UNIVERSIDAD NACIONAL DE MOQUEGUA

ESCUELA PROFESIONAL DE INGENIERIA DE SISTEMAS E INFORMÁTICA

DIAGNÓSTICO DE LA RETINOPATÍA DIABÉTICA USANDO ALGORITMOS DE CLASIFICACIÓN

TESIS

PRESENTADO POR:

ANAMELBA LISSETTE MACHACA HUANCA

PARA OPTAR EL TÍTULO PROFESIONAL DE:

INGENIERO DE SISTEMAS E INFORMÁTICA

MOQUEGUA – PERU

2021

UNAM
UNIVERSIDAD NACIONAL DE MOQUEGUA

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

Folio **Nº 002**

ACTA DE SUSTENTACIÓN

I. Modalidad:

Tesis : [X]
Trabajo de Suficiencia Profesional : [ ]
Otro: ..... : [ ]

II. Documento de aprobación para la sustentación:

Resolución Comisión Organizadora Nº 549-2021-UNAM

Fecha: 10 de junio 2021

III. El jurado ha dictaminado la (el) Tesis, TITULADO:

"Diagnóstico de la retinopatía diabética usando algoritmos de clasificación".

Bachiller: Anamelba Lissette Machaca Huanca

Asesor Principal: Mgr. Carlos Alberto Silva Delgado

Co-Asesor: .....

IV. Se dictamina:

Como: Aprobado por Unanimidad la sustentación de manera:

Presencial [ ] Virtual [X]

Con nota: Quince (15) por lo que; Sí [X] No [ ]

Opta el Título Profesional de INGENIERO DE SISTEMAS E INFORMÁTICA.

Siendo las 19:21 horas, del 18 de junio del año 2021 se procede con la firma en conformidad de lo dictaminado por el Jurado.

Presidente
Euler Tito Chura

Primer Miembro

Segundo Miembro

Anamelba Lissette Machaca Huanca
Bachiller

DEDICATORIA

La presente Tesis dedico a Dios, quien con su bendición e infinito amor guía mi vida en cada paso y a la Virgen María, espejo de virtudes.

A mis padres, Julián Machaca Hinojosa y Virginia Huanca Vilca, pilar fundamental en mi formación profesional y personal.

AGRADECIMIENTO

Agradezco a Dios, por ser fortaleza espiritual durante este trabajo. A mis padres, principales promotores de este logro.

Mi gratitud también a mis asesores Mg. Carlos Silva, Dr. Américo Rubín y docentes que contribuyeron con su apoyo y valiosos aportes durante el desarrollo de esta Tesis. A los miembros del jurado por sus recomendaciones para la mejoría del trabajo de investigación.

Finalmente, agradezco a la Universidad Nacional de Moquegua, y a la Vicepresidencia de Investigación, por permitirme ser parte del "III Concurso de Investigación de estudiantes y egresados con financiamiento del Canon Minero y regalías UNAM 2019".

RESUMEN

La Retinopatía Diabética es una complicación de la Diabetes y una de las causas principales de la ceguera debido al deterioro de la retina, que se manifiesta mediante la presencia de exudados, microaneurismas y hemorragias. La presente investigación tiene como objetivo el diagnóstico de la Retinopatía Diabética a partir del modelamiento y comparación de algoritmos de clasificación. En la primera etapa se procedió a la obtención de imágenes de retina del dataset MESSIDOR que contiene 1200 de imágenes adquiridas de 3 departamentos oftalmológicos, establecido para estudios de diagnóstico por computadora, este dataset contiene. En la segunda etapa se realizó pre-procesamiento de imágenes para eliminar el ruido del fondo y para eliminar el disco óptico, debido a que no aportan en el análisis de la imagen. Para la detección de exudados, microaneurismas y hemorragias, realizamos el procesamiento de imágenes, a fin de obtener cuatro características, como el número de exudados, área de exudados, número de microaneurismas y hemorragias y el área de microaneurismas y hemorragias. En la tercera etapa se utilizó modelos Machine Learning para la clasificación de imágenes según la severidad de la enfermedad. Finalmente, se comparó los resultados de las métricas de clasificación, donde el algoritmo SVM obtuvo el mejor resultado.

ABSTRACT

Diabetic Retinopathy is a complication of Diabetes and one of the main causes of blindness due to the deterioration of the retina, that is manifested by the presence of exudates, microaneurysms and hemorrhages. The objective of this research is to get the Diabetic Retinopathy diagnosis from modeling and comparing classification algorithms. The first part of this study was to get the retinal images from the MESSIDOR dataset that contains 1200 images acquired from 3 ophthalmic departments, established for computer diagnostic studies. The second part was to perform image pre-processing in order to remove background noise and to remove the optic disc, because of they do not contribute to the analysis of the image. To detect exudates, microaneurysms and hemorrhages, was used image processing, in order to get four characteristics, such as the number of exudates, area of exudates, number of microaneurysms and hemorrhages, and area of microaneurysms and hemorrhages. In the third part, Machine Learning models were applied to classify images according to the severity of the disease. Finally, classification reports from Machine Learning models were used to compere results, where Support Vector Machine algorithm got better results

ÍNDICE GENERAL

I. INTRODUCCIÓN 10
1.1 DESCRIPCIÓN Y FORMULACIÓN DEL PROBLEMA 10
1.2 ANTECEDENTES 11
1.3 OBJETIVOS 14
1.3.1. OBJETIVO GENERAL 14
1.3.2. OBJETIVOS ESPECÍFICOS 14
1.4 JUSTIFICACIÓN E IMPORTANCIA 14
II. MARCO TEÓRICO 15
2.1 RETINOPATÍA DIABÉTICA 15
2.1.1 CONCEPTO DE LA RETINOPATÍA DIABÉTICA 15
2.1.2 EXUDADOS 15
2.1.3 MICROANEURISMAS 16
2.1.4 HEMORRAGIAS RETINIANAS 16
2.2 VISION ARTIFICIAL (COMPUTER VISION) 17
2.2.1 VISION ARTIFICIAL 17
2.2.2 ESPACIO DE COLOR RGB 19
2.2.3 ESPACIO DE COLOR LAB 17
2.2.4 OPEN CV 20
2.2.5 HISTOGRAMA DE ECUALIZACIÓN 20
2.2.6 OPERACIONES MORFOLÓGICAS 21
2.2.7 DETECCIÓN DE BORDES CANNY 23
2.2.8 CONNECTED COMPONENT 24
2.3 MACHINE LEARNING Y ALGORITMOS DE CLASIFICACIÓN 25
2.3.1 APRENDIZAJE MÁQUINA 25
2.3.2. APRENDIZAJE SUPERVISADO 25
2.3.3. SUPPORT VECTOR MACHINE (SVM) 26
2.3.4. K NEAREST NEIGHBORS 28
2.3.5. PERCEPTRÓN MULTICAPA (MLP) 28
2.3.6. MATRIZ DE CONFUSIÓN 30
2.3.7. EXACTITUD (ACCURACY) 33
2.3.8. PRECISIÓN (PRECISION) 33
2.3.9. EXHAUSTIVIDAD (RECALL) 33
2.3.10. F1 – SCORE 34
III. MÉTODO 35
3.1 TIPO Y NIVEL DE INVESTIGACIÓN 35
3.2 POBLACIÓN Y MUESTRA 35

3.3 INSTRUMENTOS 35
3.4 PROCEDIMIENTOS 35
3.5 ANÁLISIS Y PROCESAMIENTO DE DATOS 36
3.5.1 DESCRIPCIÓN DE EL DATASET 36
3.5.2 PRE - PROCESAMIENDO DE IMÁGENES 38
3.5.3 PROCESAMIENTO DE IMÁGENES 42
3.5.4 EXTRACCIÓN DE CARACTERÍSTICAS 50
3.5.5 ANÁLISIS DE CARACTERÍSTICAS 50
3.5.6 DIVISIÓN EN DATOS EN ENTRENAMIENTO Y TEST 51
3.5.7 NORMALIZACIÓN DE DATOS 52
IV. RESULTADOS 54
4.1 MODELAMIENTO CON ALGORIMOS MACHINE LEARNING 55
4.1.1 K-NEAREST NEIGHBOR 55
4.1.2 RED NEURONAL PERCEPTRÓN MULTICAPA (MLP NN) 57
4.1.3 SUPPORT VECTOR MACHINE 59
V. DISCUSIÓN DE RESULTADOS 62
VI. CONCLUSIONES 63
VII. RECOMENDACIONES 64
REFERENCIAS 65

ÍNDICE DE IMÁGENES

FIGURA 1. RETINOPATÍA DIABÉTICA ……. 16
FIGURA 2. ESPACIO DE COLOR LAB ……..... 18
FIGURA 3. ESPACIO DE COLOR RGB ……... 19
FIGURA 4. ECUALIZACIÓN DE HISTOGRAMA ...……... 20
FIGURA 5. DETECCIÓN DE BORDES CANNY ……. 24
FIGURA 6. SUPPORT VECTOR MACHINE …… 27
FIGURA 7. MULTILAYER PERCEPTRON NEURAL NETWORK ...…… 29
FIGURA 8. CLASIFICACIÓN DE DATOS ……... 37
FIGURA 9. DATASET DE IMÁGENES DE RETINA …….. 37
FIGURA 10. RETINA ORIGINAL ……. 38
FIGURA 11. MÁSCARA DE FONDO DE LA RETINA ……..... 39
FIGURA 12. RETINA PRE-PROCESADA SIN RUIDO ……... 39
FIGURA 13. MÁSCARA DEL DISCO ÓPTICO ……... 40
FIGURA 14. MÁSCARA DEL DISCO ÓPTICO Y FONDO …… 41
FIGURA 15. RETINA CON EL DISCO ÓPTICO DETECTADO …… 41
FIGURA 16. CANAL L DEL ESPACIO DE COLOR ……..……. 42
FIGURA 17. RESTA DEL CANAL L Y LA APERTURA MORFOLÓGICA …...…... 43
FIGURA 18. RESTA DEL CANAL L Y EL CIERRE MORFOLÓGICO ……. 43
FIGURA 19. SUMA DEL CANAL L Y LA APERTURA MORFOLÓGICA ……... 44
FIGURA 20. RESTA DEL CANAL L Y LA SUMA MORFOLÓGICA ……... 44
FIGURA 21. ECUALIZACIÓN DE HISTOGRAMA LUEGO DE OPERACIONES MORFOLÓGICOS …….. 45
FIGURA 22. EXUDADOS DETECTADOS ……... 45
FIGURA 23. CANAL VERDE DE LA IMAGEN DE RETINA …….……... 46
FIGURA 24. ECUALIZACIÓN DE HISTOGRAMA EN EL CANAL VERDE ……... 47
FIGURA 25. BORDES DETECTADOS USANDO CANNY ……... 47
FIGURA 26. OPERACIÓN MORFOÓGICA EN BORDES DETECTADOS ……... 48
FIGURA 27. ELIMINACIÓN DE VASOS SANGUÍNEOS …….. 49
FIGURA 28. MICROANEURISMAS DETECTADOS ….……. 49

ÍNDICE DE TABLAS

TABLA 1. MATRIZ DE CONFUSIÓN DE DOS CLASES ……... 30
TABLA 2. MATRIZ DE CONFUSIÓN MULTICLASE – VP …….. 31
TABLA 3. MATRIZ DE CONFUSIÓN MULTICLASE – VN …….. 31
TABLA 4. MATRIZ DE CONFUSIÓN MULTICLASE – FP ……... 32
TABLA 5. MATRIZ DE CONFUSIÓN MULTICLASE – FN ……. 32
TABLA 6. NÚMERO DE DATOS CLASIFICADOS SEGÚN SU SEVERIDAD …… 36
TABLA 7. CORRELACIÓN DE CARACTERÍSTICAS ……51
TABLA 8. PROPORCIONALIDAD DE DATOS EN ENTRENAMIENTO Y TEST .. 52
TABLA 9. DIVISIÓN DE DATOS EN ENTRENAMIENTO Y TEST ……. 52
TABLA 10. DATOS NORMALIZADOS USANDO STANDARSCALER …….. 53
TABLA 11. MATRIZ DE CONFUSIÓN DEL ALGORITMO KNN …… 56
TABLA 12. REPORTE DE MÉTRICAS DE EVALUACIÓN DEL ALGORITMO KNN ……... 57
TABLA 13. MATRIZ DE CONFUSIÓN DEL ALGORITMO MLP ……..…… 58
TABLA 14. REPORTE DE MÉTRICAS DE EVALUACIÓN DEL ALGORITMO MLP ……..……. 58
TABLA 15. MATRIZ DE CONFUSIÓN DEL ALGORITMO SVM ……...……. 59
TABLA 16. REPORTE DE MÉTRICAS DE EVALUACIÓN DEL ALGORITMO SVM ……... 60
TABLA 17. MATRIZ DE CONFUSIÓN DE SVM USANDO GRIDSEARCH …… 61
TABLA 18. MÉTRICAS DE EVALUACIÓN DE SVM USANDO GRIDSEARCH ... 61
TABLA 19. REPORTE DE MÉTRICA DE EVALUACIÓN F1-CORE …….. 62

I. INTRODUCCIÓN

1.1 DESCRIPCIÓN Y FORMULACIÓN DEL PROBLEMA

La diabetes es un importante problema de salud pública y una de las cuatro enfermedades no transmisibles (ENT) seleccionadas por entidades del ámbito de la salud (OMS, Organización Mundial de la Salud; OPS, Organización Panamericana de la Salud), los cuales establecen de acuerdo a sus estudios como de carácter prioritario. En las últimas décadas han aumentado sin pausa el número de casos y la prevalencia de la enfermedad.

