



UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MÉXICO

---

---

CENTRO UNIVERSITARIO UAEM TEXCOCO

**“ANÁLISIS COMPARATIVO DE TÉCNICAS DE  
RECONOCIMIENTO FACIAL”**

**T E S I S**

**QUE PARA OBTENER EL TÍTULO DE  
INGENIERO EN COMPUTACIÓN**

**PRESENTA  
EMMANUEL CALDERÓN LÓPEZ**

**DIRECTOR  
DR. EN C. JAIR CERVANTES CANALES**

**REVISORES  
DR. EN C. FARID GARCIA LAMONT  
DR. EN C. C. JOSE SERGIO RUIZ CASTILLA**

**TEXCOCO, ESTADO DE MÉXICO, MAYO DE 2016.**

M. EN C. E. VIRIDIANA BANDA ARZATE  
SUBDIRECTORA ACADEMICA DEL  
-CENTRO UNIVERSITARIO UAEM TEXCOCO.

AT'N L. EN D. MARCO RODRIGO LÓPEZ GONZÁLEZ  
RESPONSABLE DEL DEPARTAMENTO DE TITULACIÓN.

Con base en las revisiones efectuadas al trabajo escrito titulado "Análisis comparativo de Técnicas de Reconocimiento Facial" que para obtener el título de Ingeniero en Computación presenta el sustentante Calderón López Emmanuel, con número de cuenta 0921029 respectivamente, se concluye que cumple con los requisitos teórico-metodológicos por lo que se le otorga el voto de aprobación para su trabajo sustentación, pudiendo continuar con la etapa de digitalización del trabajo escrito.

ATENTAMENTE

  
Farid Barca Lament

FIRMA

NOMBRE DEL REVISOR



FIRMA

NOMBRE DEL REVISOR

  
Jair Cervantes Canales

NOMBRE Y FIRMA DEL  
DIRECTOR

c.c.p. Calderón López Emmanuel  
c.c.p. Director Dr . En. C. Jair Cervantes Canales.  
c.c.p. Titulación L en D. MARCO RODRIGO LÓPEZ GONZÁLEZ



## AGRADECIMIENTOS

Primero y más importante, me gustaría agradecer sinceramente a mi asesor de Tesis, Dr. C. Jair Cervantes Canales, por su tiempo, su esfuerzo y su dedicación, en compartir su conocimiento. Su conocimiento, su orientación, su persistencia, su paciencia y su apoyo han sido fundamentales para realizar este trabajo.

A su manera, ha sido capaz de ganarse mi lealtad y admiración, así como sentirme en deuda con él por todo lo recibido durante el periodo de tiempo que ha durado este trabajo de Tesis.

A mis padres, por haberme proporcionado la mejor educación y sus grandes lecciones de vida.

En especial a mi madre Rosa María López Castañeda que a pesar de las circunstancias siempre nos enseñó a luchar por nuestros sueños, sin importar los obstáculos de la vida y por enseñarme a siempre confiar en mis decisiones.

En especial a mi padre Rafael Calderón Rodríguez que me enseñó que nada puede detenernos para lograr a cumplir nuestros sueños y el enseñarme a ver la vida de manera diferente.

A mi hermano Rafael Calderón López porque siempre me apoyo a seguir estudiando y a echarle ganas.

A todos mis maestros quienes con gran paciencia nos transmitieron su conocimiento.

## DEDICATORIA

A mi madre que fue un gran ejemplo de siempre salir adelante sin importar la  
circunstancia y de nunca darse por vencido ante las dificultades de la vida.  
A mi padre que me enseñó que perseverar significa alcanzar las meta no importando los  
retos que te ponga la vida durante el camino.  
A mi hermano quien siempre me apoyo para seguir adelante.  
A mi familia que siempre estuvo hay para apoyarme y alentarme.  
A mis maestros quienes con paciencia, me transmitieron su conocimiento.  
EMMANUEL calderón López

# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Planteamiento del Problema . . . . .	3
1.2. Justificación . . . . .	3
1.3. hipótesis . . . . .	4
1.4. Objetivos y Metas . . . . .	4
1.4.1. Objetivo General . . . . .	4
1.4.2. Objetivos Específicos . . . . .	4
1.4.3. Metodología . . . . .	4
1.5. Estado del Arte . . . . .	4
<b>2. Preliminares</b>	<b>8</b>
2.1. Imágenes . . . . .	8
2.2. Técnicas de pre-procesamiento . . . . .	9
2.2.1. Operaciones Lógicas, Estadísticas y Aritméticas . . . . .	26
2.2.2. Operaciones Aritméticas . . . . .	30
2.2.3. Operaciones Trigonométricas . . . . .	32
2.2.4. Operaciones Estadísticas . . . . .	34
2.2.5. Operadores Grupales . . . . .	37
<b>3. Sistemas de reconocimiento facial</b>	<b>50</b>
3.1. Análisis de Componentes Principales (PCA) . . . . .	50
3.1.1. Clasificación de caras Mediante Análisis de Componentes Principales . . . . .	52
3.2. Análisis Discriminante Lineal (LDA) . . . . .	55

3.2.1.	Clasificación de caras Mediante Análisis de Discriminante Lineal . . . . .	57
3.3.	Patrón Biario Local (LBP) . . . . .	59
3.3.1.	Clasificación de caras Mediante el algoritmo LBP . . . . .	60
<b>4.</b>	<b>Metodología</b>	<b>62</b>
4.1.	Problemas al trabajar con un Sistema de Reconocimiento Facial . . . . .	62
4.2.	Etapas de un Sistema Biométrico Facial . . . . .	64
4.2.1.	Obtención de la Imagen . . . . .	65
4.2.2.	Detección de Rostro . . . . .	66
4.2.3.	Pre-Procesamiento . . . . .	67
4.2.4.	Aplicación de Algoritmos . . . . .	68
<b>5.</b>	<b>Resultados experimentales</b>	<b>72</b>
5.1.	Conjuntos de datos y pre-procesamiento . . . . .	72
5.1.1.	Base de imágenes FEI . . . . .	72
5.1.2.	Base de imágenes Yale . . . . .	73
5.1.3.	Base de imágenes IMA . . . . .	74
5.2.	Resultados . . . . .	76
<b>6.</b>	<b>Conclusiones</b>	<b>83</b>
<b>A.</b>	<b>Guia de Usuario</b>	<b>92</b>

# Índice de figuras

2-1. Figura. Operación de Cuantificación . . . . .	11
2-2. Figura. Los operadores puntuales se aplican al pixel de la misma coordenada que tendra de salida. . . . .	11
2-3. Figura. A cada valor de X le corresponde el mismo valor de Y. . . . .	13
2-4. Figura. Función Escalon. . . . .	14
2-5. Figura. Función Escalon aplicada a una imagen. . . . .	14
2-6. Figura. Función Escalonada. . . . .	15
2-7. Figura. Función Escalonada aplicada a una imagen. . . . .	16
2-8. Figura. Función Incremento de Intensidades. . . . .	16
2-9. Figura. Función Incremento de Intensidad aplicada a una imagen. . . . .	17
2-10. Figura. Función de Reducción de Intensidad. . . . .	18
2-11. Figura. Función Reducción de Intesidad. . . . .	18
2-12. Figura. Función Dientes de Sierra. . . . .	19
2-13. Figura. Función Escalon dentro de Identidad. . . . .	20
2-14. Figura. Función Escalon dentro de Identidad aplicada a una imagen. . . . .	21
2-15. Figura. Función Expansión de Contraste. . . . .	21
2-16. Figura. Función Expansión de Contraste aplicada a una imagen. . . . .	22
2-17. Figura. Función Parabola. . . . .	23
2-18. Figura. Función Parabola aplicada a una imagen. . . . .	23
2-19. Figura. Función de Parabola Inversa aplicada a una imagen. . . . .	24
2-20. Figura. Función de Polarizado. . . . .	25
2-21. Figura. Función Polarizada aplicada a una imagen. . . . .	25

2-22. Figura. Función Expansión aplicada a una imagen. . . . .	27
2-23. Figura. Función OR aplicada a dos imagenes. . . . .	28
2-24. Figura. Función XOR aplicada a dos imagenes. . . . .	29
2-25. Figura. Función NOT aplicada a una imagen. . . . .	30
2-26. Figura. Función Trigonometrica Seno aplicada a una imange. . . . .	33
2-27. Figura. Funcipon Trigonometrica Coseno aplicada a una imagen. . . . .	33
2-28. Figura. Función Trigonometrica Tangente aplicada a una imagen. . . . .	34
2-29. Función. Se aplica operaciones al conjunto de pixeles que lo rodean para dar un nuevo valor a un pixel. . . . .	38
4-1. Etapas del Reconocimiento Facial . . . . .	64
4-2. En esta imagen se puede apreciar, los pasos a seguir para obtener la parte del rostro, del resto de la imagen. . . . .	66
4-3. Ejemplo de pre-procesamiento aplicado a una imagen. . . . .	67
5-1. Esta imagen de entrada a la cual se detectara el rostro. . . . .	75
5-2. Al aplicar el algoritmo para detectar el rostro, este selecciona el contorno del rostro, el cual esta en un recuadro de color azul. . . . .	75
5-3. En esta imagen, se aplico el detector de rostro y se recorto solo el rostro, con un tamaño de 300 x 300 pixeles . . . . .	75
6-1. Figura. Imagen ejemplo con poca iluminacion que afecto el desempeño de los algoritmos. . . . .	84
6-2. Figura. Imgaen ejemplo con poca iluminacion, que afecta el desempeño de los algoritmos. . . . .	84
A-1. Se carga la ruta donde tenemos la interfaz. . . . .	93
A-2. Se escribe el nombre de la interfaz, sin la extensión, para ejecutar. . . . .	94
A-3. Se carga la interfaz, para poder realizar las pruebas con los algoritmos que tiene. . . . .	95
A-4. Ejemplo para cargar una imagen. . . . .	96
A-5. Imagen cargada para poder trabajar con ella. . . . .	96



A-6. Ejemplo, aplicando uno de los 3 algoritmos con los que podemos aplicar en la  
intefaz. . . . . 97

# Índice de cuadros

2-1. Operación AND . . . . .	27
2-2. Operación OR . . . . .	28
2-3. Operación XOR . . . . .	28
2-4. Operación SUB . . . . .	29
2-5. Operación NOT . . . . .	30
5-1. Matriz de confusión utilizando el algoritmo LDA . . . . .	76
5-2. Matriz de confusión utilizando el algoritmo PCA . . . . .	77
5-3. Matriz de confusión utilizando el algoritmo LBP . . . . .	77
5-4. Matriz de confusión utilizando el algoritmo LDA . . . . .	78
5-5. Matriz de confusión utilizando el algoritmo PCA . . . . .	79
5-6. Matriz de confusión utilizando el algoritmo LBP . . . . .	79
5-7. Matriz de confusión utilizando el algoritmo LDA . . . . .	80
5-8. Matriz de confusión utilizando el algoritmo PCA . . . . .	81
5-9. Matriz de confusión utilizando el algoritmo LBP . . . . .	82

# **Análisis Comparativo de Técnicas de Reconocimiento Facial**

by

Calderón López Emmanuel

## **Abstract**

Facial recognition has been a studied in recent years for many researchers in several areas. The most important techniques used in face recognition can use; Principal Component Analysis (PCA), Local Binary Patterns (LBP) or Linear Discriminant Analysis (LDA). These are the techniques with the best performance in the literature. However, in the actual literature, there isn't a comparison study of these techniques. In this Thesis are comparing these techniques under specific conditions of the input images.

# Análisis Comparativo de Técnicas de Reconocimiento Facial

por

Calderón López Emmanuel

## Resumen

Los métodos de reconocimiento facial han sido estudiados en años recientes por muchos investigadores en diferentes áreas. Las técnicas más importantes en reconocimiento facial pueden utilizar: Análisis de Componentes Principales (PCA), Patrones Binarios Locales (LBP) y Análisis Discriminante Lineal (LDA). Estas son las técnicas con mejores desempeños en la literatura. Sin embargo, en la literatura actual no existe un estudio comparativo de estas técnicas. En esta tesis son comparadas estas técnicas bajo condiciones específicas en las imágenes de entrada

# Capítulo 1

## Introducción

Los seres humanos han aprendido a reconocer a otras personas por medio de sus rostros, su forma de caminar, forma de vestir e incluso por sus gestos. Como humanos hemos aprendido a reconocer a otros seres humanos inconscientemente, esto se debe a que aún en los primeros años de infancia, durante nuestro desarrollo, aprendemos a diferenciar y reconocer a las personas. Enseñarle a una computadora que realice el reconocimiento facial, ha llevado al hombre al estudio de diferentes métodos y técnicas que estudian y miden los diferentes puntos y características del ser humano, todo ello con el objetivo de determinar la identidad de una persona.

La biometría se define como el conjunto de técnicas que estudian las características físicas de una persona, por la gran variedad de técnicas existentes, éstas se pueden categorizar en dos grandes grupos, Biométrica estática y Biométrica dinámica. La primera es el estudio de características de: rostro (ojos, nariz, boca), la iris, la retina, huellas dactilares, la palma de la mano. La segunda se encarga del estudio de formas de expresión, gestos, e incluso la forma de caminar.

Dentro de la biometría existen muchas técnicas que se encargan de realizar el proceso de identificación, utilizando algoritmos matemáticos y estadísticos sobre la imagen del rostro de una persona. Enseñar a una computadora a implementar los algoritmos de biometría sin la intervención de un experto, ha llevado al desarrollo de diversas técnicas y métodos para automatizar estos procesos dentro de una computadora. Existen diversas técnicas para el reconocimiento

facial, muchas han tenido un gran avance y destacan sobre otras técnicas en puntos importantes como: tiempo de procesamiento, recursos ocupados para llevar a cabo el procesamiento y exactitud de resultados.

La biométrica es una de las herramientas más utilizadas en el área de seguridad, en empresas, organizaciones, sectores públicos, sectores privados, aeropuertos, entre muchas más.

El objetivo de la biométrica dentro de estas áreas, es el control de acceso a organizaciones, sectores, e incluso recursos, evitar fraudes, detectar terroristas en aeropuertos, en agencias policíacas y militares, con motivos de seguridad.

La utilización de la biométrica es utilizada desde 1882, en Francia por un policía de nombre Alphonse Bertillon quien desarrolló el primer sistema de identificación de las personas, que se basaba en rasgos físicos como nariz, altura y partes del cuerpo de la persona, este sistema funcionaba tomando las medidas que existían entre un componente u otro, el sistema lo probó con diferentes personas para así identificar a los criminales.

Tiempo después en el año de 1892, un jefe de policía de nombre Juan Vucetich, descubrió la técnica de las huellas dactilares, gracias a un caso en el que una mujer mató a sus dos hijas, y fue culpado el criado, Juan Vucetich descubrió que la persona culpable fue la madre y no el criado. Utilizando las huellas de sangre dejadas en el buzón, así fue como se empezó a trabajar con huellas dactilares, aunque fue hasta después de un año que esta técnica fue aceptada, ya que se dieron cuenta de que dos personas no pueden tener la misma huella dactilar.

El arte de la biometría fue utilizado por los forenses, detectando huellas y comparándolas con los sospechosos de los que ya se habían tomado sus huellas, lo que provocó un gran avance en el campo científico.

Conforme fue pasando el tiempo se vieron en la necesidad de separar las huellas en clases, esto fue llevado a la práctica por Francis Galton en 1900, que fue empleado por Edward Henry en 1992. Los sistemas biométricos obtienen las imágenes de cámaras de vigilancia, obteniendo una serie de imágenes para su procesamiento que identifican a un individuo por medio de características únicas. Al extraer las características de una persona se comparan con las características de las personas que se encuentran dentro de la base de datos, para así poder identificar al individuo, o en algunos casos agregarlos al sistema.

Tenemos muchos problemas al realizar el reconocimiento facial debido a las variaciones,

que se dividen en dos categorías; variaciones agresivas y no agresivas. Las variaciones agresivas representan un cambio notable en un corto periodo de tiempo, como lo son cirugías, el cambio de color del pelo o cicatrices. Por otro lado tenemos las variaciones no agresivas representan un cambio, el cual no es notable en un periodo corto, si no lleva tiempo en notarse, como el envejecimiento. Al realizar el reconocimiento facial se pueden trabajar con imágenes que tienen condiciones específicas, para facilitar su procesamiento y sus resultados, como puede ser el fondo blando.

En esta tesis, se implementan tres técnicas de reconocimiento facial utilizando Análisis de Componentes Principales (PCA, por sus siglas en inglés *Principal Component Analysis*), Patrón Binario Local (LBP, por sus siglas en inglés *Local Binary Pattern*) y Análisis Discriminante Lineal (LDA por sus siglas en inglés *Linear Discriminant Analysis*). Se realiza un análisis comparativo de los resultados y se discuten los resultados.

## **1.1. Planteamiento del Problema**

Dentro del estado del arte actual, existen muchas técnicas de Reconocimiento Facial, unas más eficientes que otras, con el paso del tiempo surgen más técnicas e incluso el mejoramiento de unas. Cada técnica tiene sus características como son: tiempo de procesamiento, recursos necesarios para llevar a cabo una función y precisión de resultados. Además cada técnica tiene sus ventajas y desventajas, así como el área de aplicación.

En la literatura actual no existe un estudio que compare los métodos de reconocimiento facial más conocidos bajo distintas condiciones de trabajo. En esta tesis se pretende hacer un análisis del comportamiento de los tres algoritmos bajo diferentes condiciones de trabajo.

## **1.2. Justificación**

Los resultados obtenidos permitirán una mejor selección de un algoritmo de reconocimiento facial bajo condiciones más específicas, mejorando el desempeño al utilizar con mayor precisión un algoritmo de reconocimiento facial.

### **1.3. hipótesis**

¿Es posible seleccionar con mejor exactitud un método de reconocimiento facial a partir de las condiciones de la imagen?

### **1.4. Objetivos y Metas**

#### **1.4.1. Objetivo General**

Realizar un análisis experimental para obtener las técnicas de reconocimiento facial con mejor desempeño bajo diferentes condiciones de trabajo.

#### **1.4.2. Objetivos Específicos**

1. Obtener las diferentes imágenes de diferentes individuos
2. Realizar un análisis de las mejores técnicas de reconocimiento facial.
3. Obtener los resultados y la mejor técnica que se acople a las condiciones.

#### **1.4.3. Metodología**

1. Analizar las técnicas a trabajar.
2. Obtener las imágenes con las que se van a trabajar.
3. Programar las técnicas.
4. Ejecutar las técnicas.
5. Comparar las técnicas.
6. Obtener los resultados y conclusiones.

### **1.5. Estado del Arte**

#### **Origen de Sistemas Biométricos Faciales**



En los años 60's [5] Bledsoe diseñó un sistema hombre-máquina para clasificar rostros, en base a las relaciones de ángulos, longitudes y proporciones de las fotografías. Unos años más tarde, Sakai [2] realizó un programa que se basaba en plantillas en base a rasgos en boca, la nariz y ojos. Kelly propone un método para la detección de rostros en fotografías de cuerpo completo[14]. La técnica se ha utilizado en diferentes algoritmos, incluso hoy en día. Kanade propuso el primer sistema automático de reconocimiento facial [13], sin embargo, éste obtenía una baja tasa de reconocimiento facial, en pequeñas base de datos.

## **Obtención de la Imagen**

Truk y compañía describen algunos problemas al obtener la imagen correcta para reconocimiento, uno de estos problemas se refiere a la dificultad de tomar una imagen en tiempo real, en su artículo recomiendan que el momento perfecto para tomar una imagen del rostro de una persona es cuando la persona este mirando hacia la cámara y con movimiento lento, para así obtener la imagen sin tantas imperfecciones [26]. Sung y Poggio presentaron un modelo ejemplo basado en aprendizaje, que se encargaba de trabajar en rostros con una vista frontal y vertical [20]. El sistema está basado en la distribución de patrones de la cara y el resto de la imagen que no lo era. Jeng propone un sistema que se basa en trabajar con rostros con la vista frontal y un poco inclinada, obteniendo características geométricas, el problema de este método era trabajar con imágenes que tienen más de un rostro y con imágenes de rostros con resoluciones mayores a  $80 \times 80$  pixeles[10]. Jain y Learned argumentan que el reconocimiento facial sigue siendo un reto, ya que las condiciones de iluminación, y la posición del rostro, son obstáculos que disminuyen los sistemas de reconocimiento facial [11]. Al igual que se han desarrollado grandes técnicas de reconocimiento facial, y gran número de métodos.

Adini [3] argumenta que el reconocimiento facial se ve afectado por las variaciones que existen en las imágenes, como las condiciones de luz y las posiciones. Afectando el proceso de reconocimiento por esas variaciones que provocan en imágenes de diferentes tamaños.

## Biométrie Facial

En [30], los autores mencionan que el reconocimiento facial es una de las aplicaciones más exitosas en el análisis de imágenes, ya que es aplicada en el campo comercial, como en la seguridad, que cuenta con la tecnología suficiente para su aplicación. Yang y Yhung mencionan que los métodos en la detección de rostros se clasifican en 4 categorías: métodos basadas en conocimientos, métodos características invariantes, por apariencia y juego de plantillas. Sistemas que se enfocan en los rasgos faciales, los cuales son: ojos, la nariz como centro y la boca [29].

En [26], Turk y Pentland no hablan de los sistemas basados en apariencia: los cuales son sistemas de aprendizaje automatizados.

[20] Sung y Poggio nos hablan de las características invariantes: enfocado en condiciones de iluminación o la rotación del rostro, color de piel incluso en los bordes del rostro.

Lanitis nos menciona los juegos de plantillas [15]: esta categoría nos propone un método de la detección de rostros que la correlación entre una imagen de prueba y la plantilla facial.

[12] Haiyuan y compañía trabajaron con un método para detectar rostros, el cual trabajaba con imágenes a color, este método está basado en la teoría difusa, y consiste en dos métodos generales, uno para obtener el color del pelo y otro para el color de piel. Comparando los colores obtenidos con un método de comparación de patrones basado la teoría Fuzzy para detectar en la base de datos a los rasgos con similitudes. El modelo propuesto presenta buenos resultados al detectar rostros del individuo a analizar, sin embargo presenta algunos falsos positivos.

