el análisis de imágenes con un patrón de franjas utilizando la Transformada Rápida de Fourier con la finalidad de conocer información, por ejemplo las medidas físicas de algún objeto de interés. Este método principalmente se encarga de calcular la fase de la imagen y es uno de los métodos más comunes para la obtención de la fase en un patrón de franjas, además sólo se requiere una sola imagen, comparado con los demás métodos que requieren más de dos imágenes para la obtención de la fase.[17],[18],[1]

Se puede definir una imagen con un patrón de franjas con la siguiente ecuación matemática:

$$g(x) = a(x, y) + b(x, y)\cos[2\pi f_0 x + \phi(x, y)] \quad (2.1.1)$$

donde $a(x, y)$ es el brillo o contraste de la imagen y $b(x, y)$ es la demodulación de las franjas.

La ecuación 2.1.1 se puede reescribir empleando la fórmula de Euler como:

$$g(x, y) = a(x, y) + c(x, y)e^{2\pi i f_0 x} + \bar{c}(x, y)e^{-2\pi i f_0 x} \quad (2.1.2)$$

donde $c(x, y) = \frac{1}{2}b(x, y)e^{i\phi(x, y)}$ y \bar{c} es el complejo conjugado.

Aplicando la transformada de Fourier con respecto a x a la ecuación 2.1.2 se obtiene:

$$G(f_x, y) = A(f_x, y) + C(f_x - f_0, y) + \bar{C}(f_x + f_0, y) \quad (2.1.3)$$

Tal que las letras en mayúsculas representan al espectro de Fourier y f_x a la frecuencia espacial en la dirección x .

Se puede utilizar cualquiera de los dos aspectos de la frecuencia portadora C o \bar{C} como se observa en la figura 2.2 para trasladarla al origen, en este caso se eligió $C(f_x - f_0, y)$ y desplazándolo hacia el origen una cantidad f_0 , se obtiene $C(f_x, y)$ como se muestra en la figura 2.3.

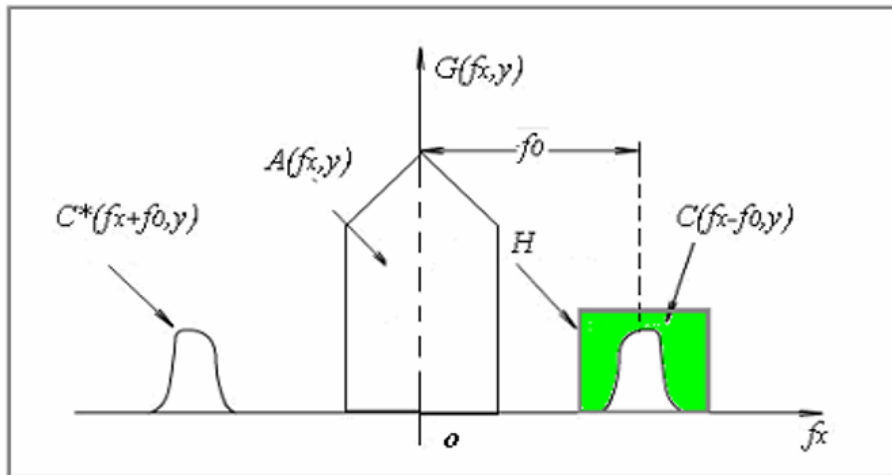


Figura 2.2: Espectros de frecuencias del patrón de franjas

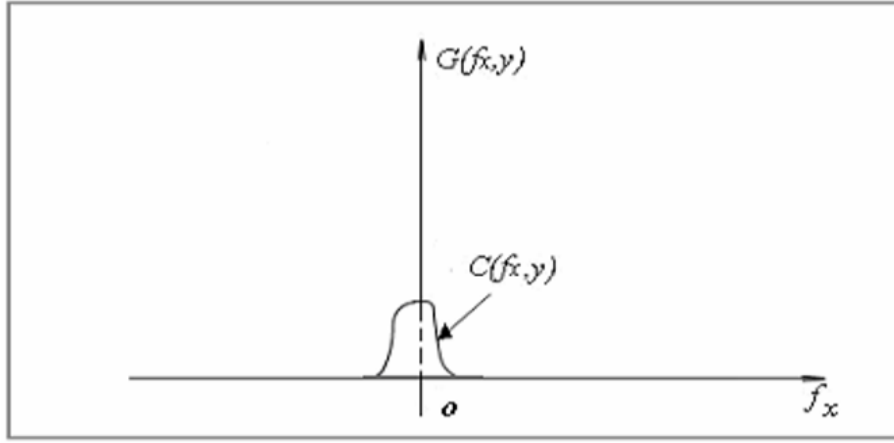


Figura 2.3: Frecuencia trasladada

Transformando el término $C(f_x, y)$ a su dominio espacial original empleando la transformada inversa de Fourier unidimensional se obtiene nuevamente $c(x, y)$:

Tal que la parte real e imaginaria del término $c(x, y)$ son:

$$\begin{aligned}\Re\{c(x, y)\} &= b(x, y)\cos(\phi(x, y)) \\ \Im\{c(x, y)\} &= b(x, y)\sin(\phi(x, y))\end{aligned}\tag{2.1.4}$$

y el valor de la fase se puede extraer mediante la siguiente ecuación:

$$\psi(x, y) = \tan^{-1} \frac{\Im\{c(x, y)\}}{\Re\{c(x, y)\}} = \phi(x, y) \bmod \pi\tag{2.1.5}$$

2.1.2. Método de Itoh para la Desenvoltura de la Fase Unidimensional

La ecuación 2.1.5 es la forma matemática para obtener la fase, la cual queda acotada o envuelta entre $-\pi$ a π , uno de los principales algoritmos para recuperar la fase, en el cual se basa la mayoría y se describirá a continuación.

El siguiente algoritmo fue obtenido del libro de Ghiglia and Pritt [16] mediante el cual se han desarrollado nuevos métodos para mejorar el proceso de la desenvoltura de la fase:

Paso 1. Calcula las diferencias de fase: $D(i) = \psi(i + 1) - \psi(i)$ para $i = 0, 1, \dots, N - 2$

Paso 2. Calcula las diferencias de las fases envueltas: $\Delta(i) = \arctan\{\sin D(i), \cos D(i)\}$, para $i = 0, 1, \dots, N - 2$

Paso 3. Inicializa el primer valor desenvuelto: $\phi(0) = \psi(0)$.

Paso 4. Desenvuelve sumando las diferencias de las fases desenvueltas: $\phi(i) = \phi(i - 1) + \Delta(i - 1)$, para $i = 1, 2, \dots, N - 1$.

El algoritmo de Itoh ha sido la base para construir nuevos algoritmos de optimización para la recuperación de la fase, éstos algoritmos se han clasificado en dos tipos y son los más populares:

1.-Algoritmos locales

2.-Algoritmos globales

Los cuales se describirán a continuación:

2.1.3. Algoritmos locales para la desenvoltura de la fase

La modalidad que utilizan las técnicas de desenvoltura local de la fase es encontrar un camino de seguimiento para desenvolver el mapa de fase envuelto por medio de esa ruta. La desenvoltura local de la fase es definida matemáticamente como:

$$\phi(b) = \phi(a) + \sum_{i=a}^{b-1} \Delta(\phi(i)) \quad (2.1.6)$$

Donde $\phi(b)$ es la fase desenvuelta en algún punto b , $\phi(a)$ es la fase desenvuelta en un punto inicial a , y Δ es el operador de diferencia tal que:

$$\Delta\{\phi(n)\} = \phi(n + 1) - \phi(n) \quad (2.1.7)$$

La ecuación 2.1.6 indica que la desenvoltura de la fase en un punto b , se define como la suma de la fase desenvuelta en un punto a más la integración de las diferencias de fases desenvueltas, a través de una ruta desde el punto a al punto b . Sin embargo, es una ecuación que puede ser aplicada a cualquier función *continua*, motivo por el cual no se puede usar directamente para obtener la solución del mapa de fases

desenvueltas y es necesario modificarla.[17]

Los valores de la fase envuelta y desenvuelta se pueden relacionar mediante la siguiente ecuación: [17],[16]

$$\psi(n) = \phi(n) + 2\pi * k(n) \quad (2.1.8)$$

$$-\pi < \psi(n) \leq \pi$$

o bien

$$\phi(n) = \psi(n) + 2\pi * \nu(n) \quad (2.1.9)$$

$$-\infty < \phi(n) < \infty$$

Tal que

$\psi(n)$ son los valores de fases envueltas

$\phi(n)$ son los valores de fases desenvueltas

$k(n)$ es la función que contiene a los números enteros que son agregados a la fase desenvuelta ϕ para que devuelva valores entre $(-\pi, \pi)$.

$\nu(n)$ es la función de valores enteros que deben de ser agregados a la fase envuelta ψ para obtener la fase desenvuelta ϕ . Cabe observar que $\nu(n) = -k(n)$.

Se define a ω como el operador *envoltura* que convierte a los valores verdaderos de la fase a valores acotados entre $(-\pi, \pi)$, matemáticamente se representa:

$$\omega\{\phi(n)\} = \arctan\left[\frac{\sin(\phi(n))}{\cos(\phi(n))}\right] \quad (2.1.10)$$

o

$$\omega\{\phi(n)\} = \psi(n) = \phi(n) + 2\pi * k(n) \quad (2.1.11)$$

Donde la diferencia de fases envueltas se definen como:

$$\Delta\{\psi(n)\} = \psi(n+1) - \psi(n) \quad (2.1.12)$$

Sustituyendo la ecuación 2.1.8 en la ecuación 2.1.11

$$\Delta\{\psi(n)\} = \Delta\phi(n) + 2\pi * \Delta k(n) \quad (2.1.13)$$

Aplicando el operador de envoltura ω a las diferencias de fases envueltas (ecuación 2.1.13) tenemos:

$$\begin{aligned} \omega\{\Delta(\psi(n))\} &= \arctan \left\{ \frac{\sin(\Delta(\psi(n)))}{\cos(\Delta(\psi(n)))} \right\} \\ &= \arctan \left\{ \frac{\sin(\Delta(\phi(n))) + 2\pi\Delta k(n)}{\cos(\Delta(\phi(n))) + 2\pi\Delta k(n)} \right\} \\ &= \arctan \left\{ \frac{\sin(\Delta\phi(n))}{\cos(\Delta\phi(n))} \right\} \\ &= \omega\{\Delta\phi(n)\} \end{aligned} \quad (2.1.14)$$

Entonces

$$\omega\{\Delta(\psi(n))\} = \omega\{\Delta\phi(n)\} \quad (2.1.15)$$

Cuando no hay sub o sobremuestreo de fases:

$$\text{Si } |\Delta\phi(n)| < \pi, \text{ entonces } \omega[\Delta\phi(n)] = \Delta\phi(n) \quad (2.1.16)$$

Entonces

$$\omega\{\Delta(\psi(n))\} = \Delta\phi(n) \quad (2.1.17)$$

Sustituyendo la ecuación de la fase envuelta 2.1.11 en la ecuación 2.1.6:

$$\phi(b) = \phi(a) + \sum_{i=a}^{b-1} \omega\{\Delta(\psi(i))\} \quad (2.1.18)$$

Donde:

$\phi(b)$ es la fase desenvuelta en el punto b .

$\phi(a)$ es la fase desenvuelta en un punto inicial arbitrario a , considerando que la fase desenvuelta de a es igual a su fase envuelta.

La ecuación 2.1.18 establece que para los algoritmos de desenvoltura local de la fase, se considera a la desenvoltura como una integración de las diferencias de las fases envueltas a lo largo de un camino desenvuelto.[17]

Al usar la ecuación 2.1.18 para desenvolver la fase, el error se propaga desde un punto que no fue correctamente desenvuelto hasta el último elemento del mapa de fases. En la actualidad, varios investigadores han desarrollado nuevos métodos para evitar la propagación del error, los cuales consisten en crear rutas que aíslan las posibles regiones de desconfianza con el fin de minimizar el error de propagación en el mapa de fases desenvueltas. Algunos ejemplos de entre los más populares de este tipo de algoritmos son:

- 1.-Algoritmos de desenvoltura por rutas de calidad
- 2.-Mapas de calidad Bidimensionales

2.1.4. Algoritmos de Desenvoltura por rutas de Calidad

Este tipo de algoritmos dependen de una medida de calidad la cual traza una ruta a seguir para realizar la desenvoltura. El objetivo es desenvolver la fase partiendo del pixel de mayor calidad hasta llegar al pixel de menor calidad y así evitar la propagación del error.

El mapa de calidad puede ser definido como un arreglo de valores que contiene la calidad de cada pixel del mapa de fases, éste mapa de calidad es utilizado para desenvolver la fase envuelta donde sólo aquellos pixeles que sobrepasen cierto valor de umbral son desenvueltos, mientras que los demás no se desenvuelven

hasta que sobrepasen el valor del umbral, el cual cambia en cada iteración.

El doctor Miguel Arevalillo-Herráez [19] fue el primero en introducir el uso de la calidad de los bordes para construir una ruta de desenvoltura.

Un borde se puede definir como la línea que conecta a dos pixeles adjuntos como se muestra en la figura 2.4, éste método de medir la calidad de los bordes ha tenido gran éxito para el proceso de la desenvoltura de la fase.

La desventaja de este algoritmo es que al crear un camino para desenvolver la fase, éste no aisle un residuo desbalanceado y agregue una asíntota falsa.

