

Universidad de Lima
Facultad de Ingeniería
Carrera de Ingeniería de Sistemas



IMPLEMENTACIÓN DE UN MODELO ENSAMBLADO PARA LA DETECCIÓN TEMPRANA DE RETINOPATÍA DIABÉTICA BASADO EN REDES NEURONALES CONVOLUCIONALES

Tesis para optar el Título Profesional de Ingeniero de Sistemas

Luis Alberto Ortiz Valdivia

Código 20172426

Asesor

Jean Pierre Tincopa Flores

Lima – Perú

Diciembre de 2025

Implementación de un Modelo Ensamblado para la Detección Temprana de Retinopatía Diabética Basado en Redes Neuronales Convolucionales

Ortiz Valdivia Luis Alberto

20172426@aloe.ulima.edu.pe

Universidad de Lima

Resumen:

La retinopatía diabética (RD) es una de las complicaciones más comunes de la diabetes, caracterizada por el daño progresivo de los vasos sanguíneos de la retina. A largo plazo, puede causar ceguera irreversible debido al exceso de glucosa en sangre; por ello, su detección temprana resulta fundamental para evitar su avance y adoptar medidas de control adecuadas. En este contexto, las tecnologías de visión computacional han demostrado una alta precisión en la identificación de lesiones ocasionadas por diversas enfermedades oculares. El objetivo de la presente investigación es determinar el nivel de retinopatía diabética mediante un modelo ensamblado que combina las redes neuronales InceptionV3, VGG16 y DenseNet121, con el fin de reducir los efectos severos de la enfermedad. La metodología aplicada incluyó la modificación del conjunto de datos, el preprocesamiento, la selección de la mejor combinación y la evaluación de los modelos. Se utilizó el dataset APTOS 2019, que contiene imágenes clasificadas en cinco grados de RD, posteriormente reorganizados en tres categorías: no RD (N), RD temprano (T) y RD avanzada (A). Durante el preprocesamiento se aplicó desenfoque gaussiano para resaltar cavidades retinianas y la técnica *Blob Detection* para identificar microaneurismas. Además, se realizaron tres pruebas de validación cruzada para seleccionar los mejores modelos. Los modelos fueron optimizados mediante el ajuste de capas adicionales y la descongelación de las últimas capas para especializar las predicciones. El ensamblaje por promedio de las tres redes alcanzó una exactitud general del 91.65% en la fase de evaluación, evidenciando la efectividad del modelo propuesto y la utilidad del *Blob Detection* en la detección temprana de la RD.

Palabras Clave: retinopatía diabética, detección de blobs, detección de microaneurismas, densenet121, inceptionv3, ensamblado, vgg16

Abstract:

Diabetic retinopathy (DR) is a common complication of diabetes which consists of damage to the retina's blood vessels. Over the long term, DR can produce blindness in diabetic people due to irreversible damage caused by excess sugar in the blood stream, therefore, early identification can prevent the complication's development and at the same time allow for the adoption of measures which help control diabetes. Advancements in computational vision have expanded to a point where they can detect lesions caused by different sicknesses which great precision. The purpose of this investigation centers on being able to determine diabetic retinopathy utilizing an assembled model through the networks: InceptionV3, VGG16, and DenseNet121 to prevent fatal effects for diabetics. To accomplish this objective, a methodology based on the medication of the dataset, preprocessing, best model selection, and testing was employed. The APTOS 2019 dataset was employed, this dataset presents images of DR in 5 degrees, of which in preprocessing, were modified through gaussian blur to highlight cavities in the retina and Blob Detection to highlight microaneurysms. Through the same means, the dataset was modified so only 3 degrees would be present: early DR (T), advanced DR (A), and no DR (N). In the investigation, Cross-Validation was used 3 times to obtain the best models for use, this process was accomplished for each individual model and in the assembly. For the training of the proposed models, extra layers were added to improve performance, and the last layers were unfrozen for each model to specialize their

predictions for the detection of DR. The averaging ensemble of the three models achieved an overall accuracy of 91.65% was obtained highlighting the efficiency of the model in detecting DR in its early stages. Therefore, the results demonstrate the effective use of Blob Detection as a tool for detecting DR and the use of averaged assembly to improve the accuracy of the models.

Keywords: diabetic retinopathy, blob detection, microaneurysm detection, densenet121, inceptionv3, ensemble, vgg16

1. INTRODUCCIÓN

La relación entre la retina y la diabetes es una complicación que fue observada por primera vez en 1856 por Eduard Jaeger, quien identificó manchas amarillentas en la retina. Actualmente, esta patología afecta a millones de personas en todo el mundo. Según datos de La Organización Mundial de la Salud WHO (2024), el número de personas con diabetes aumentó de 200 millones en 1990 a 830 millones en 2022. Además, en el año 2022 más de la mitad de los adultos mayores no recibían medicamentos para la enfermedad. Cuanto mayor sea la duración de la enfermedad en un paciente, mayor será la posibilidad de que aparezcan complicaciones. Una de las más relevantes es la Retinopatía diabética (RD), que se origina por daños en los vasos sanguíneos de la retina ocasionados por el exceso de azúcar en la sangre. En su fase inicial, Ghosh (2024) señala que la patología puede mantenerse de una manera asintomática, por lo que detectarla tempranamente puede salvar al paciente de perder la visión. A medida que la RD se agrava, es más fácil distinguir los microaneurismas, *wool spots*, lesiones rojas y exudados. Patni (2025) declaró que el reconocimiento temprano de la RD sumado a una intervención médica inmediata permite reducir las consecuencias de las lesiones; sin embargo, la detección temprana no es sencilla, ya que los cambios iniciales ocurren a una escala diminuta. Por este motivo, distintos investigadores enfocaron sus esfuerzos en conseguir distintos métodos de detección temprana de RD mediante herramientas computarizadas.

Existen varios modelos que logran detectar y/o graduar la RD empleando distintos enfoques, por un lado, se puede mencionar a Kumar (2018) o a Kanimozhi (2020) quienes evidenciaron las lesiones en la retina aplicando filtros en la imagen para posteriormente medir la gravedad de la enfermedad según la cantidad y características de las lesiones. Por otro lado, las investigaciones de Samanta (2020) o Bhardwaj (2021) representan metodologías basadas en el uso de redes neuronales convolucionales (CNN) entrenadas a partir de un dataset de imágenes de RD. Una de las ventajas que ofrecen estos modelos es que se analiza absolutamente toda la imagen, por lo que cualquier característica que sea destacable para la detección de la enfermedad será tomada en cuenta sin depender de las lesiones que se planteen detectar. Investigadores como Abini (2025) y Lands (2020) obtuvieron resultados superiores a 80% en la exactitud de sus modelos CNN, sin embargo, se presentaron dificultades al momento de detectar la RD en un estatus temprano ya que sus modelos confundían una retina sana con una RD leve o moderada. En cuanto a los *dataset* empleados para el entrenamiento de estos modelos, existen varios de uso común en la comunidad científica, entre los cuales destacan APTOS 2019 que presenta una gran cantidad de imágenes de alta calidad, pero no tan graduada, MESSIDOR que posee desbalanceo de imágenes por grados, pero con una enorme calidad, E-ophtha-EX donde se resaltan los exudados en las imágenes de RD y DIARETDB1 que posee homogeneidad de imágenes, pero con una menor cantidad.

La creación de un modelo de detección temprana de RD sigue siendo uno de los retos que se debe seguir abordado, por lo que esta investigación propone el uso de un modelo ensamblado de las redes neuronales convolucionales InceptionV3, DenseNet y VGG16 para la detección temprana de retinopatía diabética junto a un preprocesamiento enfocado en resaltar los microaneurismas en las imágenes con el objetivo de facilitar el aprendizaje de una etapa temprana.

La estructura del artículo es la siguiente: en la sección 2, se presentará en primera instancia una revisión del estado del arte; en la sección 3, el marco teórico; en la sección 4, la metodología del modelo ensamblado propuesto. Posteriormente en la sección 5, la experimentación que se siguió para la creación de este, en la sección 6, la discusión de los resultados y finalmente en la sección 7, las conclusiones.

2. ESTADO DEL ARTE

Los estudios revisados para la realización de este reporte intentaron detectar y/o graduar con mayor eficiencia la RD. Entre estos estudios se encuentran propuestas muy variadas entre sí; se pueden resaltar distintos enfoques para definir si una imagen de fondo de ojo posee signos de RD o no. Investigaciones como la de Kumar (2018) logró evidenciar RD utilizando una metodología de preprocesamiento intensivo. Por su parte, otro conjunto de estudios se apoyó en construir un modelo mediante CNN para detectar la presencia de RD a partir de un conjunto de imágenes, cómo lo hizo Zago (2020).

En el ámbito de las investigaciones dirigidas a identificar lesiones mediante el procesamiento de imágenes, destaca el estudio de Kumar (2018). Este trabajo se centró en detectar microaneurismas a partir de imágenes de fondo de ojo; para ello aplicó un preprocesamiento compuesto por ecualización de histogramas (CLAHE), extracción del canal verde, filtros de promediado, operaciones morfológicas y análisis de componentes principales (PCA). A continuación, construyó un modelo de predicción mediante SVM lineal que empleó como características el área y la cantidad de microaneurismas detectados. Este modelo logró una sensibilidad del 96 % y una especificidad del 92 %. Adal (2014) utilizó la determinante de la matriz Hessiana en cada píxel para extraer regiones circulares oscuras potenciales microaneurismas; de esta forma pudo extraer características de los candidatos mediante Speeded Up Robust Features (SURF) y validarlas con SVM obteniendo una sensibilidad de 81.08% y precisión de 92.31%. Por su parte, Kanimozhi (2020) también utilizó CLAHE para mejorar la iluminación, PCA para extraer vasos sanguíneos y procesos morfológicos para localizar microaneurismas; sin embargo, su modelo incluyó además hemorragias y exudados duros en las imágenes, alcanzando una exactitud del 97,87 %, superando al de Kumar (2018). Dharmana(2020) y Tavakoli (2020) emplearon el Laplaciano de Gaussiano (LoG) de distintas maneras. Tavakoli (2020) usó el filtro gaussiano para crear candidatos de microaneurismas una vez segmentado los vasos sanguíneos de la imagen. Por otro lado, Dharmana (2020) usó un tipo de blob detection basado en LoG para obtener la cantidad, el radio y la intensidad máxima de los blobs en imágenes de RD grave, mediana, leve o inexistente y posteriormente realizar la predicción usando Support Vector Machine. Los últimos dos autores referenciados basaron sus estudios en detectar el grado de la retinopatía diabética utilizando métodos que facilitan la detección de microaneurismas, Dharmana(2020) obtuvo una precisión de 83% para su partición de pruebas, teniendo dificultades a diferenciar la RD leve con moderada, Tavakoli(2020) comprobó la validez de los candidatos a microaneurismas usando el Log, obteniendo una sensibilidad de 90% demostrando la efectividad del uso de estos filtros para obtener candidatos potenciales.

Un rumbo que han comenzado a tomar las investigaciones más recientes es el uso de redes neuronales convolucionales (CNN) para extraer parámetros y predecir la RD. Para este tipo de modelos se puede recurrir tanto a una arquitectura preentrenada como a una construida desde cero. Abini (2025) empleó el modelo preentrenado Inception V3 para detectar RD, su propuesta se caracterizó por un método de aumento de imágenes mediante SMOTE y modificaciones en las capas finales del modelo, además de dejar congeladas todas las capas iniciales durante el entrenamiento. El modelo de Abini (2025) logró una exactitud del 91,3 %. Por otro lado, Hathwar (2019) ejecutó un proceso muy similar, pero empleó dos arquitecturas diferentes —Inception-ResNet-V2 y Xception— consiguiendo una exactitud del 88 % en ambos casos. Samanta (2020)

y Bhardwaj (2021) utilizaron modelos basados en DenseNet 121 e Inception V3 respectivamente, aunque su distinción radicó en el enfoque del preprocesamiento de las imágenes. Samanta empleó CLAHE para resaltar regiones pequeñas y Gaussian Blur para una mejor extracción de características, además de aplicar descenso de gradiente para optimizar los resultados; mientras que Bhardwaj (2021) también empleó CLAHE, pero se centró en eliminar el disco óptico mediante operaciones morfológicas, entrenando la imagen dividida en cuatro partes y cuatro modelos con arquitectura Inception V3: obtuvieron una exactitud del 97 % y del 93,3 % respectivamente. Finalmente, al igual que Abini, Hammad (2021) utilizó Inception V3, pero añadió promedio global (global average pooling) para reducir el sobreajuste, y una capa GAP seguida de normalización por lotes (batch normalization) y dropout antes de pasar a la capa densa, donde se aplicó la función de activación ReLU. También hay casos de modelos basados en arquitecturas VGG, como en el caso de Hayati (2023), que empleó CLAHE para el preprocesamiento de las imágenes y mejorar el contraste, logró una mejoría notable en el entrenamiento con VGG16 y una exactitud del 91,87 %. Yan (2026) usó el dataset de APTOS 2019 con imágenes aumentadas para el balanceo, donde para el preprocesamiento empleó el canal verde de las imágenes para posteriormente aplicar CLAHE y Laplace Sharpening con un clipLimit de 7. Yan (2026) mejoró la extracción de características globales de EfficientNetB0 añadiendo un módulo de Atención Residual Local-Global (RLGA) y concatenando los resultados de MobileNetV1 para la predicción, obteniendo un 94.58% de exactitud de evaluación con los 5 grados. Del mismo modo, Shamrat (2024) usó APTOS 2019 aumentado para el balanceo, usando un preprocesamiento basado en median filter para reducción de ruido y Gamma Correction para mejorar el contraste. El enfoque de Shamrat (2024) se basó en demostrar la superioridad de su modelo DRNet13 a comparación de 14 modelos pre-entrenados conocidos, obteniendo un 97% de exactitud en evaluación.

Existe una ramificación de los modelos creados en base a CNNs que consisten en unir diversas redes con resultados óptimos para crear una red más robusta, esta metodología es conocida como *Ensemble*. Jinfeng (2020), Hammad (2021) y Alyoubi (2021) utilizaron específicamente la técnica de Average Ensemble. Jinfeng (2020) realizó un ensamblado compuesto entre red InceptionV3, DenseNet y ResNet50; este ensamblado obtuvo una exactitud de 80%, siendo este un resultado superior que el resultado de las redes de manera individual, por otro lado, Hammad (2021) ensambló las redes DenseNet, Inception y Xception, lo cual hizo que la exactitud incrementará 0.4% con respecto a una red individual. Alyoubi (2021) combinó YOLOv3 y CNN512 para mejorar la exactitud un 0.4%. Otro caso similar es el de Karki (2021), el cual ensambló 5 redes EfficientNet entrenadas con imágenes de distintos tamaños, de este modo el modelo final obtenía un 2% adicional de exactitud que las redes trabajando de manera independiente. Pavithra (2024) realizó un modelo ensamblado empleando VGG19, InceptionV3 y ResNet50 obteniendo una exactitud de 98.47% en un modelo con 2 clases de imágenes, superando en 3% a las exactitudes obtenidas individualmente. Las investigaciones hechas con modelos ensamblados demostraron que, si se tiene 2 o más modelos con una exactitud alta, al ensamblarlos crearon un aumento leve en la exactitud, generalmente este incremento tiende a subir a medida que más modelos distintos se encuentren implicados.

