

Tesis de la Ingeniería en Tecnología Electrónica

Método de correlación digital para el
reconocimiento de imágenes

2012



Universidad de Sonora

Repositorio Institucional UNISON



“El saber de mis hijos
hará mi grandeza”



Excepto si se señala otra cosa, la licencia del ítem se describe como openAccess



UNIVERSIDAD DE SONORA
División de Ciencias Exactas y Naturales
Departamento de Investigación en Física

Método de correlación digital para el reconocimiento de imágenes

Tesis que para obtener el título de
Ingeniero en Tecnología Electrónica

presenta

Sergio Esteban Ontiveros Gallardo

Director de tesis: Dr. Ángel Coronel Beltrán.
Hermosillo, Sonora, 17 de septiembre de 2012.

Dedicatoria

A mi madre Alma Rosa Gallardo por su apoyo incondicional y amor.

*A mi novia Rosa Alejandra Ortega que se mantuvo siempre atenta
durante el desarrollo de mi trabajo.*

Agradecimientos

Al Dr. Ángel Coronel Beltrán, director de tesis, por su disposición y paciencia durante la realización de todo el trabajo.

A los miembros del comité de tesis: Dr. Alejandro García Juárez, Dr. Ángel Coronel Beltrán y Dra. Alicia Vera Marquina por sus valiosos comentarios y muestras de apoyo.

A mi madre Alma Rosa Gallardo que nunca vaciló en consentirme mientras me veía trabajar pegado a la computadora.

A mi novia Rosa Alejandra Ortega, mi mayor inspiración para acelerar la conclusión de este trabajo.

Al Dr. Julio Waissman y a mi amigo Raúl Leonel Grjalva por su apoyo y asesoría con la cual contribuyeron para un mejor diseño de este trabajo.

Índice general

Introducción	1
Objetivos	2
Objetivo general	2
Objetivos particulares	2
Organización del trabajo	3
1. Herramientas matemáticas	5
1.1. Transformada de Fourier (TF)	5
1.2. Transformada Discreta de Fourier (TDF)	8
1.3. Transformada de Mellin	10
1.4. Transformada de Escala	11
1.5. Correlación discreta	12
2. Filtros y métricas de desempeño	13
2.1. Filtro digital	13
2.1.1. Filtro no lineal (NLF)	13
2.1.2. Filtro clásico acoplado (CMF)	14
2.1.3. Filtro solo de fase (POF)	14
2.1.4. Filtro inverso (IF)	15
2.2. Métricas de desempeño	15
2.2.1. Métrica Energía de Pico de Correlación (PCE)	16
3. Metodología	17
3.1. Algoritmo	18

3.2. Representación gráfica del algoritmo de reconocimiento	19
4. Resultados	21
4.1. Elección del módulo de la TF	21
4.2. Reconocimiento de imágenes	23
4.2.1. Imágenes escaladas	26
4.2.2. Imágenes rotadas	27
4.3. Costo computacional	31
Conclusiones	33
A. Programas en Matlab	35
A.1. Programa que compara el módulo de la TF con diferentes exponentes . . .	35
A.2. Programa que compara imágenes escaladas y rotadas	36
A.3. Funciones	37
A.3.1. Función compspectrum	37
A.3.2. Función factorro	37
A.3.3. Función mapolar	38
A.3.4. Función maplog2	38
A.3.5. Función Trans_Escala	39
A.3.6. Función Metrica_PCE	39

Introducción

El procesamiento digital de imágenes está tan cerca de nosotros y muchas veces no nos damos cuenta, constantemente son más los campos de la ciencia que requieren realizar investigación apoyados por el procesamiento digital de imágenes. Uno de los principales campos en los que se realiza este tipo de investigación que posiblemente tiene más impacto en la vida cotidiana es quizás en la medicina, la identificación de tumores cerebrales a través de imágenes, la amplificación de imágenes de rayos x de los huesos, entre otros, son ejemplos claros de la aplicación del procesamiento digital de imágenes. Sin embargo, el procesamiento digital de imágenes no está limitado a la medicina, en la actualidad distintos grupos de investigación realizan trabajo de este tipo, en ramas como electrónica, astronomía, biología, física, agricultura, geografía, antropología, sólo por mencionar algunos.

Una imagen puede ser definida como una función bidimensional $f(n_1, n_2)$ donde n_1 y n_2 son coordenadas espaciales, y la amplitud de f para cada par de coordenadas (n_1, n_2) es conocida como la intensidad de la imagen en ese punto. Cuando los valores de n_1 , n_2 y f son finitos y discretos, podemos llamar a la imagen como imagen digital. Sabemos entonces que una imagen está compuesta por un número finito de elementos, los cuales tienen una posición particular y un valor. Estos elementos son popularmente nombrados como pixeles (pixel: acrónimo del inglés *picture element*, “elemento de imagen”).

Conforme la tecnología avanza, es común ver cómo se va creando una gran cantidad de robots encaminados a realizar actividades industriales, militares o cotidianas que resultan difíciles, tediosas o peligrosas para ser realizadas por humanos, la implementación de este tipo de dispositivos se presenta con mayor frecuencia estos días y conforme más actividades se les otorguen, mayor será la necesidad de que estos robots mejoren su capacidad

visual, no solo para la manipulación de productos industriales, sino para que sean capaces de procesar la información de su entorno tal como lo hacemos los humanos y de esta forma puedan aprender a interactuar más adecuadamente. Aunado a eso, es conveniente mencionar que pese a que el sentido de la vista en los humanos es un sentido altamente desarrollado, se encuentra limitado en cuanto al espectro electromagnético que es capaz de percibir, sin embargo, actualmente existen robots capaces de percibir visualmente casi en su totalidad el espectro electromagnético, en rangos que van de rayos gamma hasta las microondas. Los robots pueden trabajar con imágenes producidas por una gran cantidad de fuentes distintas a las que un humano sería capaz, lo cual trae consigo más campos de interés al procesamiento digital de imágenes.

Cuando se intenta simular el sentido de la vista de los humanos mediante algún tipo de algoritmo, es necesaria una comparación de ciertos patrones que contenga la imagen que quiera reconocerse con patrones provenientes de la imagen visualizada, este tipo de reconocimiento de patrones se hace a través de una función matemática conocida como *correlación*, la cual nos permite determinar que tan parecido es un objeto a otro de forma cuantificada. Algunos sistemas de correlación existentes en la actualidad permiten en ciertos casos reconocer objetos aun cuando este ha sido afectado en escala, posición y rotación.

Objetivos

Objetivo general

Elaborar un algoritmo computacional capaz de reconocer objetos en imágenes digitales invariante a posición, escala y rotación para la clasificación de estos mismos.

Objetivos particulares

1.- Elaborar algoritmos computacionales para la realización de un filtro de correlación no lineal.

2.- Incorporar invariancia a desplazamiento, escala y rotación al filtro de correlación diseñado.

3.- Análisis del comportamiento de la métrica PCE en el sistema no lineal invariante para imágenes digitales.

Organización del trabajo

La presente tesis fue organizada de la siguiente manera:

En el capítulo 1 se presentan los fundamentos matemáticos utilizados en esta tesis durante el procesamiento de imágenes. Entre los cuales se encuentran las transformadas de Fourier, Mellin y de escala con sus propiedades características. Además, se introduce la función de correlación.

En el capítulo 2 se presenta el concepto teórico y matemático de diferentes filtros que fueron considerados para el desarrollo del algoritmo de esta tesis (NLF, CMF, IF y POF), sus características principales y las ventajas que presenta cada uno, así como la descripción de la métrica que evaluará el desempeño del algoritmo de reconocimiento de cada filtro.

En el capítulo 3 se presenta la metodología utilizada en la realización de la correlación digital no lineal invariante a translación, rotación y escala para el reconocimiento de patrones de diferentes objetos tanto artificiales como de la vida real mediante las operaciones de la transformada de Fourier, el mapeo de coordenadas cartesianas a polares y la transformada de escala, respectivamente.

En el capítulo 4 se presentan los resultados obtenidos al comparar distintas imágenes de diatomeas escaladas y rotadas, evaluados por la métrica PCE. Se prueba la efectividad del algoritmo y se muestra el costo computacional del mismo.

Herramientas matemáticas

1.1. Transformada de Fourier (TF)

Diferentes campos incluyendo medicina, óptica, física e ingeniería electrónica usan la Transformada de Fourier como una herramienta común. En la práctica, los grupos estándar de compresión JPEG (Joint Photographic Experts Group) y MPEG (Motion Picture Experts Group) utilizan formas modificadas de la transformada de Fourier. En general, la TF de una señal nos permite ver su información en el dominio de la frecuencia en lugar de la información en el dominio del tiempo.

Dada una señal $x(t)$, la transformada de Fourier denotada por $X(w)$ está definida como[1]:

$$X(w) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt, -\infty < w < \infty \quad (1.1)$$

donde $j = \sqrt{-1}$ y w es la variable continua de frecuencia.

Se dice que una señal $x(t)$ tiene TF en el sentido ordinario, si la integral existe. La integral existe si $x(t)$ tiene un número finito de discontinuidades, máximas y mínimas, dentro de un intervalo finito de tiempo, y si $x(t)$ es completamente integrable, esto significa:

$$\int_{-\infty}^{\infty} |x(t)| dt < \infty \quad (1.2)$$

De la misma manera, teniendo la TF $X(w)$ de una señal, es posible obtener la señal

original $x(t)$ aplicando la Transformada Inversa de Fourier dada por[1]:

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(w)e^{j\omega t} dw \quad (1.3)$$

Dado que en el presente trabajo realizaremos este tipo de transformaciones a imágenes, las cuales se pueden representar como secuencias (señales) discretas de espacio con dos componentes enteros, el análisis discreto será muy importante y lo explicaremos un poco más adelante. Las secuencias que utilizaremos pueden ser representadas entonces como $g(x, y)$, las cuales están definidas para todos los valores enteros de x y y . La TF estaría dada por[2]:

$$F(g) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x, y)e^{-j2\pi(f_x x + f_y y)} dx dy \quad (1.4)$$

La transformada definida es por sí misma una función de valores complejos de dos variables independientes f_x y f_y , a las cuales nos referimos generalmente como frecuencias. De forma similar, la Transformada Inversa de Fourier de una función $G(f_x, f_y)$ será representada por[2]:

$$F^{-1}(G) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} G(f_x, f_y)e^{j2\pi(f_x x + f_y y)} df_x df_y \quad (1.5)$$

Trabajar con una señal en el dominio de la frecuencia aporta una gran cantidad de información de la señal que al ser trabajada en el dominio del tiempo sería imposible descifrar. Sin embargo, para aprovechar de mejor manera todo lo que la señal puede ofrecernos, es necesario conocer ciertas propiedades de la TF. Tales como[2]:

1.-Teorema de linealidad. $F(\alpha g + \beta h) = \alpha F(g) + \beta F(h)$; Esta propiedad nos dice que la Transformada de la suma ponderada de dos o más funciones es igual a la suma ponderada de las Transformadas de forma individual.