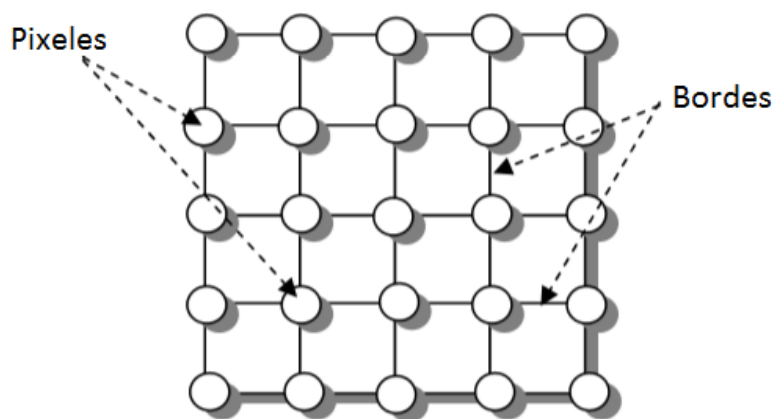


Figura 2.4: Bordes y píxeles

2.1.5. Mapas de calidad Bidimensional

El objetivo de utilizar los mapas de calidad en el proceso de la desenvoltura es encontrar un mapa confiable que indique el camino a seguir para desenvolver la fase y además que otorgue información particular de cada pixel del mapa de fases envueltas. Éstos mapas pueden ser calculados usando diferentes métodos, entre ellos los más comunes son la segunda diferencia, la varianza de la derivada de la fase, gradientes máximos de la fase y técnicas de pseudo correlación.[16]

2.1.6. Mapas de Calidad de Pseudo Correlación

El método de pseudo-correlación utiliza la información del mapa de fases envueltos para estimar la calidad de cada pixel [16],[17]. Este método calcula lo mejor que hay en cada pixel mediante la siguiente ecuación:

$$C_{\psi_{i,j}} = \frac{1}{k^2} * \sqrt{(\sum \cos \Delta(\psi))^2 + (\sum \sin \Delta(\psi))^2} \quad (2.1.19)$$

tal que k es el tamaño de la ventana, el cual tiene que ser impar y recorrerá toda la matriz de fases envueltas, $\Delta(\psi)$ son los $(k * k) - 1$ pixeles de la ventana situada alrededor del pixel a analizar, y $\psi_{i,j}$ son los valores de la matriz de fases envueltas.

En la siguiente tabla se muestra el pixel de interés $\psi_{i,j}$ y sus vecinos de la ventana de tamaño de 3×3 .

$\psi_{i-1,j-1}$	$\psi_{i-1,j}$	$\psi_{i-1,j+1}$
$\psi_{i,j-1}$	$\psi_{i,j}$	$\psi_{i,j+1}$
$\psi_{i+1,j-1}$	$\psi_{i+1,j}$	$\psi_{i+1,j+1}$

Tabla 2.1: Pixeles vecinos de una ventana de 3×3

Como se observa en la tabla 2.1, existen 8 pixeles vecinos de $\psi_{i,j}$ representados como:

$\Delta(\psi) = \{(\psi_{i,j} - \psi_{i-1,j-1}), (\psi_{i,j} - \psi_{i-1,j}), (\psi_{i,j} - \psi_{i-1,j+1}), (\psi_{i,j} - \psi_{i,j-1}), (\psi_{i,j} - \psi_{i,j+1}), (\psi_{i,j} - \psi_{i+1,j-1}), (\psi_{i,j} - \psi_{i+1,j}), (\psi_{i,j} - \psi_{i+1,j+1})\}$, los cuales proporcionarán información específica acerca de la calidad $C_{\psi_{i,j}}$ del pixel de interés.

2.1.7. Mapa de calidad de la Segunda Diferencia

Según los investigadores [17],[19], éste algoritmo es más robusto que los otros ya mencionados y hasta el momento es uno de los mejores mapas de calidad. Mide la buena y mala calidad de cada pixel central en ventanas de $k * k$ empleando la siguiente ecuación:

$$SD(m, n) = \sqrt{H^2(m, n) + V^2(m, n) + \sum D_n^2(m, n)} \quad (2.1.20)$$

Tal que

$$H(m, n) = \omega\{\psi(m-1, n) - \psi(m, n)\} - \omega\{\psi(m, n) - \psi(m+1, n)\}$$

$$V(m, n) = \omega\{\psi(m, n-1) - \psi(m, n)\} - \omega\{\psi(m, n) - \psi(m, n+1)\}$$

$$D_1(m, n) = \omega\{\psi(m-1, n-1) - \psi(m, n)\} - \omega\{\psi(m, n) - \psi(m+1, n+1)\}$$

$$D_2(m, n) = \omega\{\psi(m+1, n-1) - \psi(m, n)\} - \omega\{\psi(m, n) - \psi(m-1, n+1)\}$$

$H(m, n)$ y $V(m, n)$ son las segundas diferencias horizontales y verticales respectivamente, y D_n es la segunda diferencia de la n -ésima diagonal.

La segunda diferencia $SD(m, n)$ mide la mala calidad en un pixel, por lo tanto el recíproco del valor de la diferencia es la buena calidad del pixel.

2.1.8. Algoritmos de desenvoltura de fase global

A diferencia de los algoritmos locales, los algoritmos de desenvoltura de fase global buscan optimizar ó minimizar el recorrido de la trayectoria de la integración de la fase desenvuelta; por tal motivo muchos investigadores se refieren a ellos como algoritmos de norma mínima. El objetivo de estos algoritmos es obtener la fase desenvuelta minimizando una función de error global, dicha función se puede representar como:

$$\epsilon^p = \|\text{solución} - \text{problema}\|^p \quad (2.1.21)$$

Para los casos en donde no hayan submuestreos de datos, el gradiente de la fase desenvuelta es igual al gradiente envuelto de la fase envuelta:

$$\Delta\phi(x, y) = \omega\{\Delta\psi(x, y)\} \quad (2.1.22)$$

Los algoritmos de desenvoltura global de la fase, buscan la fase desenvuelta cuyos gradientes locales en la dirección x & y coincidan lo más cercano a los gradientes envueltos de la fase envuelta, como se representa en la siguiente ecuación:

$$\epsilon^\rho = \sum_{i=0}^{M-2} \sum_{j=0}^{N-1} |\Delta^x \phi(i, j) - \widehat{\Delta}^x \psi(i, j)|^\rho + \sum_{i=0}^{M-1} \sum_{j=0}^{N-2} |\Delta^y \phi(i, j) - \widehat{\Delta}^y \psi(i, j)|^\rho \quad (2.1.23)$$

donde

$\Delta^x \phi(i, j)$ y $\Delta^y \phi(i, j)$ son los gradientes de la fase desenvuelta en las direcciones x & y respectivamente, los cuales se definen como:

$$\Delta^x \phi(i, j) = \phi(i + 1, j) - \phi(i, j) \quad (2.1.24)$$

$$\Delta^y \phi(i, j) = \phi(i, j + 1) - \phi(i, j) \quad (2.1.25)$$

$\widehat{\Delta}^x \psi(i, j)$ y $\widehat{\Delta}^y \psi(i, j)$ son los valores envueltos de los gradientes de la fase envuelta en las direcciones x & y , respectivamente, y se definen mediante las siguientes ecuaciones:

$$\widehat{\Delta}^x \psi(i, j) = \omega\{\psi(i + 1, j) - \psi(i, j)\} \quad (2.1.26)$$

$$\widehat{\Delta}^y \psi(i, j) = \omega\{\psi(i, j + 1) - \psi(i, j)\} \quad (2.1.27)$$

Tal que $\omega\{\}$ es el operador de envoltura definido como:

$$\omega\{\bullet\} = \arctan \frac{\sin(\bullet)}{\cos(\bullet)} \quad (2.1.28)$$

Este tipo de algoritmos se pueden clasificar en tres tipos: mínimos cuadrados no ponderados, mínimos cuadrados ponderados y norma L^p .

2.1.9. Mínimos cuadrados no ponderados

Este método busca minimizar la diferencia entre los gradientes de la fase desenvuelta y los valores envueltos del gradiente de la fase envuelta, minimizando en una forma de mínimos cuadrados la ecuación 2.1.23, con

$p = 2$ como se denota en la ecuación 2.1.29.[16],[17]

$$\epsilon^2 = \sum_{i=0}^{M-2} \sum_{j=0}^{N-1} |\Delta^x \phi(i, j) - \widehat{\Delta}^x \psi(i, j)|^2 + \sum_{i=0}^{M-1} \sum_{j=0}^{N-2} |\Delta^y \phi(i, j) - \widehat{\Delta}^y \psi(i, j)|^2 \quad (2.1.29)$$

Como se describe en [16], la ecuación 2.1.29 se puede reescribir como:

$$[\phi(i+1, j) - 2\phi(i, j) + \phi(i-1, j)] + [\phi(i, j+1) - 2\phi(i, j) + \phi(i, j-1)] = \rho(i, j) \quad (2.1.30)$$

Tal que

$$\rho(i, j) = [\widehat{\Delta}^x \psi(i, j) - \widehat{\Delta}^x \psi(i-1, j)] + [\widehat{\Delta}^y \psi(i, j) - \widehat{\Delta}^y \psi(i, j-1)] \quad (2.1.31)$$

Resultando que la ecuación 2.1.31 representa a la ecuación de Poisson en una versión discreta, matricialmente se denota como:

$$\mathbf{P}\vec{\phi} = \vec{\rho} \quad (2.1.32)$$

Tal que \mathbf{P} representa el Laplaciano discreto, y $\vec{\phi}, \vec{\rho}$ los vectores unidimensionales que contienen los valores de los arreglos $\phi(x, y)$ y $\rho(x, y)$.

El objetivo de éste método es resolver la ecuación de Poisson para obtener la fase desenvuelta, en [16] usan el método de Gauss Seidel por relajación para obtener su solución. La ecuación 2.1.33 es la solución de la desenvoltura de la fase, la cual se actualiza en forma iterativa hasta que el error de la diferencia llegue a cierto valor de tolerancia.

$$\phi(i, j) = \frac{[\phi(i+1, j) + \phi(i-1, j) + \phi(i, j+1) + \phi(i, j-1)] - \rho(i, j)}{4} \quad (2.1.33)$$

2.1.10. Mínimos cuadrados ponderados

A diferencia entre el método de mínimos cuadrados no ponderados, el cual asigna pesos del mismo tamaño a todos los píxeles del mapa de fases envueltos, éste método intenta obtener la solución de la fase desenvuelta mediante la minimización de la ecuación 2.1.34 asignando pesos bajos a aquellos píxeles con ruido o residuos para reducir la propagación del error en el mapa de fases desenvueltas.

$$\epsilon^2 = \sum_{i=0}^{M-2} \sum_{j=0}^{N-1} U(i, j) x |\Delta^x \phi(i, j) - \widehat{\Delta}^x \psi(i, j)|^2 + \sum_{i=0}^{M-1} \sum_{j=0}^{N-2} V(i, j) y |\Delta^y \phi(i, j) - \widehat{\Delta}^y \psi(i, j)|^2 \quad (2.1.34)$$

Tal que $U(i, j)$ y $V(i, j)$ son los pesos asignados a los gradientes en x y y respectivamente, los cuales son definidos como:

$$U(i, j) = \min(q_{i+1, j}^2, q_{i, j}^2) \quad (2.1.35)$$

$$V(i, j) = \min(q_{i, j+1}^2, q_{i, j}^2) \quad (2.1.36)$$

Donde $q_{i, j}$ es la calidad calculada del píxel (i, j) usando algún mapa de calidad de los mencionados en la parte inicial de este capítulo.

La ecuación 2.1.34 según [16] se puede representar como:

$$\begin{aligned} & U(i, j) \Delta^x \phi(i, j) - U(i-1, j) \Delta^x \phi(i-1, j) + \\ & V(i, j) \Delta^y \phi(i, j) - V(i, j-1) \Delta^y \phi(i, j-1) = \sigma(i, j) \end{aligned} \quad (2.1.37)$$

Tal que:

$$\begin{aligned} \sigma(i, j) = & [U(i, j)\widehat{\Delta}^x\psi(i, j) - U(i - 1, j)\widehat{\Delta}^x\psi(i - 1, j)] + \\ & [V(i, j)\widehat{\Delta}^y\psi(i, j) - V(i, j - 1)\widehat{\Delta}^y\psi(i, j - 1)] \end{aligned} \quad (2.1.38)$$

Expresando a la ecuación 2.1.39 en su forma matricial:

$$\mathbf{Z}\vec{\phi} = \vec{\sigma} \quad (2.1.39)$$

Tal que \mathbf{Z} representa el Laplaciano de ponderación discreta y $\vec{\sigma}$ son los vectores unidimensionales obtenidos de la ecuación 2.1.39.

La solución de la fase desenvuelta puede ser calculada iterativamente mediante la siguiente ecuación:

$$\phi(i, j) = \frac{[U(i, j)\phi(i + 1, j) + U(i - 1, j)\phi(i - 1, j) + V(i, j)\phi(i, j + 1) + V(i, j - 1)\phi(i, j - 1)] - \sigma(i, j)}{\nu(i, j)} \quad (2.1.40)$$

Donde $\nu(i, j)$ es:

$$\nu(i, j) = U(i, j) + U(i - 1, j) + V(i, j) + V(i, j - 1) \quad (2.1.41)$$

2.1.11. Norma L^p

Éste método consiste en obtener la solución de la fase desenvuelta, que minimiza a la ecuación 2.1.23 para algún valor cualquiera p , como se menciona en [16], [17].

Como un caso general, la ecuación 2.1.23 se puede simplificar de la siguiente manera:

$$\begin{aligned} & \tilde{U}(i, j)\Delta^x\phi(i, j) - \tilde{U}(i - 1, j)\Delta^x\phi(i - 1, j) + \\ & \tilde{V}(i, j)\Delta^y\phi(i, j) - \tilde{V}(i, j - 1)\Delta^y\phi(i, j - 1) = \tilde{\sigma}(i, j) \end{aligned} \quad (2.1.42)$$

Tal que \tilde{U} y \tilde{V} son los pesos que se les asigna a los gradientes en x e y respectivamente, y se definen como:

$$\tilde{U}(i, j) = |\Delta^x\phi(i, j) - \hat{\Delta}^x\psi(i, j)|^{p-2} \quad (2.1.43)$$

$$\tilde{V}(i, j) = |\Delta^y\phi(i, j) - \hat{\Delta}^y\psi(i, j)|^{p-2} \quad (2.1.44)$$

$$\begin{aligned} \tilde{\sigma}(i, j) = & [\tilde{U}(i, j)\hat{\Delta}^x\psi(i, j) - \tilde{U}(i - 1, j)\hat{\Delta}^x\psi(i - 1, j)] + \\ & [\tilde{V}(i, j)\hat{\Delta}^y\psi(i, j) - \tilde{V}(i, j - 1)\hat{\Delta}^y\psi(i, j - 1)] \end{aligned} \quad (2.1.45)$$

La norma L^p es similar al método de mínimos cuadrados ponderados, la diferencia en estos dos es que los pesos del método de mínimos cuadrados se definen independientemente de un mapa de calidad mientras que en la norma L^p los pesos son extraídos de los mismos datos de la fase envuelta. Y para llevar a cabo su resolución se pueden usar los mismos algoritmos que se utilizan en el método de mínimos cuadrados ponderados.

