

INSTITUTO TECNOLÓGICO DE CHIHUAHUA
DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

**“REDES NEURONALES CONVOLUCIONALES
APLICADAS AL RECONOCIMIENTO DE IRIS”**

TESIS

QUE PARA OBTENER EL GRADO DE

**MAESTRO EN CIENCIAS
EN INGENIERÍA ELECTRÓNICA**

PRESENTA:

ADRIÁN ALBERTO GONZÁLEZ ESTRADA

DIRECTOR DE LA TESIS:
DR. JAVIER VEGA PINEDA



EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO NACIONAL DE MÉXICO



CHIHUAHUA, CHIH., NOVIEMBRE 2019



EDUCACIÓN

SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO

Instituto Tecnológico de Chihuahua

"2019, Año del Caudillo del Sur, Emiliano Zapata"

Chihuahua, Chih. **5 de noviembre de 2019**

**C. ADRIÁN ALBERTO GONZÁLEZ ESTRADA
PRESENTE**

Por este conducto le comunico que a propuesta del Jurado de Examen, la División de Estudios de Posgrado e Investigación ha concedido autorización para la impresión de su tesis para obtener el grado de en Ciencias en Ingeniería Electrónica, cuyo título es:

"Redes neuronales convolucionales aplicadas al reconocimiento de iris"

La tesis presenta el siguiente contenido de capítulos:

- I Introducción
- II Reconocimiento de iris
- III Segmentación de la región del iris
- IV Redes neuronales convolucionales
- V Resultados y conclusiones

ATENTAMENTE

Excelencia en Educación Tecnológica
"La técnica por el Engrandecimiento de México"

**MTRO. LUIS CARDONA CHACÓN
JEFE DE LA DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN**



Ave Tecnológico No. 2909 Col. 10 de Mayo C.P. 31310, Chihuahua, Chih. México
Tel. 01 (614) 201 2000, (614) 413 5187, Ext. 2150 e-mail: dir_chihuahua@tecnm.mx

www.tecnm.mx | www.itchihuahua.edu.mx





EDUCACIÓN

SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO NACIONAL DE MÉXICO

Instituto Tecnológico de Chihuahua

"2019, Año del Caudillo del Sur, Emiliano Zapata"

Chihuahua, Chih. **5 de noviembre de 2019**

MTRO. LUIS CARDONA CHACÓN
JEFE DE LA DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN
PRESENTE

Por medio de la presente notificamos a usted que en cumplimiento de los requerimientos para la obtención del grado de en Ciencias en Ingeniería Electrónica, el documento de tesis del **C. ADRIÁN ALBERTO GONZÁLEZ ESTRADA**, ha sido aprobado y aceptado para su impresión. El título de la tesis es:

"Redes neuronales convolucionales aplicadas al reconocimiento de iris"

Por lo que proponemos, le sea concedida la autorización de impresión correspondiente.

Agradeciendo la atención a la presente, quedamos de usted:

ATENTAMENTE

Excelencia en Educación Tecnológica
"La técnica por el Engrandecimiento de México"

DR. JAVIER VEGA PINEDA
DIRECTOR DE TESIS

DR. LUIS FRANCISCO CORRAL MARTÍNEZ
MIEMBRO DEL JURADO DE EXAMEN

Juan Ramirez
DR. JUAN ALBERTO RAMÍREZ QUINTANA
MIEMBRO DEL JURADO DE EXAMEN

DR. JOSÉ LUIS DURÁN GÓMEZ
MIEMBRO DEL JURADO DE EXAMEN





EDUCACIÓN

SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO

Instituto Tecnológico de Chihuahua

"2019, Año del Caudillo del Sur, Emiliano Zapata"

CARTA CESIÓN DE DERECHOS

En la ciudad de Chihuahua el día , el que suscribe **C. ADRIÁN ALBERTO GONZÁLEZ ESTRADA** de la Maestría en Ciencias en Ingeniería Electrónica, con número de control **G17061497**, adscrito a la **DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN**, del Instituto Tecnológico de Chihuahua, manifiesta que es autor intelectual del presente trabajo de Tesis bajo la dirección del **Dr. Javier Vega Pineda** y cede los derechos del trabajo titulado **"Redes neuronales convolucionales aplicadas al reconocimiento de iris"**, al Tecnológico Nacional de México y/o Instituto Tecnológico de Chihuahua para su difusión, divulgación, transmisión, reproducción, así como su digitalización con fines académicos y de investigación.

Adrian Gonzalez

C. ADRIÁN ALBERTO GONZÁLEZ ESTRADA



Ave Tecnológico No. 2909 Col. 10 de Mayo C.P. 31310, Chihuahua, Chih. México
Tel. 01 (614) 201 2000,(614)413 5187, Ext. 2150 e-mail: dir_chihuahua@tecnm.mx

www.tecnm.mx | www.itchihuahua.edu.mx



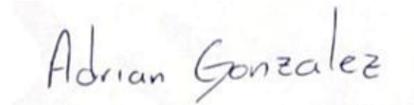
Chihuahua, Chihuahua, a 6 de noviembre de 2019

Dra. María Elena Álvarez-Buylla Roces
Directora del CONACYT

Por este conducto aprovecho la ocasión para saludarla e informarle que a la fecha he obtenido el grado de **Maestría en Ciencias de Ingeniería Electrónica** en la División de Estudios de Posgrado e Investigación del Instituto Tecnológico de Chihuahua. Agradezco todo el apoyo brindado por CONACYT ya que el otorgamiento de la beca para estudios de Posgrado, permitió dedicarme de tiempo completo al programa y con ello lograr el cumplimiento del objetivo principal del convenio establecido.

Sin otro particular por el momento, me es grato quedar de usted como su seguro servidor, no sin antes reiterar nuevamente todo mi agradecimiento.

A t e n t a m e n t e



Adrian Alberto Gonzalez Estrada
Exbecario CONACYT

“La educación es el arma más poderosa que puedes usar para cambiar el mundo” –

Nelson Mandela

“Una persona que nunca ha cometido un error nunca intenta nada nuevo” –

Albert Einstein

AGRADECIMIENTOS

Quisiera agradecer a primeramente a mis padres Víctor Manuel Gonzalez Miranda y Silvia Estrada Gonzalez por ser los principales promotores de mis sueños, gracias a ellos por cada día confiar y creer en mí y en mis expectativas, por haberme forjado como la persona que soy en la actualidad; muchos de mis logros se los debo a ustedes. Me formaron con reglas y ciertas libertades, pero al fin de cuentas, me motivaron constantemente. Gracias a mi madre por estar dispuesta a acompañarme con cada decisión difícil de tomar; gracias a mi padre por siempre desear y anhelar siempre lo mejor para mi vida con las duras reglas impuestas, gracias a los dos por cada consejo y por cada una de sus palabras que me guiaron durante mi vida.

A mis familiares que con su apoyo nunca faltan motivaciones para seguir esforzándose. A mis hermanos Gabriela Gonzalez y Miguel Gonzalez. A mi abuela Mary Gonzalez que siempre me ha apoyado en todo y a mi tía Marielena Gonzalez por haberme acogido en su hogar en la estancia de la maestría estos últimos años.

Agradezco al Tecnológico Nacional de México por el apoyo a este trabajo bajo el nombre de “redes neuronales convolucionales aplicadas al reconocimiento de iris”. También Agradezco al CONACYT por el apoyo económico brindado durante mis estudios.

Dr. Javier Vega, gracias por compartir su conocimiento y tiempo en las asesorías en estos últimos dos años. Dr. Juan Ramírez, gracias por guiarme con los conocimientos nuevos para mí en las nuevas tecnologías para concluir el trabajo deseado.

Y para finalizar agradezco a todos los que fueron mis compañeros de clase durante todos los años vividos en la estancia de maestría, por el ambiente de trabajo generado ya que gracias al compañerismo, amistad y apoyo moral han aportado en un alto porcentaje a mis ganas de seguir adelante.

RESUMEN**REDES NEURONALES CONVOLUCIONALES APLICADAS AL RECONOCIMIENTO DE IRIS**

Ing. Adrián Alberto González Estrada
Maestría en Ciencias en Ingeniería Electrónica
División de Estudios de Posgrado e Investigación
Instituto Tecnológico de Chihuahua
Chihuahua, Chih., 2019
Director de Tesis: Dr. Javier Vega Pineda

El reconocimiento de iris se refiere al proceso automático de identificar individuos con base en los patrones visuales específicos del iris. La naturaleza de las características aleatorias y distintivas de este elemento del ojo lo convierte en una señal viable para el reconocimiento biométrico. Un esquema exitoso en el reconocimiento de iris es el uso de las redes neuronales profundas especialmente en el uso de Redes Neuronales Convolucionales (CNN, por sus siglas en inglés) ya que permiten obtener características abstractas.

Sin embargo, los modelos basados en CNN utilizan una gran cantidad de recursos computacionales, por otra parte, no las hace factibles para implementaciones en sistemas embebidos/móviles. Por lo tanto, en este trabajo de tesis, se exploró el rendimiento de la utilización de modelos CNN para el reconocimiento del iris del ojo humano, para lo cual, se analizó la red CNN ganadora de premios en reconocimiento de objetos AlexNet y con base en dicha red, proponer un nuevo modelo de CNN que tiene como entrada imágenes segmentadas y normalizadas del iris y en cuanto a su arquitectura consta de 4 capas convoluciones, 3 capas totalmente conectadas, seguidas con sus respectivos mapas de activación y de agrupamiento para cada capa en específico. El modelo propuesto consta de 15 clases que se refieren a personas a identificar.

Para probar el modelo propuesto, se utilizó la base de datos UTIRIS V1 y se lograron resultados de desempeño en la red de 97% utilizando el conjunto de imágenes pruebas. Se presentan también, los resultados de la comparación de desempeño de la CNN propuesta contra dos CNN: la AlexNet y la CNN VGG-16.

CONTENIDO

CAPÍTULO 1 INTRODUCCIÓN	1
1.1. Aprendizaje Profundo en el Reconocimiento de Iris	2
1.2. Antecedentes	4
1.3. Justificación	6
1.4. Contribuciones	7
1.5. Presentación de la tesis	8
CAPÍTULO 2 RECONOCIMIENTO DE IRIS.....	9
2.1. Iris humano como reconocimiento biométrico	9
2.2. Sistema biométrico	11
2.3. Avances en los sistemas de reconocimiento de iris	12
2.4. Bases de datos	13
2.5. Problemas y limitaciones de los sistemas de reconocimiento biométricos	16
2.5.1. Factores que influyen en la calidad de la imagen	16
2.5.1.1. Desenfoco en la imagen.....	17
2.5.2. Consideraciones sobre la iluminación	18
2.5.3. Posicionamiento del iris	19
CAPÍTULO 3 SEGMENTACIÓN DE LA REGIÓN DEL IRIS.....	20
3.1. Preprocesamiento de la imagen.....	20
3.1.1. Selección de la imagen.....	21
3.1.2. Realce de bordes.....	22
3.1.3. Mejoramiento de la imagen.....	24
3.2. Mapa de bordes	26

3.2.1.	Filtro de Canny para detectar el mapa de bordes	26
3.3.	Transformada de Hough	29
3.4.	Normalización	35
3.4.1.	Proceso de Normalización.....	37
3.5.	Segmentación utilizando redes neuronales convoluciones	42
CAPÍTULO 4 REDES NEURONALES CONVOLUCIONALES		45
4.1.	Capa convolucional	46
4.2.	Capas ReLU (Unidades Lineales Rectificadas)	47
4.3.	Salto (stride).....	48
4.4.	Relleno (<i>padding</i>).....	49
4.5.	Normalización	50
4.6.	Capa de agrupamiento (<i>Pooling</i>)	51
4.7.	Capa totalmente conectada	53
4.8.	Dropout.....	53
4.9.	Bias (Sesgo)	54
4.10.	Capa clasificación Softmax	54
4.11.	Proceso de aprendizaje	55
4.11.1.	Backpropagation a través de la convolución.....	56
4.12.	Arquitectura AlexNet	57
4.12.1.	Entrenamiento.....	58
4.12.2.	Puntos principales.....	59
4.13.	Arquitectura Very Deep Convolutional Network (VGG)	59
4.13.1.	Configuraciones.....	60
4.13.2.	Entrenamiento.....	61

4.13.3. Puntos principales.....	62
4.14. Arquitectura IrisNet.....	62
4.15. Partición de los conjuntos de entrenamiento, validación y pruebas.....	66
4.16. Descripción general para la experimentación de la red neuronal	67
4.16.1. Entrenamiento de la red neuronal.....	69
CAPÍTULO 5 RESULTADOS Y CONCLUSIONES.....	71
5.1. Pruebas con diferentes algoritmos de entrenamiento	75
5.1.1. RMSprop.....	75
5.1.2. Adam.....	76
5.2. Análisis de los resultados	78
5.3. Conclusiones	79

LISTA DE FIGURAS

Figura 1.1. Método propuesto para el reconocimiento de iris. 2

Figura 2.1. Estructura del iris en el ojo humano. 10

Figura 2.2. Diagrama básico de un sistema biométrico. 12

Figura 2.3 Imágenes de Iris de la base de datos CASIA obtenida con cámara NIR [25]. 14

Figura 2.4 Imágenes de la base de datos UPOL capturadas en el espectro visible. 16

Figura 2.5 Problemas que se presentan en la calidad de las imágenes. 17

Figura 2.6 Imagen del ojo desenfocada por la captura. 18

Figura 2.7 pupila contraída y dilatada, esquema explicativo. 18

Figura 2.8 Ejemplos reales con iluminación ambiental, pupila dilatada (derecha) y pupila
contraída (izquierda). 19

Figura 2.9 a) imagen con vista frontal. b) imagen fuera de ángulo. c) esquema de visión del iris
fuera de ángulo. 19

Figura 3.1 Pasos de procesamiento de la imagen para la localización del iris y de la pupila. .. 21

Figura 3.2 Imagen sin procesar en formato RGB tomada de la base de datos UTIRIS V1. 22

Figura 3.3 Canal rojo de la imagen original. 22

Figura 3.4 Aplicación del filtro de mediana. 23

Figura 3.5 Mejoramiento de bordes para detectar los círculos del iris y de la pupila. 23

Figura 3.6 Problemas de iluminación. 24

Figura 3.7 Transformación lineal para mejorar el contraste. 25

Figura 3.8 Mejoramiento de contraste de la imagen. 25

Figura 3.9 a) Mapa de bordes sin difuminado, b) Mapa de bordes con problemas de
iluminación sin difuminado y mejoramiento de contraste. 27

Figura 3.10 Mapa de bordes aplicando contraste y difuminado. 27

Figura 3.11 Sección de la pupila por medio de un umbral. 28

Figura 3.12 Espacio de la imagen convertida al espacio de Hough. 30

Figura 3.13 Circunferencias encontradas con una pupila dilatada. 31

Figura 3.14 Ejemplo de circunferencias encontradas de las regiones del iris y de la pupila., a) Ejemplo circunferencias encontradas con una pupila mucho más pequeña que la región del iris, b) Ejemplo circunferencias encontradas con una pupila dilatada..... 32

Figura 3.15 Máscaras de las circunferencias encontradas del lado izquierdo, imagen binaria del ojo humano a comparar del lado derecho..... 33

Figura 3.16 a) Máscara de la circunferencia con el radio y centro encontrados, b) máscara de la pupila de la imagen del iris..... 34

Figura 3.17 circunferencias definidas que delimitan con más exactitud las regiones del iris y pupila..... 34

Figura 3.18 Proceso de Normalización. 36

Figura 3.19 Demostración del proceso de normalización con pocas muestras en ambas direcciones, a) visualización de las muestras obtenidas en el proceso de normalización, b) resultado final. 37

Figura 3.20 Coordenadas del límite del iris con base en el número de muestras de ángulo establecidos..... 38

Figura 3.21 Cálculos de línea recta para encontrar la intersección con los puntos de la pupila. 39

Figura 3.22 a) intersección con las primeras líneas proyectadas, b) intersecciones con todos los puntos de la circunferencia..... 39

Figura 3.23 Obtención de las muestras en dirección del radio..... 40

Figura 3.24 a) Sentido de reacomodo de los pixeles, b) resultado del reacomodo de los pixeles. 41

Figura 3.25 Imagen del iris normalizada..... 41

Figura 3.26 a) Problemas con las circunferencias fuera de rango, b) circunferencias fuera de rango sobre las circunferencias seleccionadas..... 42

Figura 3.27 Descripción del método propuesto en [37]..... 44

Figura 3.28 Bloque denso por concatenación de características [37]..... 44

Figura 4.1. Arquitectura tradicional de una CNN con las dos principales etapas. 46

Figura 4.2 Salto del filtro sobre una matriz a) entrada de tamaño 7x7 con un filtro de 3x3 aplicando con paso de una unidad, b) salida obtenida de tamaño 5x5..... 49

Figura 4.3 Un ejemplo de relleno sobre una imagen o mapa de activación.	50
Figura 4.4. Aplicación de agrupamiento máximo sobre una matriz de 7 x 7 con saltos de 1 y 2.	52
Figura 4.5. Capa totalmente conectada aplicada a una red neuronal.	53
Figura 4.6 Arquitectura AlexNet, a) sin particionamiento de GPU, b) con particionamiento de GPU [48].	58
Figura 4.7. Convolución aplicada a una imagen con relleno de ceros mediante un filtro de 3×3.	63
Figura 4.8 Arquitectura de la red neuronal convolucional implementada para el reconocimiento de iris.	65
Figura 4.9 Sección tomada para la eliminación de ruido ocasionada por oclusión de párpados.	66
Figura 4.10. Diagrama de flujo de entrenamiento de la red neuronal.	68
Figura 5.1. Los 24 filtros de tamaño 11×11 resultantes de la primera capa convolucional aplicados a la imagen original para la extracción de características.	71
Figura 5.2 Los 64 kerneles resultantes del entrenamiento de la red en la última capa convolucional.	72
Figura 5.3 Resultados obtenido con el algoritmo RMSprop con el modelo de VGG-16.	76
Figura 5.4 Resultados obtenido con el algoritmo Adam con el modelo de VGG-16.	77
Figura 5.5. Matriz de confusión aplicada al conjunto de pruebas.	79

LISTA DE TABLAS

Tabla 4.1 Configuración ConvNets. 61

Tabla 4.2 Número de parámetros (en millones). 61

Tabla 4.3 Arquitectura propuesta para el reconocimiento de iris. 64

Tabla 4.4 Configuración para la adquisición de la base de datos. 67

Tabla 4.5 Conjunto de imágenes establecidos para el reconocimiento de iris. 67

Tabla 4.6 Parámetros de entrenamiento aplicados al entrenamiento de la red neuronal
convolucional. 69

Tabla 5.1 Clases a identificar con la cantidad de imágenes que se cuentan por cada clase. 73

Tabla 5.2 Desempeños obtenidos de la red neuronal convolucional con los diferentes
conjuntos de datos establecidos. 74

Tabla 5.3 Análisis de los resultados obtenidos. 78

CAPÍTULO 1

INTRODUCCIÓN

El reconocimiento biométrico es la ciencia que se encarga de establecer la identidad de las personas con base en las características físicas únicas e intransferibles y ha tomado un papel importante en la vida moderna ya que se puede utilizar para la identificación de personas por medio del iris de sus ojos [1], por lo tanto, se utiliza en seguridad para ámbitos de accesos en una empresa, compras en internet, seguridad en la banca, etc., mientras que una contraseña puede ser olvidada o una tarjeta de identificación puede ser fácilmente extraviada.

Un órgano que se puede utilizar es el iris, ya que tiene patrones visuales que se utilizan para la detección de enfermedades (iridología), un ejemplo es la detección de la diabetes, entre otros tipos de enfermedades capaz de detectarse, de acuerdo a [2]. El iris se caracteriza por tener una textura única, estabilidad en el tiempo, accesibilidad y la posibilidad de escaneo no invasivo para la persona, además tiene la ventaja de que su variabilidad de patrones entre las personas es enorme [3].

El desarrollo de los sistemas de reconocimiento de iris a partir de una imagen digital generalmente se divide en cuatro etapas: preprocesamiento, localización de las regiones de interés, normalización y aplicación de las Redes Neuronales Convoluciones (CNN, por sus siglas en inglés), como se muestra en la Figura 1.1.

Una vez tomada la imagen del ojo, se le realiza un preprocesamiento para obtener su mapa de bordes, posteriormente para la localización se aplica la transformada de Hough en la detección de las circunferencias que determinan la posición del iris y la pupila en la imagen sobre los mapas de bordes que definen el límite interior (pupila) y exterior (iris) en el ojo.

La etapa de normalización de la imagen segmentada del iris consiste en realizar una transformación del espacio de coordenadas cartesianas (x, y) a un sistema de coordenadas polares (r, θ) para realizar computacionalmente una comparación entre imágenes de la misma persona capturadas en diferente posición de una manera más eficiente [4].

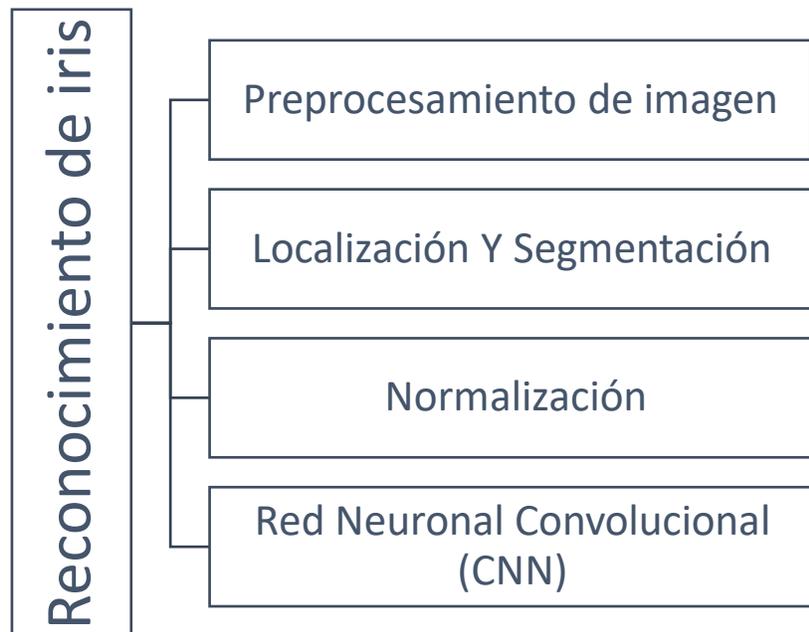


Figura 1.1. Método propuesto para el reconocimiento de iris.

Para la etapa de clasificación usando modelos CNN, la extracción de características efectivas de la imagen normalizada es uno de los elementos más importantes para el reconocimiento de objetos y tareas de visión. Por lo cual, los trabajos de investigación en esta área se han enfocado fuertemente en el diseño robusto para extracción de características en una gran variedad de tareas de clasificación de imágenes [5].

El interés de clasificar o extraer de manera eficiente las características de la imagen normalizada del iris se experimenta con las CNN, ya que el proceso de la extracción de características se realiza de manera autónoma dentro de la misma red, por consiguiente, no se sabe que realmente que se obtiene del entrenamiento de la CNN en la etapa de extracción de características.

1.1. Aprendizaje Profundo en el Reconocimiento de Iris

El aprendizaje profundo (DL, por sus siglas en inglés) permite a los modelos computacionales de múltiples capas de procesamiento que aprendan y representen datos con ciertos niveles de abstracción imitando como el cerebro percibe y entiende la información

recibida, permitiendo abarcar redes neuronales, métodos probabilísticos y una gran variedad de algoritmos de aprendizaje con modelos supervisados y no supervisados.

Una gran parte del éxito reciente de las redes neuronales de DL se explica por el hecho de que la mayor disponibilidad de datos y el poder computacional de los ordenadores modernos ha superado los límites de los algoritmos tradicionales de aprendizaje automático, que no aprovechan al máximo lo actualmente posible. El rendimiento del aprendizaje automático tradicional sigue siendo mejor a veces para conjuntos de datos más pequeños debido a más opciones, una mayor facilidad de interpretación del modelo y la tendencia a crear características interpretables que incorporen conocimientos específicos del dominio.

Se han realizado intentos de utilizar los principios de DL a la tarea de reconocimiento del iris, sin embargo, la limitación de estas técnicas al problema del reconocimiento del iris se debe al hecho de que se requiere una gran cantidad de datos de entrenamiento, las cuales no están disponibles por el momento en las bases de datos existentes en el área para su acceso por parte de los investigadores.

También DL es costoso desde el punto de vista computacional y requiere la potencia equivalente a varias unidades de procesamiento gráfico (GPU, del inglés *Graphics Processing Unit*). Esto es un fuerte impedimento para la implementación física por los recursos computacionales que requieren las redes con las limitaciones que se tienen en las tarjetas de desarrollo para los sistemas embebidos en los enfoques de aprendizaje profundo en aplicaciones tecnológicas.

Hasta la fecha no se ha entendido porqué el aprendizaje profundo debería funcionar para el reconocimiento del iris debido a la cantidad de datos del conjunto de las bases de datos que se tienen y por la desventaja de una CNN en el ámbito de no saber con exactitud qué características se obtienen del entrenamiento, y no se ha realizado un análisis sistemático para determinar la mejor manera de aprovechar los enfoques profundos modernos en el diseño de una arquitectura óptima de redes profundas para lograr una alta precisión con una baja complejidad computacional.