2.-Teorema de Similaridad. Si $F[g(x, y)] = G(f_x, f_y)$, entonces

$$F[g(ax, by)] = \frac{1}{|ab|} G\left(\frac{f_x}{a}, \frac{f_y}{b}\right) \quad (1.6)$$

Esta propiedad nos dice que cualquier estiramiento de las coordenadas en el dominio espacial (x, y) se traducirá en una contracción de las coordenadas en el dominio de la

frecuencia (f_x, f_y) más un cambio en la amplitud total del espectro.

3.-Teorema de traslación. Si $F[g(x, y)] = G(f_x, f_y)$, entonces

$$F[g(x - a, y - b)] = G(f_x, f_y)e^{-j2\pi(f_x a + f_y b)} \quad (1.7)$$

Esta propiedad nos dice que una traslación en el dominio del espacio introduce una traslación lineal en el dominio de la frecuencia.

4.-Teorema de Parseval. Si $F[g(x, y)] = G(f_x, f_y)$, entonces

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |g(x, y)|^2 dx dy = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |G(f_x, f_y)|^2 df_x df_y \quad (1.8)$$

En esta propiedad la integral de la parte izquierda puede ser interpretada como la energía contenida en la forma de onda $g(x, y)$. Lo cual lleva a la idea de que la cantidad $|G(f_x, f_y)|^2$ puede ser interpretada como la densidad de energía en el dominio de la frecuencia.

5.-Teorema de convolución. Si $F[g(x, y)] = G(f_x, f_y)$ y $F[h(x, y)] = H(f_x, f_y)$, entonces

$$F\left[\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(\xi, \eta)h(x - \xi, y - \eta)d\xi d\eta\right] = G(f_x, f_y)H(f_x, f_y) \quad (1.9)$$

Esta propiedad nos dice que la convolución de dos funciones en el dominio espacial es equivalente a la multiplicación de sus transformadas individuales.

6.-Teorema de autocorrelación. Si $F[g(x, y)] = G(f_x, f_y)$, entonces

$$F\left[\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(\xi, \eta)g^*(\xi - x, \eta - y)d\xi d\eta\right] = |G(f_x, f_y)|^2 \quad (1.10)$$

De forma similar

$$F[|g(x, y)|^2] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} G(\xi, \eta)G^*(\xi - f_x, \eta - f_y)d\xi d\eta \quad (1.11)$$

Este teorema es considerado un caso especial del teorema de correlación en el cual se correlacionan $g(x, y)$ con $g^*(-x, -y)$, donde * es el complejo conjugado

La TF está envuelta por una gran cantidad de propiedades, que para casos muy es-

pecíficos son útiles, sin embargo, para el propósito de esta tesis se tomaron solo estas seis propiedades que nos parecieron más relevantes.

1.2. Transformada Discreta de Fourier (TDF)

El análisis de señales en tiempo discreto es usualmente utilizado en el procesamiento digital de señales. Para realizar un análisis en frecuencia de una señal en tiempo discreto $x(n)$, necesitamos convertir la secuencia en el dominio del tiempo a su representación equivalente en el dominio de la frecuencia. La transformada de Fourier para la señal ya mencionada será representada entonces como $X(w)$. Sin embargo, $X(w)$ es una función de frecuencia continua y por lo tanto no es una representación conveniente para la señal $x(n)$. Como ya sabemos, computacionalmente es imposible llevar a cabo la representación de una señal continua. De forma digital, solo se harán aproximaciones a una señal continua, por eso mismo representaremos la secuencia $x(n)$ con muestras de su espectro $X(w)$, de esta manera su representación en la frecuencia nos llevaría a la Transformada Discreta de Fourier (TDF)[3], la cual es una poderosa herramienta computacional para la realización de análisis frecuenciales de señales en tiempo discreto.

Dada una señal en tiempo discreto $x[n]$, la TDF estará dada por [1]:

$$X[\Omega] = \sum_{n=-\infty}^{\infty} x[n]e^{-j\Omega n} \quad (1.12)$$

La TDF es en general una función valuada de complejos de la variable real Ω . Notar que la ecuación anterior es la equivalente en tiempo discreto a la TF, donde la integral es reemplazada por una sumatoria.

Se dice que una señal $x[n]$ en tiempo discreto tiene TDF en el sentido ordinario, si la sumatoria converge, es decir, si la sumatoria es finita para todos los valores reales de Ω , esto significa:

$$\sum_{n=-\infty}^{\infty} |x[n]| < \infty \quad (1.13)$$

Tomando en cuenta nuevamente que se trabajará con secuencias discretas bidimensionales, podemos considerar una secuencia $x_0(n_1, n_2)$, la cual es periódica con periodo de $N_1 \times N_2$. Supongamos que formamos una señal con extensión finita $x(n_1, n_2)$ que contenga

un periodo de $x_0(n_1, n_2)$ y todos los demás valores en cero. Es decir:

$$x(n_1, n_2) = x_0(n_1, n_2)R_{N_1 \times N_2}(n_1, n_2) \quad (1.14)$$

donde

$$R_{N_1 \times N_2}(n_1, n_2) = \begin{cases} 1, & 0 \leq n_1 \leq N_1 - 1, 0 \leq n_2 \leq N_2 - 1 \\ 0, & \text{cualquier otro caso} \end{cases} \quad (1.15)$$

Esto muestra claramente que podemos determinar $x_0(n_1, n_2)$ de $x(n_1, n_2)$ por:

$$x_0(n_1, n_2) = \sum_{r_1=-\infty}^{\infty} \sum_{r_2=-\infty}^{\infty} x(n_1 - r_1 N_1, n_2 - r_2 N_2) \quad (1.16)$$

Ahora, consideremos $X_0(k_1, k_2)$ la Serie Discreta de Fourier de $x_0(n_1, n_2)$. Supongamos que formamos una señal con extensión finita $X(k_1, k_2)$ que contenga un periodo de $X_0(k_1, k_2)$ y todos los demás valores en cero. Es decir:

$$X(k_1, k_2) = X_0(k_1, k_2)R_{N_1 \times N_2}(k_1, k_2) \quad (1.17)$$

Notamos aquí también que el proceso es invertible ya que podemos obtener $X_0(k_1, k_2)$ de $X(k_1, k_2)$ por:

$$X_0(k_1, k_2) = \sum_{r_1=-\infty}^{\infty} \sum_{r_2=-\infty}^{\infty} X(k_1 - r_1 N_1, k_2 - r_2 N_2) \quad (1.18)$$

Finalmente tenemos nuestro par de TDF dadas por[4]:

$$X(k_1, k_2) = \begin{cases} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} x(n_1, n_2) e^{-j(\frac{2\pi}{N_1})k_1 n_1} e^{-j(\frac{2\pi}{N_2})k_2 n_2}, & 0 \leq k_1 \leq N_1 - 1, 0 \leq k_2 \leq N_2 - 1 \\ 0, & \\ \text{cualquier otro caso} & \end{cases} \quad (1.19)$$

$$x(n_1, n_2) = \begin{cases} \frac{1}{N_1 N_2} \sum_{k_1=0}^{N_1-1} \sum_{k_2=0}^{N_2-1} X(k_1, k_2) e^{-j(\frac{2\pi}{N_1})k_1 n_1} e^{-j(\frac{2\pi}{N_2})k_2 n_2}, \\ \quad 0 \leq n_1 \leq N_1 - 1, 0 \leq n_2 \leq N_2 - 1 \\ 0, \\ \text{cualquier otro caso} \end{cases} \quad (1.20)$$

De las transformadas anteriores tenemos que un punto en $N_1 \times N_2$ de la secuencia $x(n_1, n_2)$ está representado en el dominio de la frecuencia por un punto en $N_1 \times N_2$ de la secuencia $X(k_1, k_2)$. La secuencia $X(k_1, k_2)$ es conocida como la Transformada Discreta de Fourier de $x(n_1, n_2)$, mientras que $x(n_1, n_2)$ es conocida como la Transformada Discreta de Fourier Inversa (TDF Inversa) de $X(k_1, k_2)$.

Estos algoritmos ya están incluidos en el software de MATLAB, y acceder a ellos no es difícil dado que las funciones tienen un formato muy sencillo. Para obtener la TDF en MATLAB se utiliza la función **fft** mediante la siguiente sintaxis:

$$F = \text{fft}(x, n)$$

Esta función asigna al vector F la transformada discreta de Fourier de la señal discreta x que en este caso es un vector también con n número de muestras.

De esta misma forma, es posible reconstruir la señal original en MATLAB a partir de la TDF, haciendo uso de la función **ifft** y sujetos a la siguiente sintaxis:

$$x = \text{ifft}(F, n)$$

En esta función se asigna al vector x la TDF inversa del espectro de Fourier F con n muestras.

1.3. Transformada de Mellin

La Transformada de Mellin fue desarrollada por Robert Hjalmar Mellin(1854-1933) para el estudio de funciones hipergeométricas, de la función zeta de Riemann, para la solución de ecuaciones diferenciales parciales, entre otras cosas. La transformada de Mellin se ha utilizado también en ingeniería eléctrica para el estudio del control de un sistema de motores, por ejemplo.

La Transformada de Mellin puede representar una señal en términos de la escala. La escala puede ser representada de forma similar a la frecuencia como un atributo físico de las señales. La Transformada de Mellin está dada por[5]:

$$M_f(p) = \int_0^{\infty} f(t)t^{p-1}dt \quad (1.21)$$

Donde p es una variable compleja dada por $p = -jc + \beta$. Aquí β es un parámetro fijo que pertenece a \mathbb{R} y c es una variable independiente que también pertenece a \mathbb{R} . Dado que la parte real de la variable compleja p es parametrizada, el parámetro β nos permite seleccionar entre:

- a) Una transformada invariante a escala con $\beta = \frac{1}{2}$
- b) Una transformada invariante a compresión y expansión con $\beta = 0$
- c) Una transformada invariante a forma con $\beta = -1$

La existencia de la transformada de Mellin depende de la convergencia de su integral, es decir:

$$\int_0^{\infty} |f(t)|t^{p-1}dt < \infty \quad (1.22)$$

La transformada de Mellin es muy importante en el procesamiento de imágenes. De hecho, una transformación Fourier-Mellin puede ser usada para el reconocimiento de patrones en imágenes con invariancia a translación, rotación y escala.