2.2. Sistemas de Lógica Difusa Tipo-1

En 1965 el ingeniero y matemático Lotfi Zadeh formuló la teoría de los conjuntos difusos, o más conocido como lógica difusa [20]. Su teoría fue modelada en base a la percepción humana; por ejemplo si en un salón de 40 alumnos, el profesor les pide que corten un metro de madera para un experimento, entonces habrán 40 medidas totalmente diferentes, quizás difieran por unos pocos milímetros pero cada medida de corte será realizado respecto a la percepción de cada uno de ellos, en este caso se dice que la medida es difusa.

A diferencia de la lógica booleana que devuelve valores de $\{0, 1\}$, la lógica difusa devuelve un rango de valores entre $[0, 1]$.

A continuación se muestra la representación de los componentes de un Sistema Difuso, el cual se interpreta como una función $y = f(x)$, tal que y es la función de salida con valores reales, otorgada por el cuerpo del sistema difuso, x son los valores reales de entrada, y f es una función que realiza el proceso difuso.

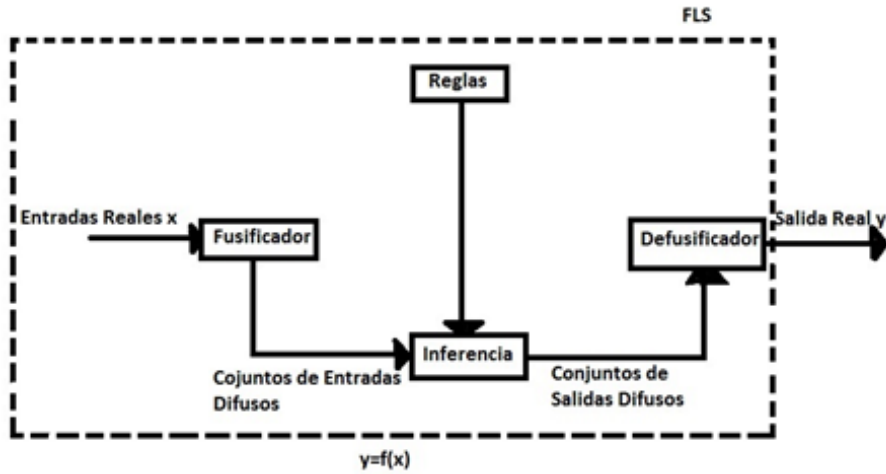


Figura 2.5: Sistema Difuso

Las *Reglas* difusas están conformadas por un conjunto de sentencias lingüísticas diseñadas y definidas en base a la experiencia y conocimiento de un experto.

Como se menciona en [21], [22], una regla difusa tiene la forma:

“SI un conjunto de condiciones se satisfacen, ENTONCES un conjunto de consecuentes se infieren” .

Supongamos que se estructuran M reglas difusas, éstas se pueden representar de manera general como:

R_j : IF x_1 is A_1^j and x_2 is A_2^j and...and x_n is A_n^j , THEN z is B^j ,

Donde $j = \{1, 2, 3, \dots, M\}$ es la regla j -ésima, x_i ($i=1,2,\dots,n$) y z son las variables de entrada y de salida, respectivamente, del sistema difuso; A_i^j y B_i^j representan los valores de la percepción otorgados por las funciones de membresía difusas $\mu_{A_i^j}$ y $\mu_{B_i^j}$ respectivamente.

La función que tiene el *Fusificador* es mapear la entrada de un conjunto de números reales $x = \{x_1, \dots, x_p\} \in X_1 \times X_2 \times \dots \times X_p = X$ a conjuntos difusos $A_x \in X$.

El rol del *Defusificador* es entregar valores $y(x) \in \Re$ como salida del Sistema de lógica difusa.

2.2.1. Lógica difusa como auxiliar para la desentvolutura de la fase

Se eligió el uso de la lógica difusa tipo-1 como una herramienta para el proceso de la desentvolutura de la fase. Una de las ventajas de esta herramienta es que los valores de salida, generados por el modelo, caen dentro del intervalo $[0, 1]$. Entonces, se sabe que para el proceso de la desentvolutura hay cierta imprecisión para conocer cuando realmente existió un cambio brusco de fase, o una asíntota de un pixel a otro.

Pues bien, se propuso un modelo difuso con la finalidad de que midiera el grado de certeza de las posibles asíntotas, y así evadir las falsas alarmas. Las reglas fueron propuestas en base a la experiencia de un experto, y las variables de entrada se eligieron de acuerdo a la información que se requiere conocer.

En esta tesis se eligió trabajar con el sistema difuso de Takagi-Sugeno-Kang (TSK) de tipo-1 de orden cero por conveniencia y ajuste a los tipos de datos a analizar, este tipo de sistema es uno de los dos más populares para las aplicaciones en ingeniería, el otro es el sistema de Mamdani.

2.2.2. Sistema de Inferencia Takagi-Sugeno-Kang

El sistema TSK de tipo-1 está conformado por un conjunto de reglas IF-THEN, en donde los antecedentes son conjuntos difusos mientras que los consecuentes son funciones lineales o constantes.[22]

Supongamos que tenemos un sistema TSK tipo-1, con m entradas $x_1 \in X_1, x_2 \in X_2, \dots, x_m \in X_m$ y una salida $z \in Z$. La i -ésima regla del modelo TSK tipo-1 de primer orden conformado por un conjunto de N reglas se puede representar como:

Rⁱ : IF x_1 is F_1^i and ... x_m is F_m^i

THEN $z^i(x) = c_0^i + c_1^i x_1 + c_2^i x_2 + \dots + c_m^i x_m$

Tal que ($i = 1, 2, \dots, N$) es el número de reglas, c_j^i ($j = 0, 1, 2, \dots, m$) son los parámetros del consecuente, F_k^i con ($k = 0, 1, \dots, m$) son los conjuntos difusos de tipo-1 del antecedente, y $z^i(x)$ es la salida de la i -ésima regla.

Por cada regla se nivela la salida $z^i(x)$ por la fuerza de disparo w^i de la regla.

La salida final del sistema de TSK es el promedio ponderado de todas salidas obtenidas por cada regla y es

calculado por la siguiente ecuación:

$$Z = \frac{\sum_{i=1}^m w^i z^i}{\sum_{i=1}^m w^i} \tag{2.2.46}$$

Una regla de Sugeno funciona como se muestran en las figuras 2.6 y 2.7. [23]

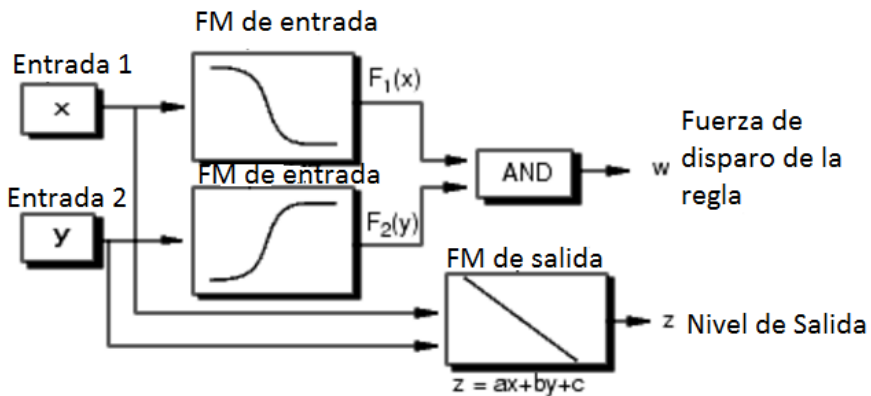


Figura 2.6: Inferencia difusa de tipo Sugeno

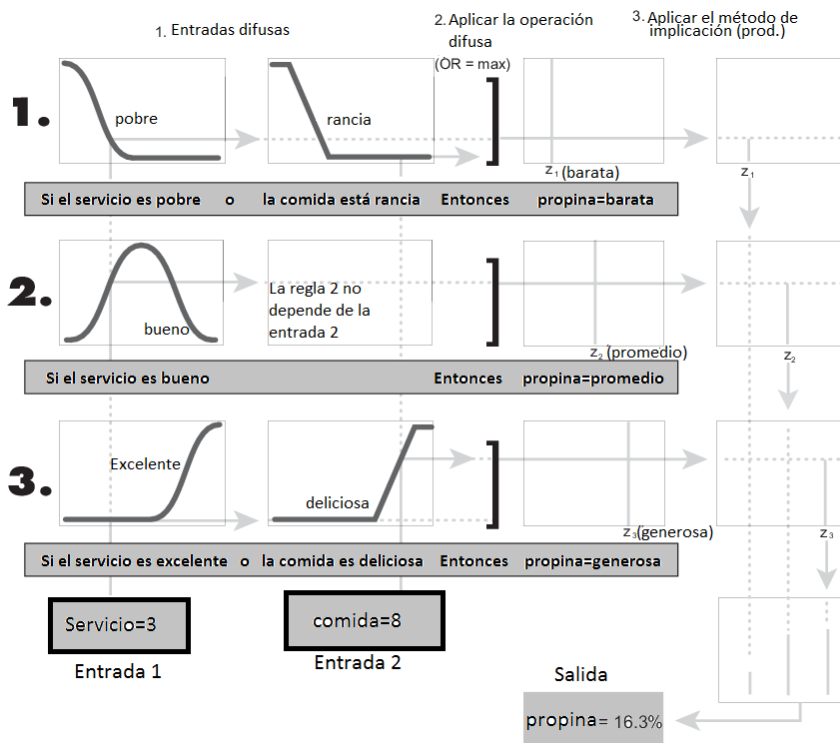


Figura 2.7: proceso de Inferencia

La conclusión en este ejemplo es que entre mejor sea el servicio y mejor sea la comida la propina será más

generosa, como se puede observar en la superficie, generado por el sistema difuso, de la figura 2.8

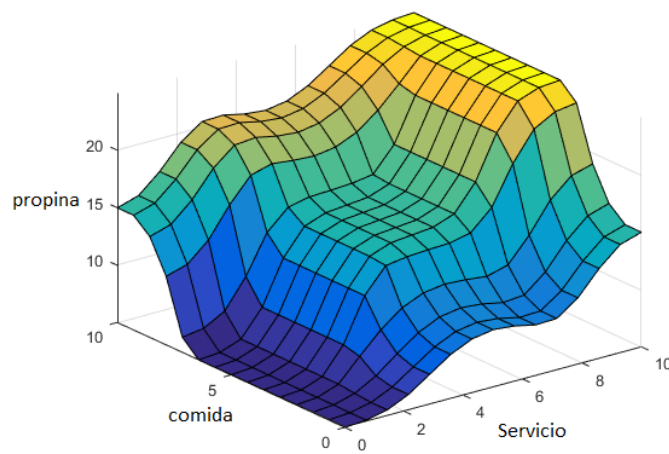


Figura 2.8: superficie generada por el modelo difuso

Capítulo 3

Desarrollo

3.1. Modelo difuso para el proceso de la Desenvoltura de la fase Envuelta

Como se mencionó en el capítulo 2, la desenvoltura de fase se define con la siguiente fórmula:

$$\phi_{i,j} = \psi_{i,j} + 2\pi\nu_{i,j}$$

Donde $\phi_{i,j}$ son los valores de la fase desenvuelta y $\psi_{i,j}$ los de las fases envueltas, en el rango de $[-\pi, \pi]$ en cada pixel i, j .

La finalidad de esta tesis es construir un modelo difuso que complemente y unifique el proceso de la desenvoltura. Existen muchísimos algoritmos enfocados a este problema, sin embargo tienen ciertas especificaciones y delimitantes para cada imagen. [24],[25],[26],[27].

El modelo se empezó a construir de acuerdo a las necesidades del problema. El problema principal es poder identificar la existencia de un cambio brusco de fase entre un punto a otro, para ello se requiere conocer la forma geométrica de la superficie que se genera a partir de la matriz de fases envueltas e identificar esos cambios abruptos.

La derivada es una herramienta que nos proporciona la información sobre los cambios que ocurren en una superficie, es por ello que se utilizó como elemento base para la entrada al sistema difuso.

Lo que se desea obtener del sistema difuso es una clasificación de los tipos de cambios que hay en la superficie de interés, por ejemplo que indique si hubo un pequeño cambio de un punto a otro y qué tan pequeño fue, o si hubo un cambio extremo.

Enseguida se presentan las metodologías de los algoritmos desarrollados para la desenvoltura de la fase utilizando Lógica Difusa.

3.2. Metodología A

Se diseñó un sistema difuso con una variable de entrada y una de salida, usando tres reglas difusas. Éstos parámetros se describirán a continuación .

3.2.1. Variable Lingüística

La variable lingüística de entrada para este modelo es la matriz de gradientes en x de la matriz de fases envueltas $\frac{\partial \psi_{x,y}}{\partial x}$, la cual contiene información de las concavidades de la superficie de interés.