El solo colocar múltiples capas de redes en el diseño de una CNN para el reconocimiento del iris sin conocimientos adecuados sobre las redes neuronales no sería factible debido a la falta de conjuntos de datos de iris a gran escala (del dominio público), sería no óptimo debido a las opciones para la arquitectura CNN (entre otras el número de capas y su configuración) e ineficiente (debido a posibles capas redundantes) [6]. Por lo cual, desde una perspectiva de DL, el problema de clasificación se puede resolver mediante una transferencia de aprendizaje de modelos pre entrenados para tareas específicas y utilizarlas para un objetivo en particular [7].

La transferencia de aprendizaje es un método en visión por computadora porque esto permite construir modelos con precisión de una manera que ahorra tiempo, en lugar de empezar desde cero se comienza desde patrones que se han aprendido al resolver un problema diferente.

En visión por computadora, la transferencia de aprendizaje usualmente es expresada a través de utilizar modelos pre entrenados, fue entrenado sobre un gran conjunto de datos de referencia para resolver un problema similar al problema que se requiere resolver. En consecuencia, debido al costo computacional de la capacitación de tales modelos, es una práctica común importar y usar modelos publicadas en la literatura.

1.2. Antecedentes

La contribución más significativa al proceso de detección y localización del iris fue hecha originalmente por Daugman [8] y consiste en un operador integro diferencial que localiza los límites interior y exterior del iris.

Diversos esfuerzos en el reconocimiento del iris revisados en la literatura relacionada se realizaron con los conjuntos de datos disponibles públicamente en las bases de datos UBIRIS V1 [9], UBIRIS V1 [10] y MICHE [11], contienen imágenes en el espectro visible en formato RGB. Los desafíos en el reconocimiento de iris asociados con imágenes sin restricción de luz fueron estudiados en [12], en donde evalúan solo la parte de la segmentación del iris y las etapas de detección de ruido, excluyendo la parte del reconocimiento permitiendo la

evaluación de las diferentes fuentes de error del reconocimiento de iris. Por medio de esto fueron construidas las bases de datos UBIRIS V1 y UBIRIS V2.

En [13] se concluyeron que es mejor trabajar con el canal rojo de la imagen en formato RGB debido a que las longitudes de ondas correspondientes a la luz roja (más cercanas al infrarrojo cercano) son las más largas en el espectro visible del ser humano y así los patrones del iris son más visibles para su mejor reconocimiento. La mayoría de los trabajos revisados en el reconocimiento de iris utilizaron bases de datos con imágenes capturadas con cámaras de infrarrojo cercano ya que proveen mejores texturas del iris que una cámara de luz visible no puede captar.

Los métodos de aprendizaje automático especialmente los modelos CNN, han conducido especialmente a avances en el desarrollo de nuevas tecnologías enfocadas en la visión artificial, por ejemplo: la detección y reconocimiento de objetos, la segmentación en videos o en imágenes. Al intentar imitar la estructura de la corteza visual humana mediante el uso de redes neuronales de múltiples capas, se ha demostrado que el uso de DL es extremadamente eficaz para automatizar los procesos de aprendizajes de representación de características a partir de los datos de entrenamiento, eliminando la tarea de búsqueda de las características más eficientes para el reconocimiento de la red [14].

Existen diseños de arquitecturas específicas para el reconocimiento de iris como las arquitecturas DeepIrisNet [15], se componen de dos modelos de arquitecturas: DeepIrisNet-A y DeepIrisNet-B. La red DeepIrisNet-A se basa en el estándar de capas convoluciones, se compone de 8 capas convolucionales (conv1 - conv8), cada una seguida por su capa de normalización por lotes y sus capas de agrupamiento máximo y finalmente por 3 capas totalmente conectadas.

La red DeepIrisNet-B utiliza el apilamiento de capas de inicio, los entrenamientos y características utilizados fueron seleccionados por medio de análisis experimental. Se apilan primero las capas convolucionales (conv1 - conv5), y luego dos capas iniciales (Inception6 e Inception7) y finalmente tres capas totalmente conectadas.

La entrada a la arquitectura DeepIris-Net son imágenes del iris de niveles de gris de 128×128 píxeles sin procesamiento previo y capturadas por el sensor LG2200. Las dos redes se entrenan utilizando solo una escala y no se realiza ningún aumento de datos. Los resultados se dividieron en tres pruebas con base en modificaciones realizadas a las imágenes de entrada (segmentación, rotación y tamaño), obteniendo como el mejor desempeño un 97.31% de clasificación correcta. En relación con los datos de entrenamiento, las arquitecturas poco profundas o muy profundas pueden resultar en ajuste insuficiente o excesivo.

1.3. Justificación

Las imágenes del iris que más se utilizan en estos enfoques de reconocimiento de iris son capturadas por la cámara NIR (Infrarrojo Cercano) ya que ofrecen muy buena visibilidad de la textura del iris, incluso para regiones muy pigmentadas. Como resultado, la región del iris extraída proporciona información más precisa, lo que implica mejores posibilidades de reconocimiento [16]. Sin embargo, debido a la complejidad de la cámara NIR surge el problema de la portabilidad y la simplicidad de posibles aplicaciones. Por otro lado, los teléfonos inteligentes con cámara (sin capacidad NIR) están al alcance de todos hoy en día.

El problema con las cámaras que capturan imágenes en el espectro de luz visible resulta en imágenes del iris menos detalladas en comparación con las cámaras NIR. Así que la pregunta sería: "¿Son lo suficientemente buenas para el reconocimiento del iris?". La utilización de cámaras de uso común, por ejemplo, cámaras de fotografía o de teléfonos celulares tomadas en un ambiente controlado pueden ser utilizadas para crear sistemas de reconocimiento de iris con desempeños mayores a 95% de exactitud sin la necesidad de recurrir a cámaras NIR.

La utilización de redes neuronales en el reconocimiento de imágenes se debe a su capacidad de aprender y poder clasificar imágenes aun con ligeras diferencias, aunque pertenezcan a una misma clase, es decir, la red neuronal no generaliza en comparación de si se clasificaran las imágenes comparándolas con una imagen de referencia y obtener la correlación entre ellas o con otros métodos para la clasificación y en caso de no ser muy idénticas no son pertenecientes a la misma clase cuando en realidad sí lo son.

Como preámbulo, el objetivo de esta investigación fue el diseñar un esquema computacional para el reconocimiento del iris por medio de redes profundas, específicamente de las CNN. Se utilizaron imágenes en el espectro de luz visible, sobre las cuales se aplica una secuencia de algoritmos de preprocesamiento para posteriormente aplicarlas como entrada a la CNN y lograr su clasificación o identificación.

1.4. Contribuciones

Las principales aportaciones de esta tesis incluyen:

- La propuesta de un método de segmentación de iris capaz de lidiar con imágenes de iris capturadas en el espectro visible en condiciones ambientales no tan controladas en el aspecto de la iluminación para la captura. El método o proceso de segmentación está compuesto por los siguientes elementos: mejora de la imagen (contraste), detección de bordes Canny, transformación Hough circular.
- La propuesta de un modelo CNN para utilizar una cantidad baja de imágenes de entrenamiento para la parte de extracción de características, tratando el problema del reconocimiento de iris.
- La reducción en complejidad computacional en la arquitectura de la red neuronal convolucional, útil para poder ser implementados como sistemas embebidos a diferencia de las otras arquitecturas que cuentan con una memoria de almacenamiento mucho más grandes.
- La utilización de poca cantidad de imágenes de iris por cada clase o persona a reconocer en el entrenamiento de la red neuronal convolucional, dado que para el entrenamiento de una CNN se requieren grandes cantidades de imágenes para obtener buenas características, teniendo en cuenta que es un problema de las bases de datos existentes que son públicas para su utilización.

En el presente trabajo de tesis se desarrolló un esquema computacional de reconocimiento de iris con imágenes capturadas con cámaras en el espectro visible (por ejemplo, cámaras de celulares o cámaras fotográficas de uso común), sin utilizar cámaras de infrarrojo cercano.

Se implementó el proceso de segmentación y normalización a la imagen del ojo y se toma como entrada a las diferentes CNN para llegar a una clasificación en términos de sus características.

La imagen normalizada del iris se adapta y se elimina información considerada como ruido (oclusión por párpados) y se introduce directamente como entrada a los modelos de CNN, luego se extraen características convenientes de esta imagen para obtener la salida deseada para su clasificación o en este caso para la identificación del individuo [5].

Se propone un diseño de un modelo CNN (derivada de la arquitectura AlexNet) denominada IrisNet y se analiza su desempeño y de otras CNN existentes en ámbito de aprendizaje profundo, consistente de 7 capas: 4 capas convolucionales y 3 capas totalmente conectadas. Se modificó la arquitectura AlexNet para adecuarse a las necesidades del reconocimiento de iris en donde se trabaja para la clasificación de N clases, es decir, la identificación de N personas diferentes utilizando DL con los modelos de CNN.

1.5. Presentación de la tesis

En los capítulos siguientes se profundiza en los detalles del trabajo desarrollado y los resultados del mismo. En el capítulo 2, se presentan los diferentes aspectos relacionados con la obtención de la imagen del iris, desde su posición en el ojo hasta los problemas que impiden obtener una imagen completa del iris. En el capítulo 3, se presenta el preprocesamiento que se aplica a la imagen del iris para su correcta segmentación y normalización. El capítulo 4, describe la aplicación de redes neurales convolucionales al reconocimiento del iris a partir de su imagen normalizada. En el capítulo 5, se presentan los resultados obtenidos al entrenar el modelo IrisNet y su comparación con los modelos AlexNet y VGG. Y por último las conclusiones del trabajo realizado.

CAPÍTULO 2

RECONOCIMIENTO DE IRIS

En el presente capítulo se describen los conceptos esenciales sobre los sistemas de reconocimientos biométricos de iris. En la sección 2.1 se habla de la estructura del ojo humano. En la sección 2.2 se describen las componentes y características principales de un sistema biométrico. En la sección 2.3 se muestran los avances recientes para la utilización de reconocimiento. En la sección 2.4 se muestran las bases de datos disponibles para el reconocimiento de iris. En la sección 2.5 se presentan los problemas de que se tienen en la captura de las imágenes del iris.

2.1. Iris humano como reconocimiento biométrico

La idea de utilizar el iris como el identificador biométrico surge en 1985, cuando los oftalmólogos Leonard Flom y Aran Safir logran la patente de un sistema de reconocimiento de iris [17]. En 1994 se presenta el primer algoritmo de reconocimiento de iris y fue patentado por el Dr. John Daugman [17].

El iris puede ser observado sin necesidad de realizar una intervención invasiva al cuerpo humano, es decir, se puede observar desde el exterior, corresponde a la sección pigmentada (sección coloreada) dentro del ojo humano, posicionándose entre la pupila y la córnea como se muestra en la Figura 2.1 [18].

El iris posee una estructura muy fina que contiene cinco capas de fibra como tejido, siendo muy complejos y se revelan en varias formas. De igual manera, la superficie llega a poseer estructuras complejas como cristales, hilos finos, manchas de cueva, surcos, rayas, entre otros [18].

Sin embargo, se posee algunas desventajas que dificultan el uso del iris en sistemas de reconocimiento biométrico:

Tamaño: es pequeño aproximadamente 1 cm de diámetro.

Ubicación: es un objeto móvil localizado en una superficie curva, húmeda y reflexiva.

Oclusión: su imagen suele estar afectada por pestañas, párpados y reflejos de luz, limitando el área de interés la cual contiene la información del iris.

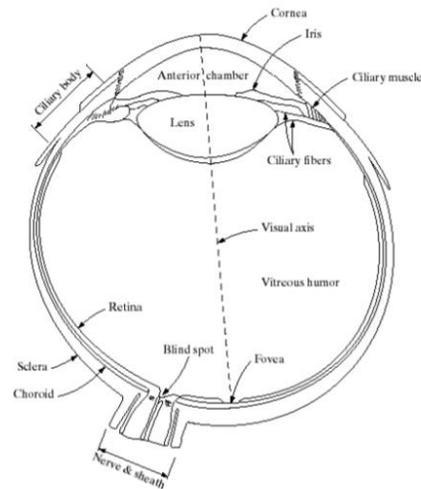
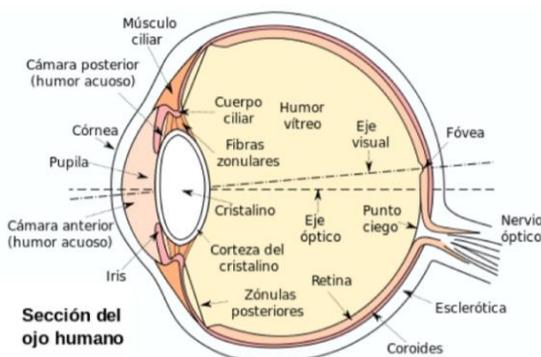
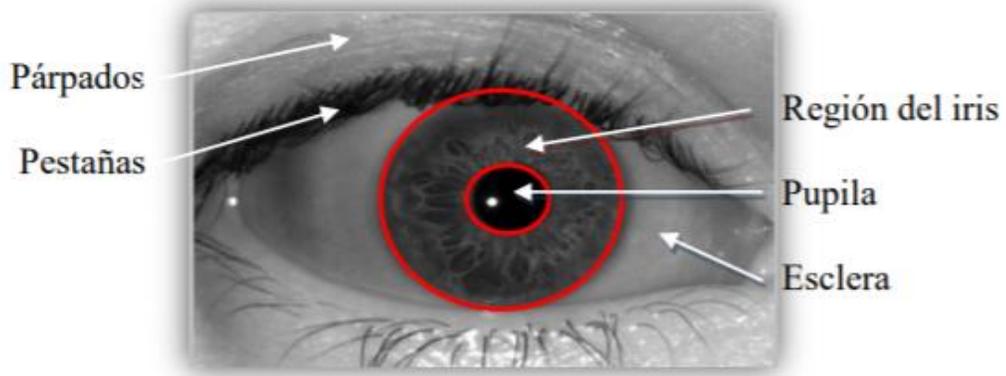


Figura 2.1. Estructura del iris en el ojo humano.

La clave en la identificación biométrica es la variación aleatoria de la naturaleza de sus características, entre diferentes personas son estadísticamente únicas y estables a lo largo del tiempo, siendo una ventaja para la utilización del reconocimiento de personas por medio del iris ya que esto es el origen de la singularidad y la base para la discriminación. Entonces, debido a las propiedades antes mencionadas, el iris humano es considerado como una de las

características biométricas más segura, confiable y adecuada para la identificación de personas, de acuerdo a [19].

En otros casos del reconocimiento biométrico, la apariencia física varía entre las personas, las huellas dactilares muestran más variación entre personas; pero en lugar de tener una textura multiescalar como los patrones de iris, su escala de estructuración está fijada por el espaciado (aproximadamente 0,5 mm) del flujo de cresta de la huella dactilar [20].

2.2. Sistema biométrico

Un sistema biométrico realiza el reconocimiento de un individuo sobre algún rasgo biológico que lo caracteriza, este rasgo biológico es llamado identificador biométrico. Algunos de los identificadores biométricos que han sido analizados son las huellas dactilares, el rostro, la voz, la geometría de la mano, la retina y el iris.

Un buen identificador es determinado por utilizar rasgos únicos, de manera que presenta una probabilidad casi nula de que exista otro individuo con el mismo patrón. Además, de no variar con el tiempo y ser fácil de capturar, básicamente los sistemas biométricos cuentan con cuatro etapas principales [21]:

- **Adquisición:** Se obtienen los datos del identificador biométrico del individuo a través de una cámara de fotos, una cámara de video, un lector de huellas digitales, etc. En esta etapa, se puede evaluar la calidad de la adquisición para favorecer el rendimiento del sistema.
- **Preprocesamiento:** Los datos adquiridos son procesados para eliminar las regiones de ruido, por ejemplo, reflexiones, oclusiones. En este último caso, más específico para el reconocimiento de iris en este proyecto es realizar un procesamiento a la imagen digital para obtener el mapa de bordes y realizar el proceso de normalización de la imagen, conservando el área útil que contiene la información para el sistema.

- **Codificación:** Se extrae la información más discriminadora, las características son extraídas en la etapa convolucional de la red neuronal, siendo prescindibles para la clasificación.
- **Comparación de códigos:** se realiza la decisión de aceptación o de rechazo como respuesta del sistema.

Se presenta el diagrama básico de un sistema biométrico. El trabajo de esta tesis adapta la etapa de codificación y comparación de desempeños y rendimiento computacional con otros modelos utilizados, como se muestra en la Figura 2.2.

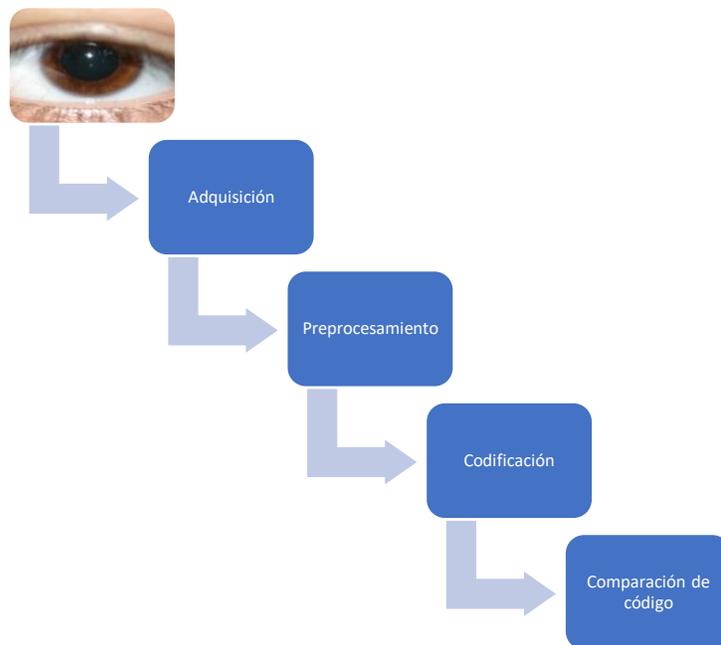


Figura 2.2. Diagrama básico de un sistema biométrico.

2.3. Avances en los sistemas de reconocimiento de iris

La idea de la utilización del iris como medida biométrica no es reciente, nace con Alphonse Bertillon [22] que presentó un método de clasificación racional de datos tomados de una persona para poder ser identificada, ya que las medidas humanas establecen una ley natural de distribución estadística.

Actualmente, después de 25 años del desarrollo de la primera patente, se han desarrollado un vasto número de algoritmos de reconocimiento de iris. No obstante, el enfoque teórico más reconocido hasta el momento ha sido el algoritmo introducido por John Daugman en [8], donde se reportan tasas de error cercanas a cero en el ámbito del desarrollo de los sistemas de reconocimiento de iris.

Las investigaciones que se relacionan con el deseo de hacer sistemas de reconocimiento del iris con menos condiciones controladas, es decir, el sistema de adquisición de la imagen del iris debe eliminar las limitaciones sobre la distancia, la iluminación, la posición y el movimiento de los usuarios.

Debido a que la mayoría de los sistemas actuales no requieren contacto con el usuario y adquieren las imágenes del iris a cierta distancia (26-36 cm), solicitan a la persona que coopere activamente con ellos, por ejemplo, permanecer inmóvil, posicionarse a una distancia para que la cámara pueda capturar de manera óptima la imagen del ojo, seguir instrucciones de indicadores de sonido o indicadores de luz para guiar sus movimientos, etc., limitándose a tener una mayor libertad de aplicaciones en los sistemas de reconocimiento [22].

2.4. Bases de datos

Una etapa difícil para los sistemas de reconocimiento biométrico es contar con una buena calidad en las imágenes de captura, en este caso, imágenes del iris que permitan ver las texturas que definen a cada individuo para tener buenos resultados al momento de definir las características y definir la clasificación del individuo.

Para ello, la implementación de un sistema de reconocimiento usando el iris como identificador biométrico requiere el desarrollo óptimo de cada una de las etapas del sistema, véase Figura 2.2. Algunos grupos de investigación se enfocan al desarrollo de una o varias etapas del sistema de reconocimiento.

Para llevar a cabo el desarrollo, los grupos de investigación requieren una base de datos de imágenes que sirva de referencia para realizar comparaciones puntuales entre los diferentes trabajos desarrollados en una o varias etapas del sistema de reconocimiento de iris.

Existen diferentes bases de datos públicas, por lo que han sido diseñadas cuidadosamente para poder estudiar y resolver los distintos problemas que se presentan al momento de implementar un sistema de reconocimiento.

La mayoría de los sistemas de reconocimiento biométrico de iris se basan en imágenes obtenidas por medio de una cámara de infrarrojo cercano (NIR, por sus siglas en inglés), de acuerdo a los artículos [23], [6], [24] se da una mejor resolución al momento de procesarlas, ya que permiten capturar detalles que una cámara de espectro visible no puede captar a diferencia de las cámaras de espectro visible.

La base de datos más referenciada en los trabajos de investigación para los sistemas de reconocimiento de iris es CASIA, es libre al público y cuenta con una gran variedad de imágenes del iris en diferentes posiciones en su captura, las imágenes están en formato JPEG, en escala de grises como se puede ver en la Figura 2.3.

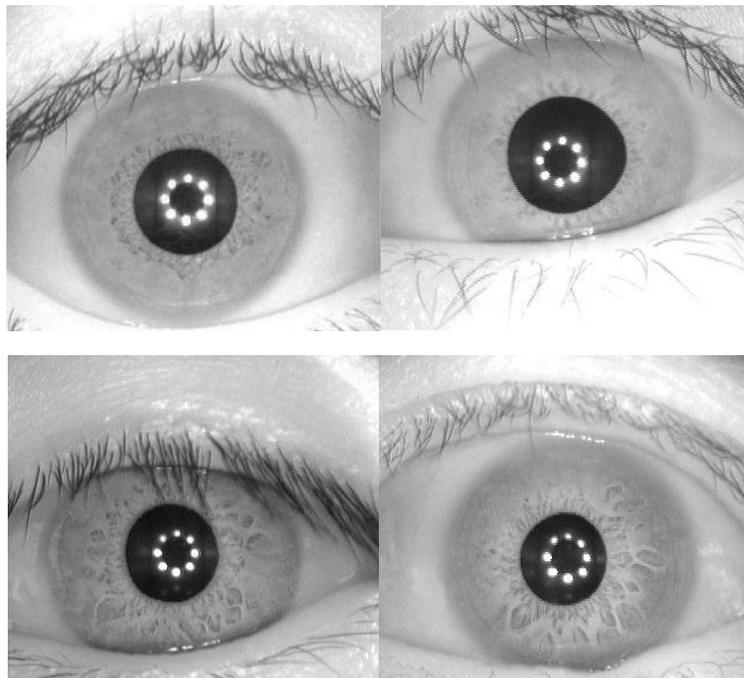


Figura 2.3 Imágenes de Iris de la base de datos CASIA obtenida con cámara NIR [25].

Además de las cámaras para la obtención de las imágenes, se reporta en [8] un sensor CMOS que utiliza una técnica de procesamiento de datos con una lógica XOR para obtener un solo bit aplicado a detección de bordes de la imagen dando como resultado una imagen binaria del iris para trabajar con ella sin dificultades.

También es posible obtener imágenes del iris capturadas en el espectro de luz visible (cámaras de celulares, cámaras de uso común), con la gran ventaja de que las cámaras de celulares son mucho más baratas en comparación con las cámaras NIR [13]. Se cuentan con bases de datos del iris en formato RGB públicas como la base de datos UBIRIS V1 [9], la base de datos UBIRIS V2 [10], UPOL (Figura 2.4) y UTIRIS [26].

El principal propósito de la base de datos UBIRIS V1 es realizar el reconocimiento automático de las personas usando imágenes adquiridas a distancia. Minimizando el grado de cooperación del usuario para lograr realizar el reconocimiento en un entorno donde el usuario no se percate que lo están identificando.

Para su adquisición se utilizó una cámara Nikon E5700 con un diámetro del lente de 71 mm, el tiempo de exposición de la imagen es de 1/30 segundo. Las dimensiones de la imagen son de 2560 x 1704 píxeles en color RGB con una resolución de 300 píxeles por pulgada.

Para difusión de las imágenes vía internet, la resolución se redujo a 400 x 300 píxeles, se realizó una transformación a escala de grises y se almacenaron en formato JPEG sin compresión, lo cual permitió obtener un menor peso de 27 Kbyte y mantener la calidad para seguir aplicándolas a los sistemas de reconocimiento.

La segunda versión de la base de datos UBIRIS V2, las imágenes fueron adquiridas de 4 a 8 metros de distancia y el usuario se encontraba en movimiento, contiene 3 x 128 imágenes de iris (personas diferentes), es decir, 3 x 64 imágenes del ojo izquierdo y 3 x 64 imágenes del ojo derecho de cada persona.

Las imágenes son de 24 bits en formato RGB, 576 x 768 píxeles, archivo en formato PNG y fueron capturadas por el dispositivo óptico TOPCON TRC50IA conectado a una cámara SONY DXC-950P 3CCD.