1.4. Transformada de Escala

Como ya lo vimos en la sección anterior, podemos obtener una transformada de escala partiendo inicialmente de la transformada de Mellin con el parámetro $\beta = \frac{1}{2}$. La transformada de escala nos es de gran utilidad debido a que presenta una mayor sensibilidad a la escala que pueda presentar en este caso una imagen. Si tomamos la variación a escala que puede tener una imagen como una variable que denotaremos con la letra c , obtenemos una pareja de transformadas definidas por[6]:

$$D_f(c) = \frac{1}{\sqrt{2\pi}} \int_0^{\infty} f(t)e^{(-jc - \frac{1}{2})lnt}dt \quad (1.23)$$

y

$$f(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} D_f(c) e^{(jc - \frac{1}{2})lnt} dc \quad (1.24)$$

En donde la primer ecuación representa la transformada de escala y la segunda ecuación representa la transformada de escala inversa.

La característica primordial de esta transformada es la invariancia que presenta a los cambios de escala que pueda presentar la imagen, la escala de una imagen puede variar comprimiendo o expandiendo sus ejes, por lo tanto si f es una imagen y g representa a la misma imagen con una variación en su escala, la transformada de escala de f y g será la misma.

1.5. Correlación discreta

Para una señal en dos dimensiones, podemos definir a la correlación discreta como[7]:

$$f(x, y) \otimes g(x, y) = \frac{1}{MN} \sum_{m=0}^M \sum_{n=0}^N f(m, n)g(x + m, y + n) \quad (1.25)$$

Lo cual podemos escribir como:

$$F[f(x, y) \otimes g(x, y)] = F(u, v)G^*(u, v) \quad (1.26)$$

Que nos indica que la TF de la correlación de dos señales en tiempo es igual al producto en la frecuencia de la TF de la primer señal por el complejo conjugado de la TF de la segunda señal. Así mismo tenemos:

$$F[f(x, y)g^*(x, y)] = F(u, v) \otimes G(u, v) \quad (1.27)$$

Aquí se obtiene que la TF del producto en el espacio de una señal por el complejo conjugado de otra señal es equivalente a la correlación de la TF de estas dos señales.

Finalmente concluimos que la correlación la podemos calcular como:

$$f(x, y) \otimes g(x, y) = F^{-1}[F(u, v)G^*(u, v)] \quad (1.28)$$

Filtros y métricas de desempeño

2.1. Filtro digital

Los filtros digitales tienen su origen teórico en los filtros análogos utilizados en electrónica. La utilización de la computadora como un medio de simulación permitió el desarrollo de una gran cantidad de procesos electrónicos de forma digital, entre ellos la realización e implementación de filtros digitales sin la necesidad de armar ni conectar un solo componente.

Existe una gran cantidad de variantes en cuanto a la forma de desarrollo de un filtro en base al dominio al que se está trabajando, sin embargo, la base matemática es la misma. Un filtro digital es básicamente un algoritmo que permite el paso de ciertos datos y reduce o elimina el valor de otros datos al ser utilizado en una señal, que para el propósito de esta tesis, es una imagen.

2.1.1. Filtro no lineal (NLF)

Un filtro no lineal es un filtro en el cual su salida se representa como función no lineal de su entrada. Los filtros no lineales se pueden representar como[8]:

$$NLF = |H(u, v)|^k e^{-j\phi(u, v)} \quad (2.1)$$

Donde $|H(u, v)|$ representa el módulo de la TF del objeto a reconocer, j es un número imaginario, k es el factor de no linealidad que puede tomar valores de $0 < k < 1$ y $\phi(u, v)$ es la fase de la TF. Al variar el valor de k dentro de este rango, podemos modificar la

capacidad discriminante del procesado no lineal y determinar el valor ideal k de los filtros no lineales. Si k toma valores fuera de este rango, tendremos filtros lineales que no son de nuestro interés para el desarrollo de esta tesis, entre ellos: filtro clásico acoplado para $k = 1$, filtro solo de fase para $k = 0$ y filtro inverso para $k = -1$.

Comúnmente el uso de un filtro no lineal presenta mayores ventajas sobre un filtro lineal, debido a que los filtros lineales son muy sensibles a pequeños cambios en la señal de entrada. Cuando se analiza un objeto con filtros lineales, el plano de correlación resultante presenta ruido lateral fuera del área del objeto, con lo cual se aumenta la probabilidad de un mal análisis de la información, haciendo a este tipo de filtro poco fiable al momento de procesar imágenes. Por otro lado, el filtro no lineal tiene una gran capacidad para discriminar objetos, el plano de correlación resultante presenta menos ruido y un pico mejor definido dentro del área del objeto.

2.1.2. Filtro clásico acoplado (CMF)

También conocido como filtro complejo u holográfico, CMF por sus siglas en inglés (Classical Matched Filter), este filtro fue propuesto por Vander Lugt en 1964, él lo describió como una función compleja de la siguiente forma[7]:

$$H(u, v)_{cmf} = |H(u, v)|e^{j\phi(u, v)} \quad (2.2)$$

En donde (u, v) son las variables en frecuencia, $|H(u, v)|$ es el módulo de la TF de la señal y $e^{j\phi(u, v)}$ representa la respuesta de fase, en esta última parte, $\phi(u, v)$ representa la fase de la TF de la señal.

Problemas comunes en la utilización de este filtro son la alta sensibilidad a distorsiones, mal funcionamiento para imágenes que contienen al objeto espacialmente separado del fondo y un pico de correlación amplio.[9]

2.1.3. Filtro solo de fase (POF)

POF por sus siglas en inglés (Phase Only Filter), este tipo de filtro se caracteriza por maximizar la eficacia de la luz y cuya principal desventaja se presenta al momento de hacer el reconocimiento de un objeto en una imagen con poco contraste y con fondo complicado.

Tomando la ecuación (2.2) del filtro clásico, de la cual sabemos que $H(u, v)$ es la magnitud y $\phi(u, v)$ la fase del filtro. Consideramos que no se presentan pérdidas a través del sistema, por lo tanto la magnitud permanece constante y solo habrá cambios en la fase, de aquí se normaliza $|H(u, v)| = 1$ y podemos representar a un filtro solo de fase como[10]:

$$H_{POF}(u, v) = e^{j\phi(u, v)} \quad (2.3)$$

2.1.4. Filtro inverso (IF)

IF por sus siglas en inglés (Inverse filter), una de las ventajas de la utilización de este tipo de filtro es la reducción del criterio PCE (Peak to Correlation Energy), que básicamente indica un pico de correlación muy angosto. Si la imagen de referencia y la imagen problema son iguales, entonces tendremos una función Delta como resultado de la correlación. Una desventaja de la utilización de este filtro es la presencia de una gran cantidad de ruido de fondo en el plano de salida. Definimos al filtro inverso como[10]:

$$H(u, v) = \beta \frac{T^*(u, v)}{|T(u, v)|^2} \equiv H_{IF(u, v)} \quad (2.4)$$

En donde β es una constante arbitraria, $T(u, v)$ es la TF de la imagen y $T^*(u, v)$ su complejo conjugado.

2.2. Métricas de desempeño

Las métricas de desempeño calculan la eficiencia con la cual está trabajando el filtro utilizado en el reconocimiento de patrones. Existe una gran cantidad de métricas para medir la eficiencia de los filtros de correlación, entre los cuales se encuentra: la relación energía del pico de correlación (PCE), la relación de la capacidad de discriminación (DC), la relación señal a ruido (SNR), y la relación de eficiencia de luz (LE), solo por mencionar las más utilizadas. Durante el desarrollo de este trabajo, utilizamos solamente la métrica PCE.

2.2.1. Métrica Energía de Pico de Correlación (PCE)

La métrica PCE, representa la relación que existe entre la energía en el origen medida en el plano de correlación y la energía total en el plano de correlación. Se define entonces como[9]:

$$PCE = \frac{|E\{c(0,0)\}|^2}{E\{|c(x,y)|^2\}} \quad (2.5)$$

De donde sabemos que la parte del numerador representa el valor esperado de la intensidad del pico de correlación y el denominador representa el valor esperado de la energía promedio en el plano de correlación.

Metodología

En este capítulo se presentará la metodología que se utilizó para desarrollar un algoritmo capaz de identificar una imagen en específico de un banco de imágenes con características muy parecidas. Además de presentar opciones en desarrollo para fortalecer la invariancia a posición, escala y rotación de imágenes de objetos de la vida real.

Esta tesis se inspiró en una estancia de investigación científica realizada en CICESE el verano del 2011, en dicha estancia conocí a un grupo de investigadores de diferentes áreas de conocimiento, los cuales trabajaban con un conjunto de algoritmos diferentes para el reconocimiento de patrones de imágenes digitales, fue así que interesado en el tema acudí a un investigador de este grupo para trabajar en un tema relacionado con estos algoritmos que estaban trabajando.

Decidimos trabajar con un conjunto de transformadas; transformada de Fourier y transformada de escala[6][5], las cuales conjuntas con un cambio de coordenadas de cartesianas a polares nos brindan la invariancia a posición, escala y rotación. La transformada de escala fue de mayor interés que la transformada de Mellin[5] en el desarrollo de esta tesis debido a que es más sensible a cambios de escala. Utilizamos un filtro no lineal dadas las ventajas descritos en la unidad anterior, las cuales consisten en un un pico de correlación bien localizado, alta capacidad para discriminar objetos y menos ruido en el plano de salida. Como lo mencionamos anteriormente, el filtro no lineal contiene un parámetro de no linealidad k que puede tomar valores de $0 < k < 1$, el valor con el cual trabajaremos a lo largo de esta tesis es $k = 0.3$ porque ha sido demostrado que este valor nos otorga los mejores resultados del filtro[7].

En el desarrollo de esta tesis, todos los algoritmos computacionales y la obtención de

los resultados fueron probados y obtenidos en MATLAB versión 7.13.0.564 R2011b a 32 bits de The Math Works, Inc.

3.1. Algoritmo

A continuación se hace una descripción del algoritmo que se utilizó para el desarrollo del programa en Matlab capaz de discriminar nuestra imagen filtro de un conjunto de imágenes problema.