A partir de la literatura revisada, se destaca el uso de *blob detection*, CLAHE y de operaciones morfológicas para detectar y resaltar los microaneurismas y exudados. Del mismo modo, el uso de modelos pre-entrenados como InceptionV3, DenseNet121, ResNet y Xception poseen antecedentes en la detección de RD, siendo bastantes de estos ensamblados mediante votación o promedio para obtener una mayor exactitud.

De igual modo, se observa una polaridad en las metodologías dirigidas a la predicción precoz de la RD. Mientras que algunos estudios emplean CNN sin aplicar un preprocesamiento específico para resaltar las malformaciones asociadas a la RD, investigaciones anteriores —como las de Adal (2014) y Kumar (2018)— se centraron en la detección de cavidades de origen

retinopático. Son escasas las aproximaciones que combinan ambas estrategias y evalúan su integración metodológica.

Resulta evidente la carencia de una metodología sistemática para la selección del modelo óptimo: la mayoría de los trabajos elige aquel que alcanza la mayor exactitud sin considerar factores cruciales como la procedencia, calidad y homogeneidad de las imágenes empleadas para entrenamiento y validación. En este sentido, el uso de oversampling para ampliar conjuntos de datos, aunque frecuente, presenta limitaciones en el ámbito de la RD: introducir imágenes recortadas, de baja resolución o generadas artificialmente produce mejoras de exactitud de carácter artificial, comprometiendo la validez del modelo.

Finalmente, las investigaciones centradas en el ensamblaje de modelos, como los trabajos de Hammad (2021) y Pavithra (2024), Shamrat(2024), tienden a combinar aquellos modelos con mejor rendimiento sin analizar suficientemente cómo un preprocesamiento optimizado podría potenciar estas combinaciones. Esta omisión deja sin resolver hasta qué punto el rendimiento podría mejorarse mediante un flujo metodológico integral que abarque desde el preprocesamiento hasta la arquitectura de ensamblado.

3. MARCO TEÓRICO

3.1 DIABETES

La Dra Molina (2012) señaló que la diabetes es un grupo de alteraciones metabólicas que se caracteriza por la hiperglucemia crónica. La hiperglucemia con el tiempo provoca daños y fallas en varios órganos. Las causas de la diabetes derivan según el tipo. Para la diabetes tipo 1, es una enfermedad autoinmune en donde no se produce suficiente insulina. La diabetes tipo 2 es causada por herencia de otros familiares. Santana (2023) declaró que entre las consecuencias que ocurre si no se mantiene un control de la enfermedad se encuentra la retinopatía, insuficiencia renal crónica, enfermedades cardiovasculares, isquemia, hipertensión arterial.

3.2 RETINOPATÍA DIABÉTICA

La RD es una de las lesiones que desarrolla la diabetes a medida que la enfermedad se encuentra avanzando. Según Yao (2021), la RD es una lesión secundaria ocasionada por daño ocasionado a los pequeños vasos sanguíneos ubicados en la retina debido al aumento de la glucosa. A la RD se le conoce como una complicación neurovascular inflamatoria con deterioro neurológico que eventualmente afectará al grado de visión del afectado, llevándolo hasta la ceguera. Mientras la RD va empeorando, se pueden observar distintas lesiones en la retina que dan indicios de la gravedad de la degeneración.

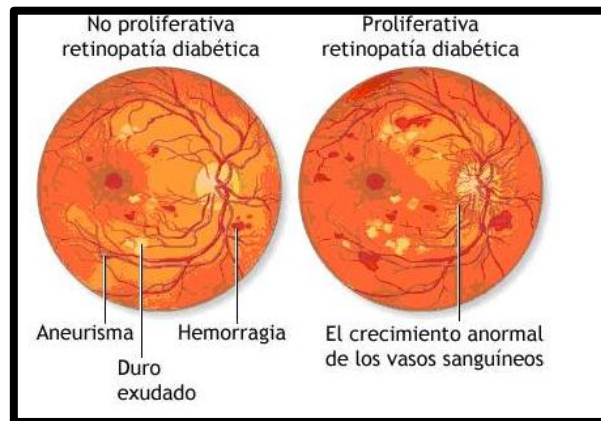
3.2.1 CARACTERÍSTICAS DE LA RETINOPATÍA DIABÉTICA

Existen características que dan indicios a la existencia de RD, algunos empiezan a formarse en una etapa más tardía y otros en una más temprana, entre estas características tenemos los mencionados por Carrera (2017) y mostrados gráficamente en la Figura 3.1.

- **Vasos Sanguíneos:** Cuando se les presenta RD, estas estructuras empiezan a tener un crecimiento anormal, por este motivo en varios estudios se obtiene la densidad de los vasos para confirmar si tienen un tamaño anormal
- **Microaneurismas:** Son el primer signo que indica la presencia de RD. Los microaneurismas son pequeños bultos ubicados en los vasos sanguíneos, que aparecen como pequeños puntos cerca a estos vasos.
- **Exudados duros:** Como menciona Yu (2017), los exudados son el resultado de la descomposición de la retina sanguínea y termina produciendo una fuga de lípidos y proteínas en los vasos sanguíneos.

Figura 3.1

Imagen de características de retinopatía diabética



Nota. Adaptado de ¿Qué es la retinopatía diabética?, por Grupo Gamma, 2013

Todas estas características mencionadas, generalmente son identificadas a través de una imagen de fondo de ojo tomada por un retinógrafo, existen dataset públicos con imágenes donde se presenta estas lesiones. En la Tabla 3.1 se muestra las siguientes correspondientes a la RD:

Tabla 3.1

Grados de retinopatía diabética

Grado	Características
Retinopatía Diabética Leve	Presencia de microaneurismas
Retinopatía Diabética Moderada	Presencia de microaneurismas en los 4 cuadrantes de la retina y exudados duros o arrosamientos venosos en solo un cuadrante
Retinopatía Diabética Severa	Más de 20 microaneurismas en cada cuadrante, diversos exudados duros y arrosamientos venosos
Retinopatía Diabética Proliferativa	Hemorragia vítrea o prerretiniana. neovascularización

Para tratar la retinopatía diabética se requiere controlar adecuadamente la diabetes con control de glucosa en sangre, presión arterial y colesterol. Posterior a lo indicado, se complementa con fármacos como el Anti-VEGF y Corticoides para parar la inflamación. También se puede tratar mediante cirugía vitrectomía en casos más avanzados.

3.3 DATASETS DE RETINOPATÍA DIABÉTICA

Entre los datasets más utilizados en la documentación, se encuentra el *Kaggle Diabetic Retinopathy Detection competition* y *APTOS 2019 Blindness Detection*. Ambos repositorios contienen imágenes de RD, clasificadas por el nivel de gravedad en que se encuentra la enfermedad. En la Tabla 3.2 se observa la cantidad de imágenes de cada dataset y los principales autores que los utilizan. Ninguno de los dataset se encuentra balanceado, y el que proviene de Kaggle posee una mayor cantidad de imágenes, mientras que el de Aptos 2019 destaca por la calidad de las imágenes

Tabla 3.2

Datasets públicos

Nombre de dataset	Cantidad de Imágenes	Autores que lo usaron	¿Se encuentra balanceado?
<i>Kaggle Diabetic Retinopathy Detection competition (2015)</i>	88702	Gayathri (2020) Hossi (2021).	No
<i>APTOS 2019 Blindness Detection</i>	3662	Nazir(2021) Abini(2025) Alyoubi(2021) Hammad(2021) Samanta (2020)	No

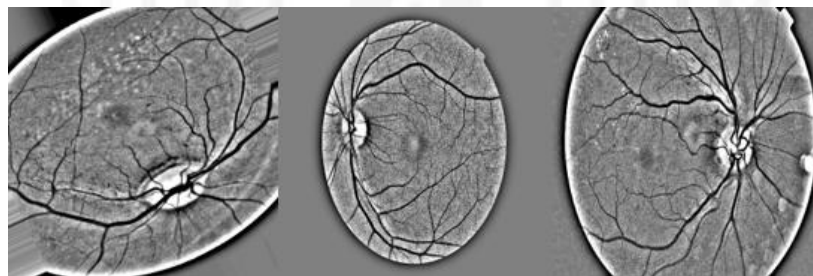
3.4 TÉCNICAS DE PREPROCESAMIENTO

Existen distintas técnicas de preprocesamiento que se realizan a las imágenes para que se pueda obtener un mejor resultado al momento de entrenar un modelo de Deep Learning. Entre las más utilizadas se encuentran las usadas por Samanta (2020) y Hemanth(2020), las cuales se listan a continuación:

- *Image Augmentation*: Es una técnica empleada por autores como Bhardwaj(2021), Qureshi(2021), Hammad(2021) entre otros. Image Augmentation consiste en la alteración de data para crear nueva y de este modo expandir el dataset. El método es empleado en la fase de entrenamiento para evitar el *overfitting* del modelo. En la Figura 3.2 se observa ejemplos de aumentaciones empleadas

Figura 3.2

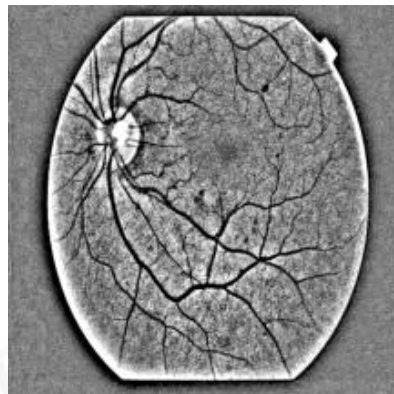
Imágenes aumentadas de fondo de ojo



- *Gaussian Blur*: Empleado por Samanta (2020), se emplea para crear una imagen más borrosa y reducir el detalle, de este modo se reduce el ruido para todas las imágenes. En la Figura 3.3 se emplea un filtro gaussiano, el cual es de paso-bajo que remueve los componentes de alta frecuencia de la imagen.

Figura 3.3

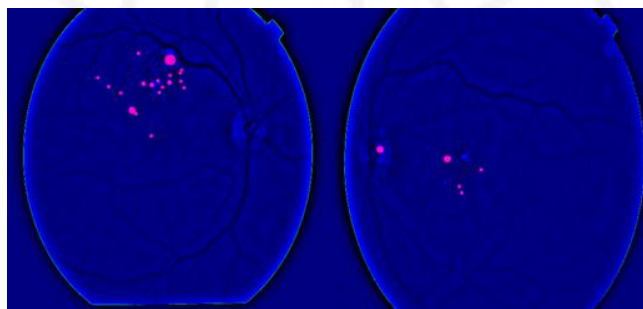
Imagen con Gaussian Blur aplicado



- Blob detection: Es una técnica cuyo propósito es detectar lugares más oscuros o claros en las imágenes. Kaspers (2011) declaró que los blobs aparecen de distintas formas dependiendo a su escala. Como se observa en la Figura 3.4 los blobs pueden ser detectados usando operaciones locales en una representación a múltiple escala de una imagen

Figura 3.4

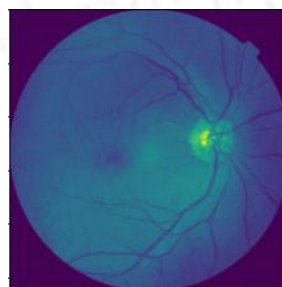
Resaltamiento de blobs



- Extracción del canal verde: Kumar (2018) usó el canal verde para mejorar el contraste de las estructuras vasculares y de las lesiones rojas características de la RD, como los microaneurismas y las hemorragias. En la Figura 3.5 se observa una mejor visualización de los vasos sanguíneos al ser también cavidades rojas.

Figura 3.5

Canal verde extraído de una imagen de fondo de ojo



Estas herramientas, modifican la imagen a procesar para que de tal manera sea más sencilla y precisa la extracción de características en las imágenes, esto último se realiza mediante una serie

de filtros que adquieran datos relevantes de las estructuras a observar, o mediante un modelo CNN que realice filtros en cada una de sus capas.

3.5 ARQUITECTURA DE REDES NEURONALES CONVOLUCIONALES

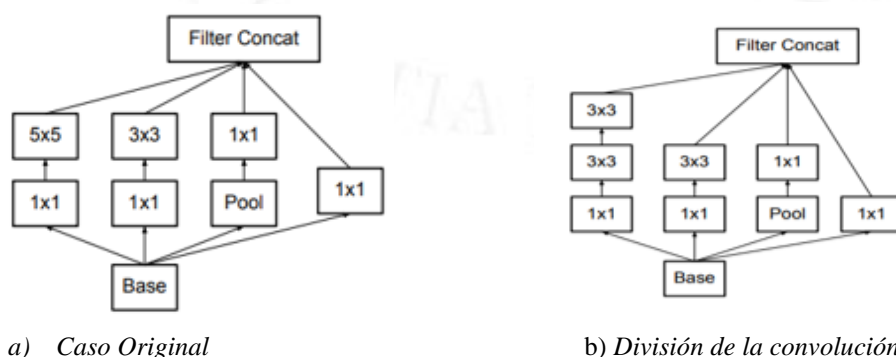
Shao (2014) definió el *Transfer Learning* como una metodología de visión computacional para crear modelos de manera más rápida y efectiva. En lugar de comenzar el entrenamiento desde cero, se parte de un modelo que ya fue entrenado en otra tarea, reutilizando sus parámetros. Esta metodología aprovecha arquitecturas preentrenadas para mejorar el rendimiento de la red. Cuando se desea cambiar el propósito de la red preentrenada, es necesario entrenar nuevamente el modelo, especialmente en las capas de clasificación, ya que los parámetros obtenidos previamente facilitan la extracción de características de la imagen. Abini (2025) mencionó que cambió la última capa de su modelo en el entrenamiento de su red Inception V3 para que pudiera aceptar predicciones de tres categorías, y a partir de esto calcular el rendimiento según algunas métricas. En esta sección se describen tres modelos ampliamente utilizados en aplicaciones médicas: Inception V3, empleada por Abini (2025), Hathwar (2019) y Kanimozhi (2020); DenseNet121, utilizada por Samanta (2020), Bhardwaj (2021) y Nazir (2021); y VGG16, aplicada por Pavithra (2024) y Hayati (2023).