Las funciones de membresía utilizadas en este modelo son funciones trapezoidales las cuales dependen de cuatro parámetros: su límite inferior a y superior d , y los límites de soporte inferior y superior b y c respectivamente, tal que $a < b < c < d$ y se define por la siguiente fórmula matemática para un vector x :

$$\mu_A(x; a, b, c, d) = \begin{cases} 0 & \text{si } (x < a) \text{ ó } (x > d) \\ \frac{x-a}{b-a} & \text{si } a \leq x \leq b \\ 1 & \text{si } b \leq x \leq c \\ \frac{d-x}{d-c} & \text{si } c \leq x \leq d \end{cases}$$

3.2.2. Funciones de Membresía

La variable lingüística de entrada al modelo es la matriz de Diferencia de Fases por renglones (D_x), y el conjunto de funciones de membresías empleado es $T(D_x) = \{Positivo, Negativo, Neutro\}$, tal que:

$$D_x(i, j) = \psi(i, j + 1) - \psi(i, j) \quad (3.2.1)$$

$$\mu_{Positivo}(x; 3, 6, 22, 38) = \begin{cases} 0 & \text{si } (x < 3) \text{ ó } (x > 38) \\ \frac{x-3}{3} & \text{si } 3 \leq x \leq 6 \\ 1 & \text{si } 6 \leq x \leq 22 \\ \frac{38-x}{16} & \text{si } 22 \leq x \leq 38 \end{cases}$$

$$\mu_{Negativo}(x; -38, -22, -6, -3) = \begin{cases} 0 & \text{si } (x < -38) \text{ ó } (x > -3) \\ \frac{x+38}{16} & \text{si } -38 \leq x \leq -22 \\ 1 & \text{si } -22 \leq x \leq -6 \\ \frac{-3-x}{3} & \text{si } -6 \leq x \leq -3 \end{cases}$$

$$\mu_{Neutro}(x; -3,5, -2, 2, 3,5) = \begin{cases} 0 & \text{si } (x < -3,5) \text{ ó } (x > 3,5) \\ \frac{x+3,5}{1,5} & \text{si } -3,5 \leq x \leq -2 \\ 1 & \text{si } -2 \leq x \leq 2 \\ \frac{3,5-x}{1,5} & \text{si } 2 \leq x \leq 3,5 \end{cases}$$

La representación gráfica de éstas funciones se observan en la figura 3.1

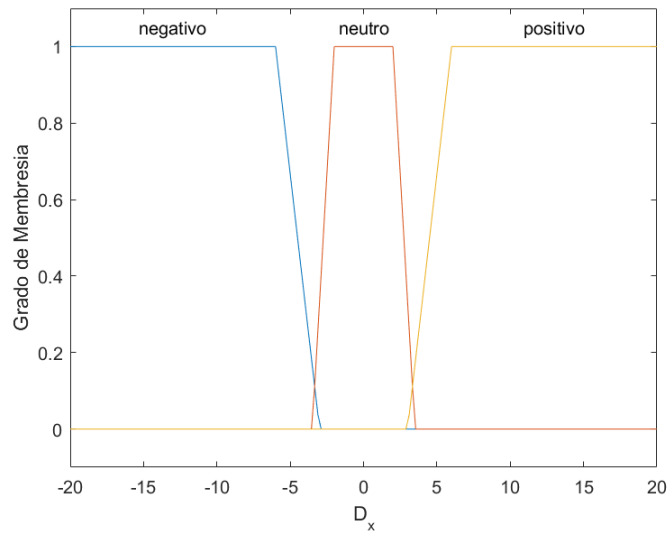


Figura 3.1: Funciones de Membresía para la variable de entrada

Para la variable lingüística de salida *Asíntota*, las funciones de membresías son:

$$T(\text{Asíntota}) = \{\text{Superior}, \text{Inferior}, \text{Nulo}\},$$

tal que la función *Superior* es la función constante 1, *Inferior* la función constante -1 , y *Nulo* es la función constante 0;

3.2.3. Reglas Difusas

1. If D_x is *Positivo* then *Asíntota* is *Superior*
2. If D_x is *Negativo* then *Asíntota* is *Inferior*
3. If D_x is *Neutro* then *Asíntota* is *Nulo*

3.3. Metodología B

Al igual que en la metodología *A*, se propone un modelo difuso de tipo Sugeno con una variable de entrada y una de salida, la diferencia en esta metodología es que se propusieron 5 reglas difusas.

3.3.1. Funciones de Membresía

La variable de entrada para corregir la fase por renglones es la variable D_x , ecuación 3.2.1, y el conjunto de funciones de membresías que lo compone es $T(D_x) = \{Positivo, Positivo2, Neutro, Negativo, Negativo2\}$, donde las funciones de membresía $Positivo2$ y $Negativo2$ indican que hubo un cambio doble ($\pm 4\pi$) en los gradientes de las fases envueltas. Las funciones de membresía para la variable de entrada se definen como:

$$\mu_{Positivo}(x; 3, 6, 9, 10) = \begin{cases} 0 & \text{si } (x < 3) \text{ ó } (x > 10) \\ \frac{x-3}{3} & \text{si } 3 \leq x \leq 6 \\ 1 & \text{si } 6 \leq x \leq 9 \\ 10-x & \text{si } 9 \leq x \leq 10 \end{cases}$$

$$\mu_{Positivo2}(x; 8,5, 12, 31,2, 40,8) = \begin{cases} 0 & \text{si } (x < 8,5) \text{ ó } (x > 40,8) \\ \frac{x-8,5}{3,5} & \text{si } 8,5 \leq x \leq 12 \\ 1 & \text{si } 12 \leq x \leq 31,2 \\ \frac{40,8-x}{9,6} & \text{si } 31,2 \leq x \leq 40,8 \end{cases}$$

$$\mu_{Neutro}(x; -3,5, -2,87, 2,87, 3,5) = \begin{cases} 0 & \text{si } (x < -3,5) \text{ ó } (x > 3,5) \\ \frac{x+3,5}{0,65} & \text{si } -3,5 \leq x \leq -2,87 \\ 1 & \text{si } -2,87 \leq x \leq 2,87 \\ \frac{3,5-x}{0,63} & \text{si } 2,87 \leq x \leq 3,5 \end{cases}$$

$$\mu_{Negativo}(x; -9,95, -8, -6, -3,5) = \begin{cases} 0 & \text{si } (x < -9,95) \text{ ó } (x > -3,5) \\ \frac{x+9,95}{1,95} & \text{si } -9,95 \leq x \leq -8 \\ 1 & \text{si } -8 \leq x \leq -6 \\ \frac{-3,5-x}{2,5} & \text{si } -6 \leq x \leq -3,5 \end{cases}$$

$$\mu_{Negativo2}(x; -31,2, -29,93, -12, -9) = \begin{cases} 0 & \text{si } (x < -31,2) \text{ ó } (x > -9) \\ \frac{x+31,2}{1,27} & \text{si } -31,2 \leq x \leq -29,93 \\ 1 & \text{si } -29,93 \leq x \leq -12 \\ \frac{-9-x}{3} & \text{si } -12 \leq x \leq -9 \end{cases}$$

Las cuales representan matemáticamente a las funciones trapezoidales, en la figura 3.2 se observa la representación gráfica de ellas:

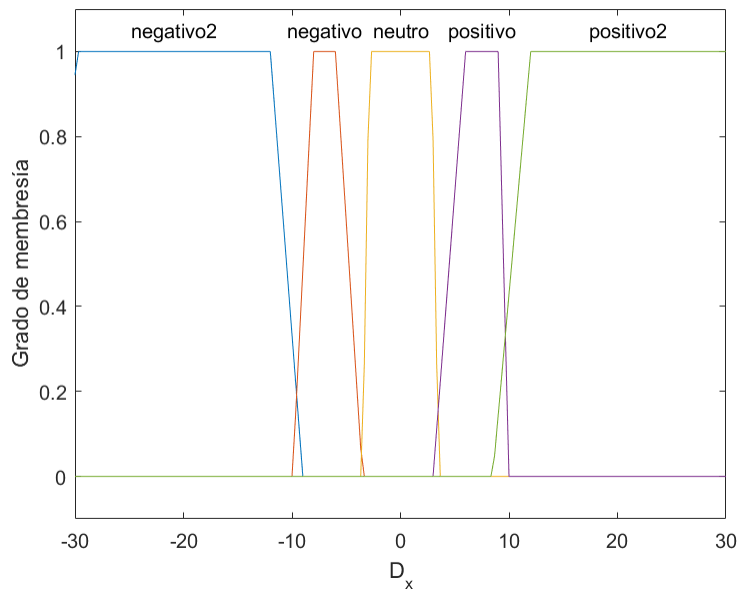


Figura 3.2: Sistema Difuso para la detección de asíntotas

La variable lingüística de salida es *Asíntota*, cuyo conjunto de funciones de membresía es: $T(Asíntota) = \{Superior, Superior2, Nulo, Inferior, Inferior2\}$

las cuales se definen como funciones constantes:

$$Superior = 1$$

$$Superior2 = 2$$

$$Nulo = 0$$

$$Inferior = -1$$

$$Inferior2 = -2$$

3.3.2. Reglas Difusas

1. If D_x is *Positivo* then *Asíntota* is *Superior*
2. If D_x is *Positivo2* then *Asíntota* is *Superior2*
3. If D_x is *Negativo* then *Asíntota* is *Inferior*
4. If D_x is *Negativo2* then *Asíntota* is *Inferior2*
5. If D_x is *Neutro* then *Asíntota* is *Nulo*

Al realizar el proceso de inferencia sobre la matriz de fases envueltas, la fase envuelta $\psi(i, j)$ será corregida por renglones. A esta modificación la nombraremos $\psi_x(i, j)$ para indicar que la fase ha sido analizada y corregida a lo largo del eje x .

Al obtener $\psi_x(i, j)$ se analizarán los gradientes en y el cual denotaremos como D_y y ésta variable será la nueva entrada al modelo difuso donde se realizará el mismo proceso de inferencia.

3.3.3. proceso de recuperación de la fase

Imaginemos que tenemos una matriz de fases envueltas con un tamaño de 3×10 , se considerará como fase verdadera a cada elemento del primer renglón, es decir la primera columna, y al calcular la derivada en x e ingresarla al modelo difuso obtenemos de salida la siguiente matriz de cambios a lo largo de x (MC_x) :

$$MC_x = \begin{bmatrix} 0 & 0,46 & -0,8 & 0,2 & 0,3 & 0,34 & -0,89 & 0,49 & 0,5 & 0,74 \\ 0 & 0,24 & -0,7 & 0,3 & -1 & 0,1 & 0,29 & 0,99 & 0,55 & 0,6 \\ 0 & 0,38 & 0,82 & 0,68 & 0,41 & -0,84 & 0,11 & 0,53 & 0,19 & -0,95 \end{bmatrix}$$

Cada valor de ésta matriz nos representa un grado de confianza con el cual podemos asegurar el tamaño del cambio que hay a lo largo del eje x en cada punto. Donde el valor 1 nos asegura que realmente hay una asíntota positiva, el valor -1 que hay una asíntota negativa, y los valores como $-0,89$ y $-0,95$ nos señalan que también esos puntos pueden considerarse como asíntotas negativas.

El programador eligirá un valor de tolerancia para decidir cuando un elemento de la matriz será considerado como una asíntota. Supongamos que se elige el valor $|0,7|$, por lo tanto se agruparán los elementos de MC_x en tres rangos, el primero son todos los valores menores o iguales a $-0,7$, el segundo son todos los valores entre el intervalo $(-0,7, 0,7)$ y el tercero son todos los valores mayores o iguales a $0,7$.

Se rastreará a cada renglón de la MC_x y se colocarán tres banderas, una para indicar que existe una asíntota positiva (1), la otra para indicar que existe una asíntota negativa (-1), y la última para indicar que no existen asíntotas (0).

El resultado del análisis anterior es una matriz cuyos elementos son $\{0, 1, -1\}$, lo que procede ahora es inspeccionar a cada renglón de la matriz resultante. Por cada valor igual a 1 que aparezca en el renglón se le sustraerá 2π a partir de ese pixel, por cada valor igual a -1 se añadirá 2π a partir de ese pixel, y si aparece el valor 0 no se añadirá nada.

Después de rastrear la MC_x del ejemplo anterior se obtiene la siguiente matriz:

$$\begin{bmatrix} 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & -1 \end{bmatrix}$$

Observamos cada renglón, y realizamos las adiciones o sustracciones según lo descrito anteriormente, resultado de ello se obtiene la siguiente matriz a la que nombraremos matriz de correcciones por renglones (MDC_x):

$$MDC_x = \begin{bmatrix} 0 & 0 & 2\pi & 2\pi & 2\pi & 2\pi & 4\pi & 4\pi & 4\pi & 2\pi \\ 0 & 0 & 2\pi & 2\pi & 4\pi & 4\pi & 4\pi & 2\pi & 2\pi & 2\pi \\ 0 & 0 & -2\pi & -2\pi & -2\pi & 0 & 0 & 0 & 0 & 2\pi \end{bmatrix}$$

Lo que procede es añadirle a la matriz de fases envueltas $\psi(i, j)$ la matriz MDC_x , con lo cual se obtendrá la fase desenvuelta por renglones, y la representaremos como sigue:

$$\phi_x(i, j) = \psi(i, j) + MDC_x \quad (3.3.2)$$

Donde

$\phi_x(i, j)$ es la matriz de fases desenvueltas por renglones

$\psi(i, j)$ es la matriz de fases envueltas

MDC_x es la matriz de correcciones por renglones.

Una vez corregidas las fases por renglones de la matriz $\psi(i, j)$, se reinicia el mismo proceso para los gradientes en y .

Sea MDC_y la matriz de correcciones a lo largo del eje y , la fase corregida por renglones y columnas será:

$$\phi_{x,y}(i, j) = \phi_x(i, j) + MDC_y \quad (3.3.3)$$

Y es con la ecuación 3.3.3 con que se determina la fase absoluta, o fase desenvuelta.

Capítulo 4

Pruebas y resultados

4.1. Resultados Obtenidos con la Metodología A

Se probó el modelo difuso de la metodología A con distintas imágenes sintéticas tomadas de [28]. La forma de calcular la fase de cualquiera de las siguientes imágenes sintéticas es mediante la ecuación:

$$\phi(x, y) = \arctan \frac{Im(Imagen)}{Re(Imagen)} \quad (4.1.1)$$

Donde $Im(Imagen)$ y $Re(Imagen)$ son la parte imaginaria y real de la imagen respectivamente.

