Clasificación de patologías oculares que causan ceguera irreversible usando arquitectura de aprendizaje profundo.

María Fernanda Cubides Jaimes

Juliana Alejandra Arenas Lobo

Erick Santiago Riaño Pérez



UNIVERSIDAD DE INVESTIGACIÓN Y DESARROLLO-UDI

FACULTAD DE INGENIERÍA

PROGRAMA DE INGENIERÍA DE SISTEMAS

BUCARAMANGA

2024

Clasificación de patologías oculares que causan ceguera irreversible usando arquitectura de aprendizaje profundo.

María Fernanda Cubides Jaimes

Juliana Alejandra Arenas Lobo

Erick Santiago Riaño Pérez

UNIVERSIDAD DE INVESTIGACIÓN Y DESARROLLO-UDI

FACULTAD DE INGENIERÍA

PROGRAMA DE INGENIERÍA DE SISTEMAS

BUCARAMANGA

2024

Clasificación de patologías oculares que causan ceguera irreversible usando arquitectura de aprendizaje profundo.

María Fernanda Cubides Jaimes, Juliana Alejandra Arenas Lobo y Erick Santiago Riaño Pérez.

Proyecto de Grado Presentado como requisito para optar al título de Ingeniero de Sistemas

Director MSc. Jorge Giovanny Castellanos Valderrama

CoDirector PhD. Crisóstomo Alberto Barajas Solano

Universidad de Investigación y Desarrollo - UDI

Facultad de Ingeniería

Programa de Ingeniería de Sistemas

Bucaramanga 2024

Nota de Aceptación del Documento Final

Dedicatoria

*A mis padres, Claudia Jaimes y Mauricio Cubides, por su constante aliento y ayuda, a mi hermano Santi y mi abuela Marta Pinto por su cariño y alegría,*

*A mi pareja Antony por su apoyo inquebrantable; a mi mejor amiga Juliana por su lealtad, a mi director y co-director por su guía.*

*Y por último a mí misma por no rendirme y superar cada obstáculo.*

**Maria Fernanda Cubides Jaimes**

*A Dios, por su guía y fortaleza en cada paso de mi camino. A mi madre y abuela, por su amor incondicional y sacrificios.*

*A mi mejor amiga María Fernanda, por su apoyo constante siempre.*

*A mi amigo Erick, por su compañía, y a mi director y co-director, por su invaluable orientación. Finalmente, a mí misma por nunca rendirme.*

**Juliana Alejandra Arenas Lobo**

*A mi familia, por su amor incondicional y apoyo en cada paso de este camino.*

*A mis compañeras, por su compañía, su constante ánimo y por ser una fuente de fortaleza en los momentos difíciles. A mi director y co-director de proyecto, por su orientación y enseñanzas que marcaron una diferencia grande en mi formación.*

**Erick Santiago Riaño Perez**

Agradecimientos

Nuestros agradecimientos a quienes nos apoyaron en la realización de este proyecto.

A la Universidad de Investigación y Desarrollo, institución de la cual somos estudiantes, gracias por darnos la facilidad y la oportunidad de acceder a este programa, así logramos terminar nuestros estudios de pregrado.

A nuestro director y co-director de proyecto por su permanente interés y acompañamiento para llevar a cabo el proyecto.

A nuestro tutor por compartir su experiencia y conocimiento.

Por último, a nuestra familia quienes nos apoyaron y animaron para la culminación de esta etapa.

**Resumen**

Este proyecto aborda la clasificación de enfermedades oculares, específicamente retinopatía diabética, cataratas y glaucoma, mediante el uso de tecnologías de aprendizaje profundo. Estas enfermedades afectan a más de 1.300 millones de personas a nivel mundial, y su detección temprana es crucial para prevenir complicaciones graves (Cuidarte, n.d). La investigación se basa en la aplicación de redes neuronales convolucionales y técnicas de transferencia de aprendizaje para mejorar la precisión diagnóstica a partir de imágenes de fondo de ojo (retinoscopía). Se utilizaron modelos disponibles en el estado del arte, junto con un conjunto de datos que incluye imágenes de retina diagnosticadas; el objetivo es seleccionar la arquitectura de aprendizaje profundo más adecuada para la clasificación de estas patologías oculares. Los resultados del proyecto incluyen la creación de un modelo que mejore la precisión diagnóstica y contribuya al avance en el campo de la inteligencia artificial aplicada a la medicina ocular.

Palabras clave: Clasificación de imágenes, aprendizaje profundo, enfermedades oculares, retinoscopía.

**Abstract**

This project addresses the classification of eye diseases, specifically diabetic retinopathy, cataracts and glaucoma, using deep learning technologies. These diseases affect more than 1.3 million people worldwide, and their early detection is crucial to prevent serious complications (Cuidarte, n.d). The research is based on the application of convolutional neural networks and transfer learning techniques to improve diagnostic accuracy from fundus images (retinoscopy). Models available in the state of the art were used, altogether with a dataset including diagnostic retinal images. The goal is to select the most suitable deep learning architecture for the classification of these ocular pathologies. The results of the project include the development of a model that offers diagnostic accuracy and contributes to the advancement in the field of artificial intelligence applied to ocular medicine.

Keywords: Image classification, deep learning, ocular diseases, retinoscopy.

Hoja de Presentación del Proyecto Grado

Universidad de Investigación y desarrollo-UDI-

Ciudad: Bucaramanga

Facultad: Ingenierías

Programa: Ingeniería de sistemas

Proyecto de Grado

Título: Clasificación de patologías oculares que causan ceguera irreversible usando arquitectura de aprendizaje profundo.

Autor Identificación

María Fernanda Cubides Jaimes 1005163255

Juliana Alejandra Arenas Lobo 1005163255

Erick Santiago Riaño Pérez 1005162937

Director del Proyecto

MSc. Jorge Giovanny Castellanos Valderrama

Fecha de Entrega: 22/11/2024

**Tabla de Contenido**

[Introducción 22](#_Toc183157786)

[1. Problemática a Investigar 23](#_Toc183157787)

[1.1. Planteamiento del problema 23](#_Toc183157788)

[2. Objetivo del proyecto 25](#_Toc183157789)

[2.1 Objetivo General 25](#_Toc183157790)

[2.2 Objetivo Específico 25](#_Toc183157791)

[3. Justificación. 26](#_Toc183157792)

[3.1 Impacto Esperado 27](#_Toc183157793)

[4. Metodología 28](#_Toc183157794)

[4.1 Tipos de Investigación 28](#_Toc183157795)

[4.2 Caracterización de Imágenes 28](#_Toc183157796)

[4.3 Hipótesis 29](#_Toc183157797)

[4.4 Método científico 29](#_Toc183157798)

[4.5 Metodología de Prototipado Rápido (MPR) 30](#_Toc183157799)

[4.6 Metodología Kanban 30](#_Toc183157800)

[4.7 Incorporación de las Metodologías (MPR, Kanban, Científica) 31](#_Toc183157801)

[4.8 Método de Investigación 32](#_Toc183157802)

[5. Marco Referencial 36](#_Toc183157803)

[5.1 Marco Teórico. 36](#_Toc183157804)

[5.1.1 Aprendizaje Automático 36](#_Toc183157805)

[5.1.2 Red neuronal artificial 37](#_Toc183157806)

[5.1.3 Aprendizaje profundo 38](#_Toc183157807)

[5.1.4 Aplicativos del Aprendizaje profundo 41](#_Toc183157808)

[5.1.5 Diferencia entre Aprendizaje profundo y Aprendizaje automático 42](#_Toc183157809)

[5.1.6 Funcionamiento del Aprendizaje profundo 46](#_Toc183157810)

[5.1.7 Redes neuronales convolucionales (CNN) 51](#_Toc183157811)

[5.1.8 Arquitecturas Aprendizaje profundo 52](#_Toc183157812)

[5.1.9 Kanban 55](#_Toc183157813)

[5.2 Marco Conceptual 59](#_Toc183157814)

[5.2.1 Conjunto de Datos (Dataset) 59](#_Toc183157815)

[5.2.2 Modelos Aprendizaje profundo 59](#_Toc183157816)

[5.2.3 Python 60](#_Toc183157817)

[5.2.4 Entorno de desarrollo 60](#_Toc183157818)

[5.2.5 Artefacto 65](#_Toc183157819)

[5.2.6 Método científico 65](#_Toc183157820)

[5.2.7 Teorema de Convolución 66](#_Toc183157821)

[5.2.8 Funcionamiento de la convolución en imágenes 66](#_Toc183157822)

[5.3 Marco Histórico 67](#_Toc183157823)

[5.3.1 Contexto histórico de la Inteligencia artificial 67](#_Toc183157824)

[5.3.2 Contexto histórico del aprendizaje profundo para clasificación de enfermedades oculares. 68](#_Toc183157825)

[5.4 Marco Normativo 69](#_Toc183157826)

[5.5 Marco Ambiental 69](#_Toc183157827)

[5.6 Marco Cultural 70](#_Toc183157828)

[5.7 Estado del Arte 70](#_Toc183157829)

[6. Referentes Metodológicos. 77](#_Toc183157830)

[6.1 Desarrollo de Metodología 82](#_Toc183157831)

[6.1.1 Fase 1: Recopilación de datos 82](#_Toc183157832)

[6.1.2 Fase 2: Selección de Modelos CNN 90](#_Toc183157833)

[6.1.3 Fase 3: Integración del Modelo CNN 100](#_Toc183157834)

[6.1.4 Fase 4: Escritura 105](#_Toc183157835)

[7. Presentación y análisis de resultados 106](#_Toc183157836)

[7.1 Desarrollo Fase 2 106](#_Toc183157837)

[7.2 Preprocesamiento de las Imágenes 106](#_Toc183157838)

[7.3 Selección de Arquitecturas para la Clasificación del Ciclo 1 107](#_Toc183157839)

[7.3.1 Dataset utilizado Ciclo 1 108](#_Toc183157840)

[7.3.2. Hiperparámetros y Configuraciones Ciclo 1 108](#_Toc183157841)

[7.3.3 Ajuste de paciencia Ciclo 1 109](#_Toc183157842)

[7.3.4 Análisis de resultados Ciclo 1 110](#_Toc183157843)

[7.4 Conclusiones y Justificación de Selección Ciclo 1 120](#_Toc183157844)

[7.4.1 Conclusiones con respecto a las tablas de varianza y tablas generales 120](#_Toc183157845)

[7.4.2 Conclusiones con respecto a las tablas de métricas de cada arquitectura y matrices de confusión: 122](#_Toc183157846)

[7.5 Selección de Arquitecturas para la Clasificación del Ciclo 2 129](#_Toc183157847)

[7.5.1 Dataset utilizado Ciclo 2 129](#_Toc183157848)

[7.5.2. Hiperparámetros y Configuraciones 130](#_Toc183157849)

[7.5.3 Ajuste de paciencia Ciclo 2 131](#_Toc183157850)

[7.6. Conclusiones y Justificación de Selección Ciclo 2 141](#_Toc183157851)

[7.6.1 Conclusiones Ciclo 2 con respecto a las tablas de varianza, tablas generales y tablas por learning rate 141](#_Toc183157852)

[7.6.2 Conclusiones Ciclo 2 con respecto a las tablas de métricas de cada arquitectura y matrices de confusión 142](#_Toc183157853)

[7.7 Selección de Arquitecturas para la Clasificación del Ciclo 3 152](#_Toc183157854)

[7.7.1 Dataset utilizado en Ciclo 3 153](#_Toc183157855)

[7.7.1 Hiperparámetros y Configuraciones 153](#_Toc183157856)

[7.7.2 Ajuste de paciencia Ciclo 3 155](#_Toc183157857)

[7.7.3 Análisis de Resultados 155](#_Toc183157858)

[7.7.4 Conclusiones Ciclo 3 con respecto a las tablas de varianza, tablas generales y tablas por learning rate 158](#_Toc183157859)

[7.7.5 Conclusiones Ciclo 3 con respecto a las tablas de métricas y matrices de confusión 158](#_Toc183157860)

[7.8 Desarrollo Fase 3 164](#_Toc183157861)

[7.8.1 Desempeño del modelo CNN en la clasificación de imágenes de fondo de ojo 164](#_Toc183157862)

[7.8.2 Matriz de confusión 164](#_Toc183157863)

[7.8.3 Desarrollo e integración de la interfaz de usuario 166](#_Toc183157864)

[7.8.4 Codificación de la interfaz 166](#_Toc183157865)

[7.8.5 Pruebas de integración y rendimiento del sistema 168](#_Toc183157866)

[7.9 Interacción de los usuarios de prueba con la interfaz 169](#_Toc183157867)

[7.9.1 Descripción de los usuarios de prueba 169](#_Toc183157868)

[7.9.2 Metodología de las pruebas 171](#_Toc183157869)

[7.9.3 Resultados 172](#_Toc183157870)

[4.9.4 Errores identificados durante las pruebas 187](#_Toc183157871)

[7.9.5 Conclusiones generales de las pruebas 187](#_Toc183157872)

[7.10 Pruebas con diferentes datasets 188](#_Toc183157873)

[8. Cumplimiento de Objetivos Específicos 193](#_Toc183157874)

[8.1 Cumplimiento de Objetivo Especifico 1 193](#_Toc183157875)

[8.2 Cumplimiento de Objetivo Especifico 2 194](#_Toc183157876)

[8.3 Cumplimiento de Objetivo Especifico 3 197](#_Toc183157877)

[8.4 Cumplimiento de Objetivo Especifico 4 199](#_Toc183157878)

[9. Conclusiones 202](#_Toc183157879)

[9.1 Conclusiones por Ciclos 202](#_Toc183157880)

[9.2 Conclusiones Generales 203](#_Toc183157881)

[10. Recomendaciones 204](#_Toc183157882)

[11.  Referencias Bibliográficas 206](#_Toc183157883)

[12. Apéndices 213](#_Toc183157884)

[Apéndice A: Tablas ciclos 213](#_Toc183157885)

[Apéndice B: Colabs 213](#_Toc183157886)

[Apéndice C: Acta de respuesta a Cambio de Título del proyecto 213](#_Toc183157887)

[Apéndice D: Acta de respuesta a Cambio de Título y Objetivos del proyecto 215](#_Toc183157888)

[Apéndice E: Datasets 216](#_Toc183157889)

[Apéndice F: Diagrama de flujo del funcionamiento de los colabs y generación de dataframes: 217](#_Toc183157890)

[Apéndice G: Diagrama de secuencia y de flujo del funcionamiento de la interfaz y su relación con el modelo: 217](#_Toc183157891)

[Apéndice H: Videos de YouTube: 217](#_Toc183157892)

[Apéndice I: Diagrama del Estado del arte con los parámetros y arquitecturas utilizadas: 218](#_Toc183157893)

[Apéndice J: Link interfaz + modelo final: 218](#_Toc183157894)

Lista de Tablas

[Tabla 1. Tabla comparativa redes neuronales clásicas y aprendizaje profundo 44](#_Toc180871446)

[Tabla 2. Comparativa de Arquitecturas de CNN 54](#_Toc180871447)

[Tabla 3. Estado del arte del conjunto de datos seleccionado 71](#_Toc180871448)

[Tabla 4. Estado del arte Clasificación con red neuronal convolucional de arquitectura VGG-19 para la detección y graduación de lesiones en retinopatía diabética basado en aprendizaje profundo. 72](#_Toc180871449)

[Tabla 5. Estado del arte Algoritmos de aprendizaje profundo para la detección y el diagnóstico de enfermedades sistémicas basados en manifestaciones oftálmicas: una revisión sistemática. 73](#_Toc180871450)

[Tabla 6. Estado del arte Clasificación de enfermedades oculares basada en aprendizaje profundo utilizando una red neuronal convolucional para imágenes OCT 74](#_Toc180871451)

[Tabla 7. Estado del arte Comparación entre los marcos de arquitectura VGG16, VGG19 y ResNet50 para la clasificación de imágenes médicas normales y procesadas CLAHE 75](#_Toc180871452)

[Tabla 8. Estado del arte Una solución novedosa del uso del aprendizaje profundo para la clasificación de glóbulos blancos: Función de pérdida mejorada con regularización y pérdida ponderada. 76](#_Toc180871453)

[Tabla 9, Cantidad de imágenes fondo de ojo 78](#_Toc180871454)

[Tabla 10. Descripción dataset 88](#_Toc180871455)

[Tabla 11. Comparativa Hiperparámetros por ciclo 98](#_Toc180871456)

[Tabla 12. Hiperparámetros ciclo 1 108](#_Toc180871457)

[Tabla 13. Varianza general VGG19 ciclo 1 110](#_Toc180871458)

[Tabla 14. Resumen general VGG19 ciclo 1 110](#_Toc180871459)

[Tabla 15. Varianza general Resnet50 ciclo 1 112](#_Toc180871460)

[Tabla 16. Resumen general Resnet50 ciclo 1 112](#_Toc180871461)

[Tabla 17. Varianza general Efficientnet ciclo 1 114](#_Toc180871462)

[Tabla 18. Resumen general Efficientnet ciclo 1 114](#_Toc180871463)

[Tabla 19. Varianza general Xception ciclo 1 116](#_Toc180871464)

[Tabla 20. Resumen general Xception ciclo 1 116](#_Toc180871465)

[Tabla 21. Varianza general InceptionV3 ciclo 1 118](#_Toc180871466)

[Tabla 22. Resumen general InceptionV3 ciclo 1 118](#_Toc180871467)

[Tabla 23. Tabla de métricas Resnet 50 ciclo 1 121](#_Toc180871468)

[Tabla 24. Tabla de métricas VGG19 ciclo1 122](#_Toc180871469)

[Tabla 25. Tabla de métricas Efficientnet ciclo 1 122](#_Toc180871470)

[Tabla 26. Configuración hiperparámetros ciclo 2 129](#_Toc180871471)

[Tabla 27. Tabla repeticiones VGG19 ciclo 2 131](#_Toc180871472)

[Tabla 28, Resumen general VGG19 ciclo 2 131](#_Toc180871473)

[Tabla 29. Resumen Resultados 0.0001 132](#_Toc180871474)

[Tabla 30. Varianza general Resnet50 ciclo 2 134](#_Toc180871475)

[Tabla 31. Resumen general Resnet50 ciclo 2 135](#_Toc180871476)

[Tabla 32. Tabla repeticiones Resnet50 ciclo 2 135](#_Toc180871477)

[Tabla 33. Varianza general Efficientnet ciclo 2 137](#_Toc180871478)

[Tabla 34. Resumen general Efficientnet ciclo 2 138](#_Toc180871479)

[Tabla 35. Repeticiones Efficientnet ciclo 2 138](#_Toc180871480)

[Tabla 36. Tabla de métricas Resnet50 ciclo 2 144](#_Toc180871481)

[Tabla 37. Tabla de métricas EfficientnetB3 ciclo 2 145](#_Toc180871482)

[Tabla 38. Tablas de métricas VGG19 ciclo 2 146](#_Toc180871483)

[Tabla 39. Modelo exactitud general 147](#_Toc180871484)

[Tabla 40. Precisión por Clase ciclo 2 148](#_Toc180871485)

[Tabla 41. Recall por Clase ciclo 2 149](#_Toc180871486)

[Tabla 42. F1-score por Clase ciclo 2 150](#_Toc180871487)

[Tabla 43. Configuración de hiperparámetros ciclo 3 153](#_Toc180871488)

[Tabla 45. Varianza general Efficientnet ciclo 3 155](#_Toc180871489)

[Tabla 46. Tabla por repeticiones Efficientnet ciclo 3 155](#_Toc180871490)

[Tabla. 47. Tabla de métricas Efficientnet ciclo 3 158](#_Toc180871491)

[Tabla 48. Exactitud general EfficientNet 159](#_Toc180871492)

[Tabla 49.Precisión por clase ciclo 2 vs ciclo 3 Efficientnet 159](#_Toc180871493)

[Tabla 50. Recall por clase ciclo 2 vs ciclo 3 Efficientnet 161](#_Toc180871494)

[Tabla 51.F1-Score por clase ciclo 2 vs ciclo 3 Efficientnet 162](#_Toc180871495)

[Tabla 52.Resultados del oftalmólogo 171](#_Toc180871496)

[Tabla 53.Sugerencias del oftalmólogo 172](#_Toc180871497)

[Tabla 54. Resultados de la médica general 173](#_Toc180871498)

[Tabla 55.Sugerencias de la médica general 174](#_Toc180871499)

[Tabla 56.Tabla de errores encontrados en las pruebas 186](#_Toc180871500)

Lista de Figuras

[Figura 1 Imagen comparativa redes neuronales clásicas (izquierda) y redes neuronales profundas (derecha). 39](#_Toc183151328)

[Figura 2. Red neuronal de 3 capas con un esquema de 5 nodos. 47](#_Toc183151329)

[Figura 3 Ejemplo de clasificación según probabilidad, con el uso de imágenes médicas (TAC cerebral) y la función de activación SOFTMAX 49](#_Toc183151330)

[Figura 4. Proceso de extracción de datos y procesamiento convolucional de una CNN 51](#_Toc183151331)

[Figura 5. Tablero Kanban 57](#_Toc183151332)

[Figura 6. Fases de la Metodología de Prototipado Rápido (MPR) 58](#_Toc183151333)

[Figura 7. Fases del Teorema Convolución 67](#_Toc183151334)

[Figura 8, Ejemplos de imágenes de fondo de ojos: Glaucoma, Retinopatía diabética, Catarata y Normal respectivamente 79](#_Toc183151335)

[Figura 9. Estructuras Anatómicas 80](#_Toc183151336)

[Figura 10. Diagrama de flujo para el procesamiento del dataset 91](#_Toc183151337)

[Figura 11. Diagrama de flujo para el entrenamiento y validación del modelo 92](#_Toc183151338)

[Figura 12. Aclaraciones de los diagramas de flujo 93](#_Toc183151339)

[Figura 13. Diagrama de flujo del funcionamiento de la interfaz 100](#_Toc183151340)

[Figura 14. Diagrama de secuencia del proceso de clasificación de imágenes y su relación entre interfaz-modelo 101](#_Toc183151341)

[Figura 15. Matriz de Confusión de Resnet50 ciclo 1 125](#_Toc183151342)

[Figura 16. Matriz de confusión de VGG19 ciclo 1 126](#_Toc183151343)

[Figura 17. Matriz de confusión EfficientnetB3 ciclo 1 127](#_Toc183151344)

[Figura 18. Gráfica pérdida y accuracy en validación vs pérdida y accuracy en entrenamiento de la mejor repetición VGG19 132](#_Toc183151345)

[Figura 19. Gráfica pérdida y accuracy en validación vs pérdida y accuracy en entrenamiento de la mejor repetición Resnet50 136](#_Toc183151346)

[Figura 20. Gráfica pérdida y accuracy en validación vs pérdida y accuracy en entrenamiento de la mejor repetición Efficientnet ciclo 2 139](#_Toc183151347)

[Figura 21. Matriz de confusión efficientnetB3 ciclo 2 141](#_Toc183151348)

[Figura 22. Matriz de confusión VGG19 ciclo 2 142](#_Toc183151349)

[Figura 23. Matriz de confusión Resnet50 ciclo 2 143](#_Toc183151350)

[Figura 24. Gráfica pérdida y accuracy en validación vs pérdida y accuracy en entrenamiento de la mejor repetición Efficientnet ciclo 3 156](#_Toc183151351)

[Figura. 25. Matriz de confusión de EfficientNet ciclo 3 157](#_Toc183151352)

[Figura 26. Matriz de confusión prueba interfaz 164](#_Toc183151353)

[Figura 27. Interfaz de usuario final del prototipo 167](#_Toc183151354)

[Figura 28. Predicción correcta de imagen borrosa de catarata en las pruebas con el oftalmólogo 175](#_Toc183151355)

[Figura 29. Predicción incorrecta de imagen borrosa de glaucoma en las pruebas con la médica general 176](#_Toc183151356)

[Figura 30. Predicción incorrecta de imagen borrosa de glaucoma en las pruebas con el oftalmólogo 177](#_Toc183151357)

[Figura 31. Predicción correcta de imagen borrosa de retinopatía diabética en las pruebas con la médica general 178](#_Toc183151358)

[Figura 32. Predicción correcta de imagen de catarata en las pruebas con el oftamólogo 179](#_Toc183151359)

[Figura 33. Predicción correcta de imagen de retinopatía diabética en las pruebas con la médica general 180](#_Toc183151360)

[Figura 34. Predicción correcta de imagen de ojo normal en las pruebas con el oftalmólogo 181](#_Toc183151361)

[Figura 35. Predicción correcta de imagen de ojo normal en las pruebas con la médica general 182](#_Toc183151362)

[Figura 36. Matriz de confusión usando imágenes de retinoscopía, evidenciada en las pruebas con ambos doctores 183](#_Toc183151363)

[Figura 37. Matriz de confusión usando imágenes de retinoscopía borrosas, evidenciada en las pruebas con la médica general 184](#_Toc183151364)

[Figura 38. Matriz de confusión usando imágenes de retinoscopía borrosas, evidenciada en las pruebas con el oftalmólogo 185](#_Toc183151365)

[Figura 39. Matriz de confusión base de datos pública 189](#_Toc183151366)

[Figura 40. Diagrama del Estado del Arte 194](#_Toc183151367)

**Glosario**

**APRENDIZAJE PROFUNDO:** Técnica de inteligencia artificial que imita el funcionamiento del cerebro humano mediante redes neuronales artificiales para procesar datos y generar patrones (Kufel, 2023).

**REDES NEURONALES CONVOLUCIONALES (CNN):** Tipo de red neuronal diseñada para el procesamiento de datos estructurados en forma de cuadrículas, como imágenes (Liu, 2021).

**TRANSFERENCIA DE APRENDIZAJE:** Método que utiliza modelos preentrenados en una tarea específica y los adapta para resolver otras tareas (LeCun, 2015).

**RETINOSCOPÍA**: Técnica médica utilizada para examinar el fondo del ojo mediante el uso de un dispositivo especializado (Matías Crespo, 2007).

**CLASIFICACIÓN DE IMÁGENES:** Proceso mediante el cual una máquina o algoritmo asigna una etiqueta o categoría a una imagen (Matías Crespo, 2007).

**GLAUCOMA**: Enfermedad ocular que daña el nervio óptico, a menudo causada por presión intraocular elevada (Matías Crespo, 2007).

**CATARATAS**: Opacidad del cristalino del ojo que conduce a la disminución de la visión (Matías Crespo, 2007).

**RETINOPATÍA DIABÉTICA:** Daño a los vasos sanguíneos de la retina en personas con diabetes, lo que puede llevar a la ceguera (Matías Crespo, 2007).

**RESNET50:** Modelo de red neuronal profunda basado en la arquitectura de redes residuales (Ortiz, 2023).

**VGG19**: Arquitectura de red neuronal profunda que se utiliza para la clasificación de imágenes complejas (Mascarenhas, 2021).

Introducción

Las discapacidades visuales, como la retinopatía diabética, las cataratas y el glaucoma, representan un desafío creciente tanto a nivel global como nacional; en Bucaramanga, Colombia, las estadísticas muestran un aumento de estas enfermedades (Cuidarte, n.d) ,denotando la necesidad urgente de soluciones innovadoras; esto es fundamental para evitar la progresión de estas patologías.

Sin embargo, el sistema actual de diagnóstico enfrenta múltiples obstáculos en términos de acceso y precisión (Salud, n.d.). Este proyecto busca abordar estas limitaciones mediante el uso de redes neuronales convolucionales (CNN), las cuales pueden apoyar el diagnóstico de enfermedades oculares. A través de este enfoque, se pretende optimizar la clasificación de imágenes relacionadas con estos padecimientos.

Con la implementación de arquitecturas avanzadas de aprendizaje profundo se espera encontrar una alta precisión en la detección y clasificación de enfermedades oculares, y así contribuir de manera significativa a la investigación en inteligencia artificial aplicada a la medicina visual.

1. Problemática a Investigar

Se centró en la relevancia de las enfermedades visuales como la retinopatía diabética, cataratas y glaucoma. En este contexto, se determinó a partir de la problemática, que existe la necesidad urgente de soluciones tecnológicas, por tal motivo se propone el uso de aprendizaje profundo para la detección y clasificación de estas patologías a partir de imágenes de retinoscopía.

1.1. Planteamiento del problema

Las discapacidades visuales que causan ceguera irreversible, tales como la retinopatía diabética, las cataratas y glaucoma, tienen una alta relevancia a nivel mundial. En 2021, la OMS estimaba que 1.300 millones de personas sufrían discapacidad visual en todo el mundo (Salud, n.d.) (Gulshan, 2016). Este panorama representa un importante desafío a nivel local y regional, ya que evidencia la necesidad urgente de soluciones eficaces para abordar y revertir la creciente prevalencia de estas patologías en la población (Gulshan, 2016).

Colombia estima que alrededor de 4000 personas por cada millón de la población requiere servicios de baja visión en donde el glaucoma contribuye a un 23% de los casos de baja visión, retinopatía diabética un 19% etc. Bucaramanga, Colombia, las estadísticas revelan un aumento de enfermedades oculares irreversibles como el glaucoma y la retinopatía, lo que supone un reto importante y evidencia la necesidad urgente de desarrollar soluciones innovadoras para abordar este problema de salud pública, al igual que en otras regiones del mundo. (Cuidarte, n.d)

La detección temprana de enfermedades oculares es crucial para prevenir complicaciones graves y evitar la pérdida de la visión, ya sea de manera total o parcial (Litjens, 2017). No obstante, la infraestructura actual de diagnóstico oftalmológico enfrenta limitaciones en términos de eficacia y accesibilidad. (Cuidarte, n.d)

Lo anterior se traduce en dificultades para realizar diagnósticos precisos de manera oportuna, afectando negativamente el tratamiento y la calidad de vida de los pacientes (Litjens, 2017). Por lo tanto, es importante desarrollar un artefacto para la clasificación de imágenes de retinoscopía.

Este enfoque da lugar a la formulación de la siguiente pregunta de investigación: *¿Cuál arquitectura de Aprendizaje profundo permite una correcta clasificación de las enfermedades oculares que causan ceguera irreversible a partir de imágenes de retinoscopía?*

Solución Propuesta

En este proyecto se propone abordar esta necesidad mediante el aprovechamiento de tecnologías avanzadas, las cuales incluyen el uso de redes neuronales convolucionales para clasificar enfermedades oculares basadas en imágenes de fondo de ojo en color.

Se implementaron técnicas de aprendizaje profundo por transferencia, adaptando modelos preentrenados a nuevas tareas (variación paramétrica de los modelos VGG19, ResNet50, EfficientNet, InceptionV3 y Xception), para así acelerar el entrenamiento y obtener resultados óptimos en menor tiempo. (Archana, 2023). Se usó el conjunto de datos Eyes Disease Classification de Guna Venkat Doddi, el cual consta de imágenes de retina normales, de retinopatía diabética, de cataratas y de glaucoma; cada clase (tipo de clasificación) tiene aproximadamente 1.000 imágenes diagnósticas (retinoscopía).

2. Objetivo del proyecto

En este apartado se presentan los respectivos objetivos específicos y general actuales del proyecto; si desea

2.1 Objetivo General

Implementar un modelo de clasificación de patologías oculares que causan ceguera irreversible (retinopatía diabética, cataratas y glaucoma) usando arquitecturas de Aprendizaje profundo.

2.2 Objetivo Específico

Seleccionar un conjunto de datos (dataset) representativos de imágenes de fondo de ojo que contenga las características morfológicas de enfermedades (retinopatía diabética, cataratas y glaucoma), así como de tejido sano, generando un conjunto de datos útil.

Revisar el estado del arte buscando modelos de Aprendizaje profundo probados en imágenes de tejidos similares a fondos de ojos, tal que puedan detectar, procesar y clasificar imágenes de retinoscopía.

Escoger un modelo de aprendizaje profundo para la clasificación de patologías oculares que causan ceguera irreversible a través de pruebas exhaustivas, usando el conjunto de imágenes seleccionado.

Integrar el modelo de Aprendizaje profundo seleccionado mediante una interfaz tipo Python notebook.

3. Justificación.

Los métodos actuales de diagnóstico de enfermedades oculares se enfrentan a varios retos que repercuten en la salud de los pacientes; los retrasos en el diagnóstico debido a la accesibilidad limitada a oftalmólogos especializados, los largos tiempos de espera para las citas y los procedimientos de cribado lentos pueden provocar la progresión de la enfermedad y la pérdida irreversible de visión son algunos de los problemas más prevalentes en la actualidad, a nivel global. (Al-Khafaji, 2023)

La falta de precisión en el diagnóstico debido a errores humanos o a conocimientos limitados también puede dar lugar a un tratamiento tardío o inadecuado, en consecuencia, se agrava aún más el problema. Además, los procedimientos de detección tradicionales pueden ser costosos, ya que requieren gastos de viaje, consultas a especialistas y estancias hospitalarias. (Cuidarte, n.d). Esto significa que los retrasos en el diagnóstico de las enfermedades oculares se deben a factores como las complicaciones geográficas al acceso a oftalmólogos especializados, que dan lugar a prolongados periodos de espera para las citas, sobre todo si el paciente reside en sectores rurales.

Una vez que los pacientes reciben una remisión, a menudo tienen que esperar más tiempo antes de someterse a las revisiones, esto aumenta el riesgo de que su vista sufra daños progresivos (Cuidarte, n.d).

Realizar el proyecto es importante debido a su potencial para mejorar significativamente la capacidad de clasificación de enfermedades oculares que causan ceguera irreversible, a partir de imágenes de retinoscopía. Además, el presente proyecto contribuirá al avance de la investigación en el campo de la inteligencia artificial aplicada a la medicina, lo que en consecuencia puede abrir nuevas oportunidades para la colaboración interdisciplinaria y el desarrollo de soluciones innovadoras en el ámbito de la salud visual.

En cuanto a la viabilidad económica, el proyecto propuesto es rentable, ya que el total de los recursos necesarios, como modelos, bibliotecas, entornos de programación y conjuntos de datos, son de libre acceso. Esto hace que el proyecto sea accesible a un amplio abanico de partes interesadas, incluidos profesionales sanitarios, investigadores, e incluso pacientes.

Las tecnologías propuestas para este proyecto incluyen Google Colab, Python y TensorFlow, las cuales son ampliamente utilizadas y están bien establecidas en el campo de la inteligencia artificial. La disponibilidad de estas herramientas y recursos hacen que el proyecto sea técnicamente viable y accesible para estudiantes e investigadores con distintos niveles de experiencia.

3.1 Impacto Esperado

Se espera que el proyecto tenga un impacto positivo en la comunidad, pues los resultados de esta investigación tendrán importantes implicaciones sociales, en consecuencia, a futuro se evidenciará un beneficio a los investigadores del área de aprendizaje profundo que deseen abarcar la clasificación de imágenes.

Adicionalmente, se estima cumplir con los requisitos académicos establecidos para aprobar las materias de Proyecto de Grado 1 y 2, así como en la ambición de alcanzar los objetivos propuestos para el proyecto, así se permitirá demostrar, por parte de los estudiantes implicados, su dominio de los conocimientos adquiridos y aplicarlos de manera práctica, consolidando así sus habilidades y competencias para su formación académica y profesional.

4. Metodología

4.1 Tipos de Investigación

El presente proyecto se fundamenta en un enfoque cuantitativo y correlacional para alcanzar sus objetivos. El enfoque cuantitativo permitirá una evaluación precisa de la efectividad de los modelos de aprendizaje profundo utilizados en la clasificación de imágenes de fondo de ojo y la detección de enfermedades oculares. Por otro lado, el enfoque correlacional se empleó para analizar las relaciones entre las diferentes variables del estudio, tales como las características de las imágenes de retinoscopía y la precisión de la clasificación.

4.2 Caracterización de Imágenes

Las imágenes de retinoscopía, o de fondo de ojo, se caracterizan por estar en formato .jpg o .jpeg; más específicamente las jpg (retinas normales, cataratas y glaucoma), y en jpeg (retinopatía diabética). Además, el tamaño de estas imágenes de fondo de ojo no es relevante, ya que los modelos escogidos interpretan los datos según se dictamina en el Teorema de la Convolución, del cual las redes neuronales convolucionales se basan; esto hace que sean flexibles, resistentes a ruido, a transformaciones, rotaciones y deformaciones (Cruz et al., 2021). Se realiza también un proceso de eliminación de ruido, más específicamente en 78 imágenes de cataratas; la cantidad de imágenes en total que se utilizarán a partir del conjunto de datos son 4217 imágenes médicas, en donde quedaron respectivamente en cataratas 1038, en retinopatía diabética 1098, glaucoma 1007 e imágenes normales 1074.