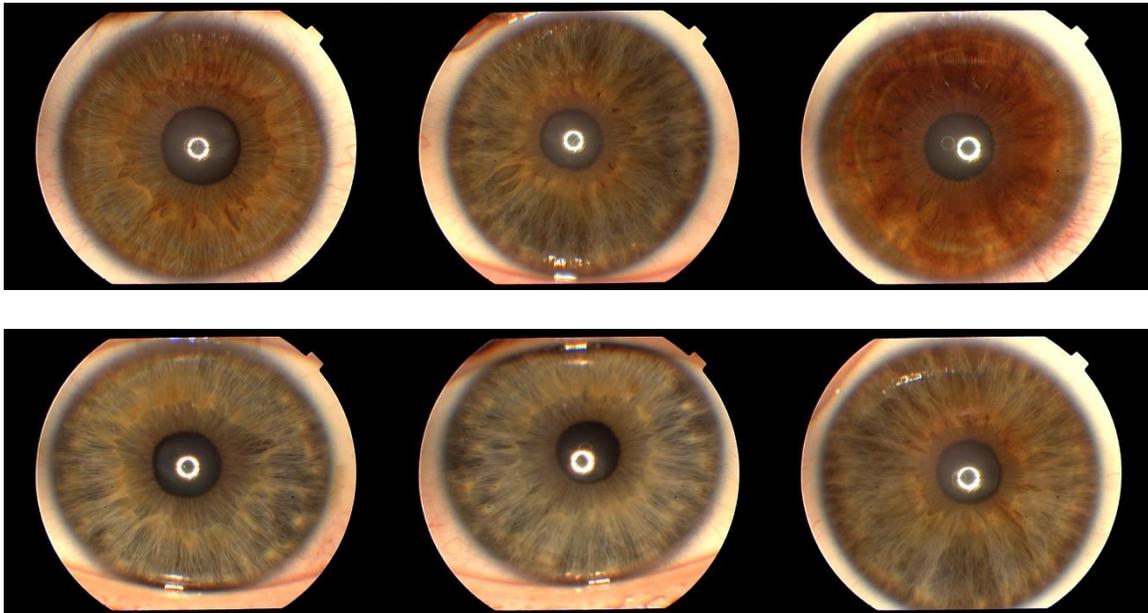


Figura 2.4 Imágenes de la base de datos UPOL capturadas en el espectro visible.

2.5. Problemas y limitaciones de los sistemas de reconocimiento biométricos

A pesar de sus evidentes ventajas, los rasgos biométricos de una persona o individuo y su representación varían considerablemente según el método de adquisición de las imágenes, en el entorno en el que se realizan las tomas de las imágenes y la forma en el que el usuario coopere con el sistema [27].

2.5.1. Factores que influyen en la calidad de la imagen

En los sistemas de adquisición de imágenes a distancia la calidad de las imágenes baja, debido a los siguientes factores: desenfoque, borrosidad por movimiento, fuera de ángulo, oclusión, reflexiones e iluminaciones. Por ello tienen un impacto negativo en el rendimiento del reconocimiento del iris.

Aunque se realice una segmentación perfecta de la región del iris, la información que provee las texturas del iris (características para la identificación del individuo) podrían ser afectadas por los problemas anteriormente mencionados.

Se muestra en la Figura 2.5 las imágenes con problemas de oclusión por parte de los párpados y las pestañas, mal posicionamiento del iris al tomar la captura de la imagen, problemas de iluminación oscureciendo la parte del iris y combinándola con la parte de la pupila, todos estos factores ocasiona un mayor pérdida de información de las características que definen a cada personas para su identificación.



Figura 2.5 Problemas que se presentan en la calidad de las imágenes.

2.5.1.1. Desenfoco en la imagen

El enfoque de la imagen es un factor óptico que depende de la profundidad de campo ubicada en el espacio por delante y por detrás del plano de enfoque. Por otro lado, el desenfoque ocurre cuando el objeto de interés está fuera de la profundidad de campo, cuanto más lejos se encuentre el objeto de la profundidad de campo, más alto será el grado de desenfoque [28].

Provocando que los detalles pequeños pierdan claridad o nitidez, en este caso lograr distinguir las regiones del iris y de la pupila con mayor facilidad. En la Figura 2.6 se muestra un ejemplo de una imagen desenfocada lo cual no se pueden observar con claridad las texturas del iris ni distinguir la parte de la pupila.

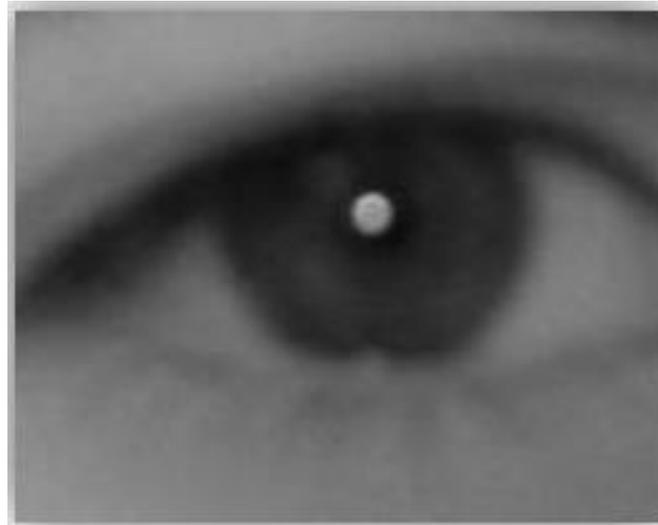


Figura 2.6 Imagen del ojo desenfocada por la captura.

2.5.2. Consideraciones sobre la iluminación

La problemática que existe en la captura del iris es en establecer un compromiso entre el suficiente contraste en patrones oscuros frente a niveles de iluminación no molestos para los usuarios.

Principalmente la iluminación provoca la dilatación de la pupila de modo que la zona del iris que es útil para obtener la información se reduce, en cambio, la falta de iluminación hace que la pupila se contraiga, ver Figura 2.7 y Figura 2.8.



Figura 2.7 pupila contraída y dilatada, esquema explicativo.



Figura 2.8 Ejemplos reales con iluminación ambiental, pupila dilatada (derecha) y pupila contraída (izquierda).

2.5.3. Posicionamiento del iris

Para obtener mejores resultados en la captura de imágenes, los usuarios deben ser cooperativos para situar adecuadamente su iris en un ángulo correcto y así obtener más información en cuanto a las características del iris. El factor de ángulo mide la orientación del iris respecto a la cámara, asumiendo que la vista frontal del iris tiene una forma circular y la vista fuera de ángulo se convierte en elipse por su posicionamiento como se ve en la Figura 2.9.

La vista fuera de ángulo no permite que el sistema de reconocimiento realice la identificación de la persona adecuadamente [29].

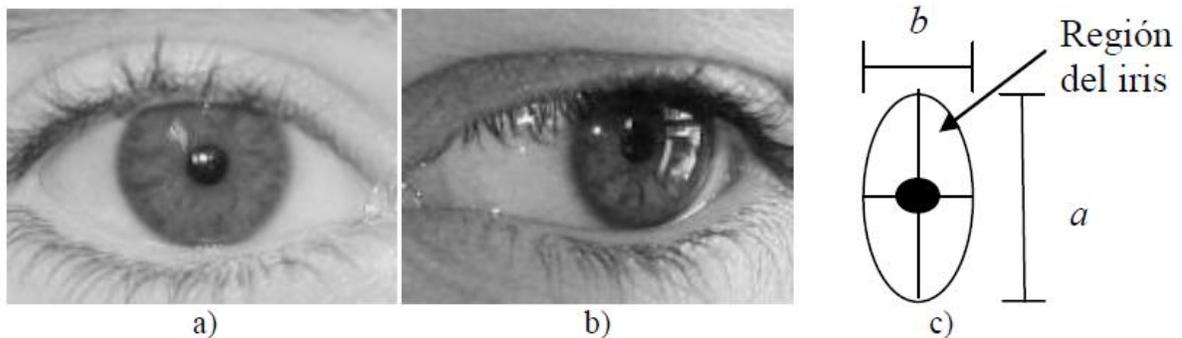


Figura 2.9 a) imagen con vista frontal. b) imagen fuera de ángulo. c) esquema de visión del iris fuera de ángulo.

CAPÍTULO 3

SEGMENTACIÓN DE LA REGIÓN DEL IRIS

La primera parte del reconocimiento de iris es localizar la región de interés del iris de la imagen digital capturada, se obtiene al considerarla como dos círculos concéntricos, un círculo enfocado para formar el límite del iris y el segundo círculo para formar el límite de la pupila.

Las pestañas, párpados y reflexiones se presentan como oclusión del iris, de tal forma que representan problemas para el reconocimiento de patrones al extraer las características, por lo cual la correcta segmentación depende mayormente de la calidad de las imágenes, así las imágenes que contengan reflexión especular pueden dificultar el proceso de segmentación. El proceso de segmentación es muy crítico ya que los datos que se han localizado incorrectamente darán como resultado tasas de reconocimiento muy bajas.

En esta etapa la imagen es procesada para localizar la región del iris y eliminar lo que no es útil, esto es con el objeto de elevar el rendimiento del sistema de reconocimiento de iris. También se normaliza el tamaño de la región del iris, debido a que generalmente varía de una imagen a otra; esto es causado por la variación de los niveles de iluminación que generan contracción o dilatación del iris, debido al cambio de distancia existente entre el ojo y la cámara o por el movimiento relativo del ojo.

Este proceso de normalización de la imagen al realizar una transformación a la imagen y obtener una imagen rectangular como resultado, permite realizar o extraer de manera más sencilla las características al tener solo el área de interés.

3.1. Preprocesamiento de la imagen

La etapa de preprocesamiento se encarga de extraer la región del iris del ojo y eliminar el ruido que afecte al entrenamiento del modelo CNN. Estos métodos forman parte del grupo de algoritmos implementados en el desarrollo de este trabajo de tesis, mostrados en la Figura 3.1.

La imagen seleccionada de la base de datos de la persona a identificar es necesario realizar el preprocesamiento de esta para la localización del iris. Los pasos que se realizan en el preprocesamiento de la imagen se definen en la Figura 3.1; entre ellos realizar un difuminado

en la imagen para poder resaltar los bordes, mejorar el contraste de la imagen para distinguir más la pupila del iris y aplicar el detector de bordes.

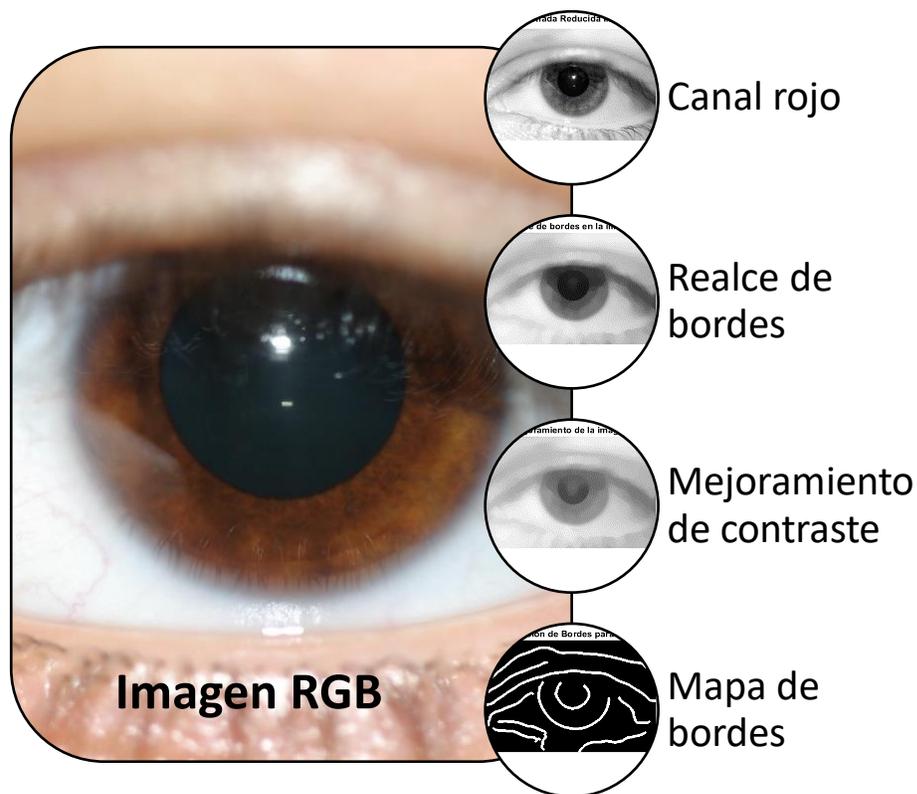


Figura 3.1 Pasos de procesamiento de la imagen para la localización del iris y de la pupila.

3.1.1. Selección de la imagen

Las imágenes provenientes del conjunto de la base de datos UTIRIS V1 se encuentran en formato RGB, (ver Figura 3.2) es decir, capturadas en el espectro visible, dando como resultado una imagen de 3 canales (rojo, verde, azul) por lo que se convierte a un solo canal para su procesamiento.

Se selecciona la capa roja del formato RGB para la extracción de características del iris como se muestra en la Figura 3.3, debido a que al trabajar en la capa de color rojo se tiene una longitud de onda visible muy cercano al infrarrojo de alrededor de los 640 nm, incluso para imágenes de iris oscuro utilizando el canal rojo se aprecia que se tiene un alto contraste entre el iris y la pupila [16].



Figura 3.2 Imagen sin procesar en formato RGB tomada de la base de datos UTIRIS V1.



Figura 3.3 Canal rojo de la imagen original.

3.1.2. Realce de bordes

Para obtener los bordes de la imagen para determinar los círculos pertenecientes al iris y a la pupila, los bordes de interés en este caso son los más superficiales del ojo, ya que aún no se pretende hacer la búsqueda de las características que poseen las texturas del iris.

Para disminuir los bordes con más detalles y definir los bordes en la imagen como lo son los bordes del iris y la pupila, se realiza un proceso de difuminado en la imagen aplicando un filtro de mediana sobre toda la imagen varias veces para eliminar los detalles. Para el resultado final, se determinó de manera experimental la aplicación de 200 veces el filtro, donde el

resultado de aplicar el primer filtro se aplica nuevamente a la salida de la imagen, y así sucesivamente hasta completar los 200 filtros aplicados a la imagen original.



Figura 3.4 Aplicación del filtro de mediana.

Este filtro funciona con la operación no lineal, en la que se toma un valor de pixel del valor de los pixeles circundantes. El cálculo se realiza ordenando el grupo de valores de intensidad de pixel y luego reemplazando los valores de píxel se procesan por el valor medio de los resultados de la secuencia [30]. La ilustración del proceso de mediana se muestra en la Figura 3.4.



Figura 3.5 Mejoramiento de bordes para detectar los círculos del iris y de la pupila.

Los detalles innecesarios en la Figura 3.5 se ven difuminados a diferencia de la Figura 3.3, lo cual al tratar de detectar los bordes en la imagen se ven reducidos considerablemente los detalles o bordes encontrados que pueden ocasionar ruido al momento de encontrar las circunferencias deseadas del iris y la pupila (ver Figura 3.9 a), dando como resultado un mapa de bordes más limpio.

3.1.3. Mejoramiento de la imagen

Las bases de datos de iris capturadas en el espectro de luz visible contienen imágenes con problemas de iluminación que hace más difícil el procesamiento de las imágenes al tratar de crear el mapa de bordes para encontrar las circunferencias que definan los límites del iris y de la pupila como se menciona en secciones anteriores.

En la Figura 3.6 se muestra el ejemplo de una imagen donde la región de la pupila es muy similar en contraste a la región del iris. En el momento de obtener el mapa de bordes (Figura 3.9) la circunferencia de la pupila se ve afectada por la iluminación y a casusa de ello el borde que la delimita no se compone de una solo línea conectada.



Figura 3.6 Problemas de iluminación.

Para los problemas de iluminación se implementa una transformación lineal sobre la imagen oscura, se compone visualmente de 3 líneas rectas unidas con diferentes pendientes como se ve en la Figura 3.7 que ayudan a hacer un mapeo de los pixeles de la imagen a un nuevo espacio con sus valores correspondientes.

Cada recta define los cambios de los pixeles, la imagen se divide en 3 secciones de tonalidades, donde los valores de los pixeles más oscuros sufren los mayores cambios de los valores de los pixeles por la pendiente, los pixeles con tonalidades grises se mantienen más

constantes al tener una pendiente cercana a 45 grados y por último los valores de los pixeles más blancos se mantienen con el mismo valor.

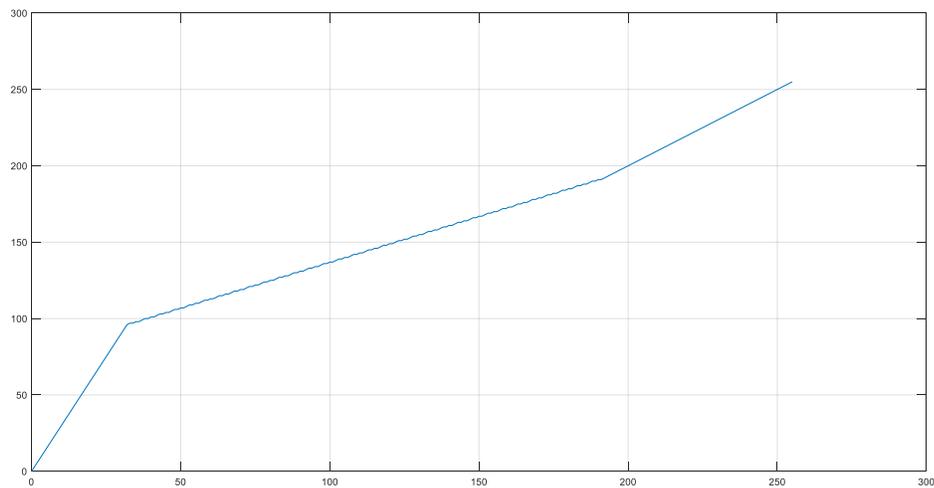


Figura 3.7 Transformación lineal para mejorar el contraste.

Para obtener los puntos para obtener los nuevos valores de los pixeles se realiza una tabla de búsqueda (Look-up Table, LUT), donde el valor correspondiente a un pixel en la imagen y se busca en el eje de las abscisas y el valor nuevo correspondiente a ese pixel será el valor indicado por el eje de las ordenadas, como resultado se obtiene una imagen mejor distribuida con los valores de los pixeles como se muestra en la Figura 3.8.

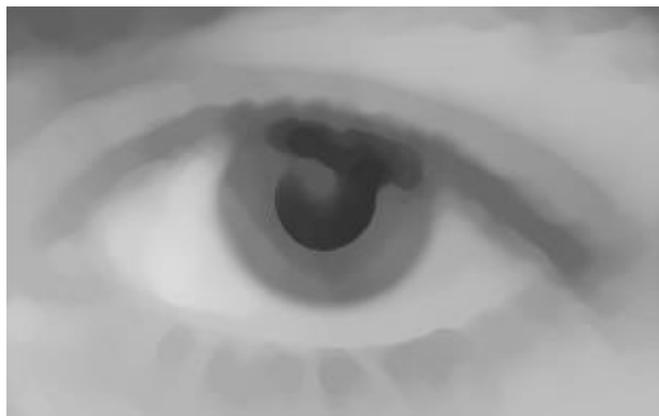


Figura 3.8 Mejoramiento de contraste de la imagen.

3.2. Mapa de bordes

Un mapa de bordes suministra información valiosa sobre los límites de los objetos y puede ser utilizada para segmentar una imagen o también en reconocimiento de patrones, en donde la mayoría de las técnicas para detectar bordes emplean operadores locales basados en distintas aproximaciones de la primera y segunda derivada de los niveles de gris de la imagen [31].

3.2.1. Filtro de Canny para detectar el mapa de bordes

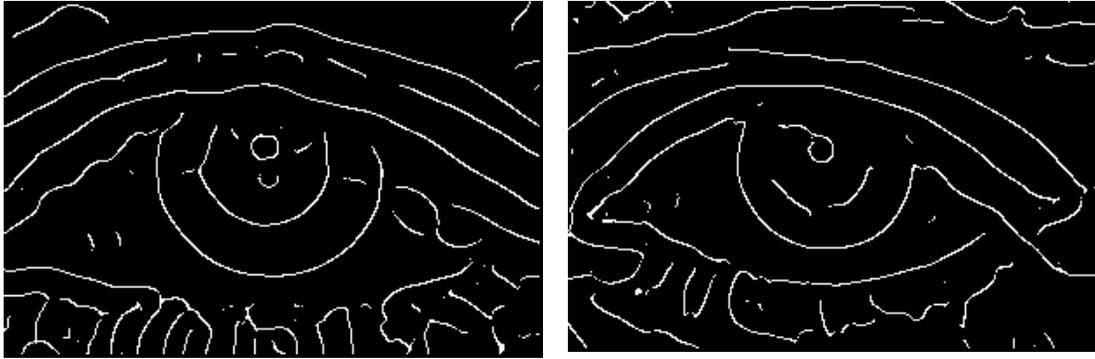
El objetivo del filtro de Canny es localizar de manera óptima los bordes en una imagen y se implementa para poder utilizar la transformada de Hough. El filtraje se lleva a cabo sobre la imagen con realce de bordes y con el mejoramiento en el contraste por falta de iluminación en las imágenes.

La localización se establece en dos etapas: localización de la pupila y localización del iris, para facilitar el proceso de localización para el sistema al tratar de buscar por la imagen información que se pueda omitir fácilmente.

La consistencia visual del color y el contorno de la pupila permite aplicar métodos fiables para calcular el centro del iris. Muchos documentos de reconocimiento de iris exploran una variedad de algoritmos de localización de pupilas; algunos métodos intercambian precisión por velocidad, y viceversa.

En la Figura 3.9 se muestran unos ejemplos de los mapas de bordes obtenidos con los problemas de iluminación y realce de bordes que se tienen en algunas de las imágenes de la base de datos, dando como resultados imágenes con bordes no necesarios para la localización del iris y de la pupila, además se obtienen bordes separados que no completan la línea para su correcta localización.

Las imágenes con el procesamiento de realce de bordes y el mejoramiento en el contraste se logra obtener correctamente el mapa de bordes. Como se menciona anteriormente, se realiza un procesamiento sobre el mapa de bordes para definirlos y ser más fácil de localizar al aplicar la transformada de Hough.



a)

b)

Figura 3.9 a) Mapa de bordes sin difuminado, b) Mapa de bordes con problemas de iluminación sin difuminado y mejoramiento de contraste.

La operación morfológica de dilatación ayuda a unir las líneas que se encuentran separadas por pocos píxeles, con lo cual se obtiene la ventaja de resaltar mejor los bordes de la imagen, dando como resultado la Figura 3.10.



Figura 3.10 Mapa de bordes aplicando contraste y difuminado.

Para casos en que la información de la pupila se pierde al crear el mapa de bordes o viceversa, se utiliza el mapa de bordes para la localización del iris, y para la pupila se utiliza un método de umbral, se asigna un umbral donde solamente se obtiene la parte de la pupila lista para obtener la circunferencia

Aprovechando la baja saturación de la pupila, se filtra razonablemente todo menos las porciones más oscuras de la imagen en escala original de grises (pupila, pestañas, cejas,

lunares), generando una imagen binaria basada en un valor de umbral de baja intensidad de pixeles como se muestra en la Figura 3.11.

El tamaño del blob que representa la conversión binaria posterior de la pupila suele ser mucho mayor que el tamaño del ruido detectado (pestañas, cejas, lunares, etc.), que luego se puede filtrar de la imagen binaria eliminando todos los blobs que tengan un área de pixeles debajo de algún valor razonable.

Luego, utilizando la transformada de Hough circular a través de la imagen binaria con una estimación del rango del radio de la pupila razonable, podemos obtener el círculo que devuelve la mayor magnitud de los picos de la matriz del acumulador sobre el blob de pupila binario aislado, lo que nos da una muy buena estimación del radio y la ubicación de la pupila.



Figura 3.11 Sección de la pupila por medio de un umbral.

De esta manera los círculos de la pupila se buscan en dos mapas de bordes dado caso que no se detecte en una de las dos imágenes binarias, teniendo prioridad la imagen generada por el mapa de bordes y asegurar encontrar la circunferencia de la pupila.

Los cambios por iluminación se presentan en ocasiones, por lo tanto, las imágenes establecidas por umbral no se definen correctamente o por detección de bordes se genera un mapa con líneas separadas evitando que determinar una circunferencia con la transformada de Hough.

3.3. Transformada de Hough

La mayoría de los trabajos están basados en la utilización de los métodos de Daugman y Wildes. Estos algoritmos se basan en que el iris posee una forma circular, buscando siempre círculos dentro de la imagen. El método de Daugman [8] se basa en el uso de un operador integral diferencial; trabaja en el espacio de tres parámetros y busca las coordenadas del centro y del radio correspondiente al círculo encontrado. El algoritmo inicia con una búsqueda amplia para luego reducirla a un área de interés. El método de Wildes [32] trabaja con imágenes con bordes binarios establecidos, después se utiliza la transformada de Hough para detectar círculos dentro de la imagen binaria establecida.

La principal desventaja de estos dos métodos es la búsqueda exhaustiva que deben realizar para determinar la posición del centro del círculo y el radio de este. En casos en que un iris no posea una forma perfectamente circular, como en casos donde se encuentra el ojo con alguna rotación, aumenta la probabilidad de error de los métodos clásicos, que buscan círculos perfectos.