[27] Viola y Jones proponen una técnica de detección de rostros utilizando el algoritmo adaBoost, que es un algoritmo de aprendizaje que se utiliza en clasificación. Aunque es más resistente al sobreajuste, es decir al sobre-entrenamiento, es sensible a los datos con ruido. Esta técnica es capaz de procesar muy rápido las imágenes, además logra incrementar la precisión de detección.

Chen y compañía desarrollan un algoritmo para detectar varios rostros en una imagen [8]. La idea principal del algoritmo está basaba en detectar los centros de los ojos y de la boca formando un triángulo, los rostros tenían que estar de frente para poder ser detectados, y no detectaba el rostro cuando los ojos eran tapados, cuando estaban cerrados e incluso con imágenes oscuras.

[16] Li y Zhang proponen un método llamado floatboost el cual entrenaba clasificadores. El algoritmo aprende una función discriminante basado en las interacciones de adaBoost, con una

mínima tasa de error.

[19] Shih y Chuang desarrollan un nuevo método que extrae las características de los ojos, nariz y boca. La idea principal del método que proponen consiste en separar el umbral alto y el umbral bajo separando el rostro del resto de la imagen y utilizando solo esta parte en la identificación.

[4] Baron nos propone una técnica en escala de grises, para el reconocimiento facial, su idea principal consiste en comparar los valores de intensidad utilizando distancia Euclidiana, esto con el propósito de encontrar las intensidades en las regiones que más se asemejan.

[9] Craw, Wiskoot y compañía realizaron un sistema de Reconocimiento Facial basado en las características faciales, como son el ancho de la nariz y su longitud, forma de la barbilla y posición de la boca. Aunque este sistema trabajaba bien con imágenes de poca resolución o imágenes con mucho ruido, era muy difícil de automatizar el sistema.

## Capítulo 2

# Preliminares

En este Capítulo se describen los conceptos básicos de imágenes y reconocimiento facial, así como las técnicas básicas de pre-procesamiento de la imagen.

### 2.1. Imágenes

Las primeras imágenes fueron realizadas en las cavernas, Realizadas por tribus, la pintura es una de las actividades más antiguas del hombre, siendo una de las formas de expresiones de la vida que llevaba el hombre en aquellas épocas. Entre el siglo XV y XVI Leonardo da Vinci fue quien empezó a desarrollar las bases de la fotografía, con la invención de las cámaras obscuras. La fotografía analógica fue un gran paso en la historia de la imagen, ya que en el año 1826 con Nicephore Niepce. Las ondas electromagnéticas se pueden considerar como ondas sinusoidales o partículas sin masa viajando o moviéndose a la velocidad de la luz. Dependiendo de la longitud de onda en metros, se representa el campo de visión donde se pueden visualizar las diferentes frecuencias como pueden ser: rayos gamma, rayos x, banda ultravioleta, banda infrarroja, microondas.

Para obtener una imagen se necesita un sensor que pueda obtener una imagen digital, del mundo real.

Una imagen es representada por una matriz formada de píxeles. Es la representación de un objeto del mundo real, panorama e incluso cosas irreal, como puede ser dibujos animados. Otra definición de imagen es una función  $f(x, y)$ , intensidad o nivel o nivel de grises en el punto

$(x, y)$ , siendo  $x$  e  $y$  las coordenadas espaciales.

Imagen Digital: Una imagen es definida de forma general como una función de dos dimensiones finitas  $f(x, y)$ , donde  $x$  e  $y$  representan las coordenadas de la matriz que componen dicha imagen, y que a su vez la intersección de coordenadas  $f(x, y)$  es un punto que representa un pixel.

Un pixel es la parte más mínima de una imagen, Un pixel representa un valor

En este Capítulo se describen los métodos básicos en procesamiento digital de imágenes. El objetivo del pre-procesamiento es mejorar la calidad de las imágenes a ser procesadas para el reconocimiento facial. Alguna de las funciones eliminar el ruido, aplicar el filtro, las distorsiones que existen en las imágenes a ser procesadas.

## 2.2. Técnicas de pre-procesamiento

El pre procesamiento, son el conjunto de técnicas que se realizan a las imágenes con el fin de mejorar las imágenes y así tener mejores imágenes de entrada para realizar su procesamiento y tener mejores resultados. Algunos pre procesamientos pueden ser la aplicación de filtros los cuales nos sirven para eliminar el ruido de una imagen.

### Cuantificacion y Muestreo

El proceso digital de una imagen consta de dos procesos fundamentales: muestreo y cuantificación. El muestreo se trata de dividir una imagen y descomponer en una serie de puntos o pixeles. En otro punto es el cambio de la resolución espacial original a otra deseada. La imagen está compuesta por una matriz de  $M \times N$  pixeles, los cuales serán descompuestos y redimensionados, tal como se muestra en la ecuación.

$$f(x, y) = \begin{bmatrix} f(0, 0) & f(0, 1) & \dots & f(0, N - 1) \\ f(1, 0) & f(1, 1) & \dots & f(1, N - 1) \\ \vdots & \vdots & \vdots & \vdots \\ f(M - 1, 0) & f(M - 1, 1) & \dots & f(M - 1, N - 1) \end{bmatrix} \quad (2-1)$$

Esta técnica provoca que en la imagen se noten aún más los pixeles, se creen efectos como el aliasing y deforme la imagen a un grado que puede no llegar a reconocerse si el tamaño a reducir

o aumentar sobrepasa el permitido por la misma imagen. Para realizar esto se tienen diversas técnicas. Aquí se mostrará una de ellas; otra técnica muy convencional en el tratamiento, además del muestreo de imágenes, es la cuantificación.

La Cuantificación, involucrada en el procesamiento de imagen la compresión de bits ( $k$ ) con pérdida que consiste en comprimir un rango de valores a un único valor dentro de las intensidades ( $L$ ) y el número de bits manejados por la misma imagen, es decir, si está en una resolución radiométrica de 8 bits, cambiarla a 2 o 7 bits, sin la pérdida de imagen.

Donde:

$$L = 2^{k-1} \quad (2-2)$$

Solamente se mueven las tonalidades. El proceso de cuantificación consiste en que cada uno de estos puntos se le asigna un valor en función de la luminosidad o color de la imagen original. Para determinar la resolución requerida, hay que determinar dos factores: cuál es la característica más pequeña a observar y cuál va a ser el campo de visión deseado a adquirir. Por tanto, habrá que determinar el factor de aumento o reducción en la imagen y el número de píxeles que debe de tener la nueva imagen resultante. Un ejemplo de ambas operaciones puede apreciarse en la siguiente Figura 2-1.

## **Operadores Puntuales**

Una imagen se define como una función  $f(x, y)$  donde cada posición representa una intensidad ya sea en una imagen a color o en una escala a grises en un punto  $(x, y)$  donde cada una de estas es una coordenada espacial dentro de la matriz. Como se muestra en la siguiente Figura 2-2.

Una imagen es también definida en forma más general como una función de dos dimensiones en un plano finito  $f(x, y)$  donde cada intersección de las coordenadas dadas en un punto representa un pixel. Las imágenes digitales son representadas en tres matrices de la forma RGB comúnmente (donde cada una representa una tonalidad en un plano Rojo, Verde y Azul) así



Figura 2-1: Figura. Operación de Cuantificación

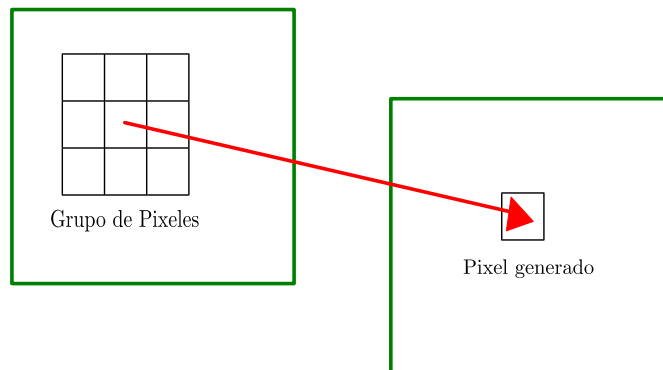


Figura 2-2: Figura. Los operadores puntuales se aplican al pixel de la misma coordenada que tendra de salida.

mismo existen diversas formas de representar la misma imagen para resaltar, opacar o simplemente corregir el color en esta. Para todo este proceso que se lleva a cabo dentro de una imagen es necesario aplicar el tratamiento digital de imágenes, donde es posible la manipulación el color, contrastes, operaciones entre estas, el cambio de intensidades, tamaño, filtros, por solo hacer mención de algunos aspectos a tratar pero todo lo anteriormente mencionado no sería posible sin el espectro electromagnético el cual al ser ondas variables que podemos manipular gracias a los grandes avances tecnológicos y así mismo a las ciencias en el campo de la física, química y con ayuda de la computación podemos plasmar el color en matrices que a su vez forman imágenes manipulables para nosotros. El campo del tratamiento digital de imágenes es basto, puesto que hoy en día todo es más visual y es por ello que es necesario tener un mejor apoyo dentro de este campo para las diversas aplicaciones, un ejemplo podría ser en la exploración geográfica por satélite, el campo de la medicina, dentro de la industria, entre varias otras. Pero dentro de la manipulación de imágenes existen también variantes para llevarlo a cabo o en este caso tres niveles:

- Nivel bajo: hace referencia a operaciones muy básicas, como el quitar ruido, elevar contraste, cambio de intensidades dentro de una imagen.
- Nivel medio: trata de la extracción de características especiales dentro de una imagen, reconocimiento de patrones.
- Nivel alto: maneja operaciones más sofisticadas en un nivel más “experto”.

En siguiente trabajo hace referencia solamente al primer nivel de los antes ya mencionados, en donde se podrá apreciar todas las técnicas vistas en clase dentro de la temática de operadores puntuales, donde esto nos habla como se aplican cambios o modificaciones en el tamaño, intensidades, variación de colores, brillo, entre otros en una imagen pero con el manejo de un solo pixel a la vez. Así mismo la implementación de una interfaz gráfica para el mismo manejo de las técnicas que a continuación se mostraran. Para ello es necesario tener en cuenta que en un plano cartesiano a como se representan las intensidades de una imagen el plano  $x$  es la imagen de entrada y el plano  $y$  es el resultante en la imagen de salida tal.

### **Funciones de los Operadores Puntuales**



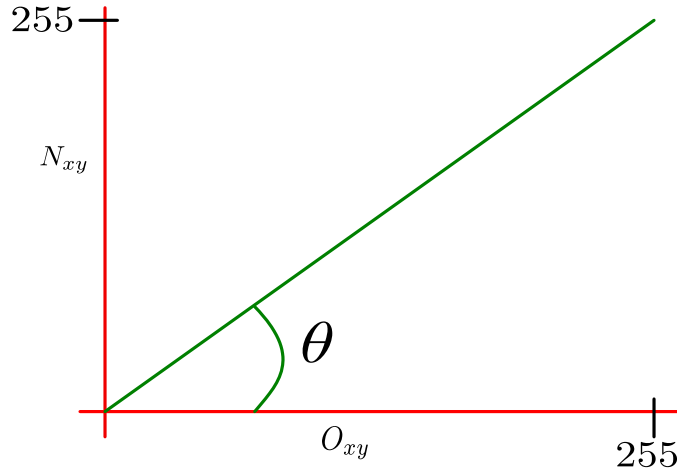


Figura 2-3: Figura. A cada valor de X le corresponde el mismo valor de Y.

Las operaciones básicas que son llevadas a cabo dentro de los operadores puntuales ejecutan una acción solamente sobre un pixel a la vez dentro de la misma imagen, obteniendo nuevos resultados y modificaciones en la imagen, ya sea cambio de contraste, intensidades o algún otro. La función básica es mostrada en la Figura 2-3. Donde cada valor contenido en el eje x, tiene un valor igual para el eje Y.

### Función Escalón

La función escalón se representa en algunos casos conocida también como función de umbralización, determina la intensidad de los pixeles en una imagen según una variable  $\alpha$ , donde si están por debajo de este límite cambian a 0 y en caso contrario, incrementan a 255, en diversos casos esta operación es utilizada para segmentar el frente de la imagen con el plano del fondo. La ecuación hace referencia a esta operación.

$$N_{xy} = \begin{cases} 0 & 0 \leq O_{xy} \leq \alpha \\ 1 & \alpha < O_{xy} \leq 255 \end{cases} \quad (2-3)$$

En la siguiente imagen Figura 2-4 se muestra la gráfica de la función Escalón dentro de la función Identidad y en la siguiente Figura 2-5 se aplica a una imagen.

### Función Escalonada

Esta función crea diversos niveles de intensidades en la imagen resultante, en base a valores

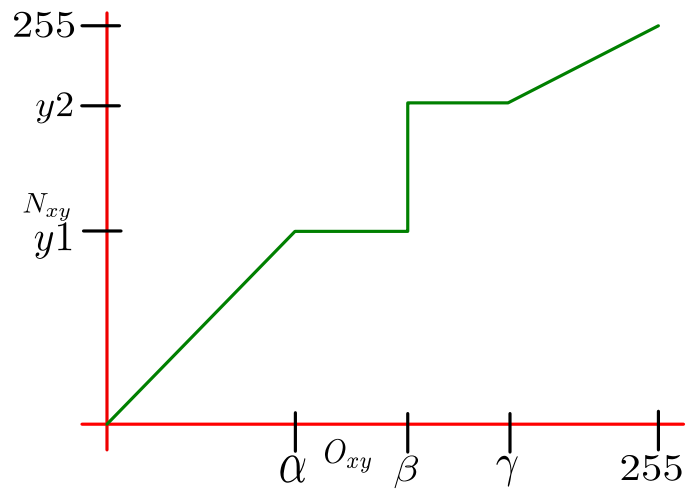


Figura 2-4: Figura. Función Escalon.



Figura 2-5: Figura. Función Escalon aplicada a una imagen.

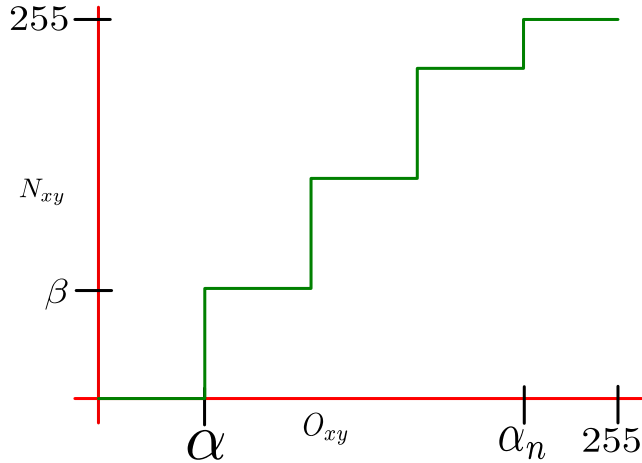


Figura 2-6: Figura. Función Escalonada.

$\alpha$  marcados sobre la imagen origen hasta un valor  $\alpha_n$  y un límite  $\beta$  en la imagen de salida. Gráficamente la función escalonada puede apreciarse en la Figura 2-6 se observa la función y aplicada a una imagen en la Figura 2-7 se aplica a una imagen.

### Incremento de Intensidades

Esta función incrementa los valores de intensidad en cada valor de una matriz, como se muestra la función representada por la gráfica en la Figura 2-8 y en la Figura 2-9 se muestra la función aplicada a una imagen.

Podemos manejar que todos los valores están dados por la ecuación

$$N_{xy} = \left\{ a \quad \alpha \leq O_{xy} \leq \alpha_n \right\} \quad (2-4)$$

Donde  $a$  es:

$$a = \left( \frac{255}{\alpha_n - \alpha} \right) O_{xy} + b \quad (2-5)$$

Teniendo así que en la imagen de entrada todos aquellos valores en el rango de  $\alpha$  hasta  $\alpha_n$  serán calculados otra vez, y en caso contrario igualados a 0.

### Reducción de Intensidades

Esta función nos permite disminuir los valores de intensidad de cada valor de una matriz,



Figura 2-7: Figura. Función Escalonada aplicada a una imagen.

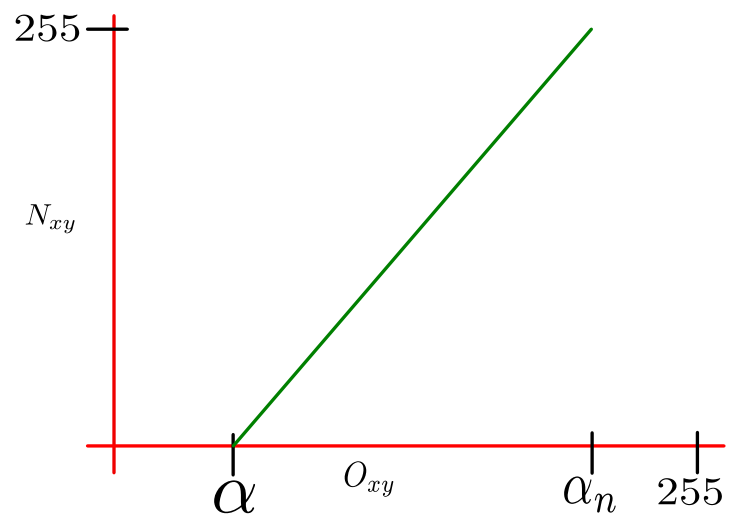


Figura 2-8: Figura. Función Incremento de Intensidades.



Figura 2-9: Figura. Función Incremento de Intensidad aplicada a una imagen.

la gráfica representa esta función, representada en la Figura 2-10 en la Figura 2-11 se aplica a una imagen.

Teniendo así que en la imagen de salida todos aquellos valores en el rango de 0 hasta  $\alpha$  serán calculados de nuevo basándose en los valores de las intensidades que se tienen en la imagen de entrada tomando en cuenta que está en un rango de 0 a 255. Representado mediante la ecuación.

$$N_{xy} = \left( \frac{\alpha}{255} \right) O_{xy} + b \quad (2-6)$$

### **Diente de sierra**

La gráfica de la Figura 2.08 representa la aplicación de esta función en la Figura. 2-12

Con esto podemos decir que todos aquellos valores que se encuentren en un rango de las intensidades de la imagen entre  $\alpha$  y  $\alpha_n$  serán calculados de nuevo hasta llegar a un punto  $\beta$  en la imagen de salida en un plano de 0 a 255. Para ello es necesario calcular un incremento de 0 a  $\frac{\alpha}{2}$  en  $y$  hasta  $\beta$  y en ese punto comenzar el decremento hasta llegar a  $\alpha$ , así mismo realizar el mismo paso con nuevos valores en cada pico según sea necesario.

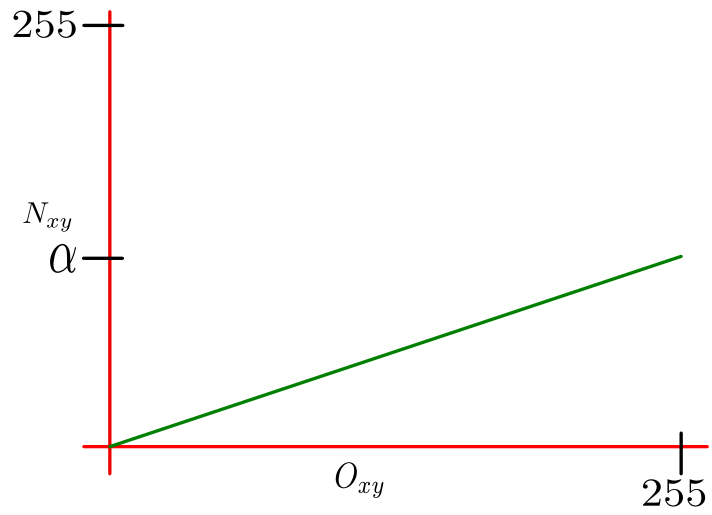


Figura 2-10: Figura. Función de Reducción de Intensidad.



Figura 2-11: Figura. Función Reducción de Intesidad.

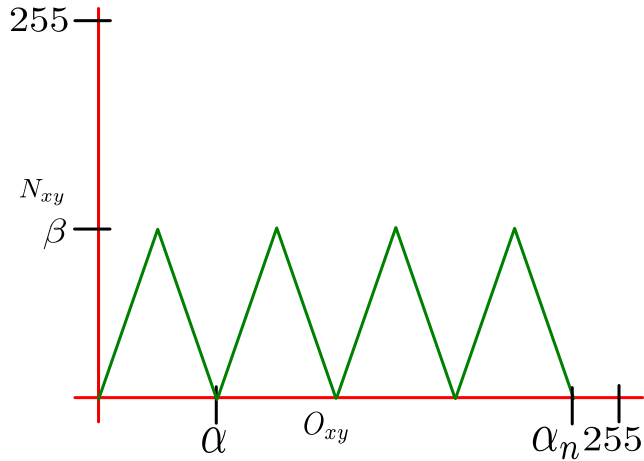


Figura 2-12: Figura. Función Dientes de Sierra.