El conjunto de datos ideal para el artefacto deberá incluir imágenes de fondo de ojo de las morfologías clave (retinopatía diabética, glaucoma y catarata); es decir, que debe ser representativo de las diversas afecciones y sus manifestaciones en la retina. Las imágenes deben ser de alta calidad, en alguno de los formatos aceptados (JPG o JPEG) y con una visibilidad clara de las siguientes estructuras anatómicas:

* Mácula
* Fóvea
* Papila del nervio óptico
* Arterias y venas retinianas
* Excavación fisiológica

El conjunto de datos debe incluir también imágenes de ojos sanos que sirvan de grupo de control para la comparación.

4.3 Hipótesis

Desarrollo basado en un artefacto de Aprendizaje profundo que permite una correcta clasificación de las enfermedades oculares que causan ceguera irreversible a partir de imágenes de retinoscopía.

4.4 Método científico

El método científico abarca todo el proceso que se utilizó en el proyecto, esto se traduce en una documentación limpia de los resultados, para así minimizar los sesgos en la investigación. Este método implica la formulación de una pregunta investigativa que abarca la hipótesis, el diseño de experimentos para probar la hipótesis, la recogida y análisis de datos y la extracción de conclusiones basadas en los resultados.

4.5 Metodología de Prototipado Rápido (MPR)

Para el desarrollo del modelo, se ha seleccionado la Metodología de Prototipado Rápido (MPR). Esta metodología se caracteriza por su enfoque iterativo mediante ciclos de desarrollo ágiles y continuos; con esta técnica se adaptará el modelo a medida que surjan nuevas necesidades y se recopilen datos adicionales. Cabe destacar que la MPR fomenta el uso de herramientas visuales, esto ayuda a garantizar un desarrollo eficiente y efectivo del artefacto.

Durante el proceso de desarrollo, se integrarán modelos de aprendizaje profundo como VGG19, ResNet50, Xception, EfficientNet e InceptionV3 en procesos de entrenamiento, validación y pruebas; se cuenta con la variación paramétrica de la tasa de aprendizaje (learning rate; , ,) y métricas establecidas(precisión o accuracy, val\_loss o función de pérdida en modo validación), así como un umbral de precisión mínima, el cuál es 0.95; también se cuenta con parámetros fijos como lo son el tamaño del lote (batch size; 16, 32), el número de épocas (epoch; 40, 60 y 100), la función de pérdida Validación Cruzada (Cross-Entropy Loss) y la función de activación Softmax. Se puede evidenciar entonces, que la combinación del enfoque cuantitativo, correlacional con la metodología MPR, el método científico y Kanban proporciona un marco sólido para el desarrollo del artefacto.

4.6 Metodología Kanban

Para verificar estos modelos se utiliza la metodología Kanban, basada en tareas divididas entre los participantes en el proyecto. Con este enfoque se garantiza que cada tarea se complete antes de pasar a la siguiente, permitiendo así un proceso de desarrollo estructurado y organizado. El tablero Kanban se utiliza para visualizar el flujo de trabajo, seguir el progreso e identificar posibles cuellos de botella; las tareas se priorizarán en función de su urgencia e importancia, garantizando que las más críticas se aborden en primer lugar. Se realizarán reuniones entre los miembros del equipo para debatir los avances, abordar cualquier problema que pueda surgir y realizar ajustes necesarios en el flujo de trabajo.

4.7 Incorporación de las Metodologías (MPR, Kanban, Científica)

El método científico se utiliza para llevar a cabo el proyecto de forma general, desde su planteamiento hasta las conclusiones y la documentación de los resultados. Para llevar a cabo el proyecto de forma más específica se utiliza la metodología de prototipado rápido (RPM) mediante ciclos de desarrollo ágiles y continuos, el modelo se adapta a medida que surgen nuevas necesidades y se recopilan datos adicionales. El RPM se utiliza para realizar las siguientes fases.

En la fase 2 Pruebas de clasificación de modelos CNN Selección del modelo final y la fase 3 en su totalidad. La metodología Kanban se utiliza para organizar las actividades y los roles encargados de las tareas necesarias que deben realizarse con el prototipado rápido.

A partir de lo anterior se proponen las siguientes fases

Fase 1: Recopilar Datos

* Revisión de conjunto de datos en imágenes de retinoscopía
* Selección del conjunto de datos en imágenes de retinoscopía
* Limpieza del conjunto de datos en imágenes

Fase 2: Escoger Modelos CNN

* Búsqueda de modelos de aprendizaje profundo a partir del estado del arte
* Selección de los modelos de aprendizaje profundo
* Pruebas de clasificación de modelos CNN
* Selección del modelo final

Fase 3: Integración del Modelo CNN

* Diseño de la interfaz
* Codificación de la interfaz
* Integración del modelo CNN con la interfaz
* Pruebas de integración del artefacto

Fase 4: Escritura

* Manual para reentrenamiento de la red neuronal
* Manual de usuario
* Libro final
* Artículo Científico

4.8 Método de Investigación

Fase 1: Recopilar Datos

Revisión de conjunto de datos en imágenes de retinoscopía

Se realizó la revisión de diversos conjuntos de datos (dataset) disponibles que contengan imágenes de retinoscopía o de fondo de ojo, relacionadas con patologías oculares; también se identificaron los conjuntos de datos más relevantes y adecuados para el proyecto.

Selección del conjunto de datos en imágenes de retinoscopía

Tras la revisión, se seleccionó el conjunto de datos que contiene imágenes específicas de retinopatías diabéticas, cataratas y glaucoma, así como casos sin enfermedades oculares (es decir, normales). Se consideraron criterios como la diversidad de casos y la calidad de las imágenes.

Limpieza del conjunto de datos en imágenes

Se realizó un proceso de limpieza y preprocesamiento del conjunto de datos seleccionado de imágenes de retinoscopía, para garantizar la consistencia y la calidad de las imágenes. Esto incluyó la eliminación de datos irrelevantes, la corrección de posibles errores, la normalización de formatos, la división del conjunto de datos en conjuntos de entrenamiento y prueba, y la extracción de características relevantes.

Fase 2: Selecionar Modelos CNN

Búsqueda de modelos de aprendizaje profundo a partir del estado del arte

Se investigaron y se evaluaron diferentes modelos de CNN a partir del estado del arte creado, adecuados para la clasificación de imágenes de retinoscopía, en donde se escogieron los modelos VGG19, ResNet50, Xception, EfficientNet e InceptionV3.

Pruebas de clasificación de modelos CNN

Se realizaron pruebas exhaustivas de clasificación utilizando los modelos de aprendizaje profundo seleccionados. Se evaluó la precisión y el rendimiento de cada modelo en la clasificación de imágenes de fondo de ojo en relación con las diferentes patologías oculares.

Selección del modelo final

Basándose en los resultados de las pruebas, se seleccionó el modelo de aprendizaje profundo que mostró el mejor rendimiento y precisión en la clasificación de imágenes de fondo de ojo para su implementación final en una interfaz.

Fase 3: Integración del Modelo CNN

Diseño de la interfaz

Se diseñó la interfaz, considerando la facilidad de uso y la presentación de los resultados de clasificación de manera clara para el usuario final.

Codificación de la interfaz

Se realizó la codificación de la interfaz utilizando tecnologías y herramientas adecuadas para su desarrollo.

Integración del modelo CNN con la interfaz

Se integró el modelo de aprendizaje profundo seleccionado previamente con la interfaz, permitiendo la clasificación automática de imágenes de fondo de ojo.

Pruebas de integración del artefacto

Se llevaron a cabo pruebas exhaustivas de integración para garantizar el correcto funcionamiento entre la interfaz y el modelo de aprendizaje profundo integrado.

Fase 4: Escritura

Manual para reentrenamiento de la red neuronal

Se elaboró un manual para el reentrenamiento de la red neuronal, que describió la arquitectura, el funcionamiento y la implementación del artefacto, además del paso a seguir para cambiar los parámetros y poder realizar variaciones de la red neuronal.

Manual de usuario

Se creó un manual de usuario para proporcionar instrucciones claras y concisas sobre cómo utilizar el artefacto; incluyendo la navegación por la interfaz, la carga de imágenes de retinoscopía y la interpretación de los resultados de clasificación.

Libro final

Se preparó el documento final del proyecto de grado, que incluyó una introducción, antecedentes, metodología, resultados, conclusiones y referencias bibliográficas, así como cualquier otro contenido requerido por las normativas institucionales.

Artículo Científico

Se redactó un artículo científico basado en los hallazgos y resultados del proyecto, con el objetivo de contribuir al conocimiento científico en el campo de la aplicación de modelos de aprendizaje profundo en la clasificación de patologías oculares.

5. Marco Referencial

5.1 Marco Teórico.

5.1.1 Aprendizaje Automático

El aprendizaje automático (Machine Learning, ML) abarca el estudio de algoritmos que mejoran automática y progresivamente a través de la experiencia; a diferencia de las tareas sencillas, en las que los algoritmos pueden programarse explícitamente para guiar a las máquinas por todos los pasos necesarios, las tareas más complejas pueden suponer un reto para los humanos a la hora de crear manualmente los algoritmos necesarios. (Li, 2023) En estos casos, puede ser más eficaz facilitar a una máquina el desarrollo de su propio algoritmo que depender únicamente de programadores humanos para definir cada paso. (Sunchalin, 2021)

Las técnicas de aprendizaje automático emplean varios enfoques para ayudar a los ordenadores a aprender a realizar tareas para las que no existe un algoritmo completamente satisfactorio. (Li, 2023)

Estos enfoques pueden clasificarse en tres grandes grupos:

Aprendizaje supervisado: Se enseña al ordenador mediante ejemplos de entradas y sus resultados deseados proporcionados por un "profesor", con el objetivo de aprender una regla general que relacione la entrada con la salida; este tipo de aprendizaje se utiliza a menudo para tareas como la clasificación, la regresión y la detección de objetos (Sunchalin, 2021)

Aprendizaje no supervisado: Los algoritmos se entrenan con datos que no tienen etiquetas ni valores de salida correspondientes; el aprendizaje no supervisado suele utilizarse cuando se desconoce el resultado deseado o cuando resulta demasiado caro o poco práctico etiquetar los datos (Sunchalin, 2021)

Aprendizaje por refuerzo: Un programa informático interactúa con un entorno dinámico en el que debe cumplir un propósito específico, recibiendo una retroalimentación similar a las recompensas que intenta maximizar a medida que navega por el espacio del problema. (Kaelbling, 1996) Los algoritmos de RL se basan en el proceso de decisión de Markov y utilizan el aprendizaje por ensayo y error para descubrir las mejores acciones para un estado determinado; este tipo de aprendizaje es especialmente útil en entornos complejos donde es difícil definir un conjunto de reglas o donde el entorno cambia constantemente. (Li, 2023) También es ventajoso porque requiere menos intervención humana que el aprendizaje supervisado, ya que aprende de sus propias interacciones con el entorno. (Kaelbling, 1996)

5.1.2 Red neuronal artificial

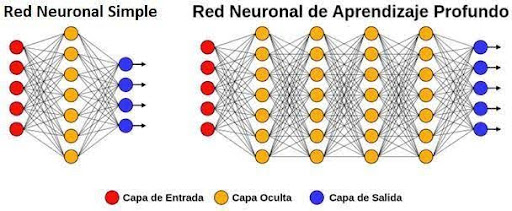
Una red neuronal artificial (Artificial Neural Network, ANN) es un modelo computacional inspirado en la estructura y función del cerebro humano, en particular en las neuronas interconectadas que procesan y transmiten información; las ANN están formadas por capas de nodos interconectados, o neuronas artificiales, diseñadas para aprender y reconocer patrones en los datos. (Pręgowska, 2021)

Las ANN no pretenden necesariamente replicar el funcionamiento óptimo del cerebro, pero pueden diseñarse para captar los errores y la capacidad de generalización que son similares al rendimiento humano. (Kar, 2022) Este enfoque puede ayudar a mejorar el rendimiento y la robustez de las ANN. Cabe destacar que su interpretabilidad, o explicabilidad, puede ser un reto, ya que a menudo carecen de la transparencia de los algoritmos tradicionales; hoy en día, los investigadores están explorando formas de alinear los componentes de las ANN con constructos neurocientíficos para mejorar su interpretabilidad. (Kar, 2022)

5.1.3 Aprendizaje profundo

El aprendizaje profundo (Deep Learning, DL), es un subconjunto del aprendizaje automático, la cual ha ganado cada vez más popularidad debido a los avances en la potencia de procesamiento y la capacidad de las ANN, también conocidas como redes neuronales profundas (Deep Neural Network, DNN); esto significa que aprendizaje profundo utiliza las DNN para simular el proceso de aprendizaje del cerebro humano y extraer características de datos a gran escala de forma no supervisada o supervisada, dependiendo del contexto, el modelo y la arquitectura usada. (Pal, 2022) Este tipo de implementación de aprendizaje automático se utiliza para automatizar tareas que normalmente requerirían inteligencia humana, como describir imágenes o transcribir archivos de audio. Los modelos de aprendizaje profundo tienen múltiples capas de nodos interconectados, y cada capa se basa en la anterior para refinar y optimizar las predicciones y clasificaciones. (Liu, 2021)

Figura 1 Imagen comparativa redes neuronales clásicas (izquierda) y redes neuronales profundas (derecha).



Nota. Imagen Tomada (Mitaritonna, 2019)

Los modelos de aprendizaje profundo se pueden entrenar para realizar tareas de clasificación y reconocer patrones en fotos, texto, audio y otros datos diversos. (Pal, 2022) La aparición de las redes neuronales hace posible el procesamiento integral de imágenes. (Liu, 2021) El Aprendizaje profundo impulsa muchas aplicaciones y servicios de inteligencia artificial (Artificial Intelligence, IA) que mejoran la realización de tareas analíticas y físicas sin intervención humana. (Shukla, 2023)

En el campo de la visión artificial, el aprendizaje profundo se utiliza principalmente en la reducción de la dimensionalidad de los datos, el reconocimiento de números manuscritos, el reconocimiento de patrones además; otros campos como el reconocimiento de imágenes, la reparación de imágenes, la segmentación de imágenes, el seguimiento de objetos, el análisis de escenas, etc., mostrando una eficacia muy alta, adquiriendo así cada vez más importancia en la investigación médica, especialmente en el campo de las bioimagenes. (Liu, 2021)

Actualmente, las investigaciones de esta era incluyen modelos que pueden diagnosticar diferentes enfermedades basándose en imágenes, algoritmos de auto-segmentación que pueden dibujar alrededor de las lesiones, modelos que pueden determinar el nivel de riesgo de lesión, y más. (Pal, 2022)

5.1.3.1 Importancia del Aprendizaje profundo

El campo del aprendizaje profundo ha experimentado un notable aumento de protagonismo debido a su profundo impacto en diversas industrias, ya que los algoritmos de aprendizaje profundo poseen la capacidad única de operar de forma autónoma, acelerando así los despliegues tecnológicos y ejecutando tareas complejas sin intervención humana. (Sunchalin, 2021) Además, su habilidad para manejar datos no estructurados establece un nuevo estándar para el procesamiento y análisis de datos; otra de las principales ventajas del aprendizaje profundo es su capacidad para automatizar tareas complejas sin intervención humana, esto significa que las empresas pueden desplegar la tecnología más rápidamente, lo que se traduce en una mayor eficiencia y productividad. (Li, 2023) Cabe destacar que los algoritmos de aprendizaje profundo son muy eficaces en el procesamiento de datos no estructurados, lo cual es una ventaja grande, pues la mayoría de los datos recuperados no están estructurados.

El Aprendizaje profundo puede tener en cuenta la variación entre las funciones de aprendizaje, reduciendo los márgenes de error en todo tipo de industrias, lo que ayuda a reducir los costes una vez que la infraestructura necesaria esté en su lugar; además, la capacidad del aprendizaje profundo para aprender sin supervisión humana impulsa la mejora continua de la precisión y los resultados, lo que permite que la analítica avanzada de datos ofrezca resultados más fiables y concisos. (eLearning, 2023) Por último, el aprendizaje profundo es altamente escalable, esto en consecuencia lo convierte en una solución rentable para procesar cantidades masivas de datos y realizar muchos cálculos; adicionalmente, la escalabilidad de los algoritmos de aprendizaje profundo beneficia directamente a la productividad empresarial al aumentar la fiabilidad de las predicciones. (eLearning, 2023)

Adicionalmente, la capacidad del aprendizaje profundo para aprender automáticamente características de los datos y manejar conjuntos de datos grandes y complejos lo convierte en una herramienta valiosa para diversas aplicaciones, como el reconocimiento de imágenes y otras tareas que implican big data. (eLearning, 2023)

5.1.4 Aplicativos del Aprendizaje profundo

El aprendizaje profundo ha revolucionado el campo de las imágenes médicas al permitir el desarrollo de algoritmos avanzados que pueden ayudar en la clasificación, la síntesis de imágenes y la detección de enfermedades. Varios estudios han puesto de manifiesto la potencia de los modelos de aprendizaje profundo, en particular las redes neuronales convolucionales (CNN), en diversas tareas de imagen médicas.

Estudios que utilizan Aprendizaje profundo en el tratamiento de imágenes médicas:

Síntesis de imágenes médicas con CNN: Se destaca un estudio centrado en la generación de imágenes de tomografía computarizada (TC) a partir de imágenes de resonancia magnética (IRM) mediante un enfoque basado en una “Generative Adversarial Network” (GAN). Mediante el empleo de una estructura de red totalmente convolucional más conocida como “Fully Convolutional Network” (FCN) y una estrategia de entrenamiento adversarial, el modelo produce imágenes de TC realistas a partir de datos de RM; además, el modelo propone una nueva función de pérdida basada en las diferencias de gradiente de la imagen para mejorar la nitidez de las imágenes generadas (Nie, 2017)

CheXNet para la detección de neumonía: Desarrollo de CheXNet, un algoritmo CNN de 121 capas diseñado para detectar neumonías en radiografías de tórax. El modelo es entrenado con el conjunto de datos ChestX-ray14, el cual contiene más de 100.000 imágenes de rayos X frontales con 14 enfermedades diferentes; CheXNet casi iguala en precisión a los radiólogos en la detección de neumonía. El algoritmo utiliza mapas de activación de clases (CAM) para visualizar las zonas de las imágenes relacionadas con la enfermedad y obtiene los mejores resultados en las 14 clases de patologías torácicas (Rajpurkar, 2017)

ResNet-18 para la detección de COVID-19: En este estudio se utiliza un modelo CNN de 18 capas basado en la arquitectura ResNet-18 para la detección de COVID-19 en imágenes de rayos X. Entrenado con 603 imágenes de alta resolución divididas en conjuntos de entrenamiento, prueba y validación, el modelo incorpora técnicas de regularización como dropout para evitar el sobreajuste; aprovechando la arquitectura ResNet-18 preentrenada de las bibliotecas Keras y Tensorflow, el modelo logra una precisión alta (94,73%) en la clasificación de casos de COVID-19 sin cálculos que consuman muchos recursos. (Gov.ar, s.f.)

Con los estudios anteriores se puede ejemplificar las diversas aplicaciones del aprendizaje profundo en imágenes médicas, mostrando su potencial para revolucionar los procesos de clasificación

5.1.5 Diferencia entre Aprendizaje profundo y Aprendizaje automático

Según el número de redes neuronales se puede distinguir entre ANN, que son redes neuronales que constan de una única capa oculta, y DNN, que son redes neuronales. redes que tienen múltiples capas ocultas (como se muestra en la Ilustración 1). Esto permite que la red comprenda e imite comportamientos más complejos y abstractos. (Kufel, 2023)

El Aprendizaje Profundo se considera una evolución del aprendizaje automático (Aprendizaje automático), siendo la principal diferencia la capacidad de acción o nivel de autonomía de la tecnología. En el caso del Aprendizaje automático, el sistema puede mejorar progresivamente sus funciones, pero aún necesita cierta orientación; si un algoritmo de inteligencia artificial devuelve una predicción inexacta, entonces un ingeniero debe intervenir y hacer ajustes. (Wolansky, 2021) En cambio, con un modelo de Aprendizaje profundo, un algoritmo puede determinar por sí mismo si una predicción es precisa o no a través de su propia red neuronal; es decir, puede aprender y mejorar a través de su propio método de computación, sin necesidad de intervención humana. (Kufel, 2023)

5.1.5.1 Diferencia entre redes neuronales clásicas y aprendizaje profundo

En esta sección se presenta un cuadro comparativo que señala las diferencias más relevantes entre estos dos conceptos

**Tabla 1.**

Tabla comparativa redes neuronales clásicas y aprendizaje profundo

|  |  |  |
| --- | --- | --- |
| **Característica** | **Redes Neuronales Clásicas** | **Aprendizaje Profundo** |
| Profundidad de la arquitectura | Poca profundidad (generalmente una o dos capas ocultas) | Arquitecturas profundas, múltiples capas ocultas |
| Representación de características | Características predefinidas o diseñadas manualmente | Características aprendidas automáticamente |
| Tamaño del conjunto de datos | Requiere grandes conjuntos de datos para aprender características efectivas | Puede beneficiarse de grandes conjuntos de datos, pero también puede funcionar con cantidades moderadas de datos gracias a la capacidad de aprendizaje jerárquico |
| Extracción de características | Manualmente diseñada o seleccionada a través de métodos heurísticos | Aprende a través del proceso de entrenamiento |
| Tareas comunes | Tareas específicas y limitadas, como clasificación o regresión | Puede abordar una amplia gama de tareas, incluyendo reconocimiento de imágenes, procesamiento de lenguaje natural, y más |
| Capacidad de generalización | Puede tener dificultades para generalizar a nuevos datos debido a que presenta problemas en su capacidad de aprender características complejas y abstractas | Tiene una gran capacidad para aprender características jerárquicas y abstractas y así facilitar la generalización a nuevos datos |
| Requisitos computacionales | Menos exigentes en términos de recursos computacionales | Más exigentes en términos de recursos computacionales, especialmente con arquitecturas profundas y grandes conjuntos de datos |
| Entrenamiento | Requiere en algunas ocasiones ajustes manuales de hiperparámetros y diseño de características | El entrenamiento puede ser más automatizado, pero a veces necesita ajustes de hiperparámetros y selección de arquitectura |
| Ejemplos de algoritmos | Redes Neuronales de una o dos capas | Redes Neuronales Convolucionales (CNN), Redes Neuronales Recurrentes (RNN), Redes Neuronales Generativas Adversariales (GAN), etc. |

Nota. Información tomada de (Kufel, 2023)

5.1.6 Funcionamiento del Aprendizaje profundo

El procesamiento en una red neuronal se describe en muchas ocasiones como un modelo de caja negra, en dónde se le da un valor de entrada y, sin saber qué procesos pasaron dentro de dicha caja, sale un valor de salida correspondiente; sin embargo, esto no aplica del todo para describir dicho funcionamiento. En las redes neuronales puede haber cientos o miles de capas ocultas con grandes cantidades de nodos (Cada neurona puede considerarse una pequeña unidad de procesamiento de información), cada una de estas capas tendrá su propia jerarquía. (Li, 2023)

Aunque los resultados de los modelos de aprendizaje profundo, como la clasificación cáncer/no cáncer y diferentes niveles de riesgo son comprensibles, su funcionamiento interno puede ser difícil de interpretar. A pesar de ello, es posible examinar los procesos matemáticos dentro de los modelos, aunque esto puede ser arduo dada la escala de los modelos actuales, que pueden tener de millones a miles de millones de pesos y cómputos. Desde una perspectiva biológica, la capa de entrada de un modelo de aprendizaje profundo es comparable a la transmisión de una imagen a través de las neuronas del cerebro. Del mismo modo, la capa de entrada toma representaciones numéricas y las propaga a través de la red, teniendo cada nodo un peso que conecta con la capa siguiente; este proceso, conocido como propagación hacia adelante, implica el cálculo y la aplicación de una función de activación antes de que la información se propague hacia adelante. (Wolansky, 2021) En la figura a continuación cabe destacar que cada nodo tendrá la capacidad de pasar información al siguiente nodo o capa, o puede incluso también descartarla.

Figura 2. Red neuronal de 3 capas con un esquema de 5 nodos.



Nota. Imagen tomada de (Pal, 2022)

5.1.6.1 Función de activación

Las funciones de activación son componentes esenciales en las arquitecturas de aprendizaje profundo, ya que determinan si se transfiere información al nodo siguiente, como se había mencionado anteriormente. Estas funciones se han desarrollado para facilitar el proceso de aprendizaje, evitar el sobreajuste, aumentar la precisión y reducir el coste computacional de los algoritmos de aprendizaje profundo; esto significa que, sin las funciones de activación adecuadas, los modelos de aprendizaje profundo pueden no alcanzar el rendimiento deseado, haciendo que se comporten como regresiones lineales; es decir, fórmulas de naturaleza

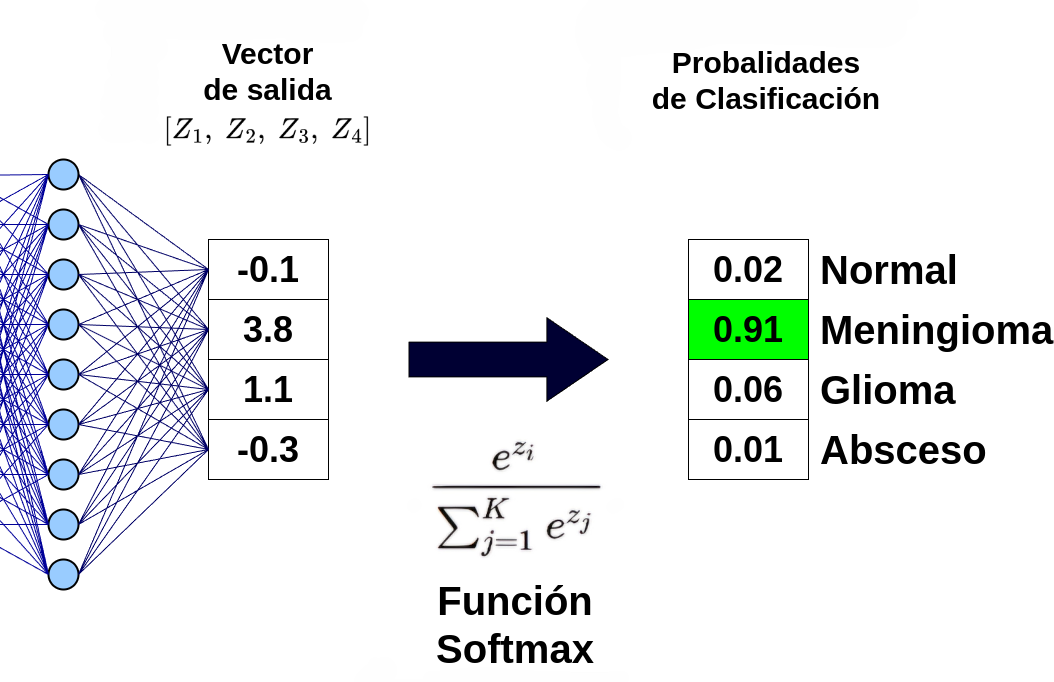
𝑦 = 𝛽0 + 𝛽1 ∗ 𝑥 . (KILIÇARSLAN, 2021)

Ecuación 1, Función de activación

Por ello mismo, se prefiere el uso de funciones de activación no lineales en las arquitecturas de redes neuronales profundas, cuyo rol principal es modificar y limitar la entrada agregada a un intervalo específico, que es crucial para determinar si la información debe transmitirse a los nodos siguientes, con el objetivo de garantizar el aprendizaje eficaz y la representación de relaciones complejas dentro de los datos (KILIÇARSLAN, 2021)

Por ejemplo, la función de activación softmax acepta varias entradas de una capa y las normaliza en una distribución de probabilidad dentro del rango [0,1]. En un escenario con cuatro resultados potenciales, los valores transformados mediante softmax sumarían 1, lo que representa una distribución de probabilidad del 100% de los resultados. A continuación, la red selecciona como salida el resultado con la mayor probabilidad, razón por la cual la función softmax suele situarse al final de una red neuronal. (Wolansky, 2021) Este proceso de selección es esencial para la capacidad de la red de tomar decisiones probabilísticas. Simplificar matemáticamente el modelo al no emplear funciones de activación no lineales en varias capas implica que se comporte de manera similar a un modelo de una sola capa. Esto se debe a que, en ausencia de estas funciones, la suma de capas resulta en un proceso lineal. (Wolansky, 2021)

Figura 3 Ejemplo de clasificación según probabilidad, con el uso de imágenes médicas (TAC cerebral) y la función de activación SOFTMAX



Nota. Imagen tomada de Softmax CNN. (s. f.). Questions And Answers In MRI. https://mriquestions.com/softmax.html

5.1.6.2 Función de pérdida

La función de pérdida es una función matemática de los parámetros del modelo y los datos de entrenamiento; su objetivo es encontrar el conjunto de parámetros que minimicen el valor de la misma, la elección de la función de pérdida depende de la tarea específica y del tipo de datos que se utilicen (Chen, 2022), por ejemplo, la función de pérdida de entropía cruzada binaria se suele utilizar para tareas de clasificación binaria, mientras que la función de pérdida de error cuadrático medio se utiliza para tareas de regresión.

En el Aprendizaje profundo, la función de pérdida se utiliza junto con un algoritmo de optimización, como el descenso de gradiente estocástico, para actualizar los parámetros del modelo durante el entrenamiento; el algoritmo de optimización ajusta iterativamente los parámetros para minimizar la función de pérdida (Chen, 2022).

En el estado del arte de (Basnet, 2020) se proponen nuevas funciones de pérdida para mejorar el rendimiento de los modelos de aprendizaje profundo en aplicaciones específicas. Por ejemplo, se destaca un estudio que propuso una función de pérdida mejorada con regularización y pérdida ponderada para la clasificación de glóbulos blancos; la cual se llama Pérdida de entropía cruzada (Cross-Entropy Loss), una función de pérdida popular utilizada en el Aprendizaje profundo para tareas de clasificación, y que adicionalmente mide la diferencia entre las probabilidades predichas y las etiquetas verdaderas.

La función de pérdida de entropía cruzada se define como:

𝐿(𝑦, 𝑡) = ∑𝑖 𝑡𝑖 ln 𝑦𝑖 (Basnet, 2020)

Ecuación 2, Función de pérdida

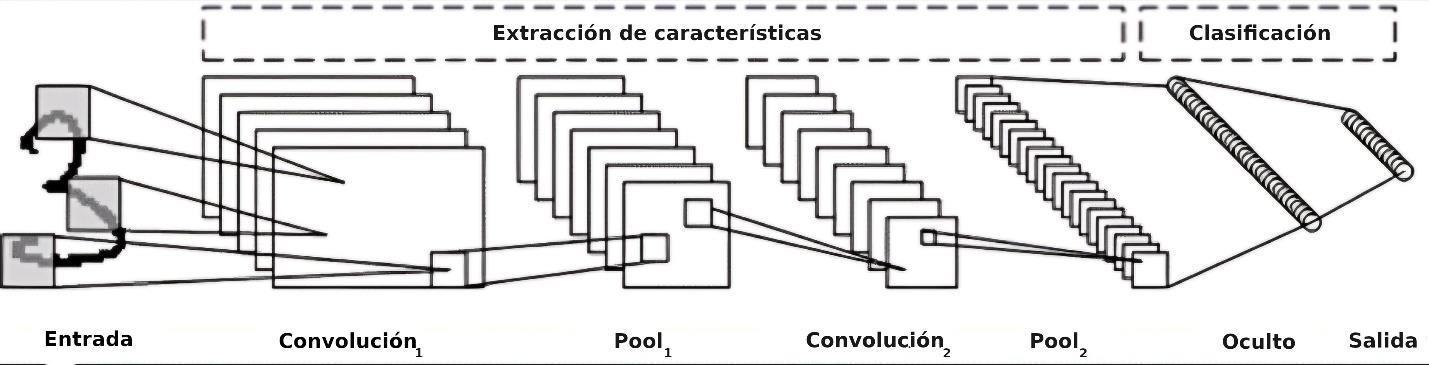
Donde 𝑦𝑖 es la probabilidad prevista de la 𝑖𝑡ℎ clase, y 𝑡𝑖 es la etiqueta verdadera de la 𝑖𝑡ℎ clase (Basnet, 2020)

La pérdida de entropía cruzada se utiliza en tareas de clasificación binaria, en las que el objetivo es predecir una de dos clases. También se utiliza en tareas de clasificación multiclase, en las que el objetivo es predecir una de varias clases; la función de pérdida de entropía cruzada es diferenciable porque facilita su optimización mediante el descenso gradiente (Basnet, 2020)

5.1.7 Redes neuronales convolucionales (CNN)

Una CNN es una red neuronal multicapa que se inspiró biológicamente en la corteza visual animal. La arquitectura es particularmente útil en aplicaciones de procesamiento de imágenes. La primera CNN fue creada por Yann LeCun; en ese momento, la arquitectura se centraba en el reconocimiento de caracteres escritos a mano, como la interpretación de códigos postales. (Venkatesan, 2017) Al tratarse de una red profunda, las primeras capas reconocen entidades (como los bordes) y las capas posteriores recombinan estas entidades en atributos de nivel superior de la entrada. (Chen, 2022)

La arquitectura de LeNet CNN se compone de varias capas que implementan la extracción de características y la clasificación.

