

**Centro de Investigación Científica y de Educación
Superior de Ensenada, Baja California**



**Maestría en Ciencias
en Ciencias de la Computación**

**Estudio de métodos para identificar signos de
retinopatía diabética en imágenes de fondo del ojo**

Tesis

para cubrir parcialmente los requisitos necesarios para obtener el grado de
Maestro en Ciencias

Presenta:

Selene Montes Fuentes

Ensenada, Baja California, México

2018

Tesis defendida por

Selene Montes Fuentes

y aprobada por el siguiente Comité

Dr. Hugo Homero Hidalgo Silva
Director de tesis

Dr. Jorge Torres Rodríguez
M.C. Jorge Enrique Preciado Velasco



Dr. Jesús Favela Vara
Coordinador del Posgrado en Ciencias de la Computación

Dra. Rufina Hernández Martínez
Directora de Estudios de Posgrado

Selene Montes Fuentes © 2018

Queda prohibida la reproducción parcial o total de esta obra sin el permiso formal y explícito del autor y director de la tesis

Resumen de la tesis que presenta Selene Montes Fuentes como requisito parcial para la obtención del grado de Maestro en Ciencias en Ciencias de la Computación.

Estudio de métodos para identificar signos de retinopatía diabética en imágenes de fondo del ojo

Resumen aprobado por:

Dr. Hugo Homero Hidalgo Silva

Director de tesis

La diabetes es una enfermedad que se presenta cuando el páncreas no produce suficiente insulina o el cuerpo no la utiliza de manera adecuada, además es una de las principales enfermedades que ha ido incrementando en los últimos años. Se pronostica que para el año 2035 el número de personas con este padecimiento sea de 370 millones. Una de las principales complicaciones de la diabetes es la retinopatía diabética (RD). La RD es la principal causa de ceguera y se presenta cuando son dañados los vasos sanguíneos que se encuentran en la retina a causa de niveles altos de glucosa en la sangre. La RD se clasifica en diversas etapas y en cada una de ellas se presenta cierta cantidad de lesiones como microaneurismas, exudados y hemorragias. Generalmente la detección de la retinopatía diabética se realiza de forma manual por medio de un diagnóstico con aparatos especiales o con imágenes de fondo del ojo, por lo que es una actividad que consume tiempo. Esta tesis propone una metodología para clasificar imágenes de fondo del ojo en las diferentes etapas de la retinopatía diabética de manera automática con el uso de redes neuronales convolucionales. Para este trabajo se utilizaron dos bases de datos de imágenes de fondo del ojo, Messidor y Messidor 2, clasificando los estados de esta enfermedad de acuerdo a la cantidad de lesiones presentadas.

Palabras clave: Diabetes, retinopatía diabética, redes neuronales convolucionales.

Abstract of the thesis presented by Selene Montes Fuentes as a partial requirement to obtain the Master of Science degree in Computer Science.

A study of computational methods for diabetic retinopathy detection on fundus images

Abstract approved by:

Dr. Hugo Homero Hidalgo Silva
Thesis Director

A study of diabetic retinopathy diagnosis aid methods using fundus images is presented. A convolutional neural network methodology is proposed for fundus images classification on several stages of diabetic retinopathy. The study is realized on Messidor and Messidor 2 databases, classifying the images according to the number of lesions.

Keywords: Diabetes, diabetic retinopathy, convolutional neural networks

Dedicatoria

A mi abuelita Esperanza.

Agradecimientos

A mi familia, por su apoyo.

A mi asesor, el Dr. Hugo Homero Hidalgo Silva por su gran apoyo y paciencia.

A los miembros de mi comité de tesis, el Dr. Jorge Torres Rodríguez y el M.C. Jorge Enrique Preciado Velasco, por sus valiosos comentarios.

A mis amigos y compañeros de CICESE.

Al Consejo Nacional de Ciencia y Tecnología (CONACyT) por brindarme el apoyo económico para realizar mis estudios de maestría.

Al Centro de Investigación Científica y de Educación Superior de Ensenada.

Tabla de contenido

	Página
Resumen en español	ii
Resumen en inglés	iii
Dedicatoria	iv
Agradecimientos	v
Lista de figuras	ix
Lista de tablas	xii
Capítulo 1. Introducción	
1.1. Antecedentes	2
1.1.1. Contribución del procesamiento de imágenes al diagnóstico de la retinopatía diabética	3
1.1.1.1. Mejora de la imagen	3
1.1.1.2. Reconocimiento de masa ocular	4
1.1.1.3. Monitoreo de la enfermedad	4
1.2. Justificación	6
1.3. Objetivos	6
1.3.1. Objetivo general	6
1.3.2. Objetivos específicos	6
1.4. Organización del documento	7
Capítulo 2. Marco Teórico	
2.1. Introducción	8
2.2. Área de la salud	8
2.2.1. Anatomía del ojo	8
2.2.2. Diabetes	9
2.2.2.1. Diabetes tipo 1	10
2.2.2.2. Diabetes tipo 2	10
2.2.2.3. Diabetes gestacional	10
2.2.2.4. Consecuencias frecuentes de la diabetes	11
2.2.2.5. Prevención de la diabetes	11
2.2.2.6. Diagnóstico y tratamiento de la diabetes	11
2.2.3. Retinopatía diabética	12
2.2.3.1. Retinopatía diabética no proliferativa (RDNP)	12
2.2.3.2. Lesiones en la RDNP	13
2.2.3.3. Retinopatía diabética proliferativa (RDP)	13
2.2.3.4. Grados en la RDNP	15
2.2.3.5. Diagnóstico de la retinopatía diabética	15
2.2.3.6. Tratamiento de la retinopatía diabética	16
2.2.4. Fotografía de fondo del ojo	17
2.3. Área de computación	20
2.3.1. Inteligencia Artificial (IA)	20
2.3.2. Aprendizaje de máquina	21

Tabla de contenido

2.3.2.1. Aprendizaje supervisado	22
2.3.3. Aprendizaje profundo	22
2.3.4. Clasificación de imágenes	23
2.3.5. Redes neuronales	25
2.3.5.1. Historia de las redes neuronales	26
2.3.5.2. Aplicaciones de las redes neuronales	28
2.3.5.3. Aumento de datos	29
2.3.5.4. Inicialización de la red	30
2.3.5.5. Funciones de activación	30
2.3.5.6. Regularizaciones	34
2.3.6. Redes Neuronales Convolucionales	36
2.3.6.1. La operación de convolución	36
2.3.6.2. Capas de las redes neuronales convolucionales	37
2.3.6.3. Capa Convolutiva	37
2.3.6.4. Capa de Pooling	38
2.3.6.5. Capa completamente conectada	39
Capítulo 3. Metodología	
3.1. Introducción	40
3.2. Bases de datos	41
3.2.1. Messidor	41
3.2.2. Messidor 2	42
3.3. Preprocesamiento de imágenes.	44
3.4. Otro tipo de preprocesamiento	45
3.5. División de las bases de datos	46
3.6. Cálculo de atributos	47
3.7. Entrenamiento	48
3.7.1. Aumento de datos	48
3.7.2. Validación cruzada	49
3.7.3. Diseño de la red neuronal convolutiva	50
3.7.4. Cálculo del porcentaje de las imágenes	52
3.7.5. Cálculo del balanceo de pesos	53
3.7.6. Secuencia de entrenamiento	53
3.8. Medición de la calidad del sistema	54
3.8.1. Matriz de confusión	54
3.8.2. Coeficiente Kappa	56
Capítulo 4. Resultados y discusiones	
4.1. Introducción	58
4.2. Consideraciones preliminares	58
4.3. Digits	58
4.3.1. Ejemplos del resultado de la clasificación	59
4.4. Tensorflow	60
4.4.1. Prueba 1	60
4.4.2. Prueba 2	61

Tabla de contenido

4.4.3. Prueba 3	62
4.4.4. Prueba 4	63
4.5. Resultados y discusiones de Messidor	64
4.5.1. Matriz de confusión de la base de datos Messidor	68
4.6. Resultados y discusiones de Messidor 2	68
4.6.1. Matriz de confusión de la base de datos Messidor 2	71
Capítulo 5. Conclusiones y trabajo futuro	
5.1. Introducción	73
5.2. Conclusiones	73
5.2.1. Aportaciones	74
5.2.2. Trabajo futuro	74
Literatura citada	75

Lista de figuras

Figura	Página
1. Anatomía del ojo. Imagen tomada de SOCV: Sociedad oftalmológica de la comunidad valenciana (2014).	9
2. Visión de una persona con RDNP. Imagen tomada de AAO: American academy of ophthalmology (2015b)	12
3. Lesiones de la RDNP. Imagen tomada de Amin <i>et al.</i> (2016)	13
4. Lesiones de la RDP. Imagen tomada de Amin <i>et al.</i> (2016).	14
5. Visión de una persona con RDP. Imagen tomada de AAO: American academy of ophthalmology (2015a)	14
6. Diagnósticos de la retinopatía diabética. Imagen tomada de Riordan Eva y Cunningham (2012).	16
7. Esquema de fondo del ojo. Imagen tomada de Riordan Eva y Cunningham (2012).	17
8. Fotografía de fondo del ojo. Imagen tomada de AAO: American academy of ophthalmology (2015a).	18
9. Línea del tiempo de la IA. Imagen tomada de NVIDIA (2016).	21
10. Modelo de aprendizaje profundo. Imagen tomada de Goodfellow <i>et al.</i> (2016).	23
11. Clasificación de la imagen de un gato. Imagen tomada de CS231 (2016) .	24
12. Variaciones de imágenes. a) Variación del punto de vista, b) Variación a escala, c) Deformación, d) Oclusión, e) Condiciones de iluminación, f) Confusión con el fondo, g) Variación intra-clase. Imagen tomada de CS231 (2016).	25
13. Topología de una red neuronal.a) Red neuronal con una capa oculta y una capa de salida, con 4 y 2 neuronas respectivamente. b) Red neuronal con 2 capas ocultas con 4 neuronas cada una y una capa de salida. Imagen tomada de Goodfellow <i>et al.</i> (2016)	26
14. Ejemplo de aumento de datos. Imagen tomada de Dieleman (2015).	29
15. ReLU. Imagen tomada de Xu <i>et al.</i> (2015)	31
16. Leaky ReLU y PReLU. Imagen tomada de Xu <i>et al.</i> (2015).	32
17. RReLU. Imagen tomada de Xu <i>et al.</i> (2015).	33
18. Función sigmoide. Imagen tomada de Xu <i>et al.</i> (2015)	33
19. Función tanh. Imagen tomada de Xu <i>et al.</i> (2015)	34
20. Ejemplo de dropout. Imagen tomada de Srivastava <i>et al.</i> (2014)	35
21. Dropout. a) durante el entrenamiento, b) durante la prueba. Imagen tomada de Srivastava <i>et al.</i> (2014).	36

Lista de figuras

Figura	Página
22. Convolución en 2D. Imagen tomada de Goodfellow <i>et al.</i> (2016).	37
23. Componentes de una capa de red convolucional.	39
24. Metodología.	40
25. Ejemplos de imágenes de acuerdo al grado de la Retinopatía diabética. Imágenes tomadas de Decencièrè <i>et al.</i> (2014)	42
26. Ejemplos de imágenes de acuerdo al grado de la Retinopatía diabética. Imágenes tomadas de Decencièrè <i>et al.</i> (2014).	44
27. a) Imagen original, b) Imagen cortada.	45
28. a) Imagen original, b) Imagen cortada.	45
29. División de las bases de datos. a) Imágenes originales, b) Imágenes cor- tadas, c) Imágenes divididas.	47
30. Validación cruzada.	50
31. Ejemplo de validación cruzada.	50
32. Secuencia de entrenamiento para las bases de datos.	54
33. Imágenes de entrenamiento en Digits.	59
34. Imágenes clasificadas correctamente	59
35. Imágenes clasificadas incorrectamente.	60
36. Imágenes clasificadas incorrectamente.	61
37. Imágenes clasificadas incorrectamente.	62
38. Imágenes clasificadas incorrectamente.	63
39. Imágenes clasificadas incorrectamente.	64
40. Pérdida de entrenamiento y validación en imágenes de 128x128 pixeles. .	65
41. Índice Kappa en el entrenamiento de imágenes de 128x128 pixeles	66
42. Pérdida de entrenamiento y validación en imágenes de 256x256 pixeles. .	66
43. Índice Kappa en el entrenamiento de imágenes de 256x256 pixeles	67
44. Pérdida de entrenamiento y validación en imágenes de 512x512 pixeles. .	67
45. Índice Kappa en el entrenamiento de imágenes de 512x512 pixeles.	68
46. Pérdida de entrenamiento y validación en imágenes de 128x128 pixeles. .	69
47. Índice Kappa en el entrenamiento de imágenes de 128x128 pixeles	69

Lista de figuras

Figura	Página
48. Pérdida de entrenamiento y validación en imágenes de 256x256 píxeles. .	70
49. Índice Kappa en el entrenamiento de imágenes de 256x256 píxeles	70
50. Pérdida de entrenamiento y validación en imágenes de 512x512 píxeles. .	71
51. Índice Kappa en el entrenamiento de imágenes de 512x512 píxeles.	71

Lista de tablas

Tabla		Página
1.	Imágenes por clase de Messidor.	42
2.	Grados de la retinopatía diabética en Messidor.	42
3.	Imágenes por clase de Messidor 2	43
4.	Grado de la retinopatía diabética en Messidor 2.	43
5.	Atributos de Messidor	48
6.	Atributos de Messidor 2	48
7.	Estructura de la red	51
8.	Tasa de aprendizaje	52
9.	Porcentaje de las imágenes de Messidor	52
10.	Porcentaje de las imágenes de Messidor 2	52
11.	Balanceo de pesos en Messidor	53
12.	Balanceo de pesos en Messidor 2	53
13.	Ejemplo de matriz de confusión	54
14.	Matriz de confusión de la prueba 1.	61
15.	Matriz de confusión de la prueba 2.	62
16.	Matriz de confusión de la prueba 3.	63
17.	Matriz de confusión de la prueba 4.	64
18.	Capas usadas en los diferentes tamaños de imágenes.	65
19.	Matriz de confusión de Messidor.	68
20.	Capas usadas en los diferentes tamaños de imágenes.	69
21.	Matriz de confusión de Messidor 2	72

Capítulo 1. Introducción

La diabetes es una enfermedad que está incrementando de manera significativa en todo el mundo, de acuerdo con la Organización Mundial de la Salud el número de personas con este padecimiento en 1980 era de 108 millones, pasando a 422 millones en el año 2014 (OMS: Organización mundial de la salud, 2017).

La retinopatía diabética es una complicación visual a causa de la diabetes y se desarrolla cuando hay niveles altos de glucosa en la sangre, por lo que se producen alteraciones en los vasos sanguíneos dañando la retina y provocando la ceguera.

Algunas ventajas de un diagnóstico asistido por computadora automático o semi-automático son:

- Disminución del tiempo de examinación.
- Disminución de recursos necesarios en términos de especialistas.

El diagnóstico de la retinopatía diabética asistido por computadora se realiza a través de procesamiento de imágenes y se divide en tres conjuntos:

- Detección automática de patologías como los microaneurismas, exudados, manchas algodonosas, hemorragias y edemas.
- Detección automática de las características de la retina como el árbol vascular y disco óptico.
- Medidas sobre patologías detectadas que son difíciles o consumen mucho tiempo para ser realizadas manualmente.

En el siguiente trabajo de investigación se presenta un método para identificar las diferentes etapas de la retinopatía diabética en imágenes de fondo del ojo, las cuales presentan lesiones como microaneurismas, exudados y hemorragias. Se presenta un método elaborado de aprendizaje profundo mediante el uso de redes convolucionales.

1.1. Antecedentes

Actualmente existen diversas técnicas para identificar la retinopatía diabética en imágenes de fondo del ojo. Sinthanayothin *et al.* (1999) identificaron los principales componentes de la retina como el disco óptico, los vasos sanguíneos y la fovea. El disco óptico fue reconocido identificando el área con una variación alta en la intensidad de píxeles adyacentes. Los vasos sanguíneos fueron identificados por medio de una red neuronal perceptron multicapa, cuya entrada se deriva de un análisis de componentes principales. La fovea fue localizada usando emparejamiento por correlación.

Posteriormente Sinthanayothin *et al.* (2002) proponen los siguientes pasos para identificar los signos de la retinopatía diabética:

- Preprocesamiento del color de la imagen: En este paso realizan una mejora de la imagen usando el modelo de color HSI.
- Identificación de los principales componentes de la retina.
- Reconocimiento de las lesiones de la retinopatía diabética: En esta etapa se identifican las principales lesiones como exudados, hemorragias y microaneurismas.

Por otro lado Abràmoff *et al.* (2008) realizan la siguiente secuencia de operaciones aplicadas a imágenes de la retina.

- Las imágenes se submuestran dejándolas en un tamaño de 640 x 640 píxeles.
- Se determina la probabilidad de que la imagen tenga una buena calidad para su evaluación.
- Se realiza segmentación automática de los vasos sanguíneos usando un clasificador de características de píxeles.
- La detección y enmascaramiento automático del disco óptico es realizada.
- Se realiza la detección automática de microaneurismas y hemorragias.
- Se realiza la detección automática de exudados.

Abramoff *et al.* (2010) proponen una serie de algoritmos que resultan ser eficientes para la identificación de los componentes en la retina. Desarrollaron un algoritmo para detectar hemorragias y anomalías vasculares con una sensibilidad del ciento por ciento.

Abramoff *et al.* (2010) también desarrollaron un algoritmo para la correcta localización del disco óptico y la fovea. El algoritmo se ejecuta en 12 segundos en una computadora estándar. Para una identificación profunda, se separan las lesiones como los exudados y las manchas algodonosas.

Oloumi *et al.* (2013) detectan los vasos sanguíneos mediante el uso de filtros Gabor, posteriormente una binarización de ellos. Un aporte importante es que desarrollan la detección de parábolas y semi parábolas usando la transformada de Hough, que es una técnica para la detección de figuras en imágenes digitales. Utilizan la base de datos Análisis de la Retina Estructurada (STARE, por sus siglas en inglés), la cual contiene imágenes de tamaño 700 x 605 píxeles.

Silberman *et al.* (2010) realizan la detección de la RD entrenando una máquina de soporte vectorial. Primero realizan un preprocesamiento que consiste en balancear el color en imágenes oscuras para obtener una buena apariencia, se aumenta la luminosidad a los píxeles, se realiza la extracción de bordes y convierte la imagen a escala de grises. Posteriormente el disco óptico es removido para la detección de exudados.

1.1.1. Contribución del procesamiento de imágenes al diagnóstico de la retinopatía diabética

La contribución del procesamiento de imágenes para el diagnóstico de la retinopatía diabética es muy importante, por lo que (Walter *et al.*, 2002) la dividen en tres partes:

1.1.1.1. Mejora de la imagen

Las imágenes tomadas por lo regular tienen ruido y bajo contraste, por lo que se usan técnicas mejorando el contraste, la nitidez y reduciendo el ruido.

1.1.1.2. Reconocimiento de masa ocular

La tarea del procesamiento de imágenes asistido por computadora más importante, es el reconocimiento de masa ocular para el diagnóstico de la retinopatía diabética.

1.1.1.3. Monitoreo de la enfermedad

Se comparan imágenes tomadas en exámenes médicos en cada paciente para evaluar la evolución de la enfermedad, así como la eficiencia de nuevos tratamientos en los tratamientos diabéticos y oftalmológicos y observar el desarrollo de las lesiones.

Ali Shah *et al.* (2016) realizan una detección automática de microaneurismas en un sistema basado en la transformada de Curvelet, la cual se propone como un método de análisis de imágenes. Para examinar estas lesiones se realizan dos pasos importantes:

- Los vasos sanguíneos son removidos en el preprocesado de la imagen y se seleccionan las imágenes candidatas con microaneurismas mediante una técnica de umbralización.
- Se estima el fondo de la imagen por medio de rasgos estadísticos.

La base de datos que utilizó Ali Shah *et al.* (2016) fue la de Retinopathy Online Challenge (ROC) que consiste de 50 imágenes. La detección automática de microaneurismas a partir de imágenes de fondo del ojo todavía sigue siendo una cuestión abierta, por ello Adal *et al.* (2014) realizan la búsqueda en regiones de interés y desarrollan una técnica de selección automática de escala local. Además proponen un enfoque de aprendizaje semisupervisado que requiere pocos ejemplos de entrenamiento para un clasificador. El sistema es desarrollado con pocas imágenes etiquetadas de forma manual y una gran cantidad de imágenes no etiquetadas.

Antal y Hajdu (2012) proponen métodos efectivos para la detección de microaneurismas tales como:

- Mejoramiento de contraste Walter-Klein: Consiste en mejorar el contraste en el fondo de una imagen aplicando una transformación de nivel de grises usando operadores.

- Ecualización del histograma adaptativo: Es una técnica biomédica popular usada en el procesamiento de imágenes. La imagen es dividida en regiones disjuntas y a cada una se aplica una ecualización del histograma.
- Removimiento de vasos y extrapolación
- Ecualización e iluminación: Este método tiene como objetivo reducir los efectos negativos causados por la iluminación desigual de las imágenes de la retina.

Lazar y Hajdu (2013) proponen un método que realiza la detección de microaneurismas a través del análisis de los perfiles de sección transversal centradas en píxeles máximos locales de la imagen preprocesada. La detección de picos se aplica en cada perfil y se calculan atributos respecto al tamaño, altura y forma del pico. La base de datos utilizada fue la de ROC y también se hace uso de una colección de imágenes privada usando las mismas características.

Santhi *et al.* (2016) proyectan tres técnicas de segmentación para detectar exudados y manchas algodonosas en imágenes de la retina. Se utiliza la base de datos STARE. Las técnicas utilizadas son:

- Región de crecimiento.
- Región de crecimiento con corrección de fondo.
- Región de adaptación creciente con corrección de fondo.

También se usa una red neuronal para la clasificación de imágenes normales y anormales. Para clasificar las lesiones brillantes se utiliza una red neuronal probabilística, una red base radial exacta y una red base radial con pocas neuronas, lo cual se logra con una precisión del 100%.