3.5.1 INCEPTION V3

La red Inception V3 fue desarrollada por Christian Szegedy y su equipo en 2016. En su diseño, cada módulo “Inception” agrupó convoluciones paralelas de distintos tamaños (1×1 , 3×3 , 5×5) y una rama de pooling, cuyas salidas se concatenaron para capturar características tanto finas como amplias. En la Figura 3.6 se presenta una representación de la manera en que se utilizaron convoluciones de 1×1 para reducir la dimensión de los canales antes de aplicar filtros mayores, y se factorizó una 5×5 en dos 3×3 buscando menor coste computacional. En la Figura 3.7 se visualiza como la red inicia con un bloque inicial (“stem”), y posteriormente se ejecutó varios módulos Inception con múltiples ramas paralelas donde al final sus salidas eran concatenadas para dar a la salida del módulo. En estos módulos Inception se aplicaba la lógica de filtros previamente explicada. Al final de la extracción de características se aplicaba un Global Average Pooling para reducir la cantidad de parámetros y empezar con producir probabilidades de clase usando softmax. En la Figura 3.6 se visualiza la factorización de filtros y en la Figura 3.7 la arquitectura completa de InceptionV3.

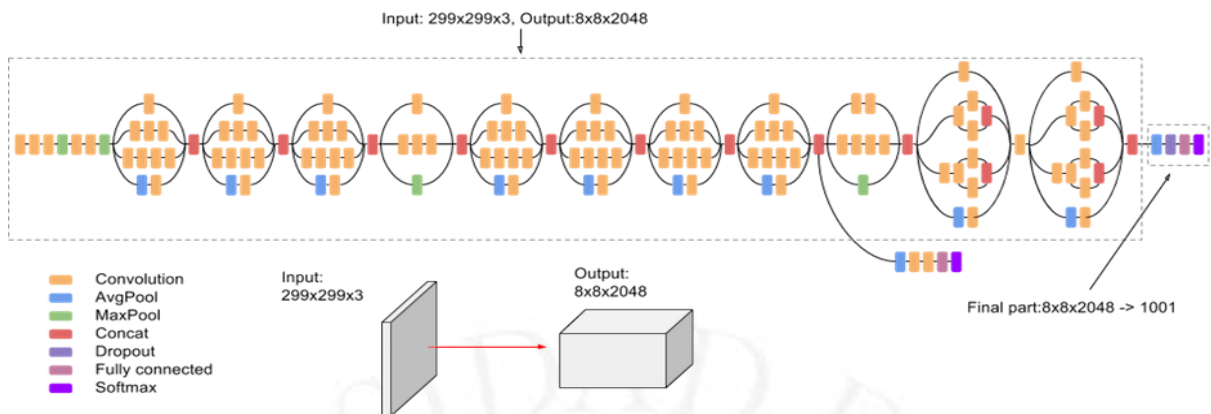
Figura 3.6

Modelo Original de Inception



Nota. Adaptado de Rethinking the inception architecture for computer vision, por C. Szegedy, 2016

Figura 3.7
Arquitectura completa de Inception

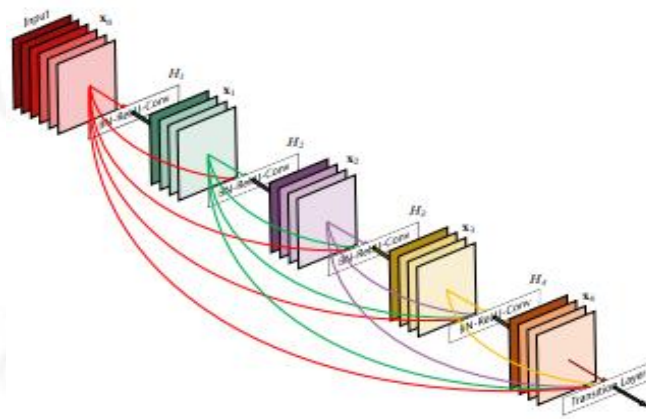


Nota. Adaptado de Rethinking the inception architecture for computer vision, por C. Szegedy, 2016

3.5.2 DENSENET121

La arquitectura DenseNet121 fue introducida por Huang (2017), consistió en una red convolucional densa, que se basa en conectar cada capa de la red, con cada capa posterior. Para cada capa de la red, los resultados de las capas previas son utilizados como inputs de la misma forma en que se muestra en la Figura 3.8.

Figura 3.8
DenseBlock de 5 capas



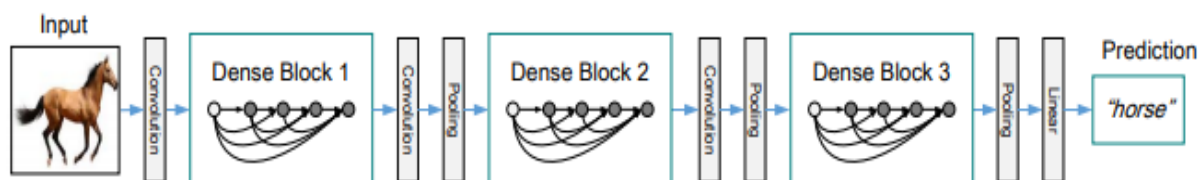
Nota. Adaptado de Densely connected convolutional networks, Huang, 2017

La Figura 3.9 es la representación de la arquitectura DenseNet121, la cual empieza con una convolución 7x7 seguida de un max pooling 3x3. Posterior a esto, viene una secuencia de cuatro bloques densos intercalados por capas de transición (una convolución junto a un average pool). En lo que respecta a los bloques densos, cada uno es compuesta por un grupo de capas densas constituidas por Batch Normalization, activación ReLU y una convolución 1x1 seguida de 3x3. Cada uno de los resultados de las capas densas terminarían siendo input para las siguientes hasta finalizar en la capa de transición.

Entre las ventajas de este modelo se encuentran fortalecer la propagación de características, fomentar la reutilización de características y reducir los parámetros empleados para el modelo,

además las conexiones densas tienen un efecto regularizador, esto permite reducir el *overfitting* con dataset pequeños. El modelo DenseNet se diferencia por la variación que existe entre la información que ingresa a las capas y las que se conservan. Las capas DenseNet son estrechas, y solo se agrega una pequeña cantidad de características a la red. La arquitectura de la red se presencia en la Figura 3.9, junto a las capas de transición (convolución y *pooling*) y la relación con los *denseblock*.

Figura 3.9
Arquitectura DenseNet121



Nota. Adaptado de *Densely connected convolutional networks*, Huang, 2017

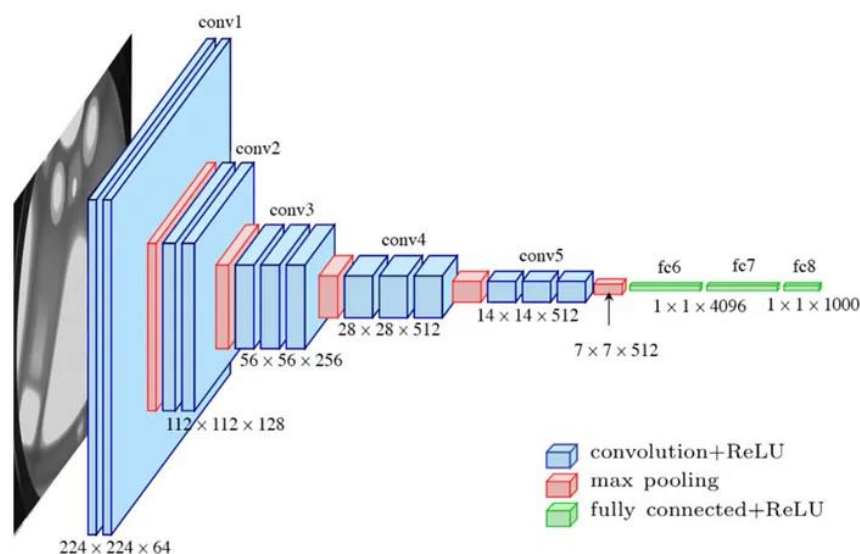
3.5.3 VGG16

El modelo fue creado por Simonyan (2014), el cual se basa en el uso de 16 capas con pesos entrenables utilizando filtros pequeños de 3x3 y capas de activación por cada filtro. Esto hace que las decisiones del modelo sean más discriminativas. Del mismo modo la ventaja que tiene este modelo es que al utilizar capas pequeñas, obtiene menos parámetros y permite el aprendizaje de representaciones más complejas que cuando se utilizan filtros grandes.

La arquitectura se basa en 5 bloques mostrados en la Figura 3.10, donde existen 13 capas convolucionales con filtros 3x3, funciones de activación Relu después de cada convolución, una capa de max pooling después de cada bloque y finalmente 3 capas fully connected seguidas que terminan obteniendo una gran cantidad de parámetros.

La principal desventaja de la arquitectura es que es más lenta que otros modelos debido a la cantidad enorme de parámetros del final, es por ello por lo que se usa menos en dispositivos de bajos recursos. Además, el modelo tiende a *overfitting* si se usa con datasets pequeños.

Figura 3.10
Arquitectura VGG16



Nota. Adaptado de *Very Deep Convolutional Networks for Large-Scale Image Recognition*, Simonyan, 2014

3.6 ENSAMBLAJE

El ensamblaje es el proceso de combinar múltiples algoritmos de aprendizaje para poder obtener un mejor desempeño en la obtención de resultados. En las investigaciones de Alyoubi (2021) y Karki (2021) se evidenció que el ensamblado de los modelos CNN mejoró significativamente la exactitud de las predicciones a comparación del obtenido de manera individual. Existen distintos tipos de algoritmos empleados para el ensamblaje, entre los más utilizados se encuentran los usados por Yazdizadeh (2019):

- Votación por promedio: Genera la predicción a partir del promedio de las probabilidades obtenidas por la clase softmax de todos los modelos.
- Votación por mayoría: Se selecciona la predicción que obtuvo más votos de los modelos creados.
- Pesos óptimos: Utilizar los pesos de los modelos que contengan un valor de pérdida más bajo, y realizar alguno de los métodos de ensamblado por votación

Existen trabajos donde se realiza un ensamblado de diversas CNNs para obtener una mucho más robusta y precisa en el momento de clasificar las imágenes, como es el caso de Karki (2021), que creó una red ensamblada de distintas versiones de EfficientNet para detectar RD, obteniendo un 2% de exactitud extra en comparación a las redes individuales.

3.7 MÉTRICAS DE DESEMPEÑO

El artículo de Fernández (2003) detalla las métricas utilizadas para evaluar la efectividad de un diagnóstico clínico, destacando entre ellas las siguientes:

- Sensibilidad: Representa el porcentaje de veces en que el modelo acierta en las clasificaciones de positivos, es decir representa que tan bien el modelo detecta esa clase.
- Especificidad: Es la medida que da el porcentaje de verdaderos negativos hay entre los negativos. Representa que tan bien se identifica una clase.
- Exactitud: Indica la cercanía que tiene el modelo para poder ser completamente verídico en sus predicciones. Es la cantidad de predicciones positivas que fueron aciertos.

$$\text{Sensibilidad} = \frac{VP}{VP + FN} \quad (1)$$

$$\text{Especificidad} = \frac{VN}{FP + VN} \quad (2)$$

$$\text{Exactitud} = \frac{VP + VN}{VP + FP + FN + VN} \quad (3)$$

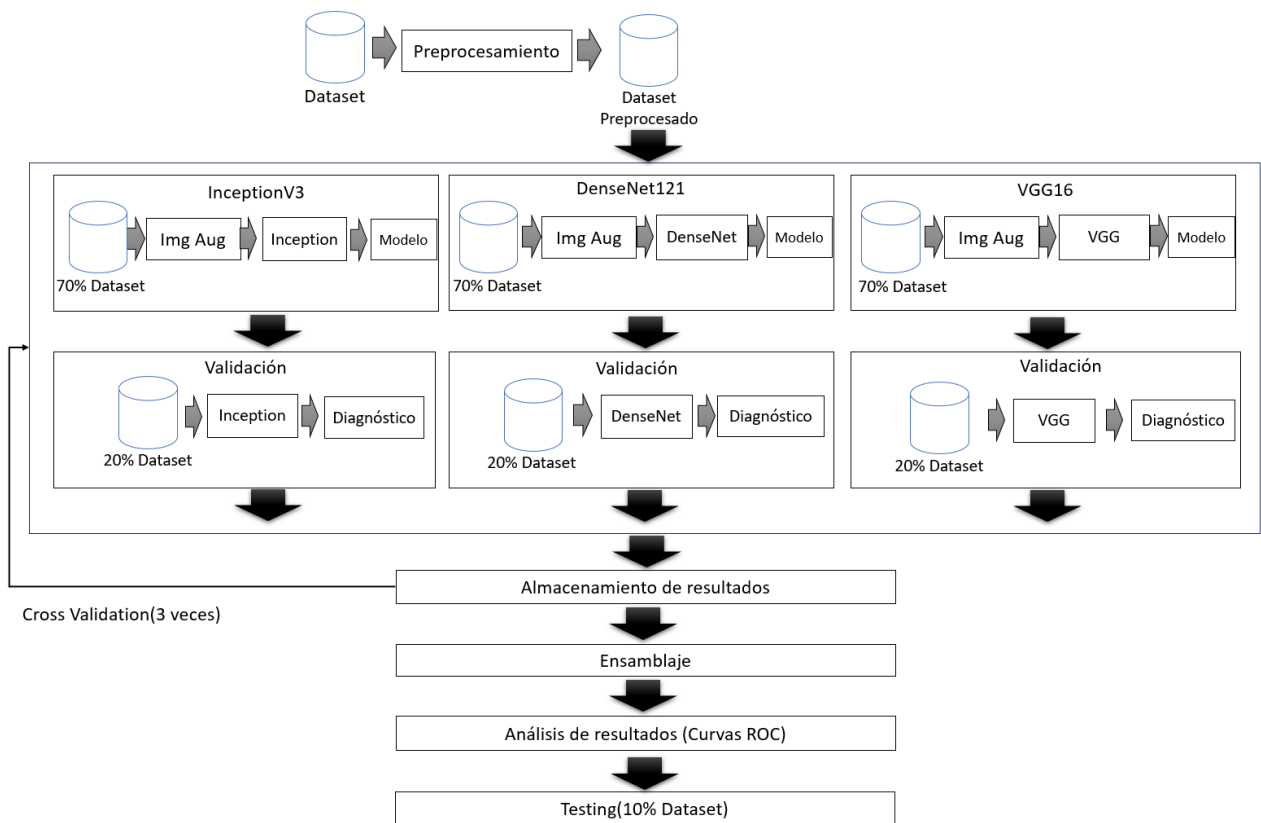
4. METODOLOGÍA

La metodología propuesta usó el dataset “*APTOS 2019 Blindness Detection*” publicado en Kaggle(2019), se desarrolló y comparó el desempeño de los modelos VGG16, DenseNet121 y InceptionV3 de manera individual y al ensamblarse. Para evaluar cada modelo, se utilizó cross validation con 3 iteraciones para cada uno, del mismo modo se realizó lo mismo para cada posible combinación de modelos para el ensamblado. Posteriormente se crearon las curvas ROC correspondientes, junto al cálculo del AUC y del umbral más óptimo por promedio. Después de la comparación de los resultados, se determinó que el ensamble de los modelos DenseNet121, InceptionV3 y VGG16 proporciona los mejores resultados.