Figura 4. Proceso de extracción de datos y procesamiento convolucional de una CNN

Nota. Imagen tomada de (Venkatesan, 2017)

La imagen se divide en campos receptivos que alimentan una capa convolucional, que luego extrae entidades de la imagen de entrada; el siguiente paso es la agrupación, que reduce la dimensionalidad de las entidades extraídas (a través de la reducción de muestreo) al tiempo que conserva la información más importante (normalmente, a través de la agrupación máxima). (Venkatesan, 2017) A continuación, se realiza otro paso de convolución y agrupación que alimenta un perceptrón multicapa totalmente conectado. La capa de salida final de esta red es un conjunto de nodos que identifican las características de la imagen (en este caso, un nodo por número identificado); la red se entrena mediante la propagación hacia atrás (Venkatesan, 2017)

El uso de capas profundas de procesamiento, convoluciones, agrupación y una capa de clasificación totalmente conectada abrió la puerta a varias aplicaciones nuevas de las redes neuronales de aprendizaje profundo; además del procesamiento de imágenes, la CNN se ha aplicado con éxito al reconocimiento de vídeo y a diversas tareas dentro del procesamiento del lenguaje natural. Las CNN constan de capas de entrada, spline, auxiliar y de salida. También detectan patrones visuales obtenidos a partir de píxeles de imágenes sin procesar utilizando capas ocultas. (Venkatesan, 2017) Las funciones no lineales extraen información sobre las características de la imagen; la 'agrupación' reduce los datos y acelera el cálculo. Esto permite encontrar características similares en la imagen para el análisis de patrones. (Kufel, 2023)

5.1.8 Arquitecturas Aprendizaje profundo

La elección de la arquitectura es una tarea fundamental a la hora de determinar el rendimiento, pues determina la eficacia con la que realiza las tareas específicas que se le encomienda. Para el proyecto que se ocupa, la selección de arquitecturas de aprendizaje profundo es una decisión crítica que influye significativamente en la precisión y eficacia del artefacto desarrollado para la tarea de clasificación de imágenes de enfermedades oculares. Entre las de arquitecturas disponibles están VGG19, InceptionV3, Xception, ResNet50 y EfficientNet, estas se destacan por sus características únicas y sus capacidades en el manejo de tareas complejas relacionadas con clasificar imágenes, en especial imágenes médicas, como se podrá evidenciar en el Estado del Arte. Estas arquitecturas seleccionadas representan una variedad de enfoques en el Aprendizaje profundo, ya que cada uno ofrece ventajas distintas que pueden aprovecharse para mejorar la precisión del artefacto. A continuación, se describirá brevemente cada una de ellas, y se presentará una tabla comparativa que ayude a diferenciarlas entre sí

VGG19: Es una extensión de VGG16, con 19 capas de peso con una estructura similar pero más profunda que VGG16. Tiene 16 capas convolucionales y 3 capas totalmente conectadas, lo que ofrece una mayor capacidad de representación a costa de mayores requisitos computacionales (Mascarenhas, 2021)

ResNet50: Es una arquitectura de red neuronal residual con 50 capas de pesos, desarrollada por Microsoft Research. ResNet introduce conexiones de salto o atajos que permiten a la red aprender funciones residuales, abordando el problema del gradiente de fuga en redes muy profundas (Mascarenhas, 2021)

Xception: Es una red neuronal convolucional conocida por su profundidad y eficiencia en términos de recursos computacionales. Emplea convoluciones separables en profundidad, que ayudan a reducir el número de parámetros manteniendo el rendimiento (Tanvir, 2023)

EfficientNet: Se centra en lograr una mayor precisión y eficiencia escalando el modelo de forma equilibrada en las distintas dimensiones. Utiliza un método de escalado compuesto para aumentar simultáneamente la profundidad, la anchura y la resolución del modelo (Zhu, 2022)

InceptionV3: También conocida como GoogLeNet, se caracteriza por su módulo de Inception que ayuda a mejorar la eficiencia computacional. Utiliza varios filtros en cada capa y concatena sus resultados para mejorar la capacidad de representación. (Pan, 2023)

**Tabla 2.**

Comparativa de Arquitecturas de CNN

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Arquitectura** | **Año** | **Nro total capas** | **Nro capas conectadas** | **Propósito** |
| EfficientNet | 2019 | 236 | 1 | Clasificación de imágenes con grandes cantidades de detalles. Bajo costo computacionales. |
| VGG19 | 2014 | 16 | 3 | Enfocada en el reconocimiento de imágenes a gran escala. |
| ResNet50 | 2015 | 50 | 1 | Interpretación de características visuales. |
| Xception | 2017 | 71 | 2 | Desarrollada por Google, paso intermedio entre la convolución regular y la operación de convolución separable en profundidad. |
| InceptionV3 | 2015 | 48 | 1 | Clasificar y detectar múltiples elementos en una sola imagen. Versión mejorada de GoogleLeNet |

Note. Esta tabla presenta una comparación entre cinco arquitecturas de redes neuronales convolucionales populares: EfficientNet, VGG19, InceptionV3, Xception y ResNet50. Esta información es crucial para comprender las diferencias y fortalezas de cada red neuronal convolucional al aplicarlas en tareas de visión por computadora. La resolución de imágen es estándar para todas las arquitecturas (224x224x3). Información tomada de (Pan, 2023) (Zhu, 2022) (Tanvir, 2023)

5.1.9 Kanban

Kanban es una metodología visual de gestión de proyectos que se centra en la mejora continua y la optimización del flujo de trabajo. También se define como un marco que se basa en parecerse, lo más posible, a una metodología ágil; fue desarrollada a finales de la década de 1940 por Taiichi Ohno, un ingeniero japonés. Kanban es conocido por su simplicidad y flexibilidad, esto lo convierte en una opción atractiva para los equipos que buscan mejorar su flujo de trabajo y sus prácticas de gestión de proyectos (Asana, 2024)

La metodología Kanban se basa en varios principios fundamentales que guían su aplicación, estos son:

Visualización del trabajo: Se hace hincapié en la importancia de visualizar el flujo de trabajo para comprender el estado actual del trabajo e identificar áreas de mejora.

Limitación de trabajo en curso: Kanban anima a los equipos a limitar la cantidad de trabajo en curso para mantener un flujo de trabajo fluido y evitar cuellos de botella.

Mantener el flujo: Tiene como objetivo mantener un flujo continuo de trabajo de "por hacer" a "haciendo" a "hecho" para garantizar que el trabajo se completa de manera eficiente y eficaz.

Mejora continua: Kanban anima a los equipos a mejorar continuamente sus procesos y flujos de trabajo para aumentar la eficiencia y la productividad. Además de sus principios básicos, Kanban también incluye varias prácticas básicas que los equipos pueden adoptar para mejorar su flujo de trabajo y la gestión de proyectos. (Asana, 2024)

Las cuales son:

Visualización del flujo de trabajo: Los tableros Kanban se utilizan para visualizar el flujo de trabajo, lo que permite a los equipos ver el estado actual del trabajo e identificar áreas de mejora.

Limitación del trabajo en curso: Kanban anima a los equipos a limitar la cantidad de trabajo en curso para mantener un flujo de trabajo fluido y evitar cuellos de botella.

Gestionar el flujo: Pretende mantener un flujo continuo de trabajo de "por hacer" a "haciendo" y a "hecho" para garantizar que el trabajo se completa de forma eficiente y eficaz.

Mejorar en colaboración: Kanban anima a los equipos a colaborar y mejorar juntos sus procesos y flujos de trabajo.

Kanban funciona utilizando un tablero visual para gestionar los elementos de trabajo, que se representan mediante tarjetas o notas adhesivas; el tablero se divide en columnas que representan distintas fases de trabajo, como "Por hacer", "En curso" y "Hecho". (Asana, 2024)

Cada tarjeta o nota adhesiva representa una tarea o elemento de trabajo específico, y se desplaza por el tablero a medida que avanza por las distintas etapas (Asana, 2024)

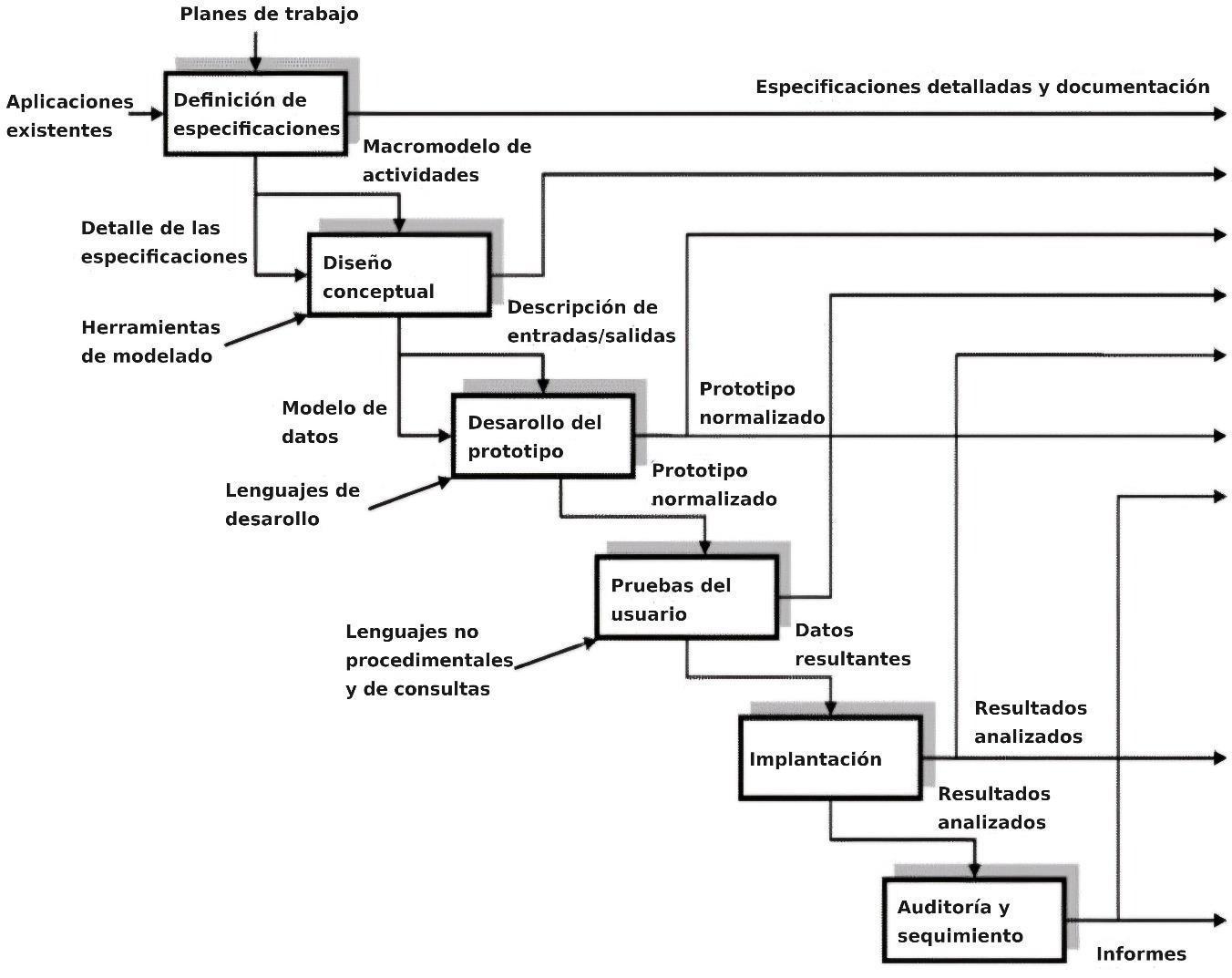
Figura 5. Tablero Kanban



Nota. Imagen tomada de Rodriguez, D. (2022, 4 noviembre).

5.1.10 Prototipado Rápido

Es un flujo de trabajo de diseño que incluye la creación de prototipos y la prueba de una solución de diseño con usuarios y partes interesadas para ayudar a los diseñadores y equipos de productos a refinar y validar ideas rápidamente; el objetivo principal es obtener información eficaz y comentarios de los usuarios desde el principio, refinando el diseño hasta que cumpla con todos los requisitos y necesidades del usuario.

Figura 6. Fases de la Metodología de Prototipado Rápido (MPR)

Nota. Imagen tomada de (Pressman, 2002).

Es una metodología que se adapta a equipos de desarrollo reducidos y que cuentan con recursos limitados, como es el caso habitual en entornos académicos; esto es debido a que permite la realización de las distintas fases de elaboración del producto final (reflejadas en la ilustración 6) en pequeños incrementos, simplificando así la complejidad del proyecto (Pressman, 2002).

Además, esta metodología hace uso de las tecnologías más avanzadas en cuanto a desarrollo de software se refiere; esto permite involucrar en el desarrollo (y también en posteriores mejoras) a estudiantes que se encuentran en la fase final de sus estudios o bien realizando el proyecto fin de carrera, ya que la adquisición de experiencia en tecnologías modernas y ampliamente utilizadas en la industria favorece su motivación (Pressman, 2002).

5.2 Marco Conceptual

5.2.1 Conjunto de Datos (Dataset)

En el contexto del aprendizaje automático y el aprendizaje profundo, un conjunto de datos se refiere a una colección de datos utilizados para entrenar, validar y probar un modelo. El conjunto de datos suele consistir en datos de entrada y sus correspondientes etiquetas de salida, que se utilizan para entrenar el modelo a fin de que reconozca patrones y haga predicciones sobre datos nuevos no antes vistos (Erdinest, 2024)

En el caso de la clasificación de imágenes en enfermedades oculares, un conjunto de datos puede consistir en imágenes médicas del ojo, junto con las etiquetas correspondientes que indican la presencia o ausencia de afecciones específicas como retinopatía diabética, glaucoma o degeneración macular asociada a la edad. (Mostafa, 2023)

5.2.2 Modelos Aprendizaje profundo

Se refiere a la representación matemática de una red neuronal que aprende a realizar tareas específicas ajustando sus parámetros internos en función de los datos de entrada (Al-Fahdawi, 2023).

Un modelo de Aprendizaje profundo aprende esencialmente a partir de datos etiquetados durante una fase de entrenamiento para optimizar sus parámetros y obtener predicciones o clasificaciones precisas; una vez entrenado, el modelo se puede utilizar para hacer predicciones sobre datos nuevos no vistos. (Al-Fahdawi, 2023) En el contexto del presente proyecto; es decir, de la clasificación de imágenes en enfermedades oculares, estos modelos se entrenan en imágenes médicas para detectar patrones indicativos de afecciones específicas como la retinopatía diabética o el glaucoma.

5.2.3 Python

Python es un lenguaje de programación potente y fácil de aprender con estructuras de datos eficientes y programación orientada a objetos. Su sintaxis elegante y escritura dinámica lo hacen ideal para secuencias de comandos y desarrollo rápido de aplicaciones. El intérprete de Python y la biblioteca estándar están disponibles de forma gratuita en las principales plataformas y se pueden ampliar con nuevas funciones y tipos de datos. (Python software, 2024)

5.2.3.1 Estructura de Python

Python es un lenguaje preferido para tareas de datos avanzadas debido a sus estructuras de datos eficientes y capacidades de programación orientada a objetos. Es fácil de aprender, lo que lo hace accesible para diversos profesionales como biólogos, estadísticos y directores de empresas emergentes. Python ofrece estructuras de datos y algoritmos integrados, lo que permite la creación de objetos de datos personalizados. El lenguaje proporciona bibliotecas internas como colecciones y objetos matemáticos, así como bibliotecas externas como SciPy para operaciones de datos avanzadas como regresión y visualización. Si bien las bibliotecas externas ofrecen soluciones listas para usar, la creación de objetos personalizados desde cero puede mejorar el rendimiento de tareas específicas. La popularidad de Python se debe a su legibilidad, flexibilidad y función de consola interactiva. (Benjamin Baka, 2018)

5.2.4 Entorno de desarrollo

5.2.4.1 Google Colab

Google Colab, también conocido como Colaboratory, es un servicio gratuito en la nube de Google diseñado para ejecutar código Python en el navegador, particularmente para aprendizaje automático, análisis de datos y fines educativos. Ofrece acceso gratuito a GPU y TPU, esenciales para tareas que requieren una potencia computacional significativa. Colab proporciona un entorno de portátil Jupyter con varios módulos como Numpy, Scipy, Pandas y marcos de aprendizaje profundo como Tensorflow, Keras y Pytorch. Esta plataforma apoya la investigación en aprendizaje automático e inteligencia artificial superando las limitaciones computacionales. (Naik, 2023)

Google Colab es una plataforma basada en la nube que elimina la necesidad de instalaciones de software locales, ofrece control automático de versiones y guarda cuadernos de Python en Google Drive.

Los archivos de Colab se pueden compartir fácilmente con varias personas para la edición en tiempo real, en consecuencia se puede acceder y revisar documentos en Google Colab sin necesidad de instalar ningún software, entre varios autores. La plataforma incluye bibliotecas para diferentes tipos de proyectos (Como de Inteligencia Artificial y/o Ciencia de Datos) y proporciona recursos de CPU, RAM, GPU y TPU en la nube para ahorrar tiempo y costos. (das, 2024)

5.2.4.2 Biblioteca TensorFlow

TensorFlow es una biblioteca de código gratuita desarrollada por Google para crear modelos de aprendizaje automático en varias plataformas, como escritorio, dispositivos móviles, web y la nube. Se centra en redes neuronales artificiales para construir y entrenar modelos para la detección y el razonamiento de patrones. TensorFlow es versátil, admite redes neuronales y ofrece subclases de Keras y API funcionales para personalización. Permite la compilación de modelos, la propagación hacia adelante y hacia atrás, la creación de capas personalizadas, activaciones y bucles de entrenamiento. Esta herramienta simplifica los procesos de aprendizaje automático, facilitando el entrenamiento y despliegue de modelos en servidores o en la web independientemente del idioma o plataforma utilizada. TensorFlow proporciona flexibilidad, control de ejecución y API intuitivas para gestionar tareas de aprendizaje automático de forma eficaz. Además, ayuda a los desarrolladores a ejecutar modelos en dispositivos integrados, móviles o IoT. (Alonso, 2022)

Es un conjunto de herramientas que ayuda a los desarrolladores a ejecutar sus modelos en dispositivos incorporados, móviles o de IoT, y les permite implementar el aprendizaje automático integrado en el dispositivo.

Características de TensorFlow

Optimizado para el aprendizaje automático integrado en el dispositivo, ya que aborda 5 limitaciones clave: latencia (no hay ida y vuelta con un servidor), privacidad (ningún dato personal sale del dispositivo), conectividad (no es necesaria una conexión a Internet), tamaño (tamaño reducido del modelo y de los objetos binarios) y consumo de energía (inferencia de alta eficiencia sin necesidad de conexiones de red)

Compatibilidad con múltiples plataformas, lo que incluye dispositivos iOS y Android, Linux incorporado y microcontroladores.

Compatibilidad con diversos lenguajes, entre los que se incluyen Java, Swift, Objective-C, C++ y Python

Alto rendimiento, con aceleración de hardware y optimización de modelos

Ejemplos de extremo a extremo de tareas comunes de aprendizaje automático, como clasificación de imágenes, detección de objetos, estimación de poses, respuestas a preguntas, clasificación de texto, etc. en múltiples plataformas. Información obtenida de (TensorFlow, 2021)

5.2.4.3 Jupyter Notebook

Jupyter Notebook es una aplicación web de código abierto que permite la creación y compartición de documentos interactivos que contienen código ejecutable, texto explicativo, ecuaciones, gráficos y figura. Si bien se utiliza principalmente con Python, admite varios lenguajes gracias a los kernels, los cuales mantienen el estado de la sesión, permitiendo ejecutar código de manera secuencial y reutilizar variables o resultados de celdas anteriores. Además, facilita la combinación de código informático, descripciones en lenguaje sencillo y datos, lo que lo convierte en una herramienta versátil para análisis de datos, visualización y generación de informes dinámicos (Jupyter, 2020)

5.2.4.4 JupyterLab

JupyterLab es una versión avanzada de Jupyter Notebook, diseñada como un entorno de desarrollo interactivo y modular. Ofrece la capacidad de trabajar con múltiples notebooks, archivos, terminales y herramientas en una interfaz unificada y personalizable, Además, cuenta con una estructura modular y proporciona una experiencia más avanzada, similar a la de un IDE (Entorno Integrado de Desarrollo) (Jupyter, Project Jupyter, 2024)

5.2.4.5 Voila

Voila es una extensión de Jupyter que permite transformar notebooks en aplicaciones web interactivas;a diferencia de Jupyter Notebook, donde tanto el código como los resultados son visibles para el usuario, Voila se enfoca en ocultar el código y mostrar únicamente los resultados de manera limpia y profesional. Así se facilita la creación de informes interactivos o aplicaciones sin exponer la lógica del código subyacente (Voila, 2020)

5.2.4.6 Widgets de IPy

IPyWidgets es una librería que permite crear controles interactivos (como deslizadores, cuadros de texto, botones, etc.) dentro de un Jupyter Notebook. Estos widgets facilitan la interacción con el código, permitiendo a los usuarios modificar variables y ver resultados en tiempo real sin tener que ejecutar el código nuevamente (Documentation, 2024)

5.2.4.7 Notebook

Un notebook es un documento que permite la combinación de código informático, descripciones en lenguaje sencillo, datos, visualizaciones enriquecidas como modelos 3D, gráficos y figuras, y controles interactivos. Junto con un editor como JupyterLab, proporciona un entorno interactivo ideal para crear prototipos, explorar y visualizar datos, explicar códigos, y compartir ideas con otros (Documentation J. N., 2024)

5.2.4.8 Ngrok

Ngrok es una herramienta que permite exponer de manera segura un servidor local a Internet mediante la creación de un túnel seguro. Este túnel conecta el servidor local directamente con una URL pública. Además, ngrok ofrece funciones avanzadas como la inspección del tráfico en tiempo real, autenticación segura y el uso de protocolos HTTPS, lo que facilita no solo el desarrollo y prueba de aplicaciones, sino también la implementación de demostraciones, pruebas de APIs, y la depuración remota de servicios en entornos de red restringidos o de difícil acceso (ngrok, n.d.)

5.2.5 Artefacto

En el contexto del presente proyecto, el artefacto se refiere al modelo final del sistema basado en aprendizaje profundo para la clasificación de imágenes de retinoscopía, que incluye la integración del modelo de red neuronal convolucional (CNN) con una interfaz. Este artefacto es una representación tangible de los objetivos del proyecto, que comprende el modelo CNN entrenado y su integración con una interfaz.

5.2.6 Método científico

El método científico es un proceso sistemático utilizado para investigar acontecimientos, verificar o construir una versión exacta y fiable de cualquier acontecimiento, implica una serie de pasos que garantizan la documentación de los resultados y minimizan los sesgos en la investigación. (Philosophy, 2021) El método científico incluye las siguientes fases:

Observación y formulación de una pregunta: Es el primer paso del método científico.

Consiste en hacer una observación y formular una pregunta sobre esa observación. (Philosophy, 2021)

Recogida de datos e hipótesis: El siguiente paso consiste en recopilar todos los datos relacionados y formular una hipótesis basada en la observación; la hipótesis puede ser la causa del acontecimiento, su efecto o su relación con cualquier otro acontecimiento. (Philosophy, 2021)

Comprobación de la hipótesis: Una vez formulada la hipótesis, hay que probarla para verificar su estado de verdad (es decir, si es verídica o no). Esto implica realizar experimentos para determinar si la hipótesis concuerda o contradice las observaciones realizadas en el mundo real. (Philosophy, 2021)

Análisis y conclusión: Este paso implica el uso de procedimientos matemáticos y otros procedimientos científicos adecuados para determinar los resultados del experimento. A partir del análisis, se puede determinar el curso de acción futuro. Si los datos hallados en el análisis concuerdan con la hipótesis, ésta se acepta. Si no, se rechaza o se modifica y se vuelve a analizar. (Philosophy, 2021)

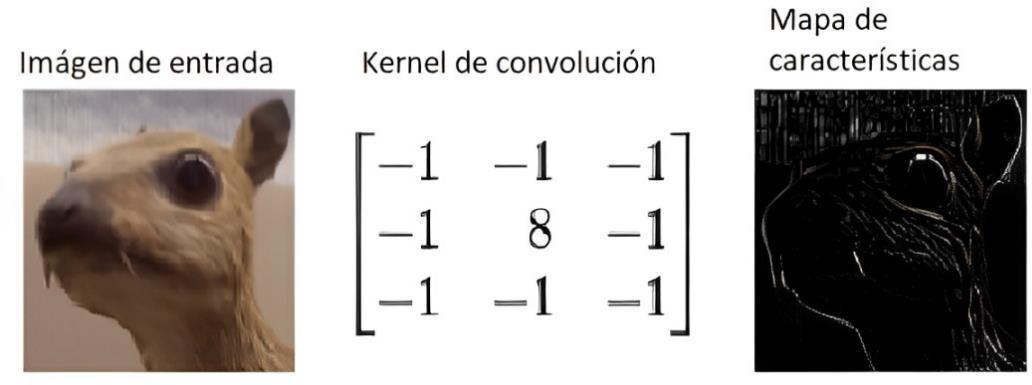
Reflexión e iteración: El último paso del método científico es reflexionar sobre nuestros resultados y utilizarlos para guiar nuestros próximos pasos. Si se confirma la hipótesis, podemos realizar pruebas adicionales para confirmarla o revisarla para hacerla más específica. Si la hipótesis no se confirmara, plantearíamos una nueva hipótesis. (Philosophy, 2021)

5.2.7 Teorema de Convolución

El teorema de convolución es un concepto fundamental en el campo de las matemáticas de ingeniería, en particular en el procesamiento de señales y los sistemas de control, así como en el aprendizaje de máquina. Es un teorema que ayuda a caracterizar y controlar sistemas basándose en su respuesta al impulso. En el procesamiento de imágenes, el filtrado convolucional se utiliza para implementar algoritmos como el desenfoque de imágenes, la detección de bordes, la nitidez o el redimensionamiento de imágenes, entre otros. Esto se consigue mediante un kernel de convolución.

5.2.8 Funcionamiento de la convolución en imágenes

Al aplicar la convolución a las imágenes, se combinan dos cubos de información: La imagen de entrada, que consta de tres matrices de píxeles que representan los canales de color rojo, azul y verde, y el kernel de convolución, una única matriz de números flotantes que sirve como herramienta para entrelazar la imagen de entrada con el kernel. La salida del kernel es la imagen alterada, a menudo denominada mapa de características en el aprendizaje profundo. Para realizar la convolución, se toma una sección de la imagen del mismo tamaño que el kernel y se multiplica elemento a elemento con el kernel. La suma de esta multiplicación da como resultado un píxel del mapa de características, y el proceso se repite para todos los píxeles del mapa de características. (Dettmers, 2021)

Figura 7. Fases del Teorema Convolución

Nota. Imagen tomada de (Dettmers, 2021)

5.3 Marco Histórico

5.3.1 Contexto histórico de la Inteligencia artificial

La historia de la inteligencia artificial (IA) puede rastrearse hasta épocas antiguas, donde filósofos trataban de explicar el sistema mental humano como un sistema simbólico. Sin embargo, la inteligencia artificial moderna realmente comenzó a tomar forma en la mitad del siglo XX. Las primeras generaciones de investigadores de IA hicieron predicciones sobre su trabajo, y en 1956, la conferencia de Dartmouth oficialmente acuñó el término "Inteligencia Artificial" (Russell, 2010) Los primeros sistemas de IA estaban centrados en reglas, lo que condujo al desarrollo de sistemas más complejos durante la década de 1970 y 1980, junto con un incremento en financiamiento (Russell, 2010). En tiempos recientes, la IA ha experimentado un renacimiento gracias a avances en algoritmos, hardware y técnicas de aprendizaje de máquina. El uso de redes neuronales convolucionales (CNN) en el campo médico comenzó en la primera década de los noventa, pero con el aumento de la disponibilidad de datos médicos digitalizados y el desarrollo de hardware potente y algoritmos de optimización, las CNN han ganado terreno en la clasificación de enfermedades oculares (LeCun, 2015)

5.3.2 Contexto histórico del aprendizaje profundo para clasificación de enfermedades oculares.

La aplicación del aprendizaje profundo en el ámbito de la oftalmología y el diagnóstico médico es producto de una evolución tecnológica que abarca décadas. Para establecer una perspectiva histórica para este proyecto se tiene en cuenta la siguiente línea temporal:

Orígenes de la Inteligencia Artificial (IA) y el Aprendizaje Automático (AM)

1956: Surge el concepto de IA, que marca el inicio de los esfuerzos por simular la cognición humana y la capacidad de resolución de problemas (Erdinest, 2024)

1980s: El aprendizaje automático comienza a ganar terreno como subconjunto de la IA, centrado en el desarrollo de modelos capaces de reconocer patrones y hacer predicciones sin ser programados explícitamente (Archana, 2023)

Desarrollo del aprendizaje profundo

Principios de la década de 2000: Los avances en la potencia de cálculo permiten el desarrollo de redes neuronales más profundas, lo que da lugar al nacimiento del aprendizaje profundo, una extensión de los enfoques tradicionales del aprendizaje automático (Erdinest, 2024)

Mediados y finales de la década de 2010: El aprendizaje profundo se hace cada vez más popular gracias a los avances en áreas como el procesamiento del lenguaje natural, el reconocimiento del habla y la visión por ordenador (Archana, 2023)

Aplicaciones en imagen médica y oftalmología

2010s-Presente: El aprendizaje profundo y las CNN empiezan a demostrar capacidades impresionantes en la interpretación de imágenes médicas, incluidas las exploraciones oftalmológicas. Algunos ejemplos son la identificación de signos tempranos de queratocono, retinopatía diabética, degeneración macular asociada a la edad, glaucoma, etc (Mostafa, 2023)

5.4 Marco Normativo

Es importante señalar que, para el presente proyecto, no se aplicará el presente marco. Considerando el contexto y los requerimientos particulares del proyecto, se ha determinado que no existe una correspondencia directa con el marco normativo.

5.5 Marco Ambiental

Es importante señalar que, para el presente proyecto, no se aplicará el presente marco. Considerando el contexto y los requerimientos particulares del proyecto, se ha determinado que no existe una correspondencia directa con el marco ambiental.

5.6 Marco Cultural

Es importante señalar que, para el presente proyecto, no se aplicará el presente marco. Considerando el contexto y los requerimientos particulares del proyecto, se ha determinado que no existe una correspondencia directa con el marco cultural.

5.7 Estado del Arte

Se presenta a continuación en esta sección el estado del arte del conjunto de datos seleccionado, así como de los modelos de aprendizaje profundo probados imágenes de tejidos similares a fondo de ojo.

**Tabla 3.**

Estado del arte del conjunto de datos seleccionado

|  |  |
| --- | --- |
| **Título del conjunto de datos** | Clasificación de las enfermedades oculares (Eyes Disease Classification) |
| **Nombres y Apellidos del Autor** | Guna Venkat Doddi |
| **Año de publicación** | 2022 |
| **Descripción del conjunto de datos**  El dataset en cuestión posee imágenes de fondo de ojo de algunas de las enfermedades más comunes, entre las cuales se encuentran el glaucoma, una enfermedad que daña el nervio óptico y puede provocar pérdida de visión; la catarata, que causa opacidad en el cristalino y afecta la visión, la retinopatía diabética, que daña los vasos sanguíneos de la retina debido a la diabetes; y glaucoma, una afección que causa que el nervio que conecta el ojo con el cerebro se dañe, generalmente debido a una presión ocular elevada.  Este conjunto de datos contiene imágenes de retina normales, retinopatía diabética, cataratas y glaucoma, con alrededor de 1,000 imágenes diagnósticas por clase. | |
| **Objetivo del conjunto de datos**  Su objetivo principal es entrenar modelos de Aprendizaje profundo, permitiendo la identificación de patrones específicos asociados con cada una de estas enfermedades oculares. Además de las imágenes, el conjunto de datos también proporciona etiquetas correspondientes que indican la presencia o ausencia de las afecciones mencionadas. | |
| **Fecha en que se elaboró este RAE** | 20/03/2024 |

**Tabla 4.**

Estado del arte Clasificación con red neuronal convolucional de arquitectura VGG-19 para la detección y graduación de lesiones en retinopatía diabética basado en aprendizaje profundo.

|  |  |
| --- | --- |
| **Título del texto** | Clasificación con red neuronal convolucional de arquitectura VGG-19 para la detección y graduación de lesiones en retinopatía diabética basado en aprendizaje profundo |
| **Nombres y Apellidos del Autor** | V. Sudha1 y T. R. Ganeshbabu |
| **Año de publicación** | 2020 |
| **Resumen del texto**  Se centra en la utilización de un clasificador de red neuronal convolucional (CNN) basado en la arquitectura VGG-19 para detectar y clasificar lesiones en la retinopatía diabética mediante técnicas de aprendizaje profundo. El estudio enfatiza el uso del mapeo de prominencia, un proceso de segmentación de imágenes que resalta cualidades únicas de los píxeles, para mejorar el diagnóstico de la retinopatía diabética. | |
| **Objetivos del texto**  El objetivo final de la investigación es mejorar la precisión del diagnóstico de la retinopatía diabética mediante métodos sofisticados de análisis de imágenes. | |
| **Fecha en que se elaboró este RAE** | 20/03/2024 |

**Tabla 5.**

Estado del arte Algoritmos de aprendizaje profundo para la detección y el diagnóstico de enfermedades sistémicas basados en manifestaciones oftálmicas: una revisión sistemática.

|  |  |
| --- | --- |
| **Título del texto** | Algoritmos de aprendizaje profundo para la detección y el diagnóstico de enfermedades sistémicas basados en manifestaciones oftálmicas: una revisión sistemática |
| **Nombres y Apellidos del Autor** | Wai Cheng Iao,Weixing Zhang,Xun Wang,Yuxuan Wu,Duoru Lin y Haotian Lin |
| **Año de publicación** | 2024 |
| **Resumen del texto**  Este estudio se enfoca en revisar y evaluar los algoritmos de aprendizaje profundo ResNet50 y EfficientNet, aplicados al diagnóstico de enfermedades sistémicas a partir de manifestaciones en imágenes oftálmicas. La investigación se enfoca en la detección de afecciones como diabetes, hipertensión y enfermedades cardiovasculares, mediante patrones en retinografías. | |
| **Objetivos del texto**  El objetivo principal es ver qué tan precisos y efectivos son estos algoritmos para el diagnóstico de enfermedades a partir de imágenes oftálmicas, cuáles son sus beneficios (como la detección temprana y sin intervención humana) y también sus limitaciones, como la necesidad de grandes cantidades de datos y la dificultad para interpretar sus resultados de manera clara para uso clínico.  **Fecha en que se elaboró este RAE** 20/03/2024 | |

**Tabla 6.**

Estado del arte Clasificación de enfermedades oculares basada en aprendizaje profundo utilizando una red neuronal convolucional para imágenes OCT

|  |  |
| --- | --- |
| **Título del texto** | Clasificación de enfermedades oculares basada en aprendizaje profundo utilizando una red neuronal convolucional para imágenes OCT |
| **Nombres y Apellidos del Autor** | Mohamed Elkholy y Marwa A. Marzouk |
| **Año de publicación** | 2024 |
| **Resumen del texto**  Se habla de la utilización de una red neuronal convolucional (CNN) basada en aprendizaje profundo para la clasificación de enfermedades oculares mediante imágenes de tomografía de coherencia óptica (OCT). | |
| **Objetivos del texto**  Desarrollar un algoritmo de aprendizaje profundo eficaz que pueda clasificar con precisión diversas enfermedades oculares, incluido el edema macular diabético (EMD), el glaucoma y la degeneración macular relacionada con la edad (DMAE), basándose en imágenes de OCT. | |
| **Fecha en que se elaboró este RAE** | 20/03/2024 |

**Tabla 7.**

Estado del arte Comparación entre los marcos de arquitectura VGG16, VGG19 y ResNet50 para la clasificación de imágenes médicas normales y procesadas CLAHE

|  |  |
| --- | --- |
| **Título del texto** | Comparación entre los marcos de arquitectura VGG16, VGG19 y ResNet50 para la clasificación de imágenes médicas normales y procesadas CLAHE |
| **Nombres y Apellidos del Autor** | Kamal Kamal |
| **Año de publicación** | 2020 |
| **Resumen del texto**  El estudio se enfoca en la búsqueda del modelo de red neuronal adecuado específicamente para la tarea de clasificación binaria para imágenes médicas de contraste normal y mejoradas mediante la técnica CLAHE. Se realizó una comparación de las arquitecturas de VGG16, VGG19 y Resnet50 en términos de su precisión, puntuación F1 y Recall para un conjunto de imágenes cerebrales seleccionadas de casos de contraste normal y mejorado con CLAHE. Después de realizar 10 experimentos en dos modos: con y sin aumento de datos, manteniendo los mismos parámetros de las simulaciones, se llega a la conclusión de que VGG16 presenta la mejor arquitectura para la clasificación de imágenes médicas. | |
| **Objetivos del texto**  Encontrar el mejor modelo para la clasificación de imágenes médicas mediante la técnica CLAHE | |
| **Fecha en que se elaboró este RAE** 20/03/2024 | |

**Tabla 8**.

Estado del arte Una solución novedosa del uso del aprendizaje profundo para la clasificación de glóbulos blancos: Función de pérdida mejorada con regularización y pérdida ponderada.

|  |  |
| --- | --- |
| **Título del texto** | Una solución novedosa del uso del aprendizaje profundo para la clasificación de glóbulos blancos: Función de pérdida mejorada con regularización y pérdida ponderada |
| **Nombres y Apellidos del Autor** | Basnet, J., Alsadoon, A., Prasad, P.W., Aloussi, S.A., & Alsadoon, O.H. |
| **Año de publicación** | 2020 |
| **Resumen del texto**  Se centra en la aplicación del aprendizaje profundo en la clasificación de los glóbulos blancos y los retos a los que se enfrenta para lograr una precisión y un tiempo de procesamiento óptimos. La idea principal de la investigación es mejorar la precisión de clasificación y predicción de imágenes de sangre utilizando una CNN con una función de pérdida modificada (Cross-Entropy Loss). El sistema también tiene en cuenta los pesos de las muestras pertenecientes a cada clase para compensar los errores derivados de conjuntos de datos desequilibrados. Se cuenta el tiempo de procesamiento de cada imagen para comprobar la mejora del tiempo. La solución propuesta consigue una precisión de clasificación del 98,92% frente al 96,1% y una disminución del tiempo de procesamiento de 0,354 a 0,216 s. El sistema también requiere menos tiempo para la convergencia del modelo, con 9 épocas frente a la media actual de 13,5 épocas. | |
| **Objetivos del texto**  Encontrar una manera de aumentar la precisión de clasificación y predicción de los glóbulos blancos utilizando un modelo de aprendizaje profundo con una función de pérdida modificada y regularización | |
| **Fecha en que se elaboró este RAE** | 20/03/2024 |

Para dar cumplimiento del Objetivo 2, revisar el estado del arte buscando modelos de Aprendizaje profundo probados en imágenes de tejidos similares a fondos de ojos, tal que puedan detectar, procesar y clasificar imágenes de retinoscopía. Se hacen referencia en el estado del arte y en el marco teórico sección arquitecturas de aprendizaje profundo, Tabla 2, Comparativa de Arquitecturas de CNN.