La detección robusta de lesiones rojas en las fotografías en color de fondo del ojo digitales es un paso crítico en el desarrollo de sistemas de detección automatizados para la retinopatía diabética. Niemeijer *et al.* (2005) presentan un nuevo método para la detección de lesiones de color rojo con dos contribuciones importantes. La primera es un sistema de detección basado en clasificación de píxeles, por lo que las lesiones son separadas del fondo de la imagen. La segunda es el uso del clasificador de los k

vecinos más cercanos.

1.2. Justificación

La detección de la retinopatía diabética es una actividad que consume tiempo y generalmente se realiza de forma manual, debido a que se requiere de un médico capacitado para evaluar imágenes a color de la retina. Un problema común es la insuficiencia de médicos entrenados y de infraestructura necesaria en regiones donde se presenta esta enfermedad. Si el número de personas con diabetes aumenta, entonces será escasa tanto la infraestructura como el personal para la prevención de la ceguera causada por la RD.

De lo anteriormente expuesto surge la necesidad de un procedimiento automatizado y completo, desarrollado mediante técnicas de clasificación de imágenes usando el aprendizaje automático y las redes convolucionales.

1.3. Objetivos

1.3.1. Objetivo general

Proponer una metodología para clasificar imágenes de fondo del ojo en las diferentes etapas de la retinopatía diabética y obtener un procedimiento eficaz que permita el diagnóstico de la enfermedad.

1.3.2. Objetivos específicos

- Estudiar procedimientos para la detección de lesiones presentadas en la RDNP.
- Diseñar una red neuronal convolucional.
- Implementar métodos basados en aprendizaje profundo para la clasificación de imágenes de fondo del ojo.

1.4. Organización del documento

Este trabajo de tesis se encuentra dividido en 5 capítulos y se distribuye de la siguiente manera:

- Capítulo I: Se presenta una breve introducción del trabajo de investigación describiendo los antecedentes, la justificación y los objetivos generales y específicos.
- Capítulo II: En este capítulo se presenta el marco teórico dividido en dos secciones; el área de la salud y el área de computación donde se describen los conceptos teóricos que se utilizan para el desarrollo de este trabajo. En el área de la salud se presenta una descripción de las partes del ojo para entender en que región se sitúan las lesiones de la enfermedad, además se describe la retinopatía diabética y el tipo de lesiones que presenta así como sus diferentes etapas. En el área de computación se describe a detalle el aprendizaje profundo y las redes convolucionales, los cuales fueron utilizadas para el desarrollo de este trabajo.
- Capítulo III: En este capítulo se describe la metodología del trabajo de investigación que se realizó y se presentan las bases de datos utilizadas.
- Capítulo IV: En este capítulo de resultados, se describe la estructura de la red convolucional implementada y se presentan los experimentos.
- Capítulo V: Se presentan las conclusiones más relevantes sobre el trabajo reportado y se proponen los trabajos futuros.

Capítulo 2. Marco Teórico

2.1. Introducción

Este trabajo de investigación tiene como objetivo clasificar imágenes de fondo del ojo en las diferentes etapas de la Retinopatía diabética. Este capítulo se divide en dos secciones; área de la salud y área de computación. En la primera sección se describen varias definiciones fundamentales como: la anatomía del ojo, la diabetes y sus tipos, la retinopatía diabética, sus lesiones y sus etapas, así como el tratamiento y diagnóstico de esta enfermedad, por último se describe la fotografía de fondo del ojo. En la segunda sección se describen las definiciones de aprendizaje profundo, redes neuronales convolucionales y métricas de evaluación.

2.2. Área de la salud

2.2.1. Anatomía del ojo

La vista es uno de los sentidos más importantes del cuerpo humano, gracias a ella se pueden percibir la forma, tamaño, posición, distancia y color de cualquier objeto. Los ojos son los receptores de este sentido. Para tener una buena comprensión de la diabetes y de las lesiones en la retinopatía diabética, se describe brevemente las partes del ojo y sus funciones.

- Esclerótica: Es una capa fibrosa que forma la parte blanca del ojo, su principal función es proteger para que soporte la presión intraocular y no se deforme. Se encuentra cubierta por una mucosa transparente denominada conjuntiva.
- Coroides: Se encarga de mantener la temperatura y nutrir estructuras del globo ocular.
- Retina: Es la capa de tejido nervioso más interna del ojo cuya función es captar la luz y convertirla en un impulso nervioso, para ello cuenta con fotorreceptores, un tipo especial de células para detectar la luz.
- Pupila: Es un orificio situado en el iris, específicamente en la parte central. En ella penetra la luz al interior del globo ocular y tiene como función regular la cantidad de luz que se introduce.

- Iris: Es la parte coloreada del ojo y puede ser de diferentes colores.
- Córnea: Parte frontal transparente del ojo cubre el iris y permite la refracción y transmisión de la luz.
- Mácula: Es una zona de la retina que se encarga de la visión central. En el centro de la mácula está la fovea y permite la visión clara de los detalles de objetos, así como leer.
- Nervio óptico: Se encarga de transportar los impulsos nerviosos originados por la luz de la retina al cerebro, donde son procesadas las imágenes. En la Figura 1 se aprecian algunas partes del ojo anteriormente mencionadas.

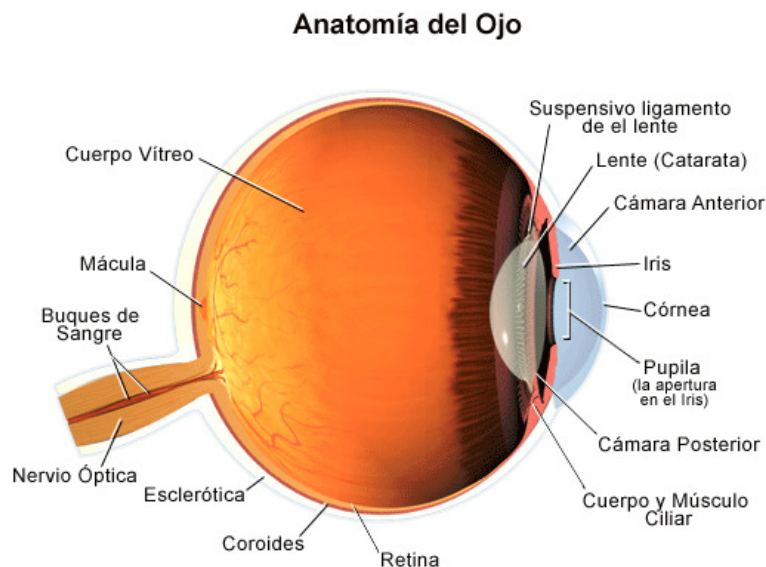


Figura 1. Anatomía del ojo. Imagen tomada de SOCV: Sociedad oftalmológica de la comunidad valenciana (2014).

2.2.2. Diabetes

La Diabetes es un padecimiento crónico que se presenta cuando el páncreas no tiene un buen funcionamiento, es decir no produce suficiente insulina y el cuerpo no la utiliza de manera adecuada. La insulina es la hormona que se encarga de regular los niveles de glucosa en la sangre, cuando estos niveles se incrementan se produce la hiperglucemia que a través del tiempo va dañando órganos y sistemas como los nervios y vasos sanguíneos (FMD: Federación mexicana de la diabetes, 2017). De acuerdo con la OMS: Organización mundial de la salud (2017), la diabetes está clasificada en 3

tipos:

2.2.2.1. Diabetes tipo 1

Este tipo de diabetes se caracteriza por la escasa producción de insulina, por lo que se necesitan dosis diarias de esta hormona. Los síntomas de la diabetes tipo 1 son:

- Excreción excesiva de orina
- Sed
- Hambre constante
- Pérdida de peso
- Trastornos visuales
- Cansancio

2.2.2.2. Diabetes tipo 2

La diabetes tipo 2 se caracteriza por el uso ineficaz de la insulina, es el tipo de diabetes que tienen más personas en todo el mundo. Los síntomas son muy parecidos a los de la diabetes tipo 1 pero son menos intensos. Este padecimiento solo se puede detectar cuando ha estado presente durante varios años y aparecen los síntomas o complicaciones.

2.2.2.3. Diabetes gestacional

En este tipo de diabetes se presenta la hiperglucemia y aparece en el embarazo, los valores que se presentan son ligeramente altos pero no lo suficiente para que se diagnostique la enfermedad.

2.2.2.4. Consecuencias frecuentes de la diabetes

Con la evolución de la enfermedad, la diabetes puede afectar al corazón, vasos sanguíneos, riñones, nervios y ojos. Algunas consecuencias son:

- Riesgo de presentar un infarto al miocardio o un accidente cerebrovascular.
- Se pueden desarrollar úlceras en los pies, causar infección y amputación.
- Padecer insuficiencia renal.
- Padecer retinopatía diabética, que es una causa importante de la ceguera.

2.2.2.5. Prevención de la diabetes

Se ha comprobado que existen medidas para prevenir la diabetes o posponer su desarrollo y se relacionan con el cuidado de la salud. Algunas recomendaciones para prevenir esta enfermedad son:

- Mantener un peso corporal bueno.
- Tener una actividad física constante.
- Tener una dieta saludable evitando azúcares y grasas saturadas.
- Evitar el consumo del tabaco.

2.2.2.6. Diagnóstico y tratamiento de la diabetes

El diagnóstico de la enfermedad se realiza cuando se presentan los síntomas más comunes y se hace mediante un análisis de sangre. El tratamiento consta de una dieta balanceada y de actividad física, algunas recomendaciones son:

- Control de glucosa en la sangre mediante la insulina para la diabetes tipo 1 o la toma de medicamentos orales para la diabetes tipo 2.
- Control de la tensión arterial.

- Cuidados podológicos.
- Control de lípidos en la sangre (regulación del colesterol).
- Detección de los signos de la nefropatía.
- Pruebas para la detección de la retinopatía diabética.

2.2.3. Retinopatía diabética

La retinopatía diabética (RD) es una complicación visual, crónica y específica de la diabetes mellitus, se presenta cuando son dañados los pequeños vasos sanguíneos que se encuentran en la retina a causa de altos niveles de glucosa en la sangre. Con el tiempo la RD puede causar una pérdida en la visión o incluso provocar ceguera. Ambos ojos se ven afectados con este padecimiento y las personas no sufren cambios en la visión al principio de la enfermedad (NEI:National eye institute, 2016).

2.2.3.1. Retinopatía diabética no proliferativa (RDNP)

La retinopatía diabética no proliferativa es la etapa más temprana de la enfermedad y se presenta cuando los vasos sanguíneos dejan escapar fluidos de sangre dentro del ojo como depósitos de colesterol, por lo que pueden introducirse en la retina. En la Figura 2 se observa como ve una persona con retinopatía diabética no proliferativa, se nota una ligera mancha que hace que la persona no vea con claridad las letras.

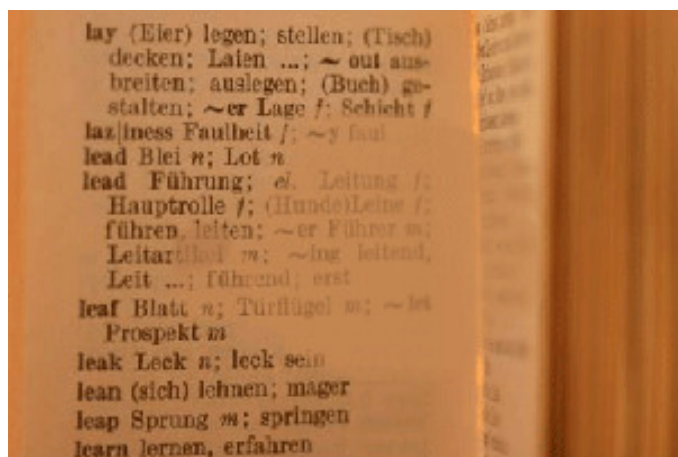


Figura 2. Visión de una persona con RDNP. Imagen tomada de AAO: American academy of ophthalmology (2015b)

2.2.3.2. Lesiones en la RDNP

La RDNP presenta diversas lesiones, tales como:

- **Microaneurismas:** Son pequeños puntos rojos en la imagen de la retina y son indicadores tempranos de coágulos de sangre, se caracterizan por ser el primer signo identificable de la retinopatía diabética.
- **Exudados:** Son manchas de color amarillo brillante que indican depósitos de grasa concentrada en la retina y se caracterizan por contribuir a la formación de coágulos y derrames.
- **Hemorragias:** Son manchas rojas (regiones manchadas de sangre).

En la Figura 3 se muestran las lesiones que presenta la retinopatía diabética no proliferativa.

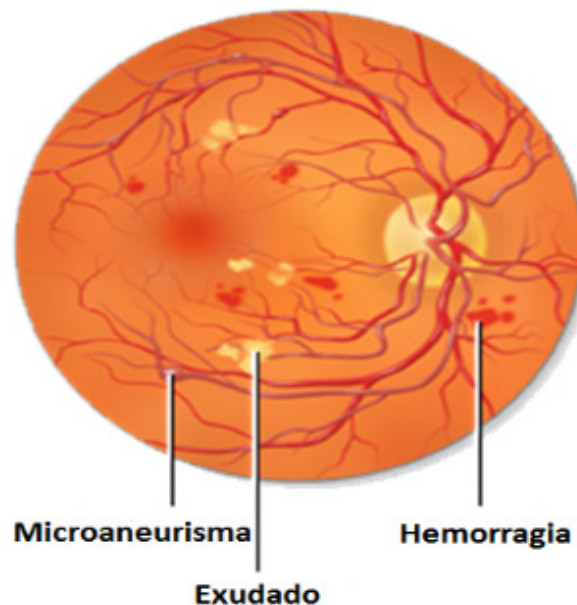


Figura 3. Lesiones de la RDNP. Imagen tomada de Amin *et al.* (2016)

2.2.3.3. Retinopatía diabética proliferativa (RDP)

Es la etapa más avanzada de la enfermedad y se produce cuando muchos de los vasos sanguíneos de la retina se tapan impidiendo un flujo suficiente de la sangre. Esto provoca el desarrollo de nuevos vasos sanguíneos anormales y frágiles. Estos vasos

tienen paredes muy delgadas y pueden escurrir sangre, lo que causaría una pérdida severa en la visión o incluso la ceguera. En la Figura 4 se muestran las lesiones que presenta la retinopatía diabética proliferativa.

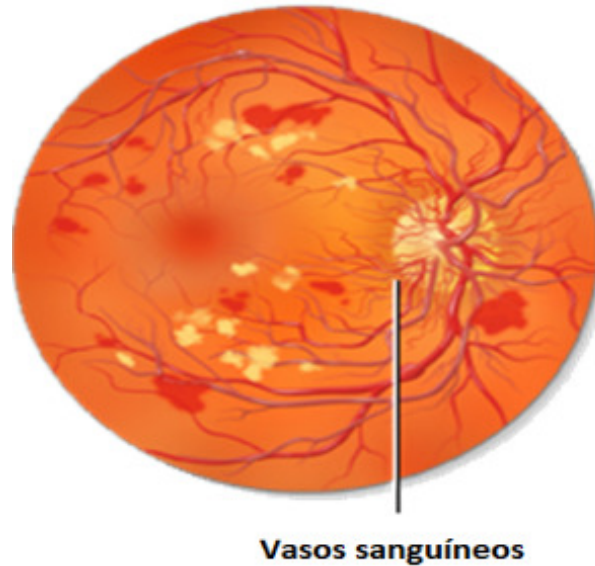


Figura 4. Lesiones de la RDP. Imagen tomada de Amin et al. (2016).

En la Figura 5 se muestra como ve una persona con retinopatía diabética proliferativa, se notan pequeñas manchas oscuras, por lo que la visión se ve muy afectada.

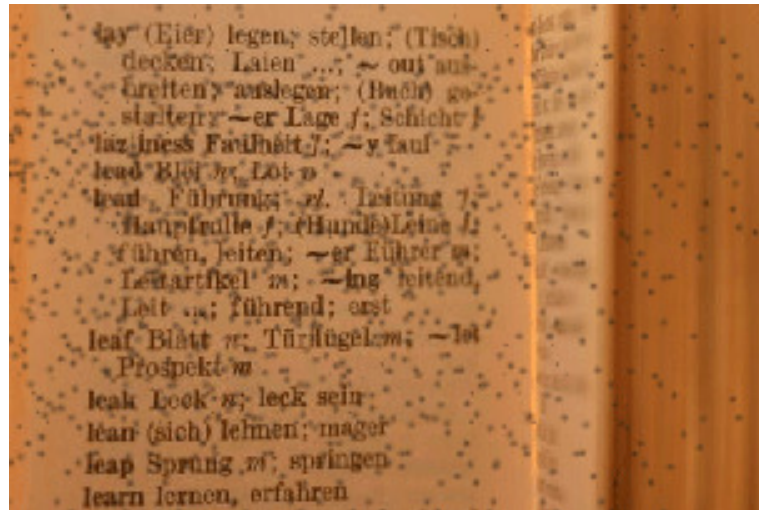


Figura 5. Visión de una persona con RDP. Imagen tomada de AAO: American academy of ophthalmology (2015a)

2.2.3.4. Grados en la RDNP

- Retinopatía no proliferativa leve: Es la primera etapa de la enfermedad donde aparecen los microaneurismas.
- Retinopatía no proliferativa moderada. En esta etapa se obstruyen algunos vasos sanguíneos que alimentan la retina.
- Retinopatía no proliferativa severa. En esta etapa se obstruyen más vasos sanguíneos que en la etapa moderada, por lo que varias regiones de la retina dejan de recibir sangre. Como consecuencia se produce la neovascularización, que es el crecimiento de nuevos vasos sanguíneos.

2.2.3.5. Diagnóstico de la retinopatía diabética

Para detectar la retinopatía diabética y saber la etapa en la que se encuentra, es necesario un examen ocular completo, el cual consiste en las siguientes pruebas:

- Prueba de agudeza visual: Consta en observar una tabla de optometría para determinar que tan bien se ven los detalles de un objeto a diferentes distancias. Esta prueba se realiza en cada ojo por separado, tapando el que no se está evaluando.
- Examen con lámpara de hendidura: Consiste en examinar con un microscopio especial la parte frontal del ojo, párpados, córnea, conjuntiva, esclera, cámara anterior, cristalino, partes de la retina y nervio óptico.
- Examen por dilatación: Con la finalidad de dilatar los ojos, son puestas unas gotas en ellos, lo cual permite observar detalladamente la retina y nervio óptico para saber si existe algún daño.
- Angiografía con fluorescencia: Es un procedimiento que usa una cámara especial para capturar imágenes de la retina, antes de esto se inyecta en una vena del brazo fluoresceína que es un tinte amarillo. El resultado de este procedimiento arroja lo siguiente:
 - Vasos sanguíneos que dejan escapar los fluidos.

- Cantidad de fluido que ha salido.
 - Existencia de vasos sanguíneos tapados.
 - Presencia de neovascularización.
- Tomografía de coherencia óptica: Consiste en un método de exploración no agresivo que provee imágenes de alta resolución de la retina para que sea evaluado el espesor de la misma, así como la presencia de algún edema macular.
 - Ultrasonido: Este procedimiento se realiza cuando no es posible observar adecuadamente la retina por alguna hemorragia, el objetivo de esta prueba es verificar si la retina ha sufrido un desprendimiento.

En la Figura 6 Se muestra una imagen con algunos diagnósticos de la retinopatía diabética. a) Cartelones para la prueba de agudeza visual, b) Examen con lámpara de hendidura, c) Angiografía con fluorescencia, d) Tomografía de coherencia óptica, e) Ultrasonido donde C, córnea, I, iris, L, superficie posterior del cristalino, O, nervio óptico, R, retina, T, tumor.

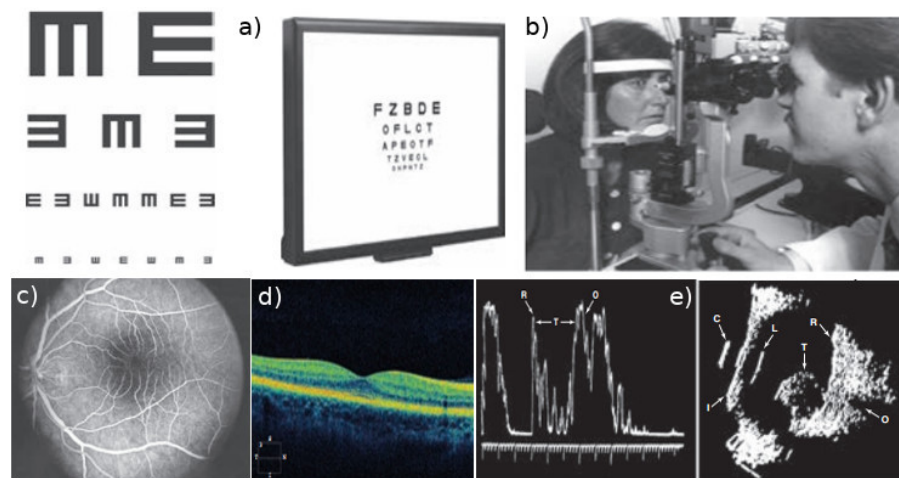


Figura 6. Diagnósticos de la retinopatía diabética. Imagen tomada de Riordan Eva y Cunningham (2012).

2.2.3.6. Tratamiento de la retinopatía diabética

Las personas con este padecimiento tienen que controlar los niveles de glucosa en la sangre, la presión arterial y el colesterol para disminuir el riesgo de la pérdida de visión. El tratamiento no cura la retinopatía diabética ni reestablece la visión normal.

Sin tratamiento la enfermedad avanza progresivamente. Existen algunas técnicas para tratar la retinopatía diabética y son:

- Cirugía láser o fotocoagulación retiniana: Ayuda a disminuir los vasos sanguíneos anormales y reduce la hinchazón macular, si existe. Durante la intervención, se coloca temporalmente en el ojo un lente de contacto para enfocar la luz del láser en la retina.
- Vitrectomía: Se realiza cuando existe una hemorragia severa, se utilizan instrumentos quirúrgicos para extraer la sangre y tejido cicatrizado con el objetivo de que haya un posicionamiento en la retina y los rayos de luz se enfoquen nuevamente en ella.
- Inyección de medicamentos: Son recetadas medicinas con esteroides o un medicamento para tratar el factor de crecimiento vascular endotelial. Después de una dilatación, el ojo es inyectado en el vítreo, así el medicamento aminora la hinchazón y el escape de fluidos, también puede mejorar la visión.

2.2.4. Fotografía de fondo del ojo

La fotografía de fondo del ojo es una técnica de obtención de una imagen de la superficie interior del ojo que incluye la retina, el disco óptico, la mácula y el polo posterior (fondo). En la Figura 7 se observa un esquema de fondo del ojo.