La búsqueda de dos círculos mediante la metodología de Wildes puede ayudar a reducir el problema, delimitando cada costado del iris por una de las circunferencias. Esto redobla el tiempo de búsqueda, pero mejora la efectividad de los algoritmos tradicionales [33].

La transformada de Hough es una herramienta de análisis y técnica de procesamiento de imágenes estándar que se utiliza para encontrar curvas y formas que se pueden representar en una forma paramétrica como líneas, círculos y polinomios dentro de una imagen [34]. El reconocimiento de algún patrón global se logra utilizando los patrones locales, por ejemplo, un círculo puede ser detectado en una imagen utilizando los bordes destacados de una imagen como patrones locales y buscando el valor máximo de la transformada de Hough circular.

Esta fase está orientada a conseguir la región del iris acotada entre la pupila y el borde exterior del iris, es la más importante, pues de ella depende la entrada del modelo CNN sea la más adecuada en cuanto a información contenida en características.

Para ubicar los límites del perímetro de la pupila y del iris, es fundamental obtener las coordenadas del centro del círculo y la longitud del radio. Este método usa la información del

mapa de bordes para transformar un conjunto de puntos que caracterizan un círculo a un conjunto de votos que son acumulados en un espacio paramétrico que representa al centro y a la longitud del radio, esto se muestra en la Figura 3.12.

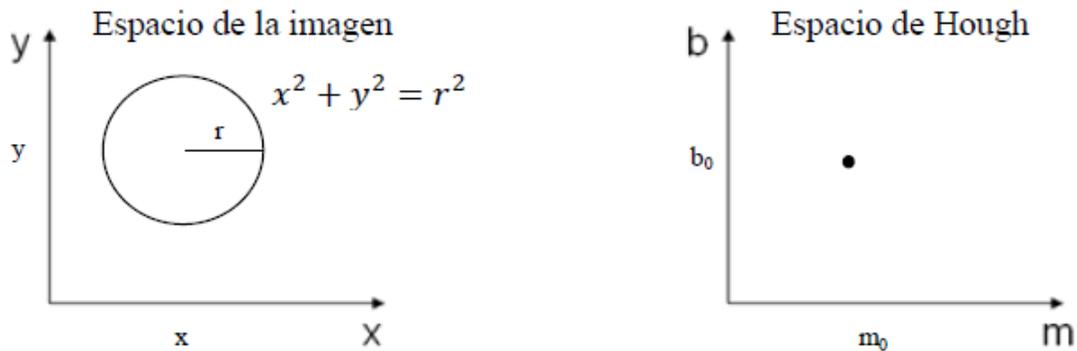


Figura 3.12 Espacio de la imagen convertida al espacio de Hough.

En otras palabras, este método toma la posición del punto en el mapa de bordes para determinar los valores del radio y las coordenadas del centro del círculo a través del cálculo de la ecuación (3.1) del círculo.

$$x^2 + y^2 = r^2 \quad (3.1)$$

El algoritmo requiere como información la posición de los píxeles en el mapa de bordes y un rango de radios para realizar la búsqueda del radio óptimo de los círculos que delimitan al iris, para ello se realiza una búsqueda en diferentes radios y asegurar el cubrir todas las regiones del iris de diferentes tamaños de las imágenes de la base de datos utilizada.

Después se localiza el círculo que delimita a la sección de la pupila de igual manera que en la región del iris, en este caso para asegurar la detección de la pupila en los diferentes tamaños y en los casos que presente alguna dilatación.

En la Figura 3.13 se muestran las circunferencias encontradas en el mapa de bordes de la Figura 3.10 al aplicar la transformada de Hough varias veces con diferentes radios de búsqueda, lo que devuelve los radios y los centros de todas las circunferencias encontradas en el mapa de bordes, ambas secciones del iris y de la pupila con sus respectivas etiquetas de

colores para identificar con facilidad el círculo que representa a cada radio de búsqueda establecido. En casos donde no se encuentra una circunferencia en un radio específico el círculo no se dibujará en la imagen.

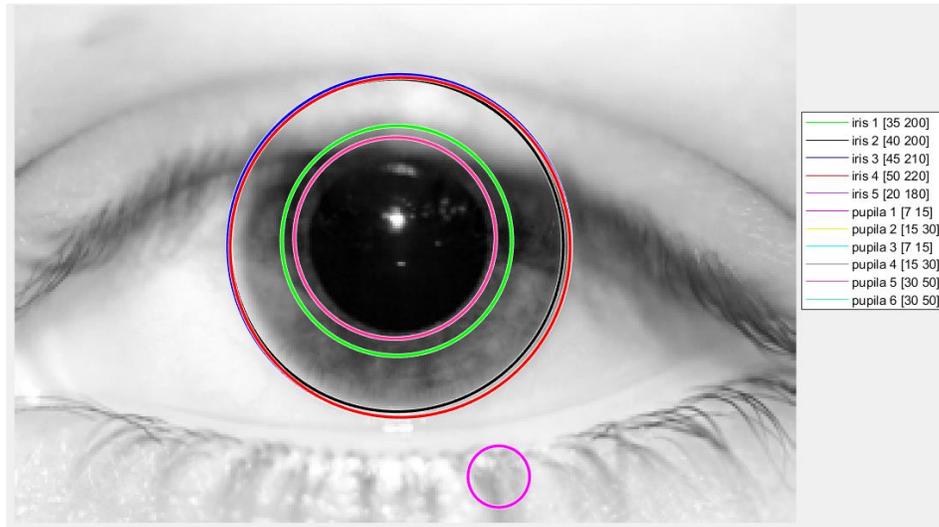


Figura 3.13 Circunferencias encontradas con una pupila dilatada.

Un problema que se presenta es la selección del círculo adecuado que delimite con más exactitud los límites deseados para no contener información que afecte la extracción de características para el desarrollo del modelo CNN.

El objetivo debido a los múltiples círculos que se encuentran en el mapa de bordes es seleccionar el círculo que contenga menos información no necesaria; como lo es la parte blanca del ojo, piel por oclusión de párpados, etc., y para el caso de la pupila la selección del círculo que no elimine información característica del iris.

Para ello, una máscara binaria circular con los radios y los centro encontrados en la transformada de Hough, (un ejemplo se muestra en la Figura 3.15 a) se compara con una imagen binaria de la Figura 3.14 mostrada en la Figura 3.15 b). De ambas imágenes se realiza una resta absoluta de la imagen y se suman los pixeles blancos restantes de la imagen dando como resultado un valor numérico que indicará el círculo ganador con el valor de radio más pequeño.

Un ejemplo de las circunferencias encontradas en el mapa de bordes sobre una imagen seleccionada de la base de datos se muestra en la Figura 3.14 a), se tiene una pupila un poco más pequeña que en la Figura 3.13, dando como resultado diferentes circunferencias encontradas muy cercanas unas de las otras en ambas regiones.

Otro ejemplo de circunferencias encontradas en donde las regiones encontradas del iris se muestra en la Figura 3.14 b), donde las circunferencias están muy cercanas una de las otras inclusive dentro de la región del iris que quita información importante para la clasificación. Las regiones de la pupila igualmente se encuentran dispersas una de las otras, facilitando la selección de las regiones adecuadas con la creación de las máscaras en la Figura 3.15 y Figura 3.16 para las regiones del iris y de la pupila respectivamente.

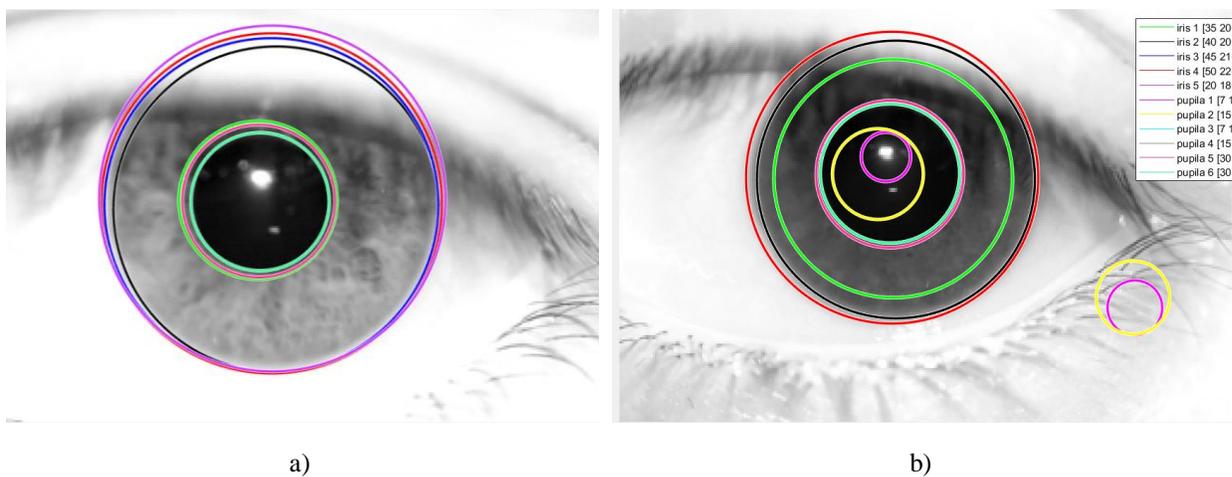


Figura 3.14 Ejemplo de circunferencias encontradas de las regiones del iris y de la pupila., a) Ejemplo circunferencias encontradas con una pupila mucho más pequeña que la región del iris, b) Ejemplo circunferencias encontradas con una pupila dilatada.

Las máscaras de las regiones del iris y de la pupila se crean mediante las funciones `Createamask` y `Createmask2` respectivamente. Los parámetros que se quieren son el centro del círculo y el radio, necesarios para definir la región binaria del círculo que representa la circunferencia encontrada y poder compararla con la imagen binaria original, la imagen del canal rojo. Además, `Createamask` utiliza la imagen con modificación de contraste y determina por medio del histograma la imagen para generar la máscara.

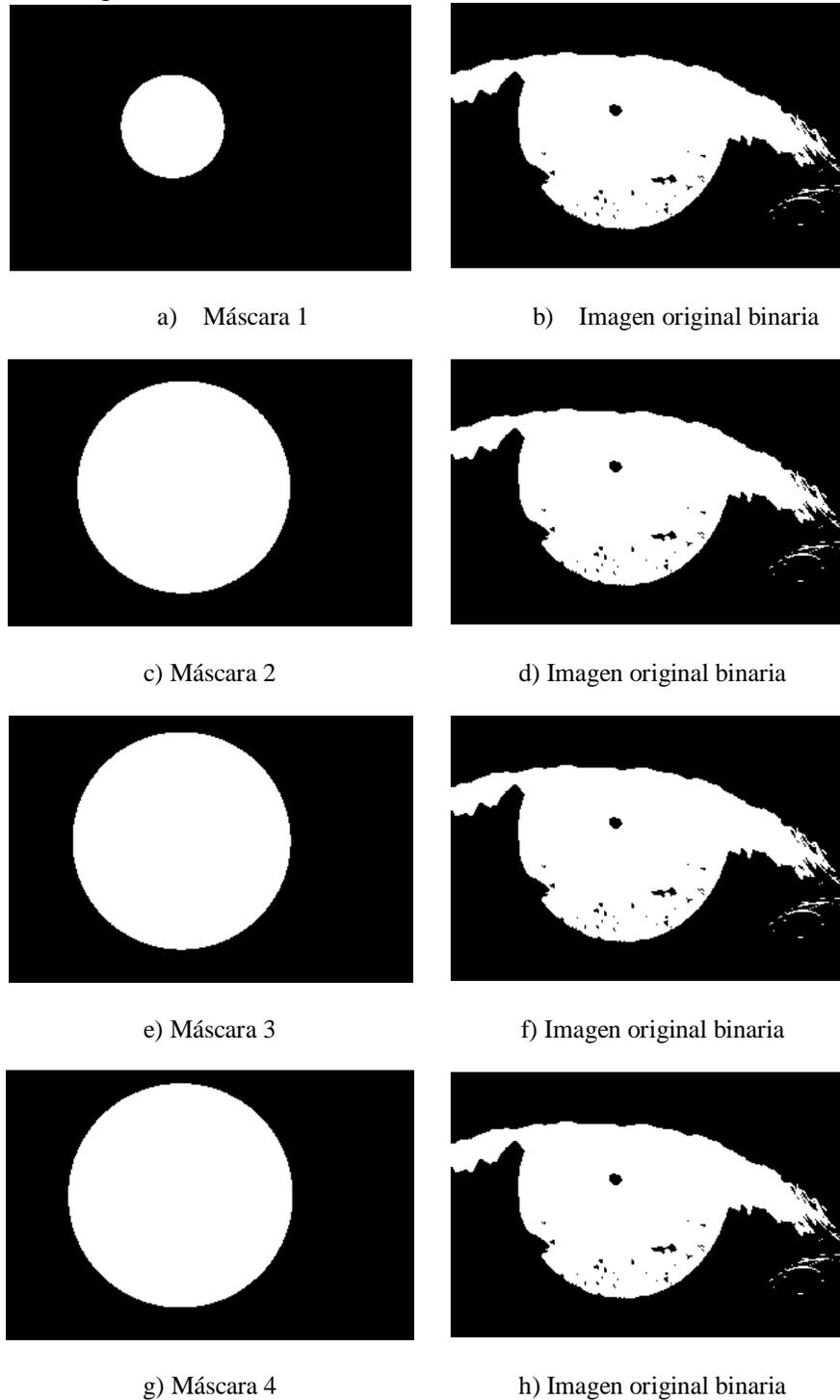


Figura 3.15 Máscaras de las circunferencias encontradas del lado izquierdo, imagen binaria del ojo humano a comparar del lado derecho.

Para la pupila, las máscaras binarias de los círculos con los radios y centros encontrados en la transformada de Hough, un ejemplo mostrado en la Figura 3.16 a), y comparándola con la

imagen binaria con un umbral específico mostrada en la Figura 3.16 b), la selección del radio de la imagen resultante con la menor cantidad de valores de los pixeles blancos totales, siendo la operación anterior una resta absoluta.

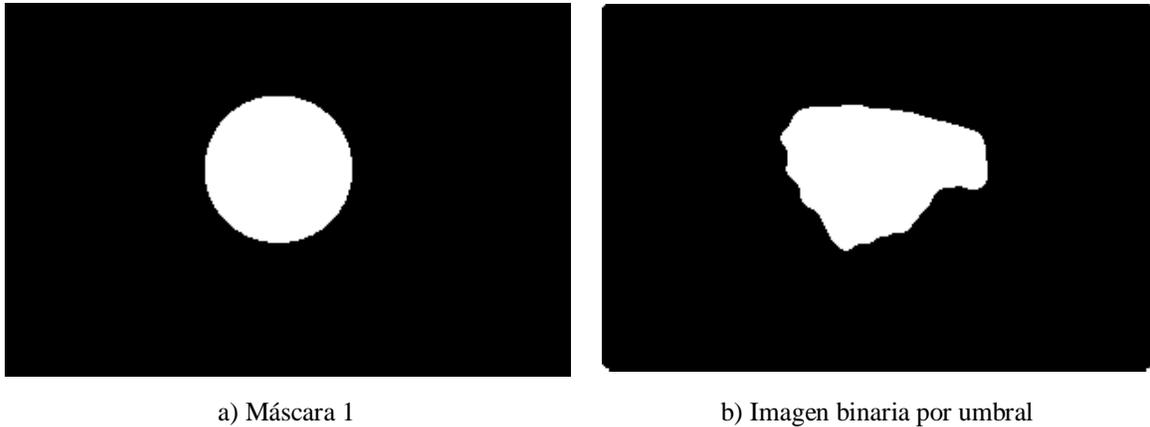


Figura 3.16 a) Máscara de la circunferencia con el radio y centro encontrados, b) máscara de la pupila de la imagen del iris.

Las circunferencias en ambas regiones más aptas con información característica se indican con una etiqueta en la imagen, lo cual dan una previsualización de la segmentación del iris que es requerida para el modelo CNN sin la intervención de información que no contenga las características del iris.

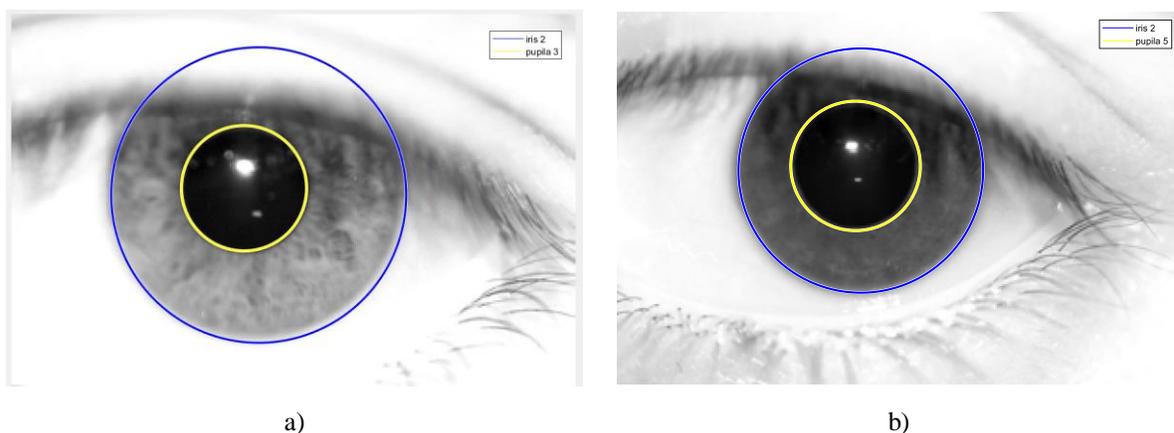


Figura 3.17 circunferencias definidas que delimitan con más exactitud las regiones del iris y pupila.

Los dos círculos corresponden las regiones que delimitan con más exactitud al resto de los círculos en el mapa de bordes con la utilización de la transformada de Hough como se muestra en la Figura 3.17 a) y en la Figura 3.17 b).

3.4. Normalización

La normalización se realiza una vez localizadas las regiones del iris y pupila en la imagen de entrada, el objetivo es la obtención de una imagen con dimensiones fijas permitiendo realizar comparaciones fácilmente en un mismo espacio.

La inconsistencia de las dimensiones a las imágenes sin normalizar es debido a la dilatación de la pupila por cambios de iluminación modificando el área de interés de la región del iris y variaciones de distancia en la que se realiza la adquisición de la imagen de entrada [4].

De modo que el proceso dará como resultados regiones de iris con las mismas dimensiones, por ejemplo, si se cuenta con dos imágenes del mismo iris se puede realizar la comparación de las regiones del iris al obtener el proceso de normalización, ya que se obtendrá las regiones del iris localizadas en una misma coordenada espacial [4].

Un punto que destacar es que las regiones del iris y la pupila no siempre son concéntricas, por lo que se tiene que tomar en cuenta si se quiere normalizar en forma de dona ya que no se va a tener un radio constante.

Introducido por Daugman [8], se realiza una transformación en la región delimitada por la pupila y el iris, consiste en hacer una reasignación de cada punto dentro de la región del iris en coordenadas de un sistema cartesiano (x, y) a un sistema normalizado de coordenadas polares (r, θ) , basado en el número de muestras de radio establecidas por un rango de $[0,1]$ y en las muestras de la dirección theta de en un rango de $[0,2\pi]$.

Se obtiene la misma cantidad de puntos en la dirección del radio a lo largo de la cantidad puntos establecidos en la dirección de theta, posteriormente se realiza la transformación espacial de los píxeles en la imagen del canal rojo, como resultado se genera una imagen rectangular de $r \times \theta$ como se muestra en la Figura 3.18.

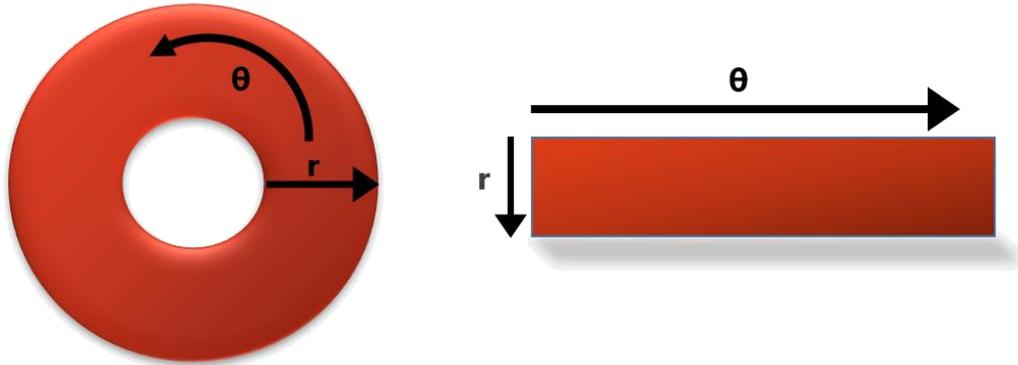


Figura 3.18 Proceso de Normalización.

Se realiza reajuste de la normalización que puede ser modelado como se muestra a continuación:

$$I(x(r, \theta), y(r, \theta)) \rightarrow I(r, \theta) \quad (3.2)$$

para encontrar los puntos en coordenadas polares se siguen las siguientes ecuaciones:

$$x(r, \theta) = (1 - r)x_p(\theta) + rx_i(\theta) \quad (3.3)$$

$$y(r, \theta) = (1 - r)y_p(\theta) + ry_i(\theta) \quad (3.4)$$

donde $I(x, y)$ es la región del iris en sus coordenadas cartesianas (x, y) , (r, θ) son las coordenadas en el sistema polares ya normalizadas, (x_p, y_p) son las coordenadas de los límites de la pupila y (x_i, y_i) son las coordenadas de los límites del iris, ambas coordenadas en dirección de theta θ .

En la Figura 3.19 se muestra una imagen del iris normalizada previamente obtenida la localización de las regiones del iris y de la pupila, donde en la Figura 3.19 a) se muestran los puntos obtenidos a lo largo del radio y las muestras a lo largo de la dirección en theta, dando como resultado una imagen rectangular de dimensiones de 6×20 .

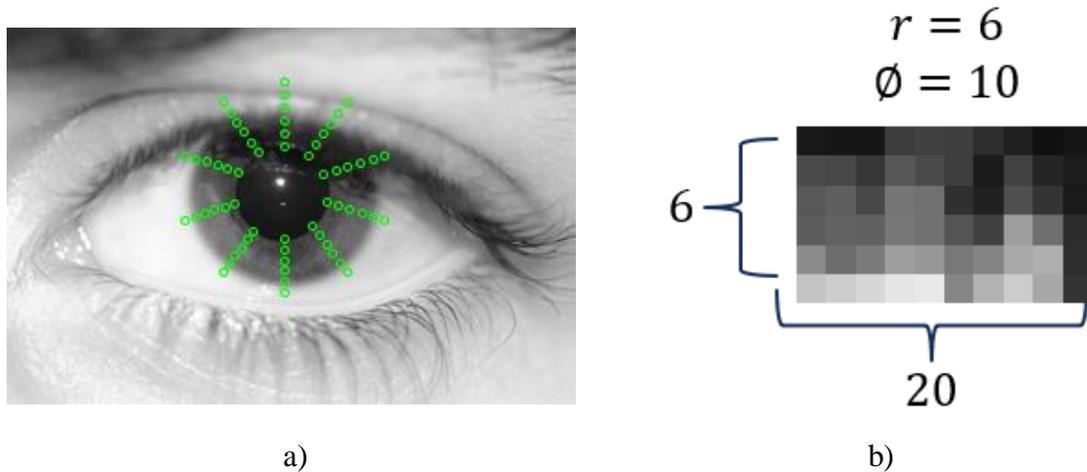


Figura 3.19 Demostración del proceso de normalización con pocas muestras en ambas direcciones, a) visualización de las muestras obtenidas en el proceso de normalización, b) resultado final.

3.4.1. Proceso de Normalización

Una vez localizada la región del iris a través de la transformada de Hough, el proceso de normalización delimita las regiones para obtener los puntos a lo largo de la circunferencia aun siendo no concéntricas ambas regiones.

Se localizan los puntos que delimitan la circunferencia del iris con base en la cantidad de muestras establecidas de ángulo theta, por medio de las siguientes ecuaciones:

$$xR = y_i + r_i * \cos(\text{angles}) \quad (3.5)$$

$$yR = x_i + r_i * \text{sen}(\text{angles}) \quad (3.6)$$

$$xL = y_i - r_i * \cos(\text{angles}) \quad (3.7)$$

$$yL = x_i - r_i * \text{sen}(\text{angles}) \quad (3.8)$$

donde x_i, y_i son las coordenadas del centro de la región del iris, r_i es el radio correspondiente a la región del iris y el argumento angles es un vector que contiene los valores establecidos de $[0, \pi)$.

Con las ecuaciones (3.5) y (3.6) se calculan las coordenadas de los puntos de la mitad de la circunferencias y las ecuaciones (3.7) y (3.8) se utilizan para calcular la segunda parte que compone a la circunferencia del iris, de esta manera las intersecciones con la circunferencia de la pupa se presentan.

Las coordenadas de los puntos que delimitan la región del iris con theta de 5 muestras, se presenta en la Figura 3.20, con esta cantidad se logran visualizar los puntos que delimitan la región en ambas partes de la circunferencia.