### Función de corrección Gamma

La corrección gamma es una función utilizada para corregir los valores de iluminación en una imagen, en otras palabras, una forma especial de aumento de contraste en áreas con más brillo o muy oscuras, esto se logra modificando los valores medios, sin afectar negros (0) o blancos (255). La corrección gamma es representada por la siguiente fórmula, donde  $N_{xy}$  es la imagen resultante y  $O_{xy}$  es la imagen de entrada a corregir. Esta función es representada mediante la ecuación.

$$N_{xy} = 255 \left( \frac{O_{xy}}{255} \right)^{\frac{1}{\gamma}} \quad (2-7)$$

### Función Escalón dentro de Identidad

La función identidad señala que todos aquellos valores que están en  $O_{xy}$  pertenecen a un valor en  $N_{xy}$ , esto es, que no tienen ninguna modificación. La función escalón dentro de la función identidad es definida por la condición, donde todos aquellos valores debajo de  $\alpha$  se conservan, los que se encuentran de  $\alpha$  a  $\beta$  son igualados al tope  $y1$  y si están de  $\beta$  a  $\gamma$  son igualados a  $y$  255, en dado caso de sobrepasar el valor  $\gamma$  se siguen conservando sin alteración alguna

:

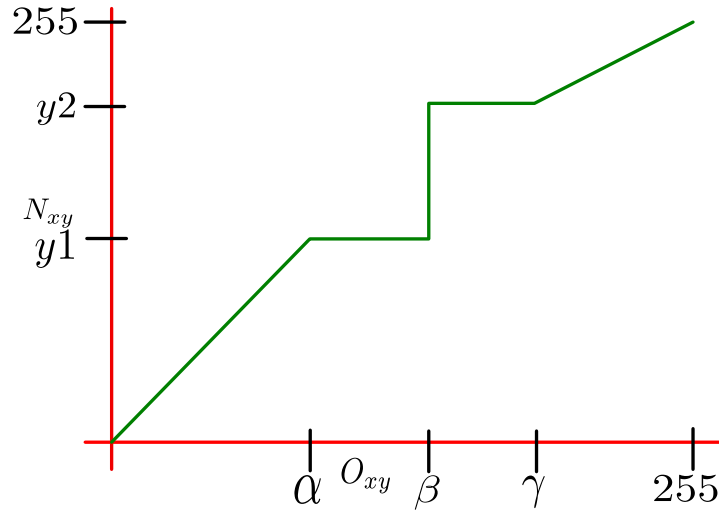


Figura 2-13: Figura. Función Escalon dentro de Identidad.

$$N_{xy} = \left\{ \begin{array}{ll} O_{xy} & 0 \leq O_{xy} \leq \alpha \\ y1 & \alpha < O_{xy} \leq \beta \\ y2; & \beta < O_{xy} \leq \gamma \\ O_{xy} & \gamma < O_{xy} \leq 255 \end{array} \right\} \quad (2-8)$$

Gráficamente se representa de la Figura 2-13 Figura y en la 2-14 Figura se aplica a una imagen.

### Expansión de Contraste

La gráfica de esta función se representa en la Figura 2-15 y en la Figura 2-16 se aplica en una imagen.

Se tiene que basados en la función identidad esta va a cambiar su inicio y final con respecto a un nuevo punto  $\alpha$  y un nuevo fin  $\beta$ . Manejando así una nueva condición definida por la función, basados en un rango de intensidades de 0 a 255, donde sí los valores están debajo de  $\alpha$  serán 0 y por encima de  $\beta$  serán 255, en dado caso que no se calculan:





Figura 2-14: Figura. Función Escalon dentro de Identidad aplicada a una imagen.

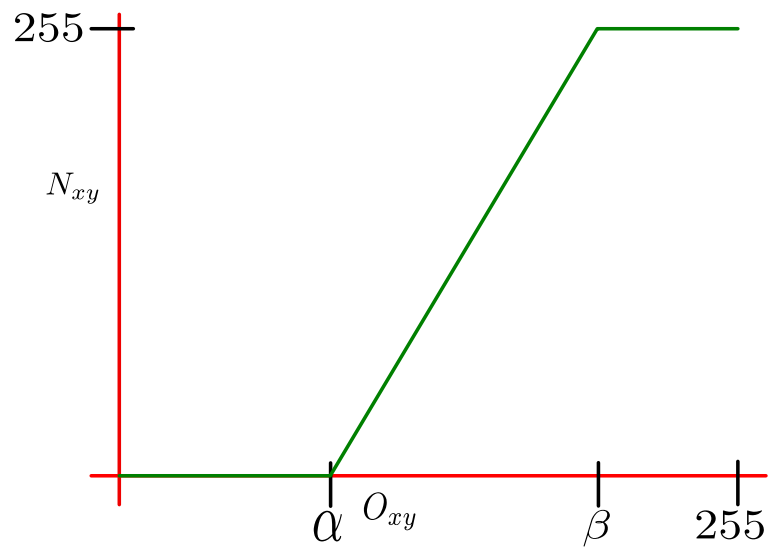


Figura 2-15: Figura. Función Expansión de Contraste.



Figura 2-16: Figura. Función Expansión de Contraste aplicada a una imagen.

$$N_{xy} = \left\{ \begin{array}{ll} 0 & 0 \leq O_{xy} \leq \alpha \\ a & \alpha < O_{xy} < \beta \\ 255 & \beta \leq O_{xy} \leq 255 \end{array} \right\} \quad (2-9)$$

Donde  $a$  es:

$$a = \left( \frac{255}{\beta - \alpha} \right) O_{xy} + b \quad (2-10)$$

### Función Parábola

La función de la parábola tiene efecto sobre las diferentes intensidades de la imagen, aumentándolas, si se encuentran en el rango manejado, la ecuación hace referencia a esta operación.

$$N_{xy} = 255 - 255 \left( \frac{O_{xy}}{128 - 1} \right)^2 \quad (2-11)$$

En la gráfica de la Figura 2-17 puede verse como es que esta función actúa sobre la imagen y en la Figura 2-18 se ve el resultado de la función aplicada.

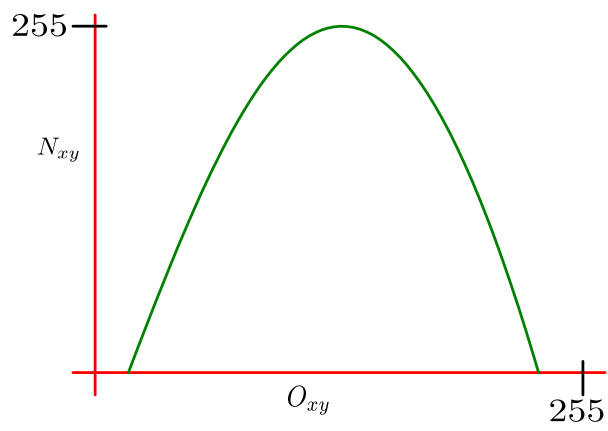


Figura 2-17: Figura. Función Parábola.



Figura 2-18: Figura. Función Parábola aplicada a una imagen.



Figura 2-19: Figura. Función de Parábola Inversa aplicada a una imagen.

### Función Parábola Inversa

De la misma forma que en la parábola, tiene ese efecto sobre el número de intensidades que se tiene pero a diferencia de la parábola, esta función invierte los valores en un rango de 0 a 255. Al decir que invierte, es que disminuye su valor.

$$N_{xy} = 255 \left( \frac{O_{xy}}{127} \right)^2 \quad (2-12)$$

En la siguiente Figura 2-19 se ve el resultado de la función Parábola Inversa.

### Función Polarizada

La función toma todos aquellos valores en un rango de 0 hasta un valor  $\theta$  de una imagen entrante  $O_{xy}$  a una resultante  $N_{xy}$ , y todos aquellos valores que sobre pasen ese rango, obtiene su valor invertido en base a la función identidad. Se maneja lo antes dicho en la función .

$$N_{xy} = \left\{ \begin{array}{ll} O_{xy} & 0 \leq O_{xy} \leq \theta \\ 255 - O_{xy} & \theta < O_{xy} \leq 255 \end{array} \right\} \quad (2-13)$$

Es representado graficamente en la Figura 2-20. Y el resultado en una imagen real puede verse reflejado sobre la Figura 2-21.

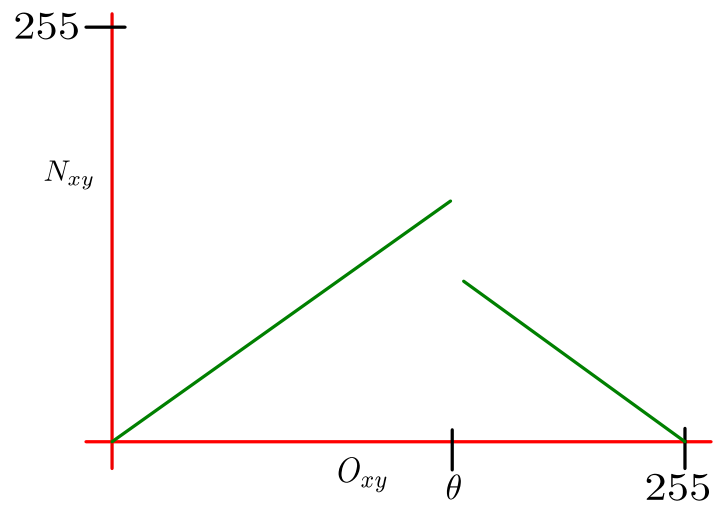


Figura 2-20: Figura. Función de Polarizado.



Figura 2-21: Figura. Función Polarizada aplicada a una imagen.

### Adición de Brillo

La función de aumento de brillo mantiene todos aquellos valores que están por debajo de un tope  $\alpha$  en la imagen entrante que se encuentra en un rango de 0 a 255 intensidades. En caso contrario que no sea así y los valores sobrepasen esta condición, tomaran nuevos valores iguales a 255. Esta condición se muestra en la función

$$N_{xy} = \left\{ \begin{array}{ll} O_{xy} & 0 \leq O_{xy} \leq \alpha \\ 255 & \alpha < O_{xy} \leq 255 \end{array} \right\} \quad (2-14)$$

Aplicando la adición de Brillo a dos imágenes tenemos como resultado de esta operación.

### Función Expansión

La operación de expansión nos dice que todo aquel valor debajo de un tope  $\theta$  será igual en la imagen resultante con respecto a una imagen de entrada y si sobrepasa este tope el nuevo valor tendrá una equivalencia del valor límite que es 255 menos el valor de la intensidad en ese punto, se representa mediante la función.

$$N_{xy} = \left\{ \begin{array}{ll} O_{xy} & O_{xy} \leq \theta \\ 255 - O_{xy} & O_{xy} > \theta \end{array} \right\} \quad (2-15)$$

La Figura 2-22 es el resultado obtenido de esta operación sobre una imagen.

## 2.2.1. Operaciones Lógicas, Estadísticas y Aritméticas

### Operaciones Lógicas

Las operaciones lógicas son las operaciones básicas and, or, xor, not y sub, las cuales se deben realizar con imágenes a 2 bits para tener una nueva imagen resultante. Estas operaciones son realizadas entre dos imágenes solamente para obtener una tercera. Al trabajar siempre con

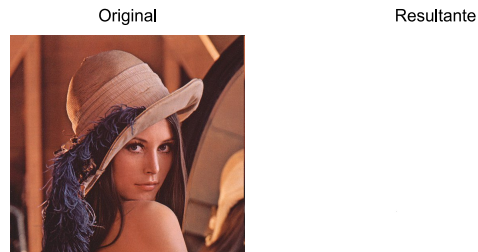


Figura 2-22: Figura. Función Expansión aplicada a una imagen.

imágenes en 8 bits, es difícil poder realizar estas operaciones por esta razón, es necesario aplicar una operación llamada Binarización.

La binarización es el cambio de una imagen que está en 8 bits, esto es que se representa con valores de 0 a 255, pasarla a una imagen de 7 bits, 2 bits, que solo contiene valores ya sean 0 o 1 o  $n$ -bits. El siguiente algoritmo representa el método aquí empleado para lograr la binarización de las imágenes. Un ejemplo de una imagen binarizada puede verse claramente en la Figura.

### Operación AND

La operación AND, como bien se sabe esta es representada mediante la siguiente tabla.

Cuadro 2-1: Operación AND

A	B	A&B
0	0	0
0	1	0
1	0	0
1	1	1

Donde si ambos valores son iguales a 1, el resultado será positivo (1) y en todo caso contrario el resultado será lo opuesto.

### Operación OR

La operación OR, esta es representada mediante la siguiente tabla.

Cuadro 2-2: Operación OR

A	B	A B
0	0	0
0	1	1
1	0	1
1	1	1



Figura 2-23: Figura. Función OR aplicada a dos imagenes.

Donde si uno solo de los valores son iguales a 1, el resultado será positivo (1) y en todo caso contrario el resultado será lo opuesto. El resultado de esta operación puede apreciarse en la Figura 2-23.

### Operación XOR

la operación XOR, se representa mediante la siguiente tabla.

Cuadro 2-3: Operación XOR

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0





Figura 2-24: Figura. Función XOR aplicada a dos imagenes.

Donde, si ambos valores son iguales el resultado será negativo (0) y en todo caso contrario el resultado es lo opuesto. El resultado de esta operación puede apreciarse en la Figura 2-24.

### Operación SUB

La operación SUB, esta es representada mediante la siguiente tabla.

Cuadro 2-4: Operación SUB

A	B	A\B
0	0	0
0	1	0
1	0	1
1	1	0

Donde, el primer valor debe ser positivo y el segundo negativo para tener un resultado verdadero (1) y en todo caso contrario el resultado es lo opuesto.

### Operación NOT

La operación NOT, se representa mediante la siguiente tabla.

Donde el valor que se tiene, es cambiado simplemente por su valor contrario. La Figura 2-25 representa esta operación.

Cuadro 2-5: Operación NOT

A	$\neg A$
0	1
1	0



Figura 2-25: Figura. Función NOT aplicada a una imagen.

### 2.2.2. Operaciones Aritméticas

Las operaciones aritméticas a diferencia de las operaciones lógicas, deben ser llevadas a cabo con imágenes que estén en escala de grises donde las intensidades de la imagen están entre el 0 y los 255. Para este caso la umbralización no es necesaria.

#### Adición

La adición o suma es llevada a cabo al igual que una suma de matrices, donde se suma de término a término al valor de  $A$  con su respectivo reflejo en  $B$ .

$$\begin{array}{c}
 \hline
 A \quad + \quad B \\
 \hline
 \left[ \begin{array}{cc} a_{11} & a_{12} \\ a_{21} & a_{22} \end{array} \right] + \left[ \begin{array}{cc} b_{11} & b_{12} \\ b_{21} & b_{22} \end{array} \right] = \left[ \begin{array}{cc} a_{11} + b_{11} & a_{12} + b_{12} \\ a_{21} + b_{21} & a_{22} + b_{22} \end{array} \right] \\
 \hline
 \end{array} \tag{2-16}$$

### Substracción

La sustracción o resta es llevada a cabo al igual que una resta de matrices, donde se resta de término a término al valor de  $A$  con su respectivo reflejo en  $B$ . Aplicando la Substracción a dos imágenes tenemos como resultante.

$$\begin{array}{c} \hline A \quad - \quad B \\ \hline \left[ \begin{array}{cc} a_{11} & a_{12} \\ a_{21} & a_{22} \end{array} \right] + \left[ \begin{array}{cc} b_{11} & b_{12} \\ b_{21} & b_{22} \end{array} \right] = \left[ \begin{array}{cc} a_{11} - b_{11} & a_{12} - b_{12} \\ a_{21} - b_{21} & a_{22} - b_{22} \end{array} \right] \\ \hline \end{array} \quad (2-17)$$

### División

La división entre matrices como tal no es posible, para ello es necesario llevar varias operaciones características de las mismas matrices, las operaciones realizadas es multiplicar la matriz  $A$  que tiene el rol de numerados, por la inversa de la matriz  $B$  siendo este el denominador.

$$C = AB^{-1} \quad (2-18)$$

### Multiplicación

La multiplicación es llevada a cabo al igual que una multiplicación de matrices, donde se multiplica cada elemento que pertenece a las filas en  $A$  por cada elemento en las columnas de  $B$ .

$$\begin{array}{c} \hline A \quad * \quad B \\ \hline \left[ \begin{array}{cc} a_{11} & a_{12} \\ a_{21} & a_{22} \end{array} \right] * \left[ \begin{array}{cc} b_{11} & b_{12} \\ b_{21} & b_{22} \end{array} \right] = \left[ \begin{array}{cc} (a_{11} * b_{11}) + (a_{12} * b_{21}) & (a_{11} * b_{12}) + (a_{12} * b_{22}) \\ (a_{21} * b_{11}) + (a_{22} * b_{21}) & (a_{21} * b_{12}) + (a_{22} * b_{22}) \end{array} \right] \\ \hline \end{array}$$

### Logaritmo

La operación del logaritmo a base 10 es solamente implementado a una matriz  $A$  en este caso esa matriz es la imagen de entrada, el resultado de la imagen es difícil de apreciar por los valores tan bajos obtenidos, siendo así que la imagen está muy oscura.

$$B = \log(A) \quad (2-19)$$

## Exponencial

La operación del exponente es implementado en la función  $e^x$ , donde  $x$  es cada valor de una imagen o matriz  $A$ , el resultado de la imagen es todo en blanco, manejando que está en una escala de 8 bit o teniendo como límite el 255, por la operación de la exponencial.

$$B = e^A \tag{2-20}$$

## Raíz Cuadrada

La operación de la raíz cuadrada está bajo la siguiente función  $\sqrt{x}$ , donde  $x$  es un punto o pixel dentro de una imagen, el resultado de la imagen es una nueva imagen más oscura.

$$B = \sqrt{A} \tag{2-21}$$

### 2.2.3. Operaciones Trigonométricas

Las operaciones trigonométricas básicas son solamente implementado a una imagen, el resultado de la imagen es difícil de apreciar ya que los valores obtenidos no pasan de 0 o 1, ya que al ser estas periódicas, sin importar el número tan elevado solo se obtendrá su equivalencia en un ciclo de 0 a  $2\pi$ .

$$\left| \begin{array}{l} B = \sin(A) \\ B = \cos(A) \\ B = \tan(A) \end{array} \right| \tag{2-22}$$

Al ejecutar esta operación se obtienen las imágenes que se muestra en la Figura 2-26 de Seno, Figura 2-27 de Coseno y Figura 2-28 de la Tangente.

## Inversa

La operación inversa debe ser llevada a cabo con ambas imágenes en la ecuación puede observarse como se realiza esta operación, donde  $A$  y  $B$  son las respectivas imágenes o matrices para obtener una nueva, el resultado de la imagen, al tener valores muy oscilante, se obtiene una nueva imagen con muchas tonalidades de blancos y algunas notorias en negro, el siguiente código representa la operación.



Figura 2-26: Figura. Función Trigonometrica Seno aplicada a una imange.



Figura 2-27: Figura. Funcipon Trigonometrica Coseno aplicada a una imagen.



Figura 2-28: Figura. Función Trigonometrica Tangente aplicada a una imagen.

$$C = (2^B - 1) - A \quad (2-23)$$

## 2.2.4. Operaciones Estadísticas

### Media

Es utilizada para obtener el promedio general que existe en un conjunto determinado de números, esto es que se calcula la frecuencia con la que un número hace presencia en este conjunto, en nuestro caso es solamente el promedio de intensidades manejadas para la imagen. Esta operación es representada mediante la ecuación:

$$\mu = \sum_i \sum_j \frac{O_{(i,j)}}{M \times N} \quad (2-24)$$

Donde:

$O_{(i,j)}$  : representa un punto en la imagen o pixel.

### Varianza

La varianza calcula la medida en como están dispersos un conjunto de valores numéricos, para este caso es poder calcular cuan separadas están las intensidades dentro de la imagen si sabemos que los valores que se tienen varían entre el 0 y 255. Esta operación es representada

mediante la ecuación

$$\sigma^2 = \sum_{y=0}^{2^L-1} (y - \mu)^2 P(y) \quad (2-25)$$

Donde:

$y$  : representa la intensidad de un punto en la imagen o pixel.

$\mu$  : es la media aritmética.

$P(y)$  : representa la probabilidad de las intensidades dentro de la imagen.

$L$  : el número de bits en los cuales la imagen se encuentra, generalmente a 8 bits.

### **Asimetría**

La asimetría nos permite identificar si la distribución de un dato dentro del conjunto numérico manejado es uniforme con respecto al punto central manejado. Para nuestro caso en el conjunto de datos se tienen los valores que contiene la imagen en cada pixel. Esta operación es representada mediante la ecuación.

$$a = \sum_{y=0}^{2^L-1} (y - \mu)^3 P(y) \quad (2-26)$$

Donde:

$y$  : representa la intensidad de un punto en la imagen o pixel.

$\mu$  : es la media aritmética.

$P(y)$  : representa la probabilidad de las intensidades dentro de la imagen.

$L$  : el número de bits en los cuales la imagen se encuentra, generalmente a 8 bits.

### **Energía**

La energía representa la suma de los cuadrados de la probabilidad con que hace presencia cada una de las intensidades dentro de la imagen. Esta operación es representada mediante la ecuación.

$$E = \sum_{y=0}^{2^L-1} (P(y))^2 \quad (2-27)$$

Donde:

$y$  : representa la intensidad de un punto en la imagen o pixel.

$P(y)$  : representa la probabilidad de las intensidades dentro de la imagen.

$L$  : el número de bits en los cuales la imagen se encuentra, generalmente a 8 bits.

### Entropía

Describe todos en una forma mas general aquel punto dentro de la variación de valores de energía emitidos por cada pixel que no emiten alguna diferencia o variación en general con respecto a los demás valores. Esta operación es representada mediante la ecuación.

$$e = \sum_{y=0}^{2^L-1} P(y) \text{Log}_2 [P(y)] \quad (2-28)$$

Donde:

$y$  : representa la intensidad de un punto en la imagen o pixel.

$P(y)$  : representa la probabilidad de las intensidades dentro de la imagen.

$L$  : el número de bits en los cuales la imagen se encuentra, generalmente a 8 bits.

### Oblicuidad

La oblicuidad se define como la relación existente entre la media y la mediana, por otro lado es que tan asimétrica es la mediana por lo tanto en manera más general define el grado en que una distribución de probabilidad de una variable aleatoria real se inclina en comparación con la mediana. Esta operación es representada mediante la ecuación.

$$\gamma_3 = \frac{1}{\sigma^3} \sum_{y=0}^{2^L-1} (y - \mu)^3 P(y) \quad (2-29)$$

Donde:

$y$  : representa la intensidad de un punto en la imagen o pixel.

$\mu$  : es la media aritmética.

$P(y)$  : representa la probabilidad de las intensidades dentro de la imagen.

$L$  : el número de bits en los cuales la imagen se encuentra, generalmente a 8 bits.

$\sigma$  : representa la varianza del conjunto de datos a tratar.

### Kurtosis



La kurtosis estudia la proporción de la varianza que se explica por la combinación de datos sumamente grandes con respecto a la mediana de los mismos. Mientras el valor de esta sea más grande define que la concentración de datos está muy cercana a la media de la distribución de estos. Esta operación es representada mediante la ecuación.

$$\gamma_4 = \frac{1}{\sigma^4} \sum_{y=0}^{2^L-1} (y - \mu)^4 P(y) - 3 \quad (2-30)$$

Donde:

$y$  : representa la intensidad de un punto en la imagen o pixel.

$\mu$  : es la media aritmética.

$P(y)$  : representa la probabilidad de las intensidades dentro de la imagen.