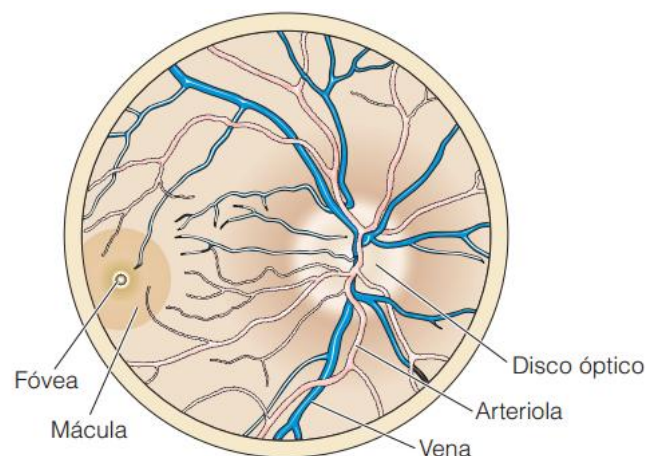


Figura 7. Esquema de fondo del ojo. Imagen tomada de Riordan Eva y Cunningham (2012).

La fotografía de fondo del ojo es usada especialmente por oftalmólogos, optometristas y personas especializadas en la medicina para el diagnóstico o progresión de enfermedades del ojo.

La historia de la fotografía de fondo del ojo surge en el año de 1862 cuando se realiza la primera fotografía de fondo del ojo de un conejo por el Dr. Noyes, posteriormente Rosenbruch fotografía la retina de un gato y en 1886 se realiza la primer fotografía de la retina humana con calidad baja, hecha por el Dr. Howe (Pérez Molina y León Veitía, 2017).

La combinación con los sistemas digitales en las cámaras de fondo del ojo facilita la manipulación y almacenamiento de las imágenes, gracias a ello existen diversas bases de datos con grandes cantidades de imágenes. Actualmente los instrumentos que se utilizan para capturar la imagen del ojo son complicados en el diseño por lo que son difíciles de fabricar con los estándares clínicos y algunos fabricantes son: Topcon, Zeiss, Canon, Nidek, Kowa. En la Figura 8 se observa una fotografía de fondo del ojo tomada con una cámara Topcon.

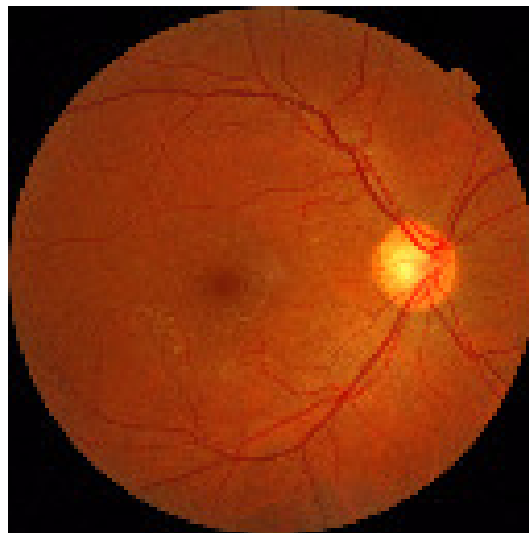


Figura 8. Fotografía de fondo del ojo. Imagen tomada de AAO: American academy of ophthalmology (2015a).

Las cámaras que se usan para hacer las fotografías de fondo del ojo son de gran utilidad para el control y diagnóstico de enfermedades oculares como la retinopatía diabética o hipertensión ocular y se clasifican en dos grupos:

- Cámaras no midriáticas: Se caracterizan por ser pequeñas y muestran una ima-

gen infrarroja de la retina, además cuentan con un flash. Para capturar la fotografía, es necesario que la retina esté alineada y enfocada. Toma imágenes de la retina y de fondo del ojo sin la necesidad de midriasis (dilatación de la pupila).

- Cámaras midriáticas: Usan la luz visible, por lo que es necesario la dilatación de la pupila del paciente, algunas cámaras están configuradas para hacer fotografías simultáneas de la retina. La separación entre la lente es fija, además se elimina el posible problema del movimiento del paciente entre dos exposiciones distintas.

2.3. Área de computación

2.3.1. Inteligencia Artificial (IA)

La inteligencia artificial surgió en la década de 1950, particularmente a partir de las conferencias en Dartmouth (McCarthy *et al.*, 2006) realizadas en 1956 por John McCarthy y Marvin L. Minsky, donde se reunieron científicos que trabajaban en el nuevo campo de la IA. Algunos problemas de la IA que se propusieron en la conferencia son:

- Computadoras Automáticas
- ¿Cómo se puede programar a una computadora para que use un lenguaje?
- Redes Neuronales
- Teoría de la dimensión de cálculo
- Autoperfeccionamiento
- Abstracciones
- Aleatoriedad y Creatividad

En los primeros días de la IA, se resolvieron problemas intelectualmente difíciles para los humanos y relativamente fáciles para las computadoras, es decir problemas que son descritos por reglas matemáticas. El gran desafío de la IA es solucionar problemas que se resuelvan intuitivamente como reconocimiento de voz y reconocimiento de rostros en imágenes.

En la Figura 9 se muestra una línea del tiempo de la IA, se aprecian dos subconjuntos de la IA; el aprendizaje de máquina y el aprendizaje profundo.

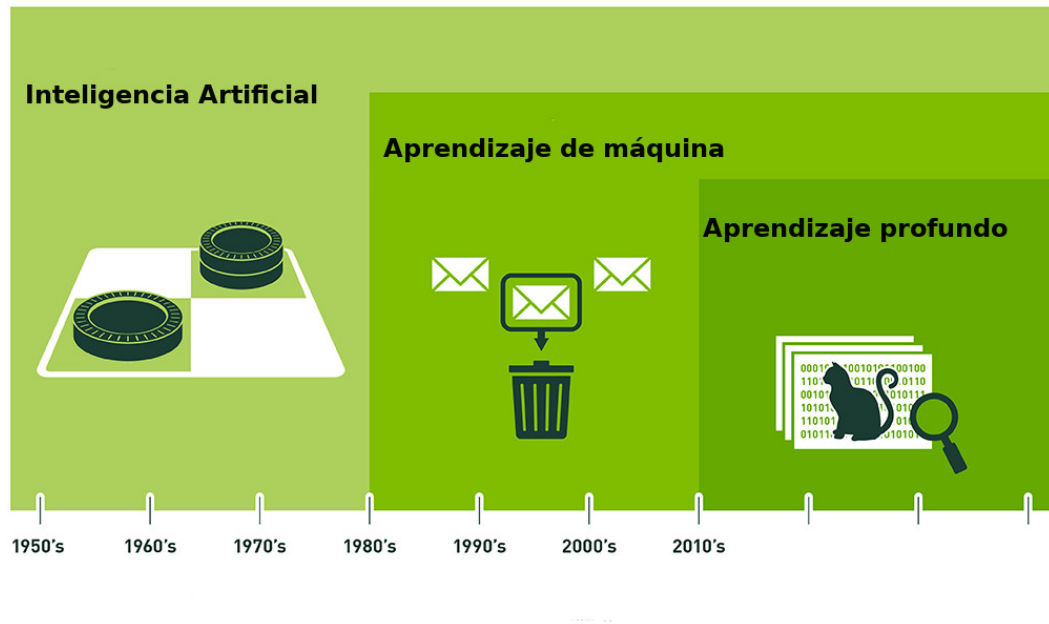


Figura 9. Línea del tiempo de la IA. Imagen tomada de NVIDIA (2016).

2.3.2. Aprendizaje de máquina

El aprendizaje de máquina se define como el campo de estudio que permite a las computadoras aprender por medio de métodos computacionales. Se dice que un programa de computación aprende de la experiencia E con respecto a una tarea T y alguna medida de rendimiento P . Lo anterior se cumple si el rendimiento en la tarea T , medido por P , mejora con la experiencia E (Mitchell, 1997).

Algunos problemas del aprendizaje de máquina son:

- Clasificación.
- Regresión o predicción.
- Ordenamiento.
- Agrupamiento.
- Reducción de la dimensionalidad.

2.3.2.1. Aprendizaje supervisado

El aprendizaje supervisado se caracteriza porque el proceso de aprendizaje se realiza mediante un entrenamiento controlado por un agente externo que determina la respuesta que debería tener la red como salida a partir de una entrada determinada. Los ejemplos que en este caso son imágenes, se encuentran etiquetados, por lo tanto se conoce la clase a la que pertenecen.

2.3.3. Aprendizaje profundo

El aprendizaje profundo es un algoritmo de aprendizaje de máquina que usa muchas capas y que permite a los sistemas informáticos mejorar con la experiencia y los datos. Cada capa aporta a la resolución del problema.

En la Figura 10 se muestra un modelo de aprendizaje profundo, donde se muestran las capas que son usadas en este proceso. La primera capa es denominada capa visible o capa de entrada porque contiene los valores que son posibles observar, en este caso la imagen de una persona se muestra como una colección de valores de los píxeles, lo cual no es fácil entender para una computadora. Las capas ocultas se encargan de extraer características cada vez más abstractas y se llaman así porque sus valores no son dados en los datos, por lo tanto el modelo debe determinar cuáles conceptos son útiles para explicar las relaciones entre los datos observados. Dados los píxeles en la entrada, la primera capa oculta puede identificar fácilmente los bordes comparando el brillo de los píxeles vecinos, después la segunda capa oculta identifica esquinas y contornos que son colecciones de bordes, la tercera capa oculta identifica partes enteras de objetos específicos encontrando colecciones de contornos y esquinas, por último se clasifica la imagen en la categoría a la que pertenece.

La función de mapear a partir de un conjunto de píxeles a una identidad objeto es complicada, aprender o evaluar este mapeo parece insuperable si se aborda directamente. El aprendizaje profundo resuelve esta dificultad al dividir el mapeo complicado en una serie de mapeos simples anidados, cada uno descrito por una capa diferente (Goodfellow *et al.*, 2016).

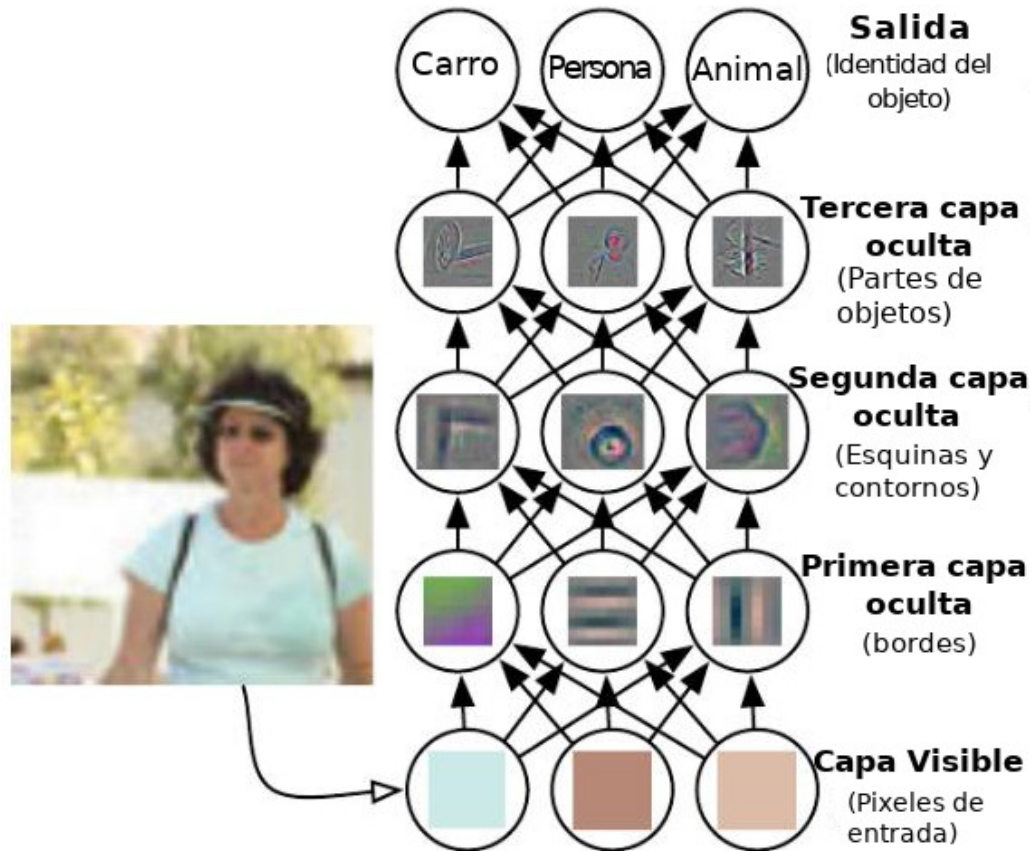


Figura 10. Modelo de aprendizaje profundo. Imagen tomada de Goodfellow *et al.* (2016).

2.3.4. Clasificación de imágenes

La clasificación de imágenes es uno de los principales problemas de visión por computadora y es una tarea que se encarga de asignar una etiqueta a una imagen de un conjunto de categorías.

La clasificación de imágenes se divide en 3 pasos:

- **Entrada:** Consiste en un conjunto de N imágenes etiquetadas con K clases diferentes, es decir el conjunto de entrenamiento.
- **Aprendizaje:** Consiste en tomar el conjunto de entrenamiento para aprender cada una de las diferentes clases, es decir aprender un modelo.
- **Evaluación:** Consiste en evaluar la calidad del clasificador entrenado con un nuevo conjunto de imágenes, de tal forma que prediga la clase a la que pertenece

una imagen de prueba. Además son comparadas las etiquetas reales de esas imágenes con las que predijo el clasificador.

En la Figura 11 se muestra una imagen de un gato y lo que percibe una computadora, en este caso son los valores de los pixeles, es decir la matriz de la imagen. Además se aprecian las posibles etiquetas a las que puede permanecer la imagen, tales como: gato, perro, sombrero y un jarrón.

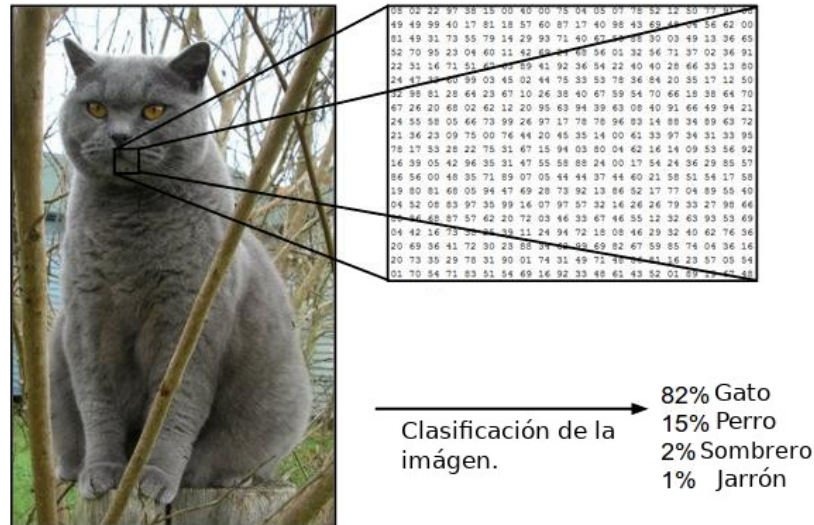


Figura 11. Clasificación de la imagen de un gato. Imagen tomada de CS231 (2016)

Para una persona es trivial clasificar un objeto, para un algoritmo de visión por computadora es más complicado, por lo tanto presenta varios desafíos, entre ellos se encuentran:

- Variación del punto de vista: Se refiere a un objeto que puede estar orientado en varias formas respecto a la cámara.
- Variación a escala: Se refiere a algunos objetos que pueden estar en diferente tamaño en el mundo real.
- Deformación: Se refiere a objetos que pueden no tener cuerpos rígidos y son deformados en diferentes formas.
- Oclusión: Es cuando algunos objetos pueden estar ocluidos, es decir algunas partes pueden no verse con claridad.

- Condiciones de iluminación: Algunos objetos pueden verse más oscuros que otros debido a los efectos de iluminación.
- Confusión con el fondo: Algunos objetos suelen mezclarse con el fondo haciendo difícil identificarlos.
- Variación intra-clase: Se refiere a que existen diferentes variedades de un solo objeto.

En la Figura 12 se muestra un ejemplo de las variaciones de imágenes.

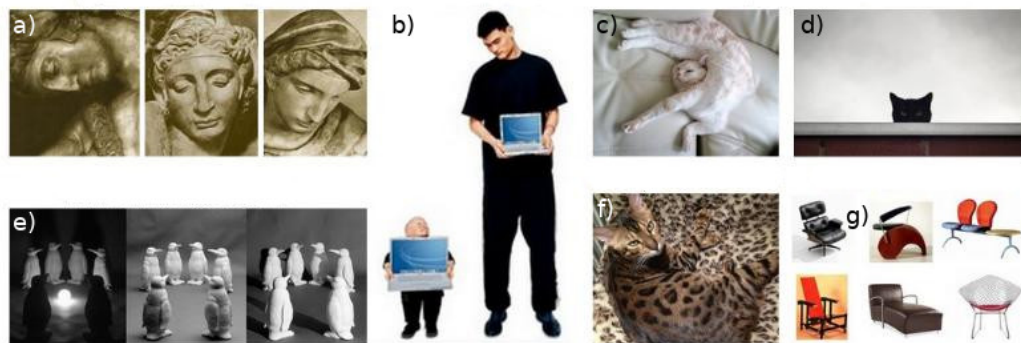


Figura 12. Variaciones de imágenes. a) Variación del punto de vista, b) Variación a escala, c) Deformación, d) Oclusión, e) Condiciones de iluminación, f) Confusión con el fondo, g) Variación intra-clase. Imagen tomada de CS231 (2016).

2.3.5. Redes neuronales

Una red neuronal es una representación inspirada en el funcionamiento del cerebro humano. Está formada por unidades de procesamiento denominadas neuronas y poseen conexiones entre ellas, además se encuentran organizadas en las siguientes capas:

- **Capa de entrada:** En esta capa se introducen los datos dependiendo de la tarea a realizar.
- **Capas ocultas:** En ellas se realiza el procesamiento de los datos y está posicionada entre la capa de entrada y la capa de salida.
- **Capa de salida:** Es llamada así porque en las tareas de clasificación, es la que se encarga de representar los valores de la clase a la que pertenece la imagen.

Las redes neuronales son modeladas como colecciones de neuronas conectadas en un grafo acíclico, por lo que, las salidas de algunas neuronas se convierten en entradas de otras. Estas redes están organizadas en distintas capas de neuronas, la más común es la completamente conectada, en la cual las neuronas entre dos capas adyacentes son conectadas y las neuronas que pertenecen a una capa no comparten conexiones. En la Figura 13 se muestran dos ejemplos de topologías de redes neuronales que usan una serie de capas completamente conectadas.

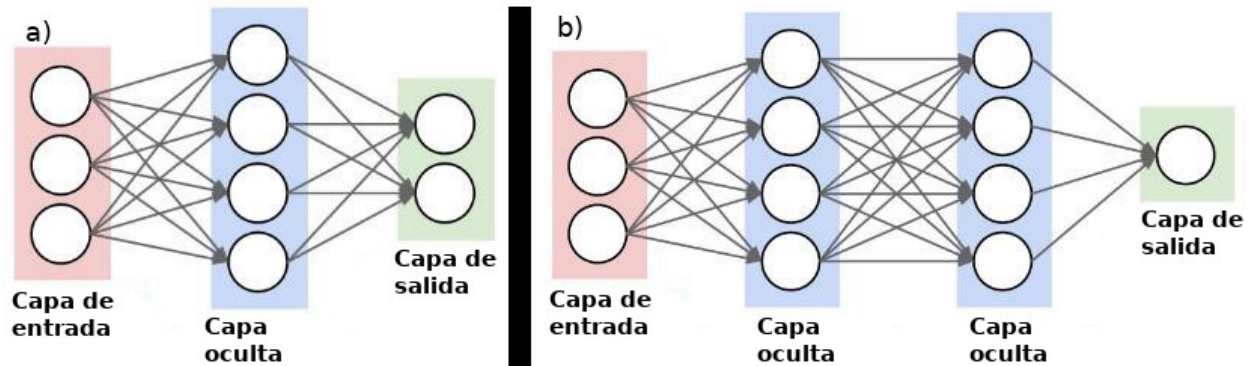


Figura 13. Topología de una red neuronal. a) Red neuronal con una capa oculta y una capa de salida, con 4 y 2 neuronas respectivamente. b) Red neuronal con 2 capas ocultas con 4 neuronas cada una y una capa de salida. Imagen tomada de Goodfellow *et al.* (2016)

2.3.5.1. Historia de las redes neuronales

1943:

El neurofisiólogo Warre McCulloh y el matemático Walter Pitts escribieron un artículo cómo las neuronas trabajan. Modelaron una red neuronal simple usando circuitos eléctricos.

1949:

Donald Hebb escribió "La organización del comportamiento", un trabajo que señala que las vías neuronales se fortalecen cada vez que se usan.

1950:

En este año fue posible simular una red neuronal, el primer paso fue por Nathalia Rochester de los laboratorios de de investigación de IBM, desafortunadamente falló.

1959:

Bernard Widrow y Marcian Hoff de Stanford desarrollaron modelos llamados ADALINE y MADALINE, los nombres se derivan por el uso de múltiples elementos lineales adaptativos. ADALINE fue diseñado para reconocer patrones binarios, si leía bits de transmisión de una línea telefónica, entonces predecía el siguiente bit. MADALINE fue la primera red neuronal aplicada a un problema del mundo real usando filtros adaptativos que eliminan el eco en las líneas telefónicas.

1962:

Bernard Widrow y Marcian Hoff desarrollaron un procedimiento de aprendizaje que examina el valor antes de que un peso se ajuste (0,1) de acuerdo a la siguiente regla:

$$\text{Cambio de peso} = \text{Valor en la línea del peso} * \frac{\text{Error}}{\text{Número de entrada}} \quad (1)$$

Esta regla se basa en la idea de que mientras un perceptron activo puede tener un gran error, se pueden ajustar los valores de peso para distribuirlos a través de la red o al menos a los perceptrones adyacentes. Aplicando la regla resulta en un error si la línea antes del peso es 0 aunque puede corregirse a sí mismo.