- 1.- Obtenemos la imagen filtro por una función de la forma $f(x, y)$.
- 2.- Utilizamos la interpolación bilineal para trabajar con imágenes de 256×256 .
- 3.- Calculamos el módulo de la transformada rápida de Fourier en dos dimensiones de esta imagen para tener la invariancia a posición.
- 4.- Aplicamos un filtraje pasa altas del módulo de la transformada de Fourier para resaltar las altas frecuencias.
- 5.- Agregamos el factor de escala \sqrt{r} , donde r es la frecuencia radial.
- 6.- Pasamos de coordenadas cartesianas a coordenadas polares para la invariancia a rotación.
- 7.- Se genera un escalamiento logarítmico en la parte radial en coordenadas polares para la invariancia a escala.
- 8.- Utilizamos la interpolación bilineal para reducir la fuga de información debido al muestreo logarítmico-polar.
- 9.- Calculamos nuevamente la transformada rápida de Fourier en dos dimensiones.
- 10.- Utilizamos el filtro no lineal el cual contiene implícitas las invariancias a rotación, escala y translación en el módulo y la fase de la transformada de Fourier.
- 11.- Realizamos los mismos cálculos para la imagen problema.
- 12.- Correlacionamos los espectros resultantes de ambas imágenes.
- 13.- Utilizamos la métrica PCE para obtener el rendimiento del filtro.
- 14.- Graficamos resultados.

Podemos interpretar el algoritmo tal y como se muestra en el diagrama de la figura 3.1, de donde $|T|$ representa el módulo la transformada de Fourier y ϕ la fase de la transformada de Fourier del objeto.

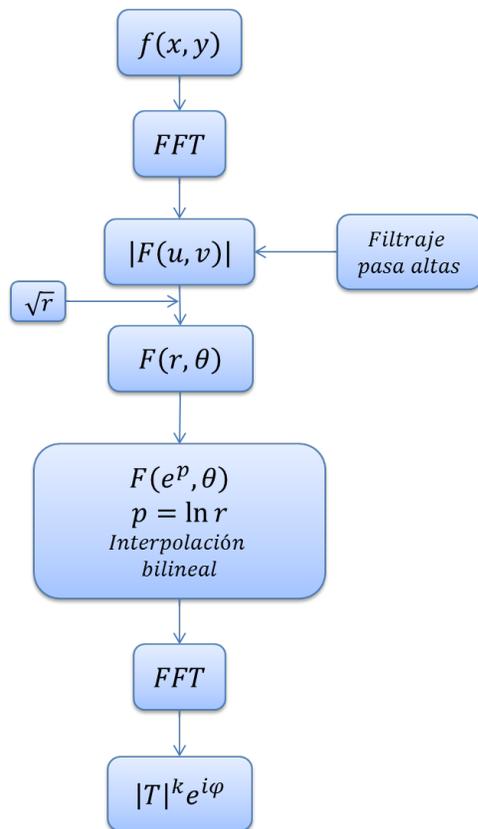


Figura 3.1: Diagrama de bloques del algoritmo de reconocimiento de patrones en imágenes digitales.

3.2. Representación gráfica del algoritmo de reconocimiento

Es importante conocer cuales son los resultados obtenidos paso a paso en la implementación del algoritmo de reconocimiento de imágenes para hacernos una mejor idea de cómo es afectado cada espectro al atravesar cada uno de los pasos que se van dando. Es por eso que en la figura 3.2 se presenta la evolución de los espectros resultantes a cada paso de una imagen que fue escalada y rotada para su comparación con la imagen original. En esta figura podemos observar que la transformada de Fourier de cada imagen rotada gira el mismo ángulo que el objeto, el mapeo logarítmico-polar se recorre horizontalmente dependiendo del ángulo de rotación del objeto, la transformada de Escala para las imágenes rotadas permanece prácticamente igual en cada caso, excepto para los casos de las imágenes escaladas, en donde sufre ciertas variaciones de intensidad.

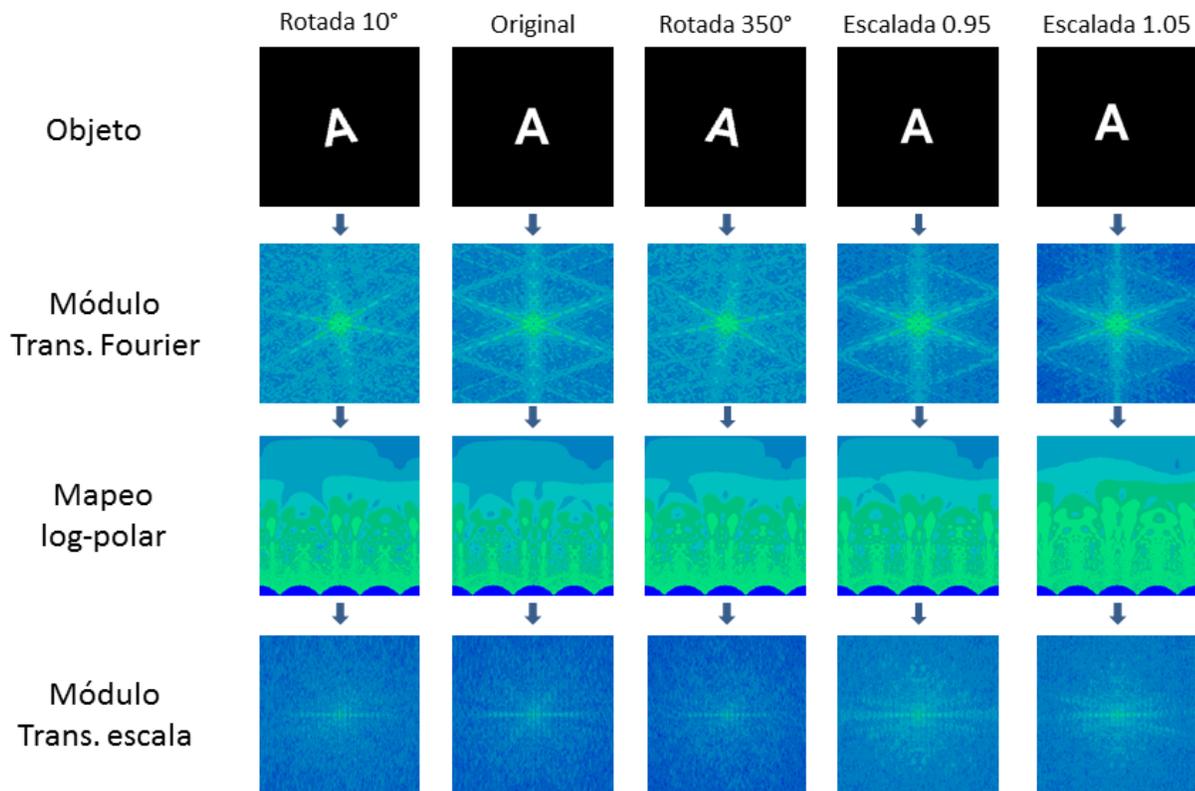


Figura 3.2: Procedimiento que sigue una imagen tratada por el algoritmo de reconocimiento para el caso de la imagen original, imagen rotada 10 grados, imagen rotada 350 grados, imagen escalada en 0.95 e imagen escalada en 1.05.

Resultados

Durante el desarrollo de este trabajo de tesis, todos los programas y resultados fueron corridos y obtenidos con una computadora portátil Dell inspiron modelo 1545, con un procesador Intel(R) Core Duo T6600 2.20GHz, con 3GB de memoria RAM y sistema operativo Windows 7 a 32 bits.

4.1. Elección del módulo de la TF

En la práctica, nos hemos dado cuenta que al afectar el módulo de la TF con un exponente los picos en el plano de correlación lucen más definidos y por lo tanto con una intensidad mayor, tal y como se muestra en la figura 4.1, en esta figura se muestra el espectro de autocorrelación de una diatomea a la cual se le fue variando el exponente del módulo de la TF de 0.2 en 0.2 hasta 2, sin embargo, al evaluar la relación de los picos de autocorrelación con el ruido que se genera en el plano de salida mediante la métrica PCE como se ve en la figura 4.2 y en el cuadro 4.1 nos damos cuenta que los mejores resultados se obtienen con el módulo elevado a la 1, en todos los demás valores que fueron considerados la calidad del reconocimiento de la imagen fue decayendo.

En la imagen de la figura 4.3 se muestra los espectros a diferente grado del proceso del algoritmo para tres potencias distintas del módulo de la TF, aquí podemos apreciar claramente que en efecto, en la práctica, el uso de estas diferentes potencias ayuda definir mejor los picos de salida en frecuencia para las imágenes procesadas, podría pensarse entonces que los mejores resultados se obtendrían para un exponente cuadrático, sin embargo, ya vimos que el exponente uno ofrece los mejores resultados.