Para realizar la creación del modelo ensamblado y del cross validation se emplearon 4 etapas presentadas más a detalle en la Figura 4.1: obtención del dataset, el preprocesamiento, selección de la mejor combinación, evaluación. Para la primera etapa que fue la obtención del dataset implicaba la descarga de las imágenes y del documento .csv con la información necesaria. El dataset se agrupó en tres grados, no RD, temprano y avanzado. En el preprocesamiento se extrajo y se trabajó con el canal verde de la imagen, ya que este canal permite resaltar las lesiones en la retina. Seguidamente se usó CLAHE con la finalidad de mejorar la calidad de la imagen y de este modo poder emplear *blob detection* para resaltar los candidatos a microaneurismas de las imágenes.

Previo a realizar el *blob detection* se ha tenido que separar los vasos sanguíneos de los posibles candidatos, ya que puede ser un falso positivo en la detección, por lo que se empleó el filtro Frangi que ayuda con la detección de estructuras tubulares. Con los microaneurismas resaltados, se realizó el cambio de espacio de color y el desenfoque gaussiano para destacar los demás síntomas de la RD. La siguiente fase consistió en la selección del mejor modelo ensamblado, lo cual se basa en grandes rasgos en el entrenamiento y las combinaciones entre los modelos InceptionV3 VGG16 y DenseNet121. Esta fase de selección nos proporcionó 7 modelos funcionales con capacidad de realizar diagnósticos, los cuales fueron evaluados mediante el AUC y exactitud promedio de cada uno. La cuarta fase fue la validación del modelo creado. Se empleó parte del dataset para evaluar el desempeño del modelo al momento de realizar diagnósticos. Se realizó con imágenes ya preprocesadas anteriormente. La última fase presentada se basa en la obtención y análisis de resultados de la evaluación del modelo ensamblado seleccionado, donde se obtuvo la exactitud del modelo y se realizaron gráficos que evidencian su progreso en cada época.

Figura 4.1
Metodología Propuesta



4.1 DATASET

El dataset que se utilizó es “*APTOS 2019 Blindness Detection*” publicado en Kaggle (2019). Este repositorio, contiene las 3662 imágenes de RD graduadas del 0 al 4 y con imágenes de distintos tamaños. Los atributos que se tiene son dos: nombre de la imagen y nivel. Las imágenes fueron proporcionadas por “*Aravind Eye Hospital*”, institución ubicada en India, donde sus doctores evaluaron y diagnosticaron el nivel de RD en cada una de las imágenes del repositorio. El dominio del nivel de gravedad de la RD en cada imagen se especifica en la Tabla 4.1. El Ministerio de Salud del Perú (2016) declara al nivel 0 como que no hay anomalías, nivel 1 cuando solo hay microaneurismas, nivel 2 cuando hay microaneurismas y hemorragias o exudados, nivel 3 cuando hay hemorragias intrarretinianas, rosarios venosos definidos, anomalías microvasculares y finalmente nivel 4 cuando hay neovascularización o hemorragia vítrea.

Tabla 4.1
Histograma de imágenes del dataset

Nivel	Descripción	Cantidad
0	Inexistencia de Retinopatía Diabética	1805
1	Retinopatía Diabética Leve	370
2	Retinopatía Diabética Media	999
3	Retinopatía Diabética Grave	193
4	Retinopatía Diabética Proliferativa	295

El Ministerio de Salud del Perú (2016) declara que la retinopatía tiene un nivel superior al 2 debe ser referido el paciente a un nivel II o III de atención, por este motivo se utilizó solo 3 grados, ver Tabla 4.2. Para el dataset se realizó una combinación de clases de tal forma que las clases sean No RD, RD temprano (leve) y RD avanzado (Media, Grave y Proliferativa). Se empleará la abreviación N para No RD, T para RD temprana y finalmente, A para RD avanzado. En la Tabla 4.2 se presenta como terminaría el dataset con lo presentado.

Tabla 4.2
Nuevos grados de retinopatía diabética

Tipo Retinopatía	Cantidad de Imágenes
Inexistencia de Retinopatía Diabética	1805
Retinopatía Diabética Temprana	370
Retinopatía Diabética Avanzada	1487

4.2 PREPROCESAMIENTO

Una vez balanceada las imágenes, es necesario el preprocesamiento de las imágenes para que el modelo pueda predecir de la mejor manera la anomalía. Por lo cual, para la primera parte del preprocesamiento, se separó el canal verde de la imagen, del mismo modo que Tavakoli (2020), ya que este canal permite resaltar distintas lesiones rojas que pueda poseer la retina como vendrían ser los vasos sanguíneos y hemorragias. Seguidamente se realizó CLAHE para mejorar el contraste de la imagen obtenida. Finalizado estos dos pasos, se empleó Blob detection junto a la extracción de vasos sanguíneos para resaltar los candidatos a microaneurismas y no confundirlos con los vasos.

La segunda parte del preprocesamiento está basada en las técnicas realizadas por Ben Graham, el ganador de la competencia de detección de RD de Kaggle por California Healthcare Foundation (2015). Lo realizado por este competidor, fue cambiar el tamaño de las imágenes a

512x512 para que se puedan ingresar correctamente a modelos de Deep learning. Finalmente se aplica Gaussian Blur y reducir la alta frecuencia de algunas partes de la imagen, haciéndolas más borrosas, este método también es empleado por Lands (2020) y Samanta (2020), donde este último nos indica que los valores empleados para los parámetros SigmaX y SigmaY es de 10.

Para que ambos preprocesamientos no sean afectados mutuamente, se mantuvo el resultado del Gaussian Blur en un canal de la imagen, mientras que las posiciones del blob detection se realizaron en otro aparte para que no se superponga la información. Un detalle más amplio acerca del proceso de preprocesamiento se encuentra en la sección 5.1

4.3 IMAGE AUGMENTATION

En base a lo presentado por Hammad (2021) y Bhardwaj (2021), utilizaron data augmentation para generar nuevas imágenes a partir de las ya existentes y evitar el *overfitting* del modelo, de este modo los resultados podrán ser considerados más generales. Se creó una imagen nueva para cada imagen que exista en la partición de entrenamiento, Se empleará 70% del dataset original. Las modificaciones fueron las siguientes: Rotación de 15 grados, *horizontal flip*. De esta forma el dataset tendrá imágenes adicionales con distintas variaciones, pero absolutamente todas deben seguir el mismo preprocesamiento. Se debe resaltar que únicamente se utilizó la augmentación para las imágenes de entrenamiento y no para todo el dataset.

No se usó el image augmentation como oversampling, ya que generar imágenes nuevas de retinopatía diabética a partir de operaciones morfológicas comunes, solo haría que el modelo entrene y valide con imágenes muy parecidas entre sí, casuística que es muy poco común en imágenes médicas. Por este motivo se optó por trabajar con el dataset desbalanceado

4.4 TRANSFER LEARNING

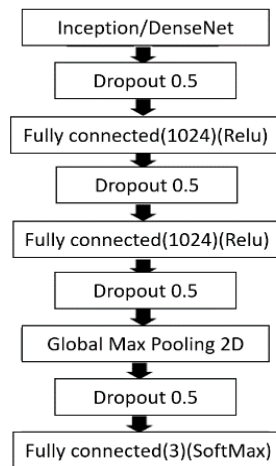
Existen muchos modelos pre-entrenados que son utilizados como arquitectura para obtener un mejor desempeño y rendimiento para el modelo. En base a las investigaciones de Hathwar (2019), Zeng (2019), Bhardwaj (2021) y Hammad (2021) se observa que la red de Inception V3 provee buenos resultados a la hora de graduar y detectar RD. Del mismo modo Nazir (2021), Lands (2020) y Samanta (2021) que emplearon el modelo DenseNet121 o a Hayati (2023) y Pavithra (2024) que usaron VGG, obtuvieron buenos resultados al usar sus redes pre-entrenadas.

Estos modelos poseen una distinta cantidad de capas, DenseNet121 posee 121 capas, InceptionV3 posee 48 capas y finalmente VGG16 posee solo 16 capas entrenadas. Para obtener unos parámetros propios para la detección de retinopatía diabética se descongelarán las últimas capas de cada modelo proporcionalmente a la cantidad que poseen en total. Esto permitirá que no solo se entrene únicamente la clasificación de imágenes. Se descongeló un 50% de las capas para el caso de InceptionV3 y DenseNet121, pero para VGG16 al presentar malos resultados con ese porcentaje se limitó a descongelar un 25%

Para la clasificación de las imágenes se utilizó como base el modelo de Samanta (2020) que detalla qué capas se agregaron para este propósito. Se agregó dos *Fully Connected Layers* con función de activación ReLu y dos *dropouts* intermedios con valor de 0.5 para evitar *overfitting*. Finalmente, un *global max pooling* para reducir la dimensionalidad del output y una *Fully Connected Layer* pero con función de clasificador, por lo que su función de activación es softmax, ya que existen más de una categoría en la clasificación. En la Figura 42 se muestran las capas agregadas.

Figura 4.2

Diagrama de capas agregadas



Se emplearon las imágenes preprocesadas para el entrenamiento y generación de pesos de estos modelos, además posteriormente se empleará el optimizador el algoritmo Adam ya que este fue elegido por Lands (2020) y Zeng (2020), debido a su poco costo computacional al entrenar y su efectividad en modelos de *Deep learning* con grandes dataset y como algoritmo para calcular el *loss* se usó *categorical cross-entropy*. El optimizador Adam utiliza un *learning rate* por defecto de 10^{-3} , el cual para el entrenamiento será utilizado como hiperparámetro inicial. Para el entrenamiento se propone que de no mejorar el porcentaje de validación después de 10 épocas se bajará el *learning rate* secuencialmente usando ReduceOnPlateau. En la Tabla 4.3 se muestran los hiperparámetros utilizados.

Tabla 4.3*Hiperparámetros empleados*

Hiperparámetro	Valor
Tamaño del Lote	8
Número de Épocas	100
Optimizador	Adam
Learning Rate	10^{-3} a 10^{-7}
Descongelamiento de Capas	DenseNet121: 60 capas InceptionV3: 24 capas VGG16: 3 capas

Con los parámetros y modelos agregados previamente, se inició el entrenamiento con las imágenes preprocesadas dando resultado a un modelo que ya pueda clasificar las imágenes que se les ingrese.

4.5 MODELO ENSAMBLADO

Para el ensamblaje, se tomará en cuenta la investigación de Jinfeng(2020) y Hammad(2021), los cuales utilizaron *ensemble avarage* para combinar sus modelos. El proceso consiste en que una vez obtenidos los resultados de cada uno del modelo DenseNet, Inception y VGG16, se suma ambos resultados, de tal forma que solo se mantenga un arreglo con las respuestas del modelo. Seguidamente, se mantendrá como valor correcto a la clase que contenga el mayor promedio, de este modo la seguridad en la predicción que mantiene cada modelo será tomada en cuenta y se reflejará en los resultados finales.

4.6 CROSS VALIDATION

Los modelos fueron entrenados 3 veces distintas con diferentes grupos del dataset dividido en 70%-20%, utilizando el mismo procedimiento de preprocesamiento, entrenamiento y ensamblaje que se utilizará para el modelo final. Debido a que se tiene 3 modelos candidatos para la predicción, se eligió emplear esta técnica para seleccionar al más efectivo para la detección temprana de retinopatía diabética, siendo cada combinación de modelos como hiperparámetro. Se realizó diagramas ROC, junto al cálculo del AUC y del umbral de clasificación óptimo para poder elegir mediante un método cuantitativo al mejor candidato para la tarea. Este análisis se realizó tanto a las versiones individuales de los modelos, como a cada combinación de ensamblaje.

4.7 EVALUACIÓN DE MODELOS

Una vez culminado el cross validation se procedió a analizar los resultados de las curvas ROC y de los umbrales óptimos conseguidos para saber cuál sería la mejor combinación de modelos posible a usar. Posterior a esto se procedió a realizar un entrenamiento adicional con el modelo óptimo y realizar la evaluación.

Se consiguieron los pesos del modelo creado en el paso anterior mediante un *checkpoint* en la época que obtuvo mejor rendimiento en la validación. Posteriormente se realizó la evaluación utilizando el 10% del dataset que no fue usado para el entrenamiento o validación. Las imágenes deben pasar previamente por el mismo preprocesamiento que el de la fase de entrenamiento. Estas imágenes fueron ingresadas a los modelos ya creados y entrenados para que posteriormente se pueda evaluar los diagnósticos otorgados. Cuando el modelo acabó de predecir, se obtuvieron todos los resultados a evaluar, los criterios de éxito de estos serán entrenar un modelo que no tienda a *overfitting*, y la obtención de resultados distintivos al usar el preprocesamiento planteado. Para el análisis de resultados se tomó en cuenta la exactitud en la validación y evaluación junto al puntaje obtenido en la función de pérdida. Se compararon los resultados de ambos modelos, y se revisó el desempeño de cada uno.