1972:

Kohonen y Anderson desarrollaron una red similar independiente, usando una matriz matemática para describir sus ideas pero lo que resultó fue una serie de circuitos analógicos ADALINE, donde se supone que las neuronas activan un conjunto de salidas en lugar de solo una.

1975:

Se desarrolla la primer red multicapa con aprendizaje no supervisado.

1982:

John Hopfield de Caltech presentó un artículo a la Academia Nacional de Ciencias, donde mostraba un enfoque más útil para las máquinas por medio del uso de líneas bidireccionales. Anteriormente la conexión entre las neuronas era en una dirección.

Reilly y Cooper usaron una red híbrida con múltiples capas, cada capa usaba una

estrategia diferente para resolver un problema.

Estados Unidos y Japón realizaron una conferencia sobre redes neuronales cooperativas y competitivas, Japón anunció la quinta generación en a las redes neuronales, por lo que, las noticias de los periódicos de Estados Unidos generaron preocupación por quedarse atrás en el campo. Cabe destacar las 5 generaciones que existen:

- Primera generación: interruptores y cables.
- Segunda generación: transistores.
- Tercera generación: Tecnología de estado sólido como los circuitos integrados y lenguajes de programación de alto nivel.
- Cuarta generación: Generadores de código
- Quinta generación: Inteligencia artificial

1986:

David Rumelhart, un ex miembro del departamento de psicología de Stanford, propuso ideas similares a lo que ahora se le conoce como retropropagación.

2.3.5.2. Aplicaciones de las redes neuronales

Algunas aplicaciones de las redes neuronales son:

- Clasificación de correo electrónico.
- Reconocimiento de voz.
- Reconocimiento de caracteres.
- Reconocimiento de rostros (Roberts, 2016).

2.3.5.3. Aumento de datos

Las redes profundas necesitan una gran cantidad de datos para obtener buenos resultados, si se cuenta con un conjunto pequeño de imágenes el aumento de datos es una buena técnica para lograr un buen desempeño en el entrenamiento de las imágenes. Existen diversas maneras para aumentar los datos y entre ellas se encuentran diferentes transformaciones como:

- Volteo
- Cortes aleatorios
- Fluctuación del color
- Escalamiento
- Rotación
- Traslación
- Aumento de la saturación y brillo
- Extender

En la Figura 14 se observa un ejemplo de diversas transformaciones para el aumento de datos de una base de datos que contiene imágenes de plancton.

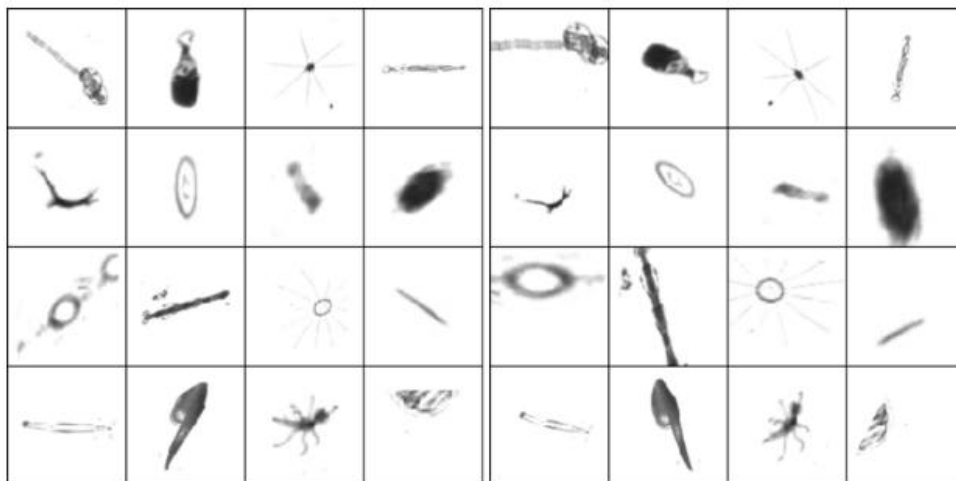


Figura 14. Ejemplo de aumento de datos. Imagen tomada de Dieleman (2015).

Otra transformación para el aumento de datos es la de Krizhevsky *et al.* (2012), donde se propone una técnica denominada análisis de componentes principales en la red Alex-Net, que tiene como objetivo alterar las intensidades de los canales RGB en las imágenes de entrenamiento, primero se realiza en el conjunto de valores de píxeles RGB en las imágenes de entrenamiento, luego para cada imagen de entrenamiento se agrega cierta cantidad a cada píxel de imagen RGB. Para el rendimiento de la clasificación esta técnica redujo significativamente el error sobre 1 % en la competencia de ImageNet 2012.

2.3.5.4. Inicialización de la red

Antes de que la red sea entrenada, los parámetros tienen que ser inicializados y existen diferentes formas para hacerlo, algunas son:

- **Inicialización en cero:** Supone que con la normalización de los datos, aproximadamente la mitad de los pesos serán positivos y la mitad serán negativos. La idea es iniciar los pesos en cero, lo que supone un buen acierto en la clasificación pero en realidad no es así, debido a que si cada neurona en la red calcula la misma salida entonces también calcularán los mismos gradientes durante la propagación hacia atrás y son sometidos a las mismas actualizaciones de parámetros, es decir no existe una asimetría si los pesos son inicializados para ser los mismos.
- **Inicialización con pequeños números aleatorios:** Es una técnica en la cual los pesos son inicializados con pequeños números aleatorios por lo que están cercanos a cero, a esto se lo conoce como ruptura de simetría. La idea es que las neuronas empiecen siendo únicas y así se calculen distintas actualizaciones, integrándose como parte de una red completa. La implementación de los pesos es: $\text{weights} = 0.0001 \times N(0,1)$, donde $N(0,1)$ es una realización de un número aleatorio con distribución gaussiana, media cero y varianza.

2.3.5.5. Funciones de activación

Una de las características de un sistema de aprendizaje profundo son las funciones de activación. Dentro de las funciones de activación se encuentra un conjunto

de 4 tipos de unidades rectificadas: rectificado lineal (ReLU), rectificado lineal con fuga (Leaky ReLU), rectificado lineal paramétrico (PReLU) y rectificado lineal aleatorio (RReLU).

- **Rectificado lineal (ReLU)**

Formalmente se define como:

$$y_i = \begin{cases} x_i & \text{si } x_i \geq 0 \\ 0 & \text{si } x_i < 0 \end{cases} \quad (2)$$

En la Figura 15 se muestra una gráfica de ReLU.

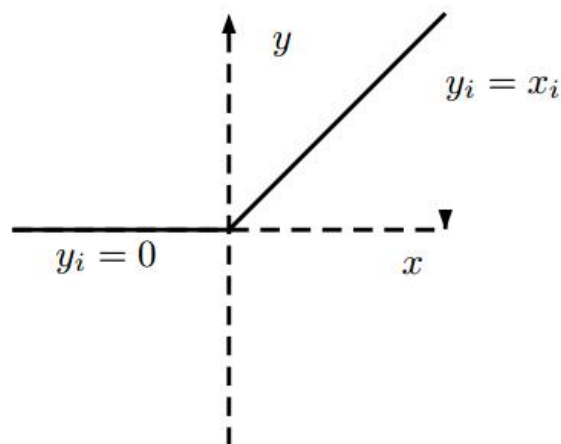


Figura 15. ReLU. Imagen tomada de Xu et al. (2015)

Comparado con sigmoide y tanh que constan de operaciones costosas como los exponenciales, ReLU puede ser implementada umbralizando una matriz de activaciones en cero. Además acelera la convergencia del gradiente estocástico comparado con las funciones anteriores. Una desventaja es que las unidades pueden ser frágiles durante el entrenamiento. Ejemplo: un gradiente grande que fluye a través de una neurona ReLU puede hacer que los pesos se actualicen, de tal forma que la neurona no se activará en ningún punto de datos.

- **Rectificado lineal con fuga (Leaky ReLU) y Rectificado lineal paramétrico (PReLU)**

Leaky ReLU es una función que trata de arreglar la desventaja de ReLU, en lugar de que la función comience en 0 cuando $x < 0$, leaky ReLU tiene un límite negativo

de 0.01 aproximadamente. Los resultados no son siempre consistentes.

PReLU fue propuesto por (He *et al.*, 2015) y se define de la misma forma que Leaky ReLU, la diferencia es que a_i es aprendida en el entrenamiento a través de la retropropagación. Leaky ReLU y PReLU se definen como:

$$y_i = \begin{cases} x_i & \text{si } x_i \geq 0 \\ \frac{x_i}{a_i} & \text{si } x_i < 0 \end{cases} \quad (3)$$

donde a_i es un parámetro ajustado en el rango $(1, +\infty)$ en Leaky ReLU. En la Figura 16 se muestra un gráfico de las funciones.

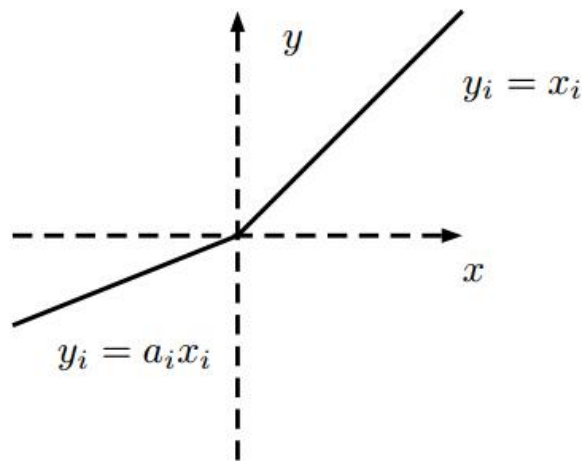


Figura 16. Leaky ReLU y PReLU. Imagen tomada de Xu *et al.* (2015).

■ Rectificado lineal aleatorio (RReLU)

Es la versión aleatoria de Leaky ReLU, durante el entrenamiento a_{ji} es un número aleatorio que pertenece a una distribución uniforme $U(l, u)$, se define como:

$$y_{ji} = \begin{cases} x_{ji} & \text{si } x_{ji} \geq 0 \\ a_{ji}x_{ji} & \text{si } x_{ji} < 0 \end{cases} \quad (4)$$

donde $a_{ji} \sim U(l, u)$, $l < u$ y $l, u \in [0, 1)$.

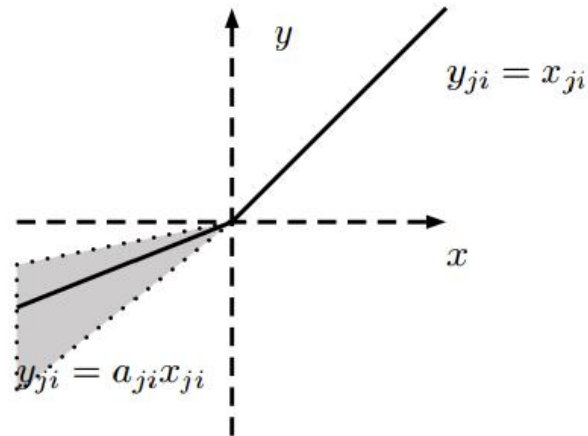


Figura 17. RReLU. Imagen tomada de Xu et al. (2015).

Sigmoide

Toma un número real y lo posiciona en un rango entre 0 y 1, los números negativos grandes se vuelven 0 y los números positivos grandes se vuelven 1. Tiene la forma matemática

$$\sigma(x) = 1/(1 + e^{-x}) \quad (5)$$

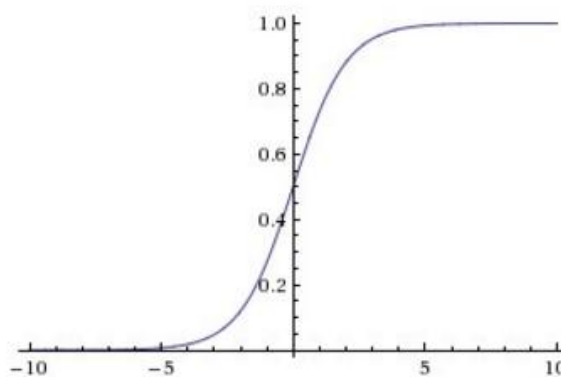


Figura 18. Función sigmoide. Imagen tomada de Xu et al. (2015)

Tanh

Toma un número real y lo posiciona en un rango entre -1 y 1. En la práctica esta función es preferible que la función sigmoide.

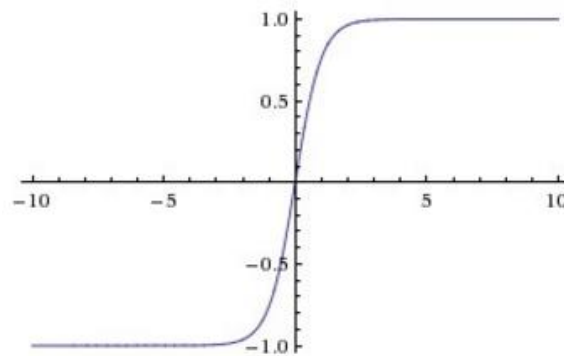


Figura 19. Función tanh. Imagen tomada de Xu *et al.* (2015)

En la práctica se observa que Leaky ReLU alcanza una mejor precisión, ReLU paramétrico es fácil de que se sobreajuste en conjuntos de datos pequeños, es decir su error de entrenamiento es el más pequeño, mientras que el error en la prueba no es satisfactorio. ReLU aleatorio es mejor que otras funciones de activación en la base de datos NDSB (Kaggle, 2015) debido a que supera el sobreajuste, ya que la base de datos contiene menos imágenes de entrenamiento que en CIFAR-10 y CIFAR-100 (Krizhevsky, 2009).

2.3.5.6. Regularizaciones

Una regularización es cualquier proceso de modificación que se realiza a un algoritmo de aprendizaje para reducir su error de generalización, pero no su error de entrenamiento.

■ Regularización L2

La regularización L2 también es conocida como decaimiento de peso y es la más común, puede ser implementada penalizando la magnitud cuadrada de todos los parámetros directamente en el objetivo, es decir por cada peso ω en la red se agrega el término $\frac{1}{2}\lambda\omega^2$ al objetivo, donde λ es la fuerza de regularización.

■ Regularización L1

Tiene la propiedad de hacer que los vectores de pesos se vuelvan escasos durante la optimización (cercano a cero). En esta regularización por cada peso ω se

agrega el término $\lambda |\omega|$ al objetivo, se pueden combinar las regularizaciones L1 y L2, es decir : $\lambda_1 |\omega| + \lambda_2 \omega^2$ (Regularización de red elástica).

■ Dropout

Dropout (Srivastava *et al.*, 2014) es una técnica de regularización que previene el sobreajuste y ayuda a mejorar otras regularizaciones. Esta técnica consiste en desconectar unidades aleatoriamente entre la capa visible o de entrada y la capa oculta. Desconectar una unidad significa que temporalmente será removida de la red con sus conexiones de entrada y de salida.

En la Figura 20 se muestra un ejemplo de dropout donde a) representa una red neuronal estándar con dos capas ocultas y b) representa la misma red aplicando dropout.

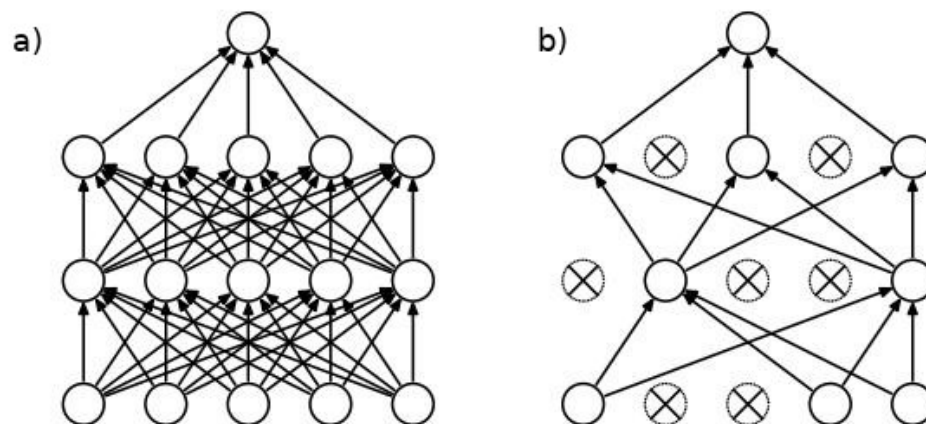


Figura 20. Ejemplo de dropout. Imagen tomada de Srivastava *et al.* (2014)

En el caso más simple, cada unidad se conserva con una probabilidad fija p independiente de otras unidades, donde p puede elegirse utilizando una validación establecida o se puede configurar en 0.5, que se considera un valor óptimo. Para las unidades de entrada la probabilidad óptima de retención es más cercana a 1 que de 0.5.

En la Figura 21 a) se muestra una unidad durante el entrenamiento que tiene una probabilidad p y está conectada con otras unidades de la siguiente capa con pesos representados como w , b) Una unidad durante la prueba donde los pesos se multiplican por la probabilidad, además la salida es la misma que la esperada durante el entrenamiento.

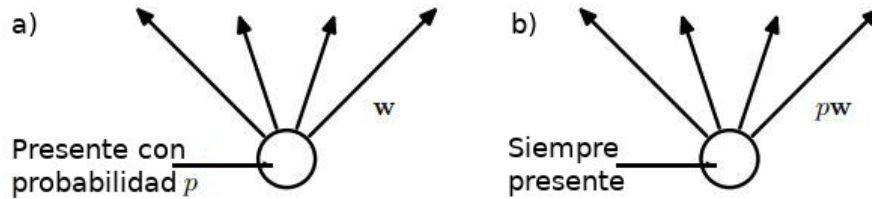


Figura 21. Dropout. a) durante el entrenamiento, b) durante la prueba. Imagen tomada de Srivastava *et al.* (2014).

2.3.6. Redes Neuronales Convolucionales

Las redes neuronales convolucionales son un tipo de redes neuronales para el procesamiento de datos, en especial de imágenes. Su nombre proviene de una operación matemática llamada convolución. Convolución es un tipo de operación lineal especializada similar a la usada por las redes convolucionales en al menos una de sus capas.

2.3.6.1. La operación de convolución

Se define por la siguiente fórmula y se denota con un asterisco:

$$s(t) = (x * w)(t) \quad (6)$$

La función x se le conoce como la entrada, la función w es el kernel y la salida se le denomina mapeo de características.

Convolución discreta:

$$s(t) = (x * w)(t) = \sum_{\alpha=-\infty}^{\infty} x(\alpha)w(t-\alpha) \quad (7)$$

En aplicaciones de aprendizaje de máquina, la entrada es usualmente un arreglo multidimensional de datos o tensores y el kernel es un arreglo de parámetros que son adaptados por el algoritmo de aprendizaje. En la Figura 22 se muestra un ejemplo de convolución en 2 dimensiones.

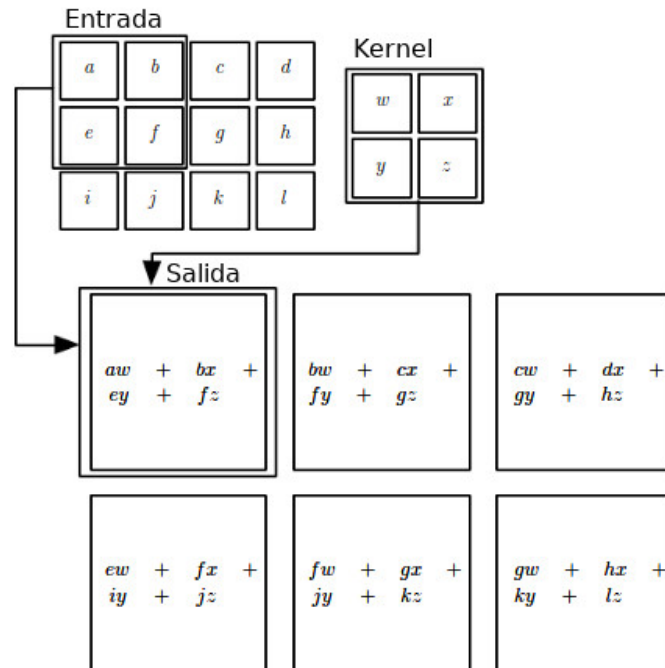


Figura 22. Convolución en 2D. Imagen tomada de Goodfellow *et al.* (2016).

2.3.6.2. Capas de las redes neuronales convolucionales

2.3.6.3. Capa Convolutiva

En lugar de utilizar la multiplicación de matrices que es aplicada generalmente, se usa la convolución. La operación de convolución recibe como entrada la imagen y luego aplica sobre ella un filtro o kernel que regresa un mapa de las características de la imagen original, de esta forma el tamaño de los parámetros es reducido. La convolución aprovecha tres ideas que ayudan a mejorar cualquier sistema de aprendizaje de máquina y son:

- **Interacciones dispersas:** Al aplicar un filtro de menor tamaño sobre la entrada original, se reduce de manera significativa la cantidad de parámetros y cálculos.
- **Parámetros compartidos:** Son los que hacen referencia a compartir los parámetros entre los distintos tipos de filtros, esto ayuda a la eficacia del sistema.
- **Representación equivariante:** Indican que si las entradas cambian entonces las salidas cambiarán de forma similar.

Por otra parte, la convolución proporciona un medio para trabajar con entradas de tamaño variable.

2.3.6.4. Capa de Pooling

La capa de pooling está situada después de la capa convolucional, su objetivo es reducir las dimensiones espaciales del volumen de entrada para la siguiente capa convolucional sin afectar la profundidad. La operación que se realiza en esta capa también es llamada reducción de muestreo debido a la reducción de tamaño y a la pérdida de información resulta favorable por la disminución en el tamaño que lleva a una menor sobrecarga en el cálculo en las próximas capas, además de que puede disminuir el sobreajuste.

Una capa en una red convolucional consiste en tres etapas, en la primera etapa representa convoluciones en paralelo para producir un conjunto de activaciones lineales, en la segunda etapa cada activación lineal se ejecuta a través de una función de activación no lineal, en la tercera etapa se usa una función de pooling para modificar la salida de la capa.

En todos los casos la capa de pooling ayuda a hacer una representación aproximada invariante a pequeñas traslaciones de la entrada. Invariante a traslaciones significa que si la entrada se traslada un poco, los valores de las salidas agrupadas no cambian.