La Organización Mundial de la Salud (OMS) ha proyectado que para el 2030 existan 500 millones de diabéticos en todo el planeta. Esta cifra se debe al crecimiento y envejecimiento de la población; así como a los malos hábitos alimenticios poco saludables y a la vida sedentaria.

Según estimaciones realizadas por dichas actividades, 422 millones de adultos en todo el mundo fueron diagnosticados de diabetes en 2014, en relación a los 108 millones de 1980. La prevalencia mundial (normalizada por edades) casi se ha duplicado desde ese año, pues ha pasado del 4.7% al 8.5% en la población adulta. En la última década, la prevalencia de la diabetes ha aumentado más deprisa en los países de ingresos bajos y medianos que en los de ingresos altos.

En el 2012, la diabetes provocó 1.5 millones de muertes. Un nivel de glucosa en la sangre superior al deseable provocó 2.2 millones de muertes, al incrementar los riesgos de enfermedades cardiovasculares y de otro tipo.

La diabetes mellitus puede provocar complicaciones en muchas partes del organismo e incrementar el riesgo general de muerte prematura. Entre las posibles complicaciones se incluyen: ataques cardiacos, accidentes cerebrovasculares, insuficiencia renal, amputación de piernas, daños neurológicos y pérdida de visión. (OMS, 2016).

De otro lado, la retinopatía diabética es una causa importante de ceguera y es la consecuencia del daño de los pequeños vasos sanguíneos de la retina que se va acumulando a lo largo del tiempo. El 2.6% de los casos mundiales de ceguera es consecuencia de la diabetes.

Considerando que el mejor tratamiento para la Retinopatía Diabética es la prevención, vinculado al control adecuado de los niveles de glucosa, dieta sana, ejercicio físico y buena educación sanitaria. Desde el punto de vista oftalmológico, la prevención secundaria se basará en la detección precoz de la retinopatía y un adecuado seguimiento.

En la retinopatía diabética se realiza un examen ocular con la pupila dilatada. Se toma fotografías del interior de los ojos para un posterior análisis de un oftalmólogo. El análisis requiere personal médico, acudiendo a su conocimiento y experiencia mediante propiedades visuales que son de utilidad para detectar la presencia de la enfermedad.

1.2 ANTECEDENTES

Estudios relacionados al diagnóstico de la Retinopatía Diabética usando Inteligencia Artificial.

- Según Karan Bhatia, Shikhar Arora, Ravi Tomar (2016), se realizó un estudio de la automatización del diagnóstico de la retinopatía diabética usando algoritmos de Machine Learning Este trabajo se enfoca en la presencia de la enfermedad aplicando algoritmos de machine learning sobre la extracción de características del procesamiento de imágenes de retina, como el diámetro del disco óptico, lesiones específicas (microaneurismas, exudados), Para la predicción de presencia de la retinopatía diabética se usó Árboles de Decisiones, AdaBoots, Naive Bayes, Random Forest y Support Vector Machine (SVM).

- En el estudio de Omer, Utku (2018), sobre el diagnóstico de la retinopatía de imágenes fundus de retina, usa procesamiento de

imágenes y redes convolucionales. Para mejorar las imágenes de retina se usó el espacio de color HSV (Hue, Saturation and Value; en español Profundidad, Saturación y Valor), el algoritmo de transformación V y técnicas de ecualización de histograma. Finalmente se aplicaron filtros Gausianos. Luego del procesamiento de imágenes se realizó la clasificación usando Redes Neuronales Convolucionales (CNN), utilizando imágenes del dataset de Kaggle. En este experimento, se obtuvo una precisión de 97%, sensibilidad de 97.67%, especificidad de 93.33%, y un F1-score de 93.33%. Los resultados obtenidos muestran que el método propuesto es muy eficiente y exitoso para diagnosticar la Retinopatía Diabética de imágenes fundes de la retina.

- En la investigación de Asti Herliana, Toni Arifin, Sara Susanti y Agung Baitul Hitmah (2018), realizaron una selección de características de la enfermedad de la retinopatía diabética usando partículas de optimización Swarm, luego para la selección de características se usó el método de clasificación de Redes Neuronales basado en la optimización de partículas Swarm (PSO) con un 76.11% como resultado. Este estudio también mostró que hay un incremento en el resultado de clasificación usando el método de selección de características de un 4.35%. de un resultado previo de 71.76% solo aplicando redes neuronales.

- El texto científico de Shailesh Kumar, Basant Kumar (2018), presenta un esquema de detección de la retinopatía diabética, extrayendo de forma precisa el área y número de microaneurismas de imágenes fundus. Inicialmente, aplica técnicas de pre-procesamiento como la extracción del canal verde, ecualización de histograma y procesos morfológicos. Para la detección de microaneurismas, se usó el análisis de componente principal (PCA), ecualización de histograma adaptativo limitado contraste (CLAHE), procesos morfológicos y filtros. Para la clasificación de la retinopatía diabética se usó Support Vector Machine (SVM). La sensibilidad y especificidad fue de 96% y 92% respectivamente.

- En el trabajo de Enrique V. Carrera, Andrés Gonzalez, Ricardo Carrera (2015), se procesó un diagnóstico asistido por computadora basado en el procesamiento digital de imágenes de la retina para ayudar en la detección de la retinopatía diabética. El principal objetivo es clasificar automáticamente el grado de la retinopatía no-proliferativa en imágenes de la retina. Para ello se inició el procesamiento de vasos sanguíneos, microaneurismas y exudados para extraer características que puedan ser usados por un Support Vector Machine para detectar el grado de retinopatía de cada imagen de retina. Esta propuesta ha sido probada en 400 imágenes de retina etiquetados de acuerdo a una escala de cuarto grado de la retinopatía diabética no-porlifeerativa. Como resultado, se obtuvo se obtuvo una sensibilidad máxima de 95% y capacidad productiva de 94%.

- La investigación de Sohini Roy Chow Dhury (2013), analizó imágenes con iluminación variada y vista de campos, y genera un grado severo de retinopatía diabética usando Machine learning. Clasificadores como el Modelo Mixto Gaussiano (GMM), k-nearest neighbor (KNN), Support Vector Machine (SVM), Y AdaBoost son analizados para la clasificación de lesiones y no lesiones de la retinopatía. Los clasificadores GMM y KNN son los mejores clasificadores para la clasificación de lesiones rojas y lesiones brillantes, respectivamente. Una contribución importante de este paper es la reducción del número de características para la clasificación de lesiones por ranking usando Adaboost donde se seleccionaron 30 de 78 características. Las lesiones brillantes son clasificados como exudados y lesiones rojas son clasificadas como hemorragias y microaneurismas. Esta clasificación de lesiones lidia con data desbalanceada. El grado de severidad de la retinopatía diabética, es testeado en 1200 imágenes del dataset MESSIDOR. Se obtuvo una sensibilidad de 100%, 53.16% de especificidad y 0.904 AUC. La reducción de características reduce el tiempo computacional por imagen, tomando de 59.54 segundos a 3.46 segundos.

1.3 OBJETIVOS

1.3.1. OBJETIVO GENERAL

- Diagnóstico de la Retinopatía Diabética a partir del modelamiento y comparación de algoritmos de clasificación.

1.3.2. OBJETIVOS ESPECÍFICOS

- Procesar imágenes de la retina para la extracción de características que determinan la presencia de la enfermedad.

- Modelar algoritmos machine learning que clasifiquen la enfermedad según la severidad de la retinopatía diabética.

- Comparar las métricas de evaluación utilizadas de los algoritmos empleados para la clasificación de la retinopatía diabética.

1.4 JUSTIFICACIÓN E IMPORTANCIA

La detección temprana de la Retinopatía Diabética es crucial para minimizar el riesgo de la ceguera. La necesidad de recursos humanos médicos y equipos médicos especializados dificulta el diagnóstico en personas de pocos recursos. Una alternativa eficiente para el diagnóstico de la enfermedad mediante la implementación de modelos computacionales que faciliten el diagnóstico de la enfermedad utilizando imágenes digitales de la retina, que no requiere una atención presencial, en tiempos donde la telemedicina va tomando importancia ante la actual pandemia.

Por este motivo proponemos la implementación de algoritmos que nos permitan clasificar el grado de la Retinopatía Diabética usando técnicas de visión artificial para el pre-procesamiento y procesamiento de imágenes. Además, que también nos permitan eliminar ruido, seleccionar zonas de interés y extraer características de la Retinopatía Diabética.

II. MARCO TEÓRICO

2.1 RETINOPATÍA DIABÉTICA

2.1.1 CONCEPTO DE LA RETINOPATÍA DIABÉTICA

Es una complicación de la diabetes que afecta los ojos, causada por el daño a los vasos sanguíneos que van al tejido sensible a la luz que se encuentra en el fondo del ojo (retina).

Inicialmente, la retinopatía diabética puede no presentar síntomas o solo presentar problemas leves de visión. A medida que la enfermedad avanza, algunos de los síntomas pueden comprender:

- Manchas o hebras oscuras que flotan en la vista.
- Visión borrosa
- Visión variable
- Visión de colores alterada
- Zonas de la visión oscuras o vacías
- Pérdida de la visión

Cualquier persona con diabetes tipo 1 o tipo 2 puede padecer este trastorno. Cuanto más tiempo hayas tenido diabetes y cuanto menos te hayas controlado el azúcar en sangre, mayor la probabilidad de presentar esta complicación en los ojos. (Mayo Foundation dor Medical Education and Researcn, 2021)

2.1.2 EXUDADOS

Los exudados son depósitos lipídicos intrarretinianos profundos de color amarillento brillante y bien definidos. En ocasiones aparecen formando un anillo en torno a un foco de microangiopatía diabética con microaneurismas y capilares incompetentes. Los exudados blandos o algodonosos son engrosamientos isquémicos de la capa de fibras nerviosas, de forma más redondeada de bordes más difusos y color blanco, tapan los vasos retinianos.

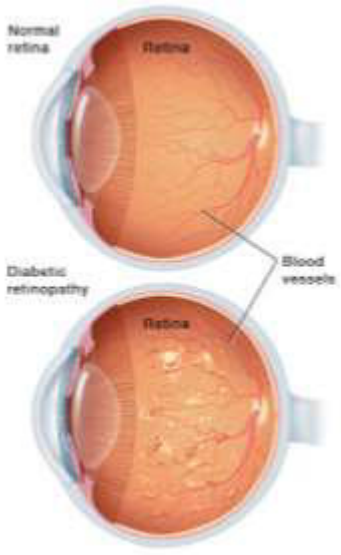
2.1.3 MICROANEURISMAS

Los microaneurismas son dilataciones de la pared de los capilares y aparecen como puntos rojos pequeños de bordes muy nítidos. Son dilataciones de las paredes de los vasos sanguíneos, posiblemente secundarias a la debilidad de la pared vascular por pérdida de pericitos o por un aumento de la presión intraluminal. Los microaneurismas pueden resolverse con el paso del tiempo o pueden permanecer estables durante mucho tiempo. Son indistinguibles en la exploración de las microhemorragias.

2.1.4 HEMORRAGIAS RETINIANAS

Pueden presentar un aspecto diferente según la capa de la retina donde asienten. Las hemorragias puntiformes (se confunden con los microaneurismas y sólo se pueden diferenciar mediante la angiografía con fluoresceína) son de localización profunda, mientras que las hemorragias en llama están localizadas a nivel más superficial en la capa de fibras nerviosas.

Figura 1.
Retinopatía Diabética



Nota: Retina saludable vs retina con de Retinopatía Diabética
Fuente: Retinopatía Diabética 2018, (www.mayoclininc.org)

2.2 VISION POR COMPUTADORA

2.2.1 VISION POR COMPUTADORA (COMPUTER VISION)

Según D. Ballard, C. Brown, 1982. La visión computacional o visión artificial es el estudio de reconocer y localizar objetos en el medio ambiente mediante el procesamiento de imágenes, para entenderlos y construir máquinas con capacidades similares.