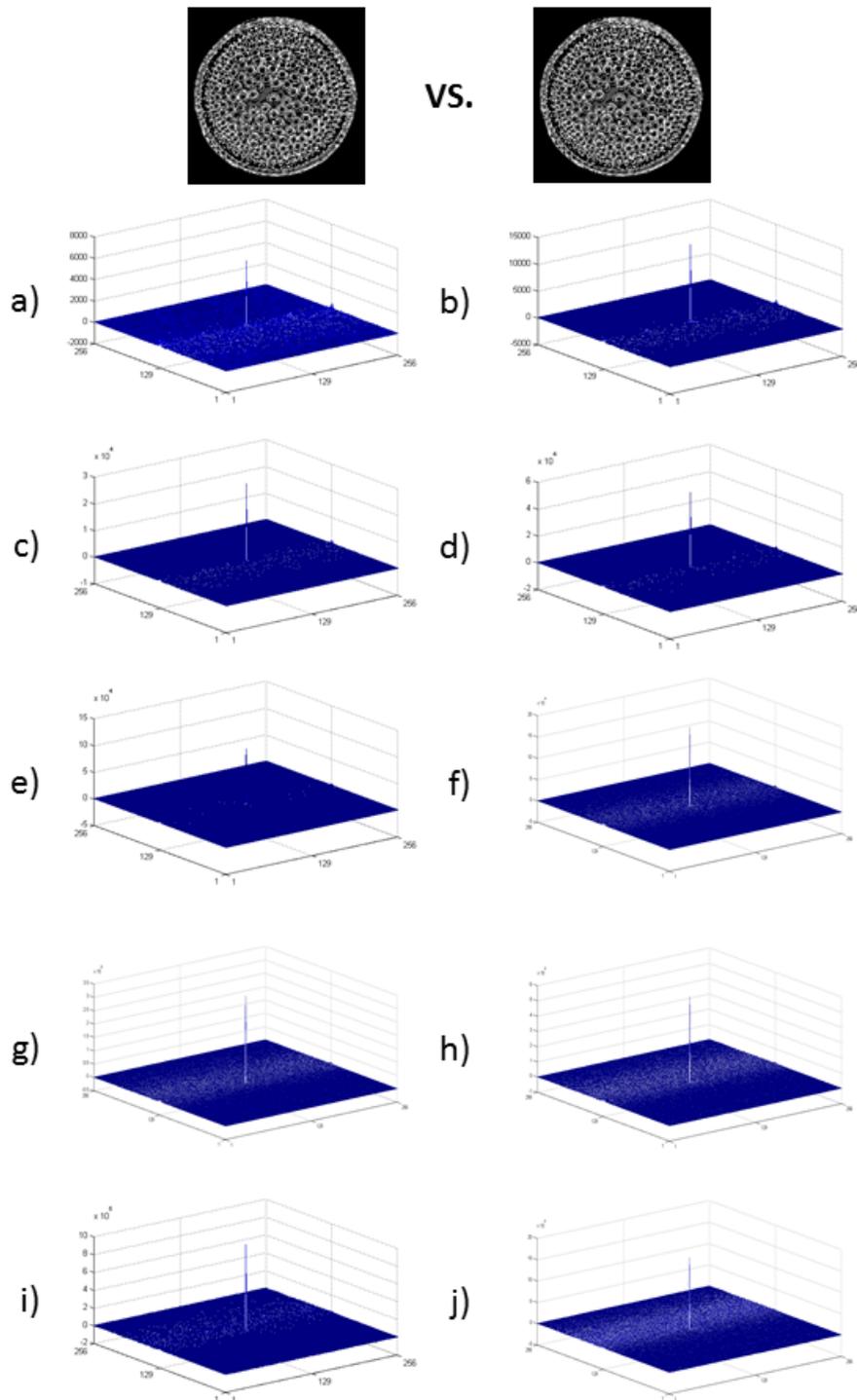


Figura 4.1: Espectros de autocorrelación con potencia en el módulo de la TF de: a)0.2 b)0.4 c)0.6 d)0.8 e)1 f)1.2 g)1.4 h)1.6 i)1.8 j)2

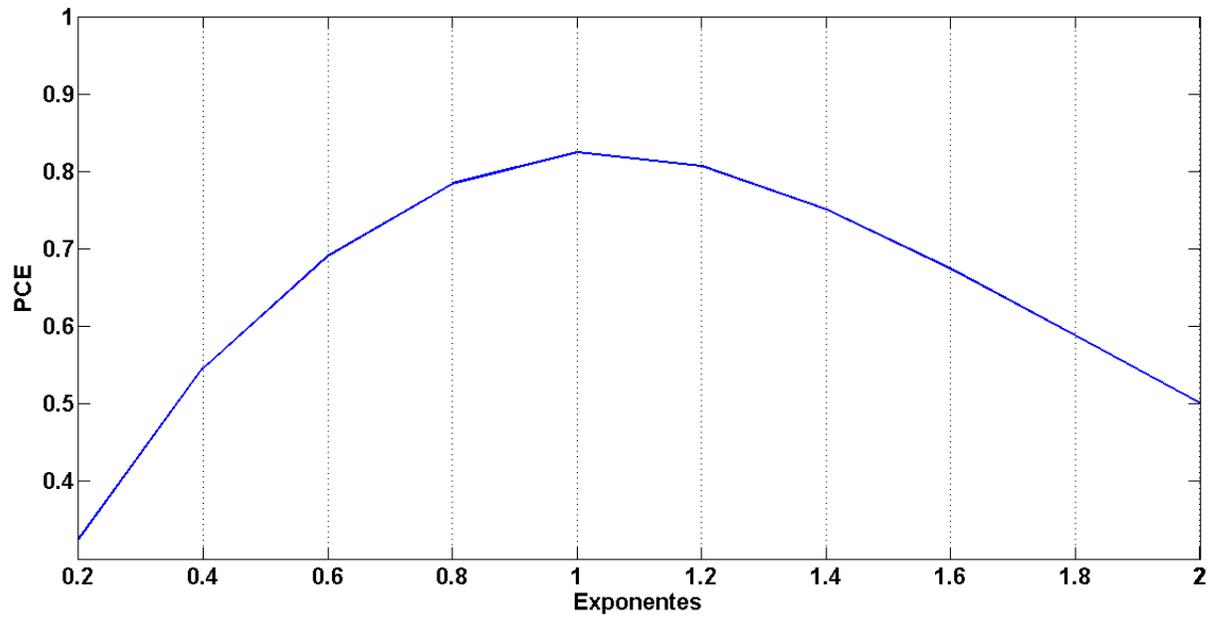


Figura 4.2: Métrica PCE para reconocimiento de imágenes con potencia en los módulos de la TF

Cuadro 4.1: Tabla de valores de la métrica PCE para cada autocorrelación realizada con potencias de los módulos de la TF

$ TF ^x$	PCE
0.2	0.3255
0.4	0.5464
0.6	0.6911
0.8	0.7841
1	0.8244
1.2	0.8071
1.4	0.7517
1.6	0.6751
1.8	0.5885
2	0.5005

4.2. Reconocimiento de imágenes

Para el desarrollo de esta tesis fue necesario realizar una elección de imágenes que presentaran características similares para evaluar el desempeño del algoritmo realizado. En la estancia de verano realizada en CICESE, se trabajó con una variedad de imágenes, entre

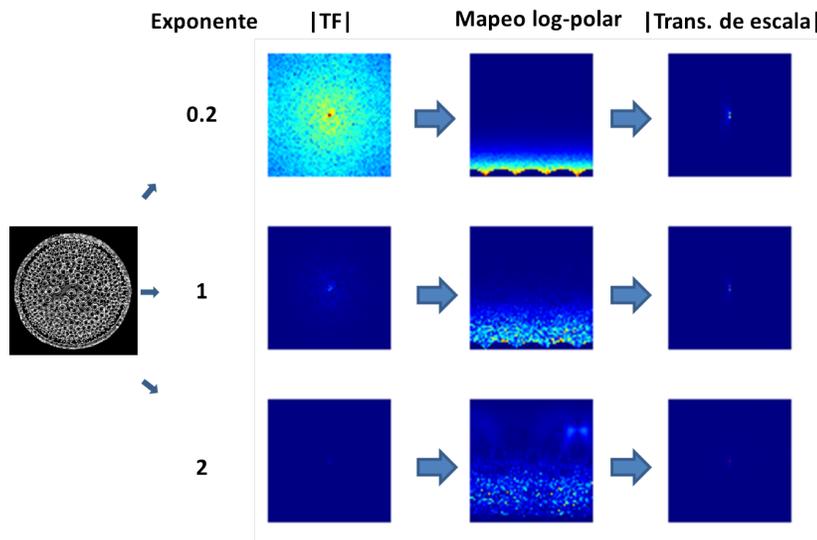


Figura 4.3: Espectros en diferentes partes del proceso de reconocimiento de imágenes para diferentes potencias del módulo de la TF

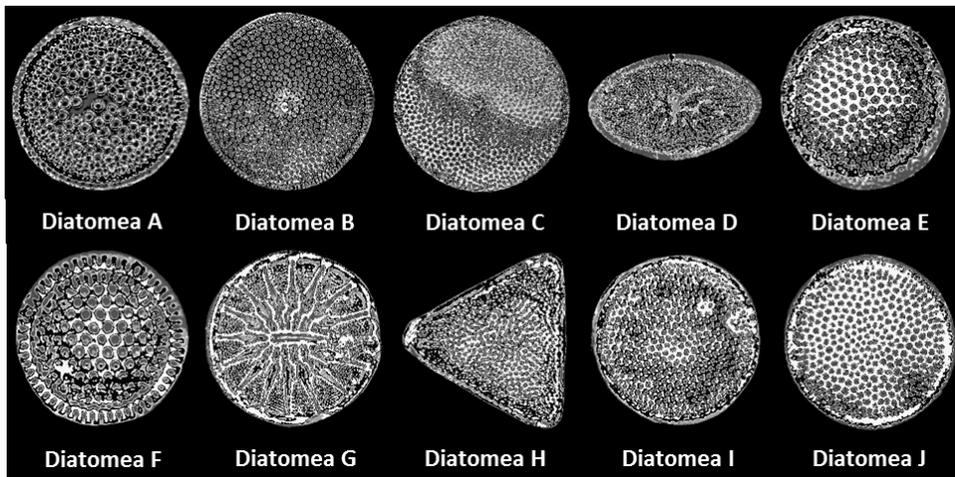


Figura 4.4: Selección de imágenes de diatomeas a reconocer mediante el algoritmo

ellas imágenes de distintas tipografías, imágenes de la vida cotidiana y de organismos microscópicos, siendo estos últimos los que más cautivaron mi interés dadas las aplicaciones médicas o científicas que pueden tener. Es así que se decidió trabajar los algoritmos desarrollados en esta tesis con imágenes microscópicas de diatomeas (algas unicelulares microscópicas), las cuales se muestran en la imagen de la figura 4.4, que para mayor comodidad fueron nombradas con letras del abecedario, así es posible referirnos a cada una de ellas de forma más sencilla durante las distintas comparaciones.

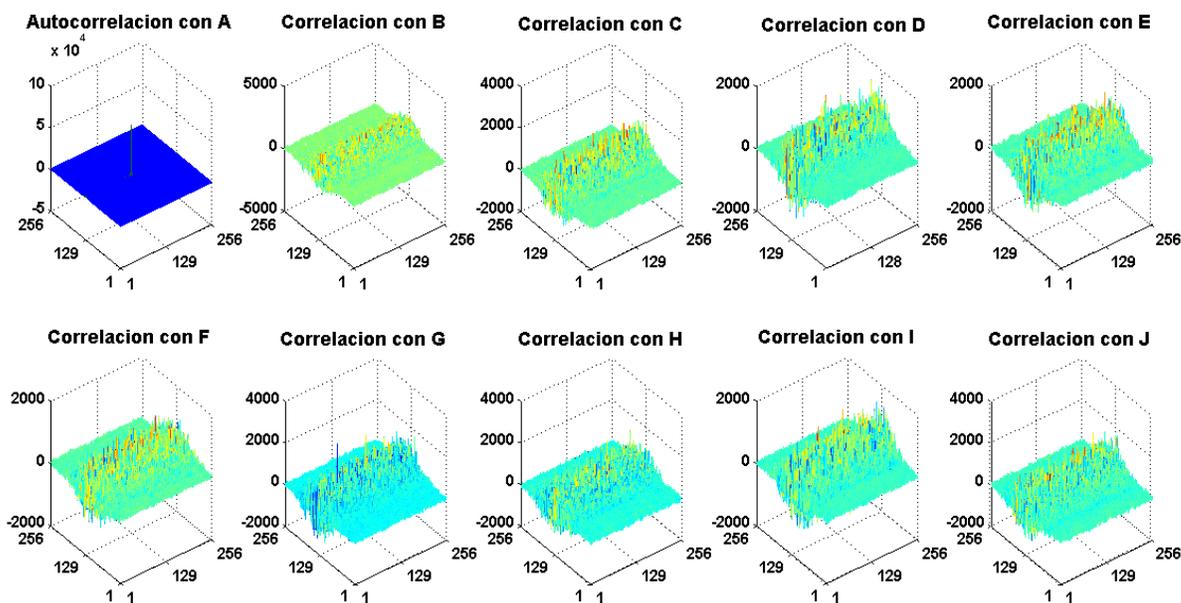


Figura 4.5: Espectro de correlación de la primer diatomea con ella misma y los nueve restantes