5 EXPERIMENTACIÓN

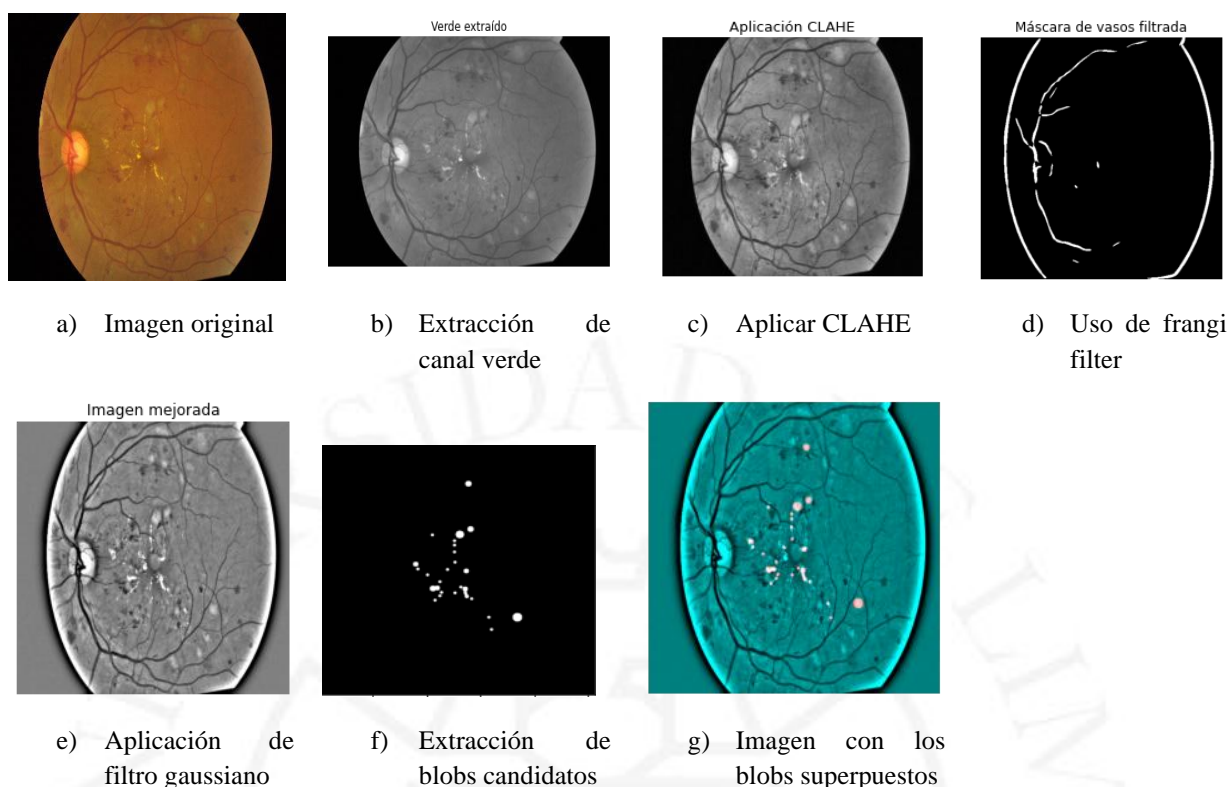
Durante la fase de experimentación se identificaron tres etapas fundamentales que fueron determinantes para alcanzar los resultados esperados: el preprocesamiento de las imágenes, el análisis de los resultados preliminares previos a la etapa de *blob detection* y la obtención de los umbrales óptimos.

5.1 PREPROCESAMIENTO DE IMÁGENES

El primer paso del preprocesamiento se evidencia en la Figura 5.1.b, que consiste en la extracción del canal verde y utilizarlo debido al realce que le da a los vasos sanguíneos y lesiones rojas de la retina. Posteriormente se empleó CLAHE evidenciado en la Figura 5.1.c, para mejorar la calidad de la imagen, pero de una manera cauta empleando un clipLimit de 2.0 para no generar mucho ruido. En la Figura 5.1.d se muestra el uso del filtro de vasos “Frangi vesselness filter” con escala de (1,2,3,4) para incluir vasos sanguíneos de distintos tamaños, de este modo se obtiene el margen de los vasos sanguíneos en la imagen que se usarán para excluirlas como candidatos en el blob detection. Se usó blob_log en la Figura 5.1.f, obviando los lugares ubicados previamente de los vasos sanguíneos. Se empleó un min_sigma de 1 con un max_sigma de 5 para detectar microaneurismas tanto grandes como pequeños con un umbral de 0.2. El siguiente paso fue emplear el suavizado gaussiano, usando una desviación estándar de 10, valor empleado por Lands (2020), el resultado se muestra en la Figura 5.1.e. Finalmente, el resultado de los blobs y del suavizado gaussiano se agregan en distintos canales de la imagen para que no se pierda la información de ninguno. En la Figura 5.1.g se visualiza la imagen final del preprocesamiento.

Figura 5.1

Proceso de preprocesamiento

**5.2 RESULTADOS PREVIOS AL BLOB DETECCION**

Se realizó una prueba al entrenar al modelo Densenet121 con 80% del dataset y validarlo con el 20% de las imágenes sin emplear *blob detection*. Se visualiza una diferenciación entre los resultados de la Tabla 5.1 y la Tabla 5.2, mostrando que hay una reducción de exactitud de 2% entre ambas casuísticas. Del mismo modo se evidencia que el f1-score que más incremento fue en la detección de RD temprana y no RD, siendo casi perfecta esta última reduciendo la posibilidad de falsos positivos a la inexistencia de RD. Se observa que, si bien mejoró la detección temprana, sigue habiendo complejidad en diferenciar una grave merecedora de un nivel más especializado de atención que de una temprana. Esto puede ocurrir debido a que existe una menor cantidad de imágenes de RD temprana y a que los microaneurismas son cavidades que aparecen en ambas clases solo que con una diferencia de cantidad.

Tabla 5.1Resultados sin usar *Blob Detection*

Clase	Precision	Recall	F1-score
No RD	96%	97%	0.96
Temprana	57%	57%	0.57
Avanzado	89%	87%	0.88
Accuracy			88%
Macro avg	81%	81%	0.81
Weighted avg	89%	89%	0.88

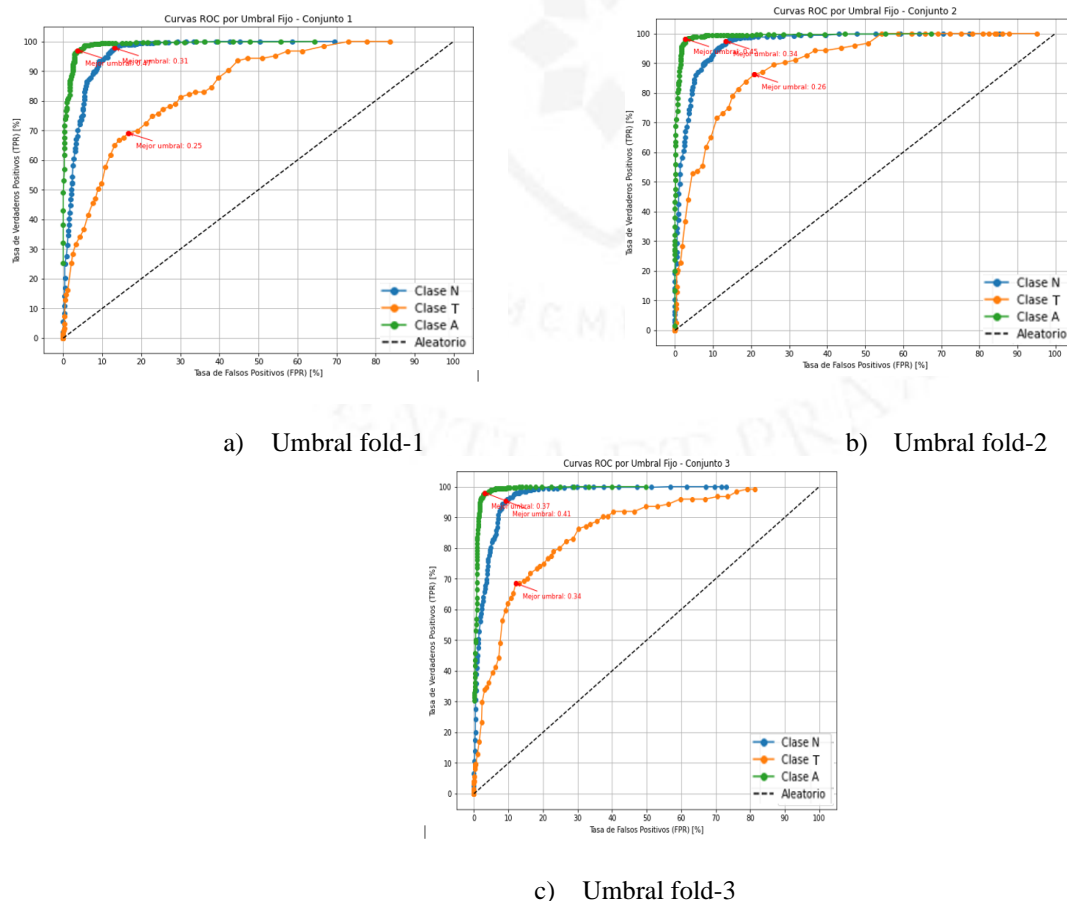
Tabla 5.2
Resultados al utilizar *Blob Detection*

Clase	Precision	Recall	F1-score
No RD	98%	99%	0.99
Temprana	57%	67%	0.62
Avanzado	90%	84%	0.87
Accuracy			90%
Macro avg	82%	83%	82%
Weighted avg	90%	89%	90%

5.3 OBTENCIÓN DEL UMBRAL ÓPTIMO

Para obtener el umbral óptimo, se calcularon las tasas de verdaderos positivos (TPR) y falsos positivos (FPR) para valores de umbral entre 0.001 y 1. El umbral óptimo se definió con el índice de Youden (1950) que consiste en la fórmula ($J=TPR-FPR$), que representa un equilibrio entre la sensibilidad y especificidad. Se realizó una curva ROC para cada clase variando el umbral de decisión. En cada curva se marcó el punto que maximiza el índice de Youden. Se obtuvieron 3 umbrales por cada fold del cross validation, siendo cada uno perteneciente a cada clase. Finalmente, estos umbrales se promediaron respectivamente para obtener su puntaje. Los resultados antes de promediar cada *fold* usados para InceptionV3 se ve en la Figura 5.2.a, donde se obtuvo un umbral óptimo en el primer *fold* de 0.47, 0.31 y 0.25 para la clase avanzada, no RD y temprana respectivamente en la Figura 5.2.b; 0.45, 0.34 y 0.26 para el segundo; finalmente 0.37, 0.41 y 0.34 para el tercero en la Figura 5.2.c.

Figura 5.2
Resultados umbrales óptimos InceptionV3



6 RESULTADOS

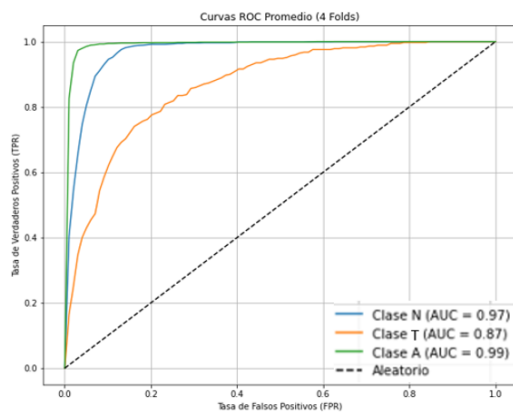
6.1 RESULTADOS DEL CROSS VALIDATION

Se utilizó un k-fold de 3 usando el 70% del dataset de entrenamiento y 20% de validación utilizando muestras aleatorias en cada iteración. La cantidad de 3 unidades es debido a la cantidad de recursos computacionales que se requería para la prueba y a que la varianza entre cada fold que se presentó era mínima y poco diferenciadora. Posteriormente se generó una curva ROC trazando la tasa de verdaderos positivos frente a la tasa de falsos positivos para distintos umbrales. Estos distintos puntos generaron la curva ROC por cada clase de RD que existe. El área bajo la curva corresponde al AUC, que es evidencia de la capacidad de discriminación del modelo independientemente del umbral.

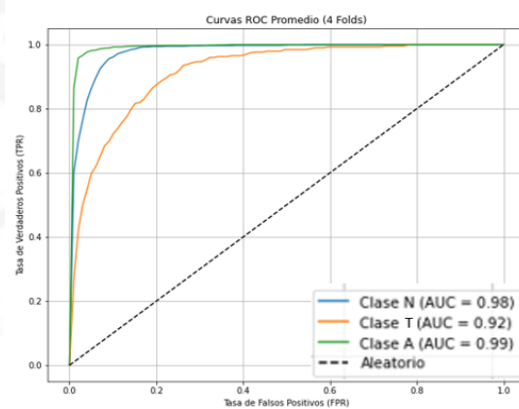
A partir de los resultados obtenidos en el Cross Validation de los modelos individuales, se concluye con la Tabla 6.1 que el modelo que posee una afinidad más balanceada para las 3 clases es DenseNet121, seguidamente VGG16 y finalmente InceptionV3. DenseNet121 es el que tiene un umbral más alto para la clase temprana de RD, mientras que por otro lado VGG16 es el que mantiene mejores umbrales en Avanzado y No RD. En lo que respecta el AUC individual de validación, en la Figura 6.1 se concluye que VGG16 es el que presenta uno superior, seguido de DenseNet121 y InceptionV3, pero poseyendo los 3 valores parecidos. Por estas semejanzas, se hizo Cross Validation con las combinaciones de ensamblado. En la Figura 6.2 se muestran los diagramas ROC de validación para los ensamblados, con el puntaje AUC junto a la matriz de confusión de cada modelo ensamblado. En la Tabla 6.1 se tiene evidencia de los mejores umbrales obtenidos para cada uno y en la Tabla 6.3 de las exactitudes respectivas a cada ensamblado.

Figura 6.1

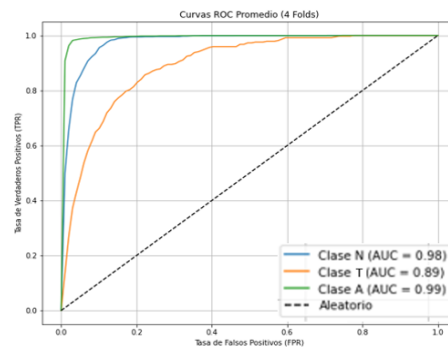
Resultados de Diagrama ROC de modelos



d) Diagrama ROC InceptionV3



e) Diagrama ROC VGG16



f) Diagrama ROC Densenet121

Tabla 6.1*Mejor umbral para cada modelo*

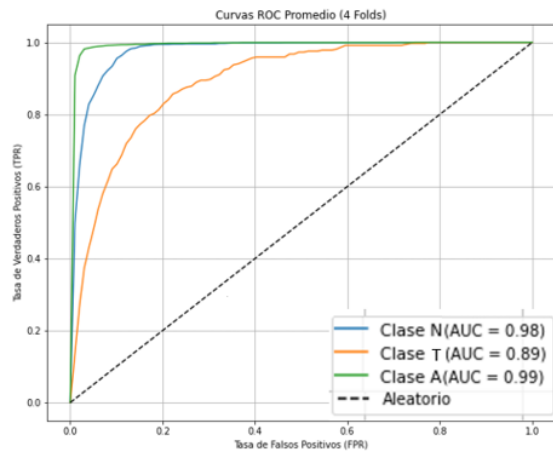
Mejor Umbral	InceptionV3	DenseNet121	VGG16
Clase A	0.44	0.43	0.43
Clase N	0.36	0.34	0.48
Clase T	0.26	0.30	0.12

Los resultados obtenidos para encontrar el mejor umbral en la Tabla 6.2 nos permiten maximizar la diferencia entre la tasa de verdaderos positivos y falsos positivos, este enfoque es ampliamente adoptando en aplicaciones médicas con problemas desbalanceados Zago (2020) ya que permite ajustar el modelo a necesidades clínicas específicas, los valores reportados indican la probabilidad mínima que una clase debe superar para ser considerada positiva. Por ejemplo, el umbral de 0.44 para la clase 'A' en InceptionV3 implica que cualquier predicción del modelo con probabilidad ≥ 0.44 para dicha clase se considera positiva.