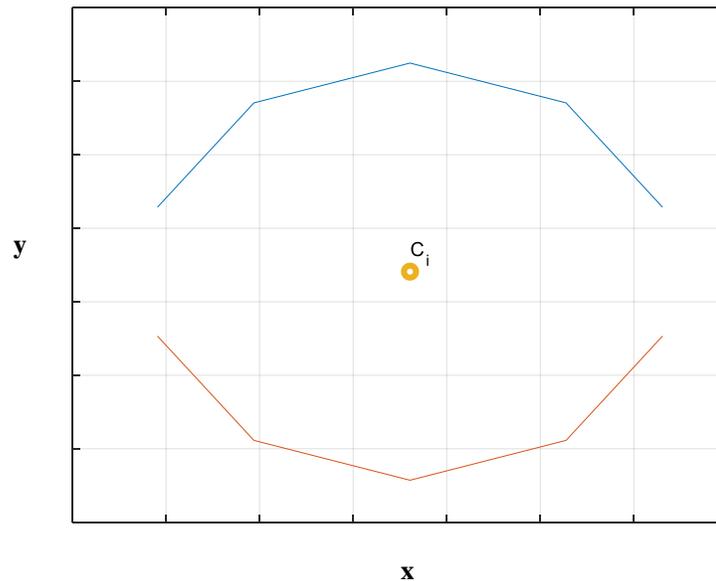


Figura 3.20 Coordenadas del límite del iris con base en el número de muestras de ángulo establecidos.

Los círculos de las regiones del iris y de la pupila no siempre son concéntricos, por consiguiente, los puntos de la pupila que se encuentren en la misma dirección de theta para cada punto de las muestras de la región del iris son los requeridos. Para esto, los cálculos de una línea recta de un punto desde el límite del iris hasta el centro de la región del iris son necesarios para obtener la intersección con la circunferencia de la pupila, esto se muestra en la Figura 3.21.

Las intersecciones con las líneas rectas del centro del iris a cada punto correspondiente generan los puntos de interés de la pupila en la misma dirección del iris sin importar si las circunferencias son concéntricas o no, esto se muestra en la Figura 3.21. Los puntos de interés

de la región de la pupila con base en la cantidad de puntos del ángulo que se establecen para el proceso de normalización.

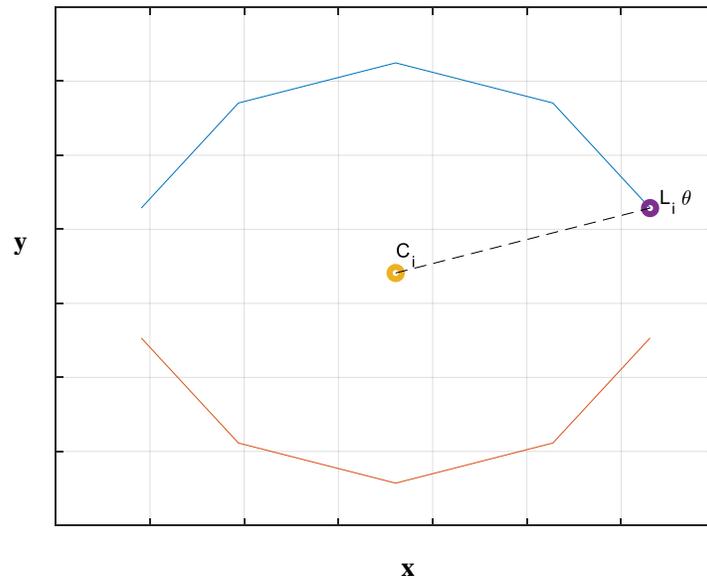


Figura 3.21 Cálculos de línea recta para encontrar la intersección con los puntos de la pupila.

En la Figura 3.22 a) se muestran los puntos correspondientes de las intersecciones que se tienen con las primeras líneas proyectadas en ambas partes de la circunferencia, generando los puntos de interés.

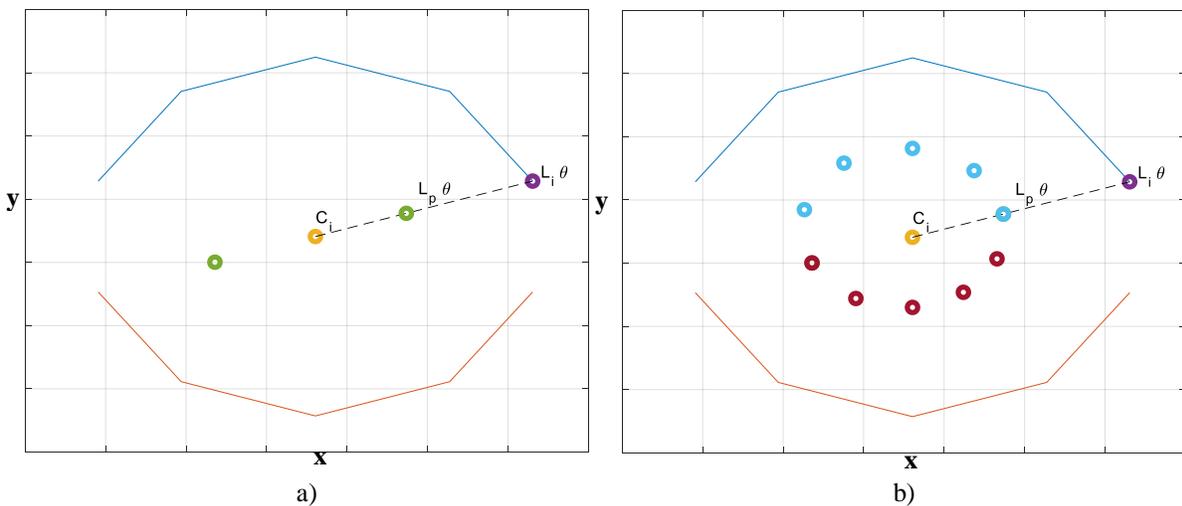


Figura 3.22 a) intersección con las primeras líneas proyectadas, b) intersecciones con todos los puntos de la circunferencia.

De igual manera en la Figura 3.22 b) se muestran todos los puntos encontrados por las intersecciones por toda la circunferencia, generando una representación de la circunferencia de la pupila representada por una cantidad específica de puntos de acuerdo con la cantidad de muestras especificadas para el valor de theta.

Los puntos de los límites de la región del iris y de la región de la pupila se emplean en las ecuaciones (3.3) y (3.4) para calcular los puntos en dirección del radio para cada eje, como se muestra en la Figura 3.23 , lo cual las coordenadas de los puntos generan las coordenadas de los pixeles en la imagen.

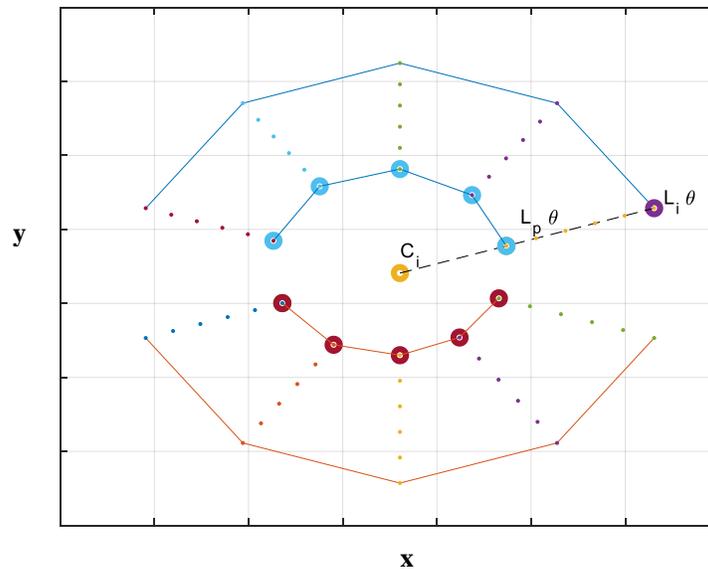


Figura 3.23 Obtención de las muestras en dirección del radio.

Las coordenadas en dirección de theta complementadas con las coordenadas en dirección del radio son requeridas para hacer el remodelado de los pixeles en la imagen. Se inicia con la primera línea encontrada en la circunferencia superior y se continua en sentido de las manecillas del reloj a lo largo de toda la circunferencia como se muestra en la Figura 3.24 a) y se acomodan en sentido de las flechas marcadas en rojo, generando como resultado final una imagen rectangular que contiene información de los puntos de interés calculados anteriormente, como se muestra en la Figura 3.24 b).

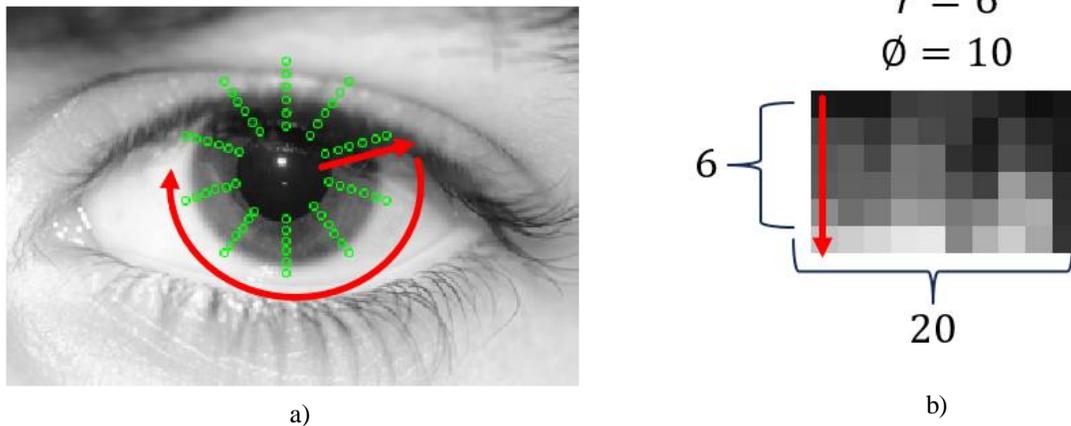


Figura 3.24 a) Sentido de reacomodo de los pixeles, b) resultado del reacomodo de los pixeles.

El tamaño de las imágenes que se obtienen correspondiente a 256 muestras en la dirección del radio y 256 muestras en la dirección de theta, es utilizado a lo largo del proyecto, de manera experimental se fue modifican los parámetros para tener la cantidad de puntos suficiente para cubrir la mayor superficie de interés de la región del iris para obtener una imagen de 256 x 512 como se muestra en la Figura 3.25.

La imagen resultante es la entrada principal a la arquitectura del modelo CNN para la identificación del individuo debido a que la imagen normalizada contiene todos los patrones o características que definen a cada individuo. La extracción de características de la imagen normalizada y de la entrada a la red neuronal se explica en el capítulo 4.



Figura 3.25 Imagen del iris normalizada.

Los problemas de realizar la normalización cuando las dimensiones de los círculos encontrados en la transformada de Hough sobrepasa las dimensiones del tamaño de la imagen generan coordenadas negativas al utilizar la plataforma Matlab, y se debe a que solo se manejan coordenadas positivas en el manejo de matrices.

En la Figura 3.26 a) se muestran las circunferencias encontradas con la transformada de Hough pertenecientes a la región del iris, los círculos negro y rojo sobresalen de la imagen y se generan coordenadas negativas, ya que no pertenecen a los límites de la imagen, lo que genera problemas en el acceso para los pixeles debido a que no pertenecen a la imagen, el mismo problema se presenta en la Figura 3.26 b) al obtener las circunferencias más adecuadas.

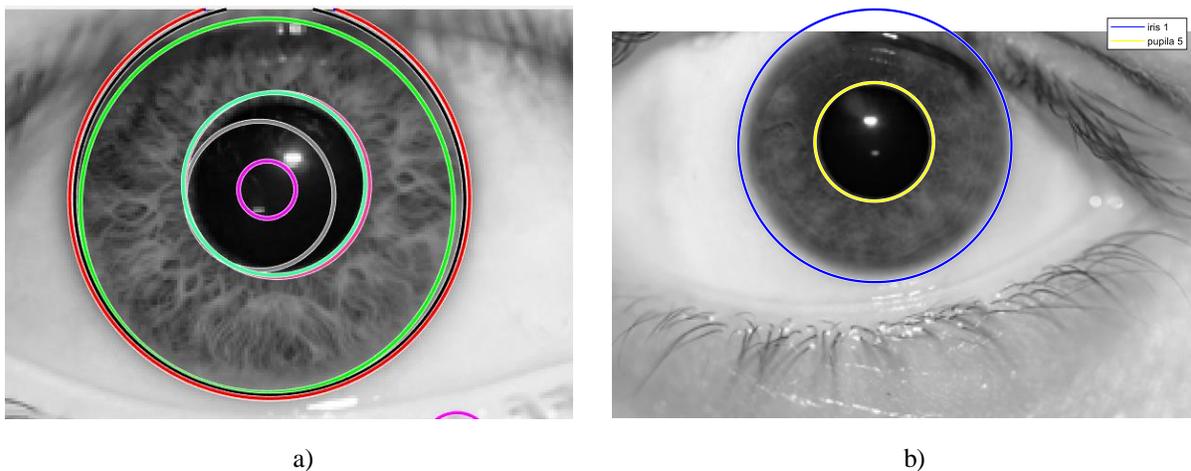


Figura 3.26 a) Problemas con las circunferencias fuera de rango, b) circunferencias fuera de rango sobre las circunferencias seleccionadas.

Para ello, se delimitan las coordenadas negativas en la imagen para el eje vertical dado que el círculo nunca sobrepasa los límites en dimensiones del eje horizontal. Las coordenadas en el eje vertical que contengan un valor menor que 1 se realiza el ajuste del valor para colocarlo en el límite de la imagen.

3.5. Segmentación utilizando redes neuronales convoluciones

Un buen algoritmo de segmentación afecta significativamente la precisión del sistema de reconocimiento de iris en general en cuanto a recolección de datos característicos y puede

manejar los errores generados por las oclusiones de las pestañas, el desenfoque de movimiento, el iris fuera de ángulo, los reflejos especulares, las distancias de separación, los anteojos y la iluminación deficiente [35]. En Investigaciones anteriores se vio que el error generado en la etapa de segmentación del iris se propaga en todas las etapas de reconocimiento posteriores [36].

Para resolver los problemas y disminuir la carga computacional de la segmentación del iris basada en los modelos CNN antes y después del procesamiento, se puede utilizar una CNN que proporciona una plataforma sólida para tareas de segmentación, esto es para poder lograr de una mejor manera los resultados esperados como se trabaja en [37] se propone un modelo CNN llamado IrisDenseNet .

El reconocimiento del iris se ha realizado principalmente con CNN. Ahuja propuso en [38] dos modelos basados en convolución para verificar un par de imágenes perioculares, incluidos los patrones del iris. Al-waisy propuso en [39] un eficiente sistema biométrico multimodal en tiempo real para la detección del iris. Gangwar y otros propusieron en [15] un modelo CNN llamado DeepIrisNet para el reconocimiento de iris en sensores cruzados. Lee y otros propusieron en [40] un método para el iris y el reconocimiento periocular basado en tres CNN.

Para garantizar la confiabilidad y precisión de las CNN, en [37] se propone una combinación de dos métodos básicos: redes convolucionales densamente conectadas con fortalecimiento de la propagación de características (DenseNet) [41] y red de decodificadores de codificador convolucional profundo (SegNet) [42].

En DenseNet, es una red neuronal hacia adelante (Feed-Forward), lo cual fortalece la propagación de características en las capas subsiguientes en un bloque denso, como se muestra en la Figura 3.28.

En SegNet es prácticamente una red totalmente convolucional para la segmentación semántica en píxeles, que utiliza el codificador de la red VGG-16 de 13 capas, eliminando las capas totalmente conectadas.

IrisDenseNet explota el potencial de la red a través de la función reutilizada de la capa anterior, lo que resulta en una mayor eficiencia. La conexión directa es básicamente la

concatenación de características lograda por las capas de concatenación con múltiples entradas y una salida, como se muestra en la Figura 3.27 [37].

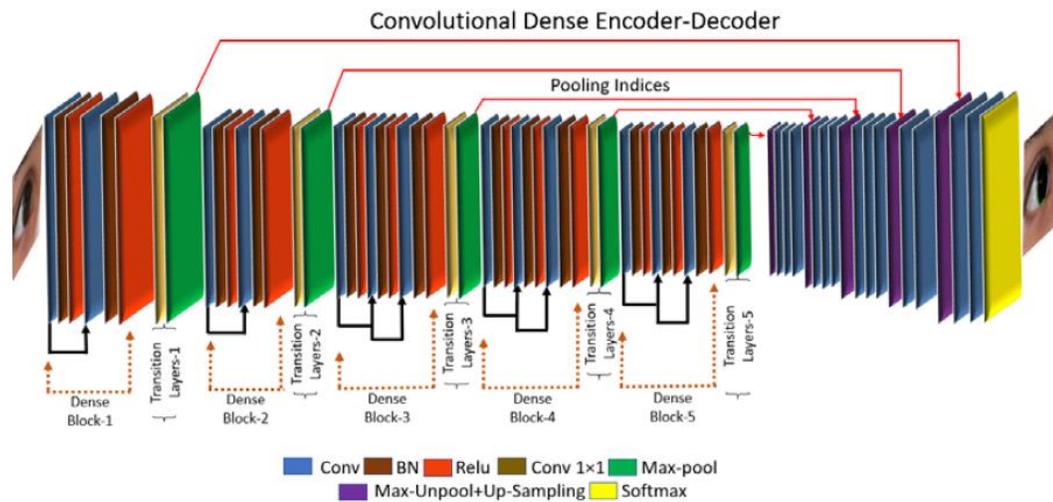


Figura 3.27 Descripción del método propuesto en [37].

En la Figura 3.28 muestra un bloque denso separado e incluye las capas convolucionales básicas (Conv), la normalización por lotes (BN) y la unidad lineal rectificada (ReLU), que incluye las conexiones de las capas a través de la concatenación.

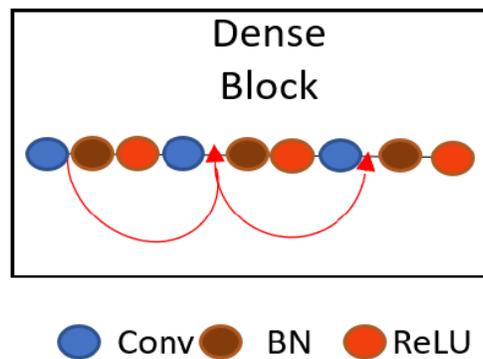


Figura 3.28 Bloque denso por concatenación de características [37].

CAPÍTULO 4

REDES NEURONALES CONVOLUCIONALES

Los modelos CNN son Redes Neuronales Profundas (DNN, por sus siglas en inglés) utilizadas en el procesamiento de imágenes y el reconocimiento de patrones con la capacidad de extraer automáticamente las características distintivas de las imágenes de entrada.

Las CNN son redes neuronales multicapa *Feed-Forward* que difiere de las redes neuronales tradicionales totalmente conectadas en que combina varias capas destinadas a la extracción de características, seguidas de una serie de capas totalmente conectadas dirigidas a la clasificación [39].

Cada módulo consta de una capa convolucional y una capa de agrupamiento. La capa convolucional comparte muchos pesos, y la capa de agrupamiento reduce los datos de la capa siguiente. El entrenamiento de los pesos en la capa convolucional, junto con los esquemas de agrupamiento elegidos apropiadamente, algunas propiedades de invariancia, por ejemplo, invariancia a rotaciones [43].

Las CNN están diseñadas para adaptarse a datos como [14]:

- una dimensión para señales y secuencias.
- dos dimensiones para imágenes o espectrogramas de audio.
- tres dimensiones para video o imágenes volumétricas.

En la última década, se ha demostrado que una CNN es una herramienta eficiente para tareas relacionadas a la clasificación de imágenes al utilizar DL, siendo utilizadas para aplicaciones de visión artificial como detección humana [5], detección de ojo abierto y cerrado [44], entre otras.

La Figura 4.1 ilustra un ejemplo de la arquitectura general para un modelo CNN, y las capas son las siguientes; capas convolucionales y capas de agrupamiento, encargadas de la extracción de características, una o más capas totalmente conectadas y la capa de clasificación.

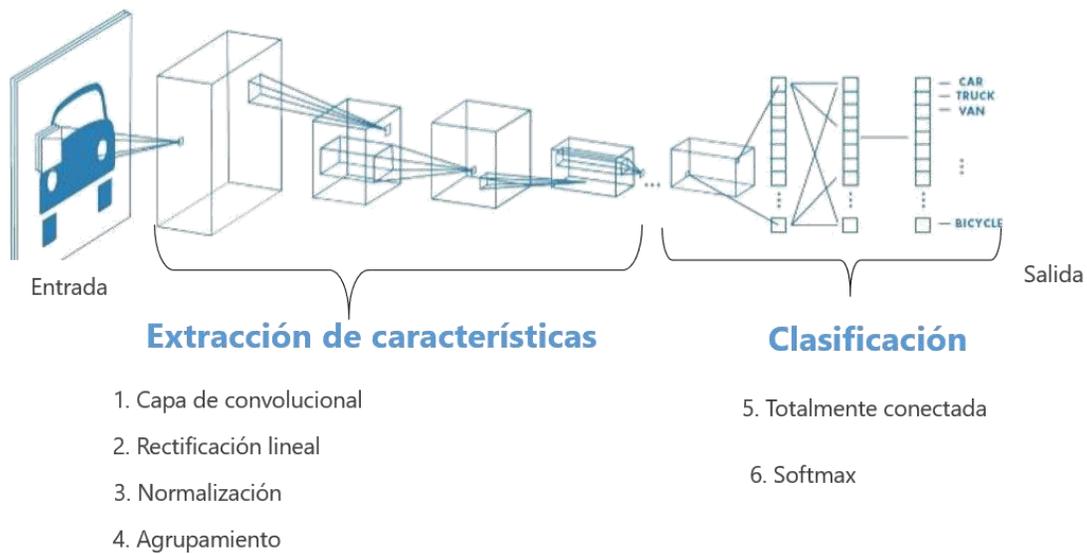


Figura 4.1. Arquitectura tradicional de una CNN con las dos principales etapas.

Además, tienen buenos desempeños en problemas de reconocimiento de patrones, pero son difíciles de entrenar. Algunos parámetros que contribuyen significativamente al entrenamiento, como lo son el tamaño de los filtros utilizados para las operaciones de convolución, el número de mapas de características en cada capa de convolución, el tipo de operación de agrupación implementada en la capa de submuestreo, la compilación de mapas de convolución, la profundidad de la red, y funciones de activación [45].

En los modelos CNN, los parámetros se organizan generalmente en conjuntos de unidades estructurales tridimensionales, conocidas como filtros [46]. El filtro suele ser cuadrado en términos de sus dimensiones espaciales, que suelen ser mucho más pequeñas que las de la capa a la que se aplica.

4.1. Capa convolucional

Esta capa realiza la convolución de los pesos con un conjunto de filtros para obtener un mapa de características y la profundidad es igual al número de filtros aplicados en esa capa.

Definir la cantidad de capas convolucionales dependerá en gran medida del tipo de datos que tenga. Los datos pueden variar según el tamaño, la complejidad de la imagen, el tipo de tarea de procesamiento de imágenes y más.

La operación de convolución coloca el filtro en cada posición posible en la imagen (o capa oculta) para que el filtro se superponga completamente con la imagen y realice un producto punto entre los parámetros de la imagen de entrada y el filtro.

Claramente, un mayor número de mapas de características es el resultado de un mayor número de filtros. Además, aumentar el número de filtros en una capa en particular aumenta el número de mapas de características (es decir, profundidad) de la siguiente capa.

La entrada a la CNN en la capa q de tamaño $L_q \times B_q \times d_q$. Donde, L_q se refiere a la altura (o longitud), B_q se refiere al ancho, y d_q es la profundidad, de igual manera para el tamaño del filtro que generalmente es cuadrado se denota por $F_q \times F_q \times d_q$. F_q se refiere a la altura (o longitud) y al ancho.

Formalmente la operación de convolución desde la capa q hasta la capa $(q + 1)$ se definen de la siguiente manera:

$$h_{ijp}^{(q+1)} = \sum_{r=1}^{F_q} \sum_{s=1}^{F_q} \sum_{k=1}^{d_q} w_{rsk}^{(p,q)} h_{i+r-1, j+s-1, k}^{(q)} + b^{(p,q)} \quad \begin{array}{l} \forall i \in \{1 \dots, L_q - F_q + 1\} \\ \forall j \in \{1 \dots, B_q - F_q + 1\} \\ \forall p \in \{1 \dots d_{q+1}\} \end{array} \quad (4.1)$$

El filtro p en la capa q tiene parámetros denotados por el tensor tridimensional $W^{(p,q)} = [w_{ijk}^{(p,q)}]$. Los índices i, j, k indican las posiciones a lo largo de la altura, el ancho y la profundidad del filtro respectivamente. El bias es representado por $b^{(p,q)}$. Los mapas de características en la capa q están representados por el tensor tridimensional $H^{(q)} = [h_{ijk}^{(q)}]$. Cuando el valor de q es 1, el caso especial correspondiente a la notación $H^{(1)}$ simplemente representa la capa de entrada.