1.-Pruebas con la Imagen Picos:

La primera imagen es la función picos, obtenida a partir de la siguiente ecuación:

$$Imagen_{picos} = 20 \exp^{-0,25(x^2+y^2)} + 2x + y$$

Su imagen correspondiente se muestra en la figura 4.1

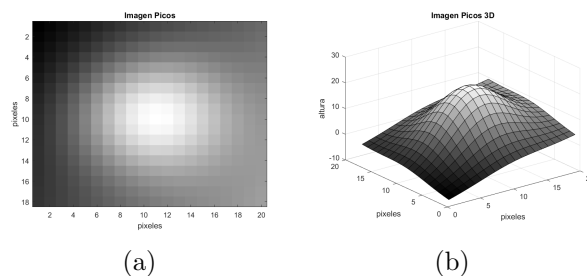


Figura 4.1: (a) Imagen Picos, (b) Imagen Picos vista en 3D

Como se había mencionado en el capítulo anterior, la variable de entrada al modelo es la matriz de diferencias de fases por renglones, en la figura 4.2 (a) se muestra la entrada vista como un arreglo de intensidades. Del lado derecho de la imagen se aprecia una barra de colores la cual indica que los tonos blancos son diferencias de fase cercanas a 2π , los tonos oscuros son diferencias de fases cercanas a -2π , y los tonos grises indican que sólo hay un ligero cambio entre un pixel y su vecino. En el inciso (b) de la misma figura se muestra la salida del modelo difuso, donde los valores que comprenden de $[-1,1]$ representan los niveles de cambios asintóticos en la matriz de fases envueltas.

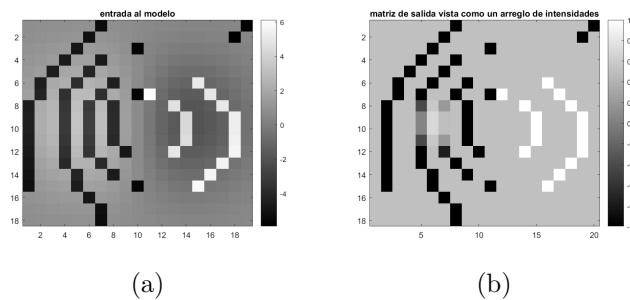


Figura 4.2: (a) entrada al modelo visto como un arreglo de intensidades, (b) salida inicial del modelo

La derivada de la matriz corregida por renglones [4.2 (b)] será la nueva entrada al mismo sistema difuso para realizar las correcciones de fase por columnas. En la figura 4.3 (a) se muestra el mapa de cambios por columnas visto como un arreglo de intensidades y en el inciso (b) se observan con tonos de intensidades la clasificación de los cambios asintóticos que el sistema detectó.

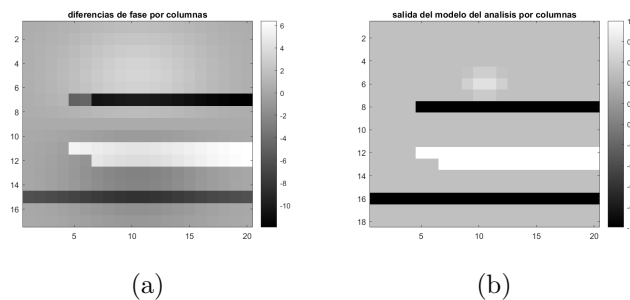


Figura 4.3: (a) Matriz de entrada vista como una imagen, (b) Matriz de salida vista como una imagen

Finalmente se corrigen las fases por columnas y se espera llegar a la solución, sin embargo como se observa en la figura 4.4 (c), la fase absoluta obtenida contiene fuentes de errores en toda la esquina inferior derecha, el detalle está en que el error se propagará a partir del elemento mal considerado.

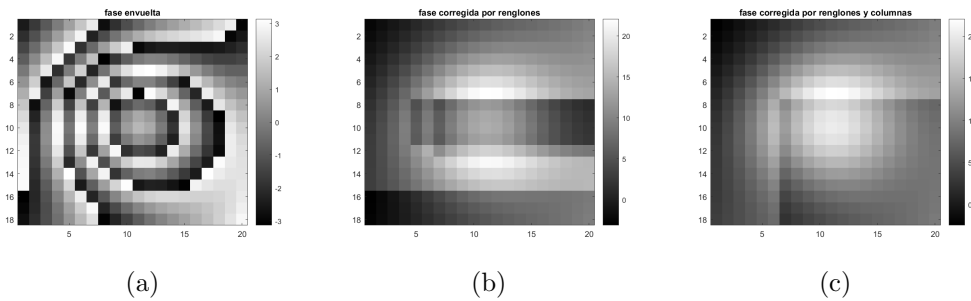


Figura 4.4: (a) fase envuelta, (b) fase desenvuelta por renglones, (c) fase desenvuelta por renglones y columnas

Si se visualizan las fases como un objeto se logrará identificar que el sistema realiza bien su trabajo pero no logra identificar un cambio asintótico negativo [figura 4.5 (b), (c) y (d)].

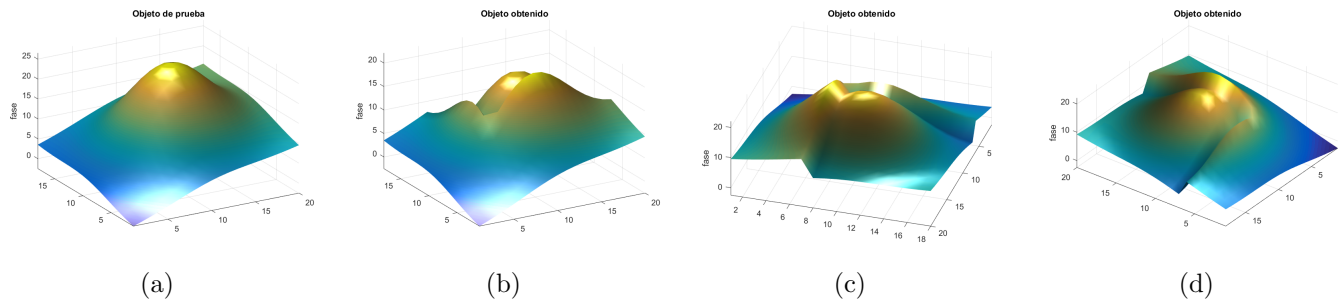


Figura 4.5: (a) fase verdadera, (b), (c) y (d) fase obtenida vista desde diferentes ángulos

2.-Pruebas con la Imagen Escalon:

La siguiente prueba del modelo se realizó con la imagen de un objeto parecido a un escalón el cual se observa en la figura 4.6

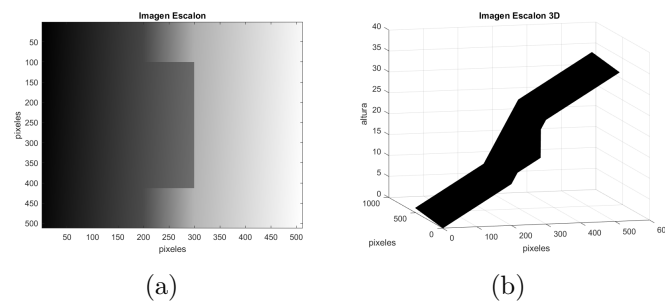


Figura 4.6: (a) Imagen Escalon, (b) Imagen Escalon vista en 3D

La entrada al modelo es la matriz de diferencias de fases por renglones visto como una imagen bidimensional, como se observa en la figura 4.7 (a), y la salida (b) del modelo del análisis por renglones.

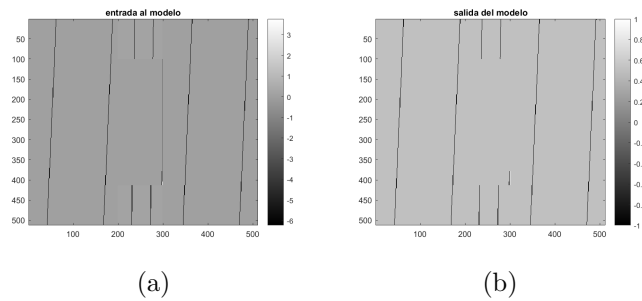


Figura 4.7: (a) entrada del modelo visto como imagen, (b) salida del modelo visto como imagen

En la figura 4.8 (b) la fase ha sido corregida por renglones, así que se calculan las diferencias de fases por columnas, el sistema difuso realiza el análisis y se corrigen las fases como se observa en la figura 4.8 (c). En

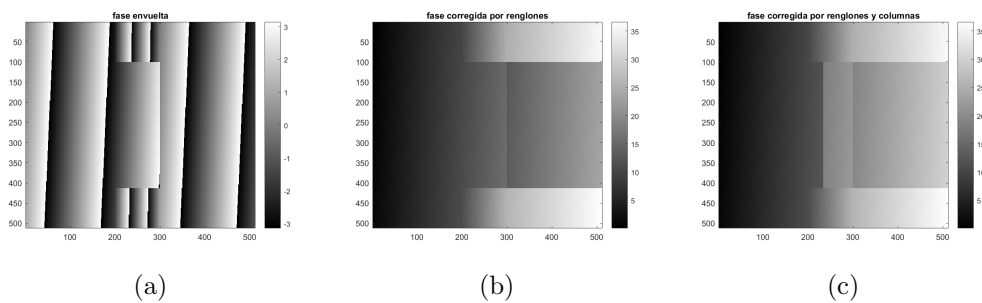


Figura 4.8: (a) fase Envuelta, (b) fase corregida por renglones, (c) fase corregida por renglones y columnas

la figura 4.9 se observan los resultados obtenidos vistos en 3D, el inciso (d) es el resultado final obtenido y puede observarse que hubo una propagación del error desde el centro de la imagen, ésto es porque consideró como fase envuelta a la fase verdadera que se encuentra en el centro de la imagen del inciso (a) de la figura 4.8

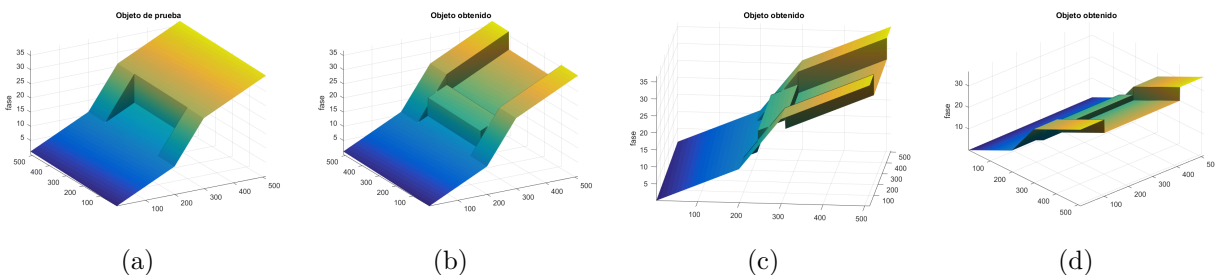


Figura 4.9: (a) objeto de prueba, (b), (c) y (d) objeto obtenido visto desde diferentes ángulos

3.-Pruebas con la Imagen Espiral:

Figura 4.10, imagen a analizar (a) y vista como un objeto (b).

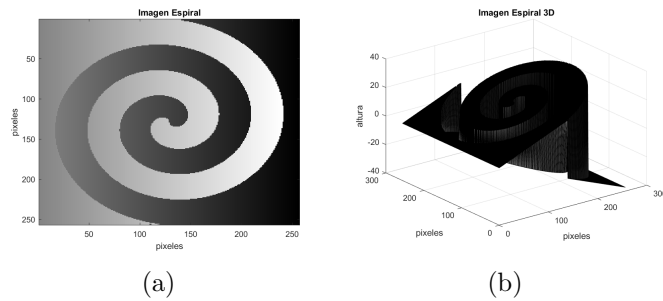


Figura 4.10: (a) Imagen Espiral,(b) Imagen Espiral vista en 3D

La espiral es una de las imágenes más complicadas por su forma natural, como se observa en la figura 4.11 (a) se muestran los cambios que existen entre un pixel y su vecino lateral pero el detalle es reconocer cuándo realmente hay un cambio asintótico entre dos pixeles y cuándo solo es el contorno de la espiral, es decir una falsa alarma.

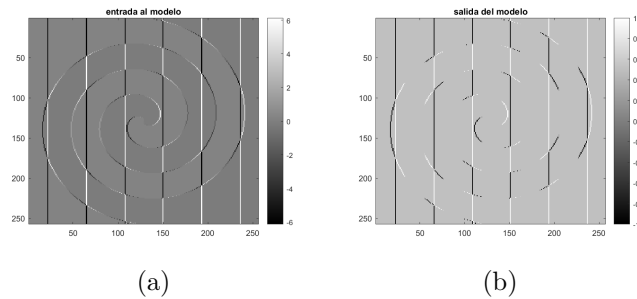


Figura 4.11: (a) entrada del modelo visto como imagen, (b) salida del modelo visto como imagen

En la figura 4.12 (b) se aprecia a partir de cuales pixeles el error se propagó hacia la derecha al momento de corregir los valores de las fases por renglones, y lo mismo ocurrió en el proceso de la desenvoltura de la fase por columnas, figura 4.12 (c).

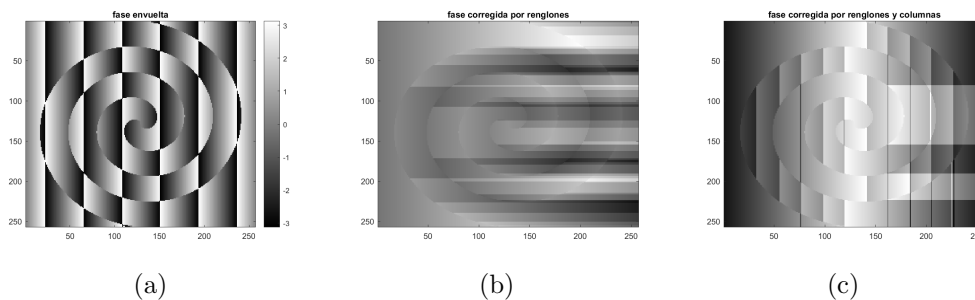


Figura 4.12: (a) fase Envuelta, (b) fase corregida por renglones, (c) fase corregida por renglones y columnas

Visualizando el proceso de la desenvoltura en 3D para la imagen de la espiral notamos que la metodología A

realmente no es la indicada para este tipo de imágenes.