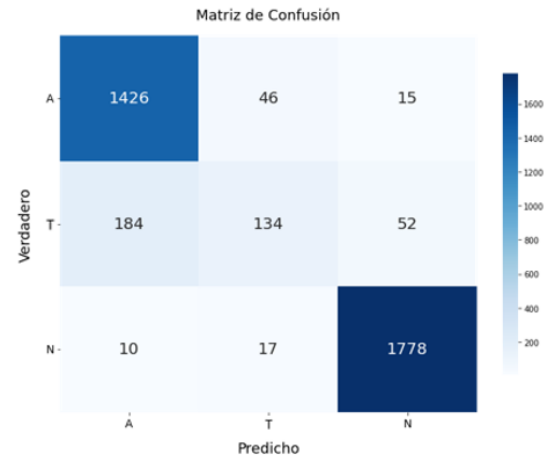
De la Figura 6.2.c y Figura 6.2.g se puede concluir que las mejores opciones para el ensamblado se encuentran entre la combinación de DenseNet121, InceptionV3 y VGG16 junto a DenseNet121 con VGG16 ya que ambas combinaciones poseen un AUC de 0.98 para N, de 0.93 para T y de 1 para A. Las matrices de confusión de la Figura 6.2.d y Figura 6.2.h evidencia errores en diferenciar entre la diferenciación entre temprano y avanzado en las predicciones de validación, pero una gran diferenciación entre Temprano y No RD. Los resultados presentados en la Tabla 6.4 demuestra los puntajes f1 para cada modelo ensamblado, demostrando que el ensamblado de DenseNet121 y VGG16, junto al ensamblado de los 3 modelos se encuentran entre los mejores candidatos según el puntaje *macro F1*, y *weighted F1* diferenciándose por milésimas. En la Tabla 6.4 se obtuvo las exactitudes de cada fold y así mismo la desviación estándar de estos, obteniendo del mismo modo que las combinaciones de VGG con Densenet121, junto a la combinación de los 3 modelos juntos son las mejores opciones a nivel de exactitudes. Así también en la Tabla 6.2 se observa que los umbrales del ensamblaje de los 3 modelos para la clase avanzada y temprana es superior a cualquier otra combinación. A partir de los resultados, se procedió a elegir el modelo ensamblado constituido por InceptionV3, Densenet121 y VGG16. La decisión se encontrará fundamentada en la sección de Discusión.

Figura 6.2

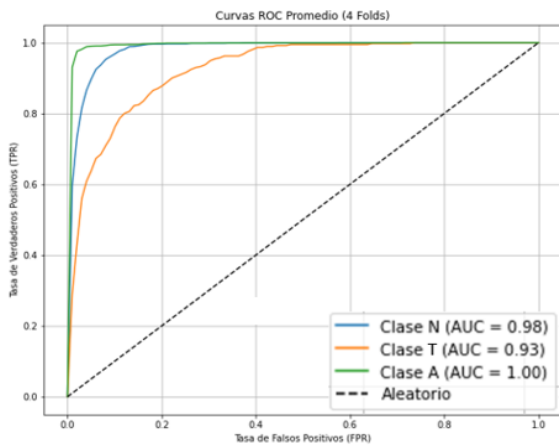
Resultados de Diagrama ROC y de confusión de modelos ensamblados



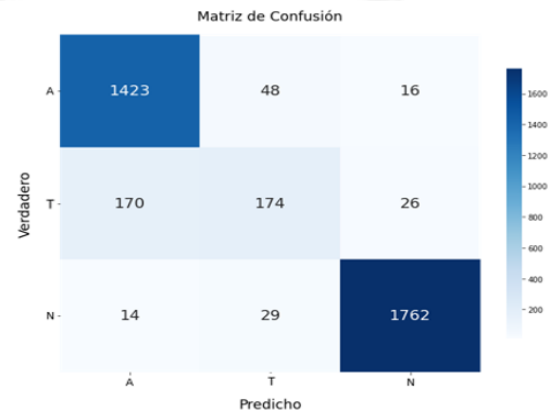
a) Diagrama ROC DenseNet+Inception



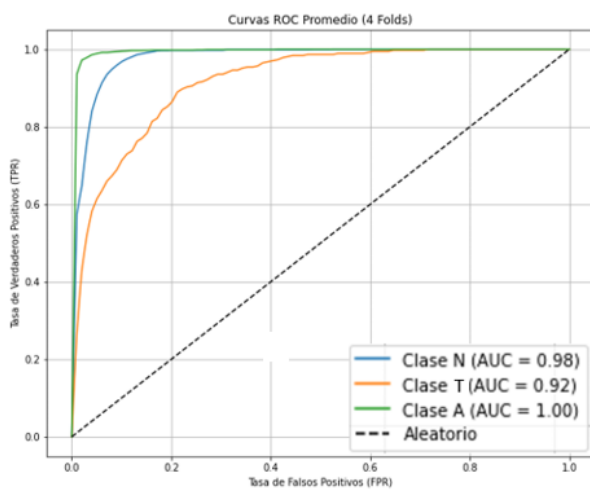
b) Diagrama de Confusión DenseNet+Inception



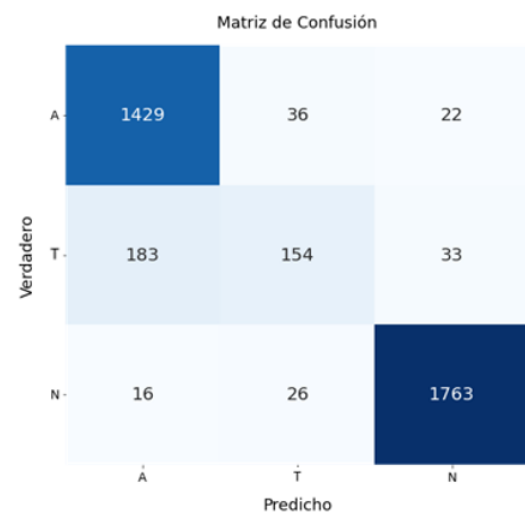
c) Diagrama ROC DenseNet + VGG



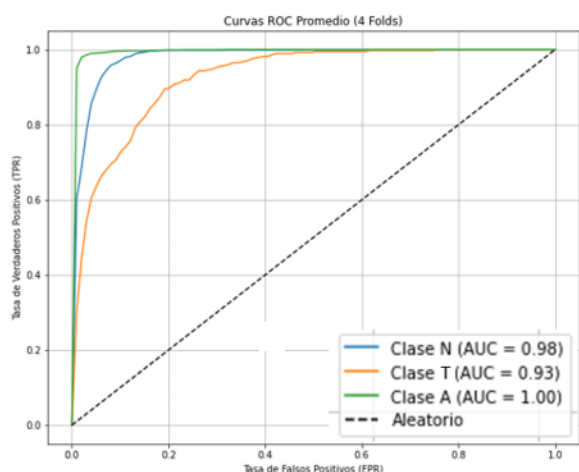
d) Diagrama de Confusión DenseNet + VGG



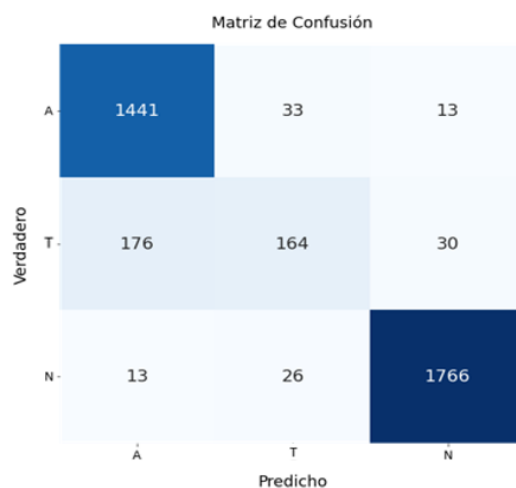
e) Diagrama ROC Inception+VGG



f) Diagrama de Confusión Inception+VGG



g) Diagrama ROC DenseNet+Inception+VGG



h) Diagrama de Confusión DenseNet + Inception +VGG

Tabla 6.2

Mejor umbral para cada modelo ensamblado

Mejor Umbral	DenseNet121+ InceptionV3	DenseNet121+ VGG16	VGG16+ InceptionV3	DenseNet121+ VGG16+InceptionV3
Clase A	0.43	0.36	0.48	0.42
Clase N	0.30	0.46	0.43	0.47
Clase T	0.34	0.24	0.17	0.22

La Tabla 6.2 presenta los umbrales óptimos para cada clase en cada combinación de modelos ensamblados, estos puntos de decisión permiten al modelo adaptarse al desbalance del dataset y priorizar la detección temprana, que es uno de los objetivos de este estudio. El cálculo de umbrales es una práctica estándar en aplicaciones de diagnósticos médico con datos desbalanceados (Zago, 2020).

Tabla 6.3

Exactitud promedio para cada modelo ensamblado

	DenseNet121+ InceptionV3	DenseNet121+ VGG16	VGG16+ InceptionV3	DenseNet121+ VGG16+InceptionV3
Exactitud	0.911	0.917	0.913	0.9205

Tabla 6.4

F1-Score para cada modelo ensamblado

	DenseNet121+ InceptionV3	DenseNet121+ VGG16	VGG16+ InceptionV3	DenseNet121+ VGG16+InceptionV3
Macro F1	0.9007	0.9129	0.9052	0.9130
Weighted F1	0.7882	0.8188	0.8054	0.8182

Tabla 6.5*Exactitud para cada fold junto a desviación estándar*

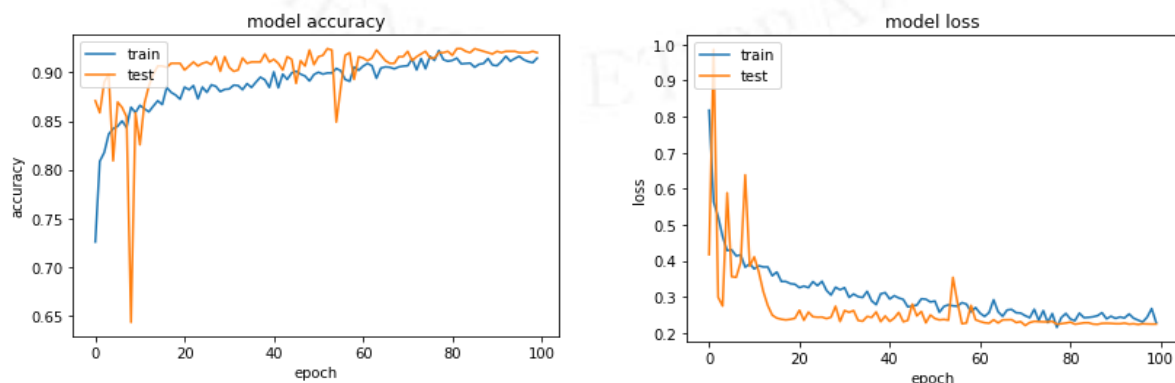
Modelo	Accuracy Fold	Accuracy Acumulado	Varianza (%)	Desv. Estándar (%)
Fold 1				
VGG	91.48%	91.48%	0.000000%	0.000000%
Inception	89.11%	89.11%	0.000000%	0.000000%
DenseNet	90.66%	90.66%	0.000000%	0.000000%
VGG + Inception	91.32%	91.32%	0.000000%	0.000000%
VGG + DenseNet	91.56%	91.56%	0.000000%	0.000000%
Inception+DenseNet	90.66%	90.66%	0.000000%	0.000000%
VGG+Inc+DenseNet	91.81%	91.81%	0.000000%	0.000000%
Fold 2				
VGG	91.81%	91.65%	0.000268%	0.16%
Inception	90.09%	89.60%	0.002415%	0.49%
DenseNet	90.75%	91.44%	0.000017%	0.05%
VGG + Inception	91.56%	91.93%	0.000151%	0.12%
VGG + DenseNet	92.30%	91.93%	0.001358%	0.37%
Inception+DenseNet	92.06%	91.36%	0.004946%	0.69%
VGG+Inc+DenseNet	92.63%	92.22%	0.001677%	0.41%
Fold 3				
VGG	90.16%	91.15%	0.005061%	0.71%
Inception	90.33%	89.84%	0.002791%	0.53%
DenseNet	89.84%	90.41%	0.001687%	0.41%
VGG + Inception	91.23%	91.37%	0.000200%	0.14%
VGG + DenseNet	91.31%	91.73%	0.0001764%	0.41%
Inception+DenseNet	90.74%	91.15%	0.0004090%	0.64%
VGG+Inc+DenseNet	91.72%	92.05%	0.001669%	0.41%

6.2 RESULTADOS DEL MODELO VGG16

Los resultados que se obtuvieron por el modelo VGG16 son los referentes a la exactitud en el entrenamiento y en la validación en cada época. El modelo de exactitud de la Figura 6.3.a muestra una estabilización en el entrenamiento antes de llegar a 92%, por otro lado, la mayor exactitud obtenida en la validación es de 92%. La fluctuación en la validación refleja el comportamiento del modelo frente a datos no vistos y probablemente ocurre debido a la poca cantidad de imágenes de entrenamiento para las imágenes tempranas.

Figura 6.3

Gráficos de Entrenamiento y Validación para Vgg16



a) Diagrama de exactitud

b) Diagrama de pérdida

En el reporte estadístico se obtuvieron los resultados mostrados en la Tabla 6.6 al momento de ingresar las imágenes de evaluación al modelo. Se destaca la exactitud de 91.12%. En el reporte se muestra una exactitud aceptable para la clasificación de grados de RD, teniendo una buena precisión y sensibilidad para los grados de No RD y Avanzado.