Disciplina científica que incluye métodos orientados a adquirir, procesar, analizar y comprender las imágenes del mundo real con el fin de producir información numérica o simbólica para que puedan ser tratados por un ordenador.

La visión artificial trata de producir el mismo efecto para que los ordenadores puedan percibir y comprender una imagen o secuencia de imágenes y actuar según convenga en una determinada situación.

Esta comprensión se consigue gracias a distintos campos como la geometría, la estadística, la física y otras disciplinas. La adquisición de datos se consigue por varios medios como secuencias de imágenes, vistas desde varias cámaras de video o datos multidimensionales desde un escáner médico.

2.2.2 ESPACIO DE COLOR LAB

El color corresponde a una percepción e interpretación subjetiva. Dos personas mirando un mismo objeto pueden usar puntos de referencia distintos y expresar el mismo color con una gran variedad de palabras diferentes, llevando a confusión y falta de comunicación internamente o a través de la cadena de abastecimiento. Para evitar esto y asegurar que una muestra cumpla con el estándar, el color debe ser expresado en términos numéricos y objetivos.

Cuando se clasifican los colores, se los puede expresar en términos de matiz (color), luminosidad (brillo) y saturación (vividez). Al crear escalas para éstos atributos, podemos expresar en forma precisa el color.

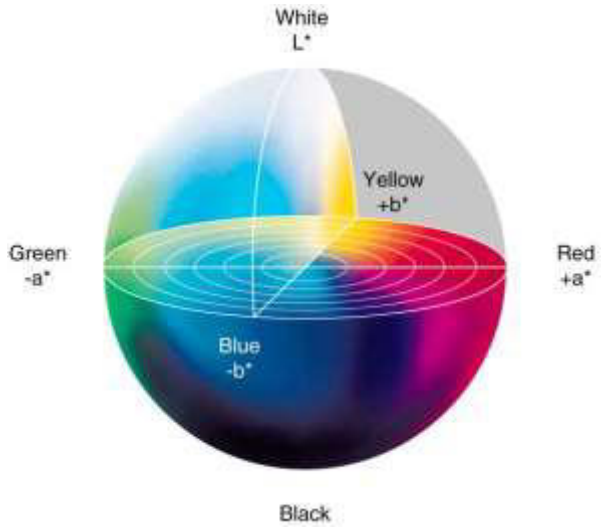
El espacio de color L*a*b* fue modelado en base a una teoría de color oponente que establece que dos colores no pueden ser rojo y verde al mismo tiempo o amarillo y azul al mismo tiempo. Como se muestra a continuación, L*indica la luminosidad y a* y b* son las coordenadas cromáticas.

A diferencia de los modelos de color RGB , CIELAB está diseñado para aproximarse a la visión humana. El componente L * se asemeja mucho a la percepción humana de la luminosidad

L* = luminosidad
a* = coordenadas rojo/verde (+a indica rojo, -aindica verde)
b* = coordenadas amarillo/azul (+b indica amarillo, -b indica azul)

Figura 2.

Espacio de color LAB



Fuente: Entendiendo el espacio de color LAB, 2012. (https://sensing.konicaminolta.us/mx/blog/entendiendo-el-espacio-de-color-cie-lab/)

2.2.3 ESPACIO DE COLOR RGB

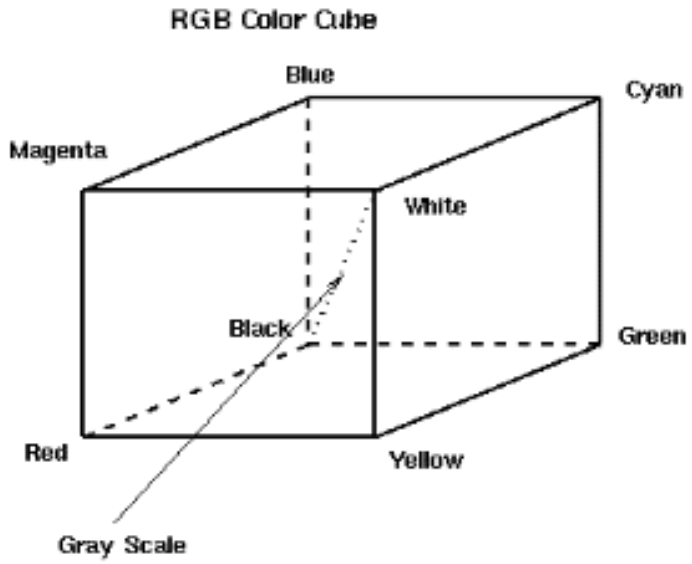
RGB (Red, Green, Blue) es un aditivo modelo de color en la que R representa el color Rojo, G representa el color Verde y B representa el color Azul. Generalmente los componentes se normalizan, obteniendo:

- r = R / (R + G+ B)
- g = G / (R + G+ B)
- b = B / (R + G+ B)

Este modelo se basa en un sistema de coordenadas cartesianas como se puede ver en la Figura 2.

Figura 3.

Cubo de color RGB



Fuente: Color image processing, 2020, (https://www.cronj.com/blog/color-space-image-processing/)

2.2.4 OPEN CV

Open Source Computer Vision es una biblioteca principal en el área visión por computadora y procesamiento de imágenes en tiempo real. Escrito en C ++ y su interfaz principal está en C ++, pero aún conserva una interfaz C más antigua, menos completa aunque extensa . Todos los algoritmos tienen una interfaz en C ++. Hay interfaces en Python , Java y MATLAB / OCTAVE .

2.2.5 ECUALIZACIÓN DE HISTOGRAMA

La ecualización de histograma es un método en el área de procesamiento de imágenes que permite ajustar el contraste de una imagen utilizando el histograma de la misma.

OpenCV tiene un algoritmo para hacer esto, cv2.equalizeHist(). Su entrada es solo una imagen en escala de grises y la salida es nuestra imagen ecualizada de histograma.

Figura 4.

Ecualización de histograma



Fuente: Histogram equalization in Open CV, 2021. (https://docs.opencv.org/master/d5/df/tutorial_py_histogram_equalization.html)

2.2.6 OPERACIONES MORFOLÓGICAS

Las operaciones morfológicas en el procesamiento de imágenes persiguen el objetivo de eliminar ruidos y texturas teniendo en cuenta la forma y estructura de la imagen. Son una colección de operaciones no lineales relacionadas con la forma morfológica de las características en una imagen. Se basan únicamente en el orden relativo de los valores de los píxeles, no en sus valores numéricos y, por lo tanto, son especialmente adecuadas para el procesamiento de imágenes binarias.

Las operaciones de Erosión y Dilatación tienen el inconveniente de disminuir o aumentar el tamaño del resto de los objetos. Este efecto puede ser subsanado con una aplicación en cascada de erosión y dilatación binaria con igual efecto

2.2.6.1 DILATACIÓN

Es un algoritmo basado en una estructura matricial que se utiliza asignado se utiliza para sondear y expandir las formas o límites de áreas específicas contenidas en la imagen de entrada. En específico, actúa como filtro máximo local. La dilatación tiene el efecto contrario a la erosión. Agrega una capa de píxeles a los límites internos y externos de las regiones en donde se aplique.

Es decir, el valor del píxel de salida es el valor máximo de todos los píxeles de la vecindad. En una imagen binaria, un píxel se establece en 1 si alguno de los píxeles vecinos tiene el valor 1. La dilatación morfológica hace que los objetos sean más visibles y se pueda rellenar pequeños agujeros en los objetos.

2.2.6.2 EROSIÓN

Algoritmo opuesto a la dilatación. La estructura matricial asignado se utiliza para sondear y reducir las formas contenidas en la imagen de entrada. En específico, actúa como filtro mínimo local. Además, los elementos de esta estructura encogen una imagen eliminando una capa de píxeles de los límites internos y externos de las regiones en donde se pasa este filtro.

Es decir, el valor del píxel de salida es el valor mínimo de todos los píxeles de la vecindad. En una imagen binaria, un píxel se establece en 0 si alguno de los píxeles vecinos tiene el valor 0. La erosión morfológica elimina las islas y los objetos pequeños de modo que solo quedan los objetos sustantivos o con mayor área dentro de una imagen.

2.2.6.3 APERTURA MORFOLÓGICA

Se consigue erosionando primero una imagen y luego dilatándola. La apertura elimina las conexiones y líneas estrechas entre dos regiones.

La operación de apertura erosiona una imagen y luego dilata la imagen erosionada, utilizando el mismo elemento de estructura matricial para ambas operaciones. La apertura morfológica es útil para eliminar objetos pequeños de una imagen mientras se conserva la forma y el tamaño de los objetos más grandes en la imagen.

$$Apertura : Erosión + Dilatación$$

2.2.6.4 CIERRE MORFOLÓGICO

Se hace dilatando primero la imagen y luego erosionándola. El orden es inverso al de apertura. El cierre rellena las regiones estrechas o los agujeros de la imagen.

La operación de cierre dilata una imagen y luego erosiona la imagen dilatada, utilizando el mismo elemento de estructura matricial para ambas operaciones. El cierre morfológico es útil para rellenar pequeños agujeros de una imagen mientras se conserva la forma y el tamaño de los objetos de la imagen.

$$Cierre : Dilatación + Erosión$$

2.2.7 DETECCIÓN DE BORDES CANNY

El algoritmo Canny es un filtro que utiliza un algoritmo de múltiples etapas para detectar una amplia gama de bordes en imágenes. El algoritmo de Canny consiste en tres grandes pasos:

- Obtención del gradiente: en este paso se calcula la magnitud y orientación del vector gradiente en cada píxel.

- Supresión no máxima: en este paso se logra el adelgazamiento del ancho de los bordes, obtenidos con el gradiente, hasta lograr bordes de un píxel de ancho.

- Histéresis de umbral: en este paso se aplica una función de histéresis basada en dos umbrales; con este proceso se pretende reducir la posibilidad de aparición de contornos falsos.

Figura 5.

Detección de bordes Canny



Fuente: Histogram equalization in Open CV, 2021. (https://docs.opencv.org/master/d5/daf/tutorial_py_histogram_equalization.html)

2.2.8 CONNECTED COMPONENT

Es un algoritmo que se utiliza en visión por computadora para detectar regiones conectadas en imágenes digitales binarias, aunque también se pueden procesar imágenes en color y datos con mayor dimensionalidad.

Cuando se integra en un sistema de reconocimiento de imágenes, el etiquetado de componentes conectados puede operar con una mayor variedad de información.

2.3 MACHINE LEARNING Y ALGORITMOS DE CLASIFICACIÓN

2.3.1 APRENDIZAJE MÁQUINA

Aprendizaje máquina (Machine learning), Disciplina científica del ámbito de la Inteligencia Artificial que crea sistemas que aprenden automáticamente. Aprender en este contexto quiere decir identificar patrones complejos a partir de un entrenamiento usando millones de datos. La máquina que realmente aprende es un algoritmo que revisa los datos y es capaz de predecir comportamientos futuros.

2.3.2. APRENDIZAJE SUPERVISADO

Los algoritmos trabajan con datos "etiquetados" (labeled data), intentado encontrar una función que, dadas las variables de entrada (input data), les asigne la etiqueta de salida adecuada. El algoritmo se entrena con un "histórico" de datos y así "aprende" a asignar la etiqueta de salida adecuada a un nuevo valor, es decir, predice el valor da salida.

El aprendizaje supervisado se suele usar en problemas de clasificación, como identificación de dígitos, diagnósticos, o detección de fraude de identidad. También se usa en problemas de regresión, como predicciones meteorológicas, de expectativa de vida, de crecimiento etc. Estos dos tipos principales de aprendizaje supervisado, clasificación y regresión, se distinguen por el tipo de variable objetivo. En los casos de clasificación, es de tipo categórico, mientras que, en los casos de regresión, la variable objetivo es de tipo numérico.

2.3.3. SUPPORT VECTOR MACHINE (SVM)

Según Bealdung (2020), las máquinas de vectores de soporte o máquinas de vector soporte (del inglés Support Vector Machine, SVM) son un conjunto de algoritmos de aprendizaje supervisado desarrollados por Vladimir Vapnik y su equipo en los laboratorios AT & T.

Estos métodos están propiamente relacionados con problemas de clasificación y regresión. Dado un conjunto de ejemplos de entrenamiento (de muestras) podemos etiquetar las clases y entrenar una SVM para construir un modelo que prediga la clase de una nueva muestra.