El algoritmo de reconocimiento de imágenes que se aplicó en esta tesis realiza sus comparaciones mediante correlaciones de espectros que pasaron por todo un proceso para agregar las distintas características que requerimos, tales como invariancia a rotación, escala y posición, por lo tanto, al momento de comparar nuestra imagen de la diatomea A, la cual llamaremos *imagen filtro* con el demás conjunto de diatomeas a las cuales llamaremos *imágenes problema*, se espera que al momento de autocorrelacionarse, el plano de correlación de salida nos arroje valores mucho más significativos que al momento de correlacionar el espectro de la imagen filtro con las demás imágenes. Es así que presento la figura 4.5 en la cual se demuestra lo antes mencionado, como se puede apreciar, la autocorrelación genera un pico muy definido en una escala de 1×10^5 , mientras que en el plano resultante de las correlaciones del espectro de la imagen de la diatomea A con el espectro de las demás imágenes de diatomeas encontramos una gran cantidad de valores en escala muy pequeña de 5×10^3 , lo cual se considera despreciable comparado con el valor de autocorrelación.

Todas las comparaciones que se hicieron en la imagen de la figura 4.5 fueron hechas con las imágenes originales, es decir, no se afectó su escala ni su rotación. En la vida diaria si tratamos de tomar fotografías microscópicas de distintas clases de diatomeas

para su posterior reconocimiento de clase mediante el presente algoritmo, nos resultaría imposible ubicar a cada diatomea con la misma escala y el mismo nivel de rotación para que aparezca igual a las imágenes originales que arrojaron los resultados ya mencionados. La mínima rotación que pueda tener una imagen o el mínimo escalamiento que presente, puede modificar de gran manera el espectro e imposibilitar el reconocimiento de la imagen. Es por eso mismo que se implementó el conjunto de la TF, la transformada de escala y se hizo el cambio de coordenadas cartesianas a polares, de esta manera se intentó evitar tales problemas y los resultados se presentan en las siguientes secciones.

4.2.1. Imágenes escaladas

Para reducir el costo computacional elegimos imágenes de un tamaño de 256×256 pixeles en formato .bmp (Bits Maps Protocole), las cuales escalamos con un valor inicial en 0.8 con saltos de 0.05 hasta la escala de 1.05, seleccionamos estos valores dado que ofrecen un rango de escala suficiente para probar nuestro algoritmo, con respecto al porqué solo se aumentó una escala al tamaño de la imagen original, esto se debe a que las imágenes de diatomeas estaban muy cercanos a los límites de 256 pixeles en su ancho y en su largo, por lo tanto una ampliación más no permitiría tener toda la imagen de la diatomea dentro del área que se trabajaría.

Para probar la efectividad del reconocimiento de imágenes escaladas, la imagen filtro se tomó tal cual se encontraba originalmente y se comparó con las 10 diatomeas, incluyéndose ella misma en las escalas ya mencionadas. Cabe recordar que la imagen filtro pertenece a la diatomea A. La figura 4.6 nos muestra el valor máximo del pico de correlación de salida para cada una de las diatomeas en las distintas escalas, como podemos ver, en todas las escalas tratadas se presenta un pico de correlación más pronunciado en la posición de la diatomea A, lo cual corrobora lo que esperábamos, en efecto el reconocimiento se logró a pesar de escalar las imágenes.

El cuadro 4.2 muestra los valores graficados en la figura 4.6, notamos que para el caso de imágenes escaladas de la diatomea A, los valores de correlación máximos son significativamente más altos que para el resto de las diatomeas escaladas.

Sin embargo, nada está dicho aún hasta que la métrica nos indique que estamos en lo correcto, en la figura 4.7 y en el cuadro 4.3 se muestran los valores obtenidos de la métrica PCE para cada una de las diatomeas escaladas, los resultados lucen muy bien para la escala original y la mayor en 1.05 en el reconocimiento de la diatomea A, para los

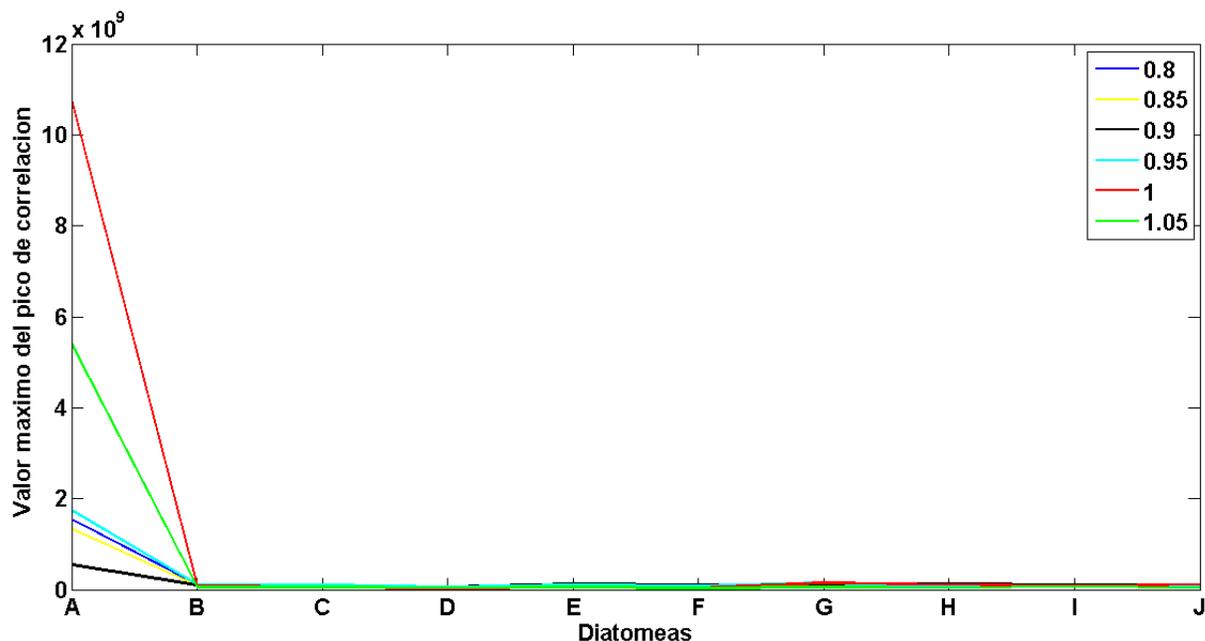


Figura 4.6: Valor máximo de correlación de imágenes de diatomeas escaladas

Cuadro 4.2: Tabla de valores de pico máximo de correlación para imágenes escaladas. Datos en la escala de 1×10^{10}

Escala	A	B	C	D	E	F	G	H	I	J
0.8	0.154	0.011	0.009	0.006	0.007	0.005	0.007	0.007	0.006	0.008
0.85	0.133	0.009	0.01	0.005	0.006	0.006	0.008	0.008	0.008	0.007
0.9	0.054	0.009	0.01	0.006	0.013	0.009	0.008	0.012	0.01	0.01
0.95	0.174	0.011	0.009	0.006	0.009	0.007	0.014	0.008	0.007	0.009
1	1.077	0.008	0.005	0.002	0.005	0.004	0.014	0.011	0.007	0.011
1.05	0.54	0.005	0.004	0.004	0.005	0.003	0.005	0.003	0.006	0.004

casos de las diatomeas en escalas menores, la métrica baja considerablemente aunque sus valores sobrepasan en gran cantidad los valores para las otras diatomeas en su respectiva escala.

4.2.2. Imágenes rotadas

Una imagen puede encontrarse rotada en una infinidad de posiciones angulares, por lo tanto, realizar el cálculo correspondiente para cada una de esas posiciones resultaría imposible. En el desarrollo de esta tesis se creyó conveniente trabajar con una rotación con

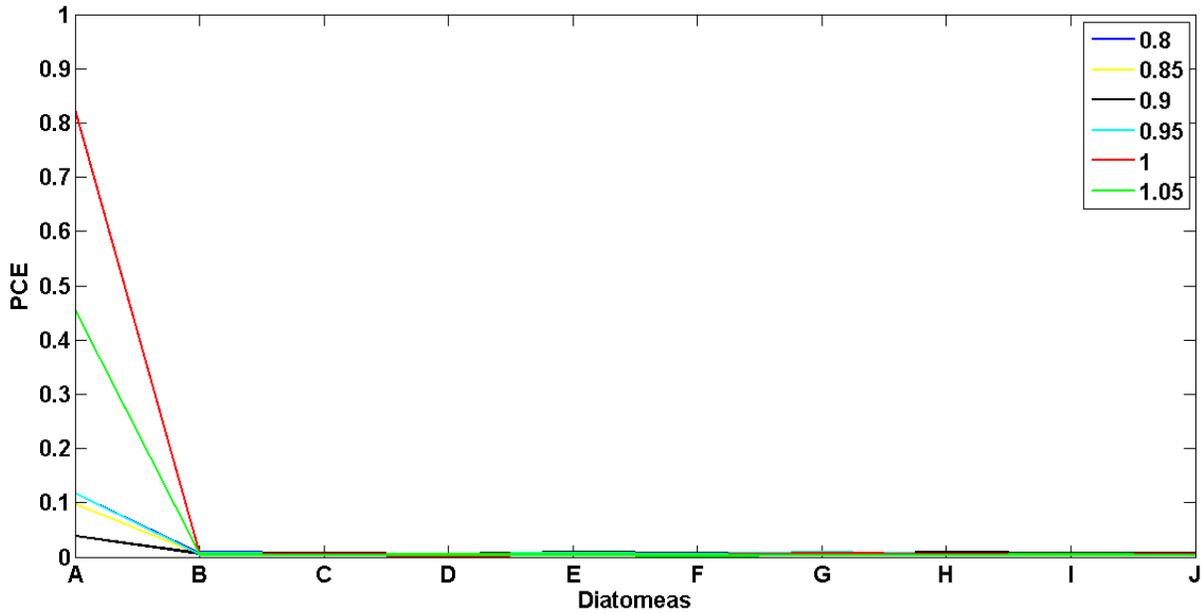


Figura 4.7: Métrica PCE para reconocimiento de imágenes de diatomeas escaladas

Cuadro 4.3: Tabla de valores de la métrica PCE para imágenes de diatomeas escaladas

Escala	A	B	C	D	E	F	G	H	I	J
0.8	0.117	0.008	0.006	0.005	0.005	0.004	0.005	0.005	0.004	0.006
0.85	0.097	0.006	0.006	0.004	0.004	0.004	0.005	0.005	0.005	0.004
0.9	0.038	0.006	0.006	0.005	0.009	0.006	0.005	0.008	0.006	0.006
0.95	0.117	0.006	0.005	0.004	0.006	0.004	0.008	0.004	0.004	0.005
1	0.824	0.005	0.004	0.002	0.003	0.003	0.006	0.005	0.005	0.005
1.05	0.455	0.003	0.003	0.004	0.004	0.002	0.003	0.003	0.004	0.003

incrementos de 20° a partir de la posición inicial la cual está posicionada en 0° hasta llegar a 180° , dados los efectos de simetría no se consideró necesario trabajar con posiciones angulares mayores.