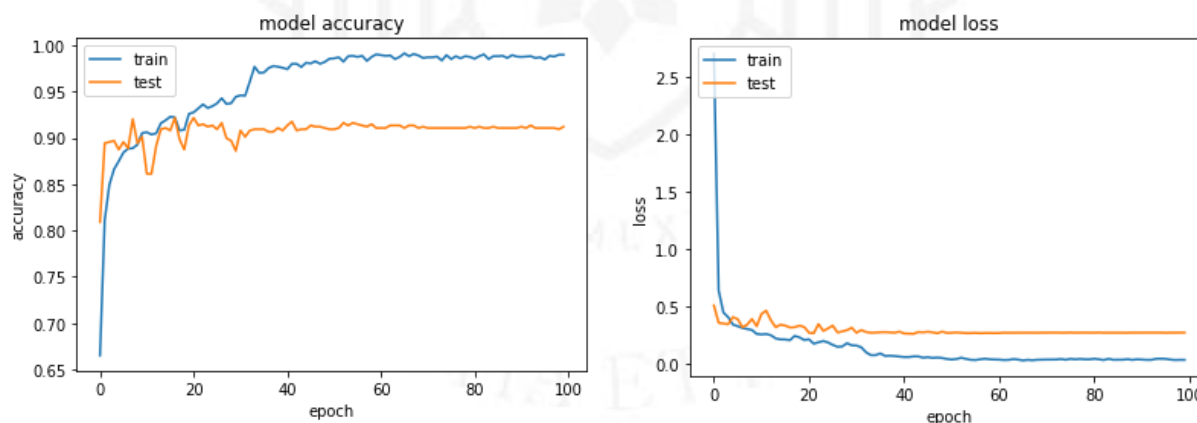
Tabla 6.6
Reporte de Clasificación VGG16 de Evaluación

Clase	Precision	Recall	F1-score
No RD	96%	97%	0.97
Temprano	62%	54%	0.58
Avanzado	91%	93%	0.92
Accuracy			91.12%
Macro avg	83%	82%	0.82
Weighted avg	91%	91%	0.91

6.3 RESULTADOS DEL MODELO INCEPTIONV3

Los resultados de la exactitud de InceptionV3 se presenta en la Figura 6.4.a, donde se observa que los valores obtenidos en el entrenamiento y evaluación son bastantes estables desde antes de llegar a la época 100, por lo que el proceso de entrenamiento del modelo surtió de una manera satisfactoria. El modelo obtuvo una exactitud en la validación de 91.25%

Figura 6.4
Gráficos de Entrenamiento y Validación para InceptionV3



a) Diagrama de exactitud

b) Diagrama de pérdida

En el reporte estadístico se obtuvieron los resultados mostrados en la Tabla 6.7 al momento de ingresar las imágenes de evaluación al modelo mostrando una exactitud de 89.65%. En el reporte se muestra que InceptionV3 posee más dificultades en la predicción de los 3 grados que VGG16, pero teniendo de igual forma buenos resultados en No RD y avanzado.

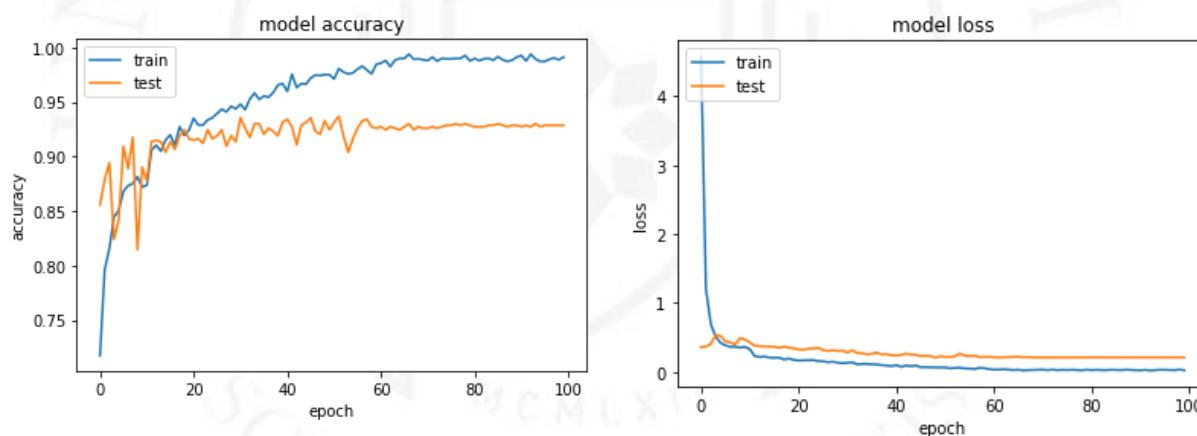
Tabla 6.7
Reporte de Clasificación InceptionV3 de Evaluación

Clase	Precision	Recall	F1-score
No RD	97%	97%	0.97
Temprano	52%	43%	0.47
Avanzado	88%	93%	0.90
Accuracy			89.65%
Macro avg	79%	78%	0.78
Weighted avg	89%	90%	0.89

6.4 RESULTADOS DEL MODELO DENSENET121

Los resultados de la exactitud de Densenet121 se observa en la Figura 6.5.a, donde se puede presenciar que los valores obtenidos en el entrenamiento y evaluación son bastantes estables desde antes de llegar a la época 100, por lo que el proceso de entrenamiento del modelo surtió de una manera satisfactoria. El modelo obtuvo una exactitud en la validación de 92.8%

Figura 6.5
Gráficos de Entrenamiento y Validación para DenseNet121



a) Diagrama de exactitud

b) Diagrama de pérdida

En el reporte estadístico se obtuvieron los resultados mostrados en la Tabla 6.8 al momento de ingresar las imágenes de evaluación al modelo, obteniendo una exactitud de 89.64%. En el reporte se muestra que los resultados son bastante parecidos a los de InceptionV3, solo que este contiene un mayor F1-score en la clase temprana.

Tabla 6.8
Reporte de Clasificación DenseNet121 de Evaluación

Clase	Precision	Recall	F1-score
No RD	97%	97%	0.97
Temprano	50%	57%	0.53
Avanzado	92%	89%	0.90
Accuracy			89.64%
Macro avg	80%	81%	0.80
Weighted avg	90%	90%	0.90

6.5 RESULTADOS DEL MODELO ENSAMBLADO POR PROMEDIO

Una vez entrenados y evaluados los tres modelos previos, proceden a ser ensamblados por promedio. El resultado de esta combinación de resultados se ven en la Tabla 6.9 donde se obtuvo el mayor porcentaje entre las demás combinaciones, con una exactitud de 91.65% en la evaluación, junto a unas sensibilidades bastante elevadas para los casos de RD Avanzada y de inexistencia de RD. En la Figura 6.6 se observa que un gran porcentaje de las fallas del modelo creado se producen al momento de diferenciar la RD Temprana de la RD Avanzada, por otro lado, se identifica de manera casi perfecta los casos de No RD y de RD avanzada. Del mismo modo, la baja cantidad de imágenes tempranas es bastante contraproducente para la evaluación. Estos resultados serán vistos a mayor detalle en la siguiente sección.

Figura 6.6
Matriz de Confusión de Modelo Ensamblado por Promedio en la Evaluación

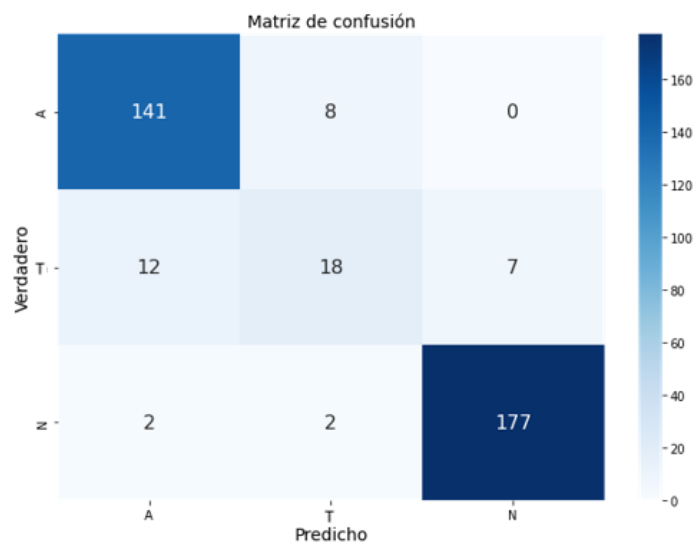


Tabla 6.9

Reporte de Clasificación de Modelo Ensamblado por Promedio

Clase	Precision	Recall	F1-score
No RD	96%	98%	0.97
Temprano	64%	49%	0.55
Avanzado	91%	95%	0.93
Accuracy			91.65%
Macro avg	84%	89%	0.82
Weighted avg	91%	92%	0.91

7 DISCUSIÓN

En la experimentación se obtuvieron diversos hallazgos sobre el aporte del Blob Detection en la investigación y finalmente el aumento de exactitud obtenido al ensamblar los modelos DenseNet121, VGG16 y InceptionV3.

La exactitud obtenida es producida gracias al uso del ensamblaje y del Blob Detection empleado en las imágenes. Comparando los resultados de la Tabla 5.1 y Tabla 5.2 se infiere que la distinción de los posibles microaneurismas produjo una diferenciación entre las imágenes de RD temprana y las retinas sanas provocando un aumento de precisión y sensibilidad de 2% para el grado de no RD, lastimosamente para el caso de la RD Temprana solo aumentó en 10% la sensibilidad, pero al mismo tiempo reduciendo la del grado avanzado en 3%. Esta casuística se debe a que los microaneurismas se presentan tanto en imágenes tempranas como en avanzadas, por lo que es una característica en común que ambos tienen y podría causar falsos positivos de diagnósticos avanzados. Acerca de los resultados individuales por modelo, si bien VGG16 ha obtenido una exactitud menor en la validación a DenseNet121, VGG16 superó a todos los modelos en la evaluación, demostrando que admite de una mejor manera datos externos. Así también fue el modelo que mantuvo un F1-score más alto para la detección de las 3 clases en la evaluación.

Se ha utilizado Cross Validation 3 veces para cada modelo y para cada posibilidad de ensamblado para verificar qué modelos son los más efectivos para la detección temprana de retinopatía diabética. El Cross Validation de la Figura 6.1 demostró que VGG16 presentaba el mayor AUC para los tres estados, demostrando los resultados de evaluación analizados previamente. Por otro lado, el AUC de InceptionV3 es bastante parecido al de DenseNet12 con diferencia en la etapa de No RD y Temprana, mientras que para la etapa avanzada comparten el mismo valor. Los resultados obtenidos en los modelos ensamblados, al revisar el AUC de la Figura 6.2.c, Figura 6.2.g y Figura 6.2.e, se puede tener como opción a la combinación de DenseNet121+VGG16, y a InceptionV3+VGG16+DenseNet121 ya que poseen los mismos valores de AUC para los 3 grados de retinopatía. Debido a las semejanzas en el puntaje, se obtuvo el promedio de exactitud en la Tabla 6.3, quedando como mejor opción InceptionV3+VGG16+DenseNet121. Del mismo modo, el ensamblado de los 3 modelos obtuvo la mayor exactitud en los resultados de los 3 folds presentados en la Tabla 6.5.

Los resultados obtenidos en la investigación mostraron que la exactitud del modelo VGG16 es superior a la de los demás modelos, pero a pesar de eso, en la Tabla 6.5 y en los cálculos de AUC correspondientes se determinó que el ensamblaje de los 3 modelos es la mejor opción. Los resultados del modelo ensamblado fueron de un 91.65%, mejorando en un 0.5% la exactitud con respecto a VGG16, pero mejorando en 2% la precisión para el estado temprano y en 2% la sensibilidad para no RD y Avanzado. Esta mejora refleja que los 3 modelos al ensamblarse por

promedio se compenentran correctamente compartiendo los beneficios de cada modelo. Esta mejora, se asemeja mucho a los resultados dados por Jinfeng (2020) o Alyoubi(2021) ya que estos obtuvieron un incremento aproximado de entre 0.5% a 3% de exactitud al ensamblar dos o más modelos CNN. Si bien la exactitud no tiene un cambio tan drástico, si fomenta una mejoría de 2% a 3% en precisión y sensibilidad por clase.

Entre las limitaciones obtenidas para el desarrollo de la metodología, se encontraría la baja cantidad de imágenes que posee el dataset de RD temprana. Esta limitación afectó considerablemente en la predicción y distinción con otras clases. El oversampling no resultó ser una solución óptima a esta limitación debido a que no se está creando imágenes nuevas que presenten un distinto tipo de lesiones. Adicionalmente, existen nuevas tecnologías para la rama de Computer Vision, como los Visual Transformer. Desafortunadamente, el uso de este tipo de modelos requiere un gran uso de requisitos computacionales para el entrenamiento, por lo que se descartó dicha opción por las limitantes técnicas de la investigación.

A partir de los resultados y la discusión presentada se rescatar del modelo propuesto ciertos puntos: Principalmente se demuestra la contribución de un preprocesamiento centrado en detectar y resaltar los síntomas de la retinopatía diabética. El *blob detection* ayudó a aumentar la precisión y sensibilidad del estado temprano, así también como reducir casi totalmente los falsos positivos para el diagnóstico de un ojo sano, permitiendo asegurar la detección de la retinopatía diabética. Usar otro tipo de operaciones morfológicas en las imágenes para detectar con mayor facilidad otro tipo de cavidades podría ser buen complemento con la realizada en esta investigación.

En la Figura 6.3, Figura 6.4 y Figura 6.5 se ve el desempeño de los modelos al momento de entrenar y validar. Se empleó 100 épocas para ver el comportamiento completo. El modelo que obtuvo una mayor exactitud en la validación fue DenseNet121 con 92.8% de la Figura 6.5, pero el modelo con un mejor comportamiento durante el entrenamiento según la gráfica es de la Figura 6.3 por parte de VGG16. Existe fluctuaciones momentáneas al momento del entrenamiento para todos los modelos, pero se encuentra dentro de lo normal, ya que posteriormente se estabilizan los resultados.

La metodología empleada en la investigación es una combinación entre las metodologías de detección antiguas basada en el preprocesamiento y realce de características, junto a las empleadas junto a CNN. En lo que respecta a la modificación de imágenes, es una combinación de la segmentación de vasos sanguíneos propuestos por Tavakoli (2020) previo a la detección de microaneurismas, junto a el uso de blob detection por Dharmana (2020). La selección del mejor modelo es distintiva a otras investigaciones como la de Shmarat (2024) o Pavithra (2024) donde únicamente se compara la exactitud de cada modelo con la misma división de información. A diferencia de Yan (2026) y Shamrat (2024) , se evitó emplear el data augmentation con métodos convencionales para generar más información como debido a que produce una inflación de resultados correctos debido a que generar imágenes de esa manera, solo incrementa de manera artificial la exactitud del modelo ya que las lesiones y características de la RD seguirán estando en posiciones semejantes a pesar de la augmentación, por lo que validar o realizar una evaluación con esas imágenes es contraproducente.