4.2. Capas ReLU (Unidades Lineales Rectificadas)

La siguiente capa después de la convolucional es generalmente la de rectificación lineal (ReLU), es conveniente aplicarla inmediatamente después una capa no lineal (o capa de activación) [47].

Cada capa de convolución debe capturar algún patrón en la imagen y se pasa a la siguiente capa de convolución, por consiguiente, los valores negativos no son importantes en el procesamiento de imágenes y, por lo tanto, se establecen en 0. Los valores positivos después de la convolución deben pasar a la siguiente capa. Es por eso por lo que la capa ReLU se está utilizando como una función de activación. Si utilizamos la función sigmoide o tanh, la información se pierde ya que ambas funciones modificarán las entradas.

Es de destacar que el uso de la función de activación ReLU es una evolución reciente en el diseño de redes neuronales. En los años anteriores, se usaban funciones de activación saturantes como sigmoide y tanh. Sin embargo, se demostró en [48] que el uso de ReLU tiene enormes ventajas sobre estas funciones de activación, tanto en términos de velocidad como de precisión. El aumento de la velocidad también está conectado a la precisión porque permite utilizar modelos más profundos y entrenarlos durante más tiempo.

La capa ReLU está definida por la siguiente ecuación:

$$f(H^{(q)}) = \max(0, H^{(q)}) \quad (4.2)$$

En términos básicos, esta capa solo cambia las activaciones negativas y las pone en 0 aumentando las propiedades no lineales del modelo y la red genera sin afectar los campos de la red convolucional.

4.3. Salto (stride)

El salto es el número de píxeles en que se desplaza el filtro sobre la matriz de entrada definido por S_q . Para un salto de 1 en la capa q , la convolución se realiza en las ubicaciones 1, $S_q + 1$, $2 S_q + 1$, y así sucesivamente a lo largo de las dimensiones espaciales de la capa, como se muestra en la Figura 4.2 a).

Los pasos más grandes pueden ser útiles en configuraciones con limitaciones de memoria o para reducir el sobreajuste si la resolución espacial es innecesariamente alta. El tamaño

espacial para la altura está definida por la ecuación (4.3) y el tamaño espacial del ancho está definido por la ecuación (4.4) como se muestra en la Figura 4.2 b) [46].

$$\frac{L_q - F_q}{S_q} + 1 \quad (4.3)$$

$$\frac{B_q - F_q}{S_q} + 1 \quad (4.4)$$

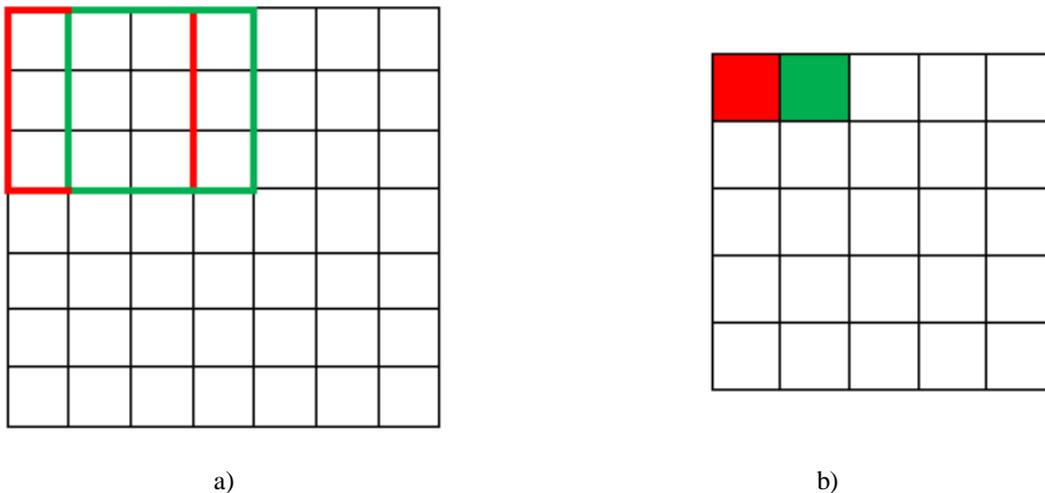


Figura 4.2 Salto del filtro sobre una matriz a) entrada de tamaño 7x7 con un filtro de 3x3 aplicando con paso de una unidad, b) salida obtenida de tamaño 5x5.

4.4. Relleno (*padding*)

La operación de convolución reduce el tamaño de la capa ($q + 1$) en comparación con el tamaño de la capa q . Este tipo de reducción de tamaño no es deseable en general, ya que tiende a perder información a lo largo de los bordes de la imagen (o del mapa de características, en el caso de capas ocultas). Este problema se puede resolver mediante el uso de relleno.

En el relleno agrega $(F_q - 1) / 2$ píxeles alrededor de los bordes del mapa de características como se muestra en la Figura 4.3, estos píxeles son realmente valores de características en el caso de rellenar capas ocultas. El valor de cada uno de estos valores de

funciones rellenas se establece en 0, independientemente de si la entrada o las capas ocultas se están rellendo.

La altura y el ancho espacial de entrada aumenta en $(F_q - 1)$, que es exactamente lo que reducen en la salida después la convolución. Las porciones rellenas no contribuyen al producto de punto final porque sus valores se establecen en 0, es decir, permiten la operación de convolución con una porción del filtro "sobresaliendo" de los bordes de la capa.

Este tipo de relleno se conoce como medio relleno, lo cual está diseñado para mantener exactamente el tamaño espacial, generalmente se realiza en todas las capas, y no solo en la primera capa donde las ubicaciones espaciales corresponden a los valores de entrada.

En la Figura 4.3 se muestra un ejemplo del relleno de un mapa de entidades único, donde se rellenan dos ceros en todos los lados de la imagen (o mapa de funciones) [46].

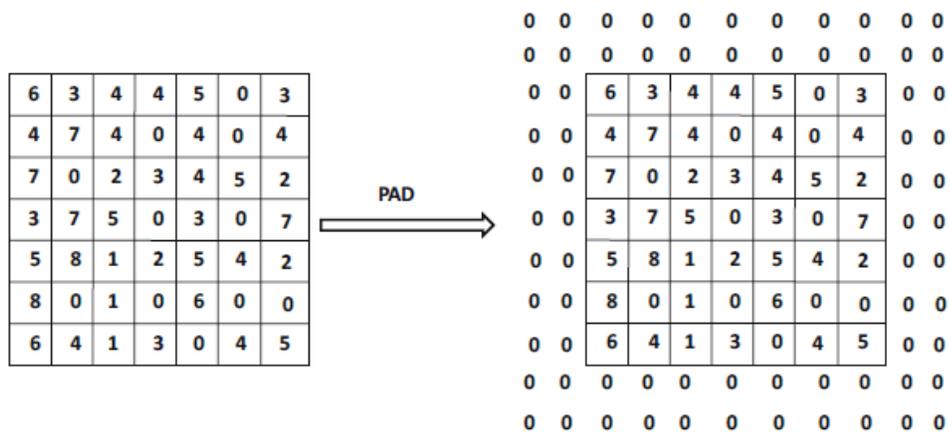


Figura 4.3 Un ejemplo de relleno sobre una imagen o mapa de activación.

4.5. Normalización

La normalización por lotes tiene un efecto benéfico en el flujo de gradiente a través de la red, al reducir la dependencia de los gradientes en la escala de los parámetros o de sus valores iniciales. Esto permite utilizar constantes de aprendizaje mucho más altas sin el riesgo de divergencia en el entrenamiento de los pesos [49].

En [48] se introduce el esquema de la normalización de la respuesta local, que siempre se usa inmediatamente después de la capa ReLU y ayuda a la generalización. La idea básica de este enfoque está inspirada en principios biológicos y tiene la intención de crear competencia entre diferentes filtros.

Considerando una capa que contiene N filtros y los valores de activación en una posición espacial particular (x, y) están dados por $H^{(1)} \dots H^{(N)}$. Luego, cada $H^{(q)}$ se convierte en un valor normalizado b_q usando la siguiente fórmula:

$$b_q = \frac{H^{(q)}}{\left(k + \alpha \sum_{j=q-\lfloor \frac{n}{2} \rfloor}^{q+\lfloor \frac{n}{2} \rfloor} (H^{(j)})^2 \right)^\beta} \quad (4.5)$$

donde n es la cantidad los conjunto filtros adyacentes en la misma capa. Las constantes k , n , α y β son hiper parámetros cuyos valores se determinan utilizando un conjunto de validación, establecidos en [48].

4.6. Capa de agrupamiento (*Pooling*)

La operación de agrupación funciona en regiones de cuadrículas pequeñas de tamaño $P_q \times P_q$ en cada capa, y produce otra capa con la misma profundidad a diferencia de los filtros.

Para el enfoque de agrupamiento máximo, cada región cuadrada de tamaño $P_q \times P_q$ en cada uno de los mapas de activación de d_q , se devuelve el máximo de los valores. Para un salto de 1, se produce una nueva capa de tamaño $(L_q - P_q + 1) \times (B_q - P_q + 1) \times d_q$.

Sin embargo, en la agrupación es más común usar un salto $S_q > 1$, por consecuencia, la longitud de la nueva capa será $(L_q - P_q) / S_q + 1$ y la amplitud será $(B_q - P_q) / S_q + 1$. Por lo tanto, la agrupación reduce drásticamente las dimensiones espaciales de cada mapa de activación.

Otro tipo de agrupación es la agrupación promedio, pero rara vez es utilizada. En [50] se diseñó la primera red convolucional llamada LeNet-5, se utiliza una variante de agrupación promedio y se denominó submuestreo2.

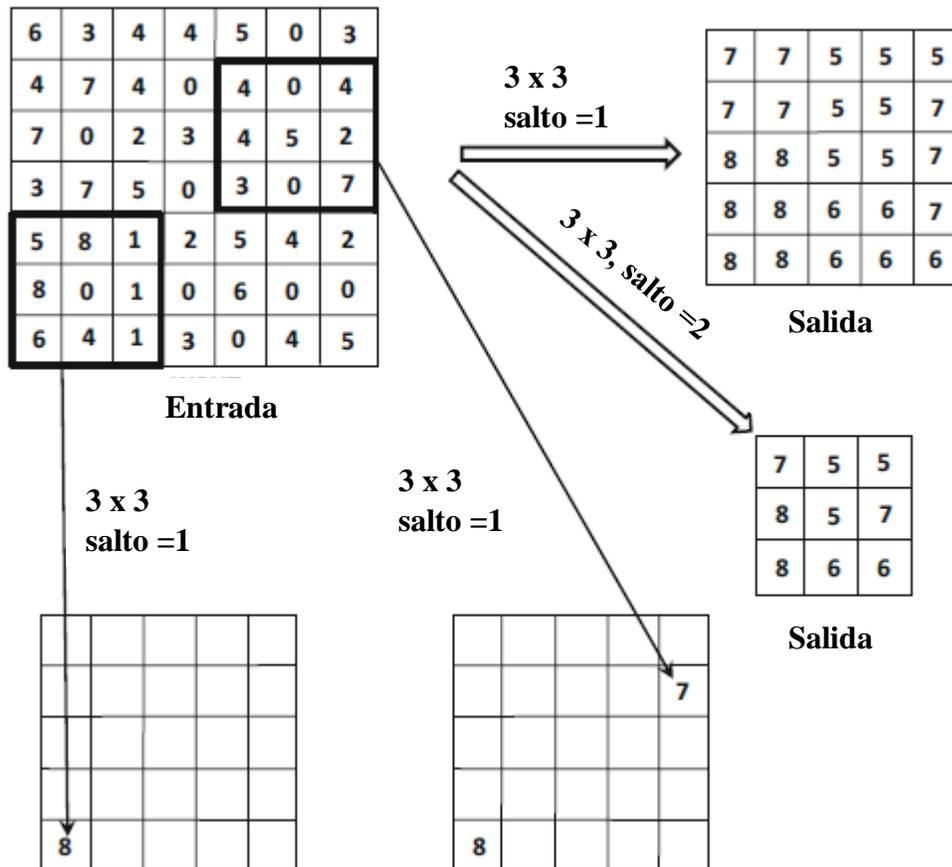


Figura 4.4. Aplicación de agrupamiento máximo sobre una matriz de 7 x 7 con saltos de 1 y 2.

En la Figura 4.4 se muestran ejemplos de agrupación con saltos de 1 y 2. Se utiliza la agrupación en regiones 3×3 . Con un paso de 2, no habría superposición entre las diferentes regiones que se están agrupando, y es bastante común usar este tipo de configuración. Sin embargo, a veces se sugiere tener al menos cierta superposición entre las unidades en las que se realiza la agrupación, porque hace que el enfoque sea menos probable que se sobreajuste.

4.7. Capa totalmente conectada

La salida de la última capa convolucional o de agrupación se alimenta a una o más capas completamente conectadas como en una red neuronal tradicional Feed-forward. En esas capas, las salidas de todas las neuronas en la $(L-1)$ capa están completamente conectadas a cada neurona en la capa L [39] y utiliza las capas anteriores para realizar la tarea de clasificación [5], como se muestra en la Figura 4.5.

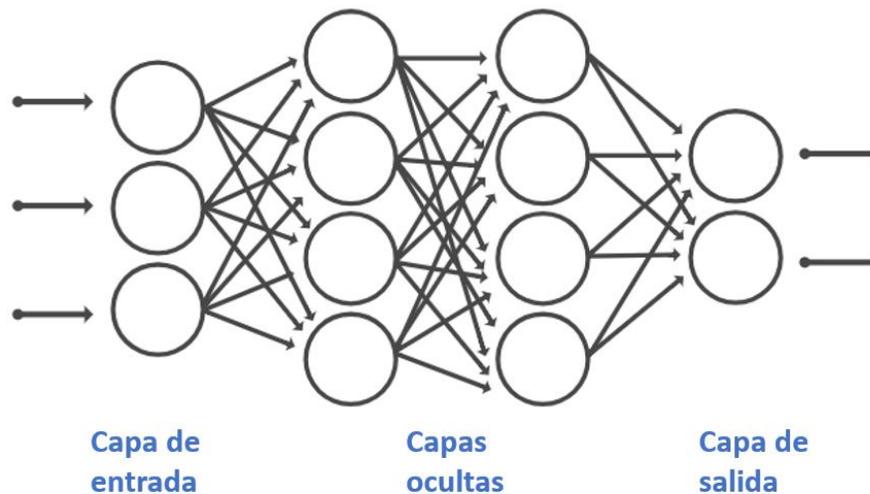


Figura 4.5. Capa totalmente conectada aplicada a una red neuronal.

Básicamente, esta capa toma una entrada (cualquiera que sea la salida de la capa convolucional o ReLU o agrupación que le precede) y genera un vector de clases de tamaño C y analiza qué características de alto nivel se correlacionan más fuertemente con una clase particular y tiene pesos particulares para que cuando calcule los productos entre los pesos y la capa anterior, obtenga las probabilidades correctas para las diferentes clases.

4.8. Dropout

La técnica dropout [51], consiste en poner a cero la salida de cada neurona oculta con probabilidad 0.5. Las neuronas a las que se aplica el dropout de esta manera no contribuyen al entrenamiento. Entonces, cada vez que se presenta una entrada, la red neuronal muestrea una arquitectura diferente, pero todas estas arquitecturas comparten pesos.

Esta técnica reduce la coadaptación complejas de las neuronas, es decir, una neurona no puede confiar en la presencia de otras neuronas particulares. Por lo tanto, aprenden características más robustas junto con muchos subconjuntos aleatorios diferentes de las otras neuronas.

En cierto modo obliga a la red a ser redundante, es decir, la red debería ser capaz de proporcionar la clasificación o salida correcta para un ejemplo en específico, incluso si se elimina algunos mapas de activación, asegurándose que la red no se esté ajustando demasiado a los datos de entrenamiento y, por lo tanto, ayuda a problema de sobreajuste [52]. Una nota importante es que esta capa solo se usa durante el entrenamiento.

4.9. Bias (Sesgo)

Cada filtro único en una capa está asociado con su propio bias. Por lo tanto, el filtro p en la capa q tiene un bias $b^{(p,q)}$. Cuando se realiza cualquier convolución con el filtro p en la capa q , el valor de $b^{(p,q)}$ agrega un producto escalar. Simplemente aumenta el número de parámetros en cada filtro en 1 y, por lo tanto, no es una sobrecarga significativa. Como todos los demás parámetros, se aprende durante la propagación hacia atrás.

Se tratar como un peso de una conexión cuya entrada siempre se establece en +1. Esta entrada especial se utiliza en todas las convoluciones, independientemente de la ubicación espacial de la convolución. Por lo tanto, el número de entidades de entrada en la capa q es $1 + L_q \times B_q \times d_q$.

4.10. Capa clasificación Softmax

La red tiene variaciones en la precisión con las que las neuronas aprenden debido a que los pesos se inicializan de forma aleatoria. Consecuentemente, se utiliza una función de pérdida de entropía cruzada denominada Softmax, se emplea para calcular el error normalizado entre la etiqueta del *ground truth* (salida deseada) y la etiqueta de predicción obtenida [53].

La capa Softmax se encuentra entre la capa de salida de la red neuronal y la capa de predicción de los resultados de la red y se define como [54]:

$$\text{softmax}(L_i^{(s)}) = L_i^{(s)} = \frac{e^{L_i^{(s)}}}{\sum_{j=1}^C e^{L_j^{(s)}}} \quad (4.6)$$

donde $L_i^{(q)}$ es la L -ésima salida de la red neuronal y la suma incluye desde la primera clase hasta la C , esto es, la cantidad de clases en la red neuronal. La salida de esta capa es la probabilidad de que la imagen pertenezca a cada una de las clases para la identificación de la persona.

4.11. Proceso de aprendizaje

Una CNN se caracteriza por una gran cantidad de parámetros, los filtros de las capas convolucionales y los pesos de la capa totalmente conectada que debe aprenderse para resolver el problema específico de interés. Dichos parámetros se inicializan aleatoriamente o con otros métodos más estables.

El proceso aprendizaje emplea el método de backpropagation, contiene dos fases principales, denominadas fases hacia adelante (forward) y hacia atrás (backward), respectivamente. La fase hacia adelante se requiere para calcular los valores de salida y las derivadas locales en varios nodos, y la fase de retroceso se requiere para acumular los productos de estos valores locales en todas las rutas desde la entrada a la salida.

El método para en pequeños lotes de imágenes de entrada, actualizando los pesos con cada lote procesado, con el objetivo de minimizar una función de costo J , que, en tareas de clasificación simples, mide el desajuste entre las salidas deseadas (*ground truth*), y la salida real de la red [55].

Para minimizar la función de costo, hay muchos solucionadores alternativos. El más extendido y utilizado es el descenso de gradiente estocástico con momento (SGDM, por sus siglas en inglés) y se incluye la constante de aprendizaje que permite controlar la magnitud del cambio en el ajuste de los pesos.

Además, la constante de momento es una estrategia para hacer que la CNN converja más rápidamente al error mínimo como se ha explicado con anterioridad al evitar los cambios

bruscos de la modificación de los pesos sinápticos al mantenerlos en una misma tendencia general que lleva la red neuronal [55].

4.11.1. Backpropagation a través de la convolución.

Para el entrenamiento de la CNN, se emplea el algoritmo de propagación hacia atrás (backpropagation). Para cada imagen de entrenamiento hay un valor de salida deseado $O_i^{(s)}(g)$, donde $g = 1, \dots, C$ describe el número de la salida. La imagen se presenta de manera aleatoria y la salida de la CNN resultante $L_i^{(s)}[t]$ se determina con base en la ecuación (4.6). El error resultante para una imagen de entrenamiento dada s en la iteración t , está dada por [56]:

$$E^s[t] = \frac{1}{2} \sum_{g=1}^C (O_i^{(s)}(g) - L_i^{(s)}(g)[t])^2 \quad (4.7)$$

Mientras que el error total de en la iteración t está dada por:

$$E[t] = \sum_{s=1}^s E^s[t] \quad (4.8)$$

donde s representa el número total de imágenes de entrenamiento, η representa el factor de aprendizaje, γ representa la constante de momento y los pesos de los filtros $w_{ijk}^{(q)}$ se derivan de la regla general de la cadena:

$$w_{ijk}^{(q)}(i, j)[t + 1] = w_{ijk}^{(q)}(i, j)[t] - \eta \Delta w_{ijk}^{(q)}(i, j)[t] + \gamma \Delta A[t] \quad (4.9)$$

donde

$$\Delta w_{ijk}^{(q)}(i, j)[t] = \frac{\partial E[t]}{\partial w_{ijk}^{(q)}(i, j)[t]} \quad (4.10)$$

$$\Delta A[t] = (w_{ijk}^{(q)}(i, j)[t] - w_{ijk}^{(q)}(i, j)[t - 1]) \quad (4.11)$$

4.12. Arquitectura AlexNet

La arquitectura de AlexNet se muestra en la Figura 4.6 a). Se compone de 8 capas, 5 capas convolucionales y 3 capas totalmente conectadas. La primera capa convolucional contiene una imagen de entrada de tamaño de $224 \times 224 \times 3$ aplicando 96 filtros convolucionales de tamaño de $11 \times 11 \times 3$ con un salto de 4 píxeles.

La segunda capa convolucional toma como entrada la salida de la primera capa convolucional y aplica 256 filtros de tamaño de $5 \times 5 \times 48$.

La tercera, cuarta y quinta capa convolucional están conectadas entre sí sin ninguna intervención. La tercera contiene 384 filtros con tamaño de $3 \times 3 \times 256$. La cuarta capa contiene 384 kernels de tamaño de $3 \times 3 \times 192$, y la quinta capa convolucional tiene 256 filtros de tamaño de $3 \times 3 \times 192$.

Los filtros de la segunda, cuarta y quinta capa convolucional están conectados solamente a los mapas de activación en la capa anterior que residen en la misma GPU. Los mapas de activación de la tercera capa convolucional están conectados a todos los mapas en la segunda capa.

Las neuronas en las capas totalmente conectadas están conectadas a todas las neuronas en la capa anterior. Las capas de normalización siguen a la primera y segunda capas convolucionales. Las capas de agrupación máxima siguen tanto las capas de normalización como la quinta capa convolucional. La capa de no linealidad ReLU se aplica a la salida de cada capa convolucional y totalmente conectada.

La limitación principal de la red está dada por la cantidad de memoria disponible en las GPU actuales y por la cantidad de tiempo de entrenamiento que se está dispuesto a tolerar, debido a que la red de la arquitectura AlexNet tuvo una duración aproximada de entre 5 y 6 días para el entrenamiento utilizando una GPU GTX 580 de 3GB.

Por lo tanto, hace imposible ajustar los cálculos intermedios en esta cantidad de espacio. La red se particionó en dos GPU que trabajan juntas para construir un modelo de entrenamiento con mayor velocidad y uso compartido de memoria, como se muestra en la Figura 4.6 b).

Es de destacar que las GPU están interconectadas solo en un subconjunto de las capas, lo que conduce a algunas diferencias entre la Figura 4.6 a) y b), específicamente, tiene menos peso porque no todas las capas tienen interconexiones, por consiguiente, reduce el tiempo de comunicación entre los procesadores y ayuda a la eficiencia [48].

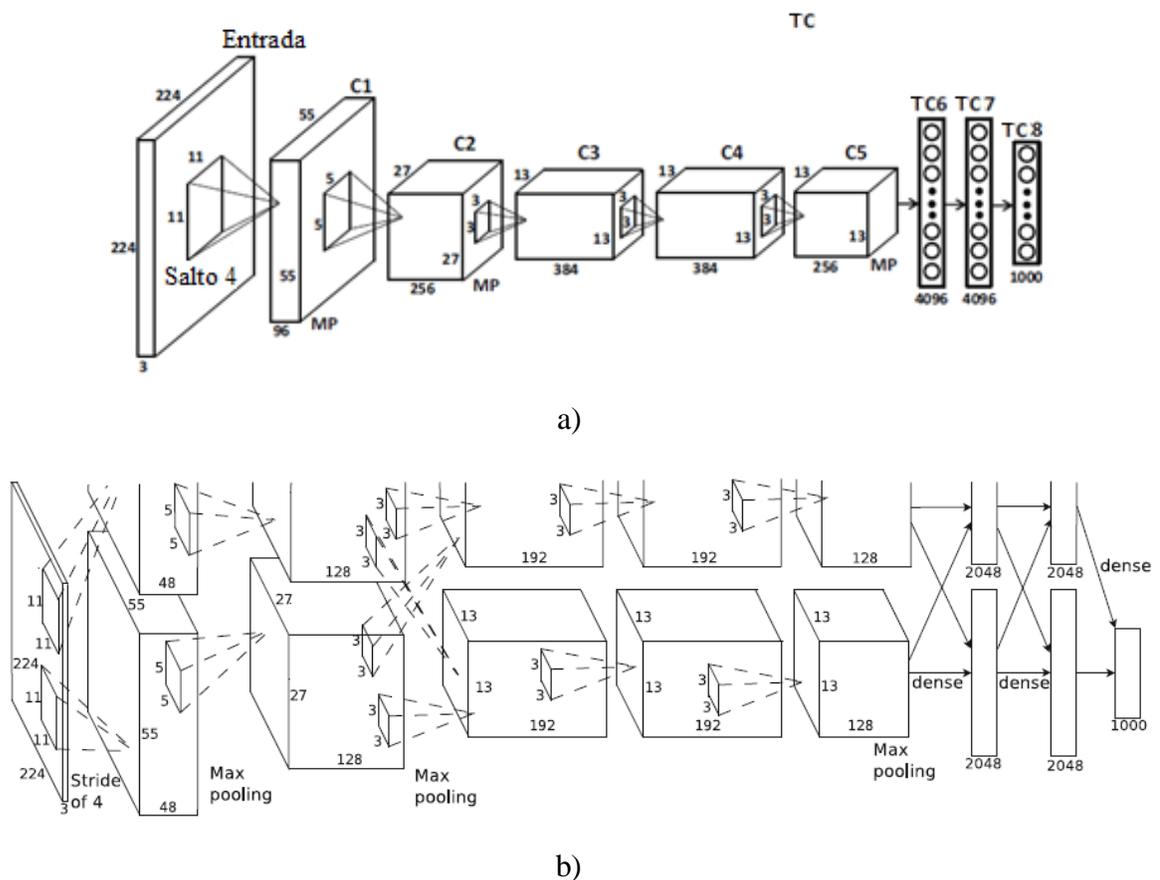


Figura 4.6 Arquitectura AlexNet, a) sin particionamiento de GPU, b) con particionamiento de GPU [48].