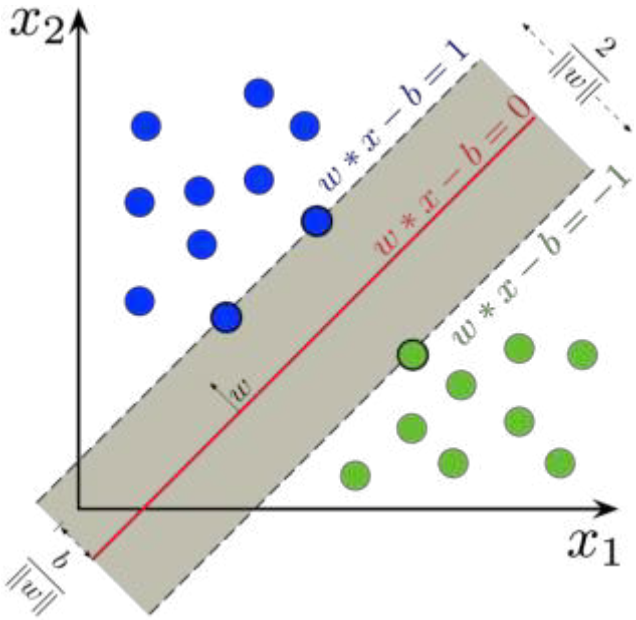
Intuitivamente, una SVM es un modelo que representa a los puntos de muestra en el espacio, separando las clases a 2 espacios lo más amplios posibles mediante un hiperplano de separación definido como el vector entre los 2 puntos, de las 2 clases, más cercanos al que se llama vector soporte. Cuando las nuevas muestras se ponen en correspondencia con dicho modelo, en función de los espacios a los que pertenezcan, pueden ser clasificadas a una o la otra clase.

Más formalmente, una SVM construye un hiperplano o conjunto de hiperplanos en un espacio de dimensionalidad muy alta (o incluso infinita) que puede ser utilizado en problemas de clasificación o regresión. Una buena separación entre las clases permitirá una clasificación correcta.

En el gráfico siguiente se muestra un hiperplano de margen máximo y márgenes para una SVM entrenada con muestras de dos clases. Las muestras en el margen se denominan vectores de soporte.

Figura 6.

Support Vector Machine



Nota: Ejemplo de gráfico de un Support Vector Machine
Fuente: Support Vector Machine, 2021.
(http://albertotb.com/curso-ml-R/Rmd/07-svm/07-svm.html)

- $W^T X - b = 0$

  Cualquier hiperplano se puede escribir como el conjunto de puntos x satisfactorio

- $W^T X - b = 1$

  Todo lo que esté dentro o por encima de este límite es una clase. Con etiqueta 1.

- $W^T X - b = -1$

  Todo lo que esté dentro o por debajo de este límite es de la otra clase, con la etiqueta -1.

2.3.4. K NEAREST NEIGHBORS

Es un método de clasificación supervisado que sirve para estimar la función densidad. En el reconocimiento de patrones, el algoritmo k-NN es usado como método de clasificación de objetos basado en un entrenamiento mediante ejemplos cercanos en el espacio de los elementos. K-NN es un tipo de aprendizaje vago (lazy learning), donde la función se aproxima solo localmente y todo el cómputo es diferido a la clasificación. La normalización de datos puede mejorar

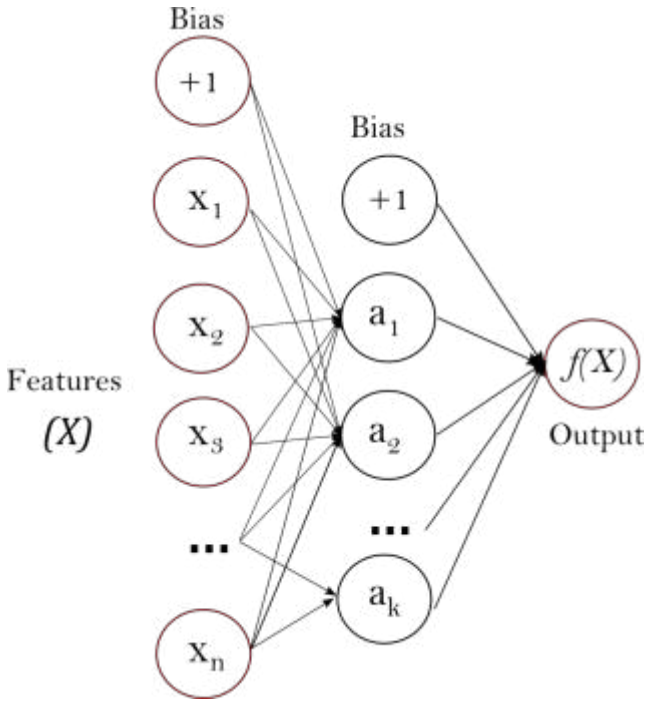
La fase de entrenamiento del algoritmo consiste en almacenar los vectores característicos y las etiquetas de las clases de los ejemplos de entrenamiento. En la fase de clasificación, la evaluación del ejemplo (del que no se conoce su clase) es representada por un vector en el espacio característico. Se calcula la distancia entre los vectores almacenados y el nuevo vector, y se seleccionan los ejemplos más cercanos. El nuevo ejemplo es clasificado con la clase que más se repite en los vectores seleccionados.

2.3.5. PERCEPTRÓN MULTICAPA (MLP)

Perceptrón multicapa es un algoritmo de aprendizaje supervisado que aprende una función $f(.) : R^m \rightarrow R^0$ entrenando en un conjunto de datos, donde m es el número de extensiones para la entrada y 0 es el número de dimensiones para la salida. Dado un conjunto de características $X = x_1, x_2, x_3, \dots, x_m$ y un objetivo y, puede aprender un aproximador de función no lineal para clasificación o regresión. Es diferente de la regresión logística, en que entre la capa de entrada y la de salida, puede haber una o más capas no lineales, llamadas capas ocultas. La figura 11 muestra un MLP de capa oculta con salida escalar.

Figura 7.

Multilayer Perceptron Neural Network



Nota: MLP de una capa escondida
Fuente: Neural Network Models. (https://scikit-learn.org/stable/modules/neural_networks_supervised.html)

La capa más a la izquierda conocida como capa de entrada, consta de un conjunto de neuronas $\{x_i x_1 x_2, \ldots, x_m\}$ que representa las características de entrada. Cada neurona de la capa oculta transforma los valores de la capa anterior con una suma lineal ponderada $w_1 x_1 + w_2 x_2 +, w_m x_m$, seguido de una función de activación no lineal $g(.) : R \rightarrow R -$ como la función de bronceado hiperbólico. La capa de salida recibe los valores de la última capa oculta y los transforma en valores de salida. MLP tiene la capacidad de aprender modelos no lineales.

2.3.6. MATRIZ DE CONFUSIÓN

La matriz de confusión visualiza el desempeño de cada modelo. El tamaño de la matriz de confusión está determinado por el número de clases que queremos predecir.

TERMINOLOGÍAS:

- **VN**: La clase real es negativa y es predicho como negativo. "**Verdadero Negativo".**
- **VP:** La clase real es positiva y es predicho como positivo. "**Verdadero Positivo".**
- **FN:** La clase real es positiva, pero es predicho como negativo. La etiqueta predicha de la clase es errada. Por lo que se considera un **"Falso Negativo"**.
- **FP:** La clase real es negativa, pero es predicho como positivo. La etiqueta predicha de la clase es errada. Por lo que se considera un **"Falso Positivo"**.

MATRIZ DE CONFUSIÓN BINARIA

Tabla 1.

Matriz de confusión de dos clases

Predicciones →

↓ Observación

	Positivos	**Negativos**
Positivos	Verdaderos Positivos (VP)	Falsos Negativos (FN)
Negativos	Falso Positivo (FP)	Verdaderos Negativos (VN)

Nota: Matriz de confusión de dos clases.
Fuente: Matriz de confusión – Evaluación de modelos de clasificación de Carlos Zelad, 2017.

MATRIZ DE CONFUSIÓN MULTICLASE

Si consideramos más de 2 clases, la diagonal indica la cantidad de clasificaciones correctas realizadas por el algoritmo, y las restantes son misclasificaciones.

Tabla 2.

Matriz de confusión multiclase – VP

Predicciones →

	A	B	C	D
A	**9**	**1**	**0**	**0**
B	**1**	**15**	**3**	**1**
C	**5**	**0**	**24**	**1**
D	**0**	**4**	**1**	**15**

↓ Observación

Nota: Matriz de confusión multiclase, Verdaderos Positivos de la clase A.
Fuente: Matriz de confusión en machine learning. Shubhanshi Singh, 2018 (https://medium.com/@shubhanshi.shubh860/confusion-matrix-in-machine-learning-cd7333d72f5d)

Tabla 3.

Matriz de confusión multiclase – VN

Predicciones →

	A	B	C	D
A	**9**	**1**	**0**	**0**
B	**1**	**15**	**3**	**1**
C	**5**	**0**	**24**	**1**
D	**0**	**4**	**1**	**15**

↓ Observación

Nota: Tabla de valores **Verdaderos Negativos** (VN) para la clase A, se expresa en la zona coloreada.
Fuente: Matriz de confusión en machine learning. Shubhanshi Singh, 2018

Tabla 4.

Matriz de confusión multiclase – FP

Predicciones →

↓ Observación	A	B	C	D
A	**9**	**1**	**0**	**0**
B	**1**	**15**	**3**	**1**
C	**5**	**0**	**24**	**1**
D	**0**	**4**	**1**	**15**

Nota: Valores **Falsos Positivos** (FP), para la clase A, se expresa en la zona coloreada.
Fuente: Matriz de confusión en machine learning. Shubhanshi Singh, 2018

Tabla 5.

Matriz de confusión multiclase – FN

Predicciones →

↓ Observación	A	B	C	D
A	**9**	**1**	**0**	**0**
B	**1**	**15**	**3**	**1**
C	**5**	**0**	**24**	**1**
D	**0**	**4**	**1**	**15**

Nota: Valores **Falsos Negativos** (FN) para la clase A, se expresa en la zona coloreada.
Fuente: Matriz de confusión en machine learning. Shubhanshi Singh, 2018

2.3.7. EXACTITUD (ACCURACY)

Para estimar la exactitud de un test, debemos calcular la proporcionalidad de True Positive y True Negative en todos los casos evaluados.

En general, ¿con qué frecuencia es correcto el clasificador?

La exactitud es calculada como el número total de predicciones correctas dividido por el total de número de datos del dataset. Matemáticamente, puede ser expresado de la siguiente manera:

$$Accuracy = \frac{TP + TN}{TOTAL}$$

2.3.8. PRECISIÓN (PRECISION)

La precisión es intuitivamente la capacidad del clasificador de no etiquetar como positiva una muestra que es negativa.

Matemáticamente, puede ser expresado de la siguiente manera:

$$Precisión = \frac{TP}{TP + FP}$$

2.3.9. EXHAUSTIVIDAD (RECALL)

Recall es intuitivamente la capacidad del clasificador de encontrar todas las muestras positivas.

Matemáticamente, puede ser expresado de la siguiente manera:

$$Recall = \frac{TP}{TP + FN}$$

2.3.10. F1 – SCORE

F-beta se puede interpretar como una media armónica ponderada de la precisión y la recuperación, donde una puntuación F-beta alcanza su mejor valor en 1 y la peor puntuación en 0.

F1 score es una métrica ideal para data desbalanceada. Considera ambas métricas la precisión y la exhaustividad para una mejor evaluación.

$$F1\ Score = \ 2 * \frac{Precision * \ Recall}{Precision + Recall}$$

2.3.11. MACRO AVERAGE

Calcula el promedio de una métrica de clasificación y devuelve el promedio sin considerar la proporción de cada etiqueta en el conjunto de datos.

$$F1_{class1} + F1_{class2} + \cdots + F1_{classN}$$

La penalización del resultado es mayor cuando el modelo no tiene un resultado considerablemente homogéneo en todas las clases y cuando se tiene datos desbalanceados.

2.3.12. WEIGHTED AVERAGE

Calcula el promedio de una métrica de clasificación y devuelve el promedio considerando la proporción de cada etiqueta en el conjunto de datos; por lo tanto favorece la clase mayoritaria.

$$F1_{class1} * W_1 + F1_{class2} * W_2 + \cdots + F1_{classN} * W_N$$

III. MÉTODO

3.1 TIPO Y NIVEL DE INVESTIGACIÓN

La presente investigación reúne las condiciones metodológicas de una investigación aplicada, en razón a que se utilizaron conocimientos de Visión por computadora y Machine Learning, necesarios para cumplir con el objetivo del presente trabajo.

3.2 POBLACIÓN Y MUESTRA

- **POBLACIÓN**

  La población motivo de esta investigación está conformada por 1200 imágenes de retina del dataset MESSIDOR.