En la imagen de la figura 4.8 y en el cuadro 4.4 se presentan la gráfica y los valores correspondientes para los picos máximos de salida en el plano de correlación. De esta imagen notamos un pico muy bien definido en las posiciones angulares de 0° y 180° , valores considerables en las posiciones de 20° y 160° y valores despreciables para los demás casos.

La métrica PCE corrobora lo mencionado, en la figura 4.9 y en el cuadro 4.5 se muestran la gráfica de la métrica y los valores graficados correspondientes para la comparación

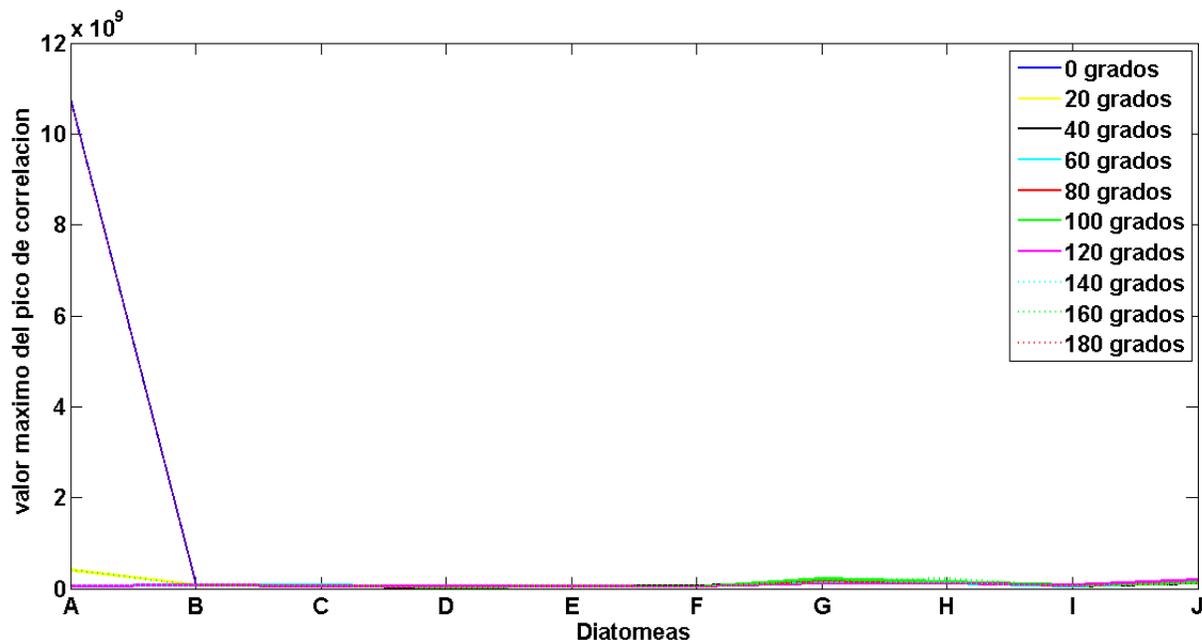


Figura 4.8: Valor máximo de correlación de imágenes de diatomeas rotadas

Cuadro 4.4: Tabla de valores de pico máximo de correlación para imágenes rotadas. Datos en la escala de 1×10^{10}

Grados	A	B	C	D	E	F	G	H	I	J
0	1.077	0.008	0.005	0.002	0.005	0.004	0.014	0.011	0.007	0.011
20	0.04	0.008	0.005	0.003	0.004	0.003	0.014	0.01	0.008	0.013
40	0.005	0.006	0.004	0.003	0.006	0.005	0.015	0.014	0.004	0.01
60	0.005	0.007	0.008	0.003	0.005	0.004	0.013	0.011	0.004	0.013
80	0.006	0.008	0.006	0.003	0.004	0.004	0.019	0.011	0.006	0.012
100	0.005	0.009	0.006	0.003	0.004	0.003	0.021	0.014	0.007	0.011
120	0.005	0.007	0.005	0.004	0.005	0.003	0.011	0.01	0.007	0.018
140	0.006	0.007	0.006	0.004	0.005	0.006	0.014	0.013	0.005	0.01
160	0.041	0.006	0.006	0.002	0.005	0.004	0.014	0.021	0.004	0.014
180	1.073	0.008	0.005	0.003	0.005	0.004	0.014	0.011	0.007	0.011

de las 10 diatomeas rotadas. Notamos valores aceptables para posiciones angulares en 0° , 20° , 160° y 180° , las demás posiciones presentan un valor en la métrica muy bajo, cercanos a 0 e inclusive superado por los valores para otras diatomeas dentro de la misma posición.

Es cierto que el algoritmo fue diseñado para trabajar con invarianza a rotación y por

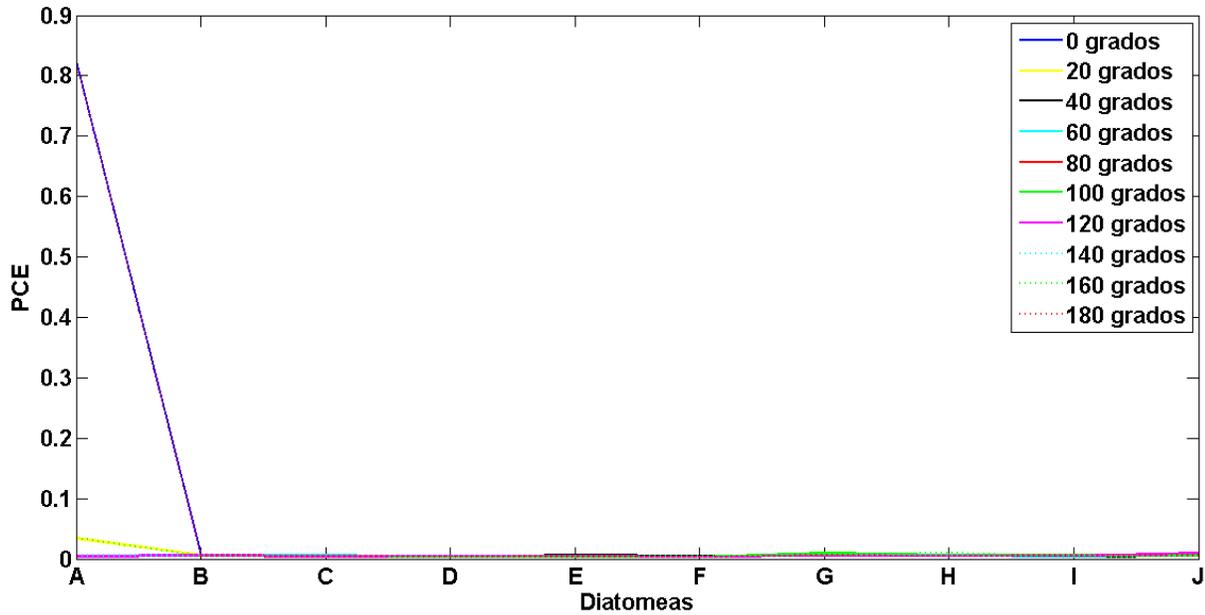


Figura 4.9: Métrica PCE para reconocimiento de imágenes de diatomeas rotadas

Cuadro 4.5: Tabla de valores de la métrica PCE para imágenes de diatomeas rotadas

Grados	A	B	C	D	E	F	G	H	I	J
0	0.824	0.005	0.004	0.002	0.003	0.003	0.006	0.005	0.005	0.005
20	0.033	0.006	0.004	0.003	0.003	0.003	0.006	0.005	0.005	0.006
40	0.004	0.004	0.003	0.003	0.005	0.003	0.006	0.007	0.003	0.005
60	0.004	0.005	0.006	0.003	0.004	0.003	0.005	0.005	0.003	0.006
80	0.005	0.006	0.004	0.003	0.003	0.003	0.008	0.005	0.004	0.005
100	0.004	0.006	0.004	0.003	0.003	0.003	0.009	0.006	0.005	0.005
120	0.004	0.005	0.004	0.004	0.004	0.003	0.005	0.005	0.005	0.009
140	0.005	0.005	0.004	0.004	0.004	0.005	0.006	0.006	0.004	0.005
160	0.034	0.004	0.004	0.002	0.004	0.003	0.006	0.01	0.002	0.007
180	0.821	0.005	0.004	0.003	0.003	0.003	0.006	0.005	0.005	0.005

lo tanto debería reconocer también las imágenes de diatomeas rotadas entre 40° y 140° , el problema se atribuye a que estamos trabajando con imágenes digitales, las cuales fueron rotadas digitalmente mediante una función de Matlab (*imrotate*), la cual genera distorsión en los contornos de las imágenes rotadas en forma de diente de sierra, lo cual se traduce en ruido en el espectro de salida, este ruido se vuelve tan determinante al grado de ser imposible el reconocimiento de la imagen. Podemos ver este efecto de diente de sierra en el contorno de la figura 4.10.a y en 4.10.b donde la imagen de la figura 4.10.c se rotó a

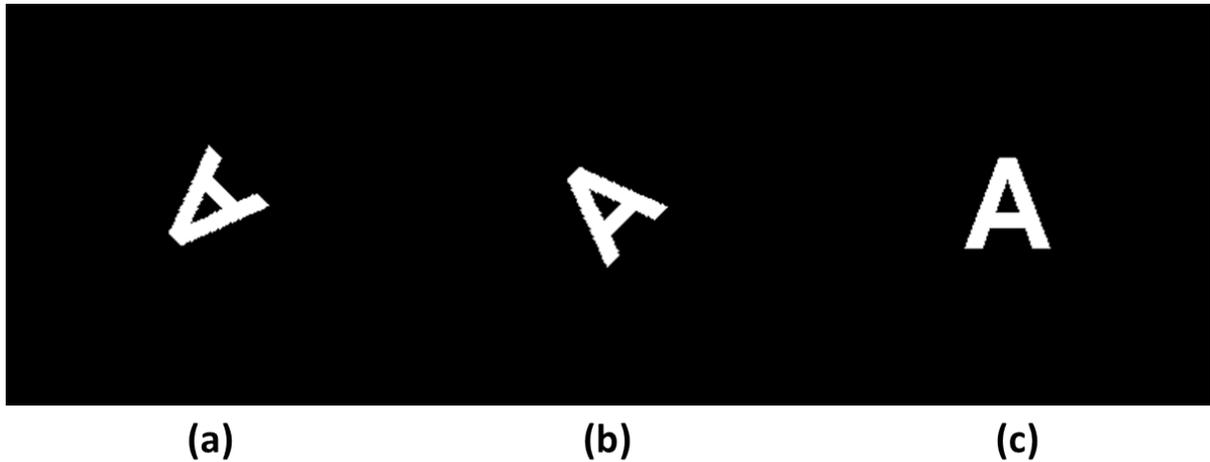


Figura 4.10: (a) Letra A rotada 135 grados. (b) Letra A rotada 45 grados. (c) Letra A sin rotar

135° y 45° correspondientemente.