En la Figura 23 se presentan los componentes de una capa en una red convolucional.

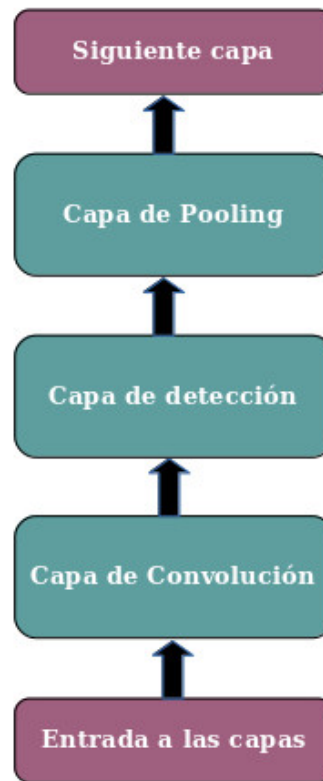


Figura 23. Componentes de una capa de red convolucional.

2.3.6.5. Capa completamente conectada

Se usa al final de las capas convolucional y de pooling, en esta capa cada pixel es considerado como una neurona separada al igual que en una red neuronal. Esta capa se encarga de clasificar y tendrá un número de neuronas que corresponde al número de clases a predecir.

Capítulo 3. Metodología

3.1. Introducción

En este capítulo se describe la metodología usada en la investigación para la clasificación de imágenes de fondo del ojo. En este trabajo se utilizaron dos bases de datos, Messidor y Messidor 2, con ellas se realizaron varios experimentos. Las imágenes originales de cada base de datos vienen en diferentes tamaños, por lo que se les aplicó un preprocesamiento, con el objetivo de manejarlas fácilmente.

Los experimentos fueron realizados en el sistema operativo Ubuntu versión 16.04 usando el lenguaje de programación Python 3.5 y las pruebas se ejecutaron en un servidor cuyas características son:

- Procesador Intel(R) Xeon(R) CPU E7-4809 v3@2.00GHz.
- Memoria RAM 1.5 TB.
- 64 núcleos.

Para realizar los experimentos se siguió la siguiente metodología representada en la Figura 24.

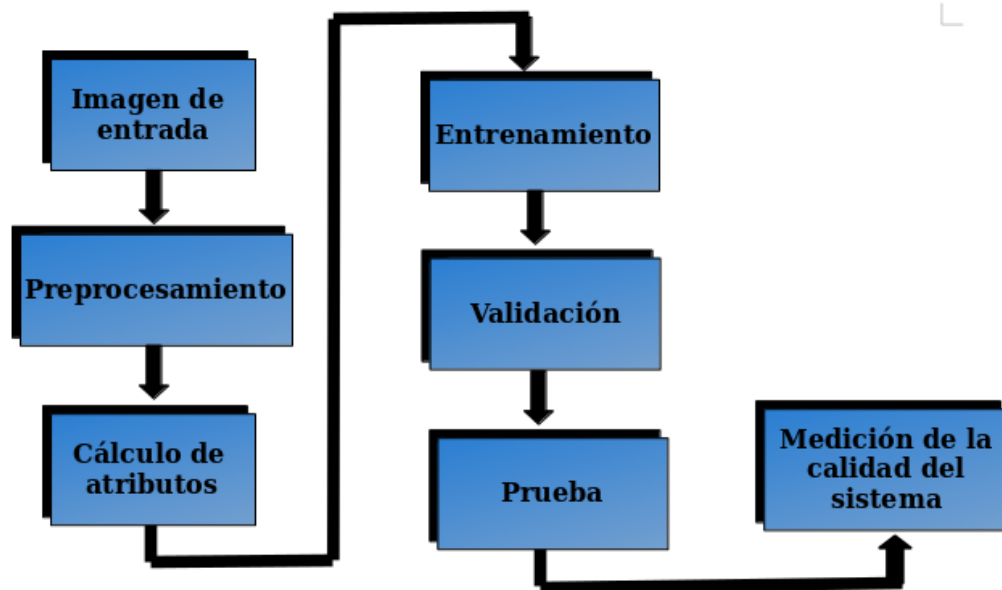


Figura 24. Metodología.

3.2. Bases de datos

3.2.1. Messidor

La base de datos Messidor (Decencière *et al.*, 2014) es proporcionada con el objetivo de facilitar los estudios en diagnósticos asistidos por computadora de la retinopatía diabética. La base de datos consta de 1200 imágenes de fondo del ojo a color obtenidas por tres departamentos de oftalmología mediante una cámara de fondo del ojo no midriática Topcon TRC NW6 con un campo de visión de 45 grados. 800 imágenes fueron adquiridas con la pupila dilatada y 400 imágenes sin dilatación.

Las imágenes fueron capturadas usando 8 bits por color y con los siguientes tamaños:

- 1440 x 960 pixeles.
- 2240 x 1488 pixeles.
- 2304 x 1536 pixeles.

Las imágenes están empaquetadas en 3 conjuntos, una por cada departamento de oftalmología y se encuentran en el formato TIFF.

En esta base de datos se clasifica a la retinopatía diabética en 4 clases, cada una con un número asignado del 0 al 3. La clase 0 representa a las imágenes que no presentan signos de RD, es decir es un ojo sano, la clase 1 representa a las imágenes con RD leve, la clase 2 representa a las imágenes con RD moderada, finalmente la clase 3 representa a las imágenes con RD severa. En la tabla 1 se muestra la cantidad de imágenes que hay en cada clase.

En la tabla 2 se describe como es medido el grado de la retinopatía diabética en la base de datos Messidor de acuerdo al número de lesiones presentadas.

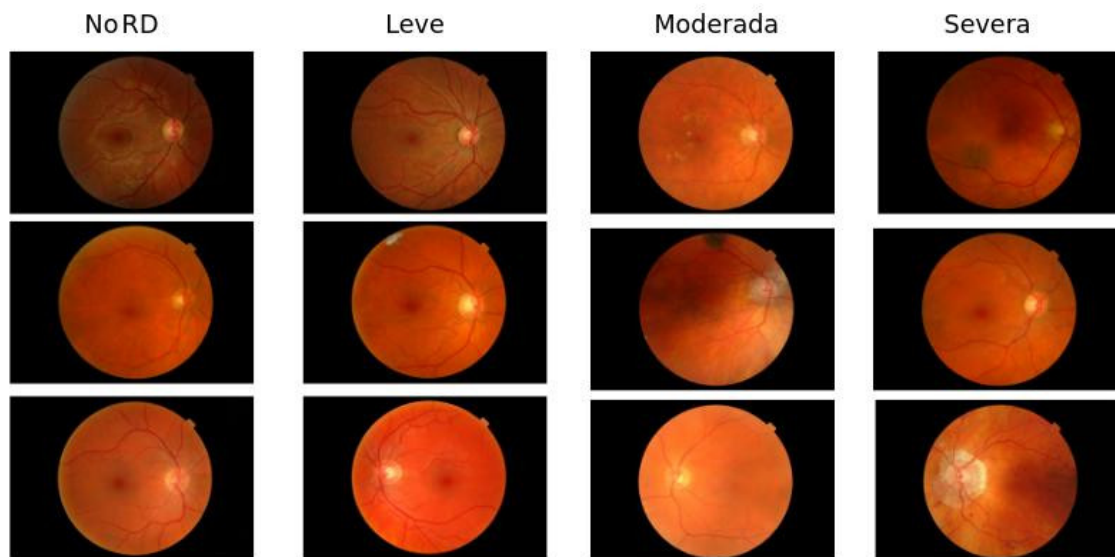
En la Figura 25 se observan algunos ejemplos de la base de datos de Messidor de acuerdo al grado de la retinopatía diabética.

Tabla 1. Imágenes por clase de Messidor.

Clase	Número de imágenes
0	547
1	153
2	246
3	254
Total	1200

Tabla 2. Grados de la retinopatía diabética en Messidor.

Clase	Nombre	Número de microaneurismas	Número de hemorragias
0	No RD	0	0
1	Leve	menor o igual a 5	0
2	Moderada	más de 5, menos de 15	menor a 5
3	Severa	mayor o igual a 15	más de 5

**Figura 25.** Ejemplos de imágenes de acuerdo al grado de la Retinopatía diabética. Imágenes tomadas de Decencière et al. (2014)

3.2.2. Messidor 2

Para hacer una extensión de la base de datos Messidor, pacientes con diabetes fueron reclutados en el departamento de oftalmología de Hospital Universitario de Brest en Francia entre el 16 de Octubre de 2009 y el 6 de Septiembre de 2010. En ese

departamento fueron adquiridos 345 pares de imágenes sin dilatación de la pupila, mediante una cámara de fondo del ojo no midriática Topcon TRC NW6 con un campo de visión de 45 grados. Cabe mencionar que la base de datos de Messidor cuenta con algunas imágenes que vienen en pares. La base de datos Messidor 2 es una colección de imágenes para el diagnóstico de la retinopatía diabética (RD), está dividida en dos clases y contiene las imágenes de Messidor que están en pares, es decir 1058 imágenes y 690 imágenes adquiridas en el Hospital Universitario de Brest haciendo un total de 1748 imágenes.

Las imágenes fueron capturadas usando 8 bits por color y con los siguientes tamaños:

- 1440 x 960 pixeles.
- 2240 x 1488 pixeles.
- 2304 x 1536 pixeles.

En la tabla 3 se muestra la cantidad de imágenes que tiene cada clase en la base de datos de Messidor 2, se observa que ahora se consideran dos clases: RD y no RD.

Tabla 3. Imágenes por clase de Messidor 2

Clase	Número de imágenes
0	1368
1	380
Total	1748

En la siguiente tabla se describe como es medido el grado de la retinopatía diabética en la base de datos Messidor 2.

Tabla 4. Grado de la retinopatía diabética en Messidor 2.

Clase	Nombre	Número de microaneurismas	Número de hemorragias
0	No RD	0	0
1	RD	más de 1	0

En la Figura 26 se observan algunos ejemplos de la base de datos de Messidor 2 de acuerdo al grado de la retinopatía diabética.

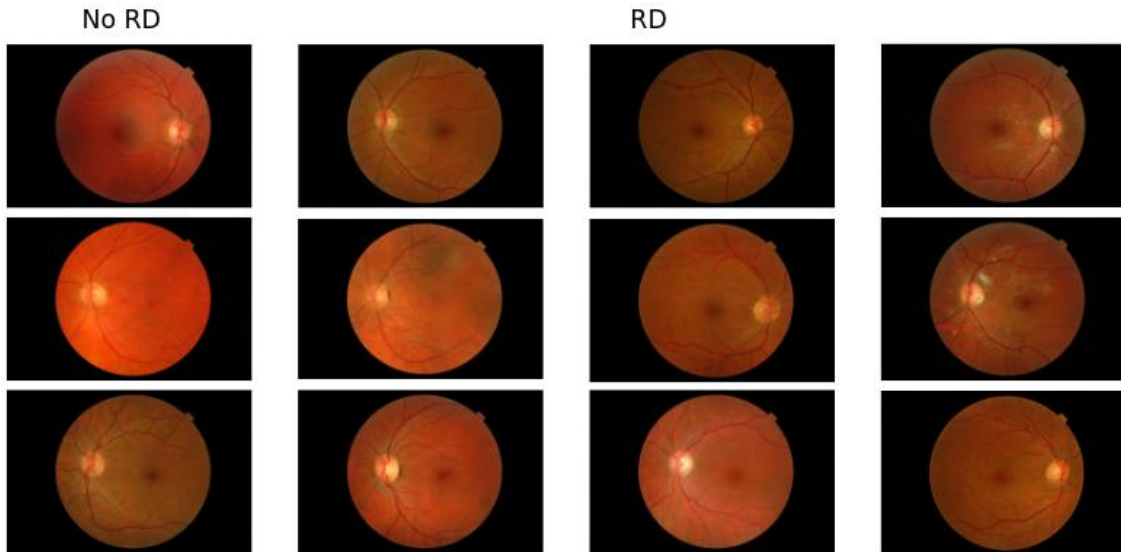


Figura 26. Ejemplos de imágenes de acuerdo al grado de la Retinopatía diabética. Imágenes tomadas de Decencière *et al.* (2014).

3.3. Preprocesamiento de imágenes.

Para definir la entrada a la red convolucional, las imágenes de cada base de datos fueron cortadas en diferentes tamaños, debido a que son demasiado grandes y rectangulares.

Se definieron los siguientes tamaños para el corte:

- 128 x 128 píxeles.
- 256 x 256 píxeles.
- 512 x 512 píxeles.

Para realizar este preprocesamiento primero se seleccionó el área más pequeña que tuviera el ojo entero y después se cortó. Este procedimiento se hizo utilizando el lenguaje de programación Python.

En la Figura 27 se observa el resultado del procesamiento en la base de datos Messidor, donde una imagen es la original y la otra es la cortada.

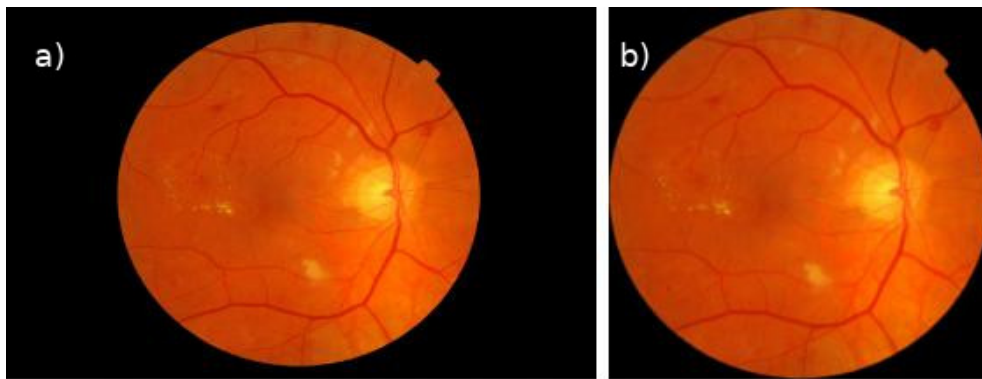


Figura 27. a) Imagen original, b) Imagen cortada.

En la Figura 28 se observa el resultado del procesamiento para la base de datos Messidor 2.

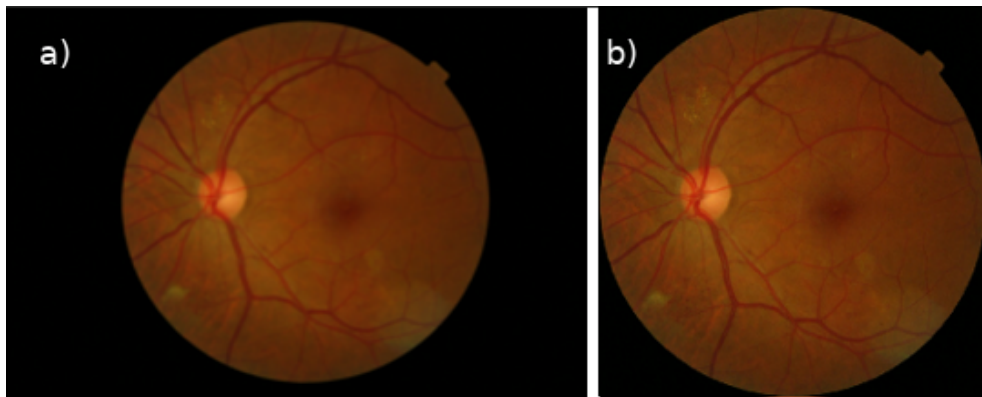


Figura 28. a) Imagen original, b) Imagen cortada.

3.4. Otro tipo de preprocesamiento

- **Substracción de la media de las imágenes:** Es una técnica que favorece a imágenes donde el color o la distribución de la intensidad no es consistente.
- **Normalización por canal:** En esta técnica sustrae la media y se divide por la desviación estándar, además es útil cuando no se puede realizar la técnica anterior.
- **Substracción de la media por canal:** Es una buena técnica para entradas de tamaño variable donde se usa la primera técnica y no se quiere hacer demasiadas suposiciones sobre la distribución.
- **Blanqueado:** Consiste en convertir la distribución en una normal.

- **Reducción de la dimensionalidad:** Consiste en transformar los datos en un espacio comprimido con menos dimensiones, donde se controla la pérdida y uso de la entrada a la red. No es una técnica común aplicada al aprendizaje profundo.

3.5. División de las bases de datos

De manera general las bases de datos se pueden dividir de la siguiente forma:

Conjunto de entrenamiento:

- Permite obtener los pesos de una red dada.
- Ajuste de parámetros.

Conjunto de validación:

- Permite comparar el desempeño de distintas arquitecturas de red.
- Ajuste de hiperparámetros.

Conjunto de prueba

- Permite estimar el desempeño de la red.

Las bases de datos fueron triplicadas. En la Figura 29 se observan 3 niveles de carpetas, el primer nivel a) se refiere a las imágenes originales que tiene cada base de datos, b) representa el triplicado de la base de datos, es decir se les aplicó el corte en los tamaños de 128 x 128, 256 x 246 y 512 x 512 pixeles, por último c) representa la división de cada base de datos en imágenes de entrenamiento y prueba.

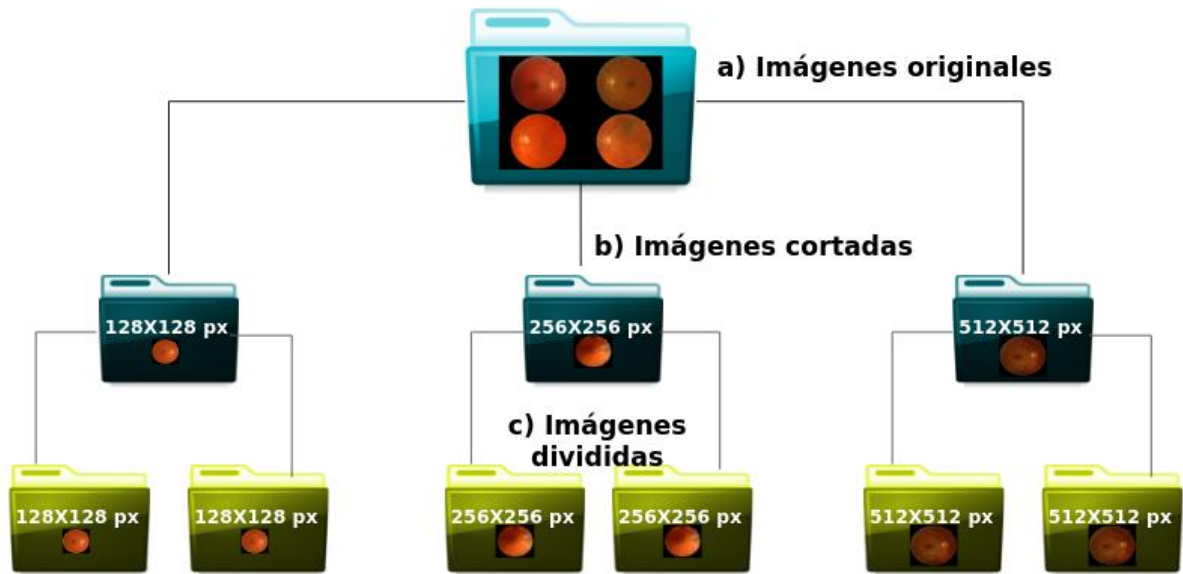


Figura 29. División de las bases de datos. a) Imágenes originales, b) Imágenes cortadas, c) Imágenes divididas.

3.6. Cálculo de atributos

Además del recorte de las imágenes se obtuvieron los siguientes atributos que se utilizan durante el entrenamiento. El primer atributo que se considera es el promedio, El siguiente atributo que se considera es la desviación estándar muestral que se define de la siguiente manera:

- **Desviación estándar:** La desviación estándar es la raíz cuadrada de la varianza. Es una de las medidas de dispersión, una medida que indica como los valores individuales pueden diferir de la media.
- **Error estándar:** El error estándar estimado de una media \bar{x} basado en n puntos independientes x_1, x_2, \dots, x_n , está dado por la siguiente fórmula:

$$\sqrt{\frac{s^2}{n}} \quad (8)$$

donde

$$s^2 = \sum_{i=1}^n (x_i - \bar{x})^2 / (n - 1) \quad (9)$$

- **Balanceo de pesos:** El balanceo de pesos es una manera de balancear la canti-

dad de imágenes que se tienen por clase dentro de un conjunto de datos (Bradley Efron y Tibshirani, 1979). Se define por la siguiente fórmula:

$$\text{Submuestreo de pesos} = \frac{1}{\text{Selección de probabilidad}} \quad (10)$$

donde la selección de probabilidad es el porcentaje de las imágenes de entrenamiento de cada clase.

En las tablas 5 y 6 se muestran los atributos calculados para el entrenamiento de la red neuronal convolucional de Messidor y Messidor 2 respectivamente. Se muestran los valores de la media, la desviación estándar, los eigenvalores y los eigenvectores de la matriz de covarianza de todas las imágenes.

Tabla 5. Atributos de Messidor

Media	134.5142742	63.843412385	21.1672838
Desviación estándar	78.58043814	39.9046363	16.62395592
Eigenvalores	1.64308206	0.64123404	0.23128457
Eigenvectores	-0.55815387	0.59118231	0.58190607
	-0.59845397	0.19899284	-0.77574304
	-0.57471242	-0.78117792	0.24275573

Tabla 6. Atributos de Messidor 2

Media	123.95193962	58.26946874	20.34519999
Desviación estándar	73.62095118	37.02279862	15.66388521
Eigenvalores	1.62162386	0.61635296	0.23422837
Eigenvectores	-0.55852078	0.62181193	0.54224674
	-0.59447729	0.15510765	-0.78578444
	-0.57827222	-0.76194457	0.28239469

3.7. Entrenamiento

3.7.1. Aumento de datos

La forma más fácil y común de reducir el sobreajuste es aumentar los datos artificialmente, usando transformaciones que preservan la etiqueta. Las imágenes no tienen que ser guardadas y son generadas en código de Python. Para el entrenamiento de las bases de datos se hicieron las siguientes transformaciones:

- Rotación.
- Traslación.
- Escala.
- Extendido.