$L$  : el número de bits en los cuales la imagen se encuentra, generalmente a 8 bits.

$\sigma$  : representa la varianza del conjunto de datos a tratar. Para este caso es la varianza existente en la imagen.

## 2.2.5. Operadores Grupales

### Introducción

Dentro del tratamiento de imágenes se tienen dos tipos de caminos para el mejoramiento de la imagen:

- Dominio espacial.
- Dominio frecuencia.

En este apartado se tomara aquellos del dominio espacial que abarca a los operadores grupales y operadores puntuales, centrándonos más en los primeros los cuales tienen efecto sobre un solo pixel en base al conjunto de pixeles que lo rodea. En la Figura 2-29 se muestra una serie de valores a los cuales se efectúa una operación para afectar a un pixel.

Para algunos casos de estos filtros la matriz en cómo se aplica llega a desbordarse para evitarlo es necesario agregar bordes, donde estos pueden ser blancos, negros, infinitos donde se replica el pixel de las orillas o envolventes cambiando los pixeles de las orillas a su lugar opuesto. El uso de filtros ayudan a resaltar o hacer notar diferentes características dentro de la

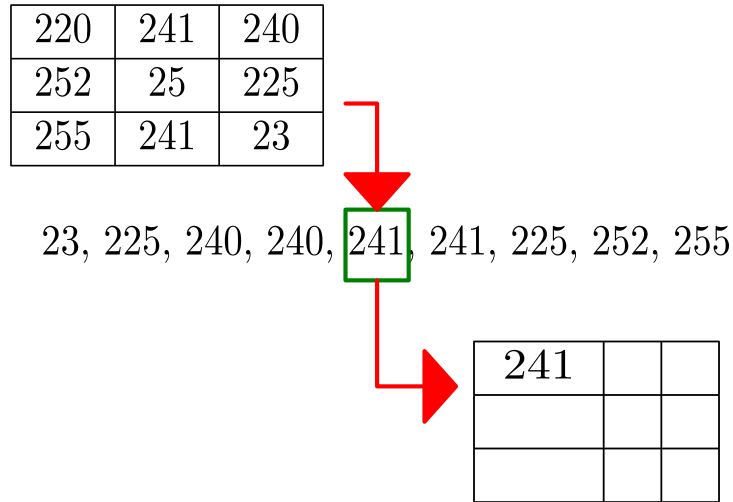


Figura 2-29: Función. Se aplica operaciones al conjunto de pixeles que lo rodean para dar un nuevo valor a un pixel.

misma imagen que se desean <<filtrar>> para poder trabajar con ellas, así mismo es posible de estas imágenes obtener sus histogramas los cuales representan la distribución de intensidades por las cuales está compuesta la imagen, contando con la facilidad de cambiar el nivel de intensidades y el contraste para una visión más clara del objeto que se desea ver o si se desea ver en un conjunto acotado de intensidades, ya sea contrayendo o expandiendo el histograma. Igual se puede acercar o alejar a un área en específico de la imagen con ayuda de diferentes interpolaciones que permiten agrandar o reducirla en base a diferentes algoritmos, los cuales, al igual que los antes ya mencionados serán mostrados en el trabajo a continuación.

### **Filtros**

Ha diferencia de los operadores puntuales que tienen efecto sobre un pixel a la vez.. El efecto es más notable mientras este filtro sea mayor en comparación a la imagen original. Estos filtros varían según el tipo que se desea ser utilizado y en el efecto que tendrá dentro de la nueva imagen creada será lo mismo. Generalmente se usa un filtro básico como se muestra en la ecuación:

$$W = \begin{bmatrix} W_1 & W_2 & W_3 \\ W_4 & W_5 & W_6 \\ W_7 & W_8 & W_9 \end{bmatrix} \quad (2-31)$$

El tamaño de la matriz como puede observarse es de  $3 \times 3$  pero este puede variar entre  $5 \times 5$ ,  $7 \times 7$ ,  $9 \times 9$  pero mientras más grande sea, más tiempo toma y más complejo se vuelve, donde el pixel más significativo y de referencia donde se aplicará el filtro es el que se encuentra en el centro, lo antes dicho está definido por la ecuación.

$$N_{xy} = W_1 O_{-1,-1} + W_2 O_{0,-1} + W_3 O_{1,-1} + W_4 O_{-1,0} + W_5 O_{0,0} + W_6 O_{1,0} + W_7 O_{-1,1} + W_8 O_{0,1} + W_9 O_{1,1}$$

Para la matriz de filtros, generalmente tienen la forma siguiente mostrada en la ecuación. Donde se aplica en algunos casos la operación a la matriz de posiciones para poder obtener el filtro deseado.

$$\begin{bmatrix} -1, -1 & 0, -1 & 1, -1 \\ -1, 0 & 0, 0 & 1, 0 \\ -1, 1 & 0, 1 & 1, 1 \end{bmatrix} \quad (2-32)$$

### **Filtro promedio o media**

Este tipo de filtro suaviza la imagen asignando la media de todos aquellos que lo rodean al pixel con el que se esté trabajando, para ello es la sumatoria de dicho conjunto de pixeles entre el número de elementos que contengan. Esto está dado por la ecuación.

$$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (2-33)$$

### **Filtro media ponderada**

Al igual que el filtro promedio o media es la suma media de todos los elementos del filtro pero dando un mayor valor o peso al pixel que se encuentra en el centro para evitar la pérdida de información detallada, la ecuación, define este filtro.

$$\frac{1}{10} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (2-34)$$

### **Filtros Gaussianos**

Los filtros Gaussianos son conocidos por los efectos de poner borrosa la imagen a la que se aplica. Quitar o aplicar ruido, por hacer mención de algunos efectos que pueden tener estos sobre la imagen utilizada.

#### **Filtro Gaussiano 3x3**

Es definido por la ecuación. Donde la aplicación al igual que el promedio es la multiplicación y sumatoria del grupo de pixeles sobre el más significativo que se encuentra al centro y dividido entre el total de los valores del filtro.

$$\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \quad (2-35)$$

Al ejecutar esta operación se obtiene la imagen que se muestra en la figura:

#### **Filtro Gaussiano 5x5**

Está definido por la ecuación. Donde al igual que el filtro promedio o media es necesario dividir la suma de las multiplicaciones entre el número total de los valores de los elementos.

$$\frac{1}{100} \begin{bmatrix} 1 & 2 & 4 & 2 & 1 \\ 2 & 4 & 8 & 4 & 2 \\ 4 & 8 & 16 & 8 & 4 \\ 2 & 4 & 8 & 4 & 2 \\ 1 & 2 & 4 & 2 & 1 \end{bmatrix} \quad (2-36)$$

Al ejecutar esta operación se obtiene la imagen que se muestra en la figura.

### **Filtros Gaussianos “Típicos”**

Están dados en base a una desviación estándar  $\sigma$  la cual tiene un efecto sobre saliente sobre cada uno de ellos, porque si esta llega a ser muy pequeña, el filtro casi no tendrá efecto sobre la imagen y en caso contrario se torna muy borrosa.

$$FiltroG(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (2-37)$$

Es definido por la ecuación, la cual permite suavizar y difuminar de alguna la imagen, todos los valores obtenidos oscilan entre 0 y 1.

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (2-38)$$

Para poder obtener el filtro, estas operaciones se aplican sobre una matriz de posiciones definida, al obtener una nueva, esta se aplica a la imagen.

Al ejecutar esta operación se obtiene la imagen que se muestra en la figura.

$$FiltroG(x, y) = e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (2-39)$$

Depende al igual que la anterior de una desviación estándar, el efecto que toma sobre una imagen es el “suavizar” o mejor dicho, poner borrosa la imagen en base a la desviación manejada, mientras esta sea mayor, más se pierde la imagen. Este filtro es definido por la ecuación.

$$G(x, y) = e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (2-40)$$

$$FiltroG(x, y) = \frac{1}{2\sqrt{2\pi}\sigma} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (2-41)$$

Se define de igual forma por una desviación estándar la cual permite que al aplicar este algoritmo sobre una imagen, esta se suavice y creando un espectro de la imagen. La ecuación, define este filtro.

$$G(x, y) = \frac{1}{2\sqrt{2\pi}\sigma} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (2-42)$$

### **Filtro mediana**

Del grupo de valores obtenidos por un filtro vacío que es llenado con los valores propios de la imagen original, se obtiene la mediana, donde primeramente este grupo se ordena de mayor a menor y si el número de elementos del grupo es impar se coloca en la nueva imagen el valor central entre estos, en caso contrario se toma el valor de la intensidad más cercada al 255. El tamaño del kernel es igualmente definido según se desee trabajar con él. Un ejemplo es dado es la figura.

### **Filtro moda**

Se calcula al igual que la moda en estadística donde la intensidad que más se repita es la que se toma, en dado caso de existir dos iguales se toma la que este más cercana al 255. La figura, ejemplifica este filtro.

### **Filtro basados en la derivada**

Estos filtros son especiales para intensificar y acentuar detalles y cambios bruscos en las intensidades dentro de una imagen, donde se hace más notable el cambio de una imagen entre blancos y negros, por dar un ejemplo, acentuando más las orillas.

### **Filtro Prewitt**

Este filtro se define por las siguientes dos ecuaciones, donde la ecuación, es para los valores en  $x$ .

$$P_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad (2-43)$$

Para valores de  $y$ .

$$P_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad (2-44)$$

### **Filtro Sobel**

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (2-45)$$

$$S_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (2-46)$$

### Filtro Laplaciano

Este filtro está basado en la segunda derivada el cual es recomendado para realzar rasgos lineales, generalmente en entornos urbanistas, donde son necesarios observar los elementos con mayor vulnerabilidad. Para la obtención de este filtro se aplican las siguientes ecuaciones, donde la ecuación, es para obtener con respecto a  $x$ .

$$\frac{\partial^2 f}{\partial x^2} \simeq f'(x) - f'(x + 1) \simeq \frac{\partial [f(i, j) - f(i, j - 1)]}{\partial x} \quad (2-47)$$

la ecuación, para los valores en  $y$ .

$$\frac{\partial^2 f}{\partial y^2} \simeq f'(y) - f'(y + 1) \simeq \frac{\partial [f(i, j) - f(i, j - 1)]}{\partial y} \quad (2-48)$$

Obteniendo así mismo las siguientes nuevas ecuaciones tanto para  $x$  y  $y$ .

$$L_x = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -2 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (2-49)$$

$$L_y = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad (2-50)$$

### Otros filtros basados en la derivada

Al igual que el filtro Laplaciano sirven para resaltar rasgos dentro de una imagen, las ecuaciones definen este tipo de filtros, generalmente diferentes autores los llaman filtros menos

media.

$$\frac{1}{4} \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (2-51)$$

$$\frac{1}{8} \begin{bmatrix} 0 & 1 & 0 \\ 1 & -8 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (2-52)$$

### Filtro LoG

Al igual que los filtros Gaussianos va en referencia a una varianza  $\sigma$ , donde se puede generar un filtro según el tamaño que se desee. La ecuación, define este filtro.

$$LoG(x, y) = \frac{1}{\pi\sigma^4} \left[ 1 - \frac{x^2 + y^2}{2\sigma^2} \right] e^{-\frac{x^2 + y^2}{2\sigma^2}} \quad (2-53)$$

### Filtro DoG

Este kernel está definido con dos valores de varianza  $\sigma_1$  y  $\sigma_2$ , para obtener nuevos valores, ya sea de  $3x3$ ,  $5x5$  o  $7x7$ . La ecuación, define el funcionamiento de este y la obtención de los nuevos valores.

$$DoG(x, y) = \frac{e^{-\frac{x^2 + y^2}{2\sigma_1^2}}}{\pi\sigma_1^2} - \frac{e^{-\frac{x^2 + y^2}{2\sigma_2^2}}}{\pi\sigma_2^2} \quad (2-54)$$

### Histograma

Veremos la representación gráfica de los datos contenidos en una imagen digital en relación al brillo ya que el histograma es la base de numerosas técnicas de procesamiento de la imagen en el dominio espacial. La manipulación del histograma puede utilizarse eficazmente para el realce de la imagen. Esto nos permite capturar la imagen con la exposición correcta.

Aquí vemos que el histograma representa una progresión de intensidades que empieza en la esquina izquierda con el punto mayor con color negro, que es el brillo y conforme se dirige hacia la esquina derecha el valor de la intensidad va aumentando hasta llegar al punto más



claro, pasando por todas la escala de tonalidades de grises. Un histograma está compuesto por 256 columnas, desde un  $\alpha = 0$  hasta  $\beta = 255$  que representan las diferentes intensidades, este puede dividirse en 5 zonas.

Nos damos cuenta que el histograma de una imagen digital con valores de gris dentro del rango  $[0, 2^L - 1]$ , donde  $L$  son el número de bits con los que está compuesta la imagen, es una función discreta del tipo  $h(r_k) = n_k$ , donde  $r_k$  es el  $k$ -ésimo nivel de gris y  $n_k$  es el número de pixeles de la imagen que tienen el nivel de gris  $r_k$ .

### Normalización del Histograma

La operación de la normalización es realizada sobre una imagen que normalmente está muy opaca o muy pálida, es ahí donde la normalización aumenta el brillo volviendo aquellos valores más bajos en negro o 0 y aclarando los más cercanos al 255, sin alterar las demás intensidades de la imagen, permitiendo de esta forma que el contraste de la imagen aumente. La forma de obtener la normalización de un histograma es mediante la ecuación.

$$N_{xy} = \frac{O_{xy}}{MxN} \quad (2-55)$$

Donde:

$N_{xy}$  = El nuevo pixel de la nueva imagen.

$O_{xy}$  = Pixel de la imagen anterior a manejar.

$MxN$  = Resolución total de la imagen.

### Ecualización del Histograma

El ecualizar una imagen consiste en obtener todas las intensidades de una manera más uniforme, convirtiendo los valores de las intensidades de la imagen en otra equivalencia modificando la anterior intensidad del pixel por otra más congruente. A continuación se van a mostrar y listar algunas de las diferentes técnicas para ecualizar una imagen y obtener su respectivo histograma.

#### Ecualización Uniforme

La ecualización uniforme está definida por la ecuación.

$$S_k = (r_{\text{máx}} - r_{\text{mín}}) \sum_{j=0}^k P(r_j) + r_{\text{mín}} \quad (2-56)$$

Donde:

$S_k$  = Nuevo valor del pixel.

$r_{\text{máx}}$  = El valor máximo de intensidades que permite la imagen según el número de bits a los que este.

$r_{\text{mín}}$  = Valor mínimo, generalmente 0.

$P(r_j)$  = Probabilidad de la aparición de un pixel a lo largo de la imagen.

### **Ecualización Exponencial**

La ecualización exponencial está definida por la ecuación

$$S_k = r_{\text{mín}} - \frac{1}{\alpha} \ln \left[ 1 - \sum_{j=0}^k P(r_j) \right] \quad (2-57)$$
$$\alpha \in [0, 1]$$

Donde:

$S_k$  = Nuevo valor del pixel.

$r_{\text{máx}}$  = El valor máximo de intensidades que permite la imagen según el número de bits a los que este.

$r_{\text{mín}}$  = Valor mínimo, generalmente 0.

$P(r_j)$  = Probabilidad de la aparición de un pixel a lo largo de la imagen.

### **Ecualización Rayleigh**

La ecualización Rayleigh está definida por la ecuación.

$$S_k = r_{\text{mín}} + \left[ 2\alpha^2 * \ln \left[ \frac{1}{1 - \sum_{j=0}^k P(r_j)} \right] \right]^{\frac{1}{2}} \quad (2-58)$$
$$\alpha \in [0, 1]$$

Donde:

$S_k$  = Nuevo valor del pixel.

$r_{\text{máx}}$  = El valor máximo de intensidades que permite la imagen según el número de bits a los que este.

$r_{\text{mín}}$  = Valor mínimo, generalmente 0.

$P(r_j)$  = Probabilidad de la aparición de un pixel a lo largo de la imagen.

### **Ecuación Hiperlogaritmica**

La ecuación hiperlogaritmica está definida por la ecuación.

$$S_k = \left[ (\sqrt[3]{r_{\text{máx}}} - \sqrt[3]{r_{\text{mín}}}) * \left( \sum_{j=0}^k P(r_j) \right) + \sqrt[3]{r_{\text{mín}}} \right]^3 \quad (2-59)$$

Donde:

$S_k$  = Nuevo valor del pixel.

$r_{\text{máx}}$  = El valor máximo de intensidades que permite la imagen según el número de bits a los que este.

$r_{\text{mín}}$  = Valor mínimo, generalmente 0.

$P(r_j)$  = Probabilidad de la aparición de un pixel a lo largo de la imagen.

### **Ecuación Hiperlogaritmica**

La ecuación hiperlogaritmica está definida por la ecuación.

$$S_k = r_{\text{mín}} \left( \frac{r_{\text{máx}}}{r_{\text{mín}}} \right)^{\sum_{j=0}^k P(r_j)} \quad (2-60)$$

Donde:

$S_k$  = Nuevo valor del pixel.

$r_{\text{máx}}$  = El valor máximo de intensidades que permite la imagen según el número de bits a los que este.

$r_{\text{mín}}$  = Valor mínimo, generalmente 0.

$P(r_j)$  = Probabilidad de la aparición de un pixel a lo largo de la imagen.

## **Expansión, Contracción y Desplazamiento del Histograma**

### **Expansión del Histograma**

La expansión del histograma suele aplicarse a imágenes con bajo contraste para que con esta operación aumente hasta un rango deseado de grises o en su defecto cubra un rango total posible de intensidades, la ecuación define la expansión.

$$N_{xy} = \left[ \frac{O_{xy} - O_{\text{mín}}}{O_{\text{máx}} - O_{\text{mín}}} \right] * (N_{\text{máx}} - N_{\text{mín}}) + N_{\text{mín}} \quad (2-61)$$

Donde:

$N_{xy}$  = Es el nuevo valor en la nueva imagen.

$N_{\text{máx}}$  = Nuevo valor máximo de expansión deseado.

$N_{\text{mín}}$  = Nuevo valor mínimo de expansión deseado.

$O_{xy}$  = pixel de la imagen original con el que se está trabajando.

### **Contracción del Histograma**

Los histogramas permites visualizar la probabilidad de las intensidades que existen dentro de una imagen, refiriéndose a esto como el número de veces que estas tienen su aparición dentro de la misma imagen, generando de esta forma una gráfica que va de un valor 0 hasta 255 y que contiene la cantidad de veces que hace presencia una intensidad en un pixel. La contracción del histograma nos permite que se acote el rango de inicio y fin de las intensidades que se manejan normalmente, es decir, que se tengan desde un valor  $\alpha$  hasta un  $\beta$ , sacando sus equivalencias dentro del mismo rango y no que sea de 0 al 255 como comúnmente se maneja. Para ello se hace uso de la ecuación:

$$N_{xy} = \left[ \frac{N_{\text{máx}} - N_{\text{mín}}}{O_{\text{máx}} - O_{\text{mín}}} \right] * (O_{xy} - O_{\text{mín}}) + N_{\text{mín}} \quad (2-62)$$

Donde:

$N_{xy}$  = Es el nuevo valor en la nueva imagen.

$N_{\text{máx}}$  = Nuevo valor máximo de compresión deseado.

$N_{\text{mín}}$  = Nuevo valor mínimo de compresión deseado.

$O_{xy}$  = pixel de la imagen original con el que se está trabajando.

### **Desplazamiento del Histograma**

Con esta técnica la imagen cambia sus intensidades originales a las nuevas equivalencias, oscureciendo la nueva imagen creada. Para la operación de desplazamiento, es definida mediante

la ecuación.

$$N_{xy} = O_{xy} + \alpha \quad (2-63)$$

Con esto podemos mover el conjunto nuevo de intensidades mostradas en el histograma a lo largo del mismo, notando así un cambio en la imagen ya sea oscureciéndola o dándole más brillo al tener este nuevo conjunto en un nuevo espacio.

### **Expansión y Modificación del Contraste.**

Existen diversas técnicas para la expansión de contraste, en el siguiente código podemos ver técnicas como el contraste es decir intensificar los colores de algún modo, la iluminación, el contraste e iluminación, segmentación por frontera y adaptación por contraste, los cuales son un ejemplo de cómo a una imagen se le puede brindar más brillo y contraste distinguible en un fondo cuando se alcance un umbral de contraste. A diferencia de los operadores puntuales, en los operadores grupales se implementa la siguiente función denotada por la ecuación:

$$\begin{aligned} f(O_{xy}) &= O_{xy} * C & (2-64) \\ 0 &\leq C < 2 \end{aligned}$$

## Capítulo 3

# Sistemas de reconocimiento facial

En éste Capítulo se describen los tres métodos más importantes utilizados en reconocimiento facial y se describen las principales variaciones y mejoras de éstos algoritmos en el estado del arte actual.

### 3.1. Análisis de Componentes Principales (PCA)

Análisis de Componentes Principales (PCA), es una técnica tradicional de proyección sobre un subespacio utilizadas para reconocimiento de caras, estadísticas de síntesis de la información, o reducción de la dimensión (números de variables). Es decir, ante un banco de datos con muchas variables el objeto es reducirlas a un menor número perdiendo la menor cantidad de información posible. Los nuevos componentes principales serán una combinación lineal de las variables originales, y además serán independientes entre si. Un objetivo principal es reducir la dimensión, ya que si tenemos muchos componentes de una imagen o muchas imágenes, es en parte complicado comparar todos los componentes, al reducir los componentes se obtienen los componentes más importantes, podremos simplificar sus componentes de cada imagen, y así realizar más rápido la comparativa o búsqueda, ya que teniendo muchas imágenes será más fácil comparar menor cantidad de componentes.

Esta técnica utilizada en reconocimiento facial consiste en encontrar transformaciones ortogonales de las variables originales, es decir extraer de un conjunto de imágenes que contienen rostros de personas, donde se realiza una extracción de características para encontrar un sub-

espacio que maximice la varianza del espacio original. Para obtener los vectores se realizan los cálculos denominados *Eigenfaces*, que se basa en la matriz de covarianza de las imágenes originales. Con la finalidad de obtener los elementos que más sobre-salen y son suficientes para poder describir la imagen y reducir la dimensión de los elementos originales.