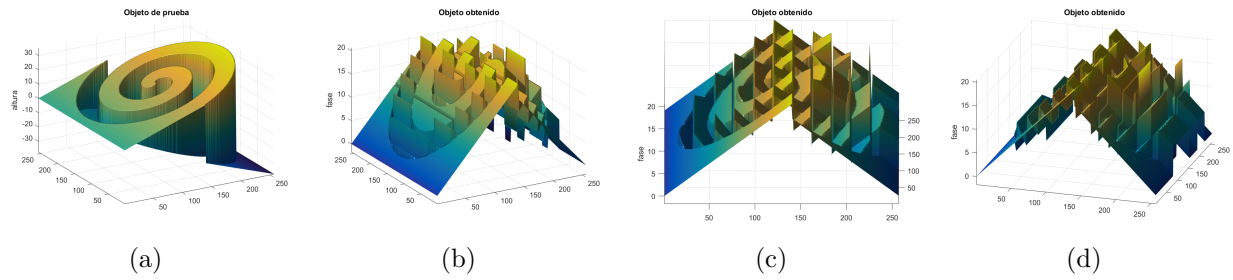


Figura 4.13: (a) objeto de prueba, (b),(c) y (d) objeto obtenido visto desde diferentes ángulos

4.2. Resultados Obtenidos con la Metodología B

Se analizaron y procesaron las mismas imágenes empleando los parámetros del sistema difuso de la metodología B, los resultados mejoraron pero el error de propagación continua siendo un problema para la imagen espiral.

1.-Imagen Picos

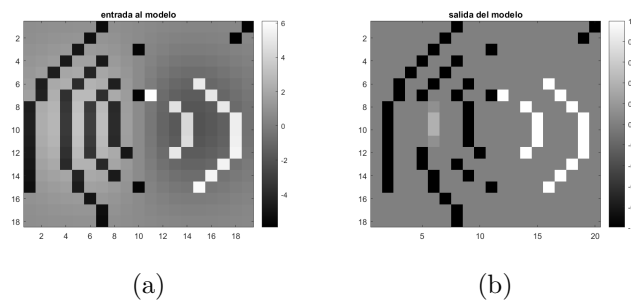


Figura 4.14: (a) entrada al modelo visto como un arreglo de intensidades, (b) salida inicial del modelo

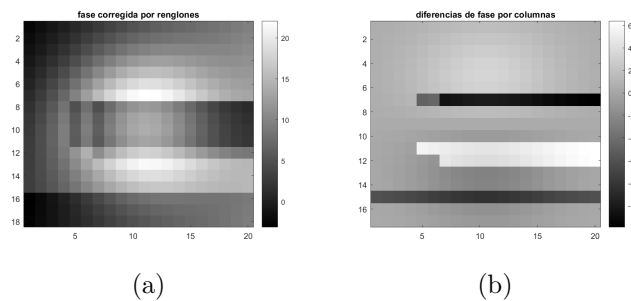


Figura 4.15: (a) Fase corregida por renglones, (b) Diferencias de fases por columnas

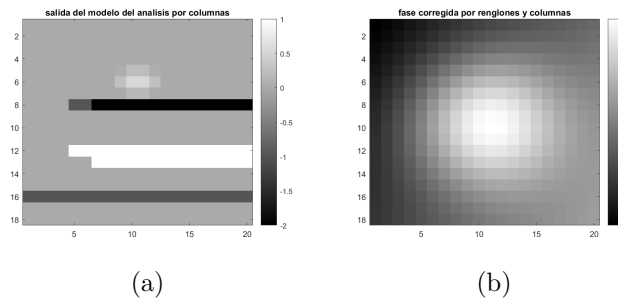


Figura 4.16: (a)Salida del modelo, (b)fase corregida por renglones y columnas

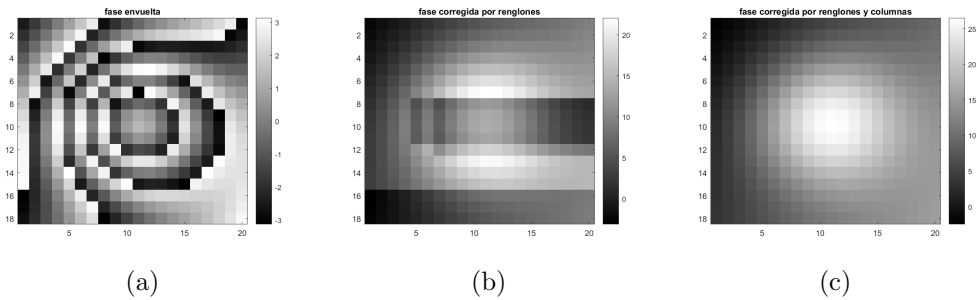


Figura 4.17: (a)fase envuelta, (b)fase corregida por renglones, (c) fase corregida por renglones y columnas

Se observa que la solución de la desenvoltura para la imagen picos ha sido obtenida, como se observa en la figura 4.18 (c), esto se debe a que existían gradientes mayores a 2π .

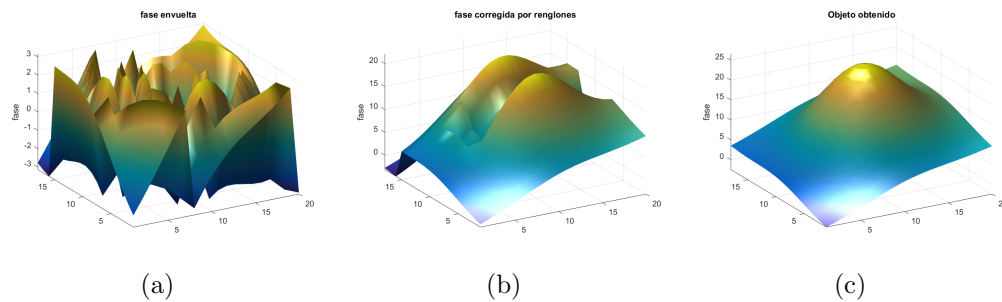


Figura 4.18: (a)fase envuelta, (b)fase corregida por renglones, (c) fase corregida por renglones y columnas

En la figura 4.19 se muestra el valor de la regresión lineal que se obtuvo al comparar la fase verdadera del objeto picos vs la fase obtenida empleando la metodología B .

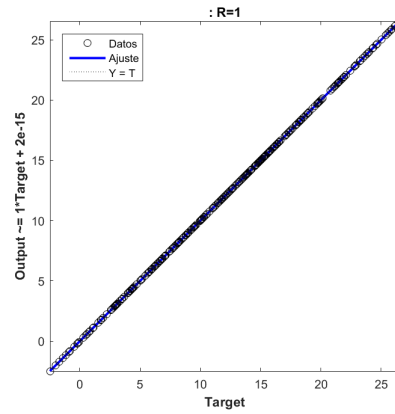


Figura 4.19: Regresión Lineal de la fase verdadera vs la fase obtenida de la imagen picos

2.-Imagen Escalón

Si observamos la imagen 4.9 (d) vemos que hubo un desnivel en la parte superior derecha, ésto se debe a que existían gradientes mayores a $|2\pi|$ y en la metodología *A* sólo considerábamos los cambios no mayores a $|2\pi|$. En la metodología *B* se anexaron estas posibilidades y por tal motivo se pudieron agregar esos cambios de fases, como se observa en la figura 4.22 (d), sin embargo no se detectaron las falsas alarmas en el centro de la imagen lo que ocasionó el desnivel en dos puntos.

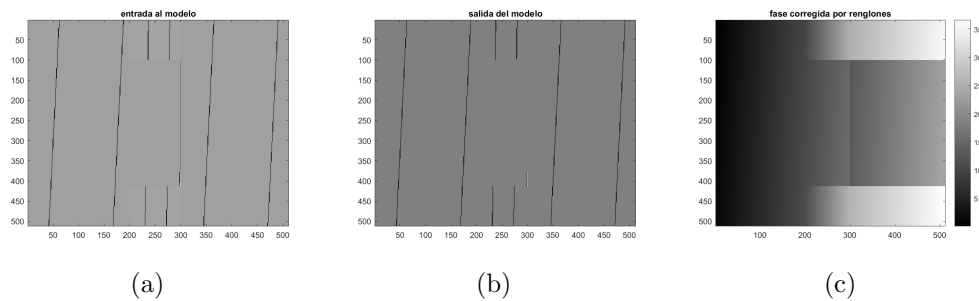


Figura 4.20: (a)Matriz de entrada al modelo difuso, (b)Matriz de salida del análisis por renglones, (c)Fase corregida por renglones

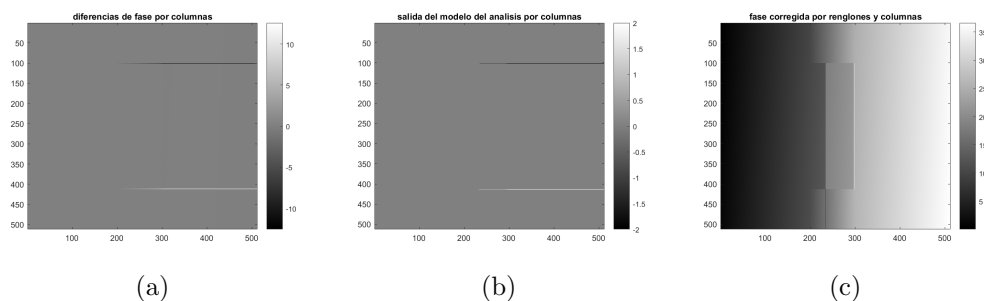


Figura 4.21: (a)Diferencia de Fases por columnas,(b)Salida del análisis por columnas,(c)Fase corregida por columnas y renglones

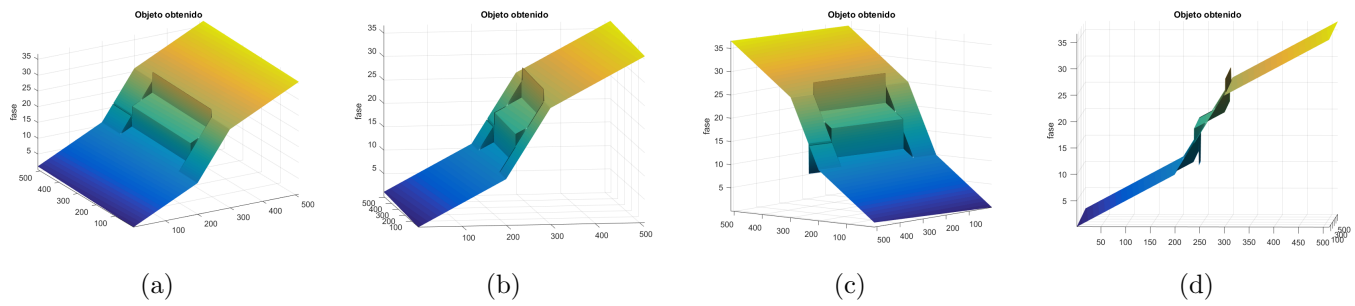


Figura 4.22: (a),(b),(c) y (d) objeto obtenido del modelo visto en 3D desde diferentes ángulos

Se obtuvo un valor de 0,98 al comparar la fase verdadera de la imagen escalon vs la fase obtenida con la metodología *B*, y se pueden apreciar los datos que no concordaron con los valores verdaderos.

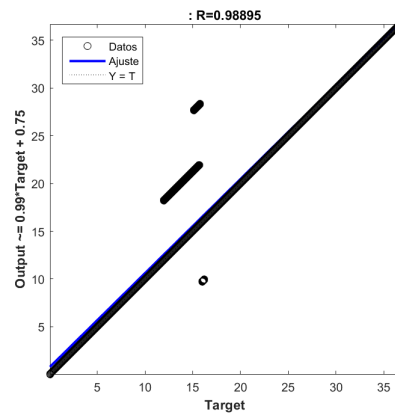


Figura 4.23: Regresión Lineal de la fase verdadera vs la fase obtenida de la imagen escalon

3.-Imagen Espiral

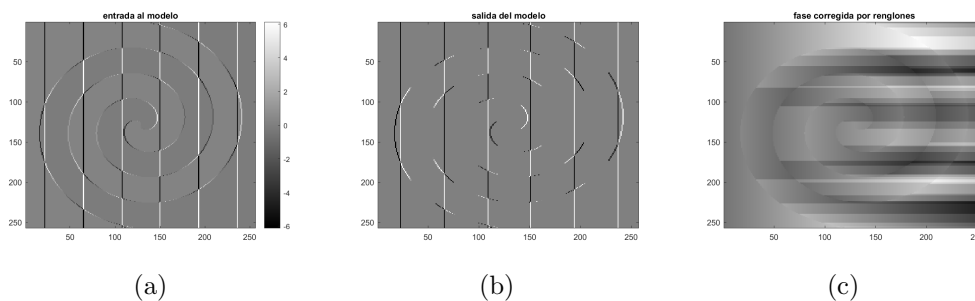


Figura 4.24: (a) entrada del modelo visto como una imagen, (b) Salida del modelo del análisis por renglones, (c) Corrección de fases por renglones

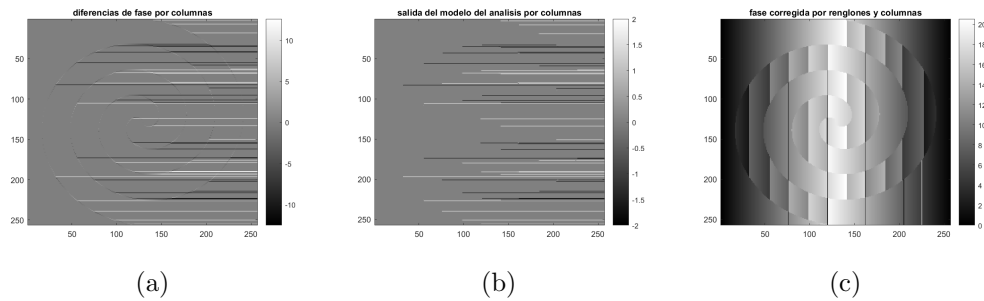


Figura 4.25: (a)Diferencias de fases por columnas,(b) Salida del modelo del análisis por columnas,(c)Corrección de fases por renglones y columnas

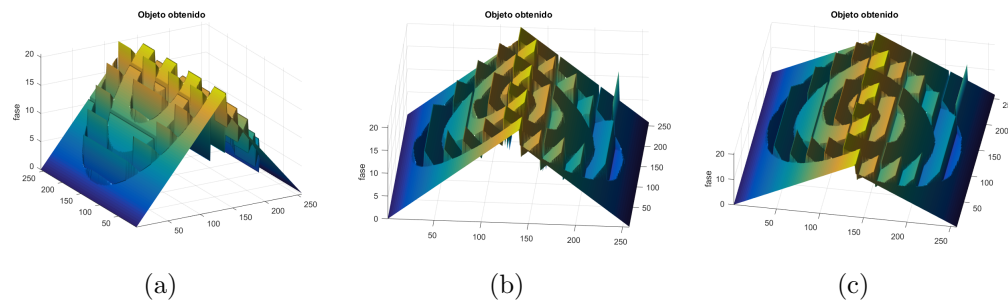


Figura 4.26: (a),(b)y (c) Objeto obtenido por el modelo difuso visto desde diferentes ángulos

Como no se contaba con la fase verdadera de la imagen espiral, se compararon las intensidades de la imagen vs los datos de la fase obtenidos en la metodología *B*. Por dicha razón hay un desajuste lineal, además de que la metodología falló totalmente en la recuperación de la fase de ésta figura.