También fue aplicada la siguiente técnica:

Aumento de color: Consiste en alterar las intensidades de los canales RGB de cada imagen de entrenamiento Krizhevsky (2009). A cada imagen se le agregan múltiplos de los componentes principales con magnitudes proporcionales a los eigenvalores correspondientes por una variable aleatoria extraída de una gaussiana con media cero y desviación estándar. Para cada pixel de la imagen $I_{xy} = [IR_{xy}, IG_{xy}, IB_{xy}]^T$ se le agrega el siguiente vector:

$$[\rho_1, \rho_2, \rho_3][\alpha_1\lambda_1, \alpha_2\lambda_2, \alpha_3\lambda_3]^T \quad (11)$$

donde ρ_i y λ_i son eigenvectores y eigenvalores de una matriz de covarianza con valores de los pixeles. Cada α_i es calculado una sola vez para todos los pixeles una imagen de entrenamiento hasta que la imagen se usa para el entrenamiento nuevamente.

3.7.2. Validación cruzada

En algunos casos donde el tamaño de los datos es pequeño se utiliza una técnica denominada validación cruzada, la cual permite ajustar los hiperparámetros. La idea es que en lugar de tomar arbitrariamente los primeros mil datos para ser puestos en el conjunto de validación y el resto en el conjunto de prueba, se puede obtener menor varianza en el error, estimando el valor de k para saber que tan bien se ajusta iterando en diferentes conjuntos de validación y promediando el rendimiento a través de ellos.

En la Figura 30 se muestra un ejemplo de la división de los datos donde el conjunto de entrenamiento se divide en 5, se usan 4 para el entrenamiento y uno para la validación donde se realizan las iteraciones donde se evalúa el rendimiento y finalmente se promedia a través de los conjuntos restantes.

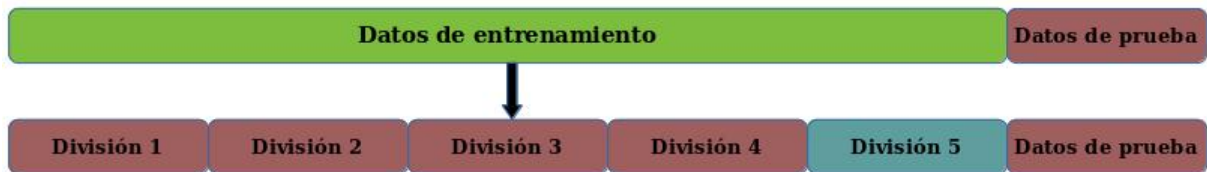


Figura 30. Validación cruzada.

En la Figura 31 se observa un ejemplo de validación cruzada, donde el conjunto de entrenamiento es dividido en 5 subconjuntos, para cada valor de k se entrena sobre cuatro conjuntos y se evalúa en el quinto. Por cada k se reciben cuatro precisiones sobre la validación, donde cada una es representada por un punto. La línea es dibujada tomando en cuenta el promedio de los resultados para cada k y la barra de error indica la desviación estándar. En este caso un buen resultado es 7 para el valor de k , por lo tanto si se usan más de 5 subconjuntos se espera que haya una menor cantidad de ruido.

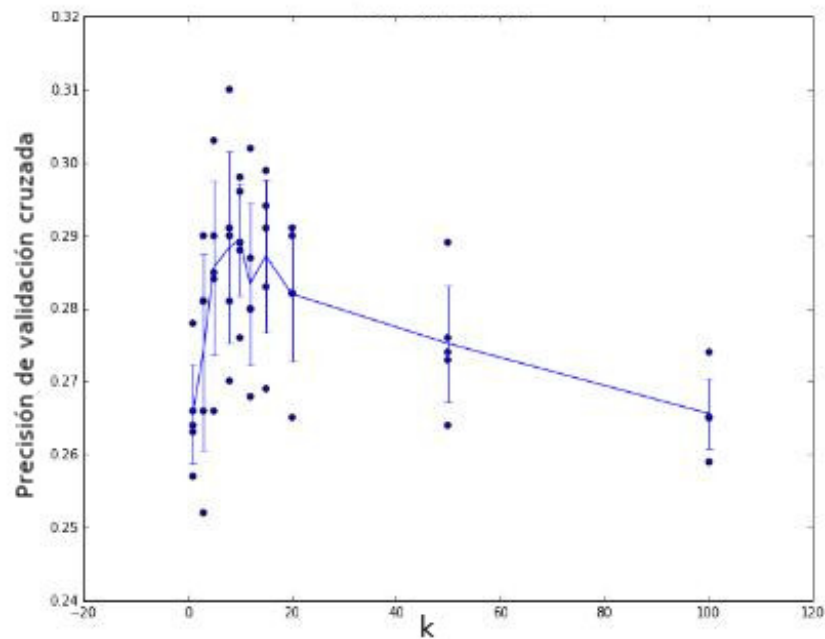


Figura 31. Ejemplo de validación cruzada.

3.7.3. Diseño de la red neuronal convolucional

En la tabla 7 se muestra la estructura de la red neuronal convolucional para el entrenamiento. La primera columna representa las capas de la red, en la segunda

columna se describe el tipo de capa, luego el número de unidades y los parámetros del filtro usado como extractor de características.

Tabla 7. Estructura de la red

	Unidades	Filtro	Stride
1	Entrada		
2	Convolutacional	32	5 2
3	Convolutacional	32	3
4	MaxPool		3 2
5	Convolutacional	64	5 2
6	Convolutacional	64	3
7	Convolutacional	64	3
8	MaxPool		3 2
9	Convolutacional	128	3
10	Convolutacional	128	3
11	Convolutacional	128	3
12	MaxPool		3 2
13	Convolutacional	256	3
14	Convolutacional	256	3
15	Convolutacional	256	3
16	MaxPool		3 2
17	Convolutacional	512	3
18	Convolutacional	512	3
19	Dropout		
20	Dense	1024	
21	Maxout	512	

Filtros: El filtro se mueve en los ejes horizontal y verticales de la imagen, lo que confiere equivalencia de traducción en ambas direcciones, el objetivo es reducir los parámetros y mejorar la precisión.

Tasa de aprendizaje: Una buena recomendación es dividir los gradientes en pequeños conjuntos. Para obtener una buena tasa de aprendizaje es bueno usar el conjunto de validación. La tasa de aprendizaje que se uso se describe en la tabla 8.

Tabla 8. Tasa de aprendizaje

Época	Tasa de aprendizaje
0	0.003
100	0.0003
200	0.00003

3.7.4. Cálculo del porcentaje de las imágenes

Se realizó el cálculo del porcentaje de las imágenes de entrenamiento para realizar un balanceo de pesos de las imágenes. Una clase que tiene un número menor de imágenes tendrá mayor peso para ser tomada. En la tablas 9 y 10 se muestra el número de imágenes por cada clase y el porcentaje de las bases de datos Messidor y Messidor 2 respectivamente.

Tabla 9. Porcentaje de las imágenes de Messidor

Clase	Número de imágenes	Porcentaje de las imágenes
0	547	45.58 %
1	153	12.75 %
2	246	20.5 %
3	254	21.16 %

Tabla 10. Porcentaje de las imágenes de Messidor 2

Clase	Número de imágenes	Porcentaje de las imágenes
0	1368	78.26 %
1	380	21.73 %

3.7.5. Cálculo del balanceo de pesos

Para realizar el cálculo del balanceo de pesos la base de datos se dividió en dos conjuntos; imágenes de entrenamiento e imágenes de prueba. En las tablas 9 y 10 se muestran los porcentajes totales y en las tablas 11 y 12 se muestra el balanceo de pesos por cada clase de cada base de datos utilizada, donde el porcentaje de las imágenes se refiere a total de las imágenes de entrenamiento y no a las totales de cada base de datos.

Tabla 11. Balanceo de pesos en Messidor

Clase	Número de imágenes	Porcentaje de las imágenes	Balanceo de pesos
0	517	47.87 %	2.0889
1	123	11.38 %	8.78
2	216	20 %	5
3	224	20.74 %	4.8216
Total	1080		

Tabla 12. Balanceo de pesos en Messidor 2

Clase	Número de imágenes	Porcentaje de las imágenes	Balanceo de pesos
0	1281	81.38 %	1.2288
1	293	18.61 %	5.37
Total	1574		

3.7.6. Secuencia de entrenamiento

En la Figura 32 se muestra la secuencia de entrenamiento de las imágenes. Primero se tomó el conjunto de imágenes de entrenamiento de las imágenes de 128 x 128 píxeles, se guardaron los pesos y se usaron para inicializar los pesos de las imágenes de 256 x 256 píxeles, se guardaron los pesos y se usaron para el entrenamiento de las imágenes de 512 x 512 píxeles.



Figura 32. Secuencia de entrenamiento para las bases de datos.

3.8. Medición de la calidad del sistema

3.8.1. Matriz de confusión

La matriz de confusión es una matriz cuadrada que se usa para la presentación y análisis del resultado de una clasificación, su principal objetivo es hacer notar la confusión entre las diferentes clases. Gracias a ella se conoce el porcentaje correcto de clasificación y la fiabilidad de cada clase. En la diagonal principal se muestran los datos que fueron correctamente clasificados.

Tabla 13. Ejemplo de matriz de confusión

		Valor real	
		Positivo	Negativo
Predicción	Positivo	Verdaderos Positivos (VP)	Falsos Positivos (FP)
	Negativo	Falsos Negativos (FN)	Verdaderos Negativos (VN)

La matriz de confusión tiene cuatro parámetros que son:

Verdaderos Positivos (VP): Indica la proporción de buena predicción para los objetos positivos.

Verdaderos Negativos (VN): Indica la proporción de buena predicción para los objetos negativos.

Falsos Positivos (FP): Indica la proporción de objetos negativos que se clasifican

como objetos positivos, también se le conoce como error de tipo I.

Falsos Negativos (FN): Indica la proporción de objetos positivos que se clasifican como objetos negativos, también se le conoce como error de tipo II.

Se observa que los primeros dos parámetros son predicciones correctas, mientras que los últimos dos son predicciones erróneas. Con la obtención de los parámetros anteriores se pueden calcular varias medidas de desempeño, algunas de ellas son:

Precisión o valor predictivo positivo: Es una medida de proporción del número total de predicciones correctas respecto al total, se determina usando la siguiente fórmula:

$$VP+ = \frac{VP}{VP + VF} \quad (12)$$

Valor predictivo negativo: Es una medida de proporción de predicciones negativas correctas, es decir, mide la calidad de la respuesta negativa del modelo de clasificación, se determina por la siguiente fórmula.

$$VP- = \frac{VN}{VN + FN} \quad (13)$$

Sensibilidad: Medida de proporción que cuantifica las predicciones positivas correctas respecto a las instancias realmente positivas, mide la capacidad de detección y se determina por la siguiente ecuación:

$$Sensibilidad = \frac{VP}{VP + FN} \quad (14)$$

Especificidad: Es una medida de proporción que indica la fiabilidad de que una predicción negativa realmente lo sea, mide la capacidad de discriminación y se determina por la siguiente fórmula:

$$Especificidad = \frac{VN}{FP + VN} \quad (15)$$

Tasa de error: Indica el número de predicciones que fueron clasificados errónea-

mente. La fórmula para calcularlo es:

$$Tasadeerror : \frac{FN + FP}{VP + VN + FN + FP} \quad (16)$$

Exactitud: Indica el porcentaje de predicciones correctas entre el total de predicciones, es decir da una proximidad entre el resultado y la clasificación exacta. Su fórmula es:

$$Exactitud = \frac{\text{predicciones correctas}}{\text{total de predicciones}} = \frac{VP + VN}{VP + FP + VN + FN} \quad (17)$$

Tasa de falsos positivos: Es una medida que cuantifica el error de tipo I. Su fórmula es:

$$Tasade falsos positivos = \frac{FP}{FP + VN} = 1 - \text{Especificidad} \quad (18)$$

Tasa de falsos negativos: Es una medida que cuantifica el error del tipo II. Su fórmula es:

$$Tasade falsos negativos = \frac{FN}{VP + FN} = 1 - \text{Sensibilidad} \quad (19)$$

Medida F: Es una media armónica de la precisión y la sensibilidad, se determina por la siguiente fórmula:

$$F_1 = \frac{2VP}{2VP + FP + FN} \quad (20)$$

3.8.2. Coeficiente Kappa

El coeficiente Kappa es una forma de medir la exactitud más precisa de una clasificación que la matriz de confusión, se define como una medida de la diferencia entre la exactitud lograda en la clasificación con un clasificador automático y la oportunidad de lograr una clasificación correcta con un clasificador aleatorio. Adquiere valores entre 0 y 1, el 0 representa falta de concordancia y el 1 una concordancia total. El índice kappa se mide por la siguiente fórmula:

$$\hat{K} = \frac{m \sum_{i=1}^{\gamma} x_{i,i} - \sum_{i=1}^{\gamma} (x_{i+} * x_{+i})}{m^2 - \sum_{i=1}^{\gamma} (x_{i+} * x_{+i})} \quad (21)$$

donde:

γ = Número de filas en la matriz

$x_{i,i}$ = Número de observaciones en el elemento de la fila i y la columna i , es decir, sobre la diagonal mayor.

$x_{i,+}$ = Total de observaciones en la fila i .

$x_{+,i}$ = Total de observaciones en la columna i .

m = Número de observaciones incluidas en la matriz.

Los valores del índice menores a 0.20 se considera una fuerza de concordancia pobre, de 0.21 a 0.40 se considera débil, de 0.41 a 0.60 se considera moderada, de 0.61 a 0.80 se considera buena y finalmente de 0.81 a 1 se considera muy buena.

Capítulo 4. Resultados y discusiones

4.1. Introducción

En este capítulo se describen los experimentos realizados para la evaluación de la metodología propuesta, así como los parámetros usados y configuraciones para la clasificación de las imágenes. Primero se describirán los resultados de la base de datos de Messidor y posteriormente los de Messidor 2.

4.2. Consideraciones preliminares

Se tomó la decisión de trabajar con el lenguaje de programación Python por su accesibilidad y la gran cantidad de librerías para el manejo de diversos problemas, en este caso la clasificación de imágenes. Al comienzo de este trabajo se utilizaron algunas tecnologías para el aprendizaje profundo como Digits y Tensorflow, más adelante se describen los resultados obtenidos.

4.3. Digits

Esta herramienta se caracteriza por tener una interfaz dinámica para el manejo y entrenamiento de los datos, es por ello que se decidió probar con ella. En este experimento se utilizaron algunas imágenes de la base de datos de Messidor. En la Figura 33 se muestra una gráfica obtenida de la interfaz donde se presenta la cantidad de imágenes de entrenamiento que fueron utilizadas. El total de imágenes de entrenamiento fueron 841.

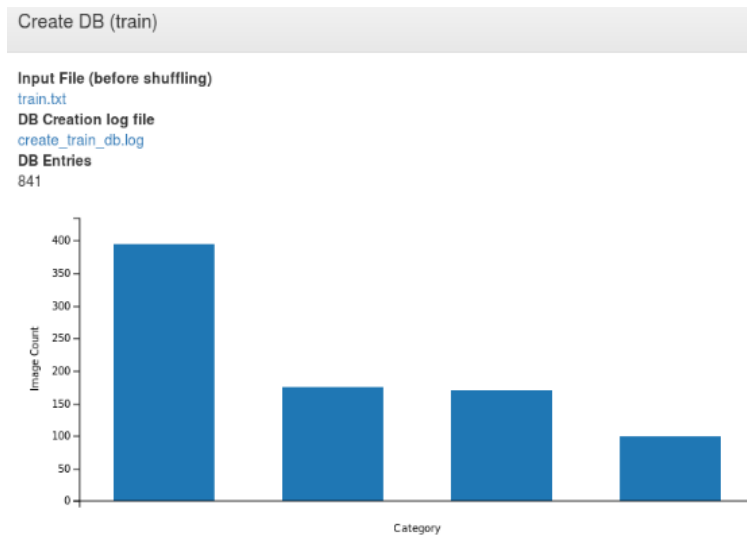


Figura 33. Imágenes de entrenamiento en Digits.

4.3.1. Ejemplos del resultado de la clasificación

En la Figura 34 se muestra el resultado de una clasificación correcta para cuatro imágenes de diferente clase, en esta figura se muestra el porcentaje que fue asignado a cada clase para clasificarla correctamente.

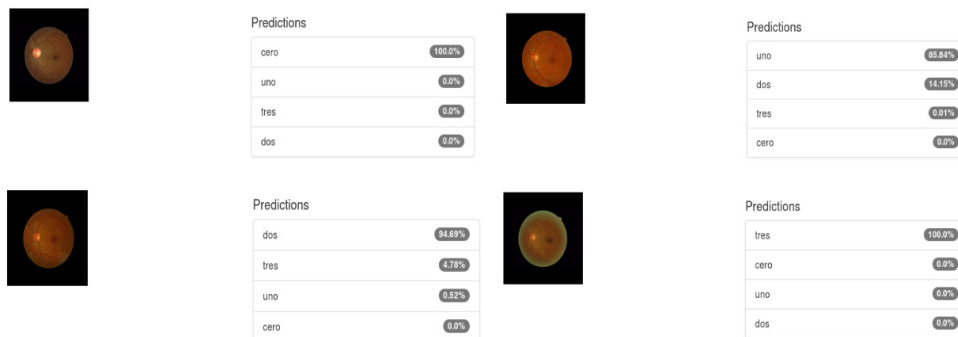


Figura 34. Imágenes clasificadas correctamente

En la Figura 35 se muestra el resultado de una clasificación incorrecta. Se aprecia que las mayoría de las imágenes fueron clasificadas con la clase 0, debido a que hay más imágenes de esa clase.

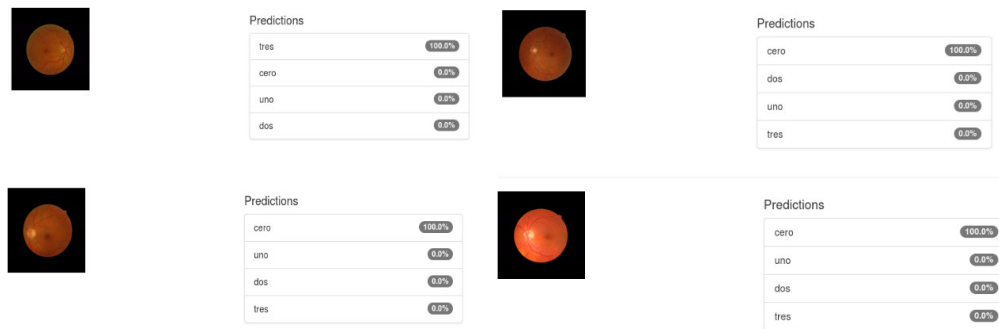


Figura 35. Imágenes clasificadas incorrectamente.

4.4. Tensorflow

Esta librería es una de las más recomendadas para la clasificación de imágenes y el uso de redes neuronales convolucionales, se realizaron algunos experimentos y se describen las más destacadas.

4.4.1. Prueba 1

En esta prueba se realizó un preprocesamiento a las imágenes que consistió en ponerlas en 50% gris, se usaron 1080 imágenes de entrenamiento y 120 imágenes de prueba. En la Figura 36 se muestra algunos ejemplos de la clasificación resultante.

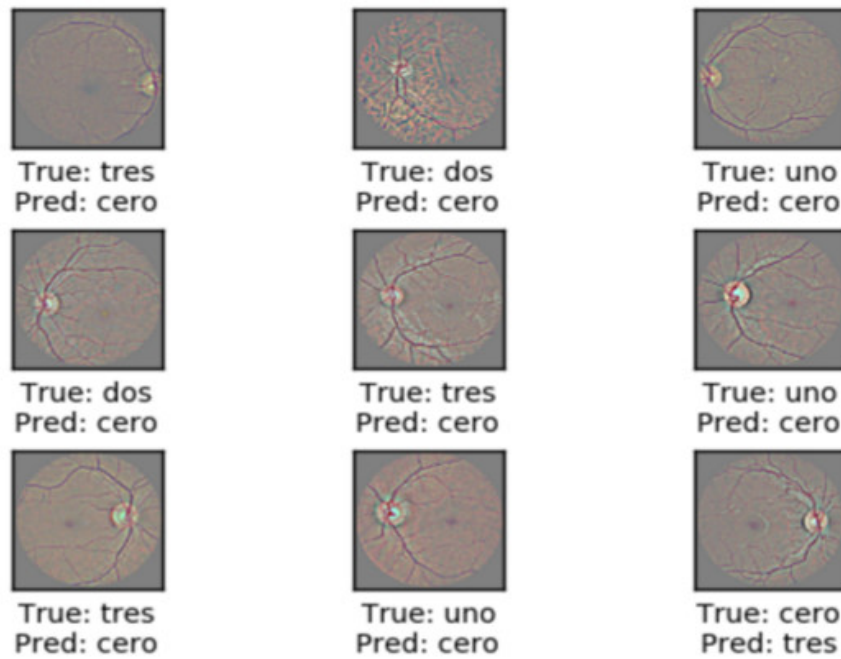


Figura 36. Imágenes clasificadas incorrectamente.

De la clasificación resultante al efectuar la prueba 1 se genera una matriz de confusión (ver tabla 14), donde se observa que una gran mayoría de las imágenes está clasificada como clase 0. Debido a que hay más imágenes en esa clase, se obtiene una precisión del 53.3% clasificando correctamente 64 imágenes.

Tabla 14. Matriz de confusión de la prueba 1.

Clase	Cero	Uno	Dos	Tres
Cero	61	2	1	1
Uno	13	0	0	2
Dos	18	1	0	1
Tres	15	1	1	3

4.4.2. Prueba 2

En esta prueba se usaron las imágenes a color y además fueron extendidas, se utilizaron 547 imágenes de entrenamiento y 61 imágenes de prueba. En la Figura 37 se muestran algunas imágenes de la clasificación resultante que fue incorrecta.

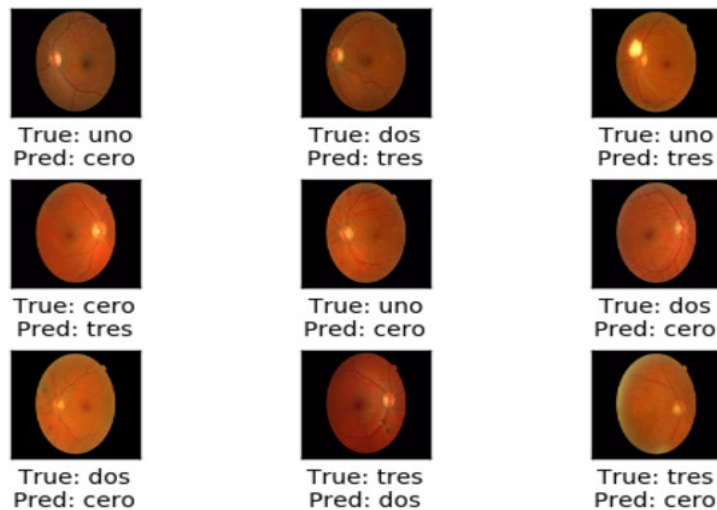


Figura 37. Imágenes clasificadas incorrectamente.