6. Referentes Metodológicos.

El conjunto de datos incluye también imágenes de ojos sanos que sirvan de grupo de control para la comparación. Se especifica la cantidad de imágenes después de la limpieza completa del dataset.

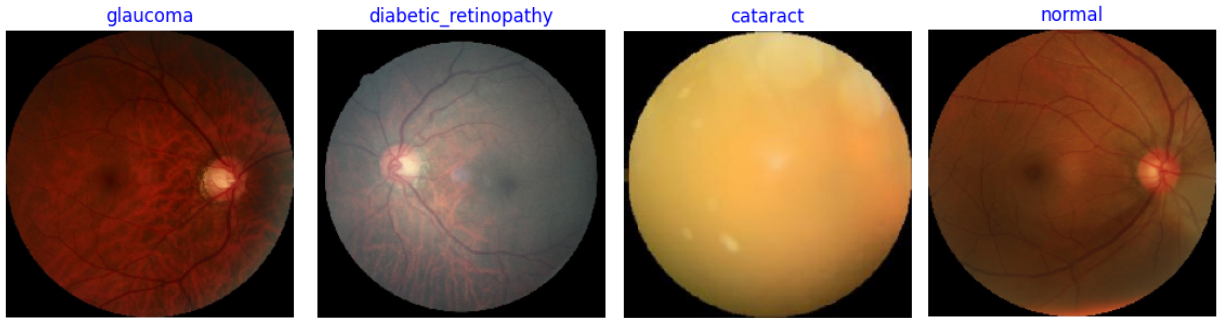
**Tabla 9.**

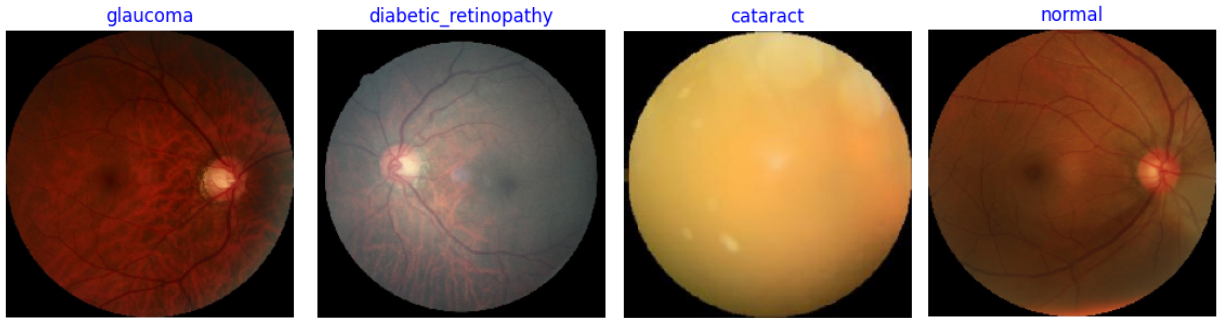
Cantidad de imágenes fondo de ojo

|  |  |
| --- | --- |
| Imágenes de Fondo de Ojo | Cantidad |
| Cataratas | 1038 |
| Retinopatía Diabética | 1098 |
| Glaucoma | 1007 |
| Sano | 1074 |
| Total de Imágenes | 4217 |

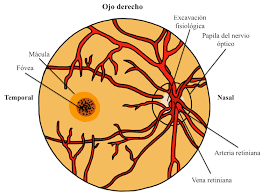
Dando un ejemplo de las diferentes imágenes de fondo de ojo están:

Figura 8, Ejemplos de imágenes de fondo de ojos: Glaucoma, Retinopatía diabética, Catarata y Normal respectivamente





Dentro de la caracterización de imágenes se presentan las siguientes estructuras anatómicas para un mayor entendimiento:

Figura 9. Estructuras Anatómicas

Nota. Tomada de (Rios Zuluaga, 2017)

* Mácula
* Fóvea
* Papila del nervio óptico
* Arterias y venas retinianas
* Excavación fisiológica

Para dar cumplimiento del Objetivo 1, seleccionar un conjunto de datos (dataset) representativos de imágenes de fondo de ojo que contenga las características morfológicas de enfermedades (retinopatía diabética, cataratas y glaucoma), así como de tejido sano, generando un conjunto de datos útil. Se hacen referencia en la sección 1.4.2 Caracterización de Imágenes, en el estado del arte: Guna Venkat Doddi, Clasificación de las enfermedades oculares (Eyes Disease Classification) y referentes metodológicos.

La metodología se desglosa en cuatro fases de las cuales en la fase número 2 en donde se selecciona el modelo CNN, para este resultado se divide en 3 ciclos específico:

Ciclo 1 (C1)

Se refiere a la evaluación inicial de los modelos. Durante este ciclo, se entrenan cinco arquitecturas diferentes de redes neuronales, cada una con una tasa de aprendizaje variable. El entrenamiento se realiza durante 40 épocas, utilizando un tamaño de lote de 16 y un conjunto de datos binario. Este proceso se repite diez veces para cada red, así se obtiene una evaluación preliminar y comparativa del rendimiento según las distintas configuraciones.

Ciclo 2 (C2)

En el segundo ciclo se centra en proporcionar el modelo final. Se seleccionan las tres mejores arquitecturas identificadas en el Ciclo 1 y se entrenan con un conjunto de datos completo y una tasa de aprendizaje fija, determinada según los resultados del ciclo anterior. El entrenamiento se extiende a 60 épocas, manteniendo el tamaño de lote en 16. Se realizan cinco repeticiones para cada red.

Ciclo 3 (C3)

El tercer ciclo se destina a entrenar el modelo final. A partir de la mejor arquitectura escogida en el ciclo anterior se realiza un entrenamiento exhaustivo, el cual se lleva a cabo durante 100 épocas, con una tasa de aprendizaje fija y un tamaño de lote de 32, utilizando el conjunto de datos completo. Se realizan cinco repeticiones, se escoge el modelo final según la mejor de estas.

6.1 Desarrollo de Metodología

6.1.1 Fase 1: Recopilación de datos

La primera fase de este proyecto se centró en la recopilación, validación y preparación del conjunto de datos utilizado para entrenar los modelos de redes neuronales convolucionales (CNN) con el fin de clasificar enfermedades oculares. Se optó por utilizar el conjunto de datos titulado "Eyes Disease Classification" de Guna Venkat Doddi (2022), el cual consta de imágenes de fondo de ojo de retinoscopia, entre las que se incluyen la retinopatía diabética, el glaucoma, las cataratas y ojos normales. Se hablará con más detalle en la siguiente sección:

6.1.1.1 Proceso de búsqueda y descarte para hallar el dataset

El diseño de un modelo eficaz que sea capaz de clasificar enfermedades oculares requiere un dataset con características específicas que garanticen un entrenamiento robusto y una evaluación y validación correcta; es por ello que es relevante revisar estudios anteriores de retinografía y aprendizaje profundo, como los realizados por (Rana, 2022), en dónde describen que un dataset ideal debe contener una amplia variedad de imágenes que representen diferentes condiciones oculares dentro de cada clase (Por ejemplo, en Retinopatía Diabética existe la clase Proliferativa, No proliferativa y la causada por edema macular), además de que las imágenes deben ser de suficiente calidad como para poder aplicarles técnicas de augmentación de datos, las características ideales de un dataset de imágenes de fondo de ojo incluyen:

* El dataset debe incluir imágenes de las cuatro clases principales: Catarata, Glaucoma, Retinopatía Diabética y Normal, con un número balanceado de imágenes por clase; se buscaba una cantidad que no fuera tan mínima, pero tampoco tan abundante (aprox. 1000 imágenes por clases) para este primer acercamiento al prototipo.
* En lo posible, las imágenes deben haberse tomado con más de una cámara retinográfica, para poder añadir diversidad de calidad de imagen al dataset.
* Las imágenes de las clases deben incluir variedad de estadios y subtipos de la imágen; ejemplo: catarata madura e inmadura dentro de la clase catarata, retinopatia diabética proliferativa y no proliferativa dentro de la clase retinopatía diabética.

La siguiente tabla presenta un análisis comparativo de los principales 10 datasets considerados durante el proceso de selección, esto para visualizar una descripción básica  de cada dataset y mostrar su relación con los criterios definidos. Entre las características evaluadas están el número de clases, la cantidad de imágenes por clase, las cámaras retinográficas utilizadas y la diversidad de estadios y subtipos de las patologías representadas.

**Tabla 10.**

Tabla comparativa de datasets en proceso de selección del conjunto de datos

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Dataset** | **Clases** | **Imágenes/Clase** | **Cámara** | **Diversidad** |
| (SMDG, a standardized Fundus Glaucoma dataset., 2023) | Normal, Retinopatía Diabética | ~100 | Zeiss Visucam 100 | Baja |
| (cataract dataset, 2019) | Normal, Enfermedad de la retina sin especificar, glaucoma, catarata | 300 normal, ~100 por el resto de clases | No especificado | Media. No especifica qué enfermedades de la retina incluyen. Se incluye si la catarata está madura o no |
| (HRF, 2020) | Normal, Retinopatía, Glaucoma | ~15 | Canon CR-1 | Baja. No especifica grado de retinopatía o glaucoma |
| (Messidor, 2024) | Retinopatía Diabética, Normal | ~1200 | Topcon TRC NW6 | Alta (Según grado de retinopatía y grado de edema macular) |
| (APTOS 2019, 2019) | Retinopatía Diabética Proliferativa Grave, avanzada y leve | ~1400 | No especificado | Media (Según grado de gravedad de retinopatía proliferativa) |
| (RIM-ONE, s.f.) | Glaucoma, Normal | 313 normal, 172 normal | Zeiss Visucam 500 | Media. Según grado de gravedad de glaucoma |
| (Kaggle EyePACS, s.f.) | Retinopatía Diabética, Normal | ~3500 | Varias | Alta (Según grado de retinopatía y grado de edema macular) |
| (Cataract Classification Dataset, 2024) | Catarata Madura, Catarata Inmadura | 214 inmaduras, 196 maduras | No especificado | Alta (Incluye si la catarata está madura o no) |
| (Doddi, s.f.) (Dataset escogido) | Normal, Retinopatía, Catarata, Glaucoma | ~1000 por clase | Zeiss Visucam 100, Canon CR-1 | Alta (Incluye recolección de imágenes de varios dataset, como IDRiD, Oculur recognition, HRF, etc.) |
| (IDiRD, 2021) | Retinopatía diabética proliferativa, Retinopatía diabética no proliferativa, Retinopatía diabética por edema macular | 516 imagenes de Retinopatía diabética proliferativa, 86 de no proliferativa y 116 por edema macular por retinopatía diabética | Zeiss Visucam 100 | Alta |

El dataset de Gunna Venkat Doddi fue seleccionado porque cumple con las tres características fundamentales necesarias para este proyecto, a diferencia de otros datasets analizados. Se argumenta más a detalle a continuación:

1. Inclusión de las cuatro clases principales con un número balanceado de imágenes (Criterio número 1):
   * Gunna Venkat incluye las clases: Catarata, Glaucoma, Retinopatía Diabética y Normal, con aproximadamente 1000 imágenes por clase, cumpliendo con el primer criterio.
   * Otros datasets tienen limitaciones importantes:
     + SMDG: Incluye solo dos clases (Normal y Retinopatía Diabética) y menos de 100 imágenes por clase.
     + HRF: Tiene tres clases, pero menos de 15 imágenes por clase.
     + APTOS 2019 y Messidor: Tienen una cantidad apta de imágenes pero solo se enfocan en subtipos de Retinopatía Diabética.

2. Diversidad en cámaras retinográficas (Criterio número dos):

* Gunna Venkat cumple con este criterio porque es el único que utiliza imágenes capturadas por más de una cámara, como Zeiss Visucam 100 y Canon CR-1. El resto de datasets solo usan una cámara o no la especifican.

3. Variedad de estadios y subtipos en las imágenes:

* Gunna Venkat cumple este criterio, pues incluye imágenes provenientes de varios dataset **(IDRiD, HRF, etc.),** esto hace que el conjunto abarque diferentes clases específicas de cada enfermedad, como cataratas maduras e inmaduras o retinopatía diabética proliferativa y no proliferativa. Otro candidato era “cataract dataset”, pero se descartó ya que no especificaba qué enfermedades de retina incluía.

Luego de escoger el dataset, se procedió a la revisión del conjunto de datos; el proceso de revisión incluyó un análisis exhaustivo de las imágenes de retinografía para evaluar su calidad, formato y relevancia para las patologías que se pretendían estudiar. Se seleccionaron las imágenes que cumplían con criterios de calidad, tales como la claridad de las estructuras anatómicas del ojo, entre las que se incluyeron la mácula, la fóvea, la papila del nervio óptico y las arterias y venas retinianas.

**Tabla 11.**

Descripción dataset

|  |  |
| --- | --- |
| **Elemento** | **Descripción** |
| Nombre del Dataset | "Eyes Disease Classification" de Guna Venkat Doddi (2022) |
| Propósito del Dataset | Clasificar enfermedades oculares mediante redes neuronales convolucionales (CNN) |
| Tipo de Imágenes | Imágenes de fondo de ojo de retinoscopía |
| Patologías Incluidas | Retinopatía diabética, glaucoma, cataratas, y ojos normales |
| Cantidad Total de Imágenes | 4217 imágenes |
| Distribución de Imágenes | - 1038 imágenes de cataratas   - 1098 imágenes de retinopatía diabética   - 1007 imágenes de glaucoma   - 1074 imágenes de ojos normales |
| Criterios de Calidad | - Claridad de estructuras anatómicas clave del ojo: mácula, fóvea, papila del nervio óptico, arterias y venas retinianas   - Imágenes en formatos compatibles con los requisitos del modelo (JPG o JPEG) |
| Limpieza del Dataset | Eliminación de imágenes duplicadas, de baja calidad o con ruido y artefactos. Se eliminaron 78 imágenes con ruido en el caso de cataratas |
| Revisión de Calidad | Análisis exhaustivo para evaluar la calidad, formato y relevancia de las imágenes. Selección basada en la claridad de estructuras anatómicas |

Uno de los primeros pasos dentro de esta fase fue la validación del conjunto de datos, esto no solo incluyó la evaluación visual de las imágenes, sino también la verificación de que las etiquetas de las imágenes correspondieran a las patologías diagnosticadas. En este punto, la colaboración con un médico fue de gran ayuda al momento de observar las imágenes.

Posteriormente, se llevó a cabo una limpieza rigurosa del conjunto de datos, en el cual se incluyó la eliminación de imágenes duplicadas o de baja calidad, así como de aquellas que contenían ruido o artefactos que pudieran interferir con el entrenamiento de los modelos CNN. En particular, se detectó una cantidad significativa de ruido en las imágenes de cataratas, lo que llevó a una revisión más detallada y a la eliminación de 78 imágenes que no cumplían con los estándares de calidad. Tras este proceso, se determinó que el conjunto de datos final estaba compuesto por 4217 imágenes médicas distribuidas de la siguiente manera: 1038 imágenes de cataratas, 1098 de retinopatía diabética, 1007 de glaucoma y 1074 imágenes de ojos normales.

El conjunto de datos seleccionado para este proyecto fue cuidadosamente examinado para garantizar que representara de manera precisa las patologías oculares de interés. Las imágenes no solo cumplían con los estándares de calidad visual, sino que también estaban en formatos compatibles con los requisitos del modelo (JPG o JPEG. Cabe destacar que el producto final fue un conjunto de datos cuyas imágenes mostrarán de manera clara las estructuras anatómicas clave que serían fundamentales para el proceso de clasificación, cumpliendo así con el primer objetivo específico.

6.1.2 Fase 2: Selección de Modelos CNN

La segunda fase de este proyecto metodológico se centró en la investigación y selección de los modelos de redes neuronales convolucionales (CNN) más adecuados para la tarea de clasificación de imágenes de fondo de ojo. El proceso de selección se lleva a partir de una revisión de la documentación existente, donde se evaluaron diferentes modelos de aprendizaje profundo que han demostrado un desempeño sobresaliente en tareas de clasificación de imágenes médicas. Adicionalmente, si desea consultar los dataset usados para el entrenamiento de los diferentes ciclos, **puede consultar el Apéndice E.**

En primera instancia se explicará la manera en cómo se codificó los notebook de colab para poder realizar las experimentaciones, para ello se expone dos diagramas, el primero es sobre el procesamiento del dataset, y el segundo sobre el entrenamiento y validación del modelo; estos se pueden visualizar con más calidad en el **Apéndice F**, además en el **Apéndice H** **sección a** se explica en video cómo ejecutar paso por paso cada colab (de entrenamiento y reentrenamiento), y en la sección d se encuentra la explicación a detalle de la codificación y su relación con el modelo final:

Figura 10. Diagrama de flujo para el procesamiento del dataset

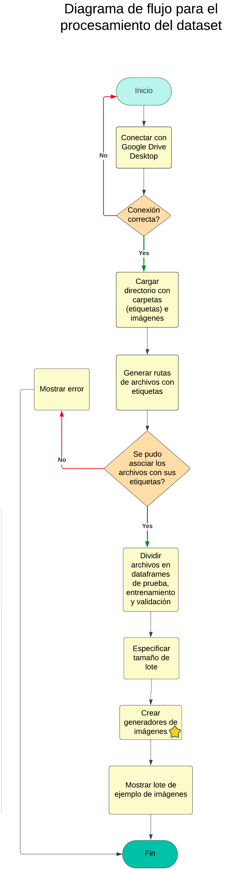
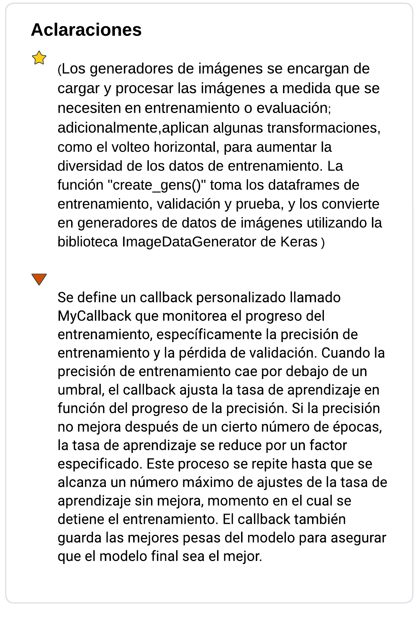


Figura 11. Diagrama de flujo para el entrenamiento y validación del modelo

Diagrama

Descripción generada automáticamente

Figura 12. Aclaraciones de los diagramas de flujo



Se llevó a cabo una investigación que incluyó la comparación de múltiples arquitecturas de redes neuronales convolucionales; para esta investigación, se tomaron en cuenta varios factores clave, entre ellos la profundidad de las capas, la capacidad de representación de las características y el balance entre precisión y eficiencia computacional. De esta revisión surgieron cinco modelos que fueron seleccionados para ser evaluados y comparados: VGG19, ResNet50, Xception, EfficientNet e InceptionV3. Cada uno de estos modelos presenta ventajas únicas que los hacen idóneos para tareas de clasificación de imágenes complejas como las de retinoscopía. Estas características son mencionadas y muy bien explicadas en los marcos conceptuales.

Tras la investigación, los cinco modelos fueron evaluados a través de tres ciclos de pruebas de clasificación para determinar cuál ofrecía el mejor rendimiento en la tarea específica de clasificación de enfermedades oculares.

Ciclo 1: Evaluación Preliminar

El primer ciclo de pruebas consistió en entrenar cada uno de los cinco modelos seleccionados utilizando un conjunto de datos de retinoscopía binario, es decir, dos clases (normal y una enfermedad). Cada modelo se entrenó con tres tasas de aprendizaje diferentes (0.01, 0.001, 0.0001), durante 40 épocas y utilizando un tamaño de lote de 16. Este proceso se repitió diez veces por cada modelo.

Los hiperparámetros y configuraciones necesarias fueron:

* Learning Rates: 0.01, 0.001, 0.0001
* Batch Size: 16
* Número de Épocas: 40
* Repeticiones: 10 por learning rate
* Función de Pérdida: Entropía cruzada
* Función de Activación: Softmax
* Optimización: Adam
* Validación Cruzada: Implementada en este ciclo

En este estudio, se utilizó una configuración del parámetro de "paciencia" (patience) que permite reducir la tasa de aprendizaje si la precisión de validación no mejora después de 3 épocas, y si no mejora el entrenamiento en 5 épocas después de ese ajuste, el entrenamiento se detiene. Esta técnica ayudó a prevenir el sobreajuste y a mantener la eficiencia del modelo.

A lo largo de este ciclo, se recopilaron métricas clave como la precisión y la pérdida de validación, Recall, F1-Score y Precisión por clase. Estos parámetros y configuraciones requeridas al final ayudaron a los resultados, en donde se determinó que VGG19, ResNet50 y EfficientNet obtuvieron las mejores precisiones, mientras que Xception e InceptionV3 mostraron una mayor variabilidad en sus métricas de desempeño, por lo cual estos dos últimos no se escogieron como modelos para el ciclo 2. Estos resultados se explican más detalladamente en la sección de resultados.

Ciclo 2: Especialización de Modelos

En el segundo ciclo, los tres mejores modelos identificados en el ciclo anterior fueron entrenados con el conjunto de datos completo, que contenía las cuatro clases de enfermedades oculares. El objetivo de este ciclo fue afinar las configuraciones y optimizar los hiperparámetros para obtener el mejor rendimiento posible de cada modelo. El entrenamiento se extendió a 60 épocas y se mantuvo la tasa de aprendizaje que proporcionó los mejores resultados en el ciclo anterior. También se usa un parámetro de "paciencia" que reduce la tasa de aprendizaje si la precisión de validación no mejora después de 5 épocas, y si no mejora el entrenamiento en 10 épocas después de ese ajuste, el entrenamiento se detiene.

Se evaluaron tres arquitecturas (VGG19, ResNet50, y EfficientNet) utilizando un dataset con imágenes de fondo de ojo de cuatro clases, utilizando la tasa de aprendizaje que permitió obtener los mejores resultados para cada arquitectura y repitiendo los experimentos cinco veces.

Los hiperparámetros y configuraciones necesarias fueron:

* Learning Rates: 0.001 para Efficientnet y Resnet50, 0.0001 para VGG19
* Batch Size: 16
* Número de Épocas: 60
* Repeticiones: 5 por arquitectura
* Función de Pérdida: Entropía cruzada
* Función de Activación: Softmax
* Optimización: Adam
* Validación Cruzada: Implementada en este ciclo 2

EfficientNet parece ser el mejor modelo en términos de precisión, recall y F1-score en las clases más problemáticas como Catarata y Glaucoma, haciéndolo ideal para su selección como modelo final. VGG19 tiene un desempeño muy sólido, especialmente en Glaucoma y Normal, pero su exactitud general es inferior a la de EfficientNet. ResNet50, aunque tiene un buen desempeño, queda rezagado en varias clases, especialmente en Glaucoma, y eso reduce su fiabilidad en comparación con EfficientNet y VGG19; estas conclusiones se explican más a detalle en el respectivo capítulo de resultados.

Ciclo 3 Entrenamiento del modelo

El tercer ciclo se destina a entrenar el modelo final. Se elige la mejor arquitectura determinada en los ciclos anteriores y se realiza un entrenamiento exhaustivo. Este entrenamiento se lleva a cabo durante 100 épocas, con una tasa de aprendizaje fija y un tamaño de lote de 32, utilizando el conjunto de datos completo. Se realizan cinco repeticiones; en este ciclo también se usa el parámetro de “paciencia” para reducir la tasa de aprendizaje si la precisión de validación no mejora después de 7 épocas, y si no mejora el entrenamiento en 10 épicas después de ese ajuste, el entrenamiento se detiene.

Cabe destacar que con la finalización de este ciclo se da cumplimiento al objetivo específico número 3.

**Tabla 12.**

Comparativa Hiperparámetros por ciclo

|  |  |  |  |
| --- | --- | --- | --- |
| **Aspecto Comparado** | **Ciclo 1: Selección de Modelos** | **Ciclo 2: Especialización de Modelos** | **Ciclo 3: Entrenamiento del Modelo Final** |
| Learning Rates | 0.01, 0.001, 0.0001 para  VGG19, ResNet50, EfficientNet y  InceptionV3 | 0.001 para EfficientNet y ResNet50, 0.0001 para VGG19 | 0.001 para EfficientNet |
| Batch Size | 16 | 16 | 32 |
| Número de Épocas | 40 | 60 | 100 |
| Repeticiones | 10 | 5 | 5 |
| Función de Pérdida | Entropía cruzada | Entropía cruzada | Entropía cruzada |
| Función de Activación | Softmax | Softmax | Softmax |
| Optimización | Adam | Adam | Adam |

6.1.3 Fase 3: Integración del Modelo CNN

Diseño de la interfaz

Se diseñó la interfaz, considerando la facilidad de uso y la presentación de los resultados de clasificación de manera clara y comprensible para el usuario final. Se buscó el medio para cumplir el objetivo tipo python notebook, como no funciono colab, se busca otra alternativa en donde se corre el código, pero no se pueda mostrarlo, pues el usuario no necesita verlo; después de una investigación se concluye el uso de Jupyter con voila, el primero proporcionó la interfaz tipo notebook y Voila se encargó de proporcionar una simulación de aplicativo web en donde solo se muestre la interfaz gráfica, con esto se superó el imprevisto de incompatibilidad que se generó el tratar de usar Colab con Ngrok para realizar la interfaz.

Codificación de la interfaz

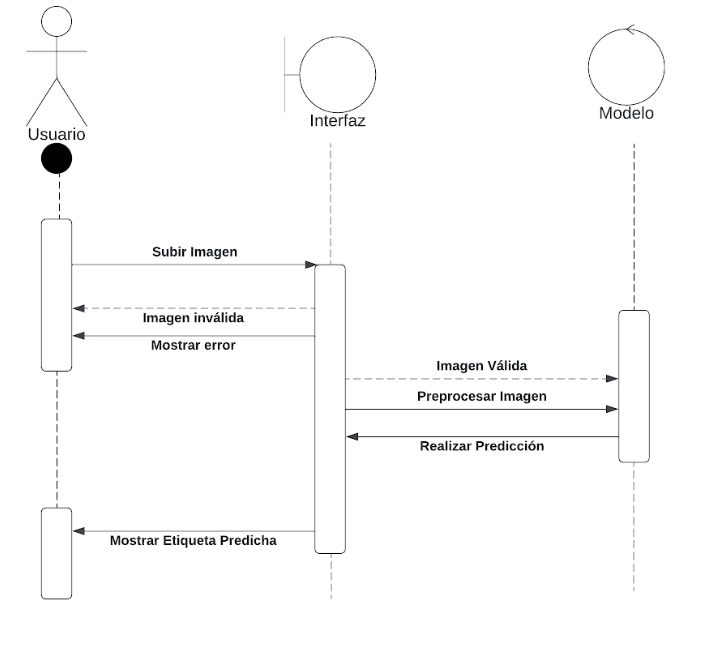
Se procedió a la codificación de la interfaz utilizando las tecnologías y herramientas adecuadas para su desarrollo. Se buscaron las librerías necesarias para poder crear la interfaz, la librería que permite subir la imagen fue ipywidgets, ésta permitió mostrar de manera amigable con el usuario los mensajes de guía, además se implementa una herramienta interactiva utilizando jupyter notebook y Voila que permite la carga de imágenes de retina en formato .jpg o .jpeg para su clasificación mediante un modelo de aprendizaje profundo preentrenado. Para poder explicar su funcionamiento y relación entre la interfaz-modelo se usarán los diagramas de secuencia y de flujo (Los cuales se pueden ver con más calidad en el **Apéndice G**), y un vídeo que explica más a detalle junto con los diagramas en el **Apéndice H, sección c**. Cabe destacar que en el **Apéndice H sección b** se explica cómo instalar las herramientas necesarias para ejecutar la interfaz, así como un manual de uso de esta última; si se desea probar la interfaz, en el **Apéndice J** se encontrará el link de descarga de esta:

Figura 13. Diagrama de flujo del funcionamiento de la interfaz

Diagrama

Descripción generada automáticamente

Figura 14. Diagrama de secuencia del proceso de clasificación de imágenes y su relación entre interfaz-modelo



A continuación, se ahonda más en cada proceso:

Importación de librerías

Se utilizan varias librerías esenciales. ipywidgets facilita la creación de widgets interactivos como el botón de carga de archivos y etiquetas para la interfaz de usuario. pil (python imaging library) permite la carga y manipulación de imágenes, mientras que numpy convierte estas imágenes en matrices numéricas para que puedan ser procesadas por el modelo. La visualización de las imágenes se realiza mediante matplotlib.pyplot, y tensorflow.keras.models se emplea para cargar el modelo de clasificación previamente entrenado.

Carga del modelo

Se carga un modelo preentrenado almacenado en el archivo 'efficientnet\_retina\_v1\_40\_learn\_3.0\_batch\_32\_rep\_3.h5', que utiliza la arquitectura de EfficientNetB3 para la clasificación de imágenes de retinoscopia. el modelo se integra de manera que pueda ser utilizado dentro del entorno interactivo para realizar predicciones en tiempo real.

Creación de la interfaz de usuario

se emplean widgets para crear una interfaz amigable. el widget fileupload genera un botón que permite a los usuarios cargar imágenes de su dispositivo, mientras que un widget de output permite mostrar la imagen cargada y los resultados del modelo. Una etiqueta informativa (label) guía al usuario, indicándole que debe cargar una imagen en el formato correcto.

Procesamiento de la imagen cargada

Cuando el usuario sube una imagen, esta es procesada por la función process\_image: Primero, se verifica que el archivo subido sea válido; luego, la imagen se carga utilizando pil.image y se convierte en un formato que el modelo puede interpretar. Para que la imagen sea compatible con el modelo, se redimensiona a 224x224 píxeles y se convierte a un array de numpy.

Predicción del modelo

Una vez que la imagen ha sido preprocesada, el modelo realiza una predicción sobre la clase a la que pertenece la imagen de retina. La predicción resulta en un array de probabilidades para cada clase posible (catarata, retinopatía diabética, glaucoma y retina normal). Se utiliza np.argmax para determinar cuál es la clase con la mayor probabilidad, y esta se mapea a una etiqueta legible para el usuario (por ejemplo, "catarata").

Visualización y resultados

La interfaz no solo permite la visualización de la imagen cargada, sino también la predicción realizada por el modelo. Si la imagen no es válida, se muestra un mensaje de error pidiendo al usuario que suba una imagen en el formato correcto. El sistema está diseñado para ser fácil de usar y proporcionar un resultado claro, mostrando tanto la imagen cargada como la predicción correspondiente.

Actualización dinámica

La funcionalidad interactiva está basada en la capacidad de los widgets para observar los cambios en la interfaz. cada vez que se carga una nueva imagen, la función process\_image se ejecuta automáticamente, eliminando cualquier salida anterior y mostrando los nuevos resultados de la predicción.

En conjunto, este código proporciona una solución que integra el modelo escogido en una interfaz accesible para los usuarios para permitir la carga, procesamiento y clasificación de imágenes de retina en tiempo real.

Integración del modelo cnn con la interfaz

Se integró el modelo de aprendizaje profundo seleccionado previamente con la interfaz, permitiendo la clasificación automática de imágenes de fondo de ojo.

Pruebas de integración del artefacto

Se llevaron a cabo pruebas exhaustivas de integración para garantizar el correcto funcionamiento y la operabilidad entre interfaz; para ello se usan las imágenes en formato jpg, luego la interfaz las redimensiona a 224x224 y las normaliza para que el modelo pueda predecir la clase. Cabe destacar que, con este último paso se da cumplimiento al último objetivo específico.

6.1.4 Fase 4: Escritura

Manual para reentrenamiento de la red neuronal

Se elaboró un manual para el reentrenamiento de la red neuronal, en donde se describió la arquitectura, el funcionamiento y la implementación del artefacto, además del paso a paso a seguir para poder añadir nuevas clases que el modelo pueda predecir.

Manual de usuario

Se creó un manual de usuario que proporciona instrucciones claras y concisas sobre cómo utilizar el artefacto; incluyendo la navegación por la interfaz, la carga de imágenes de retinoscopía y la interpretación de los resultados de clasificación.

Libro final

Se preparó el documento final del proyecto de grado, que incluyó una introducción, antecedentes, metodología, resultados, conclusiones y referencias bibliográficas, así como cualquier otro contenido requerido por las normativas institucionales.

Artículo Científico

Se redactó un artículo científico basado en los hallazgos y resultados del proyecto.

7. Presentación y análisis de resultados

7.1 Desarrollo Fase 2

Este capítulo presenta un análisis de los resultados obtenidos durante los ciclos de evaluación de arquitecturas de redes neuronales convolucionales (CNN) para la clasificación de enfermedades oculares a partir de imágenes de fondo de ojo, descritos en el Estado del Arte. Se evaluaron, inicialmente, cinco arquitecturas (VGG19, ResNet50, EfficientNetB3, Xception e InceptionV3), considerando sus capacidades para clasificar correctamente entre imágenes de ojos sanos y aquellos con patologías (retinopatía diabética, cataratas y glaucoma). Las métricas principales utilizadas en este análisis fueron la precisión (accuracy), recall, f1-score y la función de pérdida en el momento de validación (val\_loss) de experimentos para cada ciclo; además, se realizaron pruebas exhaustivas con diferentes tasas de aprendizaje y configuraciones según el ciclo.

7.2 Preprocesamiento de las Imágenes

Las imágenes de retinoscopía utilizadas durante los tres ciclos se caracterizan por estar en formato .jpg o .jpeg; las imágenes de retinas normales, cataratas y glaucoma están en formato .jpg, mientras que las de retinopatía diabética están en formato .jpeg. Aunque el tamaño de estas imágenes no es relevante debido a la flexibilidad de las redes neuronales convolucionales, se realizaron los siguientes pasos de preprocesamiento para asegurar la calidad de los datos:

Las imágenes fueron redimensionadas a 224x224 píxeles para cumplir con los requisitos de entrada de la arquitectura ResNet50 (Simonyan & Zisserman, 2014).

Se aplicó el proceso de normalización en las imágenes para tener valores de píxeles entre 0 y 1, esto ayuda a acelerar el entrenamiento y mejorar la convergencia del modelo (Elmoufidi & Amoun, 2021).

Se aplicaron técnicas de augmentación de datos como rotaciones, traslaciones, y flips horizontales para aumentar la variabilidad del dataset y mejorar la robustez del modelo (Shorten & Khoshgoftaar, 2019); esto resulta realmente útil para lidiar con el desbalance en la cantidad de imágenes por clase, presente en el ciclo 1.

7.3 Selección de Arquitecturas para la Clasificación del Ciclo 1

En el Ciclo 1 se evaluaron cinco arquitecturas (VGG19, ResNet50, Xception, EfficientNet e InceptionV3) utilizando un dataset con imágenes de fondo de ojo con Batch Size 16, llevando a cabo 10 repeticiones por tasa de aprendizaje, y 40 épocas en cada repetición; los modelos fueron evaluados utilizando tres tasas de aprendizaje: 0.01, 0.001, y 0.0001; estás y demás configuraciones se podrán evidenciar a más detalle en su respectiva sección. Con el fin de simplificar el problema y evaluar la capacidad de los modelos en la distinción básica entre ojos sanos y enfermos, se decidió realizar la clasificación en dos categorías: sano y enfermo, en lugar de las cuatro clases iniciales

7.3.1 Dataset utilizado Ciclo 1

El conjunto de datos utilizado en este estudio es el "Clasificación de las enfermedades oculares", compilado por Guna Venkat Doddi en 2022. Este dataset contiene alrededor de 1,000 imágenes diagnósticas de retina para cada clase, incluyendo retinopatía diabética, cataratas, glaucoma y ojos normales. En esta primera fase del estudio, se decidió clasificar las imágenes en dos categorías (sano y enfermo) en lugar de cuatro (glaucoma, catarata, retinopatía diabética y sano); dando como resultado un total de 3065 imágenes para la clase enfermo, y 1074 imágenes para la clase sano.