4.12.1. Entrenamiento

La red fue entrenada usando la ecuación (4.9), con un tamaño de lote de 128, utilizando un momento de 0.9 en la actualización de los pesos y un decaimiento de los pesos de 0.0005. Los

pesos se inicializaron en cada capa con una distribución Gaussiana de media cero con una desviación estándar de 0.01.

Se iniciaron los *bias* de las neuronas en la segunda, cuarta y quinta capas convolucionales, así como en las capas ocultas totalmente conectadas, con la constante 1, lo cual acelera las etapas iniciales del aprendizaje al proporcionar a las ReLU entradas positivas, los *bias* de las neuronas en las capas restantes se inicializan con la constante 0.

4.12.2. Puntos principales

- La red fue entrenada con el conjunto de imágenes ImageNet, contiene alrededor de 15 millones de imágenes de un total de más de 22 mil categorías.
- Se utilizó una capa ReLU para la función de la no linealidad (se encontró que disminuye el tiempo de entrenamiento ya que las ReLU son más rápidas que una función tanh convencional.
- Se utilizaron técnicas de aumento de datos que consistieron en traslaciones, reflexiones horizontales.
- Se implementaron capas de *dropout* para combatir el problema del sobreajuste a los datos de entrenamiento.
- Se entrenó el modelo utilizando el método del gradiente descendente por lotes, con valores específicos para el momento y la disminución de los pesos.
- Se entrenó con dos GPU GTX 580 durante cinco a seis días.

4.13. Arquitectura Very Deep Convolutional Network (VGG)

Durante el entrenamiento, la entrada de la red ConvNets es ajustada a 224×224 en formato RGB. El único procesamiento que se realiza es restar el valor RGB promedio, calculado en el conjunto de entrenamiento de cada pixel. La imagen pasa a través de una serie de capas convolucionales.

Se usan filtros pequeños de tamaño de 3×3 . En una de las configuraciones se utilizan filtros de convolución de 1×1 , que se pueden ver como una transformación lineal de los canales de entrada (seguida de la capa ReLU). El paso de desplazamiento del filtro de convolución se fija a 1 pixel; el relleno espacial de la capa convolucional entrada es tal que la resolución espacial se conserva después de la convolución, es decir, el relleno es de 1 pixel para las capas convolucionales de 3×3 .

La agrupación se lleva a cabo mediante cinco capas de agrupación máxima, que siguen algunas de las capas convolucionales (no todas las capas de convolución son seguidas por una agrupación máxima). La agrupación máxima se realiza en una ventana de 2×2 pixeles, con un salto de 2.

Una serie de capas convolucionales (que tiene una profundidad diferente en cada una de las arquitecturas) le siguen tres capas totalmente conectadas: las dos primeras tienen 4096 canales cada una, la tercera realiza una clasificación ILSVRC de 1000 canales y, por lo tanto, contiene 1000 clases. La capa final es la capa softmax.

4.13.1. Configuraciones

Las configuraciones de ConvNets, evaluadas se resumen en la Tabla 4.1, una por cada columna, refiriéndose a las redes por sus nombres (A – E). Todas las configuraciones siguen el diseño, y difieren solo en la profundidad: de 11 capas de peso en la red A (8 capas de convolución y 3 capas totalmente conectadas) a 19 capas de peso en la red E (16 capas de convolución y 3 capas de totalmente conectadas).

El ancho de convolución capas (el número de canales) es bastante pequeño, comenzando desde 64 en la primera capa y luego aumentando en un factor de 2 después de cada capa de agrupación máxima, hasta llegar a 512 [57].

En la Tabla 4.2 se muestra el número de parámetros para cada configuración. A pesar de una gran profundidad, el número de pesos las redes no son mayor que el número de pesos en una red más profunda con mayor número de capas convolucionales y campos receptivos como en [58] que cuenta con 144 M de pesos.

Tabla 4.1 Configuración ConvNets.

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224×224 RGB image)					
conv3-64	conv3-64 LRN	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 conv1-256	conv3-256 conv3-256 conv3-256	conv3-256 conv3-256 conv3-256 conv3-256
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Tabla 4.2 Número de parámetros (en millones).

Network	A, A-LRN	B	C	D	E
Número de parámetros	133	133	134	138	144

4.13.2. Entrenamiento

La inicialización de los pesos en la red es importante, donde una mala inicialización puede detener el aprendizaje debido a la inestabilidad del gradiente en las redes profundas. Para el entrenamiento de la red A (Tabla 4.1) los pesos fueron inicializados de manera aleatoria.

Luego, al entrenar arquitecturas más profundas, los pesos de las primeras cuatro capas fueron inicializados al igual que las últimas tres capas totalmente conectadas con las capas de la red A, las capas intermedias se inicializaron de manera aleatoria. No se disminuyó la tasa de aprendizaje de las capas pre inicializadas lo que les permite cambiar durante el entrenamiento.

4.13.3. Puntos principales

- El uso de solo filtros de tamaño de 3×3 es bastante diferente a los filtros de 11×11 utilizados en AlexNet en la primera capa y los filtros de 7×7 de ZFNet. El razonamiento de los autores es que la combinación de dos capas convoluciones de 3×3 tiene un campo efectivo de 5×5 . Esto a su vez simula un filtro más grande mientras mantiene los beneficios del tamaño de un filtro más pequeño, logrando una disminución en el número de parámetros.
- Tres capas convolucionales espalda con espalda tienen un campo efectivo de 7×7 .
- El número de filtros se duplica después de cada capa de Max pool. Esto refuerza la idea de reducir las dimensiones espaciales, pero la profundidad cada vez mayor.
- Funcionó bien en las tareas de clasificación y localización de imágenes.
- Se utilizó la fluctuación de escala como una técnica de aumento de datos durante el entrenamiento.
- Se utilizaron capas ReLU después de cada capa convolucional y se entrenaron con el descenso de gradiente por lotes.
- Se entrenó en cuatro GPU NVIDIA Titan Black durante dos o tres semanas.

4.14. Arquitectura IrisNet

La arquitectura propuesta IrisNet para el reconocimiento de iris, es una versión adaptación de la arquitectura AlexNet [48], se reduce la cantidad de parámetros al minimizar la

complejidad computacional debido a la reducción de una de las capas convolucionales del centro en la arquitectura establecida.

La arquitectura específicamente está compuesta por los siguientes elementos: una capa de entrada con un tamaño de la imagen de entrenamiento de 256×250 ; 4 capas convolucionales encargadas de la extracción de características; 3 capas totalmente conectadas para la clasificación.

La primera capa convolucional está compuesta por 24 filtros, teniendo como resultado una imagen de $m \times n$ pixeles con una profundidad igual al número de filtros aplicados, el tamaño del filtro es de 9×9 , se obtiene las características más superficiales en la imagen, se realiza el proceso de normalización para cada capa obtenida, seguida de una rectificación lineal, después se aplica un agrupamiento máximo a las convoluciones resultantes con un espacio de 3×3 y un salto de 2.

La segunda capa convolucional está compuesta por 64 filtros 5×5 , seguida por la normalización. Se agrega un relleno de ceros de 2×2 alrededor de la imagen, lo que significa que se agregan ceros a los mapas de activación, esto es para no modificar el tamaño resultante como se muestra en la Figura 4.7, al final es aplicada la capa de agrupamiento máximo con un de tamaño de 3×3 y un salto de 2.

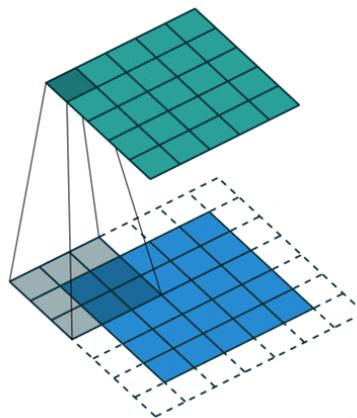


Figura 4.7. Convolución aplicada a una imagen con relleno de ceros mediante un filtro de 3×3 .

Tabla 4.3 Arquitectura propuesta para el reconocimiento de iris.

Nombre	Tipo	Activación
imageinput 256x250x1	Image input	256x250x1
conv_1 24 11x11x1 stride [4 4] padding [0 0 0 0]	Convolution	62x60x24
relu_1	ReLU	63x60x24
batchnorm_1 Batch Normalization whit 24 channels	Batch normalization	62x60x24
maxpool_1 3x3 maxpooling stride [2 2] padding [0 0 0 0]	Max Pooling	30x29x24
conv_2 64 5x5x24 stride [1 1] padding [2 2 2 2]	Convolution	30x29x64
relu_2	ReLU	30x29x64
batchnorm_2 Batch Normalization whit 64 channels	Batch normalization	30x29x64
conv_3 96 3x3x64 stride [1 1] padding [1 1 1 1]	Convolution	30x29x96
relu_3	ReLU	30x29x96
batchnorm_3 Batch Normalization whit 96 channels	Batch normalization	30x29x96
maxpool_2 3x3 maxpooling stride [2 2] padding [0 0 0 0]	Max Pooling	14x14x96
conv_4 64 3x3x96 stride [1 1] padding [1 1 1 1]	Convolution	14x14x64
relu_4	ReLU	14x14x64
batchnorm_4 Batch Normalization whit 64 channels	Batch normalization	14x14x64
maxpool_3 3x3 maxpooling stride [2 2] padding [0 0 0 0]	Max Pooling	6x6x64
fc_1 1024 fully connected layer	Fully Connected	1x1x1024
relu_5	ReLU	1x1x1024
dropout_1 50% dropout	Dropout	1x1x1024
fc_2 1024 fully connected layer	Fully Connected	1x1x1024
relu_6	ReLU	1x1x1024
dropout_2 50% dropout	Dropout	1x1x1024
fc_3 15 fully connected layer	Fully Connected	1x1x15
relu_7	ReLU	1x1x15
softmax	Softmax	1x1x15
classoutput crossentropyx with '002' and 14 other classes	Classification Output	-

La tercera capa convolucional se compone de 96 filtros de 3×3 aplicando el relleno de ceros a la imagen de 1×1 alrededor de la imagen (Figura 4.7), seguida de la capa de

normalización y la capa ReLU, al final se aplica la capa de agrupamiento máximo de tamaño de 3×3 con un salto de 2.

En la última capa convolucional de la CNN se compone de 64 filtros con un tamaño de 3×3 con relleno de ceros de 1×1 , seguida de la capa de normalización, capa ReLU y por último la capa de agrupamiento máximo de tamaño de 3×3 con un salto de 2.

En la Tabla 4.3 contiene detalladamente cada componente de la red neuronal indicando el nombre de cada capa el tipo de capa y el tamaño de cada resultado al aplicar cada operación en cada capa.

Los filtros establecidos en la arquitectura de la red neuronal propuesta son proporcionalmente menores a los filtros establecidos originalmente por la arquitectura AlexNet, para mantener por funcionamiento original.

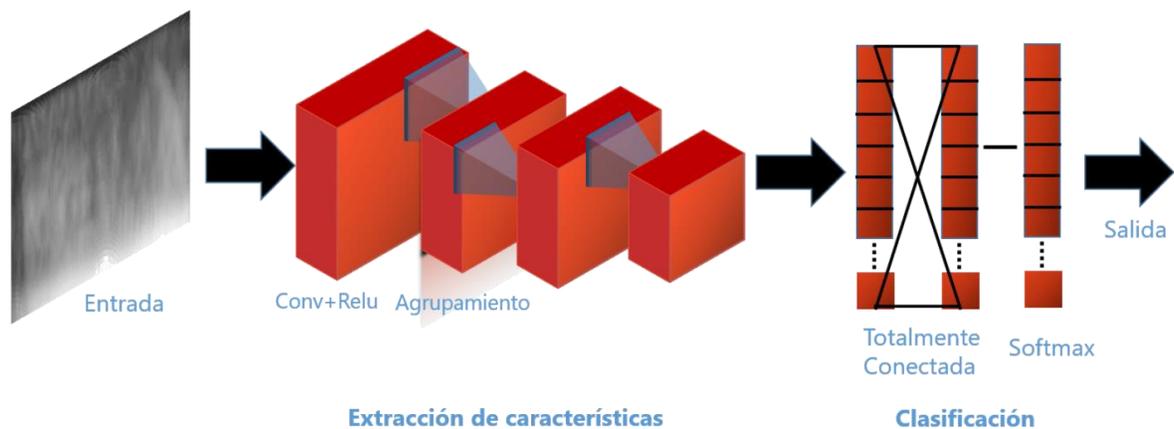


Figura 4.8 Arquitectura de la red neuronal convolucional implementada para el reconocimiento de iris.

En la Figura 4.8 se muestra una representación visual de la arquitectura IrisNet implementada para el reconocimiento de iris al introducir como imagen de entrada, una sección de la imagen del iris para asegurar de que la imagen a entrenar no contenga ruido por oclusión de párpados, sección coloreada de rojo mostrada en la Figura 4.9.

Tener una mala calidad de datos de entrenamiento da como resultado un mal entrenamiento afectando en los resultados finales para la clasificación, las pruebas realizadas de entrenar la red con la imagen directamente de la normalización, se generan resultados por debajo del 80% de desempeño.

Al eliminar parte de los párpados considerada como ruido, debido a que es parte de la piel de cada individuo que no es considerada como información que contenga características relevantes. Como resultado final se tiene una imagen en dimensiones menor, de 256×250 pixeles, que favorece en la rapidez del entrenamiento ya que contiene menos entradas a la red.

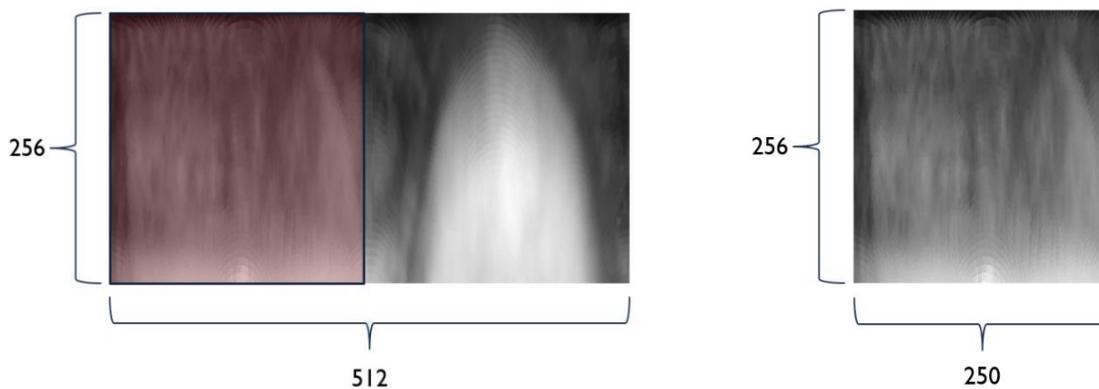


Figura 4.9 Sección tomada para la eliminación de ruido ocasionada por oclusión de párpados.

4.15. Partición de los conjuntos de entrenamiento, validación y pruebas

La base de datos utilizada es UTIRIS [26] creada por la Universidad de Teherán, es el primer banco de datos biométricos registrados en dos sesiones distintas de imágenes de longitud visible (VW) e infrarrojo cercano (NIR) del 24 al 27 de junio del 2007. La base de datos está construida con 1,540 imágenes de 79 individuos de ambos ojos, derecho e izquierdo. Los individuos se enumeran igual en sesiones NIR y VW, la base de datos está protegida por derechos de autor del centro de Excelencia de Control y Procesamiento Inteligente (CIPE, por sus siglas en inglés).

Se seleccionaron 15 clases diferentes (número de personas a identificar) dentro de los 79 individuos, cada conjunto de datos por individuo está compuesto entre 8 y 10 imágenes

incluyendo el ojo izquierdo como el derecho, especificando las características de la base de datos mostrada en la Tabla 4.4.

Tabla 4.4 Configuración para la adquisición de la base de datos.

	Onda visible	Infrarrojo cercano
Dimensión de la imagen (número de píxeles)	2048 x 1360	1000 x 776
Formato de compresión	JPEG	BMP
Dispositivo de cámara	Canon EOS 10D	ISG Lightwise LW
Espacio de color	RGB	GRAY
Longitud focal	100	
Ojos rojos	si	
Tiempo de exposición	1/60	

El conjunto de imágenes de las 15 clases se ha dividido en 3 conjuntos para entrenamiento, validación y pruebas. Primero del total de las imágenes se seleccionan 5 imágenes de cada clase de manera aleatoria por cada entrenamiento de la red para formar el conjunto de entrenamiento, dejando el resto de las imágenes para ser dividida en el conjunto de validación y prueba.

Para la selección del conjunto de validación se seleccionan 2 imágenes del conjunto restante de manera aleatoria por cada clase en cada entrenamiento que se realiza, las imágenes restantes se toman para el conjunto de pruebas, como su nombre lo indica es el conjunto con el que se probará la red neuronal y se obtendrán los desempeños.

Tabla 4.5 Conjunto de imágenes establecidos para el reconocimiento de iris.

Base de Datos	Número de clases	Cantidad de imágenes	Imágenes entrenamiento	Imágenes validación	Imágenes pruebas
UTIRIS	15	138	75	30	33

4.16. Descripción general para la experimentación de la red neuronal

La imagen adaptada mostrada en la Figura 4.9, se coloca en el modelo CNN para realizar el proceso de entrenamiento. El conjunto de validación guía a la red neuronal a la modificación de los pesos.

Al final del entrenamiento, el conjunto de pruebas que se crea en cada entrenamiento con la cantidad indicada en la Tabla 4.5 de la base de datos, se utiliza para determinar el desempeño de la CNN.

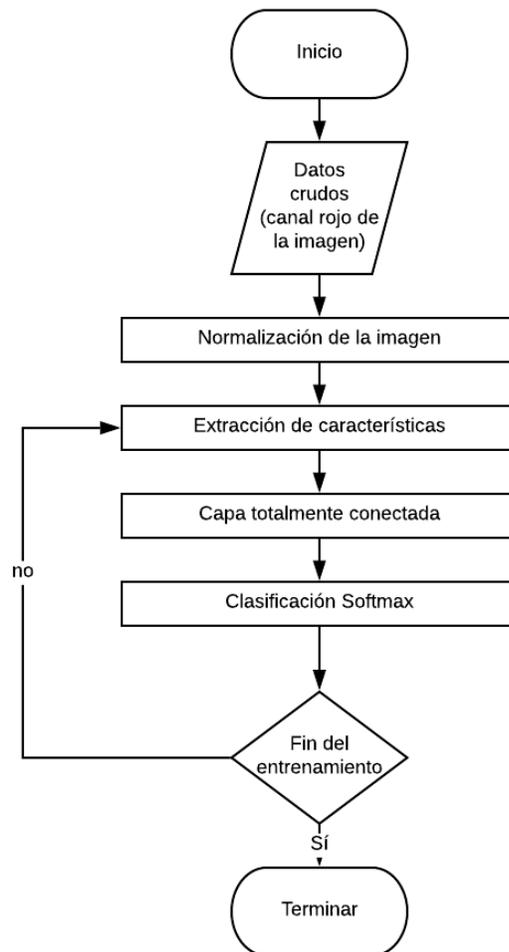


Figura 4.10. Diagrama de flujo de entrenamiento de la red neuronal.

El proceso general se muestra en la Figura 4.10, las imágenes de entrada se establecen en formato de onda visible (RGB) y se selecciona el canal rojo, ya que tiene la onda del espectro visible más cercana a la del infrarrojo cercano, se realiza el proceso de normalización, en la que se lleva a cabo una transformación de la imagen en coordenadas rectangulares a un cambio en coordenadas polares realizando un acomodo espacial de la imagen.

La imagen normalizada y adaptada es la entrada directa a la red neuronal, siendo la etapa convolucional donde se extraen las características de la imagen de acuerdo con los filtros aplicados, una vez realizada la etapa convolucional, la capa totalmente conectada se encarga de combinar todas las características extraídas para hacer más fácil su clasificación.

Por último, la capa Softmax se encarga de dar la clasificación final de la imagen, en donde se calcula el error, si el error calculado no es el deseado se continua con el entrenamiento en la extracción de características, en caso contrario que se alcance un error deseado se detendrá el entrenamiento.

4.16.1. Entrenamiento de la red neuronal

En el entrenamiento de la red neuronal se establecen una serie de parámetros para lograr el aprendizaje de la red de una manera más eficiente. Un parámetro en el entrenamiento es la tasa de aprendizaje e indica el incremento en el que se actualizan los pesos para encontrar el error mínimo deseado, también indica si se requiere usar el ambiente de desarrollo de la tarjeta gráfica específica para acelerar el proceso de entrenamiento.

Otro parámetro es el de momento, hace que la modificación de los pesos se mantenga en una misma dirección y evitar los cambios bruscos en los incrementos, evitando que la red neuronal no converja en un error mínimo, la manera como toma los datos de entrada (imágenes de entrenamiento) a la red, entre otros aspectos, se muestran con más detalle en la Tabla 4.6.

Tabla 4.6 Parámetros de entrenamiento aplicados al entrenamiento de la red neuronal convolucional.

Ambiente de ejecución	GPU GTX 1050
Constante de aprendizaje	0.001
Época máxima	500
Momento	0.9
Tamaño de mini lote	10
Aleatorio	por época
Frecuencia de validación	30

El entrenamiento de la red se realiza repetidamente para generar un promedio general con los desempeños obtenidos individualmente en cada uno de los entrenamientos, se utiliza el conjunto de imágenes de prueba para validar el desempeño del modelo, esto es debido a la toma aleatoria de los datos de entrenamiento en la red se requiere para tomar la mayoría de las combinaciones posibles en cada conjunto.

Al finalizar el entrenamiento de la red neuronal se obtiene una visualización de los filtros adaptados de la primera capa convolucional para ver qué tipo de características se obtienen y realizar un análisis con respecto al funcionamiento de la red y realizar ajustes en caso de ser necesario, también se observa la matriz de confusión de los resultados con el conjunto de imágenes de prueba y ver como categoriza la red en su clasificación.

El entrenamiento de la red neuronal convolucional se realiza en el sistema Matlab en su versión 2018b en una laptop con procesador i5 7^a generación y una GPU GTX 1050 utilizando el *toolbox* de redes profundas [59].

CAPÍTULO 5

RESULTADOS Y CONCLUSIONES

Los filtros generados en la primera capa de la CNN se muestran en la Figura 5.1, representan los 24 filtros aplicados en la extracción de características más superficiales. Sin embargo, los mapas de activación no tienen una representación para obtener una característica en específico, siendo esta una desventaja para la CNN al no saber realmente lo que se genera.

La distribución de cada filtro se debe a las características aleatoria de las características del iris, debido a que cada iris de cada individuo inclusive en ambos ojos del mismo individuo posee características únicas que los definen, por lo que la red se trata de adaptar a dichas características para su clasificación, generando los filtros para obtener características más abstractas.

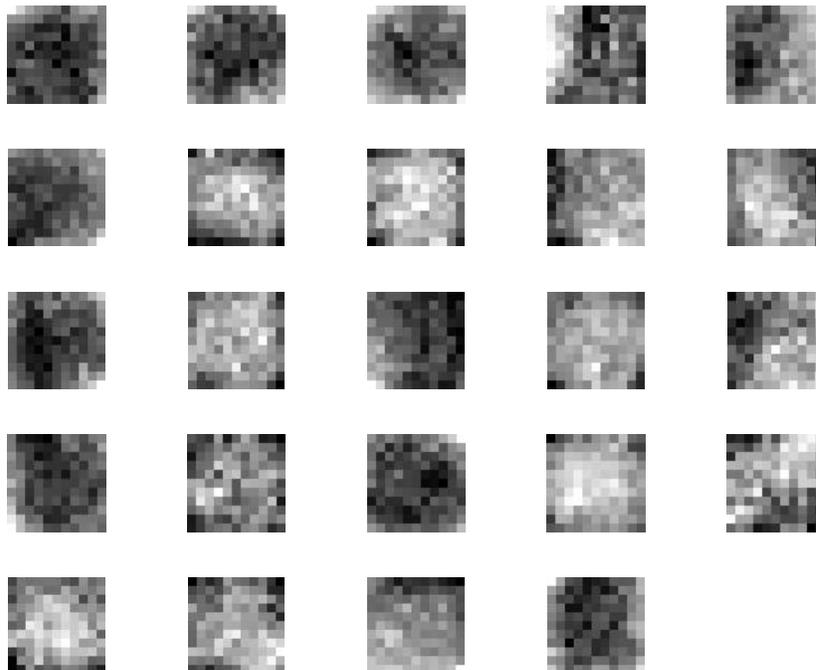


Figura 5.1. Los 24 filtros de tamaño 11×11 resultantes de la primera capa convolucional aplicados a la imagen original para la extracción de características.