- **MUESTRA**

  La muestra utilizada en la presente investigación, está conformada por el total de la población de imágenes de retina con cuatro clasificaciones de la retinopatía diabética según su severidad.

3.3 INSTRUMENTOS

- Técnicas de Visión Computacional.
- Técnicas Machine Learning.

3.4 PROCEDIMIENTOS

- Pre-procesamiento de imágenes
- Procesamiento de imágenes
- Modelamiento de Algoritmos de Clasificación
- Predicción de los modelos.

3.5 ANÁLISIS Y PROCESAMIENTO DE DATOS

En el análisis y procesamiento de datos, se describió el origen de la data y aplicó técnicas de visión computacional, que consta en el pre – procesamiento de datos que prepara la imagen, elimina o extrae regiones que distorsionan la data, para luego aplicar un procesamiento donde se detecta las zonas de interés como exudados, microaneurismas y hemorragias; características que ayudan a determinar la existencia de la enfermedad.

Luego del procesamiento de imágenes, se calculó el área y número de exudados; también el cálculo del área y número de microaneurismas y hemorragias. Dichos datos son ingresados a la Máquina de Aprendizaje (Machine Learning) para el entrenamiento y predicción.

3.5.1 DESCRIPCIÓN DE EL DATASET

La base datos de imágenes que se utilizó es MESSIDOR (Method to Evaluate Segmentation and Indexing Techniques in the field of Retinal Ophthalmology), establecido para estudios en diagnóstico por computadora de la retinopatía diabética.

El dataset contiene 1200 de imágenes adquiridas de 3 departamentos oftalmológicos, que se dividen en 4 clases según su severidad.

Tabla 6.

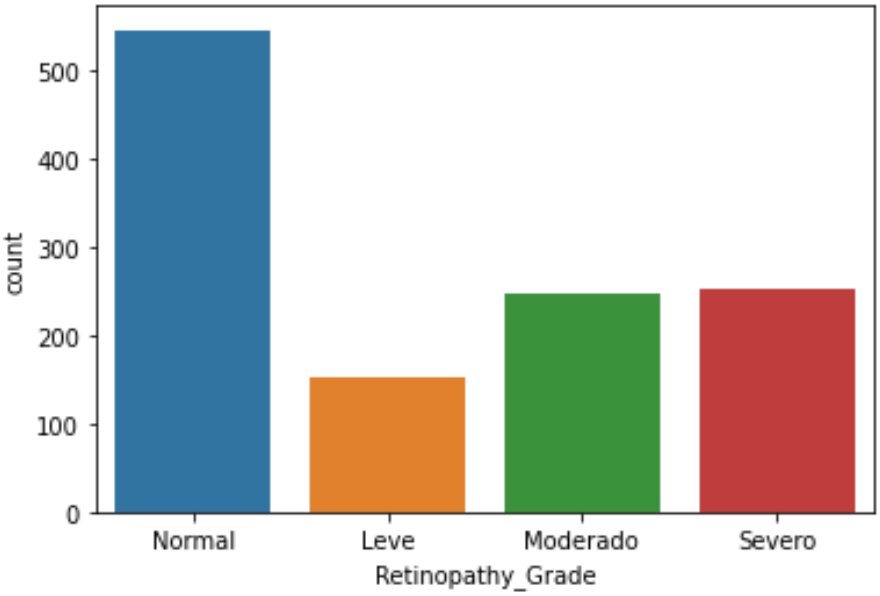
Número de datos clasificados según su severidad

CLASIFICACIÓN	CANTIDAD
Normal	546
Leve	153
Moderado	247
Severo	254

Fuente: Elaboración propia usando Python

Figura 8.

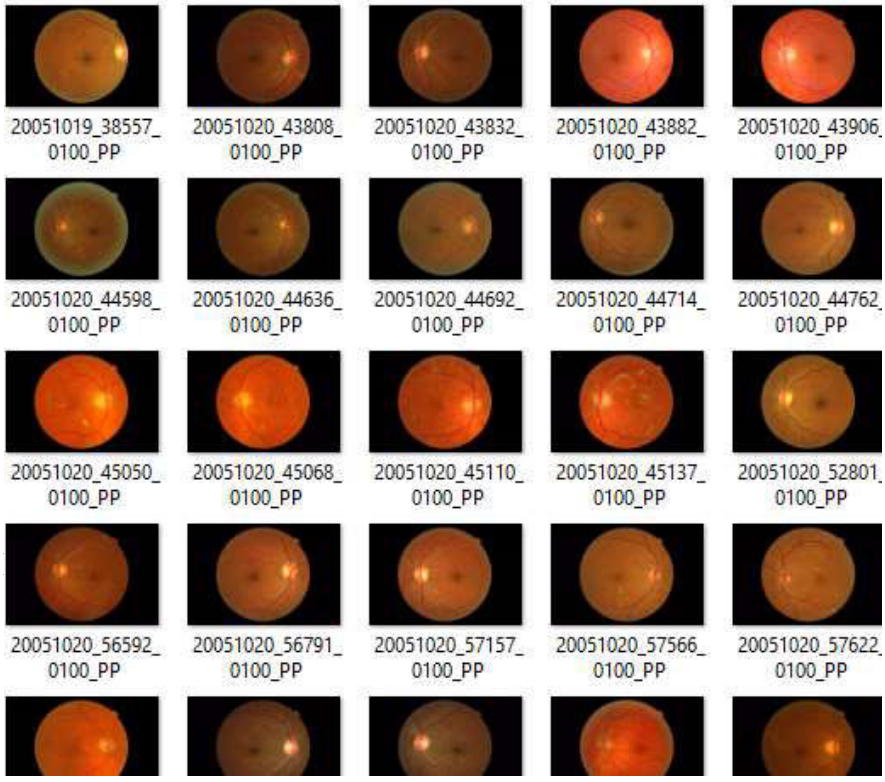
Clasificación de datos



Nota: Gráfico del número de datos clasificados según su severidad
Fuente: Elaboración propia

Figura 9.

Dataset de imágenes de retina

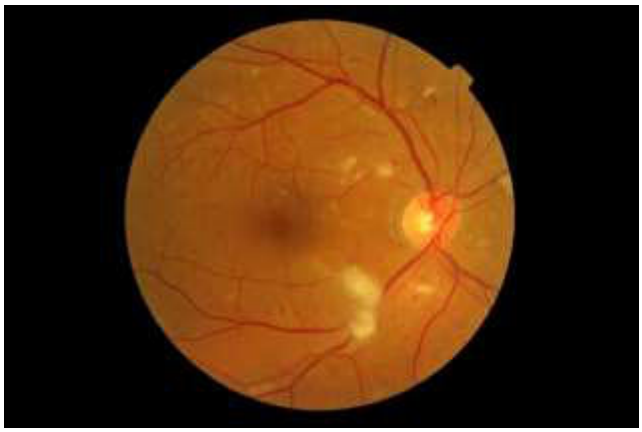


3.5.2 PRE - PROCESAMIENDO DE IMÁGENES

El dataset consta de imágenes de diferentes tamaños, para mantener la homogeneidad del tamaño de la imagen, por ese motivo se aplicó una redimensión de 2240 * 1488 píxeles, aplicando la función *resize* de la librería Open CV.

Figura 10.

Retina original



Fuente: MESSIDOR, 2021

3.5.2.1 DETECCIÓN Y LIMPIEZA DEL FONDO DE LA RETINA

En el análisis de las imágenes se observó un aparente fondo negro; sin embargo, se detectaron píxeles de color, para lo cual aplicamos un pre-procesamiento a fin de excluir ruido del fondo que podría distorsionar el procesamiento de la imagen.

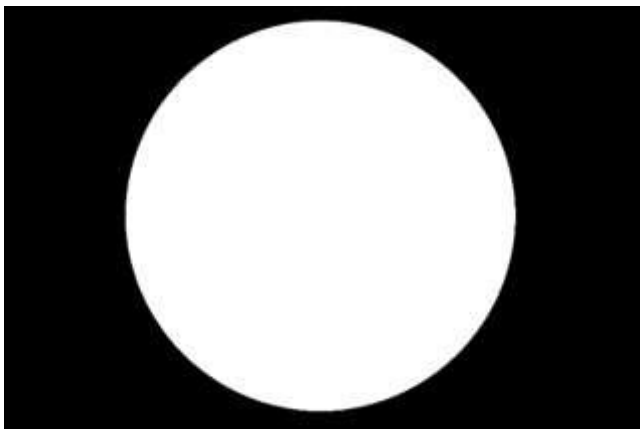
Para este proceso, se procedió a extraer el fondo de imagen original y sustituirlo por una máscara de fondo completamente negra, sin ruido. La técnica aplicada consiste en crear una matriz de 8 x 8 píxeles que recorre toda la imagen, en cada iteración se obtiene la media y la normalización del valor de píxeles de los canales RGB. Si este resultado es mayor a 10, se considera como zona que pertenece a la retina, para luego ser reemplazado por

píxeles blancos; caso contrario, si el resultado es menor a 10, se considera como zona que pertenece al fondo, y es reemplazado por píxeles negros. De esta manera se crea una máscara de color blanco y negro.

Finalmente, con el *operador and*, al unir la máscara con la imagen original, se obtiene una imagen de la retina con el fondo libre de ruido y completamente negro.

Figura 11.

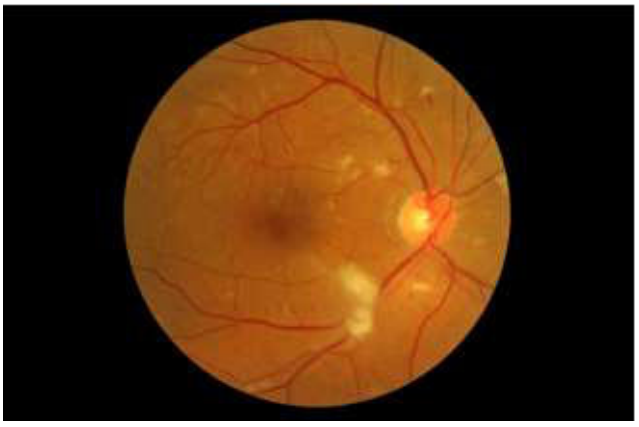
Máscara de fondo de la retina



Fuente: Elaboración propia

Figura 12.

Retina pre-procesada sin ruido



Fuente: Elaboración propia

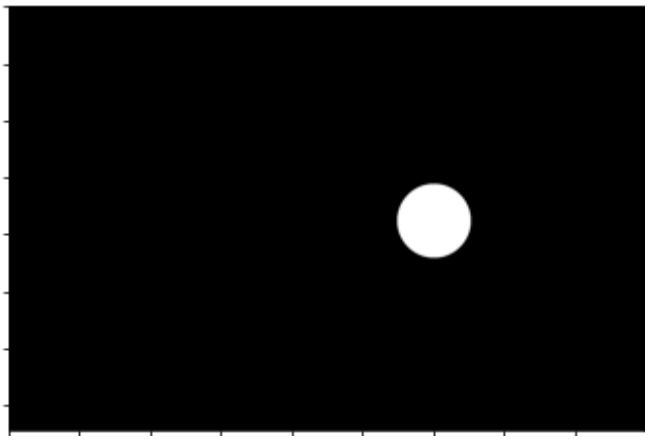
3.5.2.2 DETECCIÓN Y LIMPIEZA DEL DISCO ÓPTICO

Detectar el disco óptico es importante, debido a su estructura de forma redondeada y naturaleza brillosa y amarillenta que durante el procesamiento podría confundirse con un exudado, generando ruido en la detección de exudados.

En el procedimiento para detectar el disco óptico se enfocó en ubicar la zona más brillosa de la retina. Se empezó con la conversión de la imagen original (RGB) a una imagen binaria (blanco y negro), luego se obtuvo el píxel o píxeles de máximo valor para extraer la media de los valores. Por otro lado, se procedió a crear una máscara negra del tamaño de la imagen, donde ubicamos la coordenada del píxel extraído. Luego se creó una máscara blanca en la zona del disco óptico, como se visualiza en la figura 9.

Figura 13.

Máscara del disco óptico

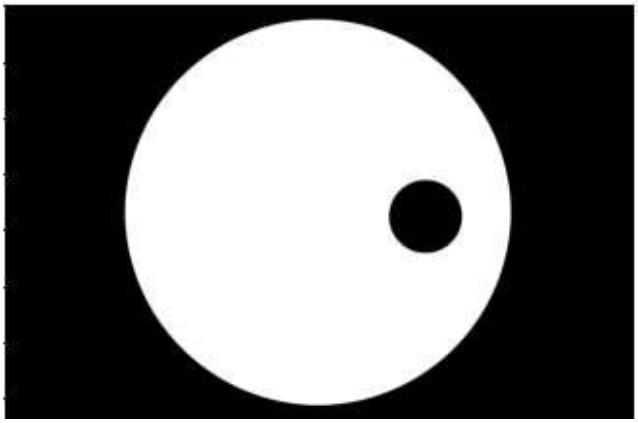


Fuente: Elaboración propia

Seguidamente se aplicó la función *xor* entre la figura 7 y 9 para obtener la máscara del fondo de la retina y la máscara del disco unido. Dando como resultado la máscara mostrada en la Fig. 14.

