



UNIVERSIDAD TÉCNICA DE COTOPAXI
DIRECCIÓN DE POSGRADO

**MAESTRÍA EN ELECTRICIDAD MENCIÓN SISTEMAS ELÉCTRICOS
DE POTENCIA**

MODALIDAD: PROYECTO DE INVESTIGACIÓN

Título:

**Identificación De Fallas En Los Aisladores De Una Línea De
Transmisión Mediante Visión Artificial.**

Trabajo de titulación previo a la obtención del título de Magíster en Electricidad
mención Sistemas Eléctricos de Potencia.

Autor:

Astudillo Cortez Vicente Paul

Tutor:

MSc. Freire Martínez Luigi Orlando

LATACUNGA-ECUADOR

2022

AVAL DEL TUTOR

En mi calidad de Tutor del Trabajo de Titulación “**Identificación de fallas en los aisladores de una línea de transmisión mediante visión artificial**”, presentado por Astudillo Cortez Vicente Paul, para optar por el título de magíster en Electricidad mención Sistemas Eléctricos de Potencia.

CERTIFICO

Que dicho trabajo de investigación ha sido revisado en todas sus partes y se considera que reúne los requisitos y méritos suficientes para ser sometido a la presentación para la valoración por parte del Tribunal de Lectores que se designe y su exposición y defensa pública.

Latacunga, Mayo 27 del 2022



.....
MSc. Freire Martínez Luigi Orlando

CC: 050252958-9

AVAL DEL TRIBUNAL

El trabajo de Titulación: “**Identificación de fallas en los aisladores de una línea de transmisión mediante visión artificial**”, ha sido revisado, aprobado y autorizado su impresión y empastado, previo a la obtención del título de Magíster en Electricidad Mención Sistemas Eléctricos de Potencia, el presente trabajo reúne los requisitos de fondo y forma para que el estudiante pueda presentarse a la exposición y defensa.

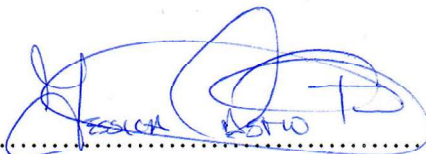
Latacunga, Mayo 27 del 2022



.....
MSc. SALAZAR ACHIG EDGAR ROBERTO

CC: 050284761-9

PRESIDENTE DEL TRIBUNAL



.....
MSc. CASTILLO FIALLOS JESSICA NATALY

CC: 060459021-6

LECTOR 2



.....
MSc. CORRALES BASTIDAS BYRON PAUL

CC: 050234776-8

LECTOR 3

DEDICATORIA

Dedico este nuevo logro en mi vida, a mi madre que con su lucha me demostró que todo se puede lograr con perseverancia y esfuerzo, a mi padre que con sus palabras, apoyo y experiencia me ha ayudado a siempre mantener la frente en alto. A mis hermanas que con sus diferentes muestras de cariño me han motivado a conseguir nuevas metas en la vida, a mi abuelita que con sus consejos siempre ha sido un ejemplo de vida, a mi abuelito Jaime que desde el cielo siempre me cuida y me guía por el arduo camino conocido como vida.

PAUL

AGRADECIMIENTO

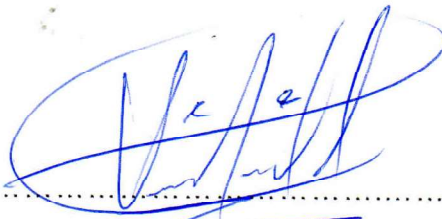
Sobran las palabras para agradecer a mi familia la cual a lo largo de mi vida me han guiado, aconsejado y formado para poder alcanzar mis metas, superar mis límites y culminar una nueva etapa de mi vida.

PAUL

RESPONSABILIDAD DE AUTORÍA

Quien suscribe, declara que asume la autoría de los contenidos y los resultados obtenidos en el presente trabajo de titulación.

Latacunga, Mayo 27 del 2022

A handwritten signature in blue ink, appearing to be 'Vicente Paul Astudillo Cortez', written over a horizontal dotted line.

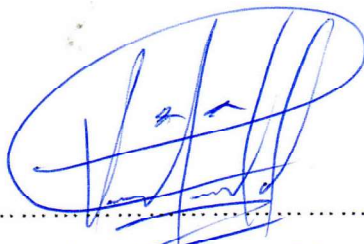
Astudillo Cortez Vicente Paul

CC. 050288778-9

RENUNCIA DE DERECHOS

Quien suscribe, cede los derechos de autoría intelectual total y/o parcial del presente proyecto de titulación a la Universidad Técnica de Cotopaxi.