Del mismo modo, con las capas agregadas para InceptionV3, DenseNet121 y VGG16, junto a los hiperparámetros utilizados se pudo obtener dos modelos con bastante precisión y sensibilidad para la clase Avanzada y sin RD. El modelo de VGG16 obtuvo precisión de 91% para RD avanzada, 96% para no RD y 62% para RD temprano; para el caso de las sensibilidades es de 93% para avanzada, 97% para inexistencia y 54% para temprano. El modelo de InceptionV3 obtuvo precisión de 88% para RD avanzada, 97% para no RD y 52% para RD temprano; para el caso de las sensibilidades es de 93% para avanzada, 97% para inexistencia y 43% para temprano. Finalmente, DenseNet121 obtuvo ha obtenido una precisión de 92% para RD avanzada, 97% para

no RD y 50% para RD temprano; para el caso de las sensibilidades es de 89% para avanzada, 97% para inexistencia y 57% para temprano. Los 3 modelos cuentan con bastante dificultad para predecir exactamente las tempranas debido a la baja cantidad de muestras y que los microaneurismas existen en cualquier etapa de la RD, solo que en distintas cantidades.

Se ha realizado una elección fundamentada para los modelos óptimos a usar en lugar de solo basarse en la exactitud de una sola agrupación. Usar cross validation permitió obtener los AUC respectivos de cada modelo y elegir el más eficiente para la investigación y adicionalmente se demostró que el ensamblaje de modelos con alta exactitud generalmente siempre permitirá mejores resultados, siendo la combinación de VGG16, DenseNet121 y InceptionV3 la mejor alternativa.

8 CONCLUSIONES

La investigación realizada aportaría al Perú una metodología para crear un modelo que permitiría un prediagnóstico eficaz de la retinopatía diabética en su etapa grave, debido a que la sensibilidad de este grado es de 95%. Una detección tan precisa de este estado permitirá que los usuarios tengan la certeza de necesitar una atención médica de alto nivel lo antes posible. Se desarrolló un modelo ensamblado de la red InceptionV3, DenseNet121 y VGG16 que otorgó una exactitud de 91.65%, este resultado demuestra que el ensamblaje de ambos modelos pre-entrenados junto a los hiperparámetros utilizados permitió crear un sistema para detectar la RD, dando apertura a distintas modificaciones del reporte para aumentar el desempeño de este.

La cantidad de 3 k-folds empleada para la investigación se definió debido a la poca varianza y desviación estándar que se obtenía de cada fold, ya que el hiperparámetro a evaluar más importante era únicamente el desempeño de cada modelo, por lo que el aumento de poder computacional para crear más folds era innecesario. El uso de Cross Validation permitió elegir los modelos y la combinación de modelos más eficaces para la detección de retinopatía diabética. Se demostró que la mejor arquitectura para esta tarea es Densenet121, y posteriormente VGG16. Se hizo la comparación de AUC entre los distintos ensamblajes y la combinación de VGG16+InceptionV3+DenseNet121 tiene 1 de AUC para RD avanzado, 0.98 de no RD y 0.93 de RD temprana. El puntaje que DenseNet121+VGG16 obtuvo fue equivalente a la combinación de los 3 modelos, pero en lo que respecta a la exactitud obtenida hay una diferenciación de 1%, además que el umbral óptimo de InceptionV3+VGG16+DenseNet121 es ligeramente mejor por 0.02 para las clases Avanzada y temprana. Por este motivo, para el preprocesamiento que se le está dando a las imágenes, la mejor combinación de modelos a usar es InceptionV3 y VGG16+DenseNet121.

El aumento producido por el ensamblaje de los dos modelos pre-entrenados reafirma la efectividad del uso del ensamblaje por promedio para aumentar la exactitud. Finalmente, a diferencia de otras investigaciones, se apostó en el uso de *Blob Detection* para resaltar los microaneurismas que aparecen en los distintos grados de la RD. Gracias a esta técnica, se obtuvo un mejor resultado en el modelo, permitiendo un aumento en la exactitud de RD en los distintos grados usados en la investigación. Esto nos ayuda a concluir que resaltar los microaneurismas de esta manera, permite diferenciar los casos de una RD temprano con una retina sana.

Para futuras investigaciones, se podría ampliar el uso de operaciones morfológicas para resaltar otras características de la RD, como podrían ser exudados o *cotton wools*. El uso del *blob detection* demostró buenos resultados detectando microaneurismas, emplear métodos enfocados específicamente a cavidades que aparecen en un grado avanzado mejoraría el resultado obtenido en esta investigación. Por otra parte, la baja cantidad de imágenes de RD en grado temprano obtenidas del dataset APTOS 2019 produjo problemas en la investigación realizada. Se recomienda la creación de un dataset amplio y balanceado o una expansión agregando imágenes

de la misma calidad provenientes de otro dataset de RD. Se recomienda también el uso de otras herramientas para el oversampling que no consista en la augmentación de imágenes, se podría probar la generación de imágenes con redes adversarias (GANs) y documentar el desempeño.

REFERENCIAS

- Kanimozhi, J., Vasuki, P., & Roomi, S. M. (2020). Fundus image lesion detection algorithm for diabetic retinopathy screening. *Journal of Ambient Intelligence and Humanized Computing*, 7407-7416.
- Abini, M., & Sathya, P. (2025). Detection and Classification of Diabetic Retinopathy Using Modified Inception V3. *International Journal Bioautomation*, 77-92.
- Adal, K. M., Sidibé, D., Ali, S., Chaum, E., Karnowski, T. P., & Mériaudeau, F. (2014). Automated detection of microaneurysms using scale-adapted blob analysis and semi-supervised learning. *Computer Methods and Programs in Biomedicine Volume 114*, 1-10.
- Akgül, İ. (2023). Deep Learning Based Models for Detection of Diabetic Retinopathy. *Tehnički Glasnik*, 581-587.
- Alyoubi, W. L., Abulkhair, M. F., & Shalash, W. M. (2021). Diabetic Retinopathy Fundus Image Classification and Lesions Localization System Using Deep Learning. *Sensors*, 21(11), 3704.
- Bhardwaj, C., Jain, S., & Sood, M. (2021). Diabetic retinopathy severity grading employing quadrantbased Inception-V3 convolution neural network. *International Journal of Imaging Systems and Technology*, 592-608.
- Bühlmann, P. (2012). Bagging, boosting and ensemble methods. En *Handbook of Computational Statistics* (págs. 985-1022). Springer Berlin Heidelberg.
- Carrera, E. V., González, A., & Carrera, R. (2017). Automated detection of diabetic retinopathy using SVM. *2017 IEEE XXIV international conference on electronics, electrical engineering and computing (INTERCON)* (págs. 1-4). Cusco: IEEE.
- Dharmana, M. M., & S., A. M. (2020). Pre-diagnosis of diabetic retinopathy using blob detection. *2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA)*, 98-101.
- Gangwar, A. K., & Ravi, V. (2021). Diabetic Retinopathy Detection Using Transfer Learning and Deep Learning. *Evolution in Computational Intelligence Volume 1176*, 679-689.
- Gayathri, S., Gopi, V. P., & Palanisamy, P. (2020). A lightweight CNN for Diabetic Retinopathy classification from fundus images. *Biomedical Signal Processing and Control*, 62, 102115.
- Ghosh, K., Sharma, S., & Choudhury, T. (2024). *Determining the Severity of Diabetic Retinopathy through Neural Network Models. In Advancement of Data Processing Methods for Artificial and Computing Intelligence*. River Publishers.
- Grupo Gamma.(s.f.). (08 de 2013). *¿Qué es la retinopatía diabética?* Obtenido de Grupo Gamma: <https://grupogamma.com/que-es-la-retinopatia-diabetica/>.
- Hammad, M., Al-Smadi, M., Baker, Q. B., & Al-Zboon, S. A. (2021). A transfer learning with deep neural network approach for diabetic retinopathy classification. *International Journal of Electrical and Computer Engineering*, 3492-3501.

- Hathwar, S. B., & Srinivasa, G. (2019). Automated Grading of Diabetic Retinopathy in Retinal Fundus Images using Deep Learning. *2019 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, 73-77.
- Hemanth, D. J., Deperlioglu, O., & Kose, U. (2020). An enhanced diabetic retinopathy detection and classification approach using deep convolutional neural network. *Neural Computing and Applications*, 32, 707-721.
- Hossi, A. E., Skouta, A., Elmoufidi, A., & Nachaoui, M. (2021). Applied CNN for Automatic Diabetic Retinopathy Assessment Using Fundus Images. *usiness Intelligence: 6th International Conference, CBI 2021* (págs. 425-433). Beni Mellal: Springer International Publishing.
- Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely Connected Convolutional Networks. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4700-4708.
- Jinfeng, G., Qummar, S., Junming, Z., Ruxian, Y., & Khan, F. G. (2020). Ensemble Framework of Deep CNNs for Diabetic Retinopathy Detection. *Computational Intelligence and Neuroscience*, 1-11.
- Karki, S. S., & Kulkarni, P. (2021). Diabetic Retinopathy Classification using a Combination of EfficientNets. *2021 International Conference on Emerging Smart Computing and Informatics (ESCI)* (págs. 68-72). Pune: IEEE.
- Kaspers, A. (2011). *Blob Detection*. UMC Utrecht.
- Kumar, S., & Kumar, B. (2018). Diabetic retinopathy detection by extracting area and number of microaneurysm from colour fundus image. *2018 5th International Conference on Signal Processing and Integrated Networks (SPIN)* (págs. 359-364). Noida: IEEE.
- Lands, A., Kottarathil, A. J., Biju, A., Jacob, E. M., & Thomas, S. (2020). Implementation of deep learning based algorithms for diabetic retinopathy classification from fundus images. *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)* (págs. 1028-1032). Tirunelveli: IEEE.
- MINSA. (2016). *Guía de práctica clínica para el diagnóstico, tratamiento y control de la diabetes mellitus tipo 2 en el primer nivel de atención*. Perú: Ministerio de Salud del Perú.
- Molina, R., & Rodríguez, C. (2012). Definición, clasificación y diagnóstico de la diabetes mellitus. *Revista Venezolana de Endocrinología y Metabolismo*, 7-12.
- Nazir, T., Nawaz, M., Rashid, J., Mahum, R., Masood, M., Mehmood, A., . . . Hussain, A. (2021). Detection of Diabetic Eye Disease from Retinal Images Using a Deep Learning Based CenterNet Model. *Sensors*, 21(16), 5283.
- Patni, K., Yagnik, S., & Patel, P. (2025). Exploring Dataset Variability in Diabetic Retinopathy Classification Using Transfer Learning Approaches. *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, 763-777.
- Pavithra, S., Jaladi, D., & Tamilarasi, K. (2024). Optical imaging for diabetic retinopathy diagnosis and detection using ensemble models. *Photodiagnosis and Photodynamic Therapy*.
- Qureshi, I., Ma, J., & Abbas, Q. (2021). Diabetic retinopathy detection and stage classification in eye fundus images using active deep learning. *Multimedia Tools and Applications*, 80, 11691-11721.

- Samanta, A., Saha, A., Satapathy, S. C., Fernandes, S. L., & Zhang, Y.-D. (2020). Automated detection of diabetic retinopathy using convolutional neural networks on a small dataset. *Pattern Recognition Letters*, 135, 293-298.
- Santana, J. C., & Licoa, J. K. (2023). *Comorbilidades asociadas a la diabetes mellitus tipo II: causas, consecuencias y prevalencia en adultos mayores*. Universidad Estatal del Sur de Manabí.
- Shamrat, J., Shakil, R., Hoque, N., Akter, B., & Ahmed, K. (2024). An advanced deep neural network for fundus image analysis and enhancing diabetic retinopathy detection. *Healthcare Analytics*.
- Shao, L., Zhu, F., & Li, X. (2014). Transfer Learning for Visual Categorization: A Survey. *IEEE transactions on neural networks and learning systems*, 26(5), 1019-1034.
- Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv*.
- Society, A. P.-O. (2019). *APTOS 2019 Blindness Detection*. Obtenido de Kaggle: <https://www.kaggle.com/c/aptos2019-blindness-detection/overview>.
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. *Proceedings of the IEEE conference on computer vision and pattern recognition* (págs. 2818-2826). IEEE.
- Tavakoli, M., Nazar, M., & Mehdizadeh, A. (2020). The efficacy of microaneurysms detection with and without vessel segmentation in color retinal images. *Medical Imaging 2020: Computer-Aided Diagnosis* (págs. 935-944). Houston: SPIE.
- WHO, W. (14 de 11 de 2024). *Implementation of a diabetic retinopathy referral network*. Obtenido de <https://www.who.int/news-room/fact-sheets/detail/diabetes>.
- Yan, S., Feng, M., & Cai, Y. (2026). An integrated deep learning model with enhanced EfficientNetB0 and MobileNetV1 for diabetic retinopathy grading. *Biomedical Signal Processing and Control*.
- Yao, X., Pei, X., Yang, Y., Zhang, H., Xia, M., Huang, R., . . . Li, Z. (2021). Distribution of diabetic retinopathy in diabetes mellitus patients and its association rules with other eye diseases. *Scientific Reports*.
- Yazdizadeh, A., Patterson, Z., & Farooq, B. (2019). Ensemble Convolutional Neural Networks for Mode Inference in Smartphone Travel Survey. *IEEE Transactions on Intelligent Transportation Systems*, 2232-2239.
- Youden, W. J. (1950). Index for rating diagnostic tests. *Cancer*, 3(1), 32-35.
- Yu, S., Xiao, D., & Kanagasigam, Y. (2017). Exudate detection for diabetic retinopathy with convolutional neural networks. *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (págs. 1744-1747). Jeju: IEEE.
- Zago, G. T., Andreão, R. V., Dorizzi, B., Ottoni, E., & Salles, T. (2020). Diabetic retinopathy detection using red lesion localization and convolutional neural networks. *Computers in biology and medicine*, 116, 103537.
- Zeng, X., Chen, H., Luo, Y., & Ye, W. (2019). Automated Diabetic Retinopathy Detection Based on Binocular Siamese-Like Convolutional Neural Network. *IEEE access*, 7, 30744-30753.






4% Similitud general

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para ca...

Filtrado desde el informe

- ▶ Bibliografía
- ▶ Texto citado

Fuentes principales

- 4%  Fuentes de Internet
- 1%  Publicaciones
- 0%  Trabajos entregados (trabajos del estudiante)

Marcas de integridad

N.º de alertas de integridad para revisión

No se han detectado manipulaciones de texto sospechosas.

Los algoritmos de nuestro sistema analizan un documento en profundidad para buscar inconsistencias que permitirían distinguirlo de una entrega normal. Si advertimos algo extraño, lo marcamos como una alerta para que pueda revisarlo.

Una marca de alerta no es necesariamente un indicador de problemas. Sin embargo, recomendamos que preste atención y la revise.