La decisión de reducir las categorías a solo dos (sano y enfermo) se basa en investigaciones que sugieren que simplificar la clasificación puede mejorar el rendimiento del modelo al facilitar el aprendizaje y la generalización. Según un estudio realizado por Babaqi et al. (2023), al reducir el número de clases en problemas de clasificación médica, se permite que los modelos se concentren en aprender características más relevantes y distintivas entre las clases principales; esto es crucial cuando se trabaja con conjuntos de datos que pueden tener un número limitado de ejemplos por clase. Este enfoque también ayuda a mitigar problemas asociados con el desequilibrio entre clases, ya que algunas patologías pueden estar subrepresentadas en comparación con otras; al enfocarse únicamente en la clasificación binaria, se puede optimizar el rendimiento del modelo antes de abordar la complejidad adicional que implica clasificar múltiples (Gong, 2023)

7.3.2. Hiperparámetros y Configuraciones Ciclo 1

Para el desarrollo de este ciclo, se utilizaron los siguientes hiperparámetros y configuraciones:

**Tabla 13.**

Hiperparámetros ciclo 1

|  |  |
| --- | --- |
| **Hiperparámetro** | **Configuración** |
| Learning Rates | Tres tasas de aprendizaje: 0.01, 0.001 y 0.0001 |
| Batch Size | 16 para todas las pruebas |
| Número de Épocas | 40 épocas para cada modelo |
| Repeticiones | 10 repeticiones para cada tasa de aprendizaje |
| Función de Pérdida | Entropía cruzada (Cross-Entropy Loss) para la clasificación |
| Función de Activación | Softmax en la capa de salida, adecuada para la clasificación multiclase |
| Optimización | Optimizador Adam, conocido por su eficiencia y adaptación del learning rate durante el entrenamiento |
| Validación Cruzada | Enfoque de validación cruzada implementado para evaluar el desempeño del modelo y reducir el riesgo de sobreajuste |

7.3.3 Ajuste de paciencia Ciclo 1

Como se había mencionado en capítulos anteriores, en este ciclo se utilizó una configuración del parámetro de "paciencia" (patience) que permite reducir la tasa de aprendizaje si la precisión de validación no mejora después de 3 épocas. Si la precisión no mejora después de 5 épocas, el entrenamiento se detiene. Esta técnica ayudó a prevenir el sobreajuste y a mantener la eficiencia del modelo.

El ajuste de la tasa de aprendizaje y la implementación de paciencia ayudaron a prevenir el sobreajuste. En algunos casos, el modelo no alcanzó las 40 épocas completas debido a esto, lo cual resultó en la finalización anticipada del entrenamiento cuando no se observó mejora significativa en la precisión de validación o la pérdida de validación.

7.3.4 Análisis de resultados Ciclo 1

Se describe, en esta sección, el análisis de los resultados obtenidos mediante tablas de diferentes categorías y gráficos, según la arquitectura evaluada:

7.3.4.1 VGG19

Varianza General:

**Tabla 14.**

Varianza general VGG19 ciclo 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **learning\_rate** | **varianza\_val\_loss** | **varianza\_val\_accuracy** | **Promedio\_val\_accuracy** | **Promedio\_val\_loss** |
| 1e-4 | 5,1E+09 | 7.14E-05 | 0.94481 | 1.03814 |
| 1e-3 | 6,8E+08 | 0.00021 | 0.92722 | 0.28714 |
| 1e-2 | 1,3E+09 | 0.00516 | 0.85277 | 0.35048 |

Resumen General:

**Tabla 15.**

Resumen general VGG19 ciclo 1

|  |  |  |  |
| --- | --- | --- | --- |
| **learning\_rate** | **repeticion** | **mejor\_val\_accuracy** | **mejor\_val\_loss** |
| 0.0001 | 6 | 0.957 | 0.9441 |
| 0.001 | 3 | 0.937 | 0.2675 |
| 0.01 | 6 | 0.928 | 0.2323 |

Discusión:

VGG19 mostró un rendimiento sólido con tasas de aprendizaje menores, como se evidencia en los resultados, se destacó por una alta precisión y baja pérdida de validación. Su capacidad para capturar características es notable, aunque la varianza aumenta con tasas de aprendizaje mayores.

VGG19 es conocida por su simplicidad en diseño, con capas convolucionales apiladas de manera uniforme (Simonyan, 2014). En los experimentos se evidencian resultados significativos como:

Val\_accuracy (Precisión en Validación): Consistentemente alta, alcanzando hasta un 0.957 en las mejores repeticiones, siendo el tercer mejor valor entre las cinco arquitecturas.

Val\_loss (Valor de pérdida en Validación): Mostró una baja pérdida de validación, esto indica una buena capacidad para generalizar sobre el conjunto de prueba.

7.3.4.2 ResNet50

Varianza General:

**Tabla 16**.

Varianza general Resnet50 ciclo 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **learning\_rate** | **varianza\_val\_loss** | **varianza\_val\_accuracy** | **promedio\_val\_accuracy** | **Promedio\_val\_loss** |
| 1e-4 | 2,7E+09 | 2.00E-05 | 0.95349 | 1.10462 |
| 1e-3 | 6,5E+08 | 5.74E-05 | 0.95662 | 0.24710 |
| 1e-2 | 1,3E+09 | 4,91E+09 | 0.926746 | 0.23518 |

Resumen General:

**Tabla 17.**

Resumen general Resnet50 ciclo 1

|  |  |  |  |
| --- | --- | --- | --- |
| **learning\_rate** | **repeticion** | **mejor\_val\_accuracy** | **mejor\_val\_loss** |
| 1e-4 | 7 | 0.964 | 0.3002 |
| 1e-3 | 3 | 0.966 | 0.2126 |
| 1e-2 | 1 | 0.882 | 0.3625 |

Discusión:

ResNet50 presentó un rendimiento superior con tasas de aprendizaje más bajas, mostrando una alta precisión y baja pérdida de validación. Su arquitectura permite el manejo eficiente de características profundas, esto se traduce en un buen desempeño en tareas de clasificación (Ortiz, 2023)

ResNet50 es conocida por sus bloques residuales, que permiten el flujo de gradiente a través de capas más profundas. (Ortiz, 2023) Se evidencia lo siguiente:

Val\_accuracy (Precisión en Validación):Alcanzó consistentemente valores altos, hasta 0.966, siendo el segundo mejor.

Val\_loss (Pérdida en Validación): Se mantuvo baja, mostrando una gran capacidad para generalizar sin caer en problemas de sobreajuste.

7.3.4.3 EfficientNet (Versión B3)

Varianza General:

**Tabla 18.**

Varianza general Efficientnet ciclo 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **learning\_rate** | **varianza\_val\_loss** | **varianza\_val\_accuracy** | **Promedio\_val\_accuracy** | **Promedio\_val\_loss** |
| 1e-2 | 6,4E+09 | 1,73E+09 | 0.79590 | 0.46728 |
| 1e-3 | 2,0E+09 | 7.32E-05 | 0.95638 | 0.28973 |
| 1e-4 | 9,9E+09 | 1,14E+09 | 0.96530 | 0.36182 |

Resumen general:

**Tabla 19.** Resumen general Efficientnet ciclo 1

|  |  |  |  |
| --- | --- | --- | --- |
| **learning\_rate** | **repeticion** | **mejor\_val\_accuracy** | **mejor\_val\_loss** |
| 0.01 | 3 | 0.942 | 0.1917 |
| 0.001 | 3 | 0.969 | 0.239 |
| 0.0001 | 2 | 0.959 | 0.6561 |

Discusión:

EfficientNet mostró una excelente precisión con tasas de aprendizaje de 0.001 y 0.0001, su versión 3 (B3) utiliza un enfoque de "compound scaling" que ajusta simultáneamente la profundidad, el ancho y la resolución de la red, esto resulta en una precisión superior (Elmoufidi, 2021). A pesar de la variación en pérdida de validación con la tasa más baja, su eficiencia en términos de parámetros y rendimiento es notable.

EfficientNet es una arquitectura más reciente que ajusta automáticamente la resolución de entrada, la profundidad y el número de filtros para optimizar el rendimiento, además de destacarse por su buen rendimiento en sistemas con limitaciones computacionales. (Elmoufidi, 2021). En los experimentos se demostró que:

Val\_accuracy (Precisión en Validación): EfficientNet fue la más consistente, con valores que llegaron hasta 0.969. siendo el mejor.

Val\_loss (Valor de Pérdida en Validación): Fue el modelo con la menor pérdida en validación, mostrando un excelente equilibrio entre capacidad de generalización y precisión.

7.3.4.4 Xception

Varianza General:

**Tabla 20.**

Varianza general Xception ciclo 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **learning\_rate** | **varianza\_val\_loss** | **varianza\_val\_accuracy** | **promedio\_val\_accuracy** | **promedio\_val\_loss** |
| 1e-2 | 2,3E+08 | 0.00904 | 0.85566 | 0.36127 |
| 1e-4 | 6,1E+09 | 2.26E-05 | 0.94481 | 0.55630 |
| 1e-3 | 2,3E+09 | 2.61E-05 | 0.94433 | 0.27205 |

**Resumen General:**

**Tabla 21.**

Resumen general Xception ciclo 1

|  |  |  |  |
| --- | --- | --- | --- |
| **learning\_rate** | **repeticion** | **mejor\_val\_accuracy** | **mejor\_val\_loss** |
| 1e-2 | 2 | 0.94 | 0.2086 |
| 1e-4 | 6 | 0.949 | 0.5279 |
| 1e-3 | 8 | 0.947 | 0.2614 |

Discusión:

Xception es una arquitectura que extiende la idea de las convoluciones separables en profundidad, optimizando el flujo de información a través de un diseño más eficiente en términos de computación (Lim, 2022) A pesar de esto, en los experimentos realizados con las imágenes de retinoscopía, los resultados obtenidos no fueron los mejores comparados con otras arquitecturas. En los experimentos se evidenció que:

Val\_accuracy (Precisión en Validación): Los valores de accuracy en Xception se mostraron inconsistentes. Aunque en algunas repeticiones lograba valores superiores a 0.80, había una alta variabilidad entre los experimentos. En general, no superó el rendimiento de EfficientNet o ResNet50, y su estabilidad fue inferior.

Val\_loss (Valor de pérdida en validación): El valor de la pérdida en validación también fluctuaba demasiado, lo que indicaba que el modelo tenía problemas de sobreajuste y generalización con los datos de validación, presentando una pérdida significativamente más alta que otros modelos después de 40 épocas.

Conclusión: Aunque es un modelo potente y ha demostrado eficacia en otros dominios, en esta tarea específica no mostró un rendimiento lo suficientemente competitivo, por ello su exclusión para el siguiente ciclo.

7.3.4.5 InceptionV3

Varianza General:

**Tabla 22.**

Varianza general InceptionV3 ciclo 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **learning\_rate** | **varianza\_val\_loss** | **varianza\_val\_accuracy** | **Promedio\_val\_accuracy** | **Promedio\_val\_loss** |
| 1e-2 | 3.6E-05 | 5.23E-07 | 0.73999 | 0.57976 |
| 1e-3 | 0.04614 | 4.55E-05 | 0.94072 | 0.40569 |
| 1e-4 | 1.07494 | 8.65E-05 | 0.93710 | 2.02466 |

**Resumen General:**

**Tabla 23.**

Resumen general InceptionV3 ciclo 1

|  |  |  |  |
| --- | --- | --- | --- |
| **learning\_rate** | **repeticion** | **mejor\_val\_accuracy** | **mejor\_val\_loss** |
| 1e-2 | 7 | 0.74 | 0.5913 |
| 1e-3 | 6 | 0.949 | 0.3377 |
| 1e-4 | 9 | 0.933 | 0.7647 |

Discusión:

InceptionV3 es una de las versiones más avanzadas de la arquitectura Inception que utiliza una mezcla de convoluciones de diferentes tamaños para capturar características a varias escalas. (Lim, 2022) No obstante, en esta aplicación específica de clasificación de imágenes de retinoscopía, los resultados fueron inferiores comparados con VGG19, ResNet50 y EfficientNet. Los experimentos mostraron que:

Val\_accuracy (Precisión en validación): InceptionV3 mostró una precisión moderada, con valores que oscilaban alrededor de 0.74 en su tasa de aprendizaje más alta, lo que lo colocaba por debajo de las arquitecturas elegidas para los experimentos finales. Aunque en ciertas pruebas se observó una convergencia rápida, no logró superar a EfficientNet o ResNet50 en consistencia o precisión final.

Val\_loss (Valor de pérdida en validación): La pérdida en validación fue más estable que en Xception, pero tampoco mostró una mejora significativa respecto a VGG19 o ResNet50. En algunos casos, el modelo parecía estancarse, y parece no estar aprovechando por completo su capacidad de clasificación de los datos de retinoscopía.

Conclusión: Aunque el rendimiento fue relativamente bueno, no logró superar las tres arquitecturas seleccionadas en términos de precisión o estabilidad, lo que justificó su exclusión.

7.4 Conclusiones y Justificación de Selección Ciclo 1

7.4.1 Conclusiones con respecto a las tablas de varianza y tablas generales

A partir lo anterior concluimos que:

ResNet50 ofrece un manejo eficiente de características profundas y logra una alta precisión, especialmente con tasas de aprendizaje más bajas. Durante los experimentos, mostró una baja varianza en los resultados de validación, mostrando una buena capacidad para generalizar. Los bloques residuales de su arquitectura permiten entrenar redes profundas sin comprometer el rendimiento, esto es una excelente opción para la clasificación de imágenes complejas como las de retinoscopía. (Ortiz, 2023) ResNet50 fue seleccionada por su equilibrio entre precisión, estabilidad y capacidad para manejar características profundas, destacando su bajo riesgo de sobreajuste en comparación con otras arquitecturas.

EfficientNet se distingue por su enfoque en la eficiencia para lograr un rendimiento competitivo con menos parámetros. Su alta precisión y buena estabilidad con tasas de aprendizaje más bajas son idóneas para la clasificación de imágenes de retinoscopía. Adicionalmente, EfficientNet optimiza la resolución de entrada, la profundidad y los filtros para obtener obtener un gran rendimiento incluso en sistemas con limitados recursos computacionales. (Elmoufidi, 2021) EfficientNet fue seleccionada por su excelente equilibrio entre eficiencia y rendimiento, sobresaliendo por su capacidad de ofrecer alta precisión.

VGG19 mostró una alta precisión con tasas de aprendizaje bajas, aunque presentó una mayor varianza en la pérdida con tasas de aprendizaje más altas. A pesar de esta variabilidad, sigue siendo una opción sólida gracias a su consistencia en tasas más bajas y su habilidad para capturar características profundas. Es evidente que la simplicidad de su arquitectura facilita su ajuste, por ello fue seleccionada por su rendimiento constante y su capacidad para captar características profundas, siendo una opción confiable para este tipo de problemas, pese a la mayor variabilidad observada en condiciones más exigentes.

Xception e InceptionV3 también lograron buenos resultados, pero mostraron una mayor variabilidad en sus métricas de desempeño. Su sensibilidad a las tasas de aprendizaje; en particular, afectó su consistencia en comparación con las arquitecturas seleccionadas. Aunque tienen ventajas en otros dominios, para esta tarea específica no ofrecieron la misma estabilidad ni precisión que ResNet50, EfficientNet y VGG19.

7.4.2 Conclusiones con respecto a las tablas de métricas de cada arquitectura y matrices de confusión:

Adicionalmente y a partir de las matrices de confusión y las tablas de métricas para los modelos finales de este ciclo 1 (ResNet50, VGG19 y EfficientNet), se puede realizar el siguiente análisis comparativo:

**Tabla 24.**

Tabla de métricas Resnet 50 ciclo 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | f1-score | support |
| Enfermo | 0.95 | 0.95 | 0.95 | 308 |
| Sano | 0.85 | 0.87 | 0.86 | 107 |
| accuracy |  |  | 0.93 | 415 |
| macro avg | 0.90 | 0.91 | 0.91 | 415 |
| weighted avg | 0.93 | 0.93 | 0.93 | 415 |

**Tabla 25.**

Tabla de métricas VGG19 ciclo1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| Enfermo | 0.97 | 0.96 | 0.97 | 308 |
| Sano | 0.90 | 0.91 | 0.90 | 107 |
| accuracy |  |  | 0.95 | 415 |
| macro avg | 0.93 | 0.94 | 0.93 | 415 |
| weighted avg | 0.95 | 0.95 | 0.95 | 415 |

**Tabla 26.**

Tabla de métricas Efficientnet ciclo 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| Enfermo | 0.99 | 0.99 | 0.99 | 308 |
| Sano | 0.98 | 0.98 | 0.98 | 107 |
| accuracy |  |  | 0.99 | 415 |
| macro avg | 0.99 | 0.99 | 0.99 | 415 |
| weighted avg | 0.99 | 0.99 | 0.99 | 415 |

7.4.2.1 Precisión

ResNet50: La precisión del modelo es del 93% en general. El modelo es capaz de identificar correctamente a los pacientes con enfermedades en el 95% de los casos, pero para los pacientes sanos, la precisión cae al 85%.

VGG19: La precisión mejora ligeramente en comparación con ResNet50, alcanzando un 95%. Para la clase enfermo, se mantiene alta en 97%, y para la clase sanos es del 90%.

EfficientNet: Es el modelo con la mayor precisión, logrando un 99%. Tanto para enfermos como sanos, la precisión es sobresaliente con un 99% y 98% respectivamente.

7.4.2.2 Recall (Sensibilidad)

ResNet50: El recall del modelo es 0.95 para los pacientes enfermos, pero es algo más bajo para los pacientes sanos, con un valor de 0.87. Esto sugiere que aunque ResNet50 es muy bueno identificando a pacientes enfermos, tiene más dificultades para detectar correctamente a los pacientes sanos.

VGG19: El recall general es 0.95. Para los pacientes enfermos, es 0.96 y para los pacientes sanos es 0.91. Esto indica que VGG19 es más equilibrado, logrando mejorar la detección de pacientes sanos en comparación con ResNet50.

EfficientNet: El mejor rendimiento en términos de recall también es para EfficientNet, con un 0.99 para pacientes enfermos y un 0.98 para pacientes sanos, lo que refleja su capacidad de generalización y su equilibrio al detectar ambas clases.

7.4.2.3 F1-Score

ResNet50: El f1-score global del modelo es 0.93, con valores de 0.95 para pacientes enfermos y 0.86 para sanos. Aunque el modelo tiene un rendimiento sólido, presenta una ligera caída al tratar de clasificar pacientes sanos.

VGG19: Este modelo tiene un f1-score general de 0.95. Al igual que con la precisión y el recall, VGG19 logra mejores resultados que ResNet50, con un f1-score de 0.97 para los enfermos y 0.90 para los sanos.

EfficientNet: EfficientNet es el que mejor se desempeña, con un f1-score global de 0.99. Esto se debe a su excelente equilibrio entre precisión y recall, con 0.99 para pacientes enfermos y 0.98 para sanos.

7.4.2.4 Exactitud General (Accuracy)

ResNet50: La exactitud global es del 93%, lo que muestra un buen rendimiento, aunque no el mejor entre los modelos evaluados.

VGG19: Mejora en comparación con ResNet50, logrando una exactitud del 95%.

EfficientNet: Este modelo destaca con una exactitud del 99%, lo que lo convierte en el más robusto y eficiente en la clasificación de las imágenes de retinoscopía.

7.4.2.5 Comparación basada en las matrices de confusión:

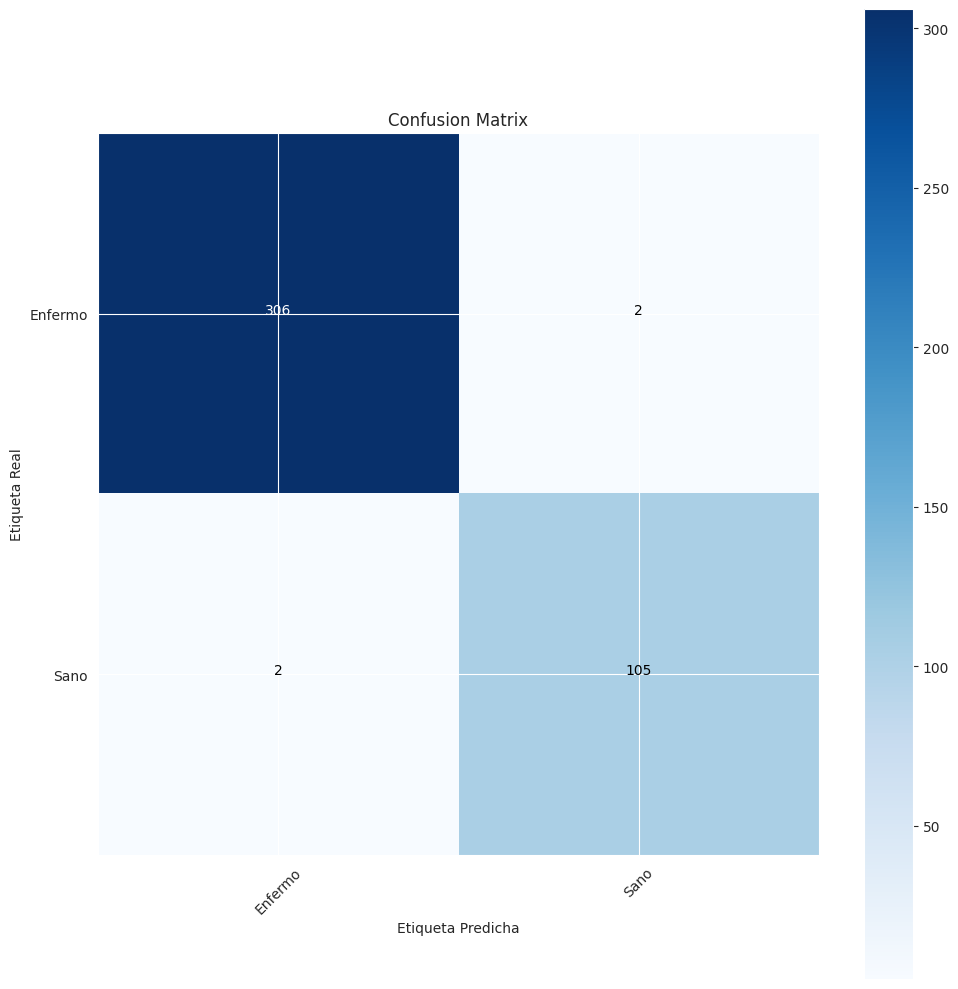
Figura 15. Matriz de Confusión de Resnet50 ciclo 1



Figura 16. Matriz de confusión de VGG19 ciclo 1



Figura 17. Matriz de confusión EfficientnetB3 ciclo 1



ResNet50:

Clasifica correctamente 297 pacientes enfermos, con solo 11 clasificaciones erróneas.

Para los pacientes sanos, clasifica correctamente 97, con 10 falsos negativos, lo que indica cierta dificultad al detectar sanos.

VGG19:

Clasifica correctamente 292 pacientes enfermos y comete 16 errores, ligeramente peor que ResNet50 en esta clase.

Sin embargo, clasifica 93 pacientes sanos correctamente, con 14 falsos negativos.

EfficientNet:

Clasifica correctamente a 306 pacientes enfermos, con solo 2 errores, y es el que mejor rendimiento tiene en esta clase.

Clasifica correctamente a 105 pacientes sanos, con solo 2 falsos negativos, superando tanto a ResNet50 como a VGG19 en ambas clases.

7.5 Selección de Arquitecturas para la Clasificación del Ciclo 2

7.5.1 Dataset utilizado Ciclo 2

Con el fin de evaluar la capacidad de los modelos para distinguir entre cada una de estas patologías, se decidió realizar la clasificación en cuatro categorías: normal, retinopatía diabética, glaucoma, catarata. El total de imágenes para la clase normal es de 1074, 1098 para retinopatía diabética, 1007 para glaucoma y 960 para catarata. La decisión de usar el dataset con cuatro clases yace en que, al aprender patrones más complejos y variados dentro de cada categoría, los modelos se vuelven más robustos frente a la variabilidad en las imágenes médicas (Gong et al., 2023).. Investigaciones previas, como las de Gulshan et al. (2016) y Kermany et al. (2018), han demostrado que un mayor número de clases puede ayudar a mejorar la capacidad del modelo para generalizar a nuevas imágenes que no ha visto antes; esto es especialmente imprescindible en aplicaciones médicas donde la diversidad en las presentaciones clínicas puede ser alta.

7.5.2. Hiperparámetros y Configuraciones

Para el desarrollo de este ciclo, se utilizaron los siguientes hiperparámetros y configuraciones:

**Tabla 27.**

Configuración hiperparámetros ciclo 2

|  |  |
| --- | --- |
| **Hiperparámetro** | **Configuración** |
| Learning Rates | 0.001 para EfficientNet y ResNet50; 0.0001 para VGG19 |
| Batch Size | 16 para todas las pruebas |
| Número de Épocas | 60 épocas para cada modelo |
| Repeticiones | 5 repeticiones para cada tasa de aprendizaje (reducido debido al tiempo limitado para realizar los experimentos) |
| Función de Pérdida | Entropía cruzada (Cross-Entropy Loss) para la clasificación |
| Función de Activación | Softmax en la capa de salida, adecuada para la clasificación multiclase |
| Optimización | Optimizador Adam, conocido por su eficiencia y adaptación del learning rate durante el entrenamiento |
| Validación Cruzada | Enfoque de validación cruzada implementado para evaluar el desempeño del modelo y reducir el riesgo de sobreajuste |

7.5.3 Ajuste de paciencia Ciclo 2

En este estudio, se utilizó una configuración de "paciencia" (patience) que permite reducir la tasa de aprendizaje si la precisión de validación no mejora después de 5 épocas. Si la precisión de validación no mejora después de 10 épocas, el entrenamiento se detiene. Esta técnica ayudó a prevenir el sobreajuste y a mantener la eficiencia del modelo, evidenciado en los resultados obtenidos.

7.5.4. Análisis de Resultados

7.5.4.1 VGG19

Varianza General:

**Tabla 28.**

Tabla repeticiones VGG19 ciclo 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **learning\_rate** | **varianza\_val\_loss** | **varianza\_val\_accuracy** | **Promedio\_val\_accuracy** | **Promedio\_val\_loss** |
| 1e-4 | 5,8E+09 | 8.6E-06 | 0.96911 | 0.41316 |

**Resumen General:**

**Tabla 29.**

Resumen general VGG19 ciclo 2

|  |  |  |  |
| --- | --- | --- | --- |
| **learning\_rate** | **repeticion** | **mejor\_val\_accuracy** | **mejor\_val\_loss** |
| 1e-4 | 3 | 0,937 | 0.3347 |

**Resumen resultados 0.0001:**

**Tabla 30.**

Resumen Resultados 0.0001

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **repeticion** | **max\_val\_accuracy** | **epoch\_max\_val\_accuracy** | **min\_val\_loss** | **epoch\_min\_val\_loss** |
| 1 | 0.97059 | 12 | 0.42501 | 59 |
| 2 | 0.97058 | 12 | 0.40113 | 58 |
| 3 | 0.97060 | 9 | 0.33470 | 59 |
| 4 | 0.96323 | 8 | 0.55172 | 58 |
| 5 | 0.970588 | 13 | 0.35324 | 41 |

Gráfica pérdida y accuracy en validación vs pérdida y accuracy en entrenamiento de la mejor repetición

Figura 18. Gráfica pérdida y accuracy en validación vs pérdida y accuracy en entrenamiento de la mejor repetición VGG19

****

Discusión:

La arquitectura VGG19 mostró resultados consistentes, alcanzando una precisión máxima de validación del 93.7% en la tercera repetición, con una pérdida mínima de validación de 0.3347. Los valores de varianza, tanto en precisión de validación como en pérdida de validación, fueron bajos: 8.65E-06 y 0.0058, respectivamente; estos valores lo posicionan en el segundo puesto en términos de varianza de pérdida de validación, último puesto en términos de mejor pérdida de validación, y segundo puesto en términos de precisión, comparado con las otras dos arquitecturas.

Observando la gráfica de su mejor repetición se puede concluir que, de los tres modelos, este fue el que presentó más variación en los valores de pérdida y accuracy, a través de las épocas, y también se resalta que fue el modelo que más se demoró en hallar una convergencia (terminar su entrenamiento); es decir, que requirió más épocas para poder finalizar su entrenamiento.

A pesar de su alto rendimiento, los valores de pérdida en algunas repeticiones fueron más elevados, lo que sugiere que la arquitectura podría estar más sensible a ciertos patrones en los datos (Simonyan, 2014). Cabe destacar un punto a favor de la arquitectura, y es que la simplicidad del diseño de VGG19, basada en capas convolucionales apiladas, permite una fácil adaptación y ajuste del modelo (Simonyan, 2014) sin embargo, las variaciones en la pérdida de validación sugieren que existe una menor estabilidad comparada con las otras arquitecturas evaluadas.

7.5.4.2 ResNet50

Varianza General:

**Tabla 31.**

Varianza general Resnet50 ciclo 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **learning\_rate** | **varianza\_val\_loss** | **varianza\_val\_accuracy** | **Promedio\_val\_accuracy** | **Promedio\_val\_loss** |
| 1e-3 | 9,7E+09 | 6.0E-06 | 0.95024 | 0.34588 |

**Resumen General:**

**Tabla 32**.

Resumen general Resnet50 ciclo 2

|  |  |  |  |
| --- | --- | --- | --- |
| **learning\_rate** | **repeticion** | **mejor\_val\_accuracy** | **mejor\_val\_loss** |
| 0.001 | 1 | 0.952 | 0.2899 |

**Resumen resultados 0.001:**

**Tabla 33.**

Tabla repeticiones Resnet50 ciclo 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **repeticion** | **max\_val\_accuracy** | **epoch\_max\_val\_accuracy** | **min\_val\_loss** | **epoch\_min\_val\_loss** |
| 1 | 0.954106 | 23 | 0.28990 | 29 |
| 2 | 0.951690 | 21 | 0.33731 | 21 |
| 3 | 0.946859 | 15 | 0.36968 | 15 |
| 4 | 0.949276 | 14 | 0.37827 | 14 |
| 5 | 0.949275 | 13 | 0.35425 | 21 |

Gráfica pérdida y accuracy en validación vs pérdida y accuracy en entrenamiento de la mejor repetición

Figura 19. Gráfica pérdida y accuracy en validación vs pérdida y accuracy en entrenamiento de la mejor repetición Resnet50

Discusión:

ResNet50 alcanzó una precisión de validación máxima del 95.2% con una pérdida de validación mínima de 0.2899 en la primera repetición. La varianza en la precisión de validación fue extremadamente baja (6.07E-06), lo que indica que el modelo mantuvo un rendimiento constante a través de las diferentes repeticiones, estos valores lo posicionan en el primer puesto en términos de varianza de pérdida de validación, segundo puesto en términos de mejor pérdida de validación, y último puesto en términos de precisión, comparado con las otras dos arquitecturas. Algunos valores de pérdida de validación fueron más altos, esto sugiere que podría haber margen para una mayor optimización en los parámetros del modelo.

Observando la gráfica se puede concluir que este modelo requirió menos épocas para hallar una convergencia, esto se evidencia también en el hecho de que sus mejores valores de pérdida y accuracy se encuentran en la época 30 y 24 respectivamente.

Cabe destacar un punto a favor de la arquitectura; su diseño está caracterizado por sus bloques residuales, esto facilita la transmisión de gradientes a través de sus capas más profundas, generalmente dando como resultado una excelente precisión sin problemas de sobreajuste (Vipas.Ai, 2024) adicionalmente, este modelo demuestra, en la mayoría de casos, una capacidad estable para manejar características complejas en las imágenes de retina, con una baja varianza en sus resultados y un buen equilibrio entre precisión y pérdida de validación (Ortiz, 2023)

7.5.4.3 EfficientNet (Versión B3)

Varianza General:

**Tabla 34.**

Varianza general Efficientnet ciclo 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **learning\_rate** | **varianza\_val\_loss** | **varianza\_val\_accuracy** | **Promedio\_val\_accuracy** | **Promedio\_val\_loss** |
| 1e-3 | 1,0E+09 | 3.6E-05 | 0.95217 | 0.30793 |

**Resumen General:**

**Tabla 35.**

Resumen general Efficientnet ciclo 2

|  |  |  |  |
| --- | --- | --- | --- |
| **learning\_rate** | **repeticion** | **mejor\_val\_accuracy** | **mejor\_val\_loss** |
| 0.001 | 3 | 0.959 | 0.26 |

**Resumen resultados 0.001:**

**Tabla 36.**

Repeticiones Efficientnet ciclo 2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **repeticion** | | **max\_**  **val\_accuracy** | **epoch\_**  **max\_val\_**  **accuracy** | **min\_**  **val\_loss** | **epoch\_**  **min\_val\_**  **loss** |
| 1 | 0.9541062713 | 13 | 0.3016205132 | 18 |
| 2 | 0.9541062713 | 17 | 0.3219943941 | 25 |
| 3 | 0.9613526464 | 25 | 0.2600095868 | 26 |
| 4 | 0.944444418 | 16 | 0.3570893109 | 16 |
| 5 | 0.9468598962 | 11 | 0.2989785075 | 21 |

Gráfica pérdida y accuracy en validación vs pérdida y accuracy en entrenamiento de la mejor repetición

****Figura 20. Gráfica pérdida y accuracy en validación vs pérdida y accuracy en entrenamiento de la mejor repetición Efficientnet ciclo 2

Discusión:

EfficientNet se destacó como la mejor arquitectura evaluada, presentando los mejores resultados en cuanto a precisión y pérdida de validación. Con una tasa de aprendizaje de 0.001, este modelo mostró una precisión constante y una pérdida de validación más baja que las demás arquitecturas, incluso si resultó obtener el último puesto en términos de varianza de pérdida de validación.

Observando la gráfica se puede concluir que comparte una similitud con VGG19: Sus variaciones resultan extremas a lo largo de su entrenamiento y validación; sin embargo, este converge en menos épocas.

Debido a la manera en la que EfficientNet ajusta de manera eficiente la resolución de entrada, la profundidad y el número de filtros, esta arquitectura es particularmente eficiente y adecuada para su implementación en sistemas con limitaciones computacionales (Elmoufidi, 2021)

7.6. Conclusiones y Justificación de Selección Ciclo 2

7.6.1 Conclusiones Ciclo 2 con respecto a las tablas de varianza, tablas generales y tablas por learning rate

A partir lo anterior concluimos que:

EfficientNet fue seleccionada por su excelente rendimiento general y su eficiencia en la utilización de recursos. Con los mejores resultados tanto en precisión como en pérdida de validación, es ideal para la clasificación de enfermedades oculares, especialmente en entornos con limitaciones de recursos computacionales.

ResNet50 presenta un equilibrio entre precisión y pérdida de validación. Su arquitectura residual permite un entrenamiento más eficiente y estable, especialmente en tareas que requieren la extracción de características complejas. Sin embargo, en algunas repeticiones mostró una pérdida de validación ligeramente más alta que EfficientNet, esta inestabilidad fue un factor clave para no seleccionarla como la arquitectura final para el ciclo 3.

VGG19, a pesar de que destaca por su simplicidad y capacidad de ofrecer resultados de alto rendimiento, mostró algunas variaciones en la pérdida, esto no lo hace un candidato final para modelo final en el ciclo 3.