Figura 14.

Máscara del disco óptico y fondo

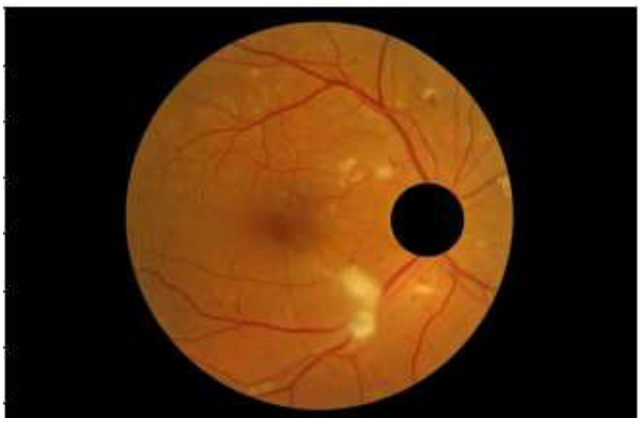


Fuente: Elaboración propia

Finalmente, para obtener la imagen de la retina con el disco óptico detectado y extraído, se aplicó la función *and* entre la máscara del disco óptico final (figura 14) y la imagen pre-procesada de la retina (figura 10).

Figura 15.

Retina con el disco óptico detectado



Fuente: Elaboración propia

3.5.3 PROCESAMIENTO DE IMÁGENES

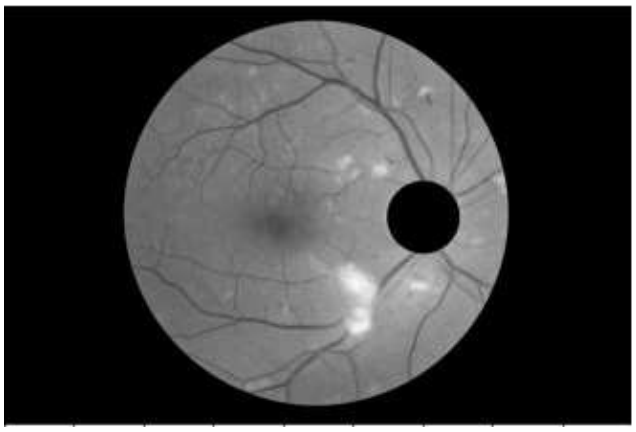
En el procesamiento de imágenes, detectamos características propias de la retinopatía. Como: los exudados (zonas brillosas), microaneurismas y hemorragias (zonas rojas y oscuras).

3.5.3.1 DETECCIÓN DE EXUDADOS

Los exudados en la retina se identifican como las zonas brillantes, de color amarillento. Esta característica se puede obtener del espacio de color LAB, en el canal L de Luminosidad. Por lo tanto, para la detección de exudados se seleccionó el canal L.

Figura 16.

Canal L del espacio de color LAB

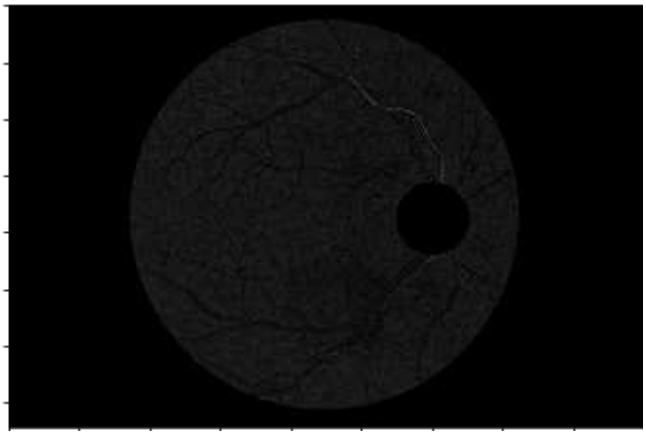


Fuente: Elaboración propia

Almacenamos en dos variables temporales la imagen del canal L, se utilizó una operación morfológica de apertura (Erosión y dilatación), luego se restó la imagen temporal con el resultado de apertura morfológica.

Figura 17.

Resta del canal L y la apertura morfológica

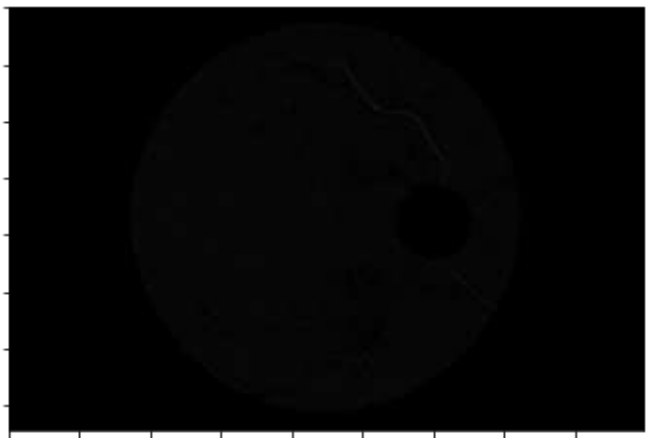


Fuente: Elaboración propia

También se realizó un cierre morfológico para restar la imagen temporal con el resultado, mostrado en la Figura 18.

Figura 18.

Resta del canal L y el cierre morfológico



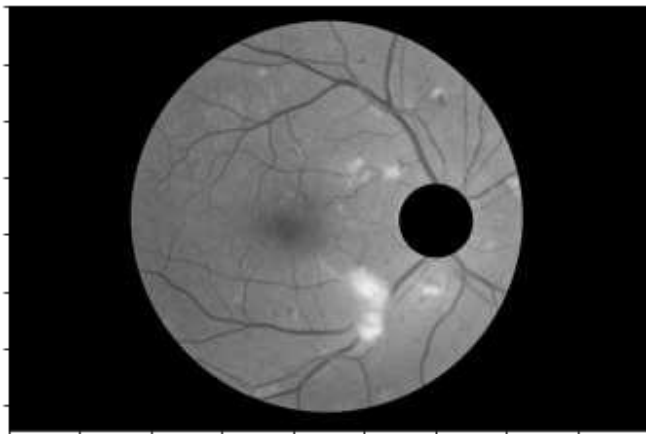
Fuente: Elaboración propia

Seguidamente se procedió a la suma del canal L (Luminosidad) y la imagen resultante de la apertura morfológica.

Por otro lado, también restamos la imagen del canal L y la imagen de resultado del cierre morfológico, almacenado en la variable del canal L.

Figura 19.

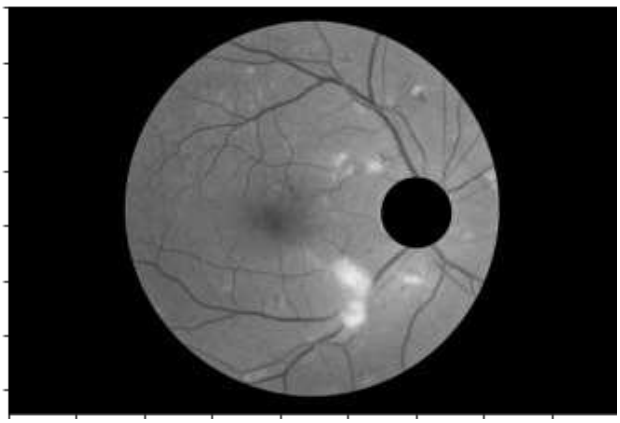
Suma del canal L y la apertura morfológica



Fuente: Elaboración propia en python

Figura 20.

Resta del canal L y la suma morfológica

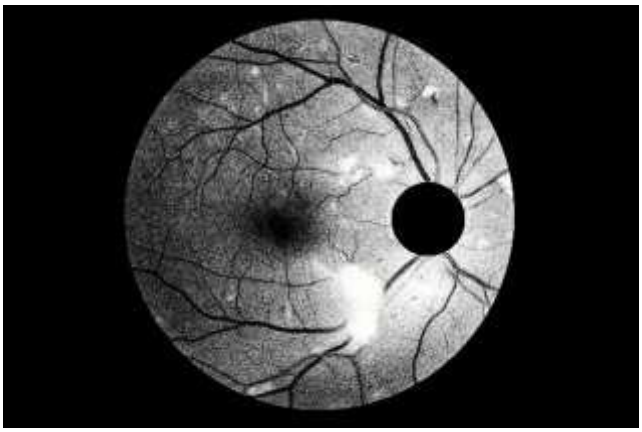


Fuente: Elaboración propia

A la actual variable del canal L se aplica la Ecualización de Histograma para mejorar el contraste de la imagen, luego se aplica un threshold para la detección final del exudado (imagen 22)

Figura 21.

Ecualización de histograma luego de operaciones morfológicas



Fuente: Elaboración propia

Figura 22.

Exudados detectados



Fuente: Elaboración propia

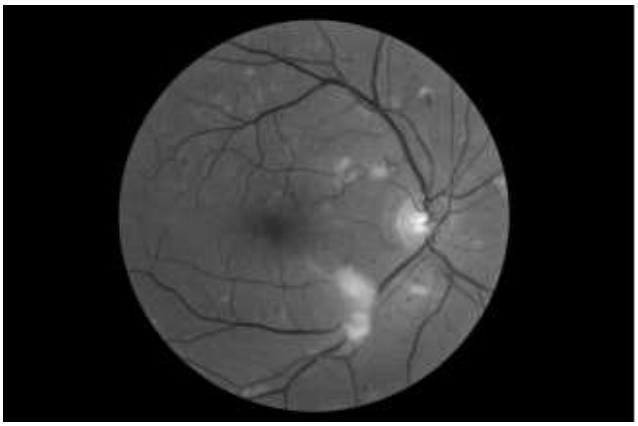
3.5.3.2 MICROANEURISMAS Y HEMORRAGIAS

Para la detección de microaneurismas y hemorragias destacamos las zonas oscuras de la retina, donde los vasos sanguíneos son parte del ruido.

Como primer paso se seleccionó el canal verde de la imagen RGB (Imagen 10), este canal es donde se observa con realce las zonas oscuras. En la Figura 23 se muestra el canal verde extraído de la imagen original.

Figura 23.

Canal verde de imagen de retina



Fuente: Elaboración propia

Con el objeto de mejorar el contraste de la imagen, aplicamos la técnica de Ecualización de Histogramas al canal verde. El resultado lo observamos en la Figura 24.

Figura 24.

Ecualización de histograma en el canal verde

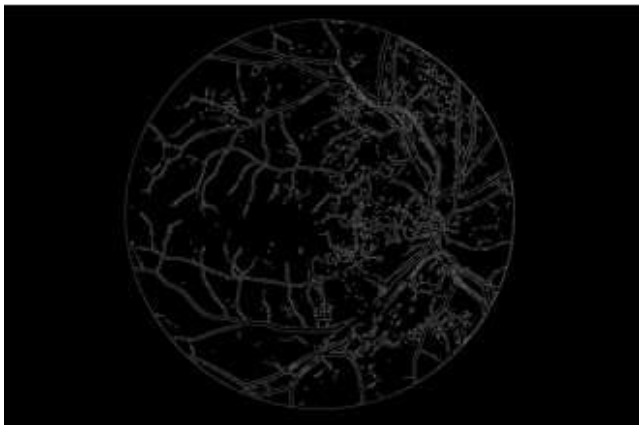


Fuente: Elaboración propia

Se aplicó un filtro de suavizado a la imagen de canal verde multiplicada por un valor escalar. Luego usamos la técnica del detector de bordes, *Canny* como se observa en la Figura 25.

Figura 25.

Detección de bordes usando Canny

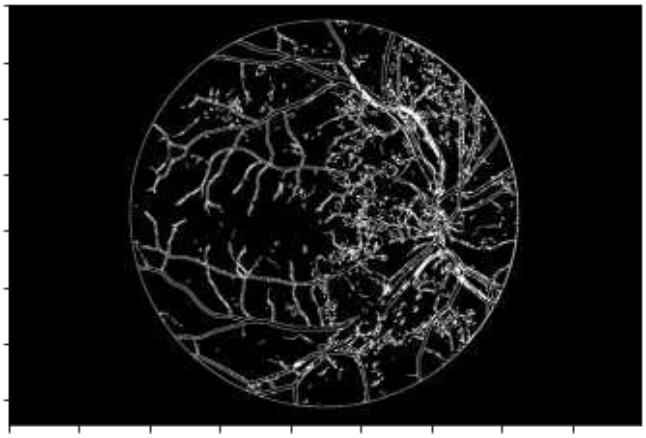


Fuente: Elaboración propia

Luego de suavizar el canal verde y detectar bordes, hallamos la suma de ambas imágenes, para realizar una operación morfológica de cierre (dilatar y erosionar) con el objetivo de resaltar los bordes, cuyo resultado se muestra en la Figura 26.