4.3. Costo computacional

Siempre que se realiza algún tipo de algoritmo enfocado a un problema en específico, además de la efectividad con la cual este solucione el problema, el costo computacional es un factor importante a considerar. El costo computacional consiste en el tiempo en el cual el algoritmo operará hasta obtener los resultados requeridos, de esta manera es posible hacer un balance entre la eficiencia del algoritmo y el tiempo que tarda en procesar la información, en algunos casos donde no se requiera de una precisión muy exigente, mejorar el tiempo de operación sería más importante que en otro caso donde se requieran cálculos más precisos pese al tiempo de procesamiento.

Es de esta manera que consideramos importante recalcar que el costo computacional de este trabajo de tesis depende totalmente del número de imágenes que se requieran comparar. En promedio calculamos que la comparación de la imagen filtro con una imagen problema toma aproximadamente 5.1473 segundos, dadas las posibilidades y limitantes que el dispositivo de cómputo empleado pudo ofrecer.

Conclusiones

Se implementó un sistema digital invariante de reconocimiento de patrones en imágenes. El sistema fue probado mediante la identificación de diversos objetos y se obtuvieron resultados favorables evaluados por la métrica PCE para variaciones en rotación y escala utilizando imágenes de diatomeas.

Se utilizó el filtro no lineal (ley k) dadas las ventajas teóricas que muestra sobre los demás filtros, el desempeño fue considerablemente bueno.

Una de las desventajas de nuestro sistema digital de correlación no lineal invariante es que no hace un buen reconocimiento de imágenes que presentan ruido proveniente al efecto sierra además el tiempo de cómputo llega a sobrepasar los 5 segundos para la comparación de una sola imagen.

Con respecto al trabajo a futuro que puede realizarse de esta tesis, podría considerarse la implementación de un mejor algoritmo de discriminación de objetos en la presencia de diferentes tipos de ruido, entre ellos el efecto sierra podría aumentar la confiabilidad del algoritmo. La reducción del tiempo de cómputo variando el tipo de interpolación utilizada, por supuesto monitoreando la eficiencia con la métrica hasta hacer un buen balance en los resultados sería un gran paso en el desarrollo de este algoritmo.

Programas en Matlab

A continuación se presentan los programas utilizados en el desarrollo de este trabajo de tesis, es conveniente mencionar que para un mejor desempeño al realizar pruebas con estos algoritmos se trabaje con imágenes en blanco y negro en formato .bmp.

A.1. Programa que compara el módulo de la TF con diferentes exponentes

```
k = 0.3;
modulo = [0.2 1 2];
xx = 0;
yy = 3;
figure,
for jj = 1 : yy
a = imread(['NombreDeLaImagen.Formato'],'Formato');
a = imresize(a, [256 256], 'bilinear');
a = double(a);
a = a - mean2(a);
a_fft = abs(fftshift(fft2(fftshift((a)))));
a_fft = a_fft .^ modulo(jj);
xx = xx + 1;
subplot(3,3,xx)
imshow(a_fft, []);
```

```

afft = compspectrum(256). * afft;
afft = maplog2(mapolar(factorro(afft)));
xx = xx + 1;
subplot(3,3,xx)
imshow(afft, []);
afft = (fftshift(fft2(fftshift(afft))));
xx = xx + 1;
subplot(3,3,xx)
imshow(abs(afft), []);
end

```

A.2. Programa que compara imágenes escaladas y rotadas

```

k = 0.3;
Escala = [80 85 90 95 100 105];
Rotacion = [0 20 40 60 80 100 120 140 160 180];
xx = 10;
yy = 10;
for jj = 5 : yy
a = imread(['NombreDeLaImagen.Formato'],'Formato');
afft = Trans_Escala(a, k);
for ii = 1 : xx;
%Escaladas
for hh = 1 : 6
b = imread(['NombreDeLaImagen.Formato'],'Formato'); %imagenproblema
bfft = Trans_Escala(b, k);
fc = bfft * conj(afft);
fcf = real(fftshift(ifft2(fftshift(fc))));
[PCE_esc(ii, hh), c_max_esc(ii, hh)] = Metrica_PCE(fcf);
end
%Rotadas

```

```

for hh = 1 : 10
b = imread(['NombreDeLaImagen.Formato'],'Formato'); %imagenproblema
bfft = Trans_Escala(b, k);
fc = bfft.*conj(afft);
fcf = real(fftshift(fft2(fftshift(fc))));
[PCE_rot(ii, hh), c_max_rot(ii, hh)] = Metrica_PCE(fcf);
end
end
end

```

A.3. Funciones

A.3.1. Función compspectrum

```

function OUT = compspectrum(rows)
r = (rows - 1)/2;
f = 2 * pi/r;
[x y] = meshgrid(-r : r);
OUT = f.*(x.*x + y.*y);

```

A.3.2. Función factorro

```

function y = factorro(f)
[m, n] = size(f);
centro = m/2;
for i = 1 : m
for j = 1 : m
r(i, j) = sqrt((sqrt((i - centro).^2 + (j - centro).^2))./(sqrt(2). * centro));
end
end
r = r + 1;
y = r.*f;

```

A.3.3. Función mapolar

```

function g = mapolar(f)
[m, n] = size(f);
for i = 1 : m
teta(i) = (2 * pi) * (i - 1)/(m - 1);
r(i) = m * (i - 1)/(sqrt(2) * (m - 1));
ind(i) = i;
end
for i = 1 : m
for j = 1 : m
x(i, j) = m/2 + r(i) .* cos(teta(j));
y(i, j) = m/2 - r(i) .* sin(teta(j));
if (x(i, j) > m || y(i, j) > m | x(i, j) < 1 | y(i, j) < 1)
g(i, j) = 0;
else
g(i, j) = f(round(y(i, j)), round(x(i, j)));
end
end
end
for i = 1 : 60
for j = 1 : 256 g(i, j) = interp2(ind', ind, f, x(i, j), y(i, j), 'linear');
end
end

```

A.3.4. Función maplog2

```

function y = maplog2(x);
[m, n] = size(x);
for i = 1 : m
ind(i) = i;

```

```
orig(i) = exp((i - 1) * ((log(m))./(m - 1)));
end
y = interp2(ind', ind, x, ind', orig);
```

A.3.5. Función Trans_Escala

```
function afft = Trans_Escala(a, k)
a = imresize(a, [256 256], 'bilinear');
a = double(a);
a = a - mean2(a);
afft = abs(fftshift(fft2(fftshift((a)))));
afft = compspectrum(256) * afft;
afft = maplog2(mapolar(factorro(afft)));
afft = (fftshift(fft2(fftshift(afft))));
afft = imresize(afft, [256 256], 'bilinear');
ffa = angle(afft);
mods = abs(afft) .^ k;
afft = mods .* exp(i * ffa);
```

A.3.6. Función Metrica_PCE

```
function [PCE, numerador] = Metrica_PCE(gcf)
cn2 = gcf;
cn2 = max(max(cn2));
numerador = abs(cn2) .^ 2;
suma = 0.0;
for ii = 1 : 256
for jj = 1 : 256
suma = suma + abs(gcf(ii, jj)) .^ 2;
end
end
deno = suma;
PCE = numerador / deno;
```


Bibliografía

- [1] Kamen, Edward W. y Bonnie S. Heck, *Fundamentos de señales y sistemas usando la Web y MATLAB*. Pearson Prentice-Hall, 2007.
- [2] Goodman, Joseph W., *Introduction to Fourier Optics*. Roberts and Company Publishers, 2005.
- [3] John G. Proakis y Dimitris G. manolakis, The discrete Fourier Transform: Its properties and Applications, *Digital Signal Processing*. Pearson Prentice-Hall, pp. 394–447, 1996.
- [4] Jae S. Lim, *Two-dimensional signal and image processing*. Prentice-Hall, 1990.
- [5] De Sena Antonio y Davide Rocchesso, A Fast Mellin and Scale Transform. Hindawi Publishing Corporation, *EURASIP Journal on Advances in Signal Processing* Article ID 89170, 9 paginas 2007.
- [6] De Sena Antonio y Davide Rocchesso, A Fast Mellin Transform with Applications in DAFX. Proc. of the 7th int, conference on Digital Audio Effects (DAFx), 2004.
- [7] Coronel Beltrán A., Reconocimiento de patrones no lineal invariante a posición, rotación, escala y ruido de imágenes digitales. Tesis Doctoral. 2010.
- [8] Coronel-Beltrán A., Álvarez-Borrego, Josué, Reconocimiento invariante de patrones con filtros no lineales. LI Congreso Nacional de Física SMF, XXI Reunión Anual de Óptica, Zacatecas, Zacatecas. 2008.
- [9] Javidi, B. and Horner, J. L., *Real-Time Optical Information Processing*. Academic Press, 1994.
- [10] Padilla Ramírez Ariel, Desarrollo de una aplicación de correlación digital que haga uso de filtros lineales y no lineales compuestos. Tesis de Maestria, 2009.
- [11] Michael Weeks, The Discrete Fourier Transform, *Digital Signal Processing using MATLAB and Wavelets*. Infinity Science press LLC pp. 191–196, 2007.