7.6.2 Conclusiones Ciclo 2 con respecto a las tablas de métricas de cada arquitectura y matrices de confusión

Adicionalmente y a partir de las matrices de confusión y las tablas de métricas para los modelos ResNet50, VGG19 y EfficientNet (las cuales se encontrarán en la hoja de cálculo adjunta en Apéndice A) , se puede realizar el siguiente análisis comparativo:

Figura 21. Matriz de confusión efficientnetB3 ciclo 2

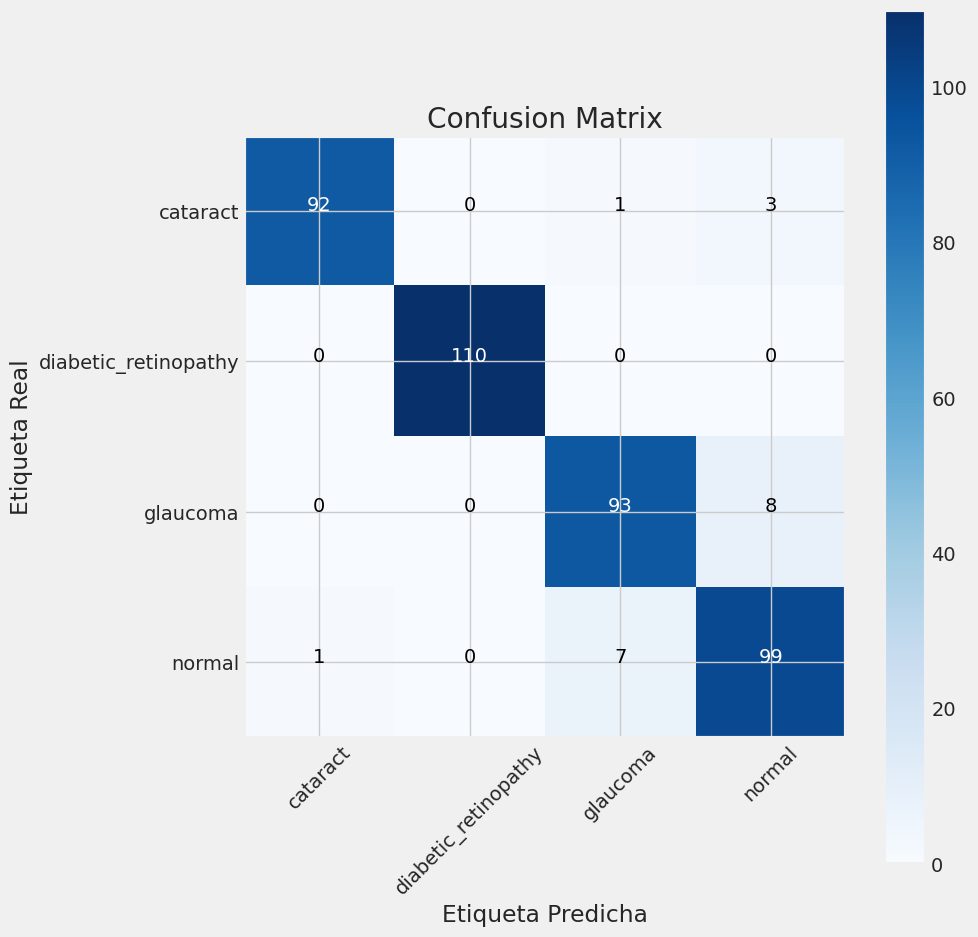


Figura 22. Matriz de confusión VGG19 ciclo 2

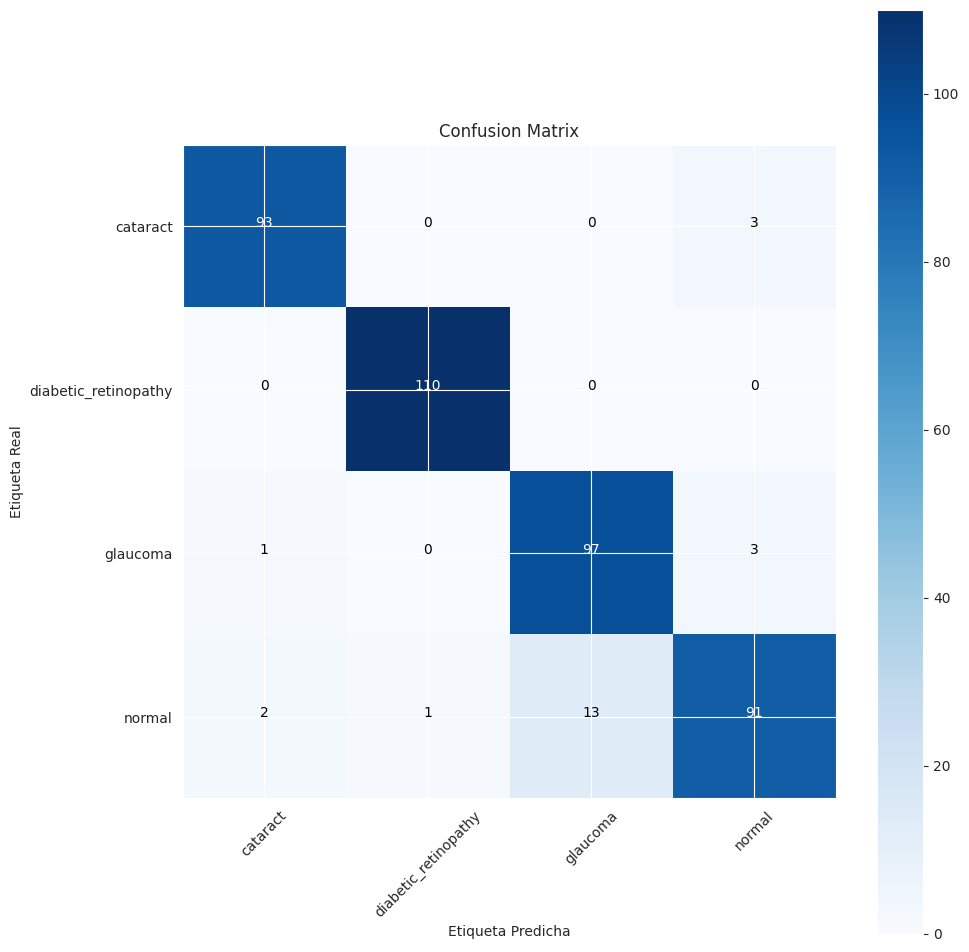
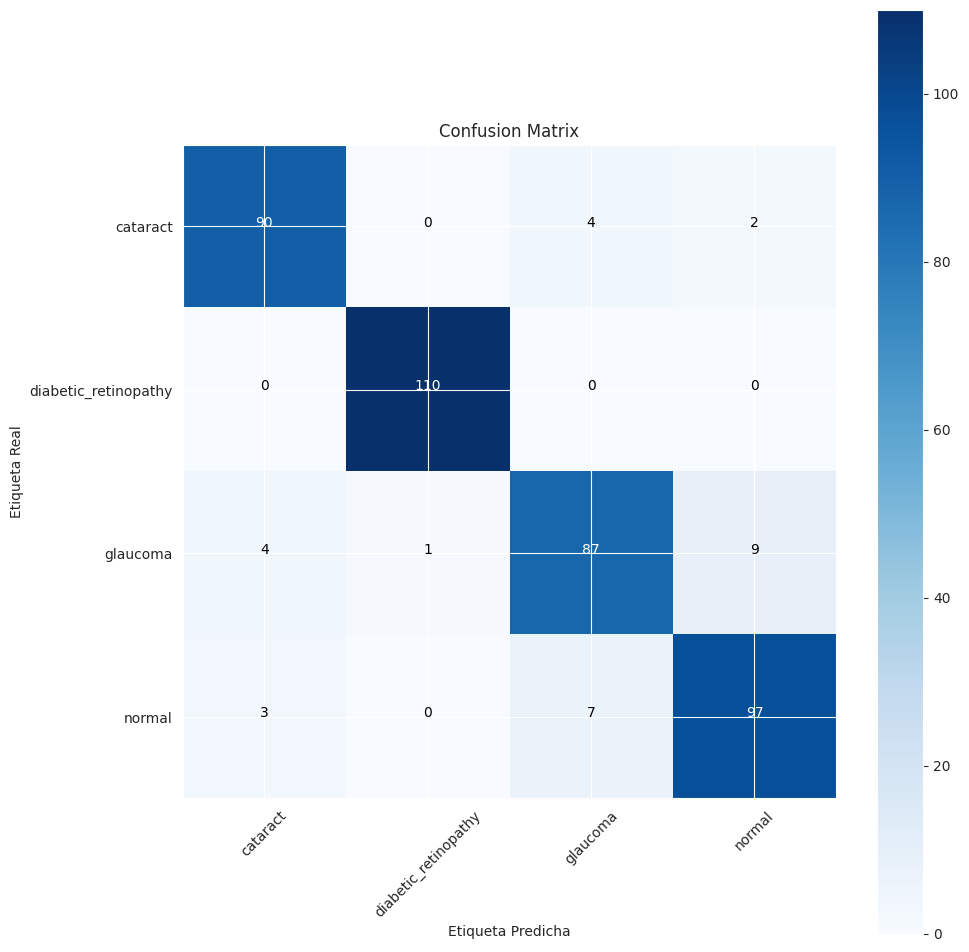


Figura 23. Matriz de confusión Resnet50 ciclo 2



**Tabla 37.**

Tabla de métricas Resnet50 ciclo 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| Cataract | 0,93 | 0,94 | 0,93 | 96 |
| diabetic\_retinopathy | 0,99 | 1,00 | 1,00 | 110 |
| glaucoma | 0,89 | 0,86 | 0,87 | 101 |
| Normal | 0,90 | 0,91 | 0,90 | 107 |
| accuracy |  |  | 0,93 | 414 |
| Macro avg | 0,93 | 0,93 | 0,93 | 414 |
| weighted avg | 0,93 | 0,93 | 0,93 | 414 |

**Tabla 38.**

Tabla de métricas EfficientnetB3 ciclo 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| Cataract | 0,99 | 0,96 | 0,97 | 96 |
| diabetic\_retinopathy | 1,00 | 1,00 | 1,00 | 110 |
| glaucoma | 0,92 | 0,92 | 0,92 | 101 |
| Normal | 0,90 | 0,93 | 0,91 | 107 |
| accuracy |  |  | 0,95 | 414 |
| Macro avg | 0,95 | 0,95 | 0,95 | 414 |
| weighted avg | 0,95 | 0,95 | 0,95 | 414 |

**Tabla 39.**

Tablas de métricas VGG19 ciclo 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| Cataract | 0.97 | 0.97 | 0.97 | 96 |
| diabetic\_retinopathy | 0.99 | 1.00 | 1.00 | 110 |
| glaucoma | 0.88 | 0.96 | 0.92 | 101 |
| Normal | 0,94 | 0.85 | 0.89 | 107 |
| accuracy |  |  | 0.94 | 414 |
| Macro avg | 0.94 | 0.94 | 0.94 | 414 |
| weighted avg | 0.95 | 0.94 | 0.94 | 414 |

7.6.2.1. Exactitud General (Accuracy)

**Tabla 40.**

Modelo exactitud general

|  |  |
| --- | --- |
| **Modelo** | **Accuracy** |
| VGG19 | 94% |
| EfficientNet | 95% |
| ResNet50 | 93% |

El modelo EfficientNet logra la mayor exactitud global en comparación con VGG19 y ResNet50, aunque las diferencias no son grandes. El desempeño de VGG19 es muy cercano al de EfficientNet, mientras que ResNet50 tiene un rendimiento algo menor en términos de exactitud global.

7.6.2.2. Precisión por Clase

**Tabla 41.**

Precisión por Clase ciclo 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Modelo** | **Cataratas** | **Retinopatía diabética** | **Glaucoma** | **Normal** |
| VGG19 | 0.97 | 0.99 | 0.88 | 0.94 |
| EfficientNet | 0.99 | 1.00 | 0.92 | 0.90 |
| RestNet | 0.93 | 0.99 | 0.89 | 0.90 |

Catarata:

EfficientNet tiene la mejor precisión en la clase Catarata, mientras que VGG19 también muestra un buen desempeño. ResNet50 tiene la precisión más baja en esta clase.

Retinopatía Diabética:

Todos los modelos tienen un excelente desempeño en la clase de Retinopatía Diabética, siendo EfficientNet el único que alcanza una precisión perfecta de 1.00.

Glaucoma:

EfficientNet también supera a los otros dos modelos en la clase de Glaucoma, con una precisión de 0.92. VGG19 tiene la menor precisión para esta clase, con 0.88.

Normal:

En la clase Normal, VGG19 sobresale con una precisión de 0.94, superando tanto a EfficientNet como a ResNet50, que comparten una precisión de 0.90.

7.6.2.3. Recall por Clase

**Tabla 42.**

Recall por Clase ciclo 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Modelo** | **Cataratas** | **Retinopatia diabetica** | **Glaucoma** | **Normal** |
| VGG19 | 0.97 | 1.00 | 0.96 | 0.85 |
| EfficientNet | 0.96 | 1.00 | 0.92 | 0.93 |
| RestNet | 0.93 | 1.00 | 0.86 | 0.91 |

Catarata:

VGG19 tiene el mejor desempeño en términos de recall para la clase de Catarata, mientras que EfficientNet y ResNet50 tienen valores ligeramente más bajos.

Retinopatía Diabética:

Todos los modelos (VGG19, EfficientNet, ResNet50) tienen un recall perfecto de 1.00.

Glaucoma:

VGG19 destaca con un recall muy alto en la clase Glaucoma, con 0.96, mientras que ResNet50 tiene el valor más bajo con 0.86. EfficientNet se encuentra en un término medio con 0.92.

Normal:

Aquí, EfficientNet tiene el mejor recall con 0.93, superando tanto a ResNet50 como a VGG19, que queda con el valor más bajo (0.85).

7.6.2.4. F1-Score por Clase

**Tabla 43.**

F1-score por Clase ciclo 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Modelo** | **Cataratas** | **Retinopatía diabética** | **Glaucoma** | **Normal** |
| VGG19 | 0.97 | 1.00 | 0.92 | 0.89 |
| EfficientNet | 0.97 | 1.00 | 0.92 | 0.91 |
| RestNet | 0.93 | 1.00 | 0.87 | 0.90 |

Catarata:

EfficientNet y VGG19 empataron con un F1-score de 0.97 para la clase de Catarata, mientras que ResNet50 tiene un valor inferior (0.93).

Retinopatía Diabética:

Todos los modelos obtienen un F1-score de 1.00.

Glaucoma:

Tanto VGG19 como EfficientNet tienen un F1-score de 0.92 para Glaucoma, mientras que ResNet50 queda por detrás con 0.87.

Normal:

EfficientNet supera a los otros dos modelos con un F1-score de 0.91 en la clase Normal. VGG19 tiene el valor más bajo (0.89), mientras que ResNet50 se queda en 0.90.

7.6.2.5 Conclusiones Comparativas según matrices de confusión

EfficientNet parece ser el mejor modelo en las clases más problemáticas como Catarata y Glaucoma, esto lo convierte en el más ideal para su selección como modelo final.

VGG19 tiene un desempeño muy sólido, especialmente en Glaucoma y Normal, pero su exactitud general es inferior a la de EfficientNet.

ResNet50, aunque tiene un buen desempeño, queda rezagado en varias clases, especialmente en Glaucoma, y esto reduce su fiabilidad en comparación con EfficientNet y VGG19.

7.7 Selección de Arquitecturas para la Clasificación del Ciclo 3

En el Ciclo 3 se evaluó una arquitectura (EfficientNet) utilizando un dataset con imágenes de fondo de ojo con Batch Size 32, llevando a cabo 5 repeticiones, y 100 épocas en cada repetición; el modelo escogido fue evaluado utilizando una tasa de aprendizaje:  0.001; está y demás configuraciones se podrán evidenciar a más detalle en su respectiva sección. Se decidió realizar la clasificación en 4 categorías: Catarata, Glaucoma, Normal y Retinopatía Diabética

7.7.1 Dataset utilizado en Ciclo 3

Con el fin de evaluar la capacidad de los modelos para distinguir entre cada una de estas patologías, se decidió realizar la clasificación en cuatro categorías: normal, retinopatía diabética, glaucoma, catarata. El total de imágenes para la clase normal es de 1074, 1098 para retinopatía diabética, 1007 para glaucoma y 960 para catarata. La decisión de usar el dataset con cuatro clases yace en que, al aprender patrones más complejos y variados dentro de cada categoría, los modelos se vuelven más robustos frente a la variabilidad en las imágenes médicas (Gong, 2023) Investigaciones previas, como las de Gulshan et al. (2016) y Kermany et al. (2018), han demostrado que un mayor número de clases puede ayudar a mejorar la capacidad del modelo para generalizar a nuevas imágenes que no ha visto antes; esto es especialmente imprescindible en aplicaciones médicas donde la diversidad en las presentaciones clínicas puede ser alta.

7.7.1 Hiperparámetros y Configuraciones

Para el desarrollo de este ciclo, se utilizaron los siguientes hiperparámetros y configuraciones:

**Tabla 44.**

Configuración de hiperparámetros ciclo 3

|  |  |
| --- | --- |
| **Hiperparámetro** | **Configuración** |
| Learning Rates | 0.001 para EfficientNet |
| Batch Size | 32  para todas las pruebas |
| Número de Épocas | 100 épocas para cada modelo |
| Repeticiones | 5 repeticiones para cada tasa de aprendizaje (reducido debido al tiempo limitado para realizar los experimentos) |
| Función de Pérdida | Entropía cruzada (Cross-Entropy Loss) para la clasificación |
| Función de Activación | Softmax en la capa de salida, adecuada para la clasificación multiclase |
| Optimización | Optimizador Adam, conocido por su eficiencia y adaptación del learning rate durante el entrenamiento |
| Validación Cruzada | Enfoque de validación cruzada implementado para evaluar el desempeño del modelo y reducir el riesgo de sobreajuste |

7.7.2 Ajuste de paciencia Ciclo 3

En este experimento también se utilizó una configuración de "paciencia" (patience) que permite reducir la tasa de aprendizaje si la precisión de validación no mejora después de 7 épocas. Si la precisión de validación no mejora después de 10 épocas, el entrenamiento se detiene.

7.7.3 Análisis de Resultados

4.7.3.1 EfficientNet

Resumen General:  
 **Tabla 45.**

Resumen general Efficientnet ciclo 3

|  |  |  |  |
| --- | --- | --- | --- |
| **learning\_rate** | **repeticion** | **mejor\_val\_accuracy** | **mejor\_val\_loss** |
| 0.001 | 3 | 0.961 | 0.2188 |

**Varianza general**

**Tabla 46.**

Varianza general Efficientnet ciclo 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **learning\_rate** | **varianza\_val\_loss** | **varianza\_val\_accuracy** | **Promedio\_val\_accuracy** | **Promedio\_val\_loss** |
| 1e-3 | 2,8E+09 | 8.7E-05 | 0.95700 | 0.26891 |

**Resumen resultados 0.001:**

**Tabla 47.**

Tabla por repeticiones Efficientnet ciclo 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **repetición** | **max\_val\_accuracy** | **epoch\_max\_val\_accuracy** | **min\_val\_loss** | **epoch\_min\_val\_loss** |
| 1 | 0.94444 | 37 | 0.341699 | 34 |
| 5 | 0.94685 | 24 | 0.324624 | 24 |
| 4 | 0.96376 | 24 | 0.237844 | 56 |
| 3 | 0.96618 | 22 | 0.218762 | 23 |
| 2 | 0.963768 | 28 | 0.221623 | 41 |

Gráfica pérdida y accuracy en validación vs pérdida y accuracy en entrenamiento de la mejor repetición

****Figura 24. Gráfica pérdida y accuracy en validación vs pérdida y accuracy en entrenamiento de la mejor repetición Efficientnet ciclo 3

Discusión:

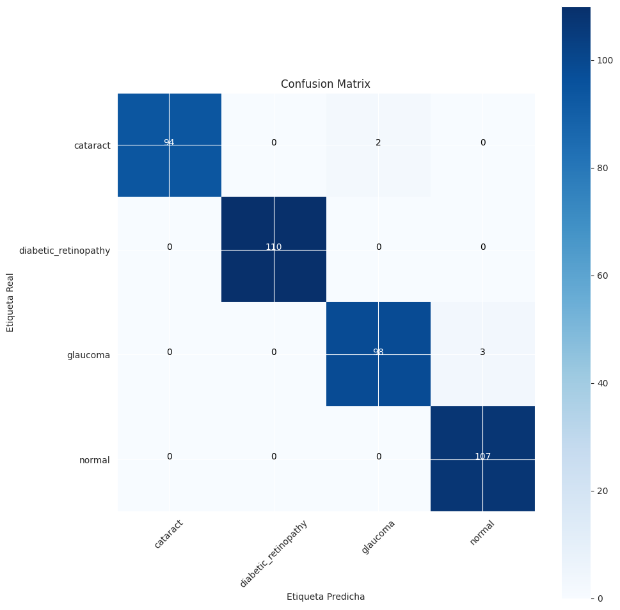
EfficientNet mostró una mejora con su rendimiento en el ciclo 3, logrando una precisión máxima de validación de 0.961 y una pérdida mínima de 0.2188 en la repetición 3, con una convergencia rápida. A lo largo de las repeticiones, se observó una ligera variación en la pérdida de validación con un valor de varianza de 0.0028, mientras que la varianza en la precisión de validación fue de 8.73E-05. La gráfica de pérdida y precisión mostró un comportamiento consistente, con mejoras continuas y pocas oscilaciones, alcanzando valores cercanos a 0.96 en precisión de validación y una disminución constante en la pérdida.

7.7.4 Conclusiones Ciclo 3 con respecto a las tablas de varianza, tablas generales y tablas por learning rate

Los resultados confirman que EfficientNet ajustó correctamente sus parámetros durante el entrenamiento, destacando su capacidad de ajuste para lograr un rendimiento óptimo en la clasificación ocular. Las tablas de varianza reflejan una notable estabilidad, mientras que las generales muestran una convergencia rápida. Además, se puede observar que la tasa de aprendizaje utilizada permitió que el modelo mantuviera un equilibrio entre rendimiento y estabilidad.

7.7.5 Conclusiones Ciclo 3 con respecto a las tablas de métricas y matrices de confusión

Figura. 25. Matriz de confusión de EfficientNet ciclo 3



**Tabla. 48.**

Tabla de métricas Efficientnet ciclo 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| Cataract | 1,00 | 0,98 | 0,99 | 96 |
| diabetic\_retinopathy | 1,00 | 1,00 | 1,00 | 110 |
| glaucoma | 0,98 | 0,97 | 0,98 | 101 |
| Normal | 0,97 | 1,00 | 0,99 | 107 |
| accuracy |  |  | 0,99 | 414 |
| Macro avg | 0,99 | 0,99 | 0,99 | 414 |
| weighted avg | 0,99 | 0,99 | 0,99 | 414 |

**7.7.5.1 Exactitud General (Accuracy)**

**Tabla 49.**

Exactitud general EfficientNet

|  |  |
| --- | --- |
| **Ciclo Efficientnet** | **Accuracy** |
| 2 | 0.95 |
| 3 | 0.99 |

El modelo EfficientNet logra una mejora significativa en su exactitud general, pasando del 95% en el ciclo 2 al 99% en el ciclo 3.

7.7.5.2 Precisión por Clase

**Tabla 50.**

Precisión por clase ciclo 2 vs ciclo 3 Efficientnet

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Ciclo Efficientnet** | **Cataratas** | **Retinopatía Diabética** | **Glaucoma** | **Normal** |
| 2 | 0.99 | 1.00 | 0.92 | 0.90 |
| 3 | 1.00 | 1.00 | 0.98 | 0.97 |

 Cataratas:

La precisión de EfficientNet mejoró del 0.99 en el ciclo 2 al 1.00 en el ciclo 3, alcanzando un rendimiento perfecto en esta clase. gracias a las actualizaciones que se realizó en las configuraciones de hiperparámetros

Retinopatía Diabética:

La precisión se mantuvo constante en 1.00 en ambos ciclos, demostrando una clasificación excelente en esta categoría.

Glaucoma:

El modelo aumentó su precisión del 0.92 en el ciclo 2 al 0.98 en el ciclo 3, mejorando su precisión en este ciclo gracias a los ajustes realizados en las configuraciones de hiperparámetros.

Normal:

 El modelo pasó del 0.90 en el ciclo 2 al 0.97 en el ciclo 3, aquí se evidencia una mejora en la capacidad para distinguir imágenes normales.

7.7.5.3 Recall por Clase

**Tabla 51.**

Recall por clase ciclo 2 vs ciclo 3 Efficientnet

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Ciclo Efficientnet** | **Cataratas** | **Retinopatía Diabética** | **Glaucoma** | **Normal** |
| 2 | 0.96 | 1.00 | 0.92 | 0.93 |
| 3 | 0.98 | 1.00 | 0.97 | 1.00 |

Cataratas:

El modelo EfficientNet mejoró su precisión del 0.99 en el ciclo 2 al 1.00 en el ciclo 3, alcanzando un rendimiento perfecto en esta clase.

Retinopatía Diabética:

El modelo mantuvo una precisión constante de 1.00 en ambos ciclos

Glaucoma:

El modelo aumentó su precisión del 0.92 en el ciclo 2 al 0.98 en el ciclo 3, una mejora significativa en la clasificación de esta clase.

Normal:

El modelo pasó del 0.90 en el ciclo 2 al 0.97 en el ciclo 3, mejorando su precisión en este ciclo y reflejando un ajuste más preciso y una mayor capacidad de generalización en la última fase, gracias a los ajustes realizados en las configuraciones de hiperparámetros

7.7.5.4 F1-Score por Clase

**Tabla 52.**

F1-Score por clase ciclo 2 vs ciclo 3 Efficientnet

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Ciclo Efficientnet** | **Cataratas** | **Retinopatía Diabética** | **Glaucoma** | **Normal** |
| 2 | 0.97 | 1.00 | 0.92 | 0.91 |
| 3 | 0.99 | 1.00 | 0.98 | 0.99 |

Cataratas:

El modelo EfficientNet incrementó su F1-Score de 0.97 en el ciclo 2 a 0.99 en el ciclo 3, acercándose a un rendimiento ideal en esta clase.

Retinopatía Diabética:

El modelo mantuvo un F1-Score perfecto de 1.00 en ambos ciclos

Glaucoma:

El modelo mejoró su F1-Score de 0.92 en el ciclo 2 a 0.98 en el ciclo 3, un avance importante en la clasificación de esta clase.

El modelo aumentó su F1-Score de 0.91 en el ciclo 2 a 0.99 en el ciclo 3, una mejora en la identificación de casos normales

7.7.5.5 Conclusiones Comparativas según matriz de confusión

Las matrices de confusión entre los ciclos 2 y 3 del modelo EfficientNet reflejan una mejora significativa en todas las clases evaluadas.

La exactitud general aumentó del 95% al 99%, aquí se evidencia un ajuste más preciso.

En términos de precisión por clase, recall y F1-Score, se observaron mejoras en casi todas las clases, especialmente en Glaucoma y Normal.

7.8 Desarrollo Fase 3

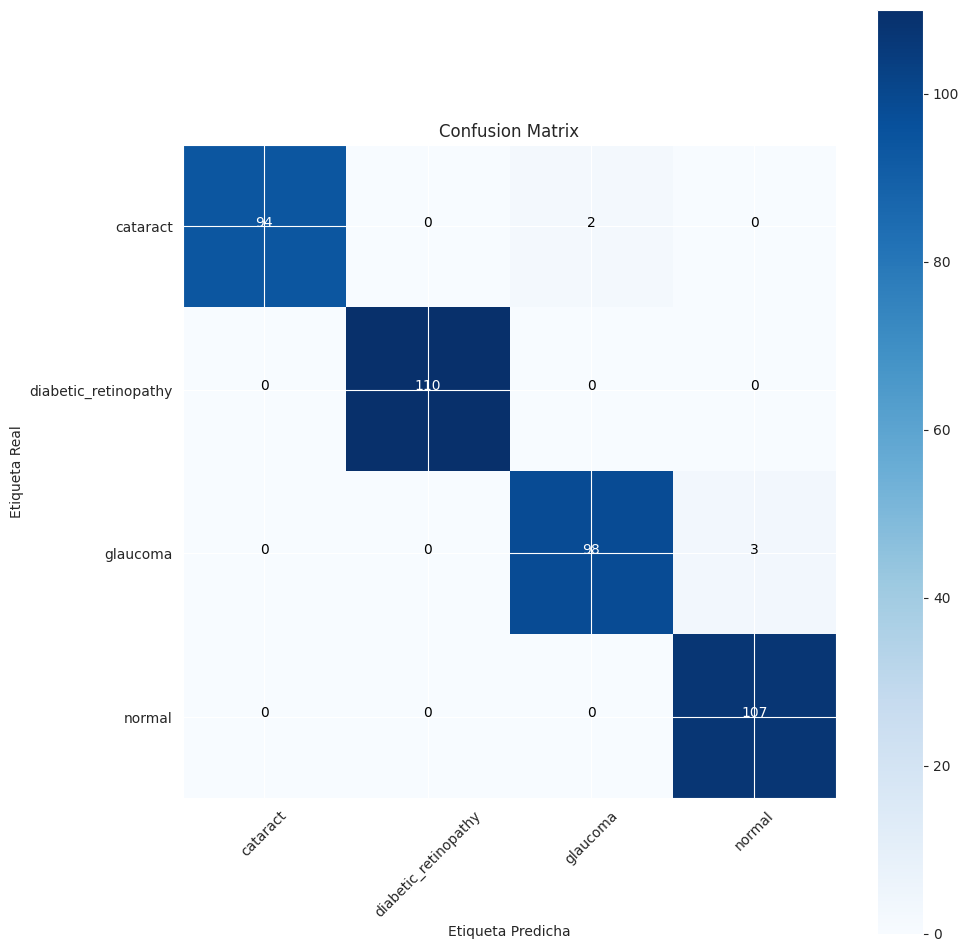
7.8.1 Desempeño del modelo CNN en la clasificación de imágenes de fondo de ojo

El modelo de aprendizaje profundo basado en la arquitectura EfficientNetB3 fue sometido a pruebas utilizando un conjunto de imágenes de fondo de ojo, clasificando las cuatro clases ya anteriormente mencionadas (Cataratas, Glaucoma, Retinopatía Diabética y Normal). Para cada una de las clases, se emplearon alrededor de 100 imágenes obtenidas de la base de datos escogida en el objetivo específico número uno, con el fin de evaluar el rendimiento del modelo en la predicción de cada categoría.

7.8.2 Matriz de confusión

En cuanto a los resultados obtenidos, el modelo de clasificación ha demostrado un desempeño altamente positivo, reflejado en la matriz de confusión generada a partir de las pruebas realizadas con aproximadamente 100 imágenes por clase. La matriz de confusión muestra que el modelo fue capaz de clasificar correctamente la gran mayoría de las imágenes, gracias a la eficiencia del modelo final. Para llevar a cabo esta evaluación, se diseñó un script que permitía probar la interfaz cargando automáticamente imágenes de las cuatro clases (Catarata, Retinopatía Diabética, Glaucoma y Normal) de forma secuencial.

Figura 26. Matriz de confusión prueba interfaz



Observando la matriz, se puede evidenciar que, de las 96 imágenes de la clase "Catarata", el modelo clasificó correctamente 94, aunque muestra solo 2 errores de clasificación hacia la clase "Glaucoma". De manera similar, en la clase "Retinopatía Diabética", el rendimiento fue impecable, ya que el modelo clasificó correctamente las 110 imágenes, sin cometer errores. Para la clase "Glaucoma", de las 101 imágenes, el modelo clasificó correctamente 98, pero se muestra un pequeño margen de error con 3 imágenes clasificadas como "Normal". Para la clase "Normal" el modelo tuvo el mejor desempeño, clasificando correctamente 107 de las 107 imágenes.

7.8.3 Desarrollo e integración de la interfaz de usuario

El desarrollo de la interfaz de usuario buscaba que el prototipo fuera accesible y fácil de usar para usuarios sin conocimientos técnicos o de programación. Tras evaluar varias alternativas y fracasar a la hora de tratar de implementarla con Ngrok y Colab, se decidió utilizar Jupyter Notebook en combinación con Voila, este último el notebook en una interfaz gráfica estilo aplicación web sin mostrar el código fuente.

Durante el proceso de diseño se implementaron características enfocadas en la usabilidad para el usuario. Se emplearon librerías como ipywidgets para permitir la interacción directa con el usuario a través de botones y áreas de carga de archivos. La interfaz guía al usuario de manera intuitiva con mensajes y permite cargar imágenes en formato .jpg o .jpeg, que luego son procesadas por el modelo para realizar la predicción.

7.8.4 Codificación de la interfaz

La interfaz fue construida bajo los siguientes pasos clave:

Carga de imágenes: Se emplea un widget de carga de archivos que permite al usuario seleccionar una imagen desde su dispositivo.

Preprocesamiento de imágenes: La imagen cargada es transformada para que sea compatible con el modelo CNN. Se redimensiona a 224x224 píxeles, y se convierte en un array numérico mediante numpy para que el modelo pueda interpretarlo y clasificarlo.

Clasificación: El modelo EfficientNetB3 realiza una predicción sobre la clase a la que pertenece la imagen.

Visualización de resultados: El sistema proporciona una visualización clara al usuario, ya que muestra la imagen cargada junto con el resultado de la predicción. Si la imagen cargada no es válida, el sistema responde con un mensaje de error e indicando al usuario que debe subir una imagen en el formato adecuado.

Figura 27. Interfaz de usuario final del prototipo

7.8.5 Pruebas de integración y rendimiento del sistema

Una vez implementada la interfaz y conectado el modelo se realizaron pruebas de integración para asegurar el correcto funcionamiento del sistema y su capacidad para interactuar con el modelo CNN en tiempo real. Estas últimas confirmaron que la interfaz es capaz de manejar imágenes en formato .jpg o .jpeg, redimensionarlas a 224x224 píxeles y generar predicciones precisas de forma dinámica.

Durante el proceso de prueba, y como anteriormente se ha mencionado, el sistema fue sometido a un conjunto de imágenes de 100 imágenes por clase aproximadamente, conformado por el dataset escogido en el primer objetivo específico. Los resultados fueron consistentes con los obtenidos previamente en las pruebas sin interfaz. Adicionalmente, se pudo evidenciar durante las pruebas que la retroalimentación que se ofrece es rápida y clara, proporcionando al usuario un flujo intuitivo desde la carga de la imagen hasta la visualización del resultado.

Se puede concluir entonces que, la interfaz desarrollada cumple con el objetivo específico número cuatro, ya que el artefacto diseñado es capaz de proporcionar una experiencia accesible para el usuario final, pues permite cargar y clasificar imágenes de fondo de ojo.

7.9 Interacción de los usuarios de prueba con la interfaz

Para evaluar la funcionalidad y usabilidad del prototipo de clasificación de imágenes de retinoscopía, se llevaron a cabo pruebas con dos profesionales de la salud: un oftalmólogo y una médica general, con el objetivo de identificar áreas de mejora tanto en la precisión del modelo como en la experiencia de usuario. Es importante resaltar que ambos profesionales han pedido permanecer anónimos, por ello sólo se mostrarán sus opiniones, resultados de las pruebas y una breve descripción de ellos sin revelar sus nombres.

7.9.1 Descripción de los usuarios de prueba

7.9.1.1 Doctor 1, Oftalmólogo:

Edad: 43 años

Experiencia: 8 años como Médico Oftalmólogo

Especialización: Supraespecialista en Segmento Anterior, Córnea y Cirugía Refractiva

Posición actual: Profesor de Posgrado en la Especialización Médica y Quirúrgica en Oftalmología en una universidad de Bucaramanga, Oftalmólogo en un centro oftalmológico.

Estudios:

* Médico y Cirujano – (2008)
* Especialista en Oftalmología – (2016)
* Supraespecialista en Segmento Anterior, Cirugía Refractiva y Córnea

Investigaciones:

Ha realizado investigaciones sobre el Poster Lente Intraocular Multifocal Posterior a Cirugía Refractiva

Relación con el prototipo:

El doctor considera que el prototipo de clasificación podría ser útil, pero no lo aplicaría en su práctica profesional, porque confía más en su experiencia clínica, exámenes directos y juicio al diagnosticar a sus pacientes.

7.9.2.2. Doctora 2: Médica general y de urgencias

Edad: 29 años

Experiencia: Médica General con experiencia en urgencias neonatales, pediátricas y tratamiento de enfermedades prevalentes no transmisibles

Posición actual: Médica General en un servicio de urgencias

Especialización: No tiene especialización en oftalmología, pero su formación incluye el tratamiento de pacientes en áreas clínicas generales.

Estudios:

Médica General – (2020)

Relación con el prototipo:

La doctora se muestra interesada en el prototipo, pero lo considera poco útil en su práctica diaria, ya que no se encuentra con frecuencia con casos que requieran la interpretación de imágenes de retinoscopía. Sin embargo, reconoce que hay potencial educativo y diagnóstico en ambientes clínicos específicos, y dice que podría ser útil para otros médicos que se enfrenten a casos de salud ocular.

7.9.2 Metodología de las pruebas

Cada profesional realizó pruebas utilizando imágenes de retinoscopías (de buena y baja calidad), se les advirtió a cada uno usar solamente imágenes de retinoscopía.

Pruebas con imágenes de retinoscopía de alta calidad: Se usaron imágenes obtenidas del sistema de archivos del oftalmólogo, así como imágenes del dataset utilizado en el desarrollo del prototipo. Se evaluó la precisión del modelo en la clasificación de las cuatro clases predichas: Catarata, Retinopatía Diabética, Glaucoma y Normal.