Una imagen puede ser expresada solo por un vector, el cual contiene los componentes principales de ahí el nombre de esta técnica, Esta técnica fue principalmente desarrollada para el campo de la estadística, pero con el tiempo se descubrió que puede ser utilizada en otros campos de investigación. Esta técnica extrae los rasgos principales y característicos, y reduce la información necesaria para identificar la variable que se está analizando, y todos estos rasgos principales son convertidos en un vector. Con este vector se crea la *Eigenface*, la cual es la representación del vector que contiene los rasgos principales. Al aplicar un algoritmo para el tratamiento de imágenes faciales, esto resulta ser ampliamente útil y con resultados muy concisos ya que son sencillos de utilizar y aplicar, así mismo con dinámicos dentro de su misma aplicación. La razón del amplio uso de este algoritmo es que tras aplicarse un cierto número de veces sobre un conjunto de imágenes que se tienen, se comienzan a apreciar de una mejor manera las características que se desean analizar y poder obtener la información deseada. Dentro del reconocimiento facial uno de los puntos que más resalta son parte de los ojos, nariz y la boca, los cuales son características analizadas comúnmente tanto por distancia y espacio como por tamaño.

**Lopez V (2009)** López nos habla de realizar un análisis de componentes principales en un conjunto de datos de  $n$  características, el sistema elimina datos repetitivos, en donde la pérdida de la información sea mínima. Las componentes principales son la menor cantidad de combinaciones de las variables originales, mientras que obtienen la mayor cantidad de variabilidad. Las componentes obtenidas puedan representar por completo todo el conjunto de datos. Los objetos pueden tener  $n$  características, y por lo tanto tienen  $n$  dimensiones. Cuando tenemos 2 características es fácil realizar el análisis e identificarla, cuando se va aumentando el número de características, realizar el análisis lleva más tiempo y es más tedioso. En el análisis de componentes principales se encarga de recoger la mayor variabilidad y obtener la mayor precisión. Esto lo hace mediante vectores que recorren el conjunto de datos y obtienen los componentes, buscando distancia entre puntos y grupos de similitud, para representar la dirección que tiene

el vector, al momento de representar sea lo más exacta a la original. En donde la representación es representada por el cuadrado de las variables de entrada y las componentes del vector.

Varios investigadores han realizado significativos avances al introducir algoritmos de reconocimiento facial utilizando PCA.

**Sung y Poggio (1998)** Sung y Poggio nos presentan un enfoque “*Based Learning*”, en este estudio los rostros de las personas son enfocados en vistas frontales y verticales, en escenas difíciles. Un modelo basado en la distribución de los patrones de la cara y un conjunto de parámetros de distancia.

**Turk y Pentland (2000)**, Turk y Pentland desarrollaron el método de *Eigenfaces*, el cual es un método importante, ya que la extracción de características discriminatorias de un conjunto de entrenamiento por etiquetado. Este método consiste en convertir los rostros a escala de grises, el rostro original lo transforma en vectores a parti de ellos se obtiene las componentes principales, y al reconstruir el rostro con estos vectores, se obtiene el mayor parecido al rostro original. Al trabajar con 40 Eigenfaces son suficientes para trabajar obtener una buena descripción del rostro de la persona ya que al reconstruir la imagen, esta tiene un mínimo de errores. **O T Toole (1993)**. Toole demostró la importancia y la eficiencia de trabajar con *Eigenfaces* con pequeños y grandes valores en los *Eigenvectores*, en sus discusiones argumenta que con esta técnica se pueden clasificar personas de diferentes sexo y raza.

### 3.1.1. Clasificación de caras Mediante Análisis de Componentes Principales

Sea una imagen bidimensional  $I(x, y)$  un arreglo de  $N \times N$  con diferentes intensidades. Una imagen puede también ser considerada como un vector de dimensión  $N^2$ , de tal forma que una imagen típica de  $128 \times 128$  es representada por un vector de 16384.

La idea principal del algoritmo de Análisis de Componentes Principales es encontrar los vectores que más influyen en la distribución de imágenes de caras en el espacio de imágenes entero. Estos vectores definen el subespacio de imágenes de caras, que es llamado "espacio de caras". Cada vector es de longitud  $N^2$  y describe una imagen de  $N \times N$  y es una combinación lineal de las imágenes de caras originales. Debido a que estos vectores son los eigenvectores de la matriz de covarianza de las imágenes originales y debido a que estos tienen la apariencia de cara, se refiere a ellos como eigen-caras.



Sea el conjunto de imágenes de caras  $\Gamma_1, \Gamma_2, \dots, \Gamma_m$ . La cara promedio de este conjunto es definida como:

$$\Psi = \frac{1}{m} \sum_{n=1}^m \Gamma_n \quad (3-1)$$

. La diferencia de cada una con respecto al promedio esta dada por el vector  $\Phi_i = \Gamma_i - \Psi$ . De este conjunto de vectores son obtenidas las componentes principales. Las componentes principales son el conjunto de  $m$  vectores ortonormales  $\mathbf{u}_n$  que mejor describe la distribución de los datos. El  $k$ -ésimo vector  $\mathbf{u}_k$  es obtenido de tal forma que:

$$\lambda_k = \frac{1}{m} \sum_{n=1}^m (\mathbf{u}_k^T \Phi_n)^2$$

es un máximo sujeto a

$$\mathbf{u}_l^T \mathbf{u}_k = \delta_{lk} = \begin{cases} 1, & \text{si } l = k \\ 0, & \text{de otra forma} \end{cases}$$

Los vectores  $\mathbf{u}_k$  y escalares  $\lambda_k$  son los eigenvectores y eigenvalores de la matriz de covarianza respectivamente.

$$\begin{aligned} C &= \frac{1}{m} \sum_{n=1}^m \Phi_n \Phi_n^T \\ &= AA^T \end{aligned}$$

donde la matriz  $A = [\Phi_1, \Phi_2, \dots, \Phi_m]$ . La matriz  $C$ , sin embargo es de  $N^2 \times N^2$  y determinar los  $N^2$  eigenvectores y eigenvalores es una tarea computacionalmente compleja para imágenes de tamaño normal, por lo que son necesarios métodos factibles para encontrar estos eigenvectores. Si el número de puntos en el espacio de imágenes es menor que la dimensión del espacio ( $M < N^2$ ), entonces serán  $m - 1$  eigenvectores significativos en lugar de  $N^2$ . Afortunadamente es posible resolver los eigenvectores de la matriz  $N^2$  al resolver los eigenvectores de la matriz  $m \times m$ , resolviendo una matriz de  $16 \times 16$  en lugar de una matriz de  $16384 \times 16384$ . Considerando

que los eigenvectores  $\mathbf{v}_i$  de  $A^T A$  tal que:

$$A^T A \mathbf{v}_i = u_i \mathbf{v}_i \quad (3-2)$$

premultiplicando ambos lados por  $A$  tenemos:

$$A A^T A \mathbf{v}_i = u_i A \mathbf{v}_i$$

de donde podemos ver que  $A \mathbf{v}_i$  son los eigenvectores de  $C = A A^T$ .

Siguiendo este análisis, construimos una matriz de  $m \times m$ .  $L = A^T A$ , donde  $L_{mn} = \Phi_m^T \Phi_n$ , y encontramos los  $m$  eigenvectores de  $\mathbf{v}_l$  de  $L$ . Estos vectores determinan la combinación lineal de  $m$  imágenes del conjunto de entrenamiento para formar las "eigen caras"  $\mathbf{u}_l$ .

$$\mathbf{u}_l = \sum_{n=1}^m \mathbf{v}_{ln} \Phi_n, l = 1, \dots, m \quad (3-3)$$

Estas eigen-caras pueden ser utilizadas para identificar una nueva imagen de una cara ( $\Gamma$ ) que es transformada en sus componentes al proyectarla dentro del espacio de caras mediante una simple operación:

$$w_k = \mathbf{u}_k^T (\Gamma - \Psi) \quad (3-4)$$

para  $k = 1, \dots, m$ . Los pesos forman un vector  $\Omega^T = [w_1, w_2, \dots, w_m]$  que describen la contribución de cada eigen-cara en representar la imagen de la cara de entrada, tratando las eigen-caras como un conjunto de imágenes de caras. El vector puede ser usado en un algoritmo de reconocimiento de patrones para encontrar cual es el número de clases de caras predefinido. El método más simple para determinar cual clase provee la mejor descripción de una imagen de entrada es encontrar la clase  $k$  que minimiza la distancia euclidiana:

$$\epsilon_k^2 = \|(\Omega - \Omega_k)\|^2 \quad (3-5)$$

donde  $\Omega_k$  es el vector que describe la  $k$ -ésima clase. Las clases  $\Omega_i$  son calculadas promediando los resultados de los eigen-caras sobre un pequeño número de imágenes de caras de cada individuo.

Una imagen es clasificada como perteneciente a una clase  $k$  cuando el mínimo  $\epsilon_k^2$  es menor a alguna frontera elegida  $\theta_\epsilon$ . De otra manera, la imagen es clasificada como desconocida y opcionalmente utilizada para crear una nueva clase.

### 3.2. Análisis Discriminante Lineal (LDA)

Esta técnica es una mejora de la técnica de PCA, la cual analiza la información, con el objetivo de reducir la dimensión de las características, aplicando LDA, donde la matriz de proyección sea máxima, la diferencia que tiene LDA con PCA, es que arroja mejores resultados en condiciones de iluminación y gestos.

Otro método es Análisis de Componentes Principales LDA (*Linear Discriminant Analysis*), es una aproximación estadística que maximiza la varianza entre usuarios y minimiza la varianza de cada usuario. Es una técnica de aprendizaje supervisado para clasificar datos. La idea central de LDA es obtener una proyección de los datos en un espacio de menor (o incluso mayor) dimensión que los datos entrantes, con el fin de la separabilidad sea la mejor posible.

Utiliza una aproximación estadística para diferenciar las muestras conocidas de los que no lo son. Sean  $x_i$  y  $s_i$  los vectores de medias y las matrices de covarianza de cada uno de los grupos  $g$  y sea  $x$  el vector de medias global del conjunto de entrenamiento.

El análisis Discriminante es una técnica multivariante cuya finalidad es analizar si existen diferencias significativas entre grupos respecto a un conjunto de variables sobre lo mismo, para el caso en que existan. Se considera como un análisis de regresiones donde la variable dependiente es categórica y tiene como categorías la etiqueta de cada uno de los grupos, mientras que las variables independientes son continuas y determinan a qué grupos pertenecen.

[23] En el trabajo GMM y LDA en la detección de enfermedades pulmonares, presentan metodologías para la detección de enfermedades en el pulmón se realizaron experimentos utilizando señales normales y anormales de la respiración. Las señales fueron tomadas de personas sanas y enfermas, el cual dio un 98 % de eficiencia. En este trabajo el LDA se divide en 2 clases que son las señales normales y anormales, donde se tienen  $x$  vectores, se busca un vector de peso  $w$  y el vector umbral  $w_0$ , con el fin de construir modelos, donde se tuvieron 3 clasificadores discriminantes, y el sistema identificaba si la señal era normal y terminaba, pero si era anormal

el sistema realizaba otro análisis para obtener que tipo de señal era y poder identificar que enfermedad era.

[67] Hau yu, Jie Yang, utiliza 2 técnicas para el reconocimiento facial: PCA y LDA. Comentan que el algoritmo PCA al realizar su procedimiento puede destacar información discriminante, a diferencia del algoritmo LDA. Nos propone utilizar LDA en un gran conjunto de imágenes, sin modificar la dimensión de las imágenes y extracción de características, aplicando directamente el criterio de Fisher, que consiste en tratar de encontrar una matriz simultánea diagonalizable. Es decir, una matriz con elementos de la diagonal ordenados decrementalmente. En su artículo, los autores mencionan el problema de trabajar con el algoritmo LDA al utilizar imágenes de grandes dimensiones y al trabajar con varias imágenes la complejidad del proceso es mayor.

[68] Changjun et al., nos hablan del reconocimiento facial, aplicando 2 algoritmos: PCA y LDA. Los autores utilizan una SVM para el reconocimiento facial. Nos plantea que el algoritmo PCA obtiene la varianza máxima de las características, y el error es mínimo al reconstruir el rostro. El algoritmo LDA reduce la dimensión en el reconocimiento de patrones, para encontrar una transformación lineal, de tal manera que las agrupaciones de entidades sean más separables entre sí. Al trabajar con PCA y LDA, se obtienen buenos resultados, aplicando el clasificador SVM.

[69] Vinay. A y compañía nos hablan del filtro de Gabor, que es una derivada del Gaussiano centrado en las frecuencias de la función sinusoidal. Tenemos 2 algoritmos que son excelentes para reducir la dimensionalidad del espacio de características. Otro punto importante es que las técnicas de LDA y PCA no se pueden adecuar a las condiciones que enfrentan el reconocimiento facial en ciertos escenarios, las imágenes contienen características como las curvas de los bordes, los píxeles alrededor de los bordes. Los autores proponen dos técnicas LDA y KDA, para medir su eficiencia, y obtener de las 2 técnicas la que mejor se desempeñe en el reconocimiento facial. Se construyeron un banco de 40 filtros (8 de orientación y 5 escalas), para obtener las características faciales, al aplicar los filtros de Gabor con LDA, la exactitud de los resultados fueron de un 93.33 %, y al utilizar los filtros de Gabor KDA se obtuvo una precisión del 95 %.

### 3.2.1. Clasificación de caras Mediante Análisis de Discriminante Lineal

Este algoritmo lleva el subespacio de rostros a una dimensión menor en donde aumenta la separabilidad de las clases. El objetivo de éste algoritmo es encontrar un sub-espacio que obtenga los discriminantes de las diferentes clases. En donde se calcula la matriz de dispersión entre las clases que son distintas (intergrupales) y la matriz de la misma clase (intragrupal).

Primero, el algoritmo necesita un conjunto de entrenamiento compuesto de un grupo relativamente grande de sujetos con diversas características faciales. La selección apropiada del conjunto de entrenamiento validará los resultados finales. La base de datos deberá contar con varios ejemplos de imágenes faciales de cada sujeto en el conjunto de entrenamiento y al menos un ejemplo en el conjunto de prueba. Estos ejemplos deberán presentar diversas vistas con variaciones de ángulo menores, incluir diversas expresiones faciales, iluminación y condiciones de fondo, así como presentar ejemplos con y sin lentes. Se asume que las imágenes ya han sido normalizadas en arreglos de  $m \times n$  y que estas contienen solo regiones donde se presenta la cara y no el cuerpo de la persona.

Sea una imagen  $I(x, y)$  bidimensional de tamaño  $N^2$ . Este vector corresponde a la representación inicial de la cara. El conjunto de todas las caras en el espacio de características es tratado como un espacio vectorial altamente dimensional. Este vector corresponde a la representación inicial de la cara.

Definiendo todos los ejemplos de la misma persona como pertenecientes a una clase y los rostros de las demás personas como pertenecientes a otras clases, para todos los sujetos en el conjunto de datos de entrenamiento, establecemos una estructura para realizar un análisis de separación de clusters en el espacio de características. También teniendo etiquetadas todos los ejemplos en el conjunto de entrenamiento y teniendo definidas todas las clases, entonces calculamos la matriz de dispersión intra-clases y matriz de dispersión entre clases

$$S_w^{(v)} = \sum_{i=1}^L \Pr(C_i) \sum i \quad (3-6)$$

$$S_b^{(v)} = \sum_{i=1}^L \Pr(C_i) (\mu - \mu_i)(\mu - \mu_i)^T$$

aquí  $S_w$  es la matriz de dispersión intra-clases,  $\sum i$  es el promedio de dispersión de los vectores

(V) de diferentes clases  $C_i$  alrededor de su media:

$$\sum_i i = E[(V - \mu_i) \times (V - \mu_i)^T | C = C_i] \quad (3-7)$$

Similarmente,  $S_b$  es la matriz de dispersión entre-clases,  $\Pr(C_i)$  es la probabilidad de la  $i$ -ésima clase. El poder discriminatorio de una representación puede ser cuantificado usando varias medidas. La matriz de separación de clases y la medida de separabilidad pueden ser calculadas como:

$$S^{(V)} = S_w^{-1} S_b$$

$$J_V = sep(V) = trace(S^{(V)})$$

La separabilidad entre clases puede ser medida, comunmente es el siguiente:

$$J(A) = \arg \max_A \frac{|AS_b A^T|}{|AS_w A^T|}$$

donde  $A$  es una matriz de  $m \times n$  con  $m \leq n$ . La ecuación anterior se resuelve con

$$S_b A^* = \lambda S_w A^*$$

la idea básica del algoritmo es encontrar una matriz  $A$  que puedan ser simultáneamente diagonalizables  $S_w$  y  $S_b$ . *i.e.*

$$AS_w A^T = I$$

$$AS_b A^T = \Delta$$

donde  $\Delta$  es una matriz diagonal con elementos sobre la diagonal ordenados de forma decreciente. Si queremos reducir la dimensión de la matriz, simplemente utilizamos las primeras filas de  $A$ , que corresponde a los eigenvalores más grandes de  $\Delta$ .

### 3.3. Patrón Biario Local (LBP)

El algoritmo Patrón Biario Local (LBP, por sus siglas en ingles *Local Binary Partten*) es un operador en la descripción de textura, donde la textura de una imagen puede proporcionar información de los contornos que se encuentran en la imagen ya sean objetos, personas, entre otras cosas. A diferencia de la percepción del ojo humano que podemos distinguir directamente los contornos y colores de los objetos. Este algoritmo puede repetir seguidamente el mismo patrón, u obtener un conjunto de valores con pequeñas variaciones.

La textura de una imagen nos da mucha información acerca del contorno de los objetos presentes en una imagen.

El algoritmo es una técnica desarrollada para trabajar con imágenes monoespectrales, en su primera versión se trabajaba con los 8 vecinos más cercanos, el resultado es una imagen donde cada nuevo valor depende de los valores vecinos que están a su alrededor, resaltando los bordes que existen dentro la imagen. Este algoritmo es modificado para resaltar y mejorar los bordes de las figuras que se encuentran dentro de la imagen, lo que permite obtener una mejor descripción de textura.

[70] LOU Yuan y compañía, tiene el objetivo de reconocer las expresiones faciales del rostro, para lograr este objetivo, proponen obtener un método híbrido entre el algoritmo PCA y LBP. Del algoritmo PCA fue implementado para extraer las características en escala de grises de la imagen del rostro con la expresión facial de los ojos. Mientras que el algoritmo LBP extrae las características de textura de la parte de la boca. Y utilizaron la SVM para el reconocimiento, al aplicar esta técnica híbrida del PCA y LBP es capaz de clasificar de mejor manera los rostros con expresiones faciales. Separando los ojos de la boca, y a cada parte del rostro se aplica una técnica diferente, la cual tiene mejores resultados en el área aplicada.

[71] En este trabajo Bodgan y Karolina, se enfocan en utilizar el algoritmo LBP. Los autores tomaron imágenes que contenían rostros faciales, dividido en dos partes: una parte de rostros sin alguna expresión y la otra con una expresión sonriente. Se aplicaron algunos filtros de suavizado cuando se tenían bordes gruesos, ya que los histogramas obtenidos variaban por causa de los bordes en los rostros. Se dividió cada imagen en subpartes para aplicar el algoritmo LBP tanto las imágenes que contenían los rostros sin expresión y los rostros que obtenían una sonrisa,

después se concatenaron los histogramas resultantes con el objetivo de construir un vector característico final. Así se obtenía la definición de cada parte del rostro, y poder identificar las diferencias que existían entre las imágenes sin expresión alguna con las que contenían la expresión de una sonrisa. Todos estos procedimientos con el objetivo de comparar la eficiencia de diferentes algoritmos basados en *LBP*.

[66] Helei y compañía nos hablan de un sistema de reconocimiento de Caras en plataforma *Symbian*, con el algoritmo LBP, y AdaBoost, utilizando la tecnología de un celular, por lo que las características para trabajar un algoritmo son limitadas, el algoritmo propuesto obtiene resultados de precisión positivos y en tiempos aceptables normalizando la iluminación de la imagen, y al utilizar, técnicas para detectar los ojos de la cara.

[72] BO Yang y Songcan Chen, nos hablan del algoritmo LBP, el cual consiste en dividir el rostro en varias sub-partes, y realizar el procedimiento adecuado a cada sub-parte, después obtener su histograma y concatenar los histogramas en un solo histograma para su reconocimiento facial, nos dicen de la técnica derivada del LBP. El histograma LBP espacialmente mejorado, por sus siglas en inglés (eLBPH), es una técnica con mejores que el algoritmo LBP, a pesar de seguir el mismo procedimiento, ya que el LBP necesita un pre-procesamiento antes de aplicar el algoritmo, el eLBPH obtiene histogramas más uniformes, es decir, más estable que el histograma obtenido con el LBP, comparan también los algoritmos hLBPI, el histograma LBP holístico ingenuo por sus siglas en inglés (hLBPH), por lo que proponen que estos algoritmos dan mejores resultados, en el campo de Reconocimiento Facial, y orientan a su aplicación en casos reales.

### **3.3.1. Clasificación de caras Mediante el algoritmo LBP**

El primer paso de este algoritmo es dividir la imagen en  $n$  partes, tomando en cuenta que mientras  $n$  sea mejor la precisión será mayor, al igual que el tiempo que llevara a realizar este proceso. Es recomendable dividir la imagen en 16 partes ya que es una buena medida en el tiempo que realizará este proceso y los resultados son aceptables. Los pasos siguientes se realizarán a cada una de las partes divididas anteriormente.

Este algoritmo trabaja con una máscara de  $3 \times 3$  respecto al píxel al píxel que se encuentra en el centro de la máscara. Aplicando la siguiente fórmula:



1;  $p_v \geq p_x$

0;  $p_v < p_x$

Donde:

$p_v$  es el pixel que se está analizando

$p_x$  es el pixel que está en el centro de la máscara

6	5	1		1	1	0
7	4	0	→	1		0
4	3	2		1	0	0

Después la matriz resultante se multiplica por la siguiente matriz.

1	2	4
128		8
64	32	16

Para obtener el nuevo pixel de la nueva imagen. Cada imagen del conjunto de caras puede considerarse como una composición de micro-patrones que se puede detectar con eficacia por el operador LBP. Para tener en cuenta la información de la forma de las caras, se reparten las imágenes de caras en  $M$  pequeñas regiones que no se solapan.