Figura 26.

Operación morfológica en bordes detectados

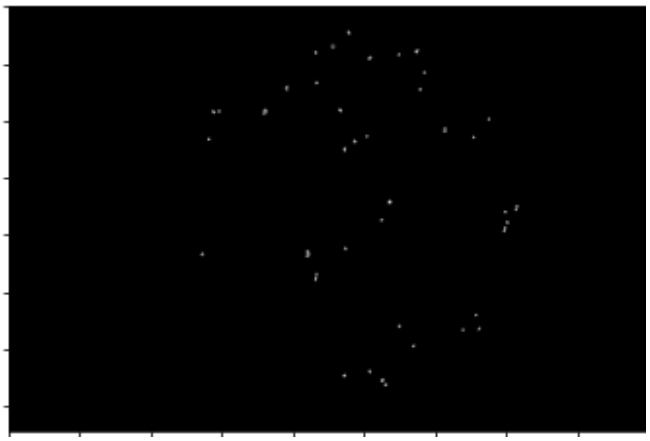


Fuente: Elaboración propia

Para la eliminación de vasos sanguíneos, aplicamos el algoritmo de *connected components with stats* a la imagen de bordes detectados. De cada componente u objeto obtenemos al ancho y alto para calcular la circularidad y el área para seleccionar únicamente los componentes de área mayor a 0.5 multiplicado por el alto y ancho y mayor a 50. Y a la vez con una circularidad en el rango mayor a 0.5 y menor a 1.5. El resultado de la imagen con los objetos que cumplen la condición se muestran en la Figura 27.

Figura 27.

Eliminación de vasos sanguíneos



Fuente: Elaboración propia

A la figura 30 ecualizada se aplicó la técnica threshold y bitwise not, a fin de resaltar la zona de microaneurismas, a este resultado y la figura 26 aplicamos la función *and* para descartar ruido que no pertenece a la zona de microaneurismas.

Figura 28.

Microaneurismas detectados



Fuente: Elaboración propia

3.5.4 EXTRACCIÓN DE CARACTERÍSTICAS

3.5.4.1 ÁREA Y NÚMERO DE CARACTERÍSTICAS

En este estudio se consideró extraer el área y número de exudados y microaneurismas a partir de las imágenes procesadas. Por lo tanto, las características que se usan para modelar son:

- Área de microaneurismas y hemorragias: se halló
- Área de exudados
- Número de microaneurismas y hemorragias
- Número de exudados

El área se halló realizando una suma de los píxeles blancos que representan el exudado, microaneurisma o hemorragia. Respecto al número de exudados, microaneurismas o hemorragias se obtuvo contando la cantidad de objetos blancos encontrados en la imagen de características detectadas.

3.5.5 ANÁLISIS DE CARACTERÍSTICAS

Analizamos la correlación estadística, donde nos muestra la relación mutua entre dos variables, cada variable en relación al Grado de la Retinopatía Diabética, con valores que oscilan entre -1 y 1. Se observó que la variable *Área de los microaneurismas y hemorragias* es más cercano a 1, esto indica que es la variable con mayor correlación al grado de Retinopatía Diabética, mientras que la variable *Número de exudados* indica menor correlación.

Tabla 7.

Correlación de las características

DATOS	ÁREA MICROANEURISMAS	NÚMERO MICROANEURISMAS	ÁREA EXUDADOS	ÁREA EXUDADOS	GRADO DE RETINOPATÍA
ÁREA MICROANEURISMAS	1.00	0.82	0.80	0.66	0.89
NÚMERO MICROANEURISMAS	0.82	1.00	0.70	0.64	0.87
ÁREA EXUDADOS	0.79	0.70	1.0	0.59	0.84
ÁREA EXUDADOS	0.66	0.64	0.59	1.00	0.68

Fuente: Elaboración propia

3.5.6 DIVISIÓN EN DATOS EN ENTRENAMIENTO Y TEST

En la preparación de los datos para ingresar a los modelos de Machine Learning, es necesario dividir los datos en grupos (entrenamiento y test). En este estudio asignaremos el 80 % para el entrenamiento y el 20 % para el test. Contamos con cuatro clases que deberá ser distribuido con la misma proporcionalidad en ambos grupos (entrenamiento y test). Esta distribución proporcionalidad se resuelvió aplicando *Stratified Shuffle Split* (Validación cruzada ShuffleSplit estratificado).

Stratified Shuffle Split. Proporciona índices de entrenamiento y prueba para dividir datos en conjuntos de entrenamiento y prueba. Este objeto es una combinación de *StratifiedKFold* y *ShuffleSplit*, que devuelve pliegues aleatorios estratificados. Los pliegues se realizan conservando el porcentaje de muestras de cada clase.

Proporcionalidad de la data general y los datos de entrenamiento y test debe ser el mismo.

Tabla 8.

Proporcionalidad de datos en entrenamiento y test

PROPORCIONALIDAD			
DATOS	**ENTRENAMIENTO**	**TEST**	**TOTAL**
NORMAL	45.5000%	45.5000%	45.5000%
LEVE	12.7500%	12.7500%	12.7500%
MODERADO	20.5833%	20.5833%	20.5833%
SEVERO	21.1667%	21.1667%	21.1667%
TOTAL	100%	100%	100%

Nota: Igualdad en la proporcionalidad en los datos de entrenamiento y test usando Stratified Shuffle Split.
Fuente: Elaboración propia

Tabla 9.

División de datos en entrenamiento y test

DATOS	**ENTRENAMIENTO 80%**	**TEST 20%**	**TOTAL 100%**
NORMAL	437	109	546
LEVE	122	31	153
MODERADO	198	49	247
SEVERO	203	51	254
TOTAL	960	240	1200

Fuente: Elaboración propia

3.5.7 NORMALIZACIÓN DE DATOS

Los datos extraídos de las características contienen valores en diferentes rangos, para normalizar y uniformizar el procesamiento es necesario transformar características escaladas a un rango determinado. Dicha normalización se realizó usando la función StandarScaler que nos devuelve valores en un rango entre – 1 y 1; los valores se muestran en la Tabla 10.

La estandarizamos StandasScaler se obtiene eliminando la media y escalando la varianza. La estandarización se calcula como:

$$z = (x - u)/s$$

Donde u es la media de los datos de entrenamiento, y s es la desviación estándar de los datos de entrenamiento.

Tabla 10.

Datos normalizados usando StandarScaler

INDEX	ÁREA MICROANEURISMAS	NÚMERO MICROANEURISMAS	ÁREA EXUDADOS	ÁREA EXUDADOS
0	-0.746285	-0.946587	-0.743803	-318316
1	-0.736463	-0.918810	-0.728221	-0.942465
2	0.201088	0.331153	-0.278926	0.805153
3	-0.439126	1.025577	0.010649	-0.068656
4	1.862786	1.497786	1.424110	1.678962
5	0.358240	0.553369	0.222960	1.179642
6	-0.735570	-0.807702	-0.728221	-0.942465
7	0.443066	0.831139	-0.283471	-0.193486
8	-0.008745	0.747808	1.179984	1.054812
9	0.850231	0.553369	0.343724	1.179642

Fuente: Elaboración propia

IV. RESULTADOS

El resultado obtenido corresponde a tres Algoritmos Machine Learning de Clasificación, estos algoritmos. Estos algoritmos son:

- K-Nearest Neighbor (KNN)
- Support Vector Machine (SVM)
- Multi-layer Perceptron neural network

Este proceso de modelamiento, se dividió en 2 etapas:

- Entrenar el modelo, utilizando la data de entrenamiento, donde aplicaremos el algoritmo seleccionado.

- Validar el modelo, dado que ya tenemos el modelo entrenado. Esto lo realizamos con la data de test. Donde se procedió a aplicar el algoritmo y evaluar los resultados.

  En el presente trabajo se usó la métrica de clasificación **F1-Score,** siendo esta métrica la más ideal para evaluar algoritmos de multiclasificación de data no balanceada. También tenemos la métrica de evaluación *Macro average* que nos muestra el promedio de cada métrica en todas las clases y la métrica de evaluación Weighted average nos muestra el promedio de cada métrica en todas las clases según su proporcionalidad.

4.1 MODELAMIENTO CON ALGORIMOS MACHINE LEARNING

4.1.1 K-NEAREST NEIGHBOR

La clasificación K-nearest neighbor evalúa los vecinos más cercanos al valor que se desea clasificar, a fin de analizar la clasificación según esta cantidad de estos.

En la presente investigación se aplicó KNN haciendo uso de la librería Scikit-learn que implementa el aprendizaje basado en K vecinos más cercanos de cada punto de consulta, donde k es un valor entero establecido por el usuario, en esta investigación se asignó a K el valor constante 23.

4.1.1.1 PREDICCIÓN DEL MODELO K-NEAREST NEIGHBOR

En la predicción de la clasificación de los datos de test, usamos la matriz de confusión del modelo K-Nearest Neighbor para analizar los datos clasificados con la siguiente interpretación:

- Se muestra la clasificación **normal,** donde 109 datos se clasificaron como Verdadero Positivo (VP), 123 datos como Verdaderos Negativos (VN), 0 datos como Falsos Negativos (FN), 8 datos como Falsos Positivos (FP).

- Se muestra la clasificación **leve,** donde 25 datos se clasificaron como Verdadero Positivo (VP), 206 datos como Verdaderos Negativos (VN), 6 datos como Falsos Negativos (FN), 3 datos como Falsos Positivos (FP).

- Se muestra la clasificación **moderada,** donde 46 datos se clasificaron como Verdadero Positivo (VP), 189 datos como

Verdaderos Negativos (VN), 3 datos como Falsos Negativos (FN), 2 datos como Falsos Positivos (FP).

- En la cuarta columna muestra la clasificación **severa,** donde 47 datos se clasificaron como Verdadero Positivo (VP), 80 datos como Verdaderos Negativos (VN), 4 datos como Falsos Negativos (FN), 0 datos como Falsos Positivos (FP).

Tabla 11.

Matriz de confusión del algoritmo KNN

	LEVE	MODERADO	NORMAL	SEVERO
LEVE	25	0	6	0
MODERADO	2	46	1	0
NORMAL	0	0	109	0
SEVERO	1	2	1	47

Nota: Matriz de confusión del algoritmo KNN, K=23
Fuente: Elaboración propia

Dentro de las métricas de evaluación se obtuvo una exactitud del 95%. En la tabla 12 observamos que la precisión más baja corresponde a la clase “leve” en comparación de las demás, en cuando al Recall se muestra mejor resultado en la clase “Normal”. Además, en cuanto a la métrica F1-Score se obtuvieron mejores resultados en la clase “Normal”.

Tabla 12.

Reporte de métricas de evaluación del algoritmo KNN

DATOS	PRECISION	RECALL	F1-SCORE
NORMAL	93	100	96
LEVE	89	81	85
MODERADO	96	94	95
SEVERO	100	92	96
MACRO AVG	95	92	93
WEIGHTED AVG	95	95	95

Fuente: Elaboración propia

4.1.2 RED NEURONAL PERCEPTRÓN MULTICAPA (MLP NN)

El clasificador de perceptrón multicapa optimiza la función de pérdida de registros utilizando LBFGS.

MLPClassifier admite la clasificación de clases múltiples aplicando la Función Tangencial hiperbólica como función de activación.

$$\tanh x = \frac{\operatorname{senh} x}{\cosh x} = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

El resultado es un vector que contiene las probabilidades de que muestras pertenecen a cada clase. La salida es la clase con mayor probabilidad.

4.1.2.1 PREDICCIÓN DEL MODELO MLP NEURAL NETWORK

En la predicción de la clasificación de los datos de test, usamos la matriz de confusión del modelo MLP para analizar los datos clasificados con la siguiente interpretación:

- Se muestra la clasificación **normal,** donde 108 datos se clasificaron como Verdadero Positivo (VP), 125 datos como Verdaderos Negativos (VN), 0 datos como Falsos Negativos (FN), 6 datos como Falsos Positivos (FP).

- Se muestra la clasificación **leve,** donde 28 datos se clasificaron como Verdadero Positivo (VP), 206 datos como Verdaderos Negativos (VN), 3 datos como Falsos Negativos (FN), 2 datos como Falsos Positivos (FP).