Pruebas con imágenes desenfocadas (baja calidad): Se modificaron imágenes obtenidas del sistema de archivos del doctor, usando una herramienta llamada PineTools, en dónde se le aplicó un desenfoque de pila con radio 55, con el objetivo de evaluar cómo actúa el prototipo ante imágenes con ruido.

7.9.3 Resultados

**Tabla 53.**

Resultados del Oftalmólogo

|  |  |  |
| --- | --- | --- |
| **Tipo de prueba** | **Descripción** | **Resultados** |
| Imágenes de retinoscopía (calidad estándar) | 10 imágenes de retinoscopía por clase, obtenidas de una carpeta, en el disco duro de su computador | El prototipo clasificó correctamente todas las imágenes en sus respectivas clases (Catarata, Retinopatía Diabética, Glaucoma y Normal). |
| Imágenes desenfocadas (calidad baja) | Imágenes con un desenfoque simulado de radio 55, aplicadas a las imágenes originales para degradar la calidad. | Catarata: 9/10 correctas, Glaucoma: 5/10 correctas, Normal: 6/10 correctas, Retinopatía Diabética: 7/10 correctas. Las imágenes desenfocadas causaron un descenso en la precisión, especialmente en las clases Glaucoma y Normal. |

**Tabla 54.**

Sugerencias del Oftalmólogo

|  |  |
| --- | --- |
| **Aspecto evaluado** | **Sugerencia** |
| Calidad de las imágenes | Mejorar la precisión en imágenes de baja calidad |
| Consentimiento informado | Falta el diseño del consentimiento informado. Incluir un documento explicativo para los usuarios, en dónde se especifique que las imágenes y datos tratados solo se usarán para fines investigativos y educativos. |
| Clasificación múltiple | Añadir la capacidad de clasificar múltiples imágenes al mismo tiempo. |
| Justificación de la clasificación | Explicar en detalle la justificación de las clasificaciones generadas por el modelo, para así proporcionar al usuario un mayor contexto sobre el porqué de la decisión. Adicionalmente, mostrar el nivel de gravedad del diagnóstico (Ejemplo: Catarata Grave) y tipo de enfermedad (Retinopatía Diabética Proliferativa Grave, Retinopatía Diabética No Proliferativa leve) |

**Tabla 55.**

Resultados de la médica general

|  |  |  |
| --- | --- | --- |
| **Tipo de prueba** | **Descripción** | **Resultados** |
| Imágenes de retinoscopía (calidad estándar) | 10 imágenes por clase del dataset usado en el desarrollo del prototipo. | El prototipo clasificó correctamente todas las imágenes |
| Imágenes desenfocadas (calidad baja) | Imágenes desenfocadas de la misma manera que en las pruebas del oftalmólogo en cuestión. | Catarata: 7/10 correctas, Glaucoma: 6/10 correctas, Normal: 5/10 correctas, Retinopatía Diabética: 6/10 correctas. La precisión disminuyó al igual que en las pruebas previas. |

**Tabla 56.**

Sugerencias de la médica general

|  |  |
| --- | --- |
| **Aspecto evaluado** | **Sugerencia** |
| Calidad de las imágenes | Mejorar la precisión en imágenes de baja calidad |
| Facilidad de instalación | Simplificar la instalación del prototipo. En lugar de implementarlo en formato notebook, convertirlo en una aplicación de escritorio. |
| Análisis detallado del ojo | Incluir características que permitan un análisis más específico de las diferentes partes del ojo (mácula, fóvea, etc.). |
| Sección educativa | Añadir una sección educativa en la interfaz que explique de manera breve las enfermedades oculares (catarata, glaucoma, retinopatía diabética) y cómo interpretar los resultados en una consulta general. |

Figura 28. Predicción correcta de imagen borrosa de catarata en las pruebas con el oftalmólogo

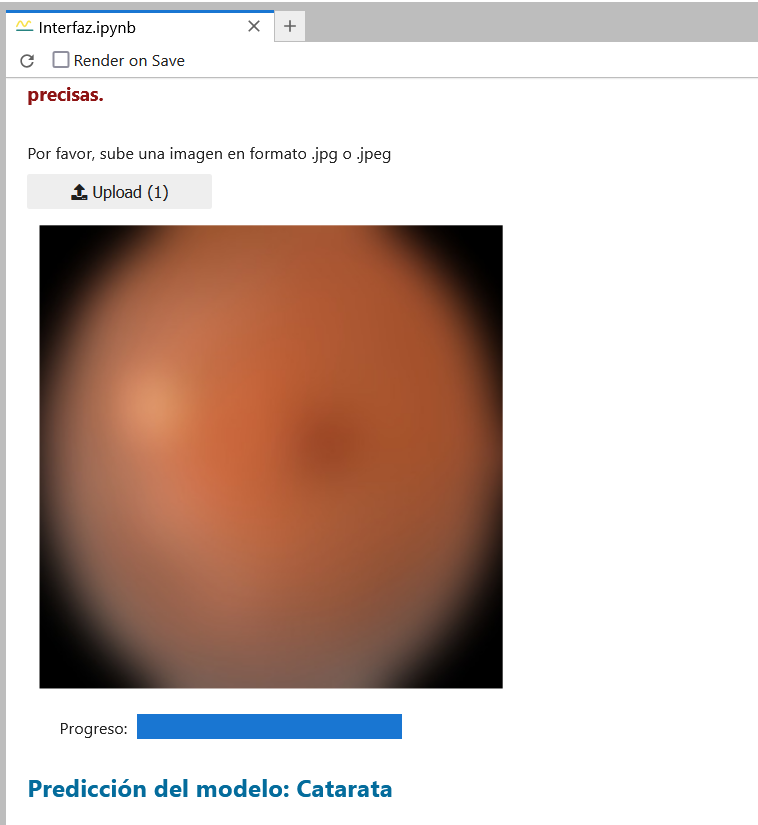


Figura 29. Predicción incorrecta de imagen borrosa de glaucoma en las pruebas con la médica general

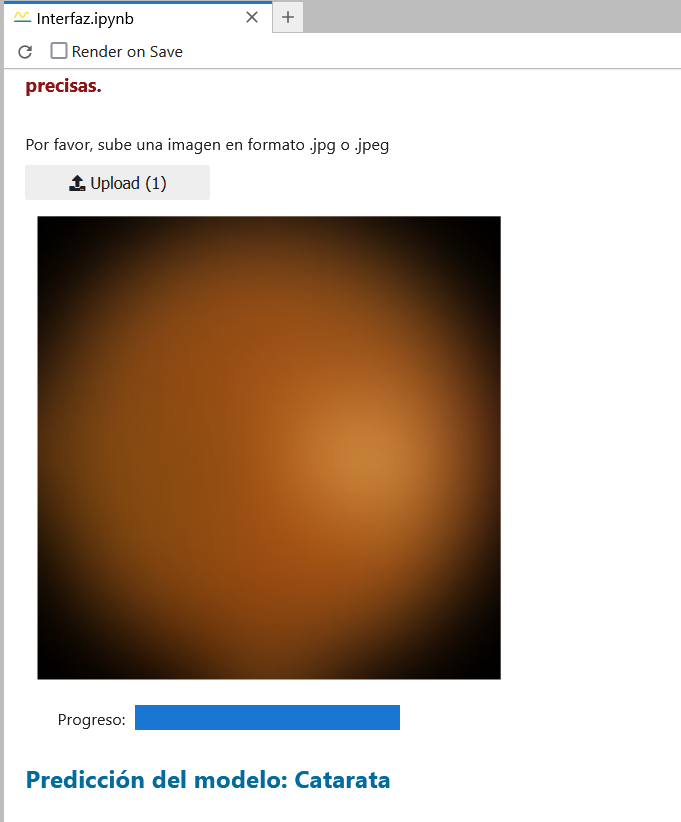


Figura 30. Predicción incorrecta de imagen borrosa de glaucoma en las pruebas con el oftalmólogo

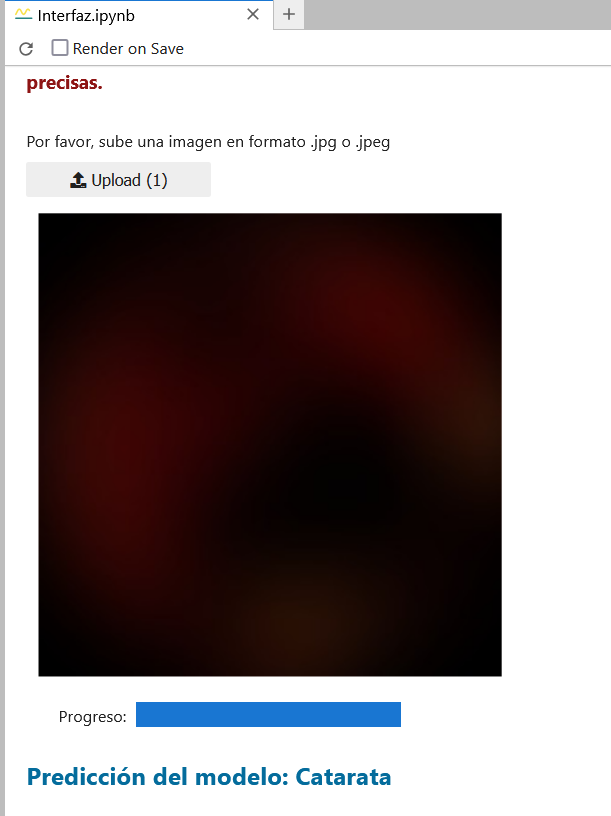


Figura 31. Predicción correcta de imagen borrosa de retinopatía diabética en las pruebas con la médica general

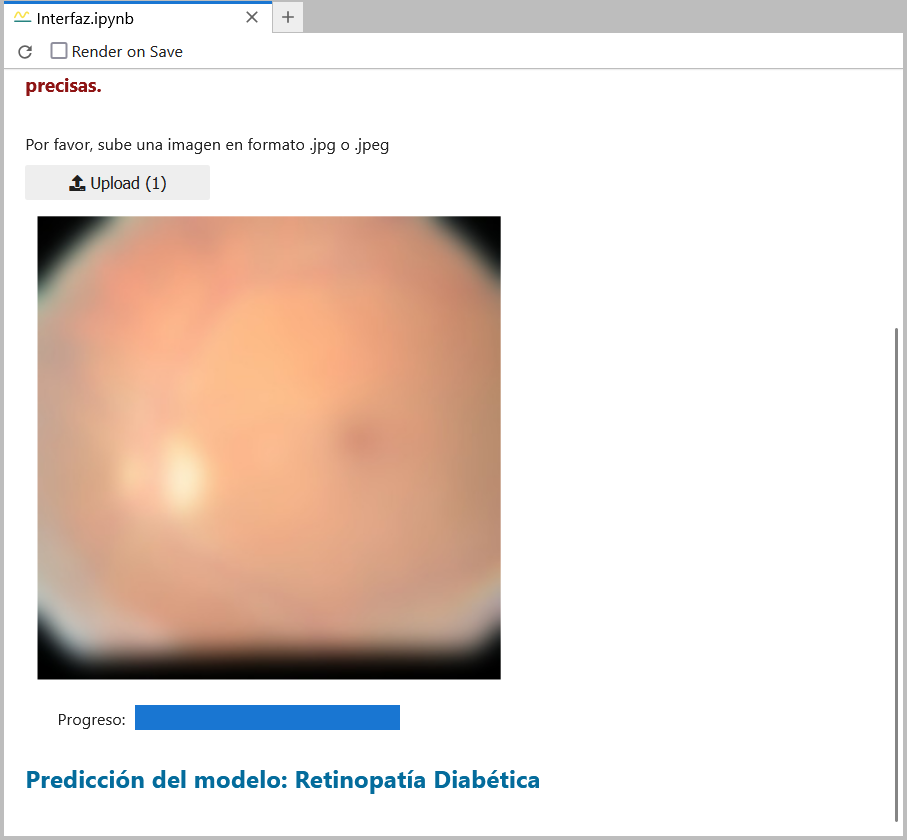


Figura 32. Predicción correcta de imagen de catarata en las pruebas con el oftamólogo

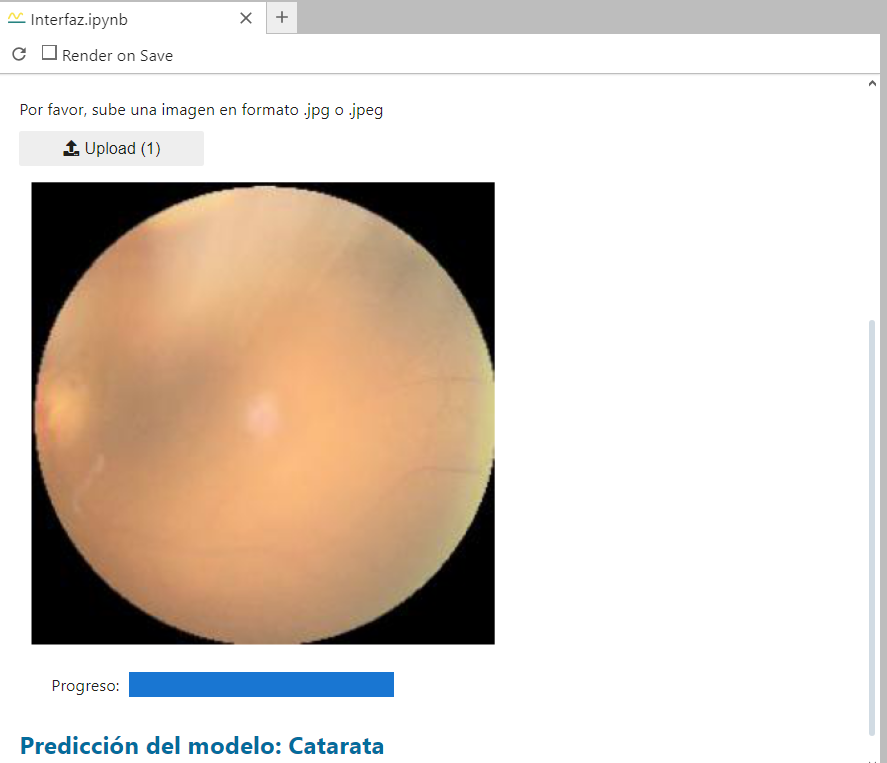


Figura 33. Predicción correcta de imagen de retinopatía diabética en las pruebas con la médica general

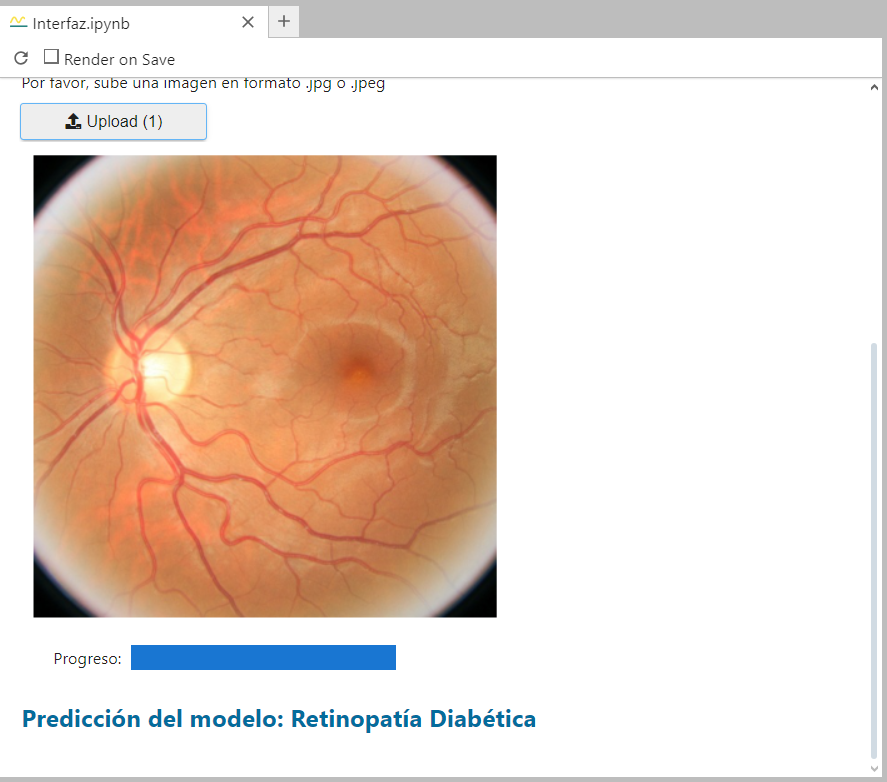


Figura 34. Predicción correcta de imagen de ojo normal en las pruebas con el oftalmólogo

Interfaz de usuario gráfica

Descripción generada automáticamente

Figura 35. Predicción correcta de imagen de ojo normal en las pruebas con la médica general

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Figura 36. Matriz de confusión usando imágenes de retinoscopía, evidenciada en las pruebas con ambos doctores

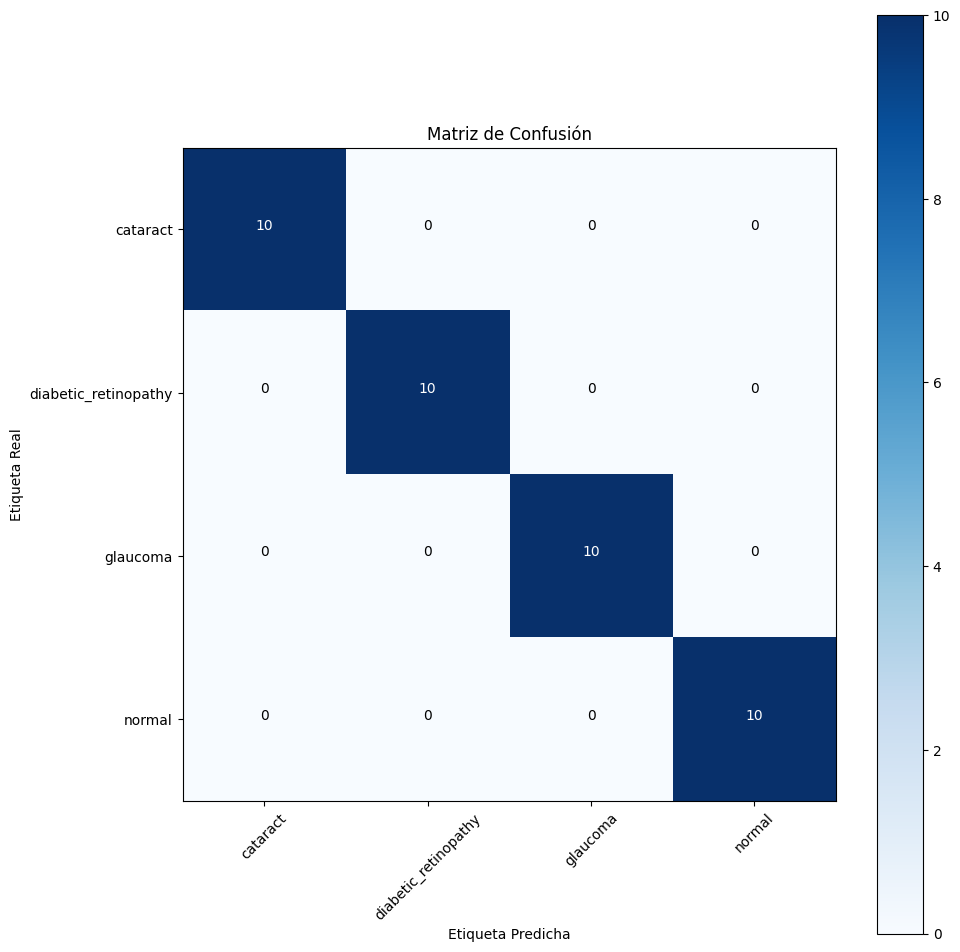


Figura 37. Matriz de confusión usando imágenes de retinoscopía borrosas, evidenciada en las pruebas con la médica general

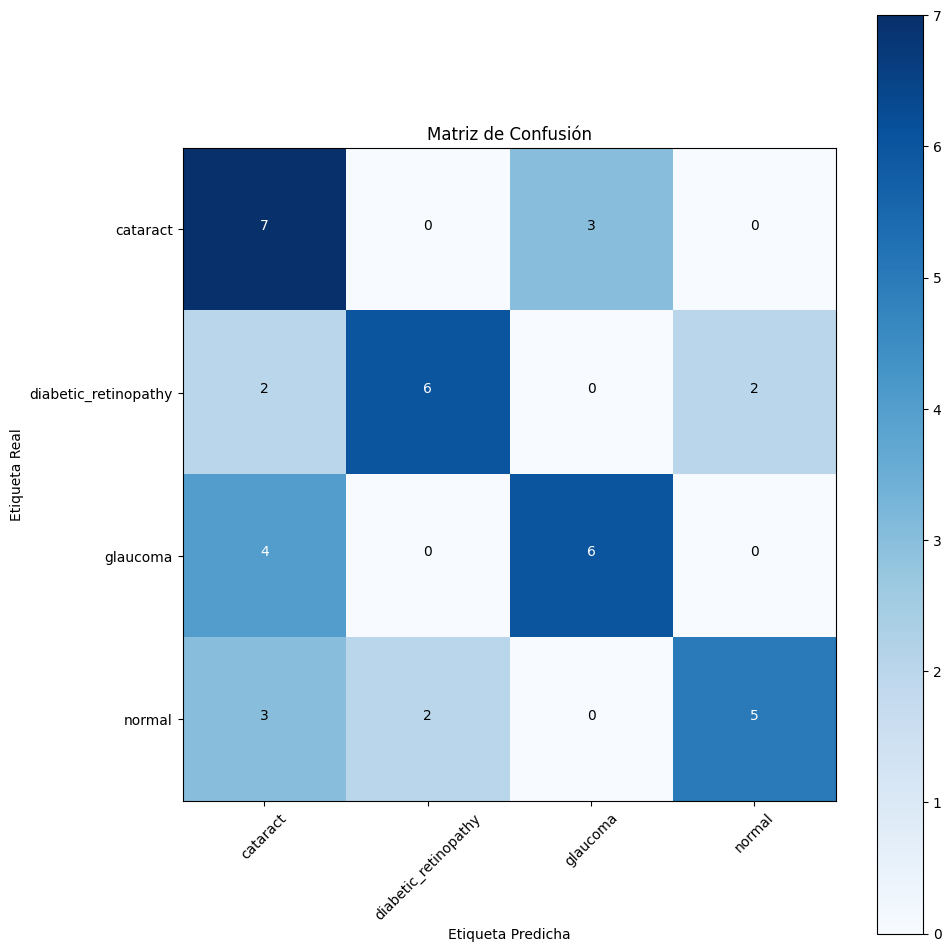
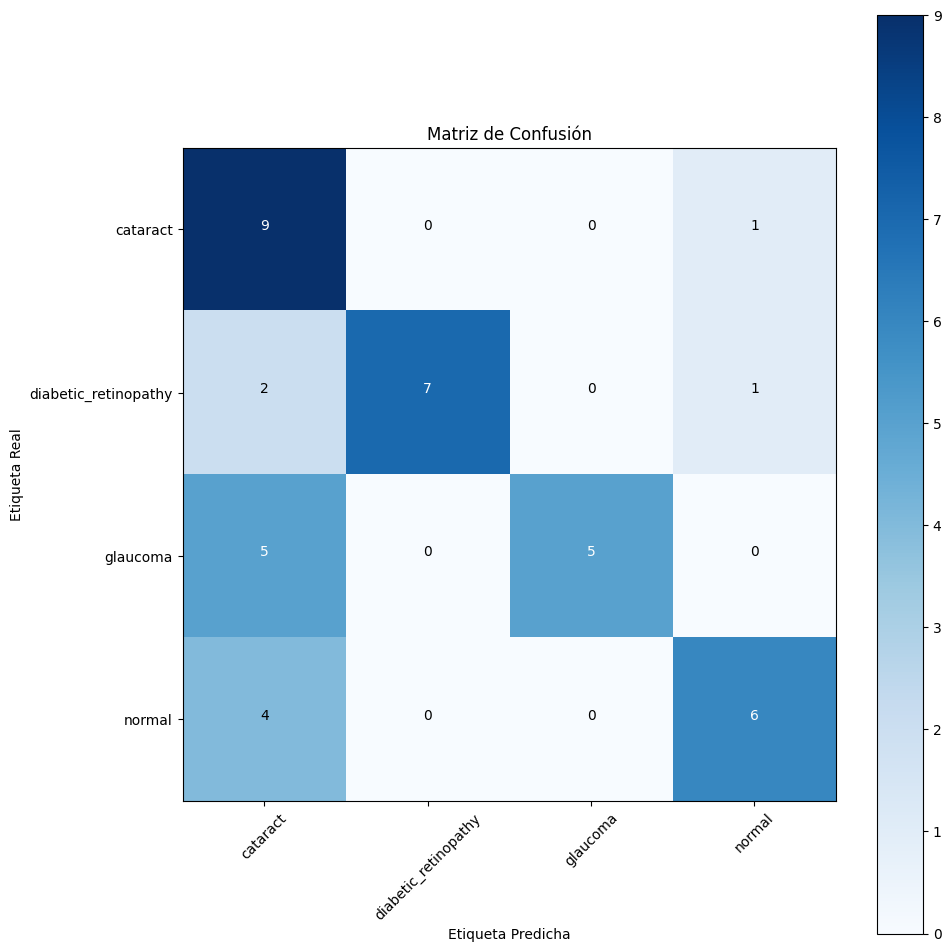


Figura 38. Matriz de confusión usando imágenes de retinoscopía borrosas, evidenciada en las pruebas con el oftalmólogo



4.9.4 Errores identificados durante las pruebas

En las pruebas se identificaron ciertos escenarios en los que la interfaz podría generar errores o comportamientos no esperados. La siguiente tabla resume estos errores, las condiciones previas que los provocan y las condiciones posteriores:

**Tabla 56.**

Tabla de errores encontrados en las pruebas

|  |  |  |
| --- | --- | --- |
| **Error** | **Precondición** | **Poscondición** |
| El sistema no identifica imágenes de baja calidad (borrosas). | El usuario carga una imagen de baja calidad o desenfocada. | El sistema la clasifica sin advertir al usuario sobre la calidad de la imagen. |
| El sistema no reconoce imágenes de formato incorrecto. | El usuario carga una imagen en un formato no compatible (.png, .gif). | El sistema arroja un error claro y no ofrece ninguna predicción. |
| Tiempo de respuesta largo en la predicción. | Se cargan imágenes de alta resolución o de gran tamaño. | El sistema tarda en procesar la imagen. Al final ofrece una predicción |

7.9.5 Conclusiones generales de las pruebas

Las pruebas realizadas con ambos profesionales de la salud arrojaron valiosa información para la mejora del prototipo:

El modelo tiene un buen rendimiento con imágenes de alta calidad, pero su precisión disminuye notablemente con imágenes desenfocadas o de baja calidad; esto indica la necesidad de entrenar a futuro el modelo con imágenes de baja calidad para poder predecir con más precisión dichas retinoscopías.

Durante las pruebas, se les advirtió a los profesionales de la salud no usar imágenes no relacionadas con retinoscopías, dado que el modelo de deep learning utilizado no tiene una capacidad explícita para manejar "clases desconocidas". Este comportamiento ha sido estudiado en trabajos como el de Hendrycks y Gimpel (2017), que explican cómo los modelos tienden a clasificar ejemplos fuera de su distribución (Out-Of-Distribution) en las clases más probables basadas en las características que reconocen, en lugar de etiquetarlos como desconocidos.

Se sugirió simplificar el proceso de instalación para que el sistema sea más accesible para cualquier profesional de salud que no esté familiarizado con plataformas como Jupyter; lo cual podría lograrse mediante una versión en formato de aplicación de escritorio.

Mientras que el oftalmólogo no considera que su uso pueda aplicarse en una consulta real, el prototipo podría ser útil para contextos de diagnóstico preliminar o educativo.

7.10 Pruebas con diferentes datasets

Para evaluar el desempeño del prototipo con imágenes distintas a las que se usaron para entrenar el modelo, se llevaron a cabo pruebas con dos conjuntos de datos distintos que contienen imágenes de fondo de ojo de pacientes con diversas condiciones oftalmológicas. A continuación, se detallan los resultados obtenidos:

Dataset 1: Base de datos pública de imágenes de fondo de ojo

Esta base de datos ha sido facilitada por el Laboratorio de Reconocimiento de Patrones (CS5), el Departamento de Oftalmología de la Universidad Friedrich-Alexander de Erlangen-Nuremberg (Alemania), y la Universidad Tecnológica de Brno, Facultad de Ingeniería Eléctrica y Comunicación, Departamento de Ingeniería Biomédica, Brno (República Checa). Contiene un total de 72 imágenes de fondo de ojo, divididas en cuatro categorías (Erlangen-Nürnberg, n.d.). Este dataset se encuentra en el **Apéndice E** con todas sus imágenes.

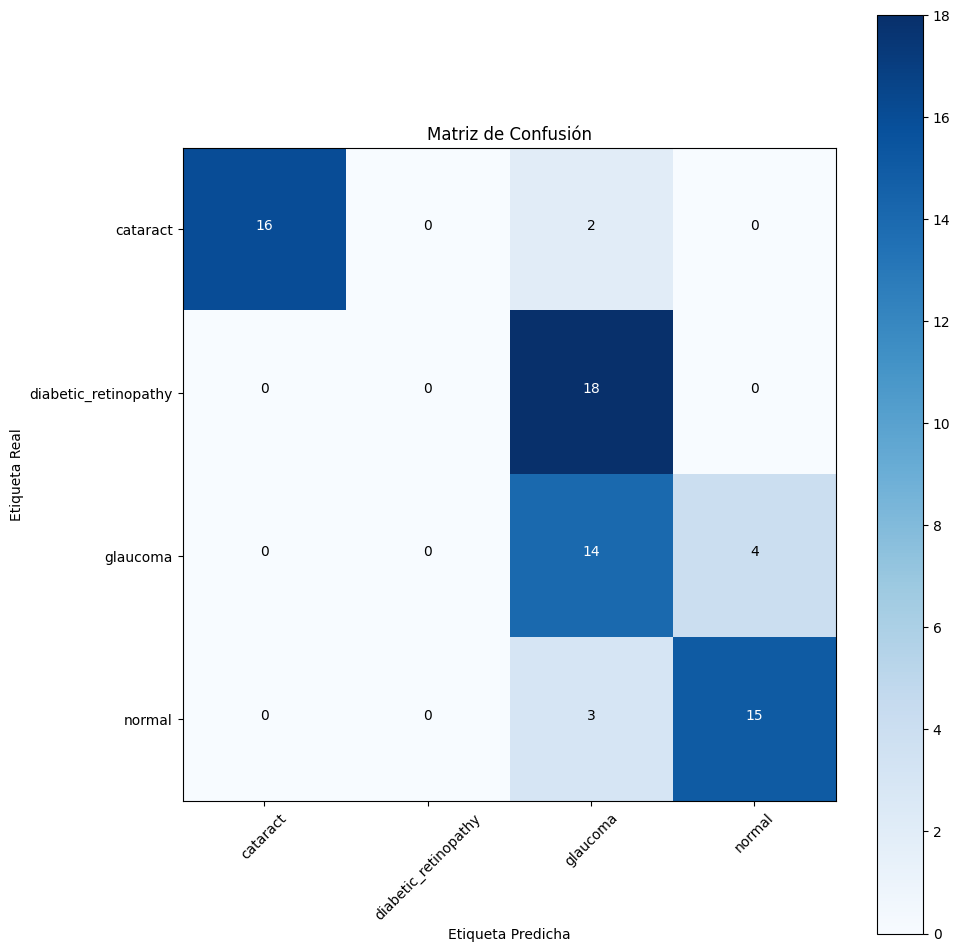
* 18 imágenes de pacientes sanos.
* 18 imágenes de pacientes con glaucoma.
* 18 imágenes de pacientes con retinopatía diabética.
* 18 imágenes de pacientes con catarata.

Cada imagen viene acompañada de una segmentación de los vasos sanguíneos y máscaras que determinan el campo de visión (FOV).

Resultados de clasificación:

* **Normales**: 15/18 imágenes fueron clasificadas correctamente como "Normal", mientras que las 3 restantes se clasificaron incorrectamente como "Glaucoma". **Precisión del 83%**
* **Glaucoma:** 14/18 imágenes fueron clasificadas correctamente como "Glaucoma", con las 4 imágenes restantes clasificadas erróneamente como "Normal". **Precisión del 78%**
* **Retinopatía Diabética:** Ninguna de las 18 imágenes de retinopatía diabética fue clasificada correctamente; todas fueron clasificadas como "Glaucoma".
* **Catarata**: 16/18 imágenes fueron clasificadas correctamente como "Catarata", con las dos imágenes restantes clasificadas como "Glaucoma". **Precisión del 90%**

Figura 39. Matriz de confusión base de datos pública



Este conjunto de resultados refleja la capacidad del modelo para reconocer bien las clases "Normal", "Catarata" y "Glaucoma", pero una nula precisión al tratar con la clase "Retinopatía Diabética"; se decide hacer una segunda prueba con un dataset que trate exclusivamente sobre esta última clase.

Dataset 2: Conjunto de imágenes de fondo de ojo para el estudio de la retinopatía diabética

Este conjunto de datos, creado por un equipo de investigadores, contiene 396 imágenes de fondo de ojo obtenidas en el Departamento de Oftalmología del Hospital de Clínicas, Facultad de Ciencias Médicas, Universidad Nacional de Asunción, Paraguay (Benítez et al., 2021). Este dataset se encuentra en el **Apéndice E** con todas sus imágenes. Las imágenes fueron adquiridas utilizando la cámara Visucam 500 de la marca Zeiss, y los oftalmólogos clasificaron las imágenes en siete categorías:

* Sin signos de retinopatía diabética (RD): 20 imágenes.
* Retinopatía diabética No Proliferativa leve: 4 imágenes.
* Retinopatía diabética No Proliferativa moderada: 58 imágenes.
* Retinopatía diabética No Proliferativa grave: 92 imágenes.
* Retinopatía diabética No Proliferativa muy grave: 83 imágenes.
* RD Proliferativa: 70 imágenes.
* RD Proliferativa Avanzada: 69 imágenes.

Resultados de clasificación:

* **Retinopatía diabética No Proliferativa leve**: 4/4 imágenes clasificadas correctamente como "Retinopatía Diabética". **Precisión del 100%**
* **Retinopatía diabética No Proliferativa Moderada:** 50/58 imágenes clasificadas correctamente, con 4 clasificadas como "Normal", 1 como "Catarata" y 3 como "Glaucoma". **Precisión del 87%**
* **Retinopatía diabética No Proliferativa grave:** 88/92 imágenes clasificadas correctamente, 2 fueron clasificadas como "Normal" y 2 como "Catarata". **Precisión del 96%**
* **Retinopatía diabética No Proliferativa muy grave:** 75/83 imágenes clasificadas correctamente, 3 como "Normal", 2 como "Catarata" y 5 como "Glaucoma". **Precisión del 90%**
* **RD Proliferativa:** 67/70 imágenes clasificadas correctamente como "Retinopatía Diabética", 2 como "Glaucoma" y 1 como "Normal". **Precisión del 96%**
* **RD Proliferativa avanzada**: 60/69 imágenes clasificadas correctamente, con 6 clasificadas como "Glaucoma" y 3 como "Catarata". **Precisión 87%**

Como se puede observar, se obtuvo un buen desempeño en las categorías "Retinopatía diabética No Proliferativa leve", "Retinopatía diabética No Proliferativa grave", "RD Proliferativa" y "RD Proliferativa Avanzada". Sin embargo, se observaron algunas confusiones en las categorías de "Retinopatía diabética No Proliferativa moderada" y "Retinopatía diabética No Proliferativa muy grave", donde algunas imágenes fueron clasificadas erróneamente como "Normal", "Catarata" o "Glaucoma". Esto sugiere la necesidad de ajustes en el modelo, o entrenarlo a futuro con más variedad de datasets, para mejorar su sensibilidad y precisión en todas las categorías de la enfermedad (Hendrycks, 2016)

8. Cumplimiento de Objetivos Específicos

En esta sección se expone un análisis conciso, sobre cómo se ha logrado cumplir con los objetivos específicos del proyecto de grado, dando así su aclaración y visualización más sencilla. Además, se destacan las secciones de sus desarrollos dentro del documento.