Los histogramas de LBP extraídos de cada subregión se concatenan en una sola, el histograma de características espacialmente mejorado se define como:

$$H_{ij} = \sum_{x,y} I(f(x,y) = i)I((x,y) \in R_j)$$

donde  $i = 0, \dots, l - 1, j = 0, \dots, m - 1$ . El histograma de características extraído describe la textura local y forma global de las caras en las imágenes.

## Capítulo 4

# Metodología

Un sistema de reconocimiento facial es un conjunto de algoritmos programados que se encargan de reconocer patrones en una imagen o videos que contienen a individuos, con la finalidad de reconocer a las personas sin la intervención de un experto humano, minimizando el tiempo de proceso, y maximizando la exactitud de los resultados. Los cuales se basan en características como lo son las manos, el rostro, la iris, la retina, la conducta, entro otros más. En este Capítulo se muestra la metodología llevada a cabo al implementar los algoritmos de reconocimiento facial.

### 4.1. Problemas al trabajar con un Sistema de Reconocimiento Facial

Existen problemas que se presentan para obtener la imagen adecuada en tiempo real y poder realizar el procesamiento, ya que al momento de obtener la imagen esta puede salir distorsionada debido a que el individuo se movió durante la captura, la imagen también llega a salir muy oscura o muy clara, y no se puede obtener el rostro claramente y se tiene que tomar otra foto, hasta tener la adecuada, este es un problema ya que cuando se realiza la captura de imagen, en ciertas áreas la persona no sabe que ésta siendo grabada o se ésta obteniendo una imagen del individuo, una de las soluciones es tomar una ráfaga de fotos en el mismo momento, para así poder seleccionar la mejor imagen con la que se va a trabajar.

Otros problemas que tenemos con las personas y los sistemas, tambien existen las variaciones

en la imagen que pueden ser clasificadas en agresivas y no agresivas. Estos son cambios que tiene un individuo en un periodo de tiempo ya sea largo o corto, dependiendo del tipo de variaciones. Las variaciones no agresivas son características que llevan tiempo en aparecer y no representan tanto problema ya que no son cambios repentinos, como el envejecimiento, el crecimiento de la persona, entre otras. Las variaciones agresivas, son características de un individuo que cambian de un momento para otro, estos son cambios bruscos en un periodo de tiempo corto en la cara de una persona debido a, cirugías, cambios en el pelo, entre otros.

Para el problema de la iluminación existen algunos métodos para mejorar la imagen y así obtener una mejor imagen para poder trabajar con ella. Otro problema que tenemos al obtener la imagen, es que el individuo lleve puesto accesorios, como son lentes, sombreros, entre otros, incluso, que en el momento que se está tomando la foto, el individuo realice gestos, por lo que, se tienen que incluir dentro de la base de datos.

Algunas de las ventajas de los sistema de reconocimiento facial son: trabajan en tiempo real, lo que significa que el individuo en algunos lugares no sabe que está siendo identificado, y esto representa una gran ventaja en sectores de seguridad, incluso en empresas, significa un ahorro de tiempo en realizar estos procesos, también el individuo no tiene detenerse para poder obtener una imagen de él, sino que se coloca la cámara en un lugar específico, y en el momento que la persona aparece en el rango de la cámara esta toma una o varias imágenes del individuo, sin que este se detenga. La detección del rostro, es el proceso que se encarga de encontrar dentro de la imagen si existe o un rostro de una persona que se va analizar, esto es eliminar el fondo, el cabello, el ruido, toda esa parte con la que no trabajaremos, y solo obtener la parte del rostro que se va a procesar para su identificación.

Para poder detectar el rostro existen varias técnicas como: detectores de bordes, segmentación, que separan las diferentes intensidades y se guían de un rango de intensidades que engloban las tonalidades de la piel, para ello, quitan todo lo que se encuentre alrededor del rostro, para solo dejar el rostro del individuo, y aplicar los algoritmos de identificación del individuo. En este fase se utilizan las técnicas de segmentación, que consisten en aislar el rostro del resto de la imagen. Este proceso puede ser realizado separando las intensidades y basándose en tonalidades del color de piel, comparando las tonalidades que se encuentran dentro de la imagen y obteniendo la tonalidad del rostro para poder aislarla del resto de la imagen.

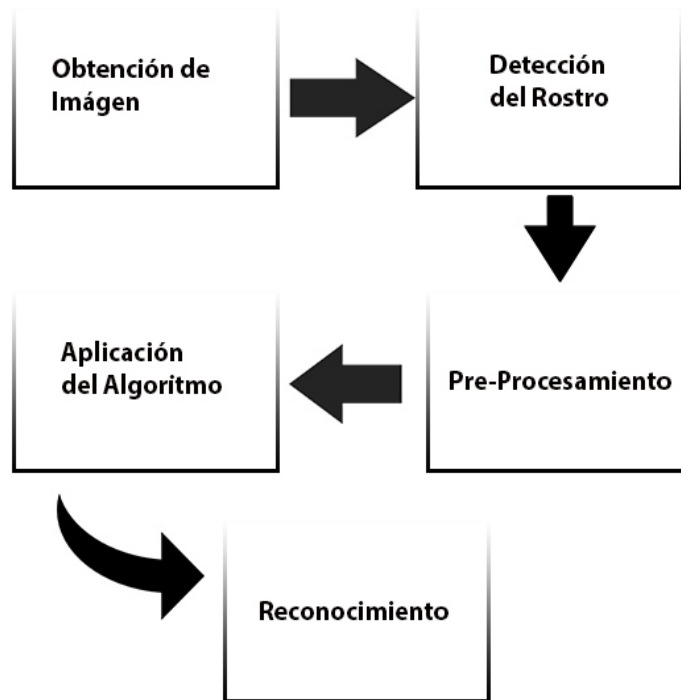


Figura 4-1: Etapas del Reconocimiento Facial

Por último tenemos la identificación, esta parte se encarga de comparar la imagen de una persona con imágenes en la base de datos. algunas técnicas de reconocimiento facial utilizan distancias que existen entre los ojos, la nariz, la boca, ya que el conjunto de todas estas distancias son únicas en cada persona, y diferentes con respecto a los demás. Estos datos obtenidos son comparados con las que se encuentran en la base de datos, encontrando la mayor semejanza de similitud en los datos obtenidos, con los que se encuentran guardados.

## 4.2. Etapas de un Sistema Biométrico Facial

De forma general, las etapas del reconocimiento facial pueden ser clasificadas en: obtención de la imagen, detección del rostro o cara de una persona en la imagen tomada, pre-procesamiento, aplicar algoritmo de reconocimiento e identificación, como se muestra en la Figura4-1. .

1. La obtención de la imagen puede ser realizada por medio de un dispositivo, como puede ser una cámara fotográfica o por medio de una video-cámara. La imagen puede no estar centrada en el rostro de la persona, que nos interesa procesar, por lo que se realiza el paso siguiente que es la detección de rostros.
2. Ya que se obtiene la imagen el siguiente paso es la detección del rostro. Este paso consiste en analizar la imagen e identificar en que parte de la imagen se encuentra el rostro con el que vamos a trabajar, al detectar el rostro, se toma en cuenta la parte de los ojos, la nariz y la boca.
3. Después de obtener la imagen y detectar el rostro, el paso siguiente es el pre-procesamiento, En este paso se realizan mejoras a la imagen con el fin de mejorar la precisión en el reconocimiento. Como se mencionó en el capítulo de pre-procesamiento existen muchas operaciones para el mejoramiento de la imagen.
4. Una vez con la imagen procesada, el paso siguiente consiste en aplicar el algoritmo de reconocimiento, existen muchas técnicas que se pueden aplicar como se menciona anteriormente en este trabajo. En nuestro caso, utilizamos las técnicas de LBP, PCA y LDA.
5. El paso siguiente es el reconocimiento, este paso consiste en comparar los resultados con los que tenemos de la base de datos, obteniendo la imagen de la base de datos que más parecido tenga a la entrada.
6. Una vez realizados los pasos anteriores, el paso siguiente la identificación, esto consiste en obtener los datos de la imagen más cercana a la imagen de entrada. Los datos pueden depender de la aplicación que se le dé al sistema, algunos datos pueden ser el nombre, dirección, id de trabajo, entre muchos datos más.

En la Figura 4-1, se muestran los pasos que se llevarán a cabo en este proyecto.

#### **4.2.1. Obtención de la Imagen**

Es necesario que las imágenes utilizadas en el presente trabajo cumplan las siguientes condiciones:

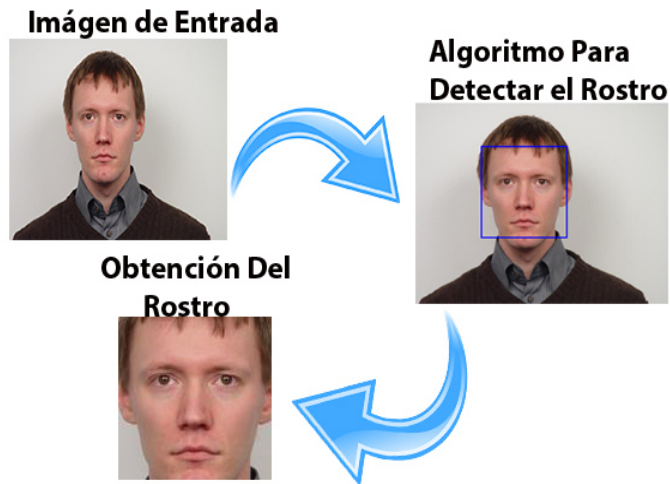


Figura 4-2: En esta imagen se puede apreciar, los pasos a seguir para obtener la parte del rostro, del resto de la imagen.

1. Rostros de frente: Los algoritmos utilizados analizaran las características con las que cuenta un rostro de frente, por lo que es necesario tener imágenes de rostros de frente, ya que el algoritmo que se aplica para la detección del rostro no funcionaria.
2. Tener imágenes de mas de 10 personas: Al tener a más de 10 personas en la base de datos de las imágenes, para tener diferentes tipos de personas, y poder medir la eficiencia de los algoritmos al trabajar con diferentes personas.
3. Tener mas de 5 imágenes de cada persona: Al tener más de 5 imágenes de cada persona, se tendran imágenes para el entrenamiento y para las prueba del sistema.
4. Imágenes claras: al tener imágenes donde el rostro sea claro, podemos aplicar pre-procesamiento para manipular las imágenes, y obtener mejores resultados con los algoritmos.

#### 4.2.2. Detección de Rostro

En este punto teniamos imágenes con resoluciones demasiado grandes y el rostro estaba en el centro de la imagen, por lo que se introducía mucho ruido, en esta imagen se muestra el proceso para la detección de rostro. 4-2.

Se realizó una detección del rostro para eliminar todo ese ruido, y obtener solo la área del

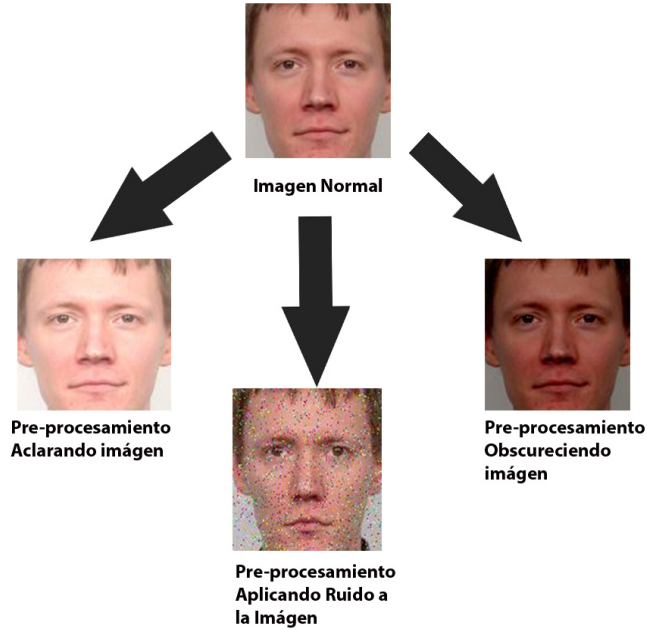


Figura 4-3: Ejemplo de pre-procesamiento aplicado a una imagen.

rostro, donde el tamaño de la nueva imagen que contenía el rostro es de 300 píxeles de ancho por 300 píxeles de alto, teniendo una imagen más pequeña. Una vez hecho todo este proceso a las imágenes, estas quedaban de diferente tamaño, para normalizar su tamaño, se redimensionaron a  $128 \times 128$  píxeles, pues se consideró que era suficiente para trabajar y obtener las características necesarias de las personas.

### 4.2.3. Pre-Procesamiento

En el pre-procesamiento se aplican algoritmos para aumentar el brillo, oscurecer e incluso aumentar ruido a la imagen, como se muestra en la Figura 4-3.

En esta parte, a las 124 imágenes se les aplicaron operaciones de pre-procesamiento, a todas las imágenes se les aplicó un filtro para oscurecer la imagen y obtener imágenes más oscura.

Se tomaron las 124 imágenes a los que se les aplicó el pre-procesamiento de aclarar las imágenes. A las 124 imágenes se les aplicó el pre-procesamiento, para aumentar el ruido a la imagen.

Estos pre-procesamiento se realizaron para analizar el eficiencia de los algoritmos que se

utilizan en esta tesis. como se muestra en la Figura. 4-3, y se obtubieron en total 496 imágenes, divididas en las 4 categorías que se les aplicó el pre-procesamiento.

#### 4.2.4. Aplicación de Algoritmos

En este punto se aplican las 3 técnicas que se utilizan en esta tesis (LDA, LBP y PCA).

##### Algoritmo PCA para reconocimiento facial

Antes de aplicar este algoritmo, las imágenes en el pre-procesamiento, deben tener un tamaño de  $128 \times 128$  pixeles, al ejecutar el algoritmo la imagen se convierte de una matriz de  $128 \times 128$ , a un vector de  $1 \times 16384$ , con el objetivo de obtener los eigenvectores de este vector, y así disminuir la dimensión, obteniendo los componentes principales. Para resumir, el enfoque de eigenfaces para reconocimiento facial involucra los siguientes pasos:

1. Recolectar un conjunto de imágenes características de individuos (conjunto de entrenamiento). Este conjunto deberá incluir un número de imágenes de cada persona, con alguna variación en expresión y luminosidad (Para 4 imágenes de 10 personas,  $m=40$ ).
2. Calcular la matriz L ( $40 \times 40$ ), encontrar sus eigenvectores y eigenvalores y elegir  $m'$  eigenvectores con los eigenvalores más altos asociados
3. Combinar el conjunto de entrenamiento normalizado de imágenes de acuerdo a la ec. 3-3 para obtener los eigencaras de  $\mathbf{u}_k$ .
4. Para cada individuo conocido calcular el vector de clases  $\Omega_k$  promediando el vector de patrones de eigencaras  $\Omega$  utilizando la ecuación 3-5. Calculada a partir de las cuatro imágenes originales de cada individuo.
5. Seleccionar una frontera  $\theta_\epsilon$  que defina la máxima distancia permitible.
6. Para cada nueva imagen a ser identificada, calcular su vector  $\Omega$ , las distancias  $\epsilon_i$  para cada clase conocida y la distancia  $\epsilon$  del espacio de caras. Si la mínima distancia  $\epsilon_k < \theta_\epsilon$  y la distancia  $\epsilon < \theta_\epsilon$  clasificar la imagen de entrada como el individuo asociado con el vector de clases  $\Omega_k$ . Si la mínima distancia  $\epsilon_k > \theta_\epsilon$  pero la distancia  $\epsilon < \theta_\epsilon$ , entonces la



imagen puede ser clasificada como desconocida y opcionalmente utilizada para una nueva clase.

### **Algoritmo LBP para reconocimiento facial**

Para trabajar con este algoritmo la imagen se divide en varias subpartes, mientras más subpartes se divida mejor precisión se obtendrá en los resultados. En este trabajo se dividió en 16 partes iguales, las imágenes de entrada son de  $128 \times 128$  píxeles, esta decisión se tomó para que la técnica tuviera un buen rendimiento en tiempo de ejecución. Dividir la imagen en 16 subpartes, hace que el proceso sea bueno en tiempo de ejecución, y en los resultados, ya que teniendo imágenes de  $128 \times 128$  píxeles no estamos metiendo muchas características al algoritmo. Como se puede observar en la siguiente Figura.

Al realizar los pasos del algoritmo tenemos 16 histogramas por imagen, este procedimiento se le aplica a la imagen que se va analizar, y al conjunto de imágenes de entrenamiento. Una vez obtenido todo este procedimiento, el paso siguiente es comparar los histogramas de cada imagen para obtener los más cercanos a nuestra imagen de entrada. En general, el algoritmo LBP utiliza los siguientes pasos:

1. Se separa el conjunto de entrenamiento en  $k$  clases.
2. Para cada clase se identifican las  $n$  imágenes de entrenamiento.
3. Para cada imagen realizamos  $w \times w$  particiones
4. Para cada partición extraemos las características LBP utilizando los pasos de codificado, mapeo y obtención de los histogramas

### **Algoritmo LDA para reconocimiento facial**

El algoritmo comienza obteniendo imágenes de matrices cuadradas con ancho = largo =  $N$ ,  $m$  se define como el número de imágenes en la base de datos,  $p$  es el número de personas en la base de datos. El enfoque LDA para reconocimiento facial involucra los siguientes pasos:

1. Recolectar un conjunto de imágenes fija de  $p$  individuos (conjunto de entrenamiento). Este conjunto deberá incluir un número de imágenes de cada persona, con alguna variación

en expresión y luminosidad (Para 4 imágenes de 10 personas,  $m=40$ ). Cada imagen es representado por un vector  $\mathbf{a}_i = [a_1^i, a_2^i, \dots, a_{N^2}^i]$ .

- Calculamos el promedio de todos los vectores de caras como:

$$\mu = \frac{1}{p \times m} \begin{bmatrix} a_1^1 + a_1^2 + \dots + a_1^{p \times m} \\ a_2^1 + a_2^2 + \dots + a_2^{p \times m} \\ \vdots \\ a_1^1 + a_2^2, \dots, a_{N^2}^{p \times m} \end{bmatrix}$$

donde  $p$  es el número de individuos o clases y  $m$  es el número de caras por individuo.

- Calculamos las matrices  $S_b$  y  $S_w$  como:

$$S_b = \sum_{i=1}^p N_i (\mu_i - \mu)(\mu_i - \mu)^T \quad (4-1)$$

$$S_w = \sum_{i=1}^p \sum_{a^j \in \mathbf{a}_i} (a^j - \mu_i)(a^j - \mu_i)^T$$

donde  $\mu_i$  representa la media de los ejemplos de entrenamiento pertenecientes a la clase  $i$ ,  $\mu$  representa la media del total de ejemplos.

- Resolver

$$S_b v_i = \lambda_i S_w v_i$$

- Calcular los eigenvectores más significativos y sus correspondientes eigenvalores de  $S_b$  mediante:

$$S_w^{-1} S_b v_i = \lambda_i v_i$$

- Para cada individuo conocido calcular el vector de clases promediando el vector de patrones de eigencaras  $v_i$  utilizando la ecuación 3-5. Calculada a partir de las cuatro imágenes originales de cada individuo.
- Seleccionar una frontera  $\theta_\epsilon$  que defina la máxima distancia permitida.
- Para cada nueva imagen a ser identificada, calcular su vector, las distancias  $\epsilon_i$  para cada

clase conocida y la distancia  $\epsilon$  del espacio de caras. Si la mínima distancia  $\epsilon_k < \theta_\epsilon$  y la distancia  $\epsilon < \theta_\epsilon$  clasificar la imagen de entrada como el individuo asociado con el vector de clases. Si la mínima distancia  $\epsilon_k > \theta_\epsilon$  pero la distancia  $\epsilon < \theta_\epsilon$ , entonces la imagen puede ser clasificada como desconocida y opcionalmente utilizada para una nueva clase.

## Capítulo 5

# Resultados experimentales

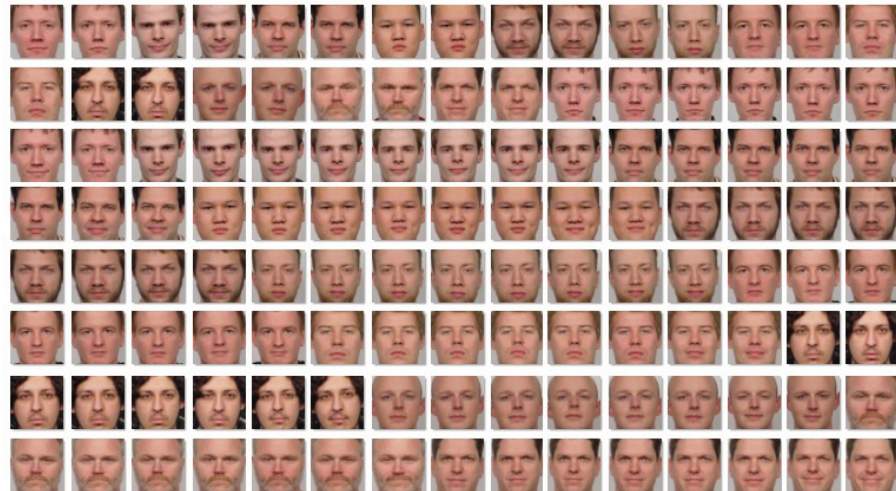
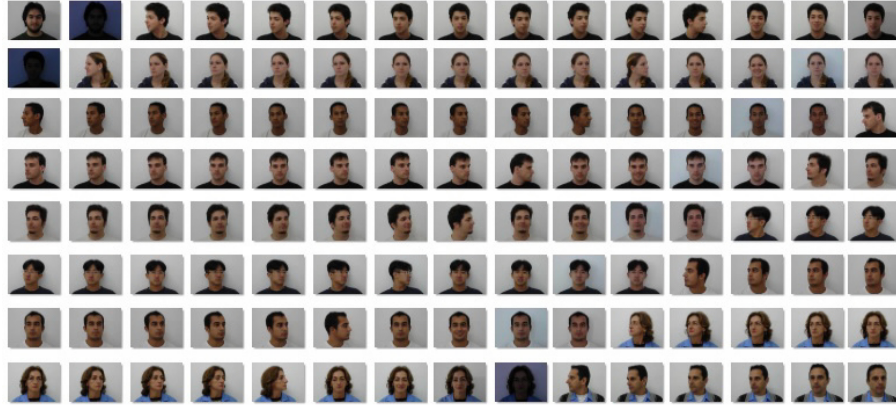
En este Capítulo se muestran los resultados obtenidos con los tres algoritmos implementados.