De igual manera, en la Figura 5.2 se muestran los 64 filtros obtenidos después del entrenamiento de la última capa convolucional, mostrando filtros abstractos complejos para ser definido, sin embargo, estos filtros al pertenecer a la última capan se encargan de obtener características de alto nivel de complejidad.

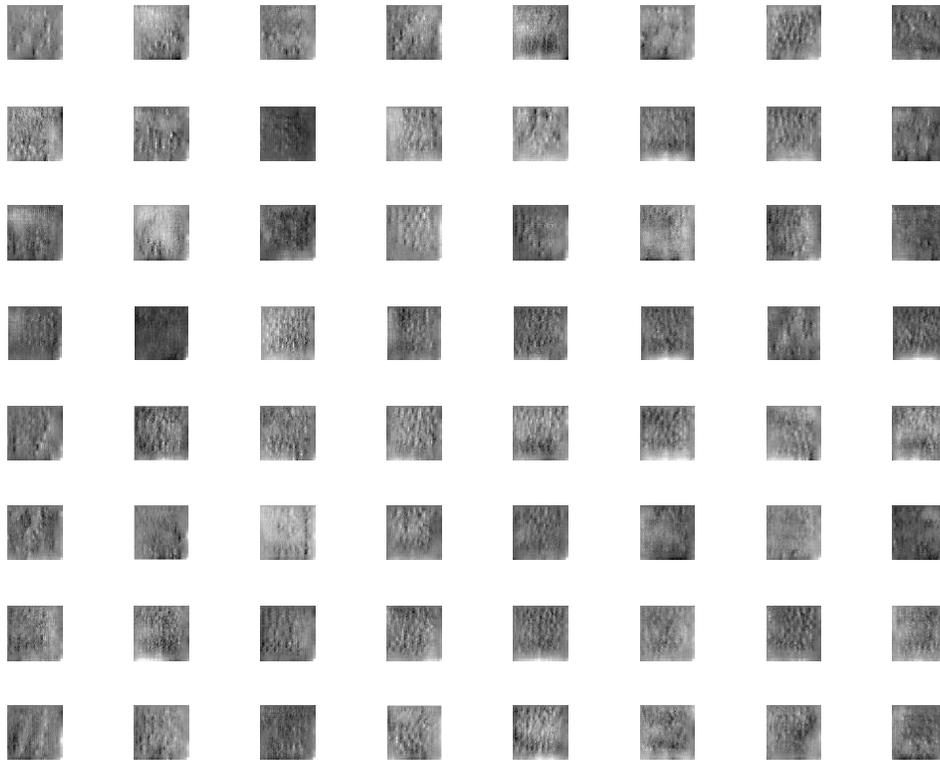


Figura 5.2 Los 64 kerneles resultantes del entrenamiento de la red en la última capa convolucional.

En el entrenamiento de las CNN, se introducen las mismas clases a identificar , como se muestra en la Tabla 5.1, en donde en cada arquitectura se ajustó el tamaño de las imágenes para adaptarse a la entrada de la arquitectura seleccionada, tomando en cuenta que para cada entrenamiento los 3 conjuntos de imágenes se toman las imágenes de manera aleatoria dentro de cada clase.

En la Tabla 5.1 se muestran las clases a identificar con las respectivas etiquetas requeridas para su identificación, se realiza un aumento de datos en las clases requeridas, se copia de manera aleatoria 1 o 2 imágenes para complementar las imágenes. Sin embargo, no es

necesario tener una gran cantidad de imágenes del iris, debido a que, las imágenes contienen idealmente las mismas características para cada persona, y si se tienen una gran cantidad de las imágenes no es lo óptimo para aplicaciones en tiempo real al tener que recolectar una gran cantidad de imágenes de cada individuo.

Tabla 5.1 Clases a identificar con la cantidad de imágenes que se cuentan por cada clase.

Etiqueta	Cantidad
002	10
004	10
006	10
007	10
008	10
019	10
022	10
023	10
032	10
034	10
050	10
065	10
070	10
075	10
079	10

El desempeño del sistema de reconocimiento de iris se muestra en la Tabla 5.2, experimentalmente la CNN fue entrenada en 10 ocasiones tomando en cada entrenamiento diferentes imágenes para cada conjunto de datos e inicializando los pesos de los filtros de manera aleatoria, para obtener los desempeños se aplica la ecuación:

$$\text{Precisión} = \frac{\sum_i V_p(i)}{T_N} \quad (5.1)$$

donde V_p son los verdaderos positivos, es decir, las imágenes que se clasificaron correctamente con los objetivos y T_N es el total de imágenes del conjunto de datos que se está clasificando.

En la Tabla 5.2 se muestran los resultados obtenidos de cada arquitectura implementada: IrisNet, AlexNet y VGG-16. Se entrenaron bajo las mismas condiciones de hardware con un procesador Intel i5 de 7a generación, una GPU GTX 1050 de 4 Gb de video dedicados y utilizando la misma versión de Matlab 2018b.

Tabla 5.2 Desempeños obtenidos de la red neuronal convolucional con los diferentes conjuntos de datos establecidos.

Arquitectura	Desempeño entrenamiento (%)	Desempeño validación (%)	Desempeño prueba (%)	Tiempo clasificación (ms)	Error (%)
IrisNet	100	84.44	93.33	171	6.67
	100	95.55	93.33	218	6.67
	100	91.11	96.66	226	3.34
	100	97.78	93.33	195	6.67
	100	91.11	96.66	239	3.34
	100	80	93.33	188	6.67
	100	86.66	90	205	10
	100	95.56	96.66	177	3.34
	100	93.33	86.66	170	13.34
	100	100	96.66	146	3.34
AlexNet	100	71.11	60	204	40
	100	64.44	40	176	60
	100	66.66	50	191	50
	100	64.44	60	182	40
	100	53.33	56.66	178	43.34
	100	60	70	188	30
	100	80	80	177	20
	100	53.33	56.66	211	43.34
	100	66.67	56.66	214	43.34
	100	77.78	73.33	214	26.67
VGG-16	6.66	6.66	6.66	1060	93.34
	6.66	6.66	6.66	990	93.34
	6.66	6.66	6.66	997	93.34
	6.66	6.66	6.66	1010	93.34
	6.66	6.66	6.66	972	93.34
	6.66	6.66	6.66	1010	93.34
	6.66	6.66	6.66	1020	93.34
	6.66	6.66	6.66	875	93.34
	6.66	6.66	6.66	877	93.34
	6.66	6.66	6.66	937	93.34

Los desempeños generados con cada conjunto de imágenes (entrenamiento, validación y prueba) se analizan para tener una idea del funcionamiento del sistema, y se consideran de mayor importancia los desempeños del conjunto de prueba debido a que no intervino en el entrenamiento.

El tiempo de clasificación del conjunto de pruebas es utilizado para tener una idea de la velocidad de procesamiento de los modelos al clasificar las imágenes, además, el error de desempeño del conjunto de pruebas se utiliza para obtener la desviación estándar y analizar su comportamiento. Los resultados de cada modelo mostrados en la Tabla 5.2 se analizan con más detalle en la sección 5.2.

5.1. Pruebas con diferentes algoritmos de entrenamiento

Se realizaron pruebas en el entrenamiento con el modelo de la red neuronal VGG-16 debido a los malos desempeños obtenidos, mostrados en la Tabla 5.2, se entrenó el algoritmo de entrenamiento de SGDM implementado durante el proceso de la investigación, se implementaron los algoritmos de RMSprop y Adam para determinar si el problema es debido al algoritmo de entrenamiento o debido a la estructura definida de la arquitectura para la solución de esta problemática.

5.1.1. RMSprop

El algoritmo de gradiente descendente estocástico con momento utiliza una tasa de aprendizaje única para todos los parámetros, en otros algoritmos de optimización se busca mejorar el entrenamiento de la red mediante el uso de tasas de aprendizaje que difieren según el parámetro y pueden adaptarse automáticamente a la función de pérdida que se está optimizando, es uno de esos algoritmos que mantiene un promedio móvil de los cuadrados de elementos de los gradientes de los parámetros.

En la Figura 5.3 se muestran los resultados al entrenar el modelo VGG-16 con el algoritmo de entrenamiento RMSprop, generando un entrenamiento bajo de 6,67% de desempeño del modelo.

5.2. Análisis de los resultados

Se presenta el análisis mostrado en la Tabla 5.3 a partir de los resultados obtenidos en la Tabla 5.2 y una comparativa de los componentes requeridos por cada arquitectura, así como el número de capas sin contar la capa de entrada ni la capa de clasificación, el espacio de memoria que ocupa, la cantidad de filtros que se aplican a las imágenes, entre otros.

Tabla 5.3 Análisis de los resultados obtenidos.

Arquitectura	Tiempo promedio (ms)	Desempeño promedio (%)	Desviación estándar del error (%)	Número de capas	Memoria ocupada (MB)	Filtros totales aplicados
IrisNet	193.5	93.66	3.313795407	7	13	248
AlexNet	193.5	60.33	11.59530599	8	136	928
VGG-16	974.8	6.66	1.49796E-14	16	488	4224

En el promediado del tiempo de clasificación la arquitectura IrisNet y la AlexNet tienen un promedio igual, teniendo ventaja sobre la VGG-16.

En la arquitectura IrisNet se obtiene un mejor desempeño con respecto de las otras, en la cual se tiene una menor variación en los desempeños con respecto a el cálculo de la desviación estándar del error.

Los resultados generados para la arquitectura VGG-16 se deben a que su estructura es demasiado grande para la cantidad de imágenes utilizadas, la red no alcanza a converger, por consecuencia se tiene un bajo entrenamiento.

Se obtiene un menor costo computacional al trabajar con menos parámetros, es decir, una menor cantidad de filtros, como resultado se consigue una menor cantidad de memoria ocupada en la arquitectura IrisNet sobre las otras arquitecturas.

En la Figura 5.5 se muestra la matriz de confusión, en ella se indica la comparación de las salidas deseadas (*ground truth*) contra las salidas proporcionadas por de la red neuronal, mostrando en recuadro verde la cantidad de imágenes clasificadas correctamente y en rojo las imágenes clasificadas en una clase distinta.

sin mencionar las capas de normalización, las capas de rectificación lineal y las capas de agrupamiento que se incluyen cada capa convolucional.

Se aplican una serie de procesamientos a las imágenes obtenidas de la base de datos UTIRIS para poder obtener la imagen normalizada del iris. Gracias a la normalización de la imagen y a la eliminación de los párpados como fuente de ruido, se logran resultados aceptables al momento de realizar el entrenamiento de la red para la clasificación, debido a que no se obtienen características que influyan en la determinación de la identificación del individuo y aun manteniendo características suficientes para su correcta clasificación.

Se presentaron problemas en la detección de las regiones del iris y de la pupila al usar la transformada de Hough debido a los cambios de iluminación que se tenían en las diferentes imágenes generando diferentes tamaños de las circunferencias presentes en las imágenes esto debido a cada individuo diferente y a las distancias de capturas de las imágenes. Se solucionó el problema de detección al buscar varias circunferencias para ambas regiones y seleccionando las regiones óptimas para su utilización.

El modelo de red propuesto tuvo un funcionamiento más acertado que AlexNet y VGG-16 teniendo un mejor desempeño promedio de un 93%, un poco por debajo de trabajos relacionados que alcanzaron un 98%. Sin embargo, considerando los recursos computacionales utilizados, la red propuesta es más factible para la creación de aplicaciones que puedan ser introducidas a un sistema embebido.

La limitación por las que no se ha trabajado mucho en el reconocimiento biométrico utilizando CNN es debido a los recursos computacionales necesarios para el entrenamiento de las redes neuronales, sobre todo en que la mayoría requieren la utilización de tarjetas gráficas de video (GPU) que son costosas en el mercado. Otra limitación para el trabajo serían las bases de datos públicas con las que se cuentan con imágenes capturadas en el espectro de onda visible, siendo uno de los propósitos de esta investigación.

La utilización de Matlab con redes neuronales facilita mucho el manejo de diseño y pruebas de las arquitecturas por las librerías y *Toolbox* que tiene, ya que de otra manera el proceso de

realización de las operaciones que realiza en el entrenamiento de los pesos sería complicado por la cantidad de entradas que posee la red como entrada.

Se concluye con los resultados obtenidos en la clasificación de los individuos que si es factible la utilización del ojo humano como identificación biométrica con imágenes capturadas en el espectro de onda visible. Lo anterior con la utilización de una buena arquitectura de una red neuronal convolucional para extraer características de manera automática que una persona no puede captar a simple vista debido a las características aleatorias y únicas que posee cada individuo.

Como complemento al trabajo de investigación sería la implementación del sistema a una máquina embebida pasando el trabajo realizado al lenguaje de Python por las librerías que se cuentan de igual manera como en Matlab y la portabilidad. También la creación de la propia base de datos debido a la poca cantidad de imágenes que se tienen por cada individuo y tener un mejor entrenamiento en la red neuronal, ya que para tener mejores resultados con las CNN se debe de contar con una gran cantidad de imágenes.

REFERENCIAS

- [1] S. Sangeetha and N. Radha, "A new framework for IRIS and fingerprint recognition using SVM classification and extreme learning machine based on score level fusion," *2013 7th Int. Conf. Intell. Syst. Control*, pp. 183–188, 2013.
- [2] P. Samant and R. Agarwal, "Machine learning techniques for medical diagnosis of diabetes using iris images," *Comput. Methods Programs Biomed.*, vol. 157, pp. 121–128, 2018.
- [3] M. Oravec, "Feature extraction and classification by machine learning methods for biometric recognition of face and iris," *ELMAR (ELMAR), 2014 56th Int. Symp.*, no. September, pp. 1–4, 2014.
- [4] M. Shamsi, P. Saad, and A. Rasouli, "Iris segmentation and normalization approach," *J. Teknol. Mklm.*, vol. 19, no. 2, pp. 88–101, 2008.
- [5] M. G Alaslani and L. A. Elrefaei, "Convolutional Neural Network Based Feature Extraction for IRIS Recognition," *Int. J. Comput. Sci. Inf. Technol.*, vol. 10, no. 2, pp. 65–78, 2018.
- [6] K. Nguyen, C. Fookes, A. Ross, and S. Sridharan, "Iris Recognition with Off-the-Shelf CNN Features: A Deep Learning Perspective," *IEEE Access*, pp. 1–9, 2017.
- [7] P. Marecelino, "Transfer learning from pre-trained models," 2018. [Online]. Available: <https://towardsdatascience.com/transfer-learning-from-pre-trained-models-f2393f124751>.
- [8] J. Daugman, "How Iris Recognition Works," *Essent. Guid. to Image Process.*, vol. 41, no. 1, pp. 715–739, 2009.
- [9] H. Proença and L. A. Alexandre, "UBIRIS: A Noisy Iris Image Database," pp. 970–977, 2005.
- [10] H. Proenca, S. Filipe, R. Santos, J. Oliveira, and L. A. Alexandre, "The UBIRIS . v2 : A

- Database of Visible Wavelength Iris Images Captured,” *Analysis*, vol. 32, no. 8, pp. 1529–1535, 2010.
- [11] M. De Marsico, M. Nappi, D. Riccio, and H. Wechsler, “Mobile Iris Challenge Evaluation (MICHE)-I, biometric iris dataset and protocols,” *Pattern Recognit. Lett.*, vol. 57, pp. 17–23, 2015.
- [12] H. Proença and L. A. Alexandre, “The NICE.I: Noisy Iris Challenge Evaluation - Part I,” *IEEE Conf. Biometrics Theory, Appl. Syst. BTAS’07*, pp. 1–4, 2007.
- [13] F. F. Khan, A. Akif, and M. A. Haque, “Iris Recognition using Machine Learning from Smartphone Captured Images in Visible Light,” pp. 26–28, 2017.
- [14] Y. Lecun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [15] A. Gangwar and A. Joshi, “Deepirisnet: Deep Iris Representation With Applications In Iris Recognition And Cross-Sensor Iris Recognition,” *2016 IEEE Int. Conf. Image Process*, vol. 2015-Novem, 2016.
- [16] M. Trokielewicz, “Iris Recognition with a Database of Iris Images Obtained in Visible Light Using Smartphone Camera,” pp. 1–6, 2018.
- [17] J. H. Hall, “BIOMETRIC PERSONAL IDENTIFICATION SYSTEM BASED ON RIS ANALYSS,” no. 19, 1992.
- [18] Z. Othman and A. Satria Prabuwono, “Preliminary study on iris recognition system: Tissues of body organs in iridology,” *Proc. 2010 IEEE EMBS Conf. Biomed. Eng. Sci. IECBES 2010*, no. December, pp. 115–119, 2010.
- [19] M. T. Ibrahim, T. M. Khan, S. A. Khan, M. A. Khan, and L. Guan, “Iris localization using local histogram and other image statistics,” *Opt. Lasers Eng.*, vol. 50, no. 5, pp. 645–654, 2012.
- [20] J. Daugman, “Probing the uniqueness and randomness of iriscodes: Results from 200 billion iris pair comparisons,” *Proc. IEEE*, vol. 94, no. 11, pp. 1927–1934, 2006.

- [21] J. Daugman, “High Confidence Visual Recog of Persons by a test of statistical significance,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 11, pp. 1148–1161, 1993.
- [22] M. Sarai and G. Alejandro, “Avances en el reconocimiento del iris: perspectivas y oportunidades en la investigación de algoritmos biométricos Advances in Iris Recognition: Perspectives and Opportunities of Research in Biometric Algorithms,” vol. 16, no. 3, pp. 267–276, 2012.
- [23] N. Ahmadi and G. Akbarizadeh, “Hybrid robust iris recognition approach using iris image pre-processing, two-dimensional gabor features and multi-layer perceptron neural network/PSO,” *IET Biometrics*, vol. 7, no. 2, pp. 153–162, 2018.
- [24] L. E. Ali, J. Luo, and J. Ma, “Iris recognition from distant images based on multiple feature descriptors and classifiers,” *Int. Conf. Signal Process. Proceedings, ICSP*, pp. 1357–1362, 2017.
- [25] N. F. Soliman, E. Mohamed, F. Magdi, F. E. A. El-Samie, and M. AbdElnaby, “Efficient iris localization and recognition,” *Optik (Stuttg.)*, vol. 140, pp. 469–475, 2017.
- [26] M. S. Hosseini, B. N. Araabi, and H. Soltanian-Zadeh, “Pigment melanin: Pattern for iris recognition,” *IEEE Trans. Instrum. Meas.*, vol. 59, no. 4, pp. 792–804, 2010.
- [27] A. K. Jain, A. Ross, and S. Prabhakar, “An Introduction to Biometric Recognition,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 1, pp. 4–20, 2004.
- [28] Y. Huang, Z. Ma, and M. Xie, “Rapid and effective method of quality assessment on sequence iris image,” *MIPPR 2007 Autom. Target Recognit. Image Anal. Multispectral Image Acquis.*, vol. 6786, p. 678620, 2007.
- [29] Z. Wei, T. Tan, Z. Sun, and J. Cui, “Robust and Fast Assessment of Iris Image Quality,” vol. 100080, pp. 464–471, 2005.
- [30] K. E. Martiana, A. R. Barakbah, S. S. Akmilis, and A. A. Hermawan, “Auto cropping

- on iris image for iridology using histogram analysis,” *2016 Int. Conf. Knowl. Creat. Intell. Comput. KCIC 2016*, pp. 42–46, 2017.
- [31] J. Canny, “A Computational Approach to Edge Detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, 1986.
- [32] H. Proenc and L.A. Alexandre, “Iris segmentation methodology for non-cooperative recognition,” *IEEE Proc. Vis.*, vol. 153, no. 1, pp. 199–205, 2006.
- [33] Y. Chen *et al.*, “A highly accurate and computationally efficient approach for unconstrained iris segmentation,” *Image Vis. Comput.*, vol. 28, no. 2, pp. 261–269, 2010.
- [34] M. Rizon *et al.*, “Object detection using circular Hough,” vol. 2, no. 12, pp. 1606–1609, 2005.
- [35] S. A. Sahmoud and I. S. Abuhaiba, “Efficient iris segmentation method in unconstrained environments,” *Pattern Recognit.*, vol. 46, no. 12, pp. 3174–3185, 2013.
- [36] H. Hofbauer, F. Alonso-Fernandez, J. Bigun, and A. Uhl, “Experimental analysis regarding the influence of iris segmentation on the recognition rate,” *IET Biometrics*, vol. 5, no. 3, pp. 200–211, 2016.
- [37] M. Arsalan, R. A. Naqvi, D. S. Kim, P. H. Nguyen, M. Owais, and K. R. Park, “IrisDenseNet: Robust iris segmentation using densely connected fully convolutional networks in the images by visible light and near-infrared light camera sensors,” *Sensors (Switzerland)*, vol. 18, no. 5, 2018.
- [38] K. Ahuja, R. Islam, F. A. Barbhuiya, and K. Dey, “A preliminary study of CNNs for iris and periocular verification in the visible spectrum,” *Proc. - Int. Conf. Pattern Recognit.*, pp. 181–186, 2017.
- [39] A. S. Al-Waisy, R. Qahwaji, S. Ipson, S. Al-Fahdawi, and T. A. M. Nagem, “A multi-biometric iris recognition system based on a deep learning approach,” *Pattern Anal. Appl.*, vol. 21, no. 3, pp. 783–802, 2018.

- [40] M. B. Lee, H. G. Hong, and K. R. Park, “Noisy ocular recognition based on three convolutional neural networks,” *Sensors (Switzerland)*, vol. 17, no. 12, 2017.
- [41] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 2261–2269, 2017.
- [42] V. Badrinarayanan, A. Kendall, and R. Cipolla, “SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, 2017.
- [43] J. Ahmad, H. Farman, and Z. Jan, “Deep Learning Methods and Applications,” *SpringerBriefs Comput. Sci.*, vol. 7, no. 2013, pp. 31–42, 2019.
- [44] K. W. Kim, H. G. Hong, G. P. Nam, and K. R. Park, “A study of deep CNN-based classification of open and closed eyes using a visible light camera sensor,” *Sensors (Switzerland)*, vol. 17, no. 7, 2017.
- [45] O. Oyedotun and A. Khashman, “Iris nevus diagnosis: Convolutional neural network and deep belief network,” *Turkish J. Electr. Eng. Comput. Sci.*, vol. 25, no. 2, pp. 1106–1115, 2017.
- [46] R. E. Neapolitan and R. E. Neapolitan, *Neural Networks and Deep Learning*. 2018.
- [47] G. E. Hinton, “Rectified Linear Units Improve Restricted Boltzmann Machines,” no. 3.
- [48] G. E. H. Alex Krizhevsky, Ilya Sutskever, “ImageNet Classification with Deep Convolutional Neural Networks,” *LSVRC-2010*, vol. 12, 2012.
- [49] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” *arXiv Prepr. arXiv1502.03167*, 2015.
- [50] Y. LeCun, Y. Bengio, and P. Haffner, “Gradient-Based Learning Applied to Document Recognition YANN,” *Biochem. Biophys. Res. Commun.*, 1998.
- [51] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, “Improving neural networks by preventing co-adaptation of feature detectors,” pp. 1–

- 18, 2012.
- [52] N. Srivastava, G. Hinton, and S. Alex Krizhevsky, Ilya Sutskever, “Dropout: A Simple Way to Prevent Neural Networks from Overfitting,” *Mach. Learn. Res.* 15, 2014.
- [53] K. Wang and A. Kumar, “Cross-spectral iris recognition using CNN and supervised discrete hashing,” *Pattern Recognit.*, vol. 86, pp. 85–98, 2019.
- [54] W. Zhang, C. Wang, and P. Xue, “Application of convolution neural network in Iris recognition technology,” *2017 4th Int. Conf. Syst. Informatics, ICSAI 2017*, vol. 2018-Janua, no. Icsai, pp. 1169–1174, 2018.
- [55] F. Ma, G. Po, S. Sa, and L. Ve, “A deep learning approach for iris sensor model identificatio,” vol. 46, no. 3, pp. 1–8, 2016.
- [56] A. Meyer-Baese and V. Schmid, “Specialized Neural Networks Relevant to Bioimaging,” *Pattern Recognit. Signal Anal. Med. Imaging*, pp. 325–351, 2014.
- [57] [19] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks For Large-Scale Image Recognition,” *arXiv Prepr. arXiv1409.1556*, 2014.
- [58] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, “OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks,” 2013.
- [59] M. H. Beale, M. T. Hagan, and H. B. Demuth, “Deep Learning Toolbox TM Reference How to Contact MathWorks,” 2018.
- [60] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” *Publ. as a Conf. Pap. ICLR 2015*, pp. 1–15, 2014.