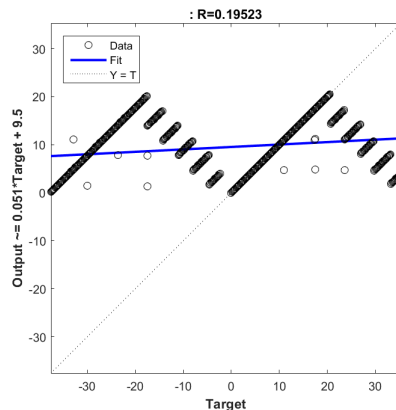


Figura 4.27: Regresión Lineal de la fase verdadera vs la fase obtenida de la imagen espiral

4.2.1. Aplicación

La idea inicial era obtener la fase en imágenes reales empleando el método de Perfilometría de Fourier, sin embargo es complicado obtener la solución, o lo más cercano de la fase desenvuelta. Se empleó la primera metodología para recuperar la fase de un objeto real, el cual fue capturado usando una cámara común, el objeto es una porción del hueso de un pez marlin de 351 x 656 pixeles el cual se observa en la imagen 4.28.



Figura 4.28: Objeto de prueba: porción del hueso de un pez marlin

Se proyectaron franjas cosenoidales a la imagen del hueso mediante un ordenador, cuya frecuencia es de $f_0 = 1/30$, y se creó un plano de referencia del mismo tamaño los cuales se observan en la figura 4.29, las imágenes se procesaron empleando el método de la transformada de Fourier el cual genera la fase envuelta. Al aplicar el método de desenvoltura de la metodología *A* se obtuvieron los siguientes resultados:

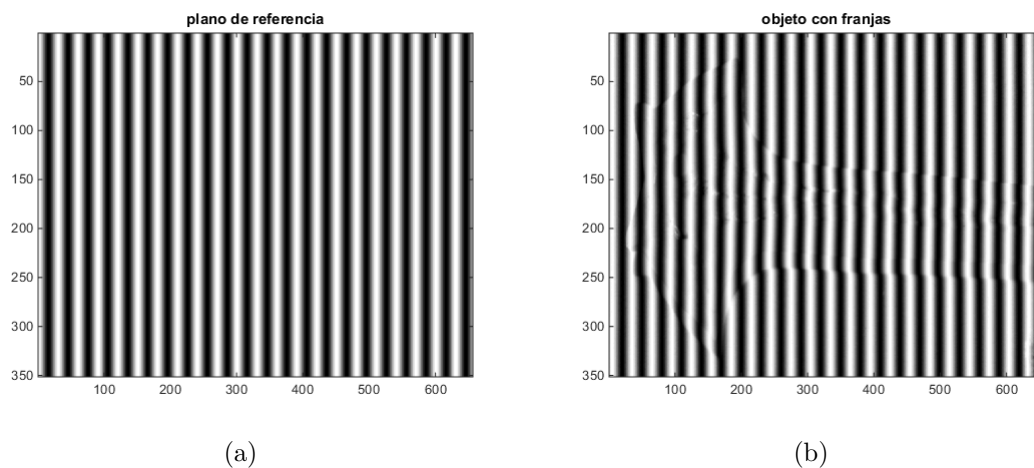


Figura 4.29: (a) y (b) Plano de referencia y Objeto con proyección de franjas cosenoidales, respectivamente

En la figura 4.30(a) y (b) se muestran las fases desenvueltas por renglones y columnas del plano de referencia, y puesto que el plano fue creado por computadora la imagen no contiene fuentes de ruido.

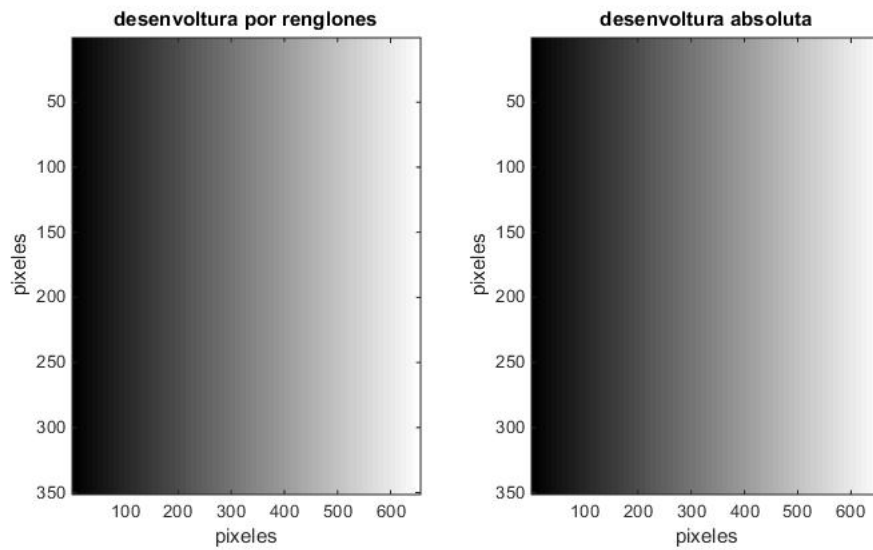


Figura 4.30: (a) Fase desenvuelta por renglones de Plano Referencia, (b) Fase desenvuelta por renglones y columnas

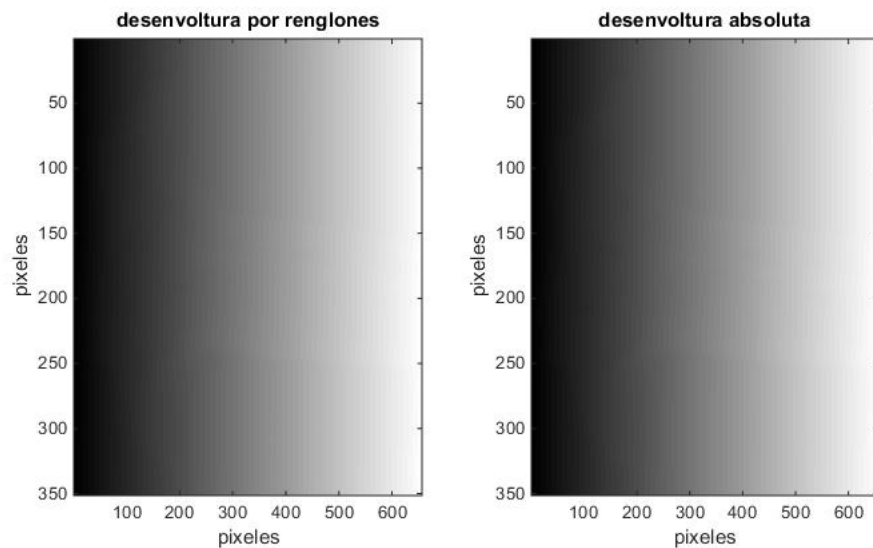


Figura 4.31: (a) Fase desenvuelta por renglones de imagen Hueso, (b) Fase desenvuelta por columnas de Imagen Hueso

La fase absoluta de la imagen del hueso de pez marlin se presenta como un arreglo de intensidades en la figura 4.32

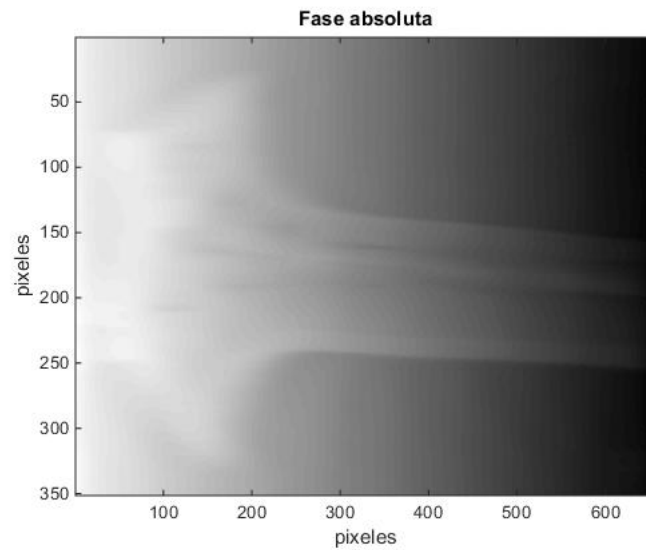


Figura 4.32: Fase de la imagen de pez marlin visto como un arreglo de intensidades

En esta última aplicación se puede observar que el modelo difuso pudo recuperar sin problemas la fase de la imagen del hueso, el detalle aquí es que tanto las franjas proyectadas en la imagen del hueso como el plano de referencia fueron sintéticas.

Capítulo 5

Conclusiones y trabajo futuro

5.0.2. Conclusiones

En comparación con los algoritmos tradicionales (Goldstein, Rutas de Calidad, Cortes por Máscaras), el modelo difuso propuesto tiene la ventaja de darle un valor o grado de pertenencia, valores entre $(0, 1)$, a cada elemento de la matriz de fases envueltas para representar el cambio que hubo de un elemento a otro, valores que a diferencia de los demás algoritmos pueden pertenecer a uno o más cambios del valor del gradiente, y esto nos ayudará a encontrar posibles falsas alarmas. Un ejemplo donde puede ocurrir alguna falsa alarma, es cuando existen diferencias de fases muy cercanos a los valores de -2π y 2π y que algún algoritmo tradicional lo tomaría como un cambio brusco cuando en realidad no lo es, en comparación con el algoritmo difuso, éste puede considerarlo de las dos formas con los grados de pertenencia adecuados, i.e. qué tanto cambio hubo entre un elemento a otro.

En resumen, un sistema difuso nos da la libertad de analizar y tomar la decisión de acuerdo a nuestra experiencia y observaciones acerca de cuales son las posibles condiciones para que un valor de fase se considere como una asíntota o muy cercano a ello, claro está que debe de ir de la mano con algún algoritmo de desevoltura.

5.0.3. Trabajo Futuro

En un inicio se buscaba trabajar con imágenes capturadas con una cámara común empleando el método de perfilometría de Fourier, el problema de trabajar con este tipo de imágenes es que en la mayoría de los casos se genera una fuente de ruido, el cual se propaga en todo el proceso de la obtención de la fase, y es realmente complicado deshacerse de él, sin embargo existen diversos algoritmos que se encargan específicamente en tratar el ruido de las imágenes reales.

Una buena opción para tratar el ruido sería con el uso de Lógica Difusa Tipo 2, ya que este tipo de Sistemas se centra en tratar principalmente con fuentes de incertidumbre, el ruido en imágenes es un ejemplo de una fuente de incertidumbre y es muy posible que los resultados mejoren.

Se espera que teniendo una imagen limpia, el proceso de la obtención de la fase para imágenes reales mejore.

Por otra parte, observando los resultados obtenidos con las metodologías presentadas en esta tesis nos damos cuenta que el algoritmo implementado es un algoritmo de desvoltura en forma continua, motivo por el cual el error se propaga. La solución aquí sería buscar o probar un algoritmo de desvoltura en forma discreta que evite la propagación del error.

Tomando en cuenta lo anterior, se podrían analizar imágenes reales sin tanto problema y reconstruir su forma tridimensional, empleando un sólo método para cualquier imagen.

Capítulo 6

Anexos

Las imágenes utilizadas en el capítulo 4 fueron extraídas del artículo [28], y se generan con el siguiente código en *Matlab*.