- Se muestra la clasificación **moderada,** donde 46 datos se clasificaron como Verdadero Positivo (VP), 189 datos como

Verdaderos Negativos (VN), 3 datos como Falsos Negativos (FN), 1 dato como Falsos Positivos (FP).

- En la cuarta columna muestra la clasificación **severa,** donde 48 datos se clasificaron como Verdadero Positivo (VP), 188 datos como Verdaderos Negativos (VN), 3 datos como Falsos Negativos (FN), 0 datos como Falsos Positivos (FP).

Tabla 13.

Matriz de confusión del modelo MLP

	LEVE	MODERADO	NORMAL	SEVERO
LEVE	28	0	3	0
MODERADO	2	46	1	0
NORMAL	0	0	108	0
SEVERO	0	1	2	48

Fuente: Elaboración propia

Dentro de las métricas de evaluación se obtuvo una exactitud del 96%. En la tabla 14 observamos que la precisión más baja corresponde a la clase “Leve” en comparación de las demás, en cuando al Recall se muestra mejor resultado en la clase “Normal”. Además, en cuanto a la métrica F1-Score se obtuvieron mejores resultados en la clase “Normal”.

Tabla 14.

Reporte de métricas de evaluación del modelo MLP

DATOS	PRECISION	RECALL	F1-SCORE
NORMAL	95	99	97
LEVE	93	90	92
MODERADO	98	94	96
SEVERO	98	94	96
MACRO AVG	96	94	95
WEIGHTED AVG	96	96	96

Fuente: Elaboración propia

4.1.3 SUPPORT VECTOR MACHINE

El modelo de clasificación SVM, es uno de los mejores en la clasificación de imágenes. Trabaja muy bien con datasets relativamente pequeños.

En la presenta investigación se aplicó Support Vector Classification multiclass usando sckit-learn library basado en LIBSVM con un enfoque de "uno contra uno", se utilizó el Kernel RBF debido a la complejidad del modelo y aplicación de 4 dimensiones, considerando los parámetros C y gamma.

4.1.3.1 PREDICCIÓN DEL MODELO SVM

Tabla 15.

Matriz de confusión del algoritmo SVM

	LEVE	MODERADO	NORMAL	SEVERO
LEVE	27	0	4	0
MODERADO	2	46	1	0
NORMAL	0	0	109	0
SEVERO	0	2	2	47

Fuente: Elaboración propia

Dentro de las métricas de evaluación se obtuvo una exactitud del 95%. En la tabla 16 observamos que la precisión más baja corresponde a la clase "Leve" en comparación de las demás, en cuando al Recall se muestra mejor resultado en la clase "Normal". Además, en cuanto a la métrica F1-Score se obtuvieron mejores resultados en la clase "Normal".

Tabla 16.

Reporte de métricas de evaluación del modelo SVM

DATOS	PRECISION	RECALL	F1-SCORE
NORMAL	95	100	97
LEVE	90	87	89
MODERADO	96	94	95
SEVERO	100	92	96
MACRO AVG	95	93	94
WEIGHTED AVG	95	95	95

Fuente: Elaboración propia

4.1.3.2 PREDICCIÓN DEL MODELO SVM APLICANDO GRIDSEARCH CV

Hacemos uso de GridSearchCV para buscar y probar múltiples hiperparámetros a la vez, realizando todas las combinaciones posibles. Encontrar los mejores valores para gamma, y potencialmente el parámetro de regularización C. Se observó que los valores ideales para *C* es 1, y el valor ideal para *gamma* es 5, mostrando mejores resultados.

En la predicción de la clasificación de los datos de test, usamos la matriz de confusión del modelo SVM para analizar los datos clasificados con la siguiente interpretación:

- Se muestra la clasificación **normal,** donde 108 datos se clasificaron como Verdadero Positivo (VP), 129 datos como Verdaderos Negativos (VN), 1 dato como Falsos Negativos (FN, 2 datos como Falsos Positivos (FP).

- Se muestra la clasificación **leve,** donde 30 datos se clasificaron como Verdadero Positivo (VP), 208 datos como Verdaderos Negativos (VN), 1 datos como Falsos Negativos (FN), 1 datos como Falsos Positivos (FP).

- Se muestra la clasificación **moderada,** donde 47 datos se clasificaron como Verdadero Positivo (VP), 188 datos como Verdaderos Negativos (VN), 2 datos como Falsos Negativos (FN), 3 datos como Falsos Positivos (FP).

- En la cuarta columna muestra la clasificación **severa,** donde 47 datos se clasificaron como Verdadero Positivo (VP), 187 datos como Verdaderos Negativos (VN), 1 datos como Falsos Negativos (FN), 2 datos como Falsos Positivos (FP).

Tabla 17.

Matriz de confusión del algoritmo de SVM usando GridSearch

	LEVE	MODERADO	NORMAL	SEVERO
LEVE	30	0	1	0
MODERADO	1	47	0	1
NORMAL	0	0	108	1
SEVERO	0	3	1	47

Fuente: Elaboración propia

Dentro de las métricas de evaluación se obtuvo una exactitud del 97%. En la tabla 18 observamos que la precisión más baja corresponde a la clase "Moderado" en comparación de las demás, en cuando al Recall se muestra mejor resultado en la clase "Normal". Además, en cuanto a la métrica F1-Score se obtuvieron mejores resultados en la clase "Normal".

Tabla 18.

Métricas de evaluación del modelo SVM usando GridSearch

DATOS	PRECISION	RECALL	F1-SCORE
NORMAL	98	99	99
LEVE	93	90	92
MODERADO	94	96	95
SEVERO	96	92	94
MACRO AVG	96	96	96
WEIGHTED AVG	97	97	97

Fuente: Elaboración propia

V. DISCUSIÓN DE RESULTADOS

Este estudio analiza datos no balanceados, lo que podría generar mejores resultados de precisión en la clase con mayor número de datos y menor en las clases de menor cantidad de data. En este caso el resultado de precisión total podría dar un resultado no real.

En esta evaluación se analiza F1 Score, esta métrica predice bien en la multiclasificación de data no balanceada debido a que se evalúa métricas, Precision y Recall a la vez,

A continuación, observamos el resultado de la métrica F1 – Score para cada modelo aplicado.

Tabla 19.

Reporte de métricas de evaluación F1-Score

F1 - SCORE			
	KNN	**MLP**	**SVM**
NORMAL	96 %	97 %	99 %
LEVE	85 %	92 %	97 %
MODERADO	95 %	96 %	95 %
SEVERO	96 %	96 %	95 %
WEIGHTED AVG	95 %	96 %	97 %

Nota: Reporte de métrica F1-Score en 3 algoritmos de clasificación según su severidad
Fuente: Elaboración propia

VI. CONCLUSIONES

- Se observa que en este estudio el algoritmo que visualiza mejores resultados en comparación al resto, es Support Vector Machine usando GSearch. La métrica analizada es el F1- Score con un resultado en la clasificación normal de 99%, leve con un 97%, moderado con un 95% y severo con un 94%.

- En la detección de características, se desarrolló el procesamiento de imágenes aplicando técnicas de visión computacional. Las patologías clínicas que se detectaron fueron los exudados, microaneurismas y hemorragias. Las características que se obtuvo luego de procesar las imágenes es el número de exudados, el área de exudados, el número y área de hemorragias y microaneurismas.

- Se desarrolló técnicas de visión computacional en el pre – procesamiento de imágenes, que consta de dos etapas. El primero es la detección del fondo de la retina para eliminar ruido. La segunda etapa es la detección y exclusión del disco óptico, debido a su característica brillosa que puede ser confundido como exudado.

VII. RECOMENDACIONES

- Implementar la telemedicina como alternativa en el diagnóstico de algunas enfermedades, optimizando recursos como el tiempo y personal médico presencial, pues en algunos lugares no hay personal suficiente para atender la demanda de pacientes que padecen la enfermedad.

- Contar con equipos oftalmológicos especializados que capturen y almacenen imágenes de la retina en la Región Moquegua, para posibilitar un posterior estudio con el uso de estas imágenes, y hacer posible el diagnóstico en pacientes de nuestra región, sin necesidad de ser transferidos a otras ciudades.

- Incrementar la base de datos para un mejor entrenamiento y evaluación en los algoritmos de Machine Learning, y evitar data desbalanceada.

- Mejorar e investigar nuevas técnicas que se desarrollen a fin de mejorar el procesamiento de imágenes, reducir ruido y optimizar la detección de características.

REFERENCIAS

Asti Herliana, Toni Arifin, Sara Susanti y Agung Baitul Hitmah. (2018). Selección de características de la enfermedad retinopatía diabética usando partículas de optimización Swarm y Redes Neuronales.

Ajay Kulkarni, Feras A. Batarseh, in Data Democracy. (2020). Foundations of data imbalance and solutions for a data democracy

Bealdung. (9 de diciembre, 2020). Multiclass Classification using Support Vector Machine.

D. Aliseda, Berástegui, Hospital Navarra (2008). Diabetic Retinopathy.

Desarrolladores de Scikit-Learn (2020). Neural Network Models. https://scikit-learn.org/stable/modules/neural_networks_supervised.html

Edward TETOUHE KILIMOU. (19 de Octubre, 2019). Image Processing: Segmentation and objects counting with python and OpenCV.

Enrique Alegre, Gonzalo Pajares y Arturo de la Escalera. (Junio, 2016). Conceptos y métodos en visión por computador.

Enrique V. Carrera, Andrés Gonzalez, Ricardo Carrera. (2015). Detección automática de la retinopatía diabética usando SVM.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, Édouard Duchesnay. (2011). Scikit-learn: Machine Learning in Python

Jose Martinez Heras. (09 de Octubre del 2020). Precision, Recall, F1, Accuracy of classification.

Johanna Orellana Alvear. (12-16 noviembre 2018). Decision Tree and Random Forest

Karan Bhatia, Shikhar Arora, Ravi Tomar. (2016). Diagnóstico de la retinopatía diabética usando algoritmos de Clasificación de Machine Learning.

Khalid K. Al-jabery, Donald C. Wunsch II. (2000). Computational learning approaches to data analytics in biomedical applications, 2020 data analysis and machine learning tools in MATLAB and Python.

Konica Minolta Sensing Americans. (2020). Espacio de color CIE L*A*B

Konica Minolta (2020). Entendiendo El Espacio de Color CIE L*A*B*. https://sensing.konicaminolta.us/mx/blog/entendiendo-el-espacio-de-color-cie-lab/

MayoClinic. (2018). Retinopatía diabética. https://www.mayoclinic.org/es-es/diseases-conditions/diabetic-retinopathy/symptoms-causes/

Nickson Joram. (1 de Enero, 2020), Morphological operations in image processing.

Organización Mundial de la Salud, (2016). Informe mundial sobre la diabetes.

Omer, Utku. (2018). Diagnóstico de la Retinopatía Diabética usando Procesamiento de imágenes y Redes Convolucionales.

Olivier Grisel, Guillaume Lemaitre, Jérémie du Boisberranger and Chiara Marmo. Scikit-Learn Consortium at Inria Foundation

Omer, Utku. (2018). Diagnóstico de la Retinopatía Diabética usando Procesamiento de imágenes y Redes Convolucionales

Open Source Computer Vision. (2021). Histogram Equalization. https://docs.opencv.org/master/d5/daf/tutorial_py_histogram_equalization.html

Russell, Stuart; Norvig, Peter. (2009). *Inteligencia Artificial: Un enfoque moderno* (3rd edición). p. 229.

Richard Szeliski. (5 de Agosto, 2020) Computer Vison: Algorithms and applications.

R. C. Gonzalez and R. E. (2018). Woods, Digital Image Processing, Third Edition

Sarang Narkhede. (9 de Mayo, 2018). Understanding Confusion Matrix.

Scikit-learn developers (2020), Neural Network Models. (https://scikit-learn.org/stable/modules/neural_networks_supervised.html)

Shailesh Kumar, Basant Kumar. (2018). Detección de la Retinopatía Diabética por la extracción del área y número de microaneurismas de imágenes fundus,

Sohini Roy chow dhury, 2013 Análisis de la retinopatía diabética usando Machine Learning.

Sohini Roy Chow Dhury. (2013). DREAM: Análisis de la retinopatía diabética usando Machine Learning.

Shailesh Kumar, Basant Kumar, (2018). Detección de la retinopatía diabética por la extracción del área y número de microaneurismas de imágenes fundus.

Tim Morris. (2004). Computer Vision and Image Processing. Palgrave Macmillan.

Tutor de programación. (2017). OpenCV y uso de funciones Split y Merge. http://acodigo.blogspot.com/2018/01/opencv-uso-de-las-funciones-split-y.html