En la tabla 15 se presenta la matriz de confusión creada a partir de los resultados de la prueba donde se nota que ya no se confunde tanto con la clase 0. En este experimento se obtuvo el 63.9% de precisión, clasificando 39 imágenes correctas.

Tabla 15. Matriz de confusión de la prueba 2.

Clase	Cero	Uno	Dos	Tres
Cero	10	2	1	2
Uno	3	5	3	1
Dos	2	2	17	2
Tres	3	0	1	7

4.4.3. Prueba 3

En este experimento se usaron 1440 imágenes de entrenamiento y 160 imágenes de prueba, las transformaciones realizadas fueron rotar las imágenes y extenderlas. En la Figura 38 se muestra un ejemplo de imágenes clasificadas incorrectamente.

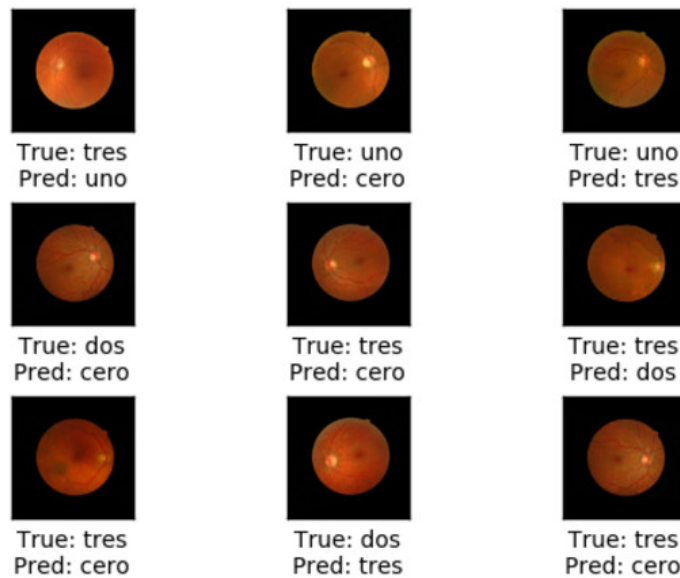


Figura 38. Imágenes clasificadas incorrectamente.

En la tabla 16 se muestra la matriz de confusión y se obtiene 63.7% de predicción, es decir 102 imágenes fueron clasificadas correctamente.

Tabla 16. Matriz de confusión de la prueba 3.

Clase	Cero	Uno	Dos	Tres
Cero	71	3	7	2
Uno	8	10	5	2
Dos	10	0	13	8
Tres	6	3	4	8

4.4.4. Prueba 4

En esta prueba se rotaron las imágenes y se extendieron, se usaron 756 imágenes de entrenamiento y 84 imágenes de prueba. En la Figura 39 se muestra un ejemplo de clasificación incorrecta de las imágenes.

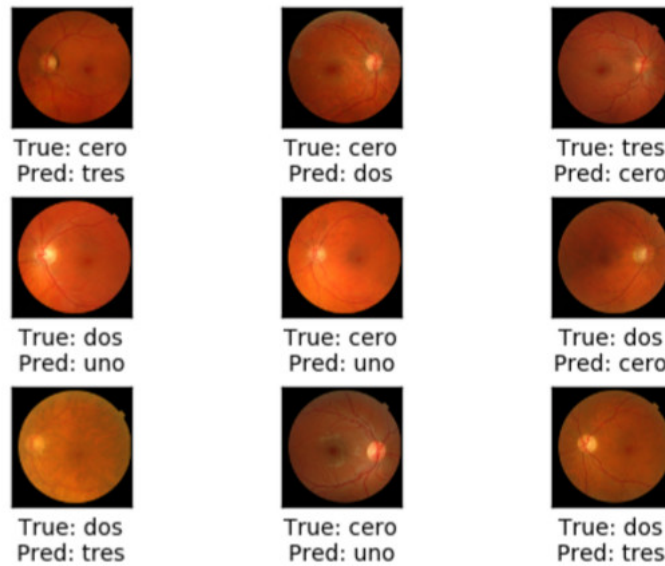


Figura 39. Imágenes clasificadas incorrectamente.

En la tabla 17 se presenta la matriz de confusión, donde se obtiene un 65.5% de precisión, clasificando 55 imágenes correctas.

Tabla 17. Matriz de confusión de la prueba 4.

Clase	Cero	Uno	Dos	Tres
Cero	14	6	3	3
Uno	1	19	3	2
Dos	2	1	6	3
Tres	3	1	1	16

Como conclusión, de los experimentos anteriores se comenta que el tamaño de la red neuronal convolucional fue pequeño ya que el número de capas usadas fue de 2 a 4, además de utilizar diferentes tamaños de kernel, siendo la mayoría grandes como por ejemplo de tamaño 9. No es conveniente aumentar las imágenes de forma manual. Conviene usar un mayor número de imágenes de entrenamiento y un mayor número de capas.

4.5. Resultados y discusiones de Messidor

Como se mencionó en el capítulo 3 se realizó un corte a las imágenes de las bases de datos, seleccionando en rectángulo que tomara en cuenta al ojo entero. Estas

imágenes fueron la entrada de la red neuronal convolucional.

Tomando como base la tabla 7 del capítulo 3, en la tabla 18 se muestra el número de capas que fueron utilizadas en cada conjunto de imágenes.

Tabla 18. Capas usadas en los diferentes tamaños de imágenes.

Número de capas	Tamaño de imágenes
1 a 11 y 19 a 21	128 x 128 pixeles
1 a 15 y 19 a 21	256 x 256 pixeles
Todas	512 x 512 pixeles

En la Figura 40 se muestra la gráfica de pérdida de entrenamiento y validación de las imágenes de 128 x 128 pixeles.

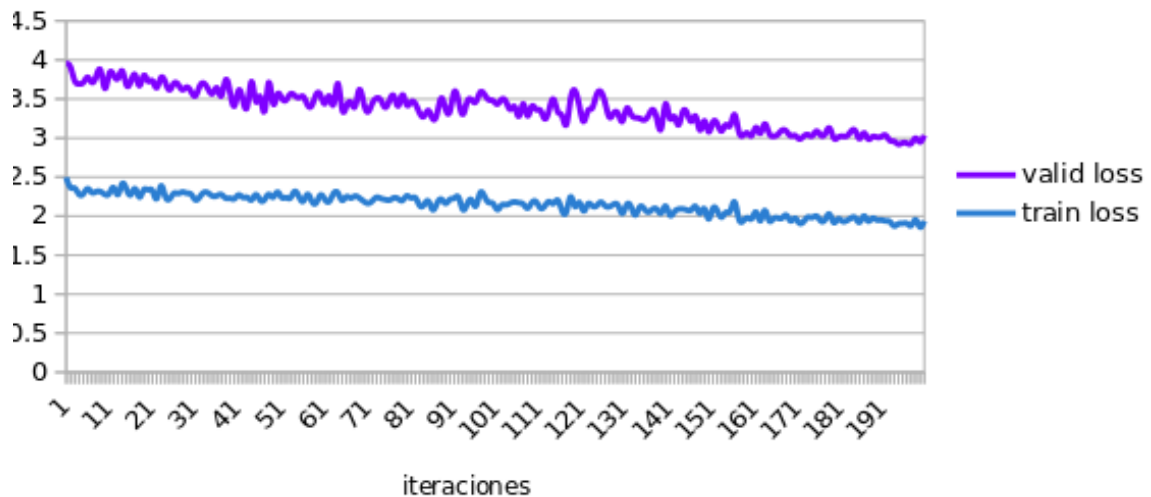


Figura 40. Pérdida de entrenamiento y validación en imágenes de 128x128 píxeles.

En la Figura 41 se muestra la gráfica que indica el valor del índice kappa que se obtuvo en el entrenamiento de las imágenes de 128 x 128 píxeles, en este caso el puntaje del índice kappa fue de 0.45.

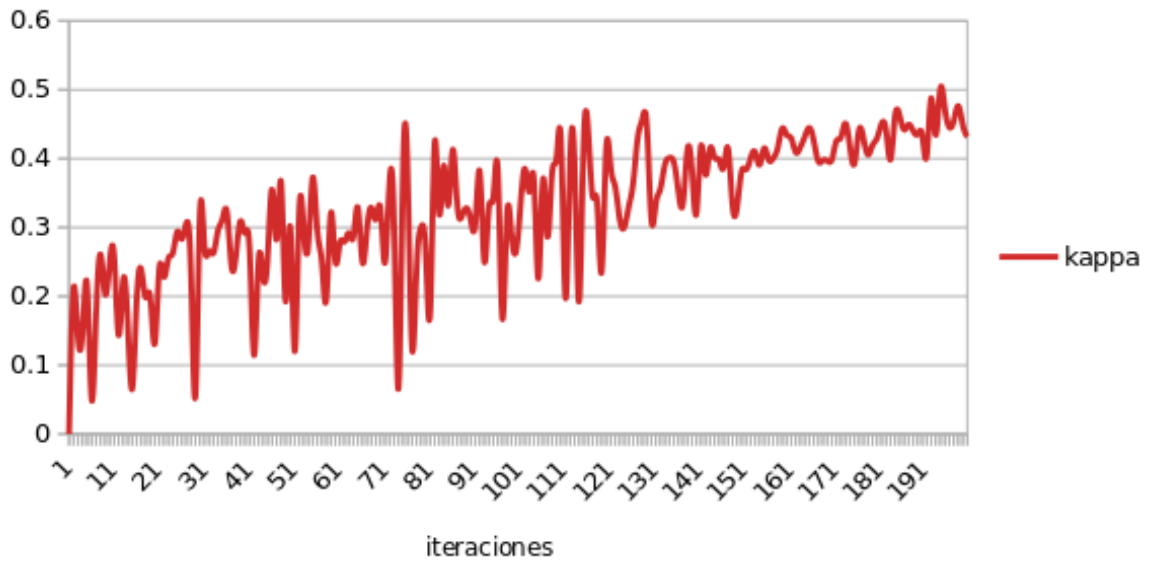


Figura 41. Índice Kappa en el entrenamiento de imágenes de 128x128 píxeles

En la Figura 42 se muestra la gráfica de pérdida de entrenamiento y validación de las imágenes de 256 x 256 píxeles.

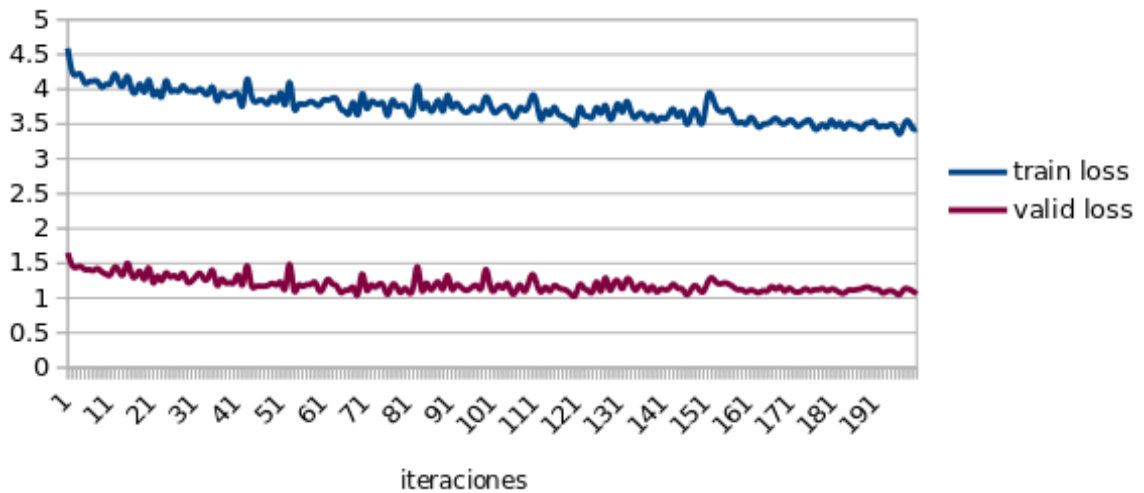


Figura 42. Pérdida de entrenamiento y validación en imágenes de 256x256 píxeles.

En la Figura 43 se muestra la gráfica que indica el valor del índice kappa que se obtuvo en el entrenamiento de las imágenes de 256 x 256 píxeles, en este caso el puntaje del índice kappa fue de 0.45.

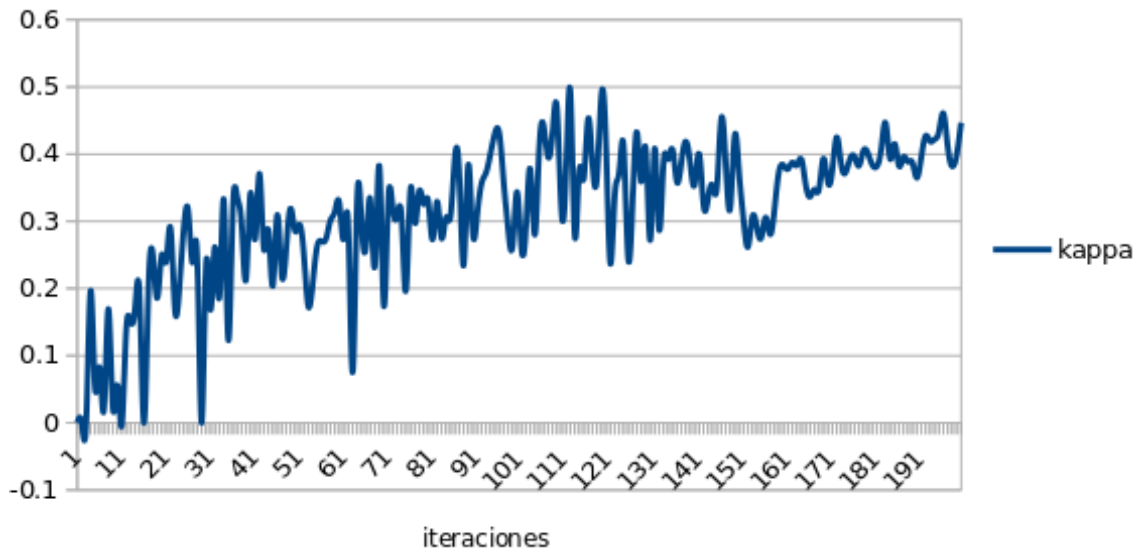


Figura 43. Índice Kappa en el entrenamiento de imágenes de 256x256 píxeles

En la Figura 44 se muestra la gráfica de pérdida de entrenamiento y validación de las imágenes de 512 x 512 píxeles.

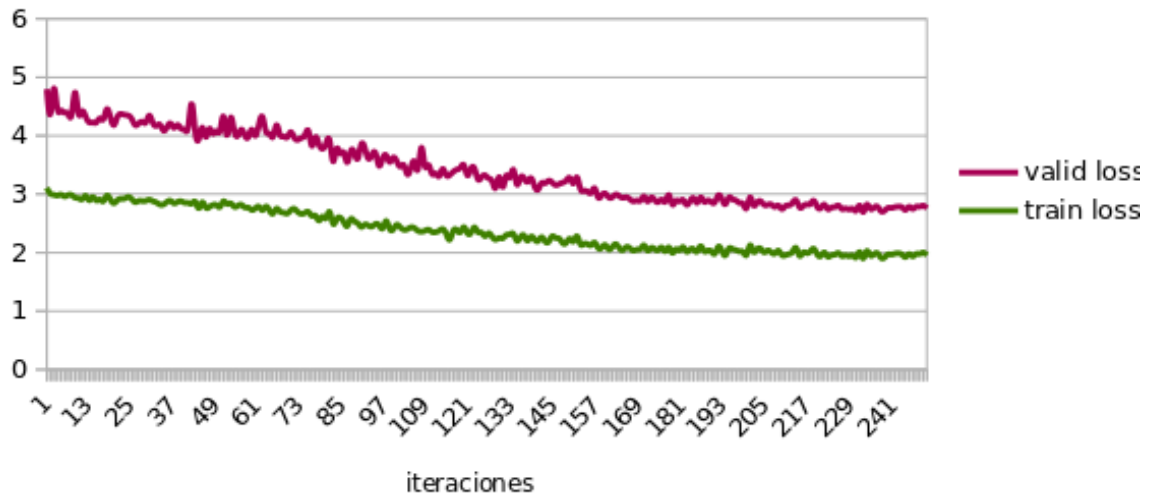


Figura 44. Pérdida de entrenamiento y validación en imágenes de 512x512 píxeles.

En la Figura 45 se muestra la gráfica que indica el valor del índice kappa que se obtuvo en el entrenamiento de las imágenes de 512 x 512 píxeles, en este caso el puntaje fue de 0.64.

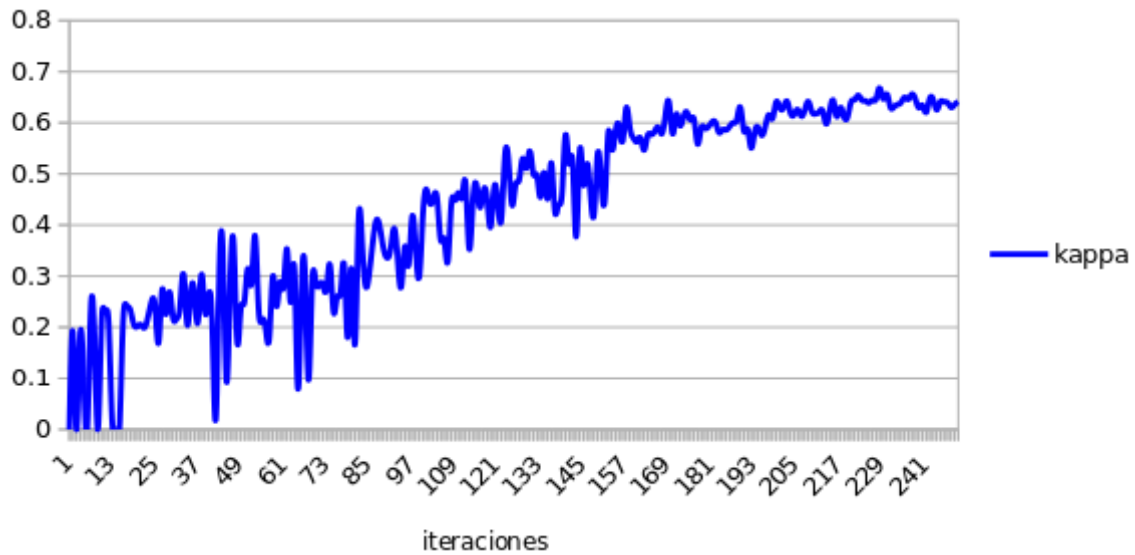


Figura 45. Índice Kappa en el entrenamiento de imágenes de 512x512 píxeles.

4.5.1. Matriz de confusión de la base de datos Messidor

En la tabla 19 se muestra la matriz de confusión donde se muestran la clasificación de las imágenes por clase. Se aprecia que los mayores errores se obtuvieron para las clases 0 y 1

Tabla 19. Matriz de confusión de Messidor.

Clase	0	1	2	3
0	32	18	2	0
1	4	6	1	1
2	4	10	6	2
3	1	4	8	9

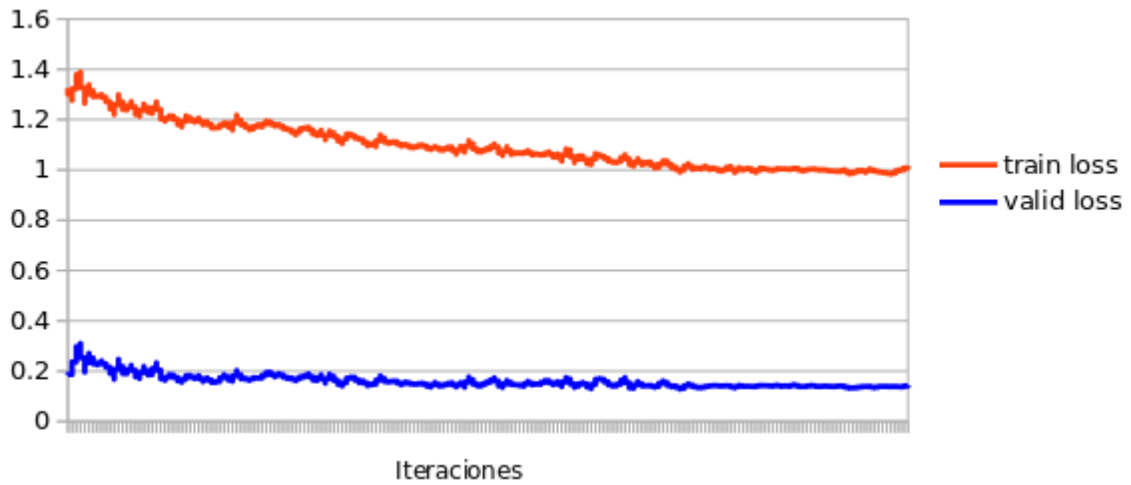
4.6. Resultados y discusiones de Messidor 2

En la tabla 20 se muestra las capas que fueron utilizadas en cada conjunto de imágenes.