1.-Imagen Picos

```
1  % codigo que genera la imagen picos
2  clc; close all; clear
3  NRx = 20; NRy=18; % Tamano de la imagen
4  tx = linspace(-3,3,NRx);
5  ty = linspace(-3,3,NRy);
6  [x,y]=meshgrid(tx,ty);
7  imagen1 = 20*exp(-0.25*(x.^2 + y.^2)) + 2*x + y;
8  figure, colormap(gray(256)), imagesc(tx,ty,imagen1)
9  title('Imagen_original_vista_como_un_arreglo_de_intensidades')
10 xlabel('pixeles'), ylabel('pixeles')
```

2.-Imagen Escalón

```
1  clc;close all;clear
2  N = 512;
3  [x,y] = meshgrid(1:N);
4  shape = zeros(N,N);
```

```

5 phaseChange = 10;
6 shape(:,300:end) = phaseChange;
7 x1=meshgrid(1:100);
8 shape(1:100,200:299) = x1*phaseChange/100;
9 shape(413:512,200:299) = x1*phaseChange/100;
10 escalon = shape + 0.05*x + 0.002*y;
11 figure,imagesc(escalon);colormap(gray(256));
12 title('Imagen original vista como un arreglo de intensidades');
13 xlabel('pixeles');ylabel('pixeles')

```

3.-Imagen Espiral

```

1 clc; close all; clear
2 N = 257;
3 fid = fopen('spiral.257x257.surf','r','b');
4 espiral = fread(fid, N * N, 'float');
5 fclose(fid);
6 espiral = reshape(espiral, N, N);
7 espiral = double(espiral);
8 figure, imagesc(espiral),colormap(gray(256));
9 title('imagen original vista como un arreglo de intensidades');
10 xlabel('pixeles');ylabel('pixeles')

```

6.1. ALGORITMOS

A continuación se describe el algoritmo empleado en la metodología A:

```

1 %funcion que corrige los cambios bruscos de la matriz envuelta tanto en renglones como en
   columnas eligiendo un valor de certeza vc para decidir si existe una asintota o no.
2 function [xx,yy]=fisMA(fase_envuelta,vc)
3     [r,c]=size(fase_envuelta); %tamano de imagen
4     dx=diff(fase_envuelta)'; %diferencia de fase por renglones

```

```
5 dx=[zeros(r,1),dx];
6 % la primera columna se considera como fase verdadera, por lo tanto no hay cambios
   abruptos y su derivada es cero.
7 vx=reshape(dx',1,r*c); % vectorizar la matriz de diferencias para introducirla al
   sistema difuso
8 fis=readfis('des_metA.fis'); % leer el Modelo Difuso
9 evfisx=evalfis(vx,fis); % evaluar el modelo en el vector de diferencias
10
11 indpx=find(evfisx>vc); % buscador de asintotas dependiendo del valor de certeza
12 mx=zeros(1,length(vx));
13 mx(indpx)=1;
14 indnx=find(evfisx<=-vc);
15 mx(indnx)=-1;
16 asinx=(reshape(mx,c,r))'; % asintotas encontradas en los renglones
17
18 suml=0;
19 for i=1:r
20     for j=1:c
21         sumax(i,j)=suml-2*pi*asinx(i,j); % correccion de fases
22         suml=sumax(i,j);
23     end
24     suml=0;
25 end
26 desx=fase_envuelta+sumax;
27 xx=desx; % fase desenvuelta por renglones
28
29 dy=diff(desx);
30 dy=[zeros(1,c);dy];
31 vy=reshape(dy,1,r*c);
32 evafisy=evalfis(vy,fis);
```

```

33     indpy=find ( evafisy >vc );
34     indny=find ( evafisy <-vc );
35     my=zeros ( 1 ,length ( vy ) );
36     my(indpy)=1;
37     my(indny)=-1;
38     asiny=reshape ( my , r , c );
39
40     sum2=0;
41     for k=1:c
42         for l=1:r
43             sumay ( l , k ) = sum2 - 2 * pi * asiny ( l , k );
44             sum2 = sumay ( l , k );
45         end
46         sum2=0;
47     end
48     desy=desx+sumay;
49     yy=desy; % fase desenvuelta por renglones y columnas
50 end

```

Código para la prueba de la metodología A:

```

1  %probando la metodologia A
2  clc , close all , clear ;
3  load 'picos.mat' ; load 'escalon.mat' ; load 'espiral.mat'
4  fase_envuelta_picos=atan2 ( sin ( picos ) , cos ( picos ) );
5  fase_envuelta_escalon=atan2 ( sin ( escalon ) , cos ( escalon ) );
6  fase_envuelta_espiral=atan2 ( sin ( espiral ) , cos ( espiral ) );
7
8  %comparaci n entre la superficie real y la superficie obtenida
9  %A.-PICOS

```

```
10 [fxpicos , fypicos]=fisMA(fase_envuelta_picos ,0.85); %el segundo elemento es el valor de
    certeza que nosotros elegimos
11 figure , imagesc(fase_envuelta_picos) , colormap(gray(256)); title('fase_envuelta')
12 figure , imagesc(fxpicos) , colormap(gray(256)); title('fase_corregida_por_renglones')
13 figure , imagesc(fypicos) , colormap(gray(256)); title('fase_corregida_por_renglones_y_columnas')
14
15 figure , surf(fase_picos , 'FaceColor' , 'interp' , 'EdgeColor' , 'none' , 'FaceLighting' , 'phong')
16 view(-30,30) , camlight left , axis tight , title('Objeto_de_prueba') , zlabel('fase')
17
18 figure , surf(fypicos , 'FaceColor' , 'interp' , 'EdgeColor' , 'none' , 'FaceLighting' , 'phong')
19 view(-30,30) , camlight left , axis tight , title('Objeto_obtenido') , zlabel('fase')
20
21 %.-ESCALON
22 [fxescalon , fyescalon]=fisMA(fase_envuelta_escalon ,0.85); %el segundo elemento es el valor de
    certeza que nosotros elegimos
23 figure , imagesc(fase_envuelta_escalon) , colormap(gray(256)); title('fase_envuelta')
24 figure , imagesc(fxescalon) , colormap(gray(256)); title('fase_corregida_por_renglones')
25 figure , imagesc(fyescalon) , colormap(gray(256)); title('fase_corregida_por_renglones_y_columnas
    ')
26
27 figure , surf(fase_escalon , 'FaceColor' , 'interp' , 'EdgeColor' , 'none' , 'FaceLighting' , 'phong')
28 view(-30,30) , camlight left , axis tight , title('Objeto_de_prueba') , zlabel('fase')
29
30 figure , surf(fyescalon , 'FaceColor' , 'interp' , 'EdgeColor' , 'none' , 'FaceLighting' , 'phong')
31 view(-30,30) , camlight left , axis tight , title('Objeto_obtenido') , zlabel('fase')
32
33 %.-ESPIRAL
34 [fxespiral , fy espiral]=fisMA(fase_envuelta_espiral ,0.85); %el segundo elemento es el valor de
    certeza que nosotros elegimos
35 figure , imagesc(fase_envuelta_espiral) , colormap(gray(256)); title('fase_envuelta')
```

```

36 figure,imagesc(fxespiral),colormap(gray(256));title('fase_corregida_por_renglones')
37 figure,imagesc(fyespiral),colormap(gray(256));title('fase_corregida_por_renglones_y_columnas
    ')
38
39 figure,surf(espiral,'FaceColor','interp','EdgeColor','none','FaceLighting','phong')
40 view(-30,30),camlight left,axis tight,title('Objeto_de_prueba'),zlabel('altura')
41
42 figure,surf(fyespiral,'FaceColor','interp','EdgeColor','none','FaceLighting','phong')
43 view(-30,30),camlight left,axis tight,title('Objeto_obtenido'),zlabel('fase')

```

El código es el mismo para la metodología B, a excepción de la función del proceso de desenvoltura empleando otro sistema difuso:

```

1 function [xx,yy]=fisMB(fase_envuelta,vc) %vc=valor de certeza
2 [r,c]=size(fase_envuelta);
3 dx=diff(fase_envuelta)';
4
5 figure,imagesc(dx),colormap(gray(256)),title('entrada_al_modelo')
6
7 dx=[zeros(r,1),dx];
8 vx=reshape(dx',1,r*c);
9 fis=readfis('meto2.fis');
10 evfisx=evalfis(vx,fis);
11
12 ax=(reshape(evfisx,c,r))';
13 figure,imagesc(ax),colormap(gray(256)),title('salida_del_modelo')
14
15 indpx=find(evfisx>vc & evfisx<(vc+1));
16 mx=zeros(1,length(vx));
17 mx(indpx)=1;
18 indnx=find(evfisx<-vc & evfisx>(-vc-1));

```

```
19 mx(indnx)=-1;
20 indp2x=find(evfisy >(vc+1));
21 mx(indp2x)=2;
22 indn2x=find(evfisy <(-vc-1));
23 mx(indn2x)=-2;
24 asinx=(reshape(mx,c,r))';
25
26 sum1=0;
27     for i=1:r
28         for j=1:c
29             sumax(i,j)=sum1-2*pi*asinx(i,j);
30             sum1=sumax(i,j);
31         end
32         sum1=0;
33     end
34 desx=fase_envuelta+sumax;
35 xx=desx;
36
37 dy=diff(desx);
38 figure, imagesc(dy), colormap(gray(256)), title('diferencias_de_fase_por_columnas')
39 dy=[zeros(1,c);dy];
40 vy=reshape(dy,1,r*c);
41 evafisy=evalfis(vy,fis);
42
43 ay=reshape(evafisy,r,c);
44 figure, imagesc(ay), colormap(gray(256)), title('salida_del_modelo_del_analisis_por_columnas')
45
46 indpy=find(evafisy>vc & evafisy<(vc+1));
47 indny=find(evafisy<=-vc & evafisy>(-vc-1));
48 indp2y=find(evafisy>vc+1);
```

```
49 indn2y=find ( evafisy <-vc -1);
50 my=zeros (1 ,length (vy));
51 my( indpy )=1;
52 my( indny )=-1;
53 my( indp2y )=2;
54 my( indn2y )=-2;
55 asiny=reshape (my, r , c);
56
57 sum2=0;
58     for k=1:c
59         for l=1:r
60             sumay (l ,k)=sum2-2*pi* asiny (l ,k);
61             sum2=sumay (l ,k);
62         end
63     sum2=0;
64 end
65 desy=desx+sumay;
66 yy=desy;
67
68 end
```

Bibliografía

- [1] Mitsuo Takeda, Hideki Ina, and Seiji Kobayashi. Fourier-transform method of fringe-pattern analysis for computer-based topography and interferometry. *JosA*, 72(1):156–160, 1982.
- [2] Liang-Chia Chen and Chu-Chin Liao. Calibration of 3D surface profilometry using digital fringe projection. *Measurement Science and Technology*, 16(8):1554, 2005.
- [3] Hong Zhao, Wenyi Chen, and Yushan Tan. Phase-unwrapping algorithm for the measurement of three-dimensional object shapes. *Applied optics*, 33(20):4497–4500, 1994.
- [4] Zhaoyang Wang, Dung A Nguyen, and John C Barnes. Some practical considerations in fringe projection profilometry. *Optics and Lasers in Engineering*, 48(2):218–225, 2010.
- [5] Pedraza-Ortega Jesus Carlos, Aceves-Fernandez Marco Antonio, Gorrostieta-Hurtado Efren, Ramos-Arreguin Juan Manuel, Sotomayor-Olmedo Artemio, Tovar-Arriaga Saul, and Vargas-Soto Jose Emilio. *Three Dimensional Reconstruction Strategies Using a Profilometrical Approach based on Fourier Transform*. INTECH Open Access Publisher, 2011.
- [6] Giovanni Simone, Alfonso Farina, Francesco Carlo Morabito, Sebastiano B Serpico, and Lorenzo Bruzzone. Image fusion techniques for remote sensing applications. *Information fusion*, 3(1):3–15, 2002.
- [7] Kannan Achan, Brendan J Frey, Ralf Koetter, and David Munson. Unwrapping phases by relaxed mean field inference. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, volume 3, pages 1901–1904. IEEE, 2001.

- [8] Ralf Koetter, Brendan J Frey, Nemanja Petrovic, and D C Munson. Unwrapping phase images by propagating probabilities across graphs. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP'01). 2001 IEEE International Conference on*, volume 3, pages 1845–1848. IEEE, 2001.
- [9] Jan Burke, Thorsten Bothe, Wolfgang Osten, and Cecil F Hess. Reverse engineering by fringe projection. In *International Symposium on Optical Science and Technology*, pages 312–324. International Society for Optics and Photonics, 2002.
- [10] Mark Jenkinson. Fast, automated, N-dimensional phase-unwrapping algorithm. *Magnetic resonance in medicine*, 49(1):193–197, 2003.
- [11] Hong Zhang, Hongjun Su, and X Y Su. Phase unwrapping algorithm for 3D reconstruction. In *Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 2007. SNPD 2007. Eighth ACIS International Conference on*, volume 2, pages 611–615. IEEE, 2007.
- [12] Howard A Zebker and Yanping Lu. Phase unwrapping algorithms for radar interferometry: residue-cut, least-squares, and synthesis algorithms. *JOSA A*, 15(3):586–598, 1998.
- [13] J M Huntley. Noise-immune phase unwrapping algorithm. *Applied Optics*, 28(16):3268–3270, 1989.
- [14] Salah A Karout, Munther A Gdeisat, David R Burton, and Michael J Lalor. Two-dimensional phase unwrapping using a hybrid genetic algorithm. *Applied optics*, 46(5):730–743, 2007.
- [15] Munther Gdeisat and Francis Lilley. One-dimensional phase unwrapping problem. *signal*, 4:6, 2012.
- [16] Dennis C. Ghiglia Pritt and Mark D. *Two-Dimensional Phase Unwrapping: theory, algorithms, and software*. Wiley-Interscience, Canada, 1998.
- [17] Hussein Abdul-Rahman. *Three-dimensional Fourier fringe analysis and phase unwrapping*. PhD thesis, Citeseer, 2007.

- [18] Mitsuo Takeda and Kazuhiro Mutoh. Fourier transform profilometry for the automatic measurement of 3-D object shapes. *Applied optics*, 22(24):3977–3982, 1983.
- [19] Miguel Arevallilo Herráez, David R Burton, Michael J Lalor, and Munther A Gdeisat. Fast two-dimensional phase-unwrapping algorithm based on sorting by reliability following a noncontinuous path. *Applied Optics*, 41(35):7437–7444, 2002.
- [20] L.A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338 – 353, 1965.
- [21] Li-Xin Wang and Jerry M Mendel. Fuzzy basis functions, universal approximation, and orthogonal least-squares learning. *Neural Networks, IEEE Transactions on*, 3(5):807–814, 1992.
- [22] Jerry M. Mendel. *Uncertain Rule-Based Fuzzy Logic Systems: introduction and new directions*. Prentice Hall PTR Upper Saddle River, 2001.
- [23] Michio Sugeno. *Industrial applications of fuzzy control*. Elsevier Science Inc., 1985.
- [24] Lei Huang, Qian Kemao, Bing Pan, and Anand Krishna Asundi. Comparison of Fourier transform, windowed Fourier transform, and wavelet transform methods for phase extraction from a single fringe pattern in fringe projection profilometry. *Optics and Lasers in Engineering*, 48(2):141–148, 2010.
- [25] R Cusack and N Papadakis. New robust 3-D phase unwrapping algorithms: application to magnetic field mapping and undistorting echoplanar images. *Neuroimage*, 16(3):754–764, 2002.
- [26] Tak-Wai Hui and Grantham Kwok-Hung Pang. 3-D measurement of solder paste using two-step phase shift profilometry. *IEEE Transactions on Electronics Packaging Manufacturing*, 31(4):306–315, 2008.
- [27] Thomas Wolf, Bernd Gutmann, and H Weber. Fuzzy logic: a new tool for 3D displacement measurements. In *Lasers, Optics, and Vision for Productivity in Manufacturing I*, pages 363–376. International Society for Optics and Photonics, 1996.

- [28] Francis Gdeisat, Munther and Lilley. Two-dimensional phase unwrapping problem. *ljmu. ac. uk*, 1:1–32, 2007.