### 5.1. Conjuntos de datos y pre-procesamiento

El conjunto de imágenes con las que se trabajaron fueron obtenidas de la pagina: <http://www.cs.umass.edu/lfw>. Las imágenes 5.1.1, 5.1.2 y 5.1.3 muestran las bases de datos utilizadas en los experimentos llevados a cabo.

#### 5.1.1. Base de imágenes FEI

La base de datos que se ocupó contiene 199 imágenes con 14 personas diferentes, en los experimentos llevados a cabo se utilizaron 42 imágenes de prueba y 157 imágenes para entrenamiento. Cada una de las imágenes fué tomada con una expresión facial, ángulo diferente e incluso con una tonalidad diferente. Algunas de las imágenes del conjunto de datos incluso estan de perfil. Una de las principales debilidades de algunos sistema de reconocimiento facial es debida a el ángulo en el que se encuentra el rostro que queremos reconocer. Estudios han confirmado que el reconocimiento actúa correctamente en imágenes de frente, sin embargo, en imágenes de perfil algunos sistemas tienen problemas para reconocer a una persona correctamente. La Figura 5.1.1 muestra algunas imágenes de ejemplo de la base de datos FEI.



### 5.1.2. Base de imágenes Yale

La base de imágenes de rostros Yale contiene 165 imágenes de 15 individuos. Para cada individuo en la base de datos existen 11 imágenes, cada imagen con una expresión facial diferente. A diferencia del conjunto de datos IMA y FEI todas las imágenes del conjunto de datos se tomaron de frente, la única variación entre imágenes del mismo individuo es la expresión facial. La complejidad de este conjunto de datos radica en que algunos individuos son muy parecidos entre si. La Figura 5.1.2 muestra algunas imágenes de ejemplo de la base de datos Yale.



### 5.1.3. Base de imágenes IMA

La base de datos que se utilizó contiene 199 imágenes, el conjunto de datos contiene 14 personas diferentes. Se utilizaron 42 imágenes para prueba y 157 imágenes para entrenamiento, Cada una de las imágenes fué tomada con una expresión facial, ángulo diferente e incluso con una tonalidad diferente. Algunas de las imágenes del conjunto de datos incluso estan de perfil. La Figura 5.1.3 muestra algunas imágenes de ejemplo de la base de datos IMA.

Para obtener unicamente el rostro de cada imagen se aplica un algoritmo que enfoca los rostros que se encuentren en una imagen identificando el tono de la piel en conjunto con las formas de ojos, boca y la nariz. La Figura 5-1 tiene un tamaño de  $800 \times 600$  pixeles de altura. Sin embargo una gran parte de la imagen contiene información que no es útil y podría afectar el desempeño del algoritmo, es por ello que se aplica el algoritmo propuesto para recuperar unicamente la parte que nos interesa.

Aplicando el algoritmo eliminamos todo el fondo de la imagen e incluso el torax y parte del cabello de las personas se elimina. Aún cuando son eliminadas parte de la persona, estas no son utilizadas para reconocer a la persona, pues los algoritmos implementados unicamente utilizan el rostra de la persona para identificar a la persona. La Figura 5-2 muestra como es eliminada gran parte de la imagen y sólo es obtenido el rostro de la persona en la imagen.

Al obtener el rostro de la imagen realizamos un recorte y cambiamos el tamaño de la imagen con un tamaño de  $300 \times 300$  pixeles. En la Figura 5-3 se muestra el resultado.



Figura 5-1: Esta imagen de entrada a la cual se detectara el rostro.

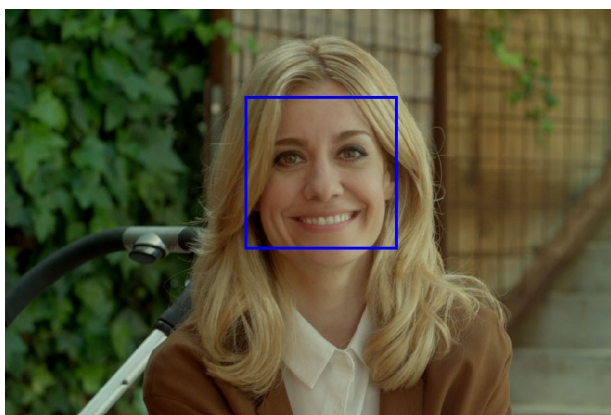


Figura 5-2: Al aplicar el algoritmo para detectar el rostro, este selecciona el contorno del rostro, el cual esta en un recuadro de color azul.

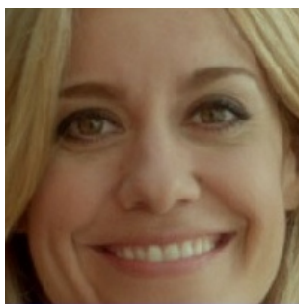


Figura 5-3: En esta imagen, se aplico el detector de rostro y se recorto solo el rostro, con un tamaño de 300 x 300 pixeles

## 5.2. Resultados

Existen diferentes métricas que pueden ser utilizadas para evaluar la precisión de los sistemas de reconocimiento de patrones. Para evaluar los sistemas biométricos son a menudo utilizadas las métricas Taza de Falsos Positivos y Taza de Verdaderos Positivos. El uso de estas métricas tiene como objetivo principal identificar los grupos de individuos sobre los que el sistema tiene un mal desempeño o tiene un desproporcionado número de errores de autenticación. Un análisis particular de estos individuos y sus propiedades comunes podría exponer las debilidades fundamentales del sistema biométrico utilizado y conocer las debilidades del sistema ayudaría a desarrollar sistemas biométricos mas robustos. Las Tablas 5-1 muestra los resultados obtenidos con el conjunto de datos Yale utilizando la base de datos de Yale. Los resultados muestran un buen desempeño del sistema únicamente en los individuos 1,3,4,8 y 10, en los demás individuos en algunos casos no reconoce al menos un individuo, los peores casos son para los individuos 2, 5, 11 y 12. El desempeño general del sistema con el algoritmo LDA es de solo 62 %.

Cuadro 5-1: Matriz de confusión utilizando el algoritmo LDA

1	2	3	4	5	6	7	8	9	10	11	12
2	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	1	1	0	0	0	2
0	0	2	0	0	0	0	0	0	0	0	3
0	0	0	2	0	0	0	0	0	0	0	4
0	0	0	1	0	0	1	0	0	0	0	5
0	0	0	0	0	1	0	0	0	1	0	6
0	0	0	1	0	0	1	0	0	0	0	7
0	0	0	0	0	0	0	2	0	0	0	8
0	0	0	0	0	0	0	0	1	0	0	9
0	0	0	0	0	0	0	0	0	2	0	10
1	0	0	0	0	0	0	0	1	0	0	11
0	0	0	2	0	0	0	0	0	0	0	12

La Tabla 5-2 muestra la matriz de confusión obtenida utilizando el algoritmo PCA en el conjunto de datos de Yale. En este caso el desempeño del algoritmo es mejor en comparación con LDA, el sistema solo tiene un fallo al identificar el individuo 2, que confunde con el individuo 8 en una prueba. El desempeño general del sistema utilizando PCA es de 95.83 %.

La Tabla 5-3 muestra la matriz de confusión obtenida utilizando el algoritmo LBP en el



Cuadro 5-2: Matriz de confusión utilizando el algoritmo PCA

	1	2	3	4	5	6	7	8	9	10	11	12
2	0	0	0	0	0	0	0	0	0	0	0	1
0	1	0	0	0	0	0	0	1	0	0	0	2
0	0	2	0	0	0	0	0	0	0	0	0	3
0	0	0	2	0	0	0	0	0	0	0	0	4
0	0	0	0	2	0	0	0	0	0	0	0	5
0	0	0	0	0	2	0	0	0	0	0	0	6
0	0	0	0	0	0	2	0	0	0	0	0	7
0	0	0	0	0	0	0	2	0	0	0	0	8
0	0	0	0	0	0	0	0	2	0	0	0	9
0	0	0	0	0	0	0	0	0	2	0	0	10
0	0	0	0	0	0	0	0	0	0	2	0	11
0	0	0	0	0	0	0	0	0	0	0	2	12

conjunto de datos de Yale. En este caso el desempeño del algoritmo es mejor en comparación con los dos algoritmos anteriores (PCA y LDA), el sistema no tiene un solo fallo. El desempeño general del sistema utilizando PCA es de 100%. Aunque el conjunto de datos pareciera muy trivial, la presencia de factores de iluminación y expresiones faciales pueden reducir el desempeño de los sistemas a niveles inaceptables.

Cuadro 5-3: Matriz de confusión utilizando el algoritmo LBP

	1	2	3	4	5	6	7	8	9	10	11	12
2	0	0	0	0	0	0	0	0	0	0	0	1
0	2	0	0	0	0	0	0	0	0	0	0	2
0	0	2	0	0	0	0	0	0	0	0	0	3
0	0	0	2	0	0	0	0	0	0	0	0	4
0	0	0	0	2	0	0	0	0	0	0	0	5
0	0	0	0	0	2	0	0	0	0	0	0	6
0	0	0	0	0	0	2	0	0	0	0	0	7
0	0	0	0	0	0	0	2	0	0	0	0	8
0	0	0	0	0	0	0	0	2	0	0	0	9
0	0	0	0	0	0	0	0	0	2	0	0	10
0	0	0	0	0	0	0	0	0	0	2	0	11
0	0	0	0	0	0	0	0	0	0	0	2	12

Para el conjunto de datos FEI, la Tabla 5-4 muestra los resultados obtenidos con el conjunto de datos. Los resultados muestran un mal desempeño del sistema. Únicamente en los individuos 2 y 6 son identificados plenamente, en los demás individuos en algunos casos no reconoce al

menos un individuo. El desempeño general del sistema con el algoritmo LDA sobre esta base de datos es de solo 42.82%.

Cuadro 5-4: Matriz de confusión utilizando el algoritmo LDA

1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	2	1	0	0	0	0	0	0	0	0	0	0	0
0	3	0	0	0	0	0	0	0	0	0	0	0	0
0	0	2	0	0	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	1	0	0	0	1	0	0	0
0	0	0	0	1	1	1	0	0	0	0	0	0	0
0	0	0	0	0	3	0	0	0	0	0	0	0	0
0	0	0	0	0	0	2	0	0	0	0	1	0	0
0	0	0	0	0	0	0	1	0	0	0	2	0	0
0	0	0	0	0	0	0	2	1	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	2	0	0	0
0	0	0	0	0	0	0	0	1	0	2	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	1	0	0	1	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	2	1	0

La Tabla 5-5 muestra la matriz de confusión obtenida utilizando el algoritmo PCA en el conjunto de datos de FEI. En este caso el desempeño del algoritmo es también mejor en comparación con LDA, el sistema solo tiene un fallo al identificar el individuo 14, que confunde con el individuo 13 en sólo una prueba. El desempeño general del sistema utilizando PCA es de 97.61%

La Tabla 5-6 muestra la matriz de confusión obtenida utilizando el algoritmo LBP en el conjunto de datos de FEI. En este caso el desempeño del algoritmo es una vez más mejor en comparación con los dos algoritmos anteriores (PCA y LDA), el sistema no tiene un solo fallo. El desempeño general del sistema utilizando PCA es de 100%. Aunque el conjunto de datos pareciera muy trivial, la presencia de factores de iluminación y expresiones faciales pueden reducir el desempeño de los sistemas a niveles inaceptables.

La Tabla 5-7 muestra los resultados obtenidos con el conjunto de datos IMA. Los resultados muestran una vez más un mal desempeño en este conjunto de datos. Sin embargo, en comparación con los dos conjuntos anteriores, este es mejorado significativamente. El desempeño general del sistema con el algoritmo LDA es de solo 79.31%.

La Tabla 5-8 muestra los resultados obtenidos con el conjunto de datos IMA. Los resultados

Cuadro 5-5: Matriz de confusión utilizando el algoritmo PCA

1	2	3	4	5	6	7	8	9	10	11	12	13	14
3	0	0	0	0	0	0	0	0	0	0	0	0	0
0	3	0	0	0	0	0	0	0	0	0	0	0	0
0	0	3	0	0	0	0	0	0	0	0	0	0	0
0	0	0	3	0	0	0	0	0	0	0	0	0	0
0	0	0	0	3	0	0	0	0	0	0	0	0	0
0	0	0	0	0	3	0	0	0	0	0	0	0	0
0	0	0	0	0	0	3	0	0	0	0	0	0	0
0	0	0	0	0	0	0	3	0	0	0	0	0	0
0	0	0	0	0	0	0	0	3	0	0	0	0	0
0	0	0	0	0	0	0	0	0	3	0	0	0	0
0	0	0	0	0	0	0	0	0	0	3	0	0	0
0	0	0	0	0	0	0	0	0	0	0	3	0	0
0	0	0	0	0	0	0	0	0	0	0	0	3	0
0	0	0	0	0	0	0	0	0	0	0	0	0	2

Cuadro 5-6: Matriz de confusión utilizando el algoritmo LBP

1	2	3	4	5	6	7	8	9	10	11	12	13	14
3	0	0	0	0	0	0	0	0	0	0	0	0	0
0	3	0	0	0	0	0	0	0	0	0	0	0	0
0	0	3	0	0	0	0	0	0	0	0	0	0	0
0	0	0	3	0	0	0	0	0	0	0	0	0	0
0	0	0	0	3	0	0	0	0	0	0	0	0	0
0	0	0	0	0	3	0	0	0	0	0	0	0	0
0	0	0	0	0	0	3	0	0	0	0	0	0	0
0	0	0	0	0	0	0	3	0	0	0	0	0	0
0	0	0	0	0	0	0	0	3	0	0	0	0	0
0	0	0	0	0	0	0	0	0	3	0	0	0	0
0	0	0	0	0	0	0	0	0	0	3	0	0	0
0	0	0	0	0	0	0	0	0	0	0	3	0	0
0	0	0	0	0	0	0	0	0	0	0	0	3	0
0	0	0	0	0	0	0	0	0	0	0	0	0	3

muestran un excelente desempeño en este conjunto de datos. El desempeño general del sistema con el algoritmo PCA es de 100%.

La Tabla 5-9 muestra los resultados obtenidos con el conjunto de datos IMA. Los resultados muestran un excelente desempeño en este conjunto de datos. Al igual que el algoritmo PCA, el desempeño general del sistema con el algoritmo LBP es de 100%.

Cuadro 5-7: Matriz de confusión utilizando el algoritmo LDA

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Cuadro 5-8: Matriz de confusión utilizando el algoritmo PCA

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Cuadro 5-9: Matriz de confusión utilizando el algoritmo LBP

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

## Capítulo 6

# Conclusiones

En esta tesis se implementan 3 algoritmos de reconocimiento facial y se comparan los desempeños obtenidos con cada algoritmo con tres conjuntos de datos. En los resultados experimentales es posible ver que la mejor precisión es obtenida con el método LBP, ya que en los tres conjuntos de datos se obtuvo la mejor precisión con este algoritmo.

Al trabajar con imágenes con poca iluminación la eficiencia de los algoritmos disminuía considerablemente con los algoritmos, por lo cual se eliminaron todas las imágenes que afectaran el desempeño de los algoritmos, a continuación se muestran algunas imágenes con poca iluminación 6-1 y 6-2.

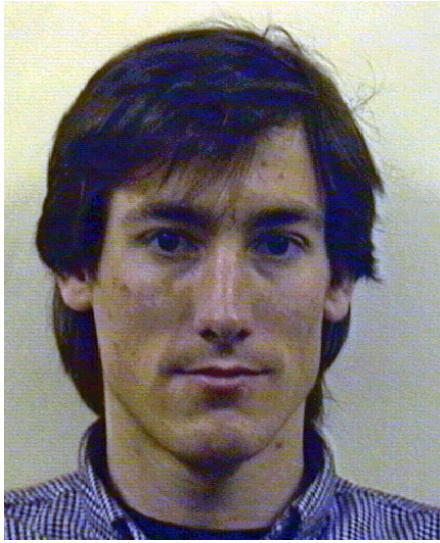


Figura 6-1: Figura. Imàgen ejemplo con poca iluminaciòn que afecto el desempeño de los algoritmos.

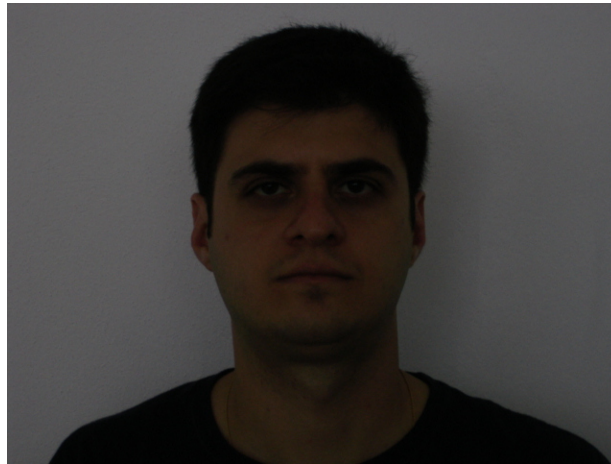


Figura 6-2: Figura. Imgaen ejemplo con poca iluminaciòn, que afecta el desempeño de los algoritmos.



# Bibliografía

- [1] **Anil K. Jain**, *Fundamentals of Digital Image Processing*, Prentice Hill, Englewood Clifft, 1989.
- [2] T. Sakai, M. Nagao, and S. Fujibayashi, "Line Extraction and Pattern Detection in a Photograph", *Pattern Recognition*, 1969
- [3] **Adini, T., Moises, Y. y Ullman, S.** "Face Recognition the probleman of Compensating on Pattern Analysis and Machine Intelligence, Vol 19, pp. 721 - 732, 1997.
- [4] **Baron, R. J.** "Mechanisms of human facial recognition", *International Journal of Man Machine Studies*, Vol. 15, pp. 137- 178, 1998.
- [5] **Bledsoe, W.W.** "The model method in face recognition". Technical, Report PRI:15, Panoramic Research In., Palo Alto, 1966.
- [6] **Burel, G. and Carel. D.** "Detection and Localization of Faces on Digital Images", *Pattern Recognition Letters*, Vol 15, No. 10, pp. 963 - 967, 1994.
- [7] **Cabello Pardos Enrique**, "Técnicas de reconocimiento facial mediante redes neuronales", Tesis Doctoral Faculta de Onfirmática abril 2004
- [8] **Chen, T. and Wu, H.R.** "Adaptive impulse detection using center weighted mediam filters", *IEEE Signal Processing Letters*, Vol. 8, No.1, pp 1-3, 2001.
- [9] **Craw, I., Elias, H. and lishman, J.R.** "Automatic Extraction of Face Feactures", *Pat- tern Precognition Letters*, Vol. 5, No. 183 - 187, 1987.

- [10] **Jeng, S.H., MarLio, H.Y., Han, C.C., Chern, M.Y. and Lui, Y.** "Facial Feature Detection using Geometrical Face Model: an Efficient Approach", Pattern Recognition, Vol 31, pp.273-282, 1998.
- [11] **Jain V. and Learned\_Miller, E.** "FDDB; A Benchmark for Face Detection in Unconstrained Setting", Technial report, University of Massachusetts, Amhert, 2010".
- [12] **Haiyuan Wu, Qian Chen and Massahiko Yachida,** "Face Detection From COlor Images Using a Fuzzy Pattern Matching Method"IEEE Transactions on Pettern Analysis and Machine Intelligence, Vol. 21, No. 6, pp. 55 - 563, 1999.
- [13] **Kanade, T.** "Picture processing by computer complex and recongnition of human faces", Technical report, Kyoto University, Dept. of Information Science, 1973
- [14] **Kelly , M.D.** "Visual Identification of People by Computer", Stanford Artificial Intelligence Project Memo AI-130, July, 1970.
- [15] **Lanitis, A., Taylor, C.J. and Cootes, T.F.** .An Automatic Face Identification System Using Flexible Appearance Models", Image and Vision COmputing, Vol. 13, No. 5, pp. 393 - 401, 1995.
- [16] **Li, S.Z. and Zhang, Z.Q.** "Floatboost learning and statistical face detection,"IEEE Transaction on Pattern Analysis and Machine Intellingence, Vol. 26, No. 9, pp. 1112 - 11123, 2004.
- [17] **Penev,P.S.andAtick,J.J.** "Local Feature Analysis: a General Statistical Theory for Object Representation", Network: Computation in Neural System, Vol. 7, pp. 447 - 500, 1996.
- [18] **Rowley, H.A., Baluja, S. and Kanade, T.** "Neural Network - Based Face Detection", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vo. 20, No. 1, pp. 23 - 38, 1998.
- [19] **Shin, F.Y. and Chuang, C.F.** .Automatic Extraction of Head and Face Boundaries and Facial Feactures", Information Sciences, Vol. 158, pp. 117 - 130, 2004.
- [20] **Sung, K.K. and Poggio, T.** .Example-Based Learning for View-Based Human Face intelligence, Vo., 20, No. 1, pp. 39 - 51, 1998.