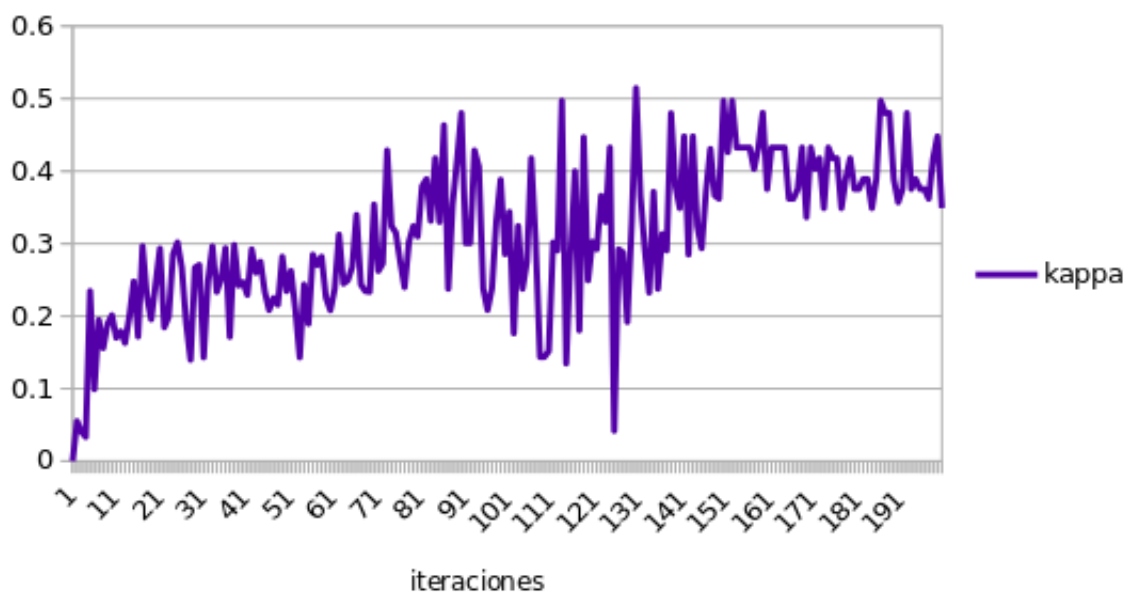
En la Figura 46 se muestra las gráfica de pérdida de entrenamiento y validación de las imágenes de 128 x 128 píxeles.

Tabla 20. Capas usadas en los diferentes tamaños de imágenes.

Número de capas	Tamaño de imágenes
1 a 11 y 19 a 21	128 x 128 pixeles
1 a 15 y 19 a 21	256 x 256 pixeles
Todas	512 x 512 pixeles

**Figura 46.** Pérdida de entrenamiento y validación en imágenes de 128x128 píxeles.

En la Figura 47 se muestra la gráfica que indica el valor del índice kappa que se obtuvo en el entrenamiento de las imágenes de 128 x 128 píxeles, en este caso el puntaje fue de 0.33.

**Figura 47.** Índice Kappa en el entrenamiento de imágenes de 128x128 píxeles

En la Figura 48 se muestra la gráfica de pérdida de entrenamiento y validación de las imágenes de 128 x 128 píxeles.

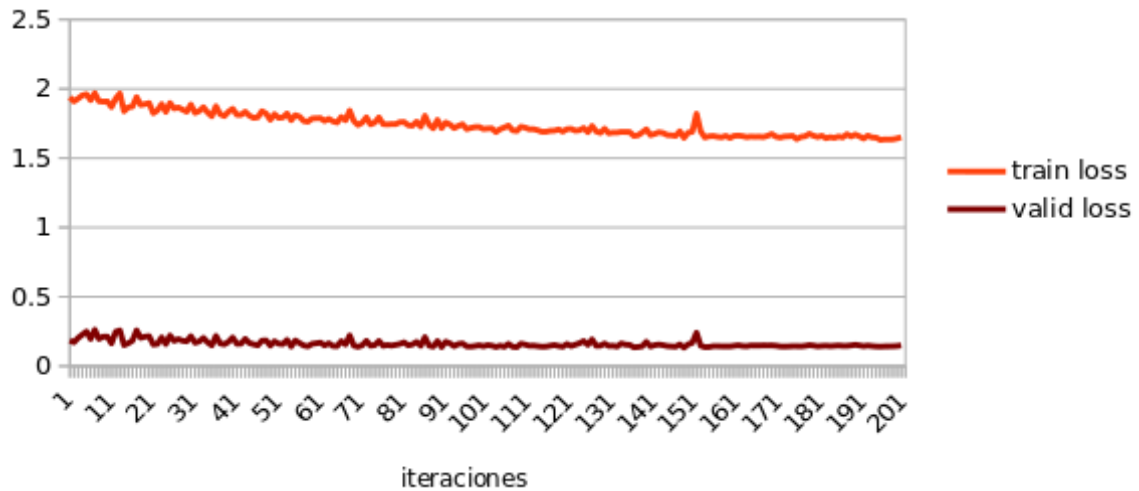


Figura 48. Pérdida de entrenamiento y validación en imágenes de 256x256 píxeles.

En la Figura 49 se muestra la gráfica que indica el valor del índice kappa que se obtuvo en el entrenamiento de las imágenes de 256 x 256 píxeles, en este caso el puntaje fue de 0.34.

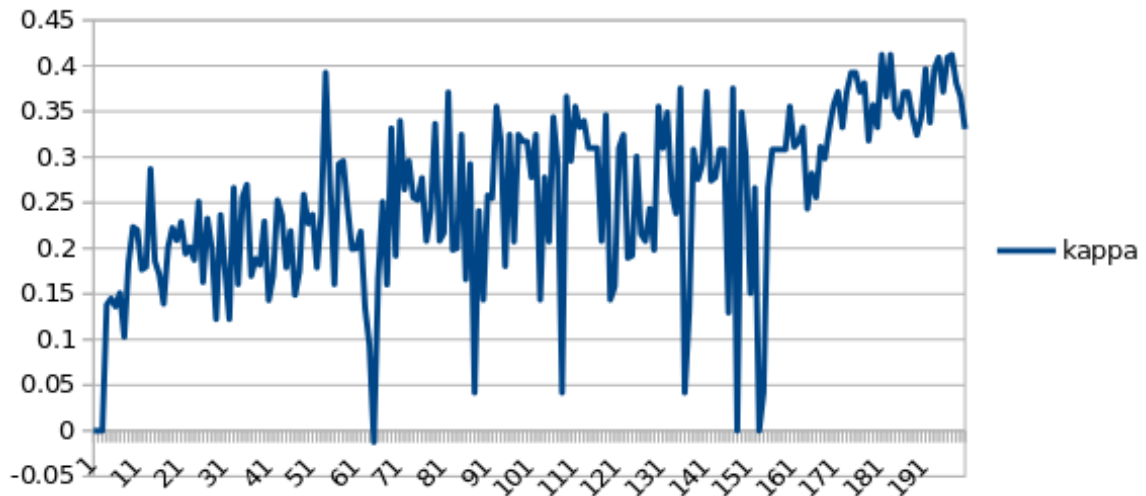


Figura 49. Índice Kappa en el entrenamiento de imágenes de 256x256 píxeles

En la Figura 50 se muestra la gráfica de pérdida de entrenamiento y validación de las imágenes de 128 x 128 píxeles.

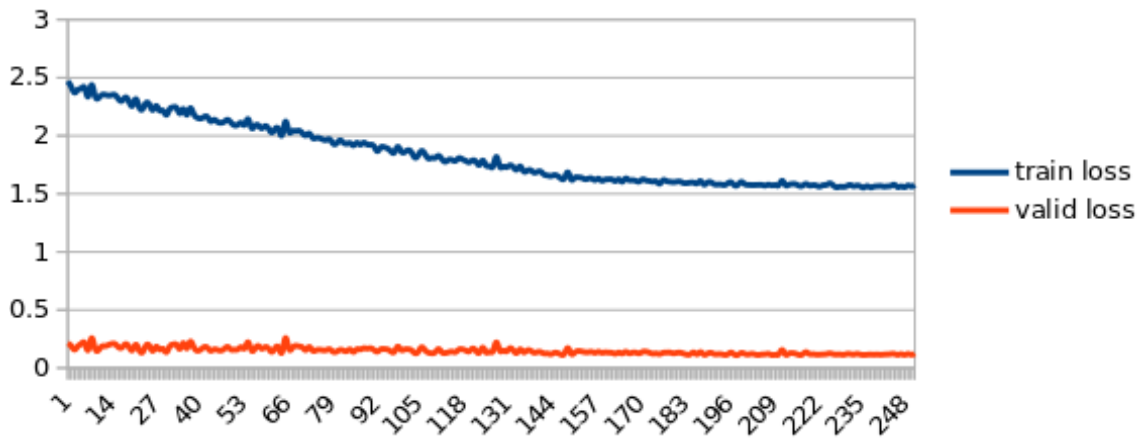


Figura 50. Pérdida de entrenamiento y validación en imágenes de 512x512 píxeles.

En la Figura 51 se muestra la gráfica que indica el valor del índice kappa que se obtuvo en el entrenamiento de las imágenes de 256 x 256 píxeles, en este caso el puntaje fue de 0.56.

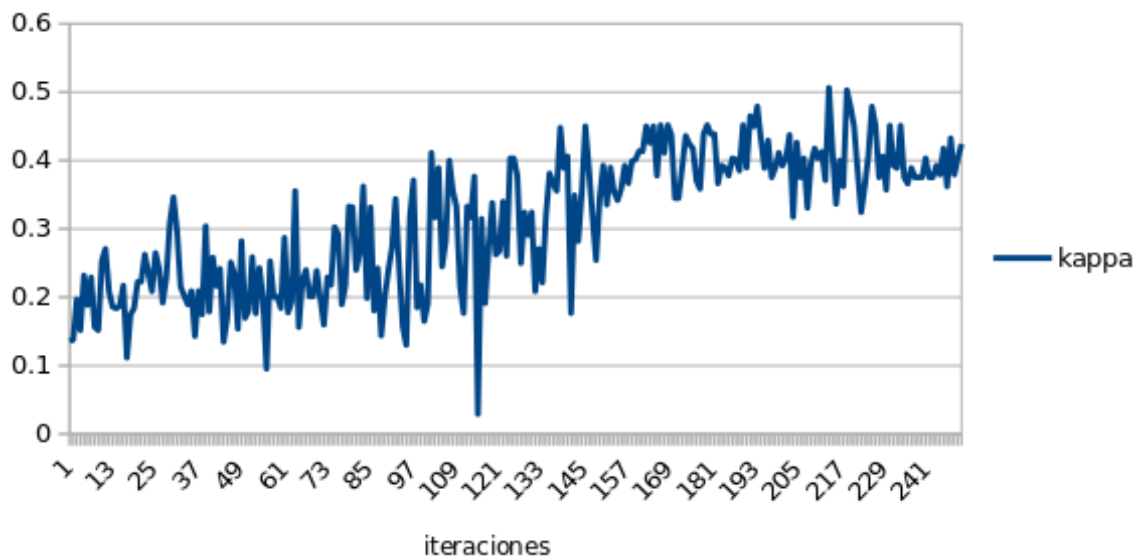


Figura 51. Índice Kappa en el entrenamiento de imágenes de 512x512 píxeles.

4.6.1. Matriz de confusión de la base de datos Messidor 2

En la tabla 21 se muestra la matriz de confusión donde se muestran la clasificación de las imágenes por clase.

- Clase 0: 118 imágenes de la clase 0 fueron clasificadas correctamente y 11 se confundieron con la clase 1.

- Clase 1: 19 imágenes fueron clasificadas correctamente y 10 se confundieron con la clase 0.

Tabla 21. Matriz de confusión de Messidor 2

Clase	0	1
0	118	11
1	10	19

De acuerdo con los resultados de la base de datos de Messidor, se observa que arrojaron un puntaje de 0.64, es decir tiene una fuerza de concordancia buena, mientras que la base de datos de Messidor 2 obtuvo un puntaje de 0.56, que significa una fuerza de concordancia moderada.

Capítulo 5. Conclusiones y trabajo futuro

5.1. Introducción

Este capítulo tiene el objetivo de exponer las conclusiones del trabajo de investigación, así como las aportaciones, y el trabajo a futuro.

5.2. Conclusiones

El procedimiento para diagnosticar la retinopatía diabética es un proceso tardado si se realiza de forma manual.

En este trabajo de investigación se propuso diseñar una metodología que permita la clasificación de imágenes de fondo del ojo donde se presentan diversas lesiones tales como: microaneurismas, exudados y hemorragias. El objetivo fue clasificar esas imágenes en las diferentes etapas de la retinopatía diabética.

Se diseñó una metodología que permitiera cumplir con el objetivo, por lo que se hizo la propuesta de hacer un pre-procesamiento que consistió en adecuar las imágenes para la entrada a la red neuronal. Al encontrar el problema de que había más imágenes en la clase 0 se propuso el balanceo de pesos correspondiente a cada clase, lo cual ayudó a obtener mejores resultados.

Una forma de aumentar la precisión en los resultados es realizar el triplicado de las bases de datos y probar con diferentes tamaños de imágenes para entrenar en secuencia, sin embargo es un proceso tardado.

No contar con un equipo de cómputo equipado con tarjetas gráficas para realizar las pruebas hizo que el tiempo de ejecución de la propuesta no fuera el deseado. Para tareas como la clasificación de imágenes y el manejo de grandes cantidades de datos se recomienda hacer uso de esta tecnología.

Sin embargo los resultados de la base de datos de Messidor arrojaron un puntaje de 0.64 de acuerdo a la valoración del índice kappa, lo cual muestra que tiene una fuerza de concordancia buena, mientras que para la base de datos de Messidor 2 el puntaje fue de 0.56, que significa una fuerza de concordancia moderada.

Este trabajo de investigación no se puede comparar con el estado del arte debido a que no se usaron grandes cantidades de imágenes y no se realiza otro preprocesamiento para resaltar las lesiones presentadas en la enfermedad.

5.2.1. Aportaciones

Las principales aportaciones de este trabajo de investigación son:

- Desarrollo de una metodología para la clasificación de etapas de la retinopatía diabética utilizando imágenes de fondo del ojo.
- Diseño de una red neuronal convolucional para la clasificación de la retinopatía diabética.

5.2.2. Trabajo futuro

Como trabajo futuro se proponen las siguientes actividades:

- Utilizar una base de datos con mayor cantidad de imágenes para el entrenamiento y prueba de la red neuronal convolucional.
- Probar otras técnicas de procesamiento de imágenes para mejorar y/o resaltar características de la imagen como las lesiones presentadas en la enfermedad.
- Probar con diferentes parámetros en el entrenamiento de la red neuronal convolucional.
- Trabajar conjuntamente con una clínica especializada en la detección de la retinopatía diabética para hacer un análisis en un ambiente médico real.

Literatura citada

- AAO: American academy of ophthalmology (2015a). Simulador: Retinopatía diabética proliferativa - American academy of ophthalmology. Recuperado de: <https://www.aao.org/salud-ocular/enfermedades/simulador-retinopatia-diabetica-proliferativa>, Enero 2018.
- AAO: American academy of ophthalmology (2015b). Simulador: Retinopatía diabética no proliferativa - American academy of ophthalmology. Recuperado de: <https://www.aao.org/salud-ocular/enfermedades/simulador-retinopatia-diabetica-no-proliferativa>, Enero 2018.
- Abràmoff, M. D., Niemeijer, M., Suttorp-Schulten, M. S. A., Viergever, M. A., Russell, S. R., y Van Ginneken, B. (2008). Evaluation of a system for automatic detection of diabetic retinopathy from color fundus photographs in a large population of patients with diabetes. *Diabetes Care*, **31**(2): 193–198.
- Abramoff, M. D., Niemeijer, M., y Russell, S. R. (2010). Automated detection of diabetic retinopathy: barriers to translation into clinical practice. *Expert Review of Medical Devices*, **7**(2): 287–296.
- Abràmoff, M. D., Folk, J. C., Han, D. P., Walker, J. D., Williams, D. F., Russell, S. R., Massin, P., Cochener, B., Gain, P., Tang, L., Lamard, M., Moga, D. C., Quellec, G., y Niemeijer, M. (2013). Automated analysis of retinal images for detection of referable diabetic retinopathy. *JAMA Ophthalmology*, **131**(3): 351–357.
- Adal, K. M., Sidibé, D., Ali, S., Chaum, E., Karnowski, T. P., y Mériaudeau, F. (2014). Automated detection of microaneurysms using scale-adapted blob analysis and semi-supervised learning. *Computer Methods and Programs in Biomedicine*, **114**(1): 1–10.
- Ali Shah, S. A., Laude, A., Faye, I., y Tang, T. B. (2016). Automated microaneurysm detection in diabetic retinopathy using curvelet transform. *Journal of Biomedical Optics*, **21**(10).
- Amin, J., Sharif, M., y Yasmin, M. (2016). A Review on recent developments for detection of diabetic retinopathy. *Scientifica*, **2016**: 1–20.
- Antal, B. y Hajdu, A. (2012). An ensemble-based system for microaneurysm detection and diabetic retinopathy grading. *IEEE Transactions on Biomedical Engineering*, **59**(6): 1720–1726.
- Bradley Efron, R. y Tibshirani, R. (1979). *Amazon.com: An introduction to the bootstrap*. Chapman & Hall. p. 456.
- CS231 (2016). CS231n Convolutional neural networks for visual recognition. Recuperado de: <http://cs231n.github.io/convolutional-networks/>, Noviembre 2018.
- Decencière, E., Zhang, X., Cazuguel, G., Lay, B., Cochener, B., Trone, C., Gain, P., Ordonez, R., Massin, P., Erginay, A., Charton, B., y Klein, J.-C. (2014). Feedback on a publicly distributed image database: the messidor database. *Image Analysis & Stereology*, **33**(3): 231.
- Dieleman, S. (2015). Classifying plankton with deep neural networks. Recuperado de: <http://benanne.github.io/2015/03/17/plankton.html>, Noviembre 2018.

- FMD: Federación mexicana de la diabetes (2017). Federación mexicana de diabetes – ¿Qué es la diabetes? Recuperado de: <http://fmdiabetes.org/que-es-la-diabetes-2/>, Enero 2018.
- Goodfellow, I., Bengio, Y., y Courville, A. (2016). *Deep Learning*. MIT Press. Recuperado de: <http://www.deeplearningbook.org>, Noviembre 2018.
- He, K., Zhang, X., Ren, S., y Sun, J. (2015). Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. *The Journal of cell biology*, **86**(3): 1–5.
- Kaggle (2015). National data science bowl. Recuperado de: <https://www.kaggle.com/c/datasciencebowl>, Noviembre 2017.
- Krizhevsky, A. (2009). Learning Multiple Layers of Features from Tiny Images. *Science Department, University of Toronto, Tech.*, pp. 1–60.
- Krizhevsky, A., Sutskever, I., y Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Advances In Neural Information Processing Systems*, pp. 1–9.
- Lazar, I. y Hajdu, A. (2013). Retinal microaneurysm detection through local rotating cross-section profile analysis. *IEEE Transactions on Medical Imaging*, **32**(2): 400–407.
- Li, Fei-Fei; R, Fergus; Pietro, P. (2004). Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories. Recuperado de: http://www.vision.caltech.edu/Image_Datasets/Caltech101/, Noviembre 2018.
- Mathers, C. D. y Loncar, D. (2006). Projections of global mortality and burden of disease from 2002 to 2030. *PLoS Medicine*, **3**(11): 442.
- McCarthy, J., Minsky, M. L., Rochester, N., y Shannon, C. E. (2006). A proposal for the Dartmouth summer research project on artificial intelligence. *AI Magazine*, **27**(4): 12–14.
- Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill.
- NEI: National eye institute (2016). La retinopatía diabética, lo que usted debe saber | National Eye Institute. Recuperado de: <https://nei.nih.gov/health/espanol/retinopatia>, Noviembre 2017.
- Niemeijer, M., van Ginneken, B., Staal, J., Suttorp-Schulten, M. S. A., y Abramoff, M. D. (2005). Automatic detection of red lesions in digital color fundus photographs. *IEEE transactions on medical imaging*, **24**(5): 584–592.
- NVIDIA (2016). The difference between AI, machine Learning, and deep learning? | NVIDIA Blog. Recuperado de: <https://blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai/>, Diciembre 2018.
- Oloumi, F., Rangayyan, R. M., y Ells, A. L. (2013). Computer-aided diagnosis of proliferative diabetic retinopathy via modeling of the major temporal arcade in retinal fundus images. *Journal of digital imaging*, **26**(6): 1124–1130.

- OMS: Organización mundial de la salud (2017). OMS | Diabetes. Recuperado de: <http://www.who.int/mediacentre/factsheets/fs312/es/>, Noviembre 2017.
- Pérez Molina, E. C. y León Veitía, L. (2017). La fotografía de fondo de ojo como método de diagnóstico en el glaucoma. *Medicentro Electrónica*, **21**: 3–10.
- Riordan Eva, P. y Cunningham, E. (2012). *Oftalmología general Vaughan Y Asbury*. p. 533.
- Roberts, E. (2016). Neural networks - History. Recuperado de: <https://cs.stanford.edu/people/eroberts/courses/soco/projects/neural-networks/History/history1.html>, Noviembre 2018.
- Santhi, D., Manimegalai, D., Parvathi, S., y Karkuzhali, S. (2016). Segmentation and classification of bright lesions to diagnose diabetic retinopathy in retinal images. *Biomedizinische Technik. Biomedical engineering*, **61**(4).
- Silberman, N., Ahlrich, K., Fergus, R., y Subramanian, L. (2010). Case for Automated Detection of Diabetic Retinopathy. pp. 85–90.
- Sinthanayothin, C., Boyce, J. F., Cook, H. L., y Williamson, T. H. (1999). Automated localisation of the optic disc, fovea, and retinal blood vessels from digital colour fundus images. *The British journal of ophthalmology*, **83**(8): 902–910.
- Sinthanayothin, C., Boyce, J. F., Williamson, T. H., Cook, H. L., Mensah, E., Lal, S., y Usher, D. (2002). Automated detection of diabetic retinopathy on digital fundus images. *Diabetic medicine : a journal of the British Diabetic Association*, **19**(2): 105–112.
- SOCV: Sociedad oftalmológica de la comunidad valenciana (2014). Anatomía del ojo - SOCV - Sociedad oftalmológica de la comunidad valenciana. Recuperado de: <http://www.socv.org/anatomia-del-ojo/>, Diciembre 2017.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., y Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, **15**: 1929–1958.
- Walter, T., Klein, J.-C., Massin, P., y Erginay, A. (2002). A contribution of image processing to the diagnosis of diabetic retinopathy—detection of exudates in color fundus images of the human retina. *IEEE transactions on medical imaging*, **21**(10): 1236–1243.
- Wei, X.-S. (2015). Must know tips/tricks in deep neural networks. Recuperado de: <http://lamda.nju.edu.cn/weixs/project/CNNTricks/CNNTricks.html>, Noviembre 2018.
- Xu, B., Wang, N., Chen, T., y Li, M. (2015). Empirical evaluation of rectified activations in convolution network. *ICML Deep Learning Workshop*, pp. 1–5.