8.1 Cumplimiento de Objetivo Especifico 1

A continuación, se muestra el objetivo específico 1 del cual se dará explicación en esta sección:

Seleccionar un conjunto de datos (dataset) representativos de imágenes de fondo de ojo que contenga las características morfológicas de enfermedades (retinopatía diabética, cataratas y glaucoma), así como de tejido sano, generando un conjunto de datos útil.

Para dar cumplimiento se hacen referencia en la sección 1.4.2 Caracterización de Imágenes, en el estado del arte: Guna Venkat Doddi, Clasificación de las enfermedades oculares (Eyes Disease Classification), referentes metodológicos y anexos; adicionalmente, se puede evidenciar su cumplimiento también en la sección 6.1.1 Fase 1: Recopilación de datos; en dónde se describen los tres criterios más importantes para escoger un dataset para la clasificación de enfermedades oculares, luego se categorizan y describen diez dataset candidatos y se justifica la elección del dataset de Gunna Venkat.

8.2 Cumplimiento de Objetivo Especifico 2

A continuación, se muestra el objetivo específico 2 del cual se dará explicación en esta sección:

Revisar el estado del arte buscando modelos de Aprendizaje profundo probados en imágenes de tejidos similares a fondos de ojos, tal que puedan detectar, procesar y clasificar imágenes de retinoscopía.

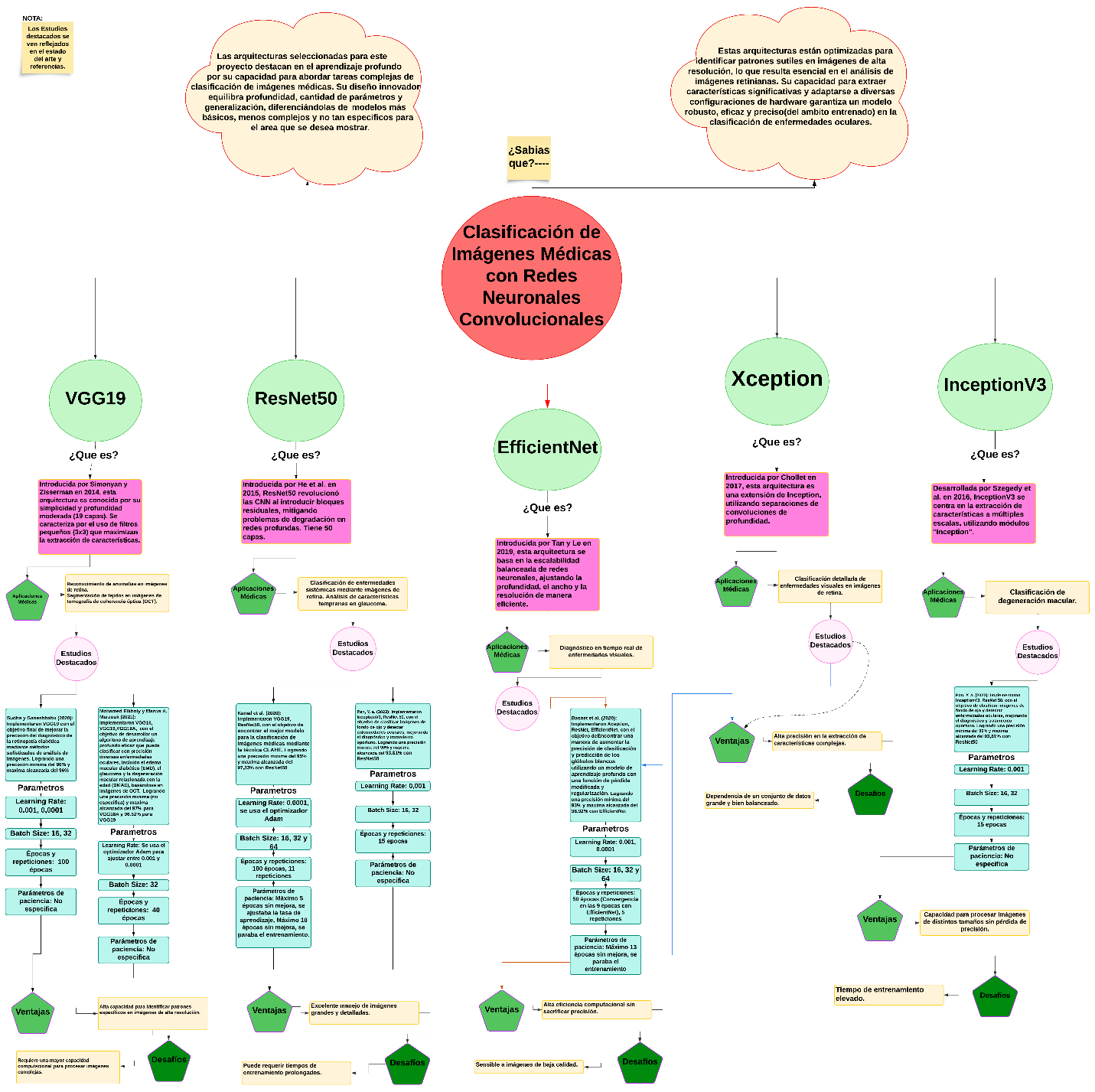
Se realiza la búsqueda de diferentes modelos de aprendizaje profundo enfocados en los procesos requeridos para trabajar con imágenes de tejidos similares a fondos de ojos. Este análisis permitió identificar los cinco modelos más prometedores, dando lugar a la muestra de los parámetros mas importantes en cada investigación y encontrando las similitudes de estos, a partir de esta selección, se elaboró un diagrama que resume las similitudes, características y parámetros esenciales de los modelos escogidos a partir del estado del arte.

Es importante destacar que las arquitecturas no seleccionadas no fueron incluidas en el estado del arte debido a su menor eficacia o relevancia para los objetivos del proyecto. Las arquitecturas elegidas se distinguen por su capacidad de abordar tareas complejas de clasificación de imágenes médicas, optimizando la identificación de patrones sutiles en imágenes de alta resolución. Su diseño equilibra profundidad, cantidad de parámetros y capacidad de generalización, superando a modelos más básicos o menos especializados.

Estas arquitecturas también destacan por su adaptabilidad a diferentes configuraciones de hardware, garantizando un modelo robusto (para el tiempo que se llevó a cabo) y eficaz para la clasificación precisa de enfermedades oculares.

El diagrama que ilustra estas comparaciones se presenta a continuación y, para mayor detalle, se encuentra disponible en el **Apéndice I** del documento.

Figura 40. Diagrama del Estado del Arte



8.3 Cumplimiento de Objetivo Especifico 3

A continuación, se muestra el objetivo específico 3 del cual se dará explicación en esta sección:

Escoger un modelo de aprendizaje profundo para la clasificación de patologías oculares que causan ceguera irreversible a través de pruebas exhaustivas, usando el conjunto de imágenes seleccionado.

El objetivo 3 se cumple en la sección 6.1.2 Fase 2: Selección de Modelos CNN, en dónde se resume el proceso de experimentación en los colabs y resultados de los 3 ciclos; cabe destacar que dicho resumen está acompañado de dos diagramas de flujo, de los cuales uno explica el proceso de generación de dataframes para el entrenamiento, y el otro explica el proceso de entrenamiento y prueba del modelo; en esta sección se menciona sus respectivos apéndices, entre estos el video explicativo desde el código de los colabs de entrenamiento, y el manual de uso de los colabs de entrenamiento y reentrenamiento.

También en la sección 7.2 Desarrollo Fase 2 se especifica más dicha selección del modelo final, el cual resultó ser EfficientNet, en ella se evidencia según ciclos:

**Ciclo 1**:

En este ciclo se evaluaron cinco arquitecturas (VGG19, ResNet50, Xception, EfficientNetB3 e InceptionV3) clasificando imágenes de fondo de ojo en dos categorías: **Sano** y **Enfermo**. Se realizaron 10 repeticiones con tres tasas de aprendizaje (0.01, 0.001 y 0.0001), utilizando un tamaño de lote de 16 y 40 épocas por repetición.

* **VGG19**: Mostró una precisión promedio del **94%** con su mejor tasa de aprendizaje (0.0001). Aunque tuvo variabilidad en tasas más altas.
* **ResNet50**: Alcanzó un promedio de precisión del **95.6%**.
* **EfficientNetB3**: Fue el mejor modelo con una precisión promedio de **96.9%**.
* **Xception**: Mostró resultados variables, una precisión promedio inferior al **80%**. Se descarta para ciclo 2
* **InceptionV3**: Tuvo el peor desempeño con una precisión promedio del **74%** y la pérdida de validación más alta. Se descarta para ciclo 2.

**Ciclo 2**

En esta etapa se evaluaron las arquitecturas seleccionadas (VGG19, ResNet50 y EfficientNetB3) clasificando imágenes en cuatro categorías: **Normal, Catarata, Glaucoma y Retinopatía Diabética**. Se usaron 60 épocas, un tamaño de lote de 16 y tasas de aprendizaje determinadas en el ciclo anterior.

* **VGG19**: Logró una precisión de **93.7%** con una pérdida de validación de 0.3347. Aunque tuvo un rendimiento bueno, fue menos consistente que los otros modelos.
* **ResNet50**: Alcanzó una precisión de **95.2%** y destacó por su baja varianza en precisión y pérdida, aunque no superó a EfficientNetB3 en términos de rendimiento global.
* **EfficientNetB3**: Lideró con una precisión general de **95%**, el balance entre rendimiento y eficiencia que este modelo presenta lo posicionó como el modelo más eficaz para integrarlo en la interfaz.

**Ciclo 3**

En esta etapa se entrena y optimiza el modelo final para su posterior implementación en la interfaz, en esta EfficientNetB3 fue evaluado con un tamaño de lote de 32, una tasa de aprendizaje de 0.001 y 100 épocas por repetición, manteniendo las cuatro categorías de clasificación. Se observa lo siguiente:

* Obtuvo una precisión de **96.1%** y una pérdida de **0.2188**, esto significa que mejoró en su precisión y velocidad de convergencia en comparación con EfficientNet de los ciclos anteriores.
* El modelo incrementó su exactitud del **95%** en el ciclo 2 al **99%**, demostrando mayor capacidad de generalización.

8.4 Cumplimiento de Objetivo Especifico 4

A continuación, se muestra el objetivo específico 4 del cual se dará explicación en esta sección:

Integrar el modelo de Aprendizaje profundo seleccionado mediante una interfaz tipo Python notebook.

El objetivo 4 se cumple en la sección 6.1.3 Fase 3: Integración del Modelo CNN, en dónde se describe brevemente el proceso de diseño y codificación de la interfaz, usando un diagrama de secuencia para poder describir el proceso de clasificación de imágenes de retinoscopía, así como cuál es la relación entre la interfaz y el modelo, también se usa un diagrama de flujo para describir las funcionalidades y procesos que hace la interfaz. Se referencian apéndices de los videos explicativos que acompañan dichos diagramas, así como el de Manual de Usuario.

Se evidencia más a detalle su cumplimiento en la sección 7.9 Desarrollo Fase 3, en dónde se realizan pruebas con dos diferentes profesionales de la medicina, las cuales se hicieron con imágenes de retinoscopía de buena y mala calidad; el resultado fue una clasificación del 100% para 10 imágenes de fondo de ojo de cada clase, en cada prueba con los profesionales. Adicionalmente, se realizaron pruebas con dos diferentes dataset; el primero contenía 18 imágenes de cada clase, en el cual se evidenció una precisión de 83% en Normal, 78% en Glaucoma y 90% en Catarata, pero ninguna imagen de retinopatía diabética se clasificó correctamente, por lo que se decidió realizar una evaluación con un segundo dataset enfocado específicamente en esta clase.

En los resultados del segundo dataset se puede evidenciar que, si bien la clase de Retinopatía Diabética no tuvo un buen rendimiento con el primer dataset de prueba, en el segundo su puntaje de precisión fue bastante alto para las distintas clases específicas de retinopatía diabética, como se muestra a continuación:

* **Retinopatía diabética No Proliferativa leve**: 4/4 imágenes clasificadas correctamente como "Retinopatía Diabética". **Precisión del 100%**
* **Retinopatía diabética No Proliferativa Moderada:** 50/58 imágenes clasificadas correctamente, con 4 clasificadas como "Normal", 1 como "Catarata" y 3 como "Glaucoma". **Precisión del 87%**
* **Retinopatía diabética No Proliferativa grave:** 88/92 imágenes clasificadas correctamente, 2 fueron clasificadas como "Normal" y 2 como "Catarata". **Precisión del 96%**
* **Retinopatía diabética No Proliferativa muy grave:** 75/83 imágenes clasificadas correctamente, 3 como "Normal", 2 como "Catarata" y 5 como "Glaucoma". **Precisión del 90%**
* **RD Proliferativa:** 67/70 imágenes clasificadas correctamente como "Retinopatía Diabética", 2 como "Glaucoma" y 1 como "Normal". **Precisión del 96%**
* **RD Proliferativa avanzada**: 60/69 imágenes clasificadas correctamente, con 6 clasificadas como "Glaucoma" y 3 como "Catarata". **Precisión 87%**

9. Conclusiones

9.1 Conclusiones por Ciclos

Ciclo 1

VGG19 fue seleccionada debido a su simplicidad y rendimiento constante, además de ser más fácil de ajustar y tener una facilidad de adaptación más notable que otras arquitecturas más complejas.

ResNet50 fue seleccionada por su balance entre profundidad y precisión, aprovechando sus bloques residuales para entrenar redes más profundas sin comprometer el rendimiento.

EfficientNet fue seleccionada por su eficiencia generalizada, logrando un rendimiento excelente con una arquitectura más optimizada, lo que la hace ideal para su despliegue en sistemas con recursos computacionales limitados.

Ciclo 2

EfficientNet fue seleccionada por su excelente rendimiento general y su eficiencia en la utilización de recursos. Con los mejores resultados tanto en precisión como en pérdida de validación; este modelo es ideal para la clasificación de enfermedades oculares, especialmente en equipos con limitaciones de recursos computacionales.

ResNet50 presenta un equilibrio entre precisión y pérdida de validación. Su arquitectura residual permite un entrenamiento más eficiente y estable, especialmente en tareas que requieren la extracción de características complejas como la clasificación de enfermedades oculares. Sin embargo, en algunas repeticiones mostró una pérdida de validación ligeramente más alta que EfficientNet, esta inestabilidad fue un factor clave para no seleccionarla como la arquitectura final para el ciclo 3.

VGG19, a pesar de que destaca por su simplicidad y capacidad de ofrecer resultados de alto rendimiento, mostró algunas variaciones en el valor de pérdida, esto y su sensibilidad a patrones de datos no lo hacen un candidato final para modelo final en el ciclo 3.

Ciclo 3

Los resultados confirman que EfficientNet ajustó correctamente sus parámetros durante el entrenamiento, destacando su capacidad de ajuste para lograr un rendimiento óptimo en la clasificación. Las tablas de varianza reflejan una notable estabilidad, mientras que las generales muestran que su precisión aumentó con respecto al ciclo anterior. Además, se puede observar que la tasa de aprendizaje utilizada permitió que el modelo mantuviera un equilibrio entre rendimiento y estabilidad.

9.2 Conclusiones Generales

Se puede evidenciar un avance positivo en este proyecto, tanto teórica como en la práctica. Desde una perspectiva teórica, se refuerza la idea de que los modelos de aprendizaje profundo pueden adaptarse y aprender patrones complejos en datos médicos, algo clave para abrir vías para la investigación en diagnósticos automatizados. Desde una perspectiva práctica, la implementación de estos modelos en entornos investigativos y académicos sientan una base para estudiar el uso de modelos de aprendizaje profundo para la clasificación de imágenes médicas.

Se concluye que EfficientNet es la arquitectura más eficaz, dentro de los límites para los que se diseñó, para la clasificación de enfermedades oculares en este proyecto. Aunque VGG19 y ResNet50 demostraron ser modelos sólidos con ventajas notables —como se puede evidenciar según las investigaciones documentadas en el Estado del Arte— EfficientNet se destacó en cada ciclo por su eficiencia en la utilización de recursos computacionales y su alta precisión y estabilidad en la validación; además, la capacidad de EfficientNet para mantener un equilibrio constante entre rendimiento y eficiencia, sobre todo en equipos con limitaciones de recursos.

10. Recomendaciones

A pesar de los resultados prometedores obtenidos en las pruebas, es evidente que el modelo aún enfrenta desafíos para clasificar correctamente ciertas imágenes, especialmente en la clase de retinopatía diabética, ya que el modelo suele confundirla con otras clases. Para mejorar la precisión y sensibilidad del modelo, se concluyó que es necesario llevar a cabo futuros trabajos que aborden estas limitaciones.

Una de las principales áreas de mejora identificadas es el entrenamiento del modelo con una mayor variedad de datasets; incluir un conjunto más amplio y diverso de imágenes de fondo de ojo que cubran diferentes patologías y estadios de las enfermedades podría aumentar la capacidad del modelo para distinguir mejor entre las clases (Hendrycks, 2016) Esto incluye:

* Expandir las categorías de las enfermedades para incorporar otras condiciones oftalmológicas que puedan ser relevantes en el diagnóstico diferencial.
* Ampliar la cantidad de imágenes por clase, incluyendo diferentes estadíos de la enfermedad, asegurando que el modelo tenga suficientes ejemplos para aprender de las variaciones dentro de cada categoría.
* Introducir imágenes de diferentes retinoscopios y cámaras, para entrenar al modelo sobre variaciones en la calidad y el formato de las imágenes.
* Se concluye entonces, a partir de las pruebas realizadas junto con profesionales de la salud, que es imprescindible mejorar a futuro la interfaz de usuario y la visualización de los resultados del diagnóstico para facilitar la interpretación de los mismos. Esto puede incluir un diseño más intuitivo, con gráficos que muestren la probabilidad de cada clase o una guía visual que indique la fiabilidad del diagnóstico en cada caso, lo cual sería de gran ayuda para los profesionales en la toma de decisiones.

11.  Referencias Bibliográficas

Al-Fahdawi, S. a.-W. (2023). Fundus-DeepNet: Multi-label deep learning classification system for enhanced detection of multiple ocular diseases through data fusion of fundus images. *Information Fusion, 102*, 102059.

Al-Khafaji, J. (2023). *Improving Diagnostic Safety and Quality.* <https://psnet.ahrq.gov/perspective/improving-diagnostic-safety-and-quality>

Alonso, J. L. (15 de 06 de 2022). *¿Qué es TensorFlow y para qué sirve?* incentro. <https://www.incentro.com/es-ES/blog/que-es-tensorflow>

Archana, E. a. (2023). Short Analysis of Machine Learning and Deep Learning Techniques used for Glaucoma Detection. *2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT).* 968--975.

Asana. (2024). *¿Qué es Kanban? Guía para principiantes para equipos ágiles*. <https://asana.com/project-management/kanban>

Basnet, J. a. (2020). A Novel Solution of Using Deep Learning for White Blood Cells Classification: Enhanced Loss Function with Regularization and Weighted Loss (ELFRWL). *Neural Processing Letters, 52*, 1517--1553.

Benjamin Baka, D. B. (2018). *Hands-On Data Structures and Algorithms with Python.* Packt Publishing.

Chen, X. a. (2022). A novel loss function of deep learning in wind speed forecasting. *Energy, 238*, 121808.

Cuidarte, R. (n.d). *Vista de Etiología de baja visión y ceguera en siete centros de referencia en Colombia entre los años 2012 a 2017.* <https://revistas.udes.edu.co/cuidarte/article/view/2036/2481>

das, t. (22 de 08 de 2024). *Google Colab: todo lo que necesita saber*. GeekFlare. <https://geekflare.com/es/google-colab/>

Dettmers, T. (31 de 12 de 2021). *Understanding Convolution in Deep Learning*. <https://timdettmers.com/2015/03/26/convolution-deep-learning/>

Documentation, J. N. (2024). *jupyter Notebook Documentation — Jupyter Notebook 7.3.0b0 documentation*. <https://jupyter-notebook.readthedocs.io/en/latest/>

Documentation, J. W. (2024). *Jupyter Widgets — Jupyter Widgets 8.1.5 documentation*. [https://ipywidgets.readthedocs.io/}](https://ipywidgets.readthedocs.io/%7d)

eLearning, E. E. (2023). *Qué es Aprendizaje profundo*. <https://editorialelearning.com/blog/deep-learning/>

Elmoufidi, A. a. (2021). EfficientNetB3 Architecture for Diabetic Retinopathy Assessment using Fundus Images. <https://doi.org/10.21203/rs.3.rs-609899/v1>

Erdinest, N. a. (2024). Artificial Intelligence in Ophthalmology. *Harefuah, 163*(1), 37--42.

Erlangen-Nürnberg, L. F. (n.d.). *High-Resolution FundUS (HRF) image database*. <https://www5.cs.fau.de/research/data/fundus-images/>

Gong, Y. a. (2023). A survey on dataset quality in machine learning. *Information And Software Technology, 162*, 107268. <https://doi.org/10.1016/j.infsof.2023.107268>

*Gov.ar*. (s.f.). <https://ri.conicet.gov.ar/bitstream/handle/11336/213918/CONICET_Digital_Nro.0c21222c-2fe7-4beb-b073-1ac133109e88_B.pdf?sequence=2&isAllowed=y>

Hendrycks, D. a. (2016). *A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks*. <https://arxiv.org/abs/1610.02136>

Jupyter, P. (2020). *Project Jupyter Documentation — Jupyter Documentation 4.1.1 alpha documentation*. Jupyter. <https://docs.jupyter.org/en/latest/#what-is-a-notebook>

Jupyter, P. (2024). *Project Jupyter*. <https://jupyter.org/>

Kaelbling, L. P. (1996). Reinforcement Learning: A Survey. *Journal of Artificial Intelligence Research, 4*, 237--285. <https://doi.org/10.1613/jair.301>

Kar, K. a. (2022). Interpretability of artificial neural network models in artificial intelligence versus neuroscience. *Nature Machine Intelligence, 4*, 1065--1067. <https://doi.org/10.1038/s42256-022-00592-3>

KILIÇARSLAN, S. a. (2021). An overview of the activation functions used in machine learning algorithms. *Journal of New Results in Science, 10*(3). <https://doi.org/10.54187/jnrs.1011739>

Kufel, J. a.-Ł. (2023). What Is Machine Learning, Artificial Neural Networks and Deep Learning?—Examples of Practical Applications in Medicine. *Diagnostics, 13*(15), 2582. <https://doi.org/10.3390/diagnostics13152582>

LeCun, Y. a. (2015). Aprendizaje profundo. *Nature, 521*(7553), 436--444. <https://doi.org/10.1038/nature14539>

Li, S. (2023). *Reinforcement Learning for Sequential Decision and Optimal Control.* Springer Verlag, Singapore. <https://doi.org/10.1007/978-981-19-7784-8>

Lim, W. a.-Y.-C.-W.-K.-J.-S.-L. (2022). Use of multimodal dataset in AI for detecting glaucoma based on fundus photographs assessed with OCT: focus group study on high prevalence of myopia. *BMC Medical Imaging, 22*. <https://doi.org/10.1186/s12880-022-00933-z>

Liu, X. a. (2021). A review of deep-learning-based medical image segmentation methods. *Sustainability (Switzerland), 13*(3). <https://doi.org/10.3390/su13031224>

Mascarenhas, S. a. (2021). A comparison between VGG16, VGG19 and ResNet50 architecture frameworks for Image Classification. En *2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON)* (págs. 96--99).

Mitaritonna, A. (2019). *¿Qué es y para qué sirve el Aprendizaje Profundo (Aprendizaje profundo)?* <https://es.linkedin.com/pulse/qu%C3%A9-es-y-para-sirve-el-aprendizaje-profundo-deep-mitaritonna>

Mostafa, K. a. (2023). Deep Learning-Based Classification of Ocular Diseases Using Convolutional Neural Networks. En *2023 Intelligent Methods, Systems, and Applications (IMSA)* (págs. 446--451).

Naik, D. P. (2023). *Conceptualizing Python in Google COLAB Hands-on Practical Sessions.* Shashwat Publication.

ngrok. (n.d.). *ngrok | API Gateway, IoT Device Gateway, Secure Tunnels for Containers, Apps & APIs*. <https://ngrok.com/>

Nie, D. a. (2017). Medical Image Synthesis with Context-Aware Generative Adversarial Networks. *Medical Image Computing and Computer Assisted Intervention − MICCAI 2017. MICCAI 2017. Lecture Notes in Computer Science, 10435*. <https://doi.org/10.1007/978-3-319-66179-7_48>

Ortiz, S. a. (2023). Deep Learning-Based Ocular Disease Classification in Fundus Images. 1--6. <https://doi.org/10.1109/C358072.2023.10436234>

Pal, S. a. (2022). Artificial Intelligence and Machine Learning in Manufacturing. En S. S. Manufacturing, *Digital Twin – Fundamental Concepts to Applications in Advanced Manufacturing* (págs. Springer, Cham). 10.1007/978-3-030-81815-9\_6.

Pan, Y. a. (2023). Fundus image classification using Inception V3 and ResNet-50 for the early diagnostics of fundus diseases. *Frontiers in Physiology, 14*, 1126780. <https://doi.org/10.3389/fphys.2023.1126780>

Philosophy, S. E. (2021). *Scientific Method*. <https://plato.stanford.edu/entries/scientific-method/>

Pręgowska, A. a. (2021). What Is An Artificial Neural Network And Why Do We Need It? *Frontiers for Young Minds*.

Python software, f. (2024). Python Tutorial Release 3.12.2 Guido van Rossum and the Python development team. *The Python Tutorial*.

Rajpurkar, P. a. (2017). CheXNet: Radiologist-level pneumonia detection on chest X-rays with aprendizaje profundo. *ArXiv, abs/1711.05225*.

Rios Zuluaga, J. D. (2017). Pautas para el examen oftalmológico. Enfoque para el estudiante de medicina y el médico general. *Universitas Médica, 58*(2). <https://doi.org/10.11144/javeriana.umed58-2.ofta>

Russell, S. J. (2010). Artificial intelligence: A modern approach. En *Artificial intelligence: A modern approach.* Prentice Hall.

Salud, O. P. (n.d.). *Salud visual.* <https://www.paho.org/es/temas/salud-visual>

Shukla, S. a. (2023). *Introduction to Deep Learning.* Springer. <https://doi.org/10.1007/978-3-031-13577-4_15>

Simonyan, K. a. (2014). *Very Deep Convolutional Networks for Large-Scale Image Recognition*. <https://arxiv.org/abs/1409.1556>

Sunchalin, A. M. (2021). Basic Elements of Machine Learning. *Fundamental Research*, 93-97. <https://doi.org/10.17513/fr.43044>

Tanvir, K. a. (2023). *Clinical Insights through Xception: A Multiclass Classification of Ocular Pathologies.* Tuijin Jishu/Journal of Propulsion Technology. <https://doi.org/10.52783/tjjpt.v44.i4.2018>

TensorFlow. (01 de 09 de 2021). *TensorFlow lite*. TensorFlow.org. <https://www.tensorflow.org/lite/guide?hl=es-419>

Venkatesan, R. a. (2017). *Convolutional Neural Networks in Visual Computing: A Concise Guide.* CRC Press. <https://doi.org/978-1-351-65032-8>

Vipas.Ai. (2024). *RESNet-50: A robust Image Classification Model*. <https://medium.com/@vipas.ai/resnet-50-a-robust-image-classification-model-9bd3fffbc3e3>

Voila. (2020). *Table of contents — Voila 0.5.8 documentation*. Voila. <https://voila.readthedocs.io/>

Wolansky, I. (2021). A Deep Dive Into the Basics of Deep Learning},. *Proceedings of the Shevchenko Scientific Society. Medical Sciences, 65*(2). <https://doi.org/10.25040/ntsh2021.02.23>

Zhu, S. a. (2022). Screening of Common Retinal Diseases Using Six-Category Models Based on EfficientNet. *Frontiers in Medicine, 9*. <https://doi.org/10.3389/fmed.2022.808402>

12. Apéndices

Apéndice A: Tablas ciclos

A continuación, se referencia la carpeta en dónde se encontrarán todas las tablas de cada ciclo:

<https://drive.google.com/drive/folders/1CyxsNqgrOtgnm7TnzoChGdlmbFfC2asE?usp=drive_link>

Apéndice B: Colabs

A continuación, se referencia la carpeta en dónde se encontrarán todos los notebook creados para avanzar en cada ciclo de la fase 2; en donde los que contienen v1 pertenecen al ciclo 1, v2 a ciclo 2 y v3 a ciclo 3:

<https://drive.google.com/drive/folders/1PHim_PD6GTXtqLOYB84IZSaTXoMZWLHI?usp=drive_link>

Apéndice C: Acta de respuesta a Cambio de Título del proyecto

A continuación, se encuentra el acta de la respuesta a solicitud de cambio de Título del proyecto de grado.

**1/1**

Bucaramanga, 16 de abril de 2024

MARIA FERNANDA CUBIDES JAIMES

C.C. N° 1.005.163.255

JULIANA ALEJANDRA ARENAS LOBO

C.C. N° 1.005.107.810

ERICK SANTIAGO RIAÑO PEREZ

C.C. N° 1.005.162.937

**Programa Pregrado – Ingeniería de Sistemas**

REF: Respuesta a solicitud de cambio de Titulo del proyecto de grado. Cordial saludo.

De acuerdo a su solicitud radicada a través de la cuenta de correo [secre.investigaciones@udi.edu.co,](mailto:secre.investigaciones@udi.edu.co) nos permitimos informar que: El Comité de Investigaciones, luego de revisar la solicitud y la argumentación descrita por Los estudiantes, APRUEBA la propuesta de Titulo del Proyecto de Grado registrado en la Institución bajo el código **IS-2024-1-08** Por lo tanto, las modificaciones son las siguientes:

**Titulo**

* ***Aplicación de técnicas de deep learning para el diagnóstico asistido de patologías oculares que causan ceguera irreversible.***

Los estudiantes deben enviar la siguiente documentación a la Dirección de Investigaciones a través de la cuenta de correo (secre.investigaciones@udi.edu.co):

* Copia de esta comunicación, en donde se evidencia la autorización de la modificación de título y/o objetivos de su Proyecto de Grado.
* Copia digital del Anteproyecto de Grado, en donde se incluya la modificación realizada y los ajustes a que haya lugar en otros apartes de dicho documento (los cambios solicitados pueden modificar, entre otros, el diseño metodológico del proyecto).

Los estudiantes deben anexar copia de esta comunicación en el documento investigativo final. Para mayor información comuníquese con la Dirección de Investigaciones al correo electrónico: [secre.investigaciones@udi.edu.co o](mailto:secre.investigaciones@udi.edu.coo) al PBX 6352525 EXT. 214.

Atentamente,

Apéndice D: Acta de respuesta a Cambio de Título y Objetivos del proyecto

A continuación, se encuentra el acta de la respuesta a solicitud de cambio de Título y objetivos del proyecto de grado.

Bucaramanga, 20 de septiembre de 2024



MARIA FERNANDA CUBIDES JAIMES C.C. N° 1.005.163.255

JULIANA ALEJANDRA ARENAS LOBO C.C. N° 1.005.107.810

ERICK SANTIAGO RIAÑO PEREZ C.C. N° 1.005.162.937

**Programa Pregrado – Ingeniería de Sistemas**

REF: Respuesta a solicitud de cambio de Titulo del proyecto de grado. Cordial saludo.

De acuerdo a su solicitud radicada a través de la cuenta de correo [secre.investigaciones@udi.edu.co,](mailto:secre.investigaciones@udi.edu.co) nos permitimos informar que: El Comité de Investigaciones, luego de revisar la solicitud y la argumentación descrita por Los estudiantes, APRUEBA la propuesta de Titulo del Proyecto de Grado registrado en la Institución bajo el código **IS-2024-1-08** Por lo tanto, las modificaciones son las siguientes:

Titulo

* ***Clasificación de patologías oculares que causan ceguera irreversible usando arquitectura de aprendizaje profundo.***

Objetivos específicos

1. *Seleccionar un conjunto de datos(dataset) representativos de imágenes de fondo de ojo que con tenga las características morfológicas de enfermedades (retinopatía diabética, cataratas y glaucoma), así como de tejido sano, generando un conjunto de datos útil.*
2. *Revisar el estado del arte buscando modelos de Aprendizaje profundo probadas en imágenes de tejidos similares a fondos de ojos, tal que puedan detectar, procesar y clasificar imágenes de retinoscopía.*
3. *Escoger un modelo de Aprendizaje profundo para la clasificación de patologías oculares que causan ceguera irreversible a través de pruebas exhaustivas, usando el conjunto de imágenes seleccionado.*
4. *Integrar el modelo de Aprendizaje profundo seleccionado mediante una interfaz tipo Python notebook*

Los estudiantes deben enviar la siguiente documentación a la Dirección de Investigaciones a través de la cuenta de correo (secre.investigaciones@udi.edu.co):

* Copia de esta comunicación, en donde se evidencia la autorización de la modificación de título y/o objetivos de su Proyecto de Grado.
* Copia digital del Anteproyecto de Grado, en donde se incluya la modificación realizada y los ajustes a que haya lugar en otros apartes de dicho documento (los cambios solicitados pueden modificar, entre otros, el diseño metodológico del proyecto).

Los estudiantes deben anexar copia de esta comunicación en el documento investigativo final. Para mayor información comuníquese con la Dirección de Investigaciones al correo electrónico: [secre.investigaciones@udi.edu.co](mailto:secre.investigaciones@udi.edu.co) o al PBX 6352525 EXT. 214.



Atentamente,

Apéndice E: Datasets

A continuación, se encuentran los dataset usados en el ciclo 1 (Dataset Binario) y ciclo 2, 3 (Dataset de 4 clases). Adicionalmente, también se encuentran los datasets usados durante la fase de pruebas de la interfaz:

**DATASET 2 CLASES:** <https://drive.google.com/drive/folders/1LLMej2ygqKgYBnij3S36EXynOE2SgyEl?usp=sharing>

**DATASET 4 CLASES:** <https://drive.google.com/drive/u/3/folders/1aW7JHfhAosBFQkuhOFFYHZbyEb_Rk7Si>

**Dataset prueba 1: Erlangen-Nürnberg, L. F. High-Resolution FundUS (HRF) image database:** <https://drive.google.com/drive/u/2/folders/1Z30LCgc5V2z_bujxj1UmuUDFFrjSoyOM>

**Dataset prueba 2: Conjunto de imágenes de fondo de ojo para el estudio de la retinopatía diabética:** <https://drive.google.com/drive/u/3/folders/1dVRvNZEAHh64kVRInfrjm9iiw7tThg9t>

Apéndice F: Diagrama de flujo del funcionamiento de los colabs y generación de dataframes:

<https://drive.google.com/file/d/187RxnuvyfeEIaZwdxG0Rce9F7pS9YtmC/view?usp=sharing>

Apéndice G: Diagrama de secuencia y de flujo del funcionamiento de la interfaz y su relación con el modelo:

<https://drive.google.com/file/d/1SGRryjJWA2uxiwwV2G3DJ0pfmSZJK-UX/view?usp=sharing>

Apéndice H: Videos de YouTube:

* 1. **Manual de uso colabs todos los ciclos y reentrenamiento (Cómo ejecutar los colabs):** <https://www.youtube.com/watch?v=0qeK9bVvN4U>
  2. **Manual de usuario interfaz (Cómo ejecutar la interfaz):** <https://youtu.be/BesBzNgZOwA>
  3. **Código de la interfaz (Explicación a detalle y con diagramas el funcionamiento y relación entre interfaz-modelo): Relación entre modelo interfaz y usuario:** <https://www.youtube.com/watch?v=KhGH2u4FrDs>
  4. **Código colabs (Explicación a detalle y con diagramas del funcionamiento de los colabs, cómo funcionan, qué relación tiene con los hiperparámetros y modelo final):** <https://www.youtube.com/watch?v=ht0h_tF6T5Y>

Apéndice I: Diagrama del Estado del arte con los parámetros y arquitecturas utilizadas:

<https://drive.google.com/file/d/14gjHMrXUA5Hftttb68DzLiuRInDNvFuO/view?usp=sharing>

Apéndice J: Link interfaz + modelo final:

https://drive.google.com/drive/u/3/folders/12Q8SwTEA0qRNaUjPNd1Rvj62i-bit2HU