- [21] **Timo, A., Adbenour , H. and Matti. P.** "Face Recognition with Local Binary Pattern", European Conference on Computer Vision, pp. 469 - 481, 2004.
- [22] **O ´Toole, A.J., Abdi, H., Deffenbcher, K.A. and Valentin, D.** "A Low-Dimensions of the Representation of Faces in the Higher Dimensions of the Space", Journal of the Optical Society of America, Vol. 10, pp. 405 - 411, 1993
- [23] **P. Mayorga Ortiz, C. Druzgalski, M. A. Criollo Arellano, H. O. González Arriega,** "GMM y LDA aplicado a la detección de enfermedades pulmonares", Revista Mex, ing Biómed vol.34 no.2 México Agosto 2013.
- [24] **Mauricio Delbracio, Matías Mateu,** "Trabajo Final de Reconocimiento de Patrones", 8 de Marzo del 2006.
- [25] **Turk, M. and Pentland, A.** "Face Recognition using Eigenface", Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Maui, Hawaii, pp. 586 - 591, 199b.
- [26] **Turk, M. and Pentland, A.** "Face Recognition using Eigenfaces", Proc. IEEE Conference on Computer Vision and Pattern Recognition, pp. 586 - 591, 2000.
- [27] **Viola, P. and Jones, M. J.** "Rapid Object Detection Using a Boosted Cascade of Simple Features", Proc. Conference Vision and Pattern Recognition (CVPR), 2001.
- [28] **Wiskott, L., Fellous, J.M., Krunger, N. and Malsburg, C.** "Face Recognition by Elastic Bunch Graph Matching", IEEE Transaction on. Pattern Analysis and Machine Intellegence, Vol. 19, No. 7, pp. 755 - 779, 1997.
- [29] **Yang. G, and Haung, T.S.** "Human Face Detection in Complex Background", Pattern Recognition, Vol. 27, No. 9, pp 712 - 735, 1997.
- [30] **Zhao, W., Chellappa, R., Rosenfled, A. and Philips, P.J.** "Face Recognition: A Literature Survey", ACM Computing Surveys, pp. 399 - 458, 2003.
- [31] **Villardón José López,** Análisis de Componentes Principales (PCA), 2009.

- [32] **Enrique L. Sucar y Giovani Gómez**, *Visión Computacional, Instituto Nacional de Astrofísica, Óptica y Electrónica. Fotointerpretación y Teledetección, Universidad de Murcia.*
- [33] **Cuevas Erik, Zaldivar Daniel, Pérez Marco**, *Procesamiento Digital de Imágenes con Matlab y Simulink*, Alfaomega, 2010.
- [34] **Carlos Platero Dueñas**, *Apuntes de Vision Artificial*, Depto. Electronica, Automatica e Informatica Industrial, 2009.
- [35] **Craig S. Lent**, Wiley, *Learning to Program with Matlab*, Building GUI Tools, 2013.
- [36] **Erik Cuevas, Daniel Zaldivar, Marco Perez**, *Procesamiento Digital de imágenes con Matlab y Simulink*, Alfaomega, 2010
- [37] **Depool Rivero Ramon e Ing. Monasterio Dioscoro**, *Probabilidad y Estadística. Aplicaciones a la Ingeniería*, Unexpo. 2013.
- [38] **Deisy Chaves, PhD. Maria Trujillo y PhD. Andres Rojas**, *Muestreo y Cuantificación de imágenes de Carbonizados*, 2008.
- [39] **Boris Escalares Ramirez**, *Procesamiento Digital de imágenes*, Agosto 2006.
- [40] **Noda La Serna Palomino y Ulises Roman Concha**, *Técnicas de Segmentación en Procesamiento Digital de imágenes*, Facultad de Sistemas e Informatica, Universidad Nacional Mayor de San Marcos, Diciembre 2009.
- [41] **Sameer Ruparelia**, *Implementation of Watershed Based Image Segmentation Algorithm in FPGA*, *Tesis de Maestria: Sistemas Embebidos*, Marzo 2012.
- [42] **Manisha Gupta, Abdul Wahid Ansari, Mannika Garg**, *Mathematical Morphology based gray scale Image Segmentation using improved watershed transform*, *Institute of Research Engineers and Doctors*, 2013.
- [43] **Clifford A. Shaffer, Hanann Sameyn**, *Optimal Quadtree Construction Algorithms*, *Computer Science Department and Center for Automation Research, University of Maryland*, 1987.

- [44] **Oscar E. Ramos, Babak Rezaei**, Scene Segmentation and Interpretation Image Segmentation using Region Growing, *MSc in Computer Vision and Robotics Universitat de Girona*, 2010.
- [45] **Anali J. Alfaro Alfaro, Ivan A. Sipiran Mendoza**, Diseño de un Algoritmo de Segmentación de imágenes aplicando el Funcional de Mumford-Shah para mejorar el desempeño de los Algoritmos Clasicos de Segmentacion, *Tesis de Ingenieria Informatica*.
- [46] **Jos Roerdink, Arnold Meijster**, The Watershed Transform: Definitions, Algorithms and Parallelization Strategies, 2010.
- [47] **Norberto Malpica, Carlos Ortiz de Solorzano, Juan Jose Vaquero, Andres Santos, Isabel Vallcorba, Jose Miguel Garacia-Sagredo, Francisco del Pozo**, Applying Watershed Algorithms to the Segmentation of Cluster ed Nuclei, *Grupo de Bioingenieria y Telemedicina, Universidad Politecnica de Madrid*, Abril 1997.
- [48] **Alvaro Suarez Bravo**, Analisis de Metodos de Procesamiento de imágenes Estereocopicas Forestales, *Tesis de Maestria: Ingenieria Informatica para la Industria*, 2009.
- [49] **Malik Sikandar Hayat Khiyal, Aihab Khan, Amna Bibi** , Modified Watershed Algorithm for Segmentation of 2D Images, 2006.
- [52] **Bryan S. Morse**, Thresholding, *Brigham Young University*, 2000.
- [51] **Salem Saleh Al-amri1, N.V. Kalyankar, Khamitkar S.D**, Image Segmentation by Using Thershod Techniques, *Journal Of Computing*, 2010.
- [52] **Bryan S. Morse**, Segmentation (Region Based), *Brigham Young University*, 2000.
- [53] **Nora La Serna Palomino, Luzmila Pro Concepcion**, Watershed: un algoritmo eficiente y flexible para segmentacion de imÁgenes de geles 2-DE, *Revista de Investigacion de Sistemas e Informatica*, 2010.
- [54] **Robert M. Haralick, K. Shanmugan, Its'hak Dinstein**, Textural Features for Image Classification, 1973.

- [55] **Yan Quin Chen, Mark S. Nixon, David W. Thomas**, Statical Geometrical Features for Texture Classification, Department of Electronics and Computer Science, University of Southampton, U.K., 1994.
- [56] **Luis Enrique Sucar**, Clasificadores Bayesianos: de Datos a Conceptos, Instituto Nacional de Astrofisica, Optica y Electronica Tonantzintla, Puebla, Mexico, 2014.
- [57] **M.A. Vicente, C. Fernández, R. Puerto, A. Gil**, Universidad Miguel Hernández, España, Extracción de características para reconocimiento visual, 2004.
- [58] **Rafael C. Gonzalez, Richard E. Woods, Prentice Hall**, Digital Image Processing, 2001.
- [59] **W. Zhao, R. Chellapa, P. J. Phillips, A. Rosenfeld**, Universidad de Meryland, Face Recognition: A Literature Survey, 2000.
- [60] **Muhammad Sharif, Sajjad Mohsin, Muhammad Atif Ali**, Universidad Nacional de Ciencia y Tecnologia, Pakistan, Single Image Face Rognition Using Laplacian of Gaussian and Discrete Cosine Transforms,2012.
- [61] **Bruno Rodriguez Rodriguez**, Reconocimiento Facial, Universidad Oberta de Catalunya, 2011.
- [62] **Hugo Armando Gualdron Colmenares**, Implementacion de un detector de rostros en posicion frontal mediante un algoritmo de Boosting, Universidad Industrial de Santander, Bucaramanga, 2011.
- [63] **Maria Esther Garcia Chang**, Diseño e implementación de una herramienta de detección facial, Centro de Innovación y Desarrollo Tecnologico en Computo, IPN, 2009.
- [64] **Luis Blazquez Perez**, Reconocimiento Facial Basado en Puntos Caracteristicos de la Cara en entornos no controlados, Universidad Autonoma de Madrir, 2013.
- [65] **Cecilia Aguerrebere, German Capdehourat**, Reconocimiento de Caras con Caracteristicas Locales, Universidad de la Republica, 2006.

- [66] **Helei Sun, Jie Shen, Bin Chen**, LBP Based Fast Face Recognition System on Symbian Platform, AASRI Conference on Computational Intelligence and Bioinformatics, 2012.
- [67] **Hua Yu, Jie Yang**, A direct LDA algorithm for high-dimensional data with application to face recognition, Interactive System Labs, School of Computer Science, 23 Octobre 2000
- [68] **Changjun Zhou, Lan Wang, Qiang Zhang, Xiaopeng Wei**, Face recognition based on PCA image reconstruction and LDA, Key Laboratory of Advanced Design and Intelligent Computing, Dalian University, China, 2013.
- [69] **Vinay. A, Vinay. S. Shehar, K. N. Balasubramanya Murthy, S. Natarajan**, Performance Study of LDA and KFA for Gabor Based Face Recognition System, 3rd International Conference on Recent Trends in Computing 2005.
- [70] **LUO Yuan, WU Cai-ming, ZHANG Yi**, Facial expression feature extration using hybrid PCA and LBP, The Journal of China Universities of Posts and Telecommunications, abril 2013.
- [71] **Bogdan Smolka, y Karolina Nurzynska**, Power LBP: A novel texture perator for smiling and neutral facial display classification, ICCS 2015 International Conference On Computational Science, 2015.
- [72] **Bo Yang, Songcan Chen**, A comparative study on local binary pattern (LBP) based face recognition: LBP histogram versus LBP image, Neurucomputing 120, 2013.

# Apéndice A

## Guia de Usuario

En este Apéndice se muestra la guía de usuario para la interfaz desarrollada (SistBiom).

Para empezar abrimos Matlab, en nuestro caso utilizamos la versión 2013A, posicionamos el directorio raíz en la ruta donde tenemos los programas (.m). Como se ve en la siguiente Figura A-1.

En la ventana de Comandos solo escribimos el nombre del programa que es la interfaz, este programa es fácil de identificar por que existen 2 archivos con ese nombre uno con la extensión .m y el otro con extensión .fig, en este caso escribimos "SistBiom". Como se muestra en la siguiente Figura.

Damos enter, y esperamos a que cargue la interfaz, una vez que cargo tendremos una pantalla como muestra la siguiente Figura A-3

Ya que cargo la interfaz lo primero que haremos sera dar clic en el boton de Cargar Imágenz seleccionaremos la imagen que vamos a ocupar y damos aceptar. En la Figura A-4 se muestra el ejemplo.

Ya que le dimos aceptar, solo esperamos a que cargue la imagen, una vez cargada la imagen, solo es cuestión de seleccionar el Algoritmo con el que queremos hacer pruebas. En las Figuras A-5 y A-6 se hace el ejemplo con PCA.

Y para hacer otra prueba con la misma imagen solo seleccionamos otro algoritmo, si queremos cambiar la imagen solo volvemos a dar clic en Cargar Imágenz aplicamos el algoritmo que deseemos.



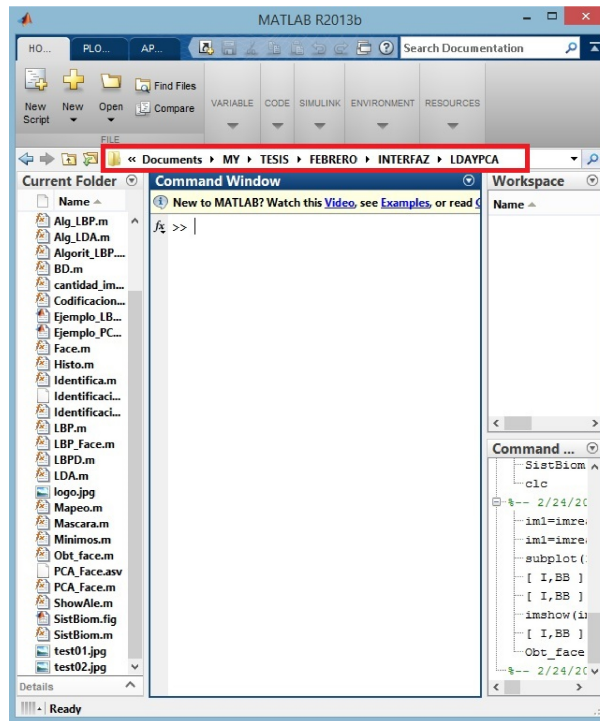


Figura A-1: Se carga la ruta donde tenemos la interfaz.

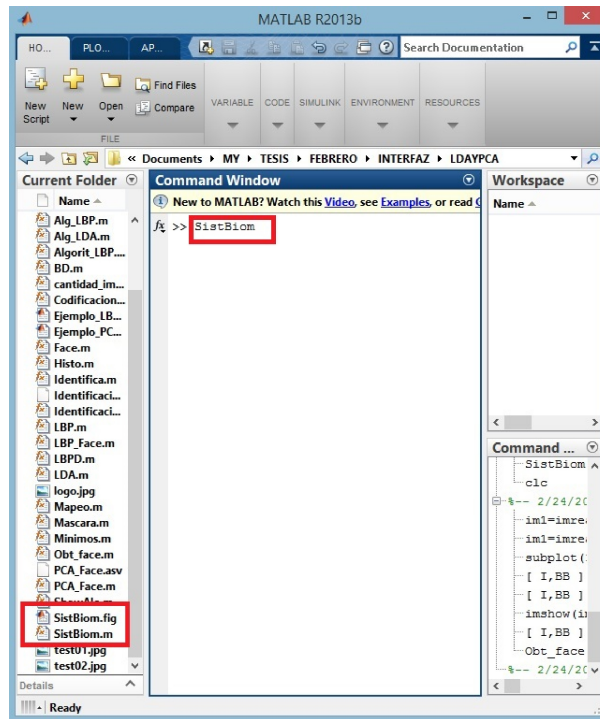


Figura A-2: Se escribe el nombre de la interfaz, sin la extensión, para ejecutar.

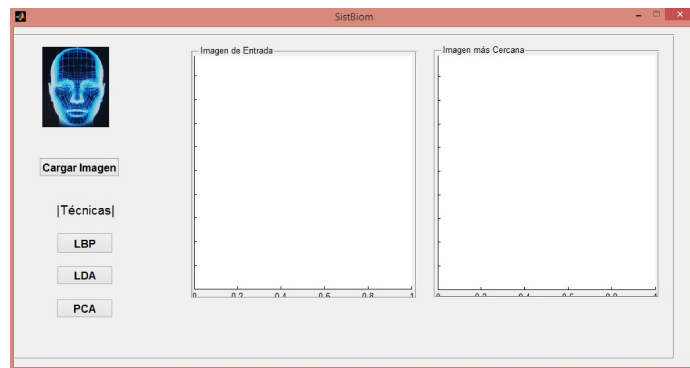


Figura A-3: Se carga la interfaz, para poder realizar las pruebas con los algoritmos que tiene.

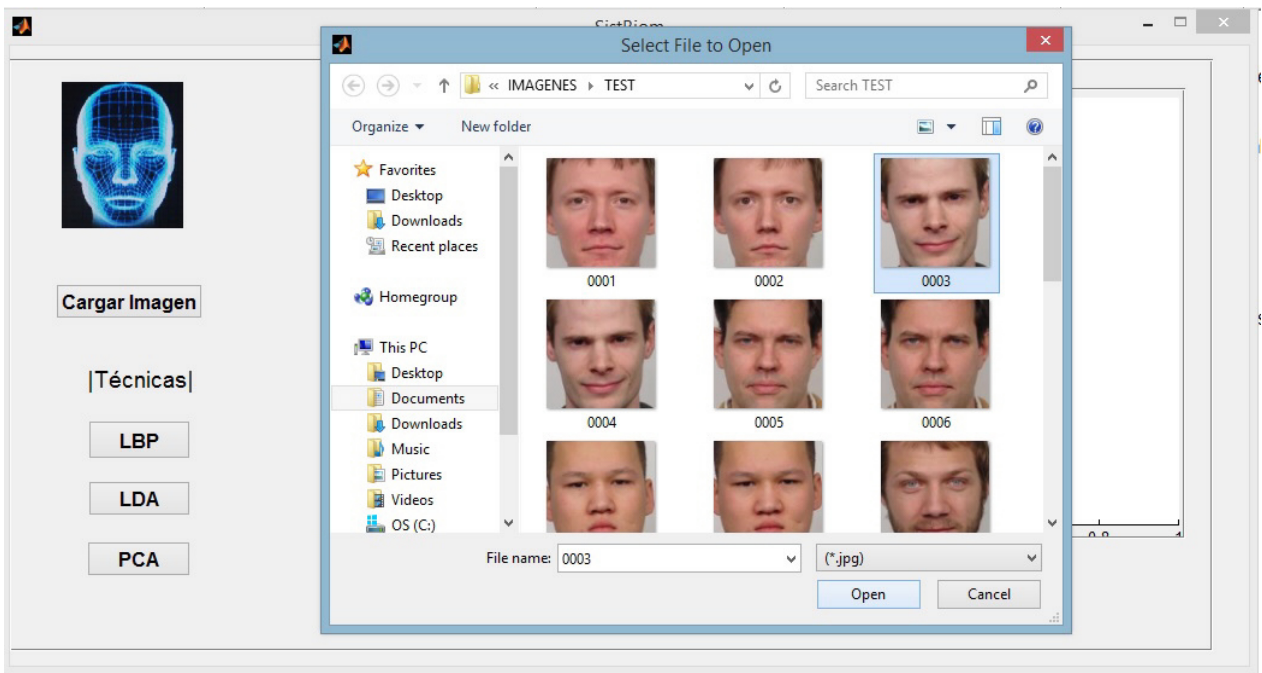


Figura A-4: Ejemplo para cargar una imagen.

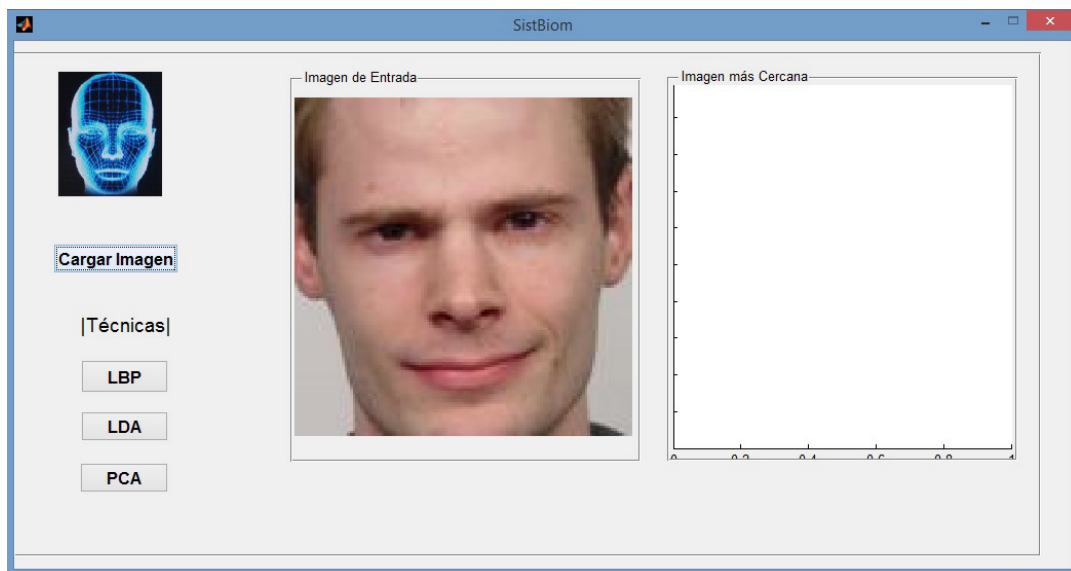


Figura A-5: Imágen cargada para poder trabajar con ella.

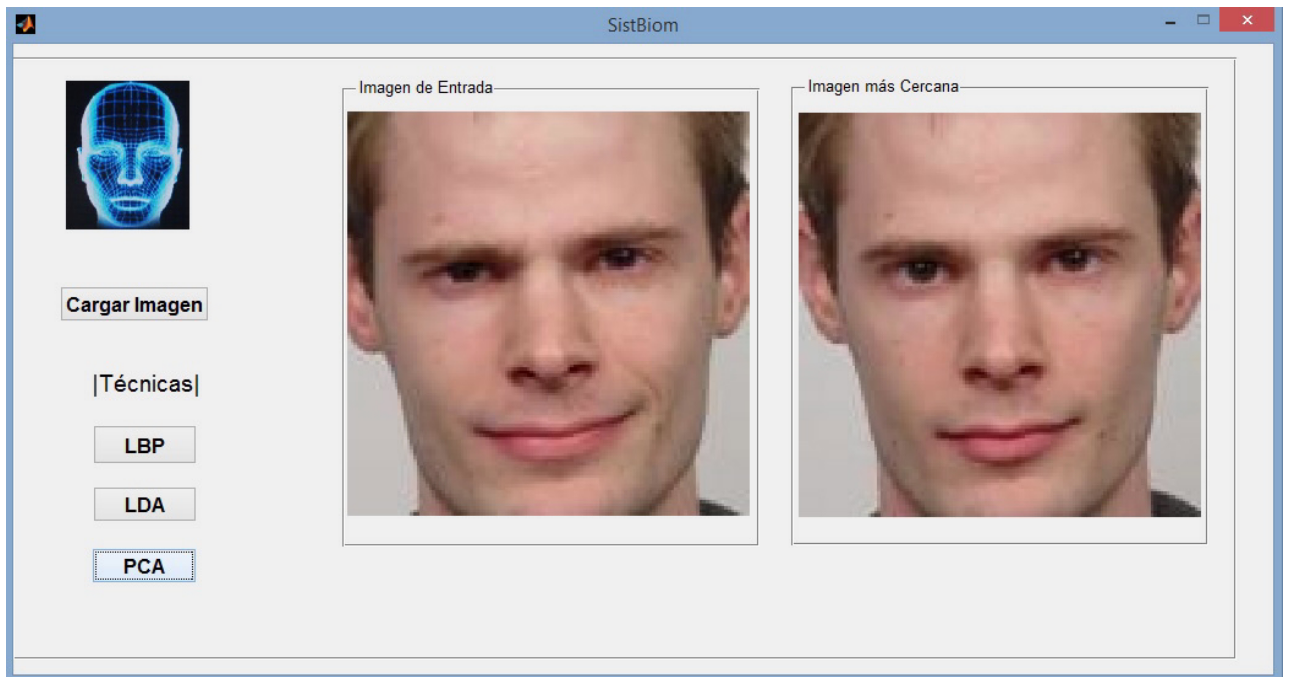


Figura A-6: Ejemplo, aplicando uno de los 3 algoritmos con los que podemos aplicar en la interfaz.