Latacunga, Mayo 27 del 2022



Astudillo Cortez Vicente Paul

CC. 050288778-9

AVAL DEL PRESIDENTE DEL TRIBUNAL

Quien suscribe, declara que el presente Trabajo de Titulación: “**Identificación de fallas en los aisladores de una línea de transmisión mediante visión artificial**”, contiene las correcciones a las observaciones realizadas por los lectores en sesión científica del tribunal.

Latacunga, Mayo 27 del 2022



.....
MSc. Salazar Achig Edgar Roberto

CC: 050284761-9

UNIVERSIDAD TÉCNICA DE COTOPAXI
DIRECCIÓN DE POSGRADOS
MAESTRÍA EN ELECTRICIDAD
MENCIÓN SISTEMAS ELÉCTRICOS DE POTENCIA

Título: Identificación de fallas en los aisladores de una línea de transmisión mediante visión artificial.

Autor: Astudillo Cortez Vicente Paul

Tutor: MSc. Freire Martínez Luigi Orlando

RESUMEN

La presente investigación se basa en la identificación superficial de fallas de aisladores de porcelana de la línea de transmisión de 13.8 kV de las ciudades de Latacunga y Salcedo mediante visión artificial para conocer el estado de dichos elementos de una manera fácil y sencilla. Se inicio el desarrollo con una base de datos de 3000 imágenes de aisladores divididas en tres estados: sucios/flameados, rotos y en buen estado. Con dichas imágenes se procedió a entrenar a Yolo como algoritmo de identificación el cual se encuentra basado en redes neuronales, la identificación funciona tanto en tiempo real como en imágenes y videos pregrabados. Para que su manejo sea lo más sencillo posible, se realizó una interfaz gráfica con la finalidad de ser usado por un solo operario. Las pruebas de campo realizadas, demostraron que hay un gran porcentaje de aisladores que necesitan un mantenimiento dado que se encuentran rotos y/o sucios/flameados con un porcentaje del 50% de las muestras, adicionalmente el algoritmo posee una exactitud del 87.5% cuando trabaja en tiempo real, al aplicarlo a imágenes se obtuvo una exactitud de 94%, finalmente en videos pregrabados del 89%. En conclusión, es factible su implementación dentro de los planes de mantenimiento de las líneas de transmisión, disminuyendo los costos, dado que no es necesario contar con un carro grúa para la inspección, simplemente un operario que cuente con un dron y un computador personal para tomar la decisión de dar o no mantenimiento a los aisladores cerámicos.

Palabras clave: Visión Artificial, Algoritmo, Aisladores, Redes Neuronales, Líneas de Transmisión.

COTOPAXI TECHNICAL UNIVERSITY
POSTGRADUATES MANAGEMENT
ELECTRICITY MASTER DEGREE
MENTION ELECTRICAL POWER SYSTEMS

Title: Identification of failures in the isolators of a transmission line through artificial vision.

Author: Astudillo Cortez Vicente Paul

Tutor: MSc. Freire Martinez Luigi Orlando

ABSTRACT

This research is based on the superficial fault's identification in the porcelain insulators of the 13.8 kV transmission line of the cities of Latacunga and Salcedo through artificial vision, to know the status of these elements in an easy and simple way. Development began with a database of 3000 images of insulators divided into three states: dirty/flamed, broken and in good condition. With these images, Yolo was trained as identification algorithm, which is based on neural networks, the identification algorithm works both in real time and on pre-recorded images and videos. To make it's handling as easy as possible, a graphical interface was designed to be used by a single operator. The field tests carried out showed that there is a high percentage of insulators that need maintenance since they are broken and/or dirty/flamed with a percentage of 50% of the samples, additionally the algorithm has an accuracy of 87.5% when works in real time, while the algorithm applied to images had an efficiency of 94% finally in videos of 89%. In conclusion, its implementation is feasible within the maintenance plans of the transmission lines, lowering costs, because it is not necessary to have a crane car for the inspection, in fact, it is just necessary an operator with a drone and a personal computer to make the decision of whether or not to perform maintenance.

Keywords: Computer Vision, Algorithm, Insulator, Neural Networks, Transmission Line.

MARIA FERNANDA AGUAIZA IZA con cédula de identidad número 050345849-9, MAGISTER EN LINGÜÍSTICA APLICADA A LA ENSEÑANZA DEL IDIOMA INGLES COMO LENGUA EXTRANJERA con número de registro de la SENESCYT: 1020-2021-2354184: CERTIFICO: haber revisado y aprobado la traducción al idioma inglés del resumen del trabajo de investigación con el título: **Identificación de fallas en los aisladores de una línea de transmisión mediante visión artificial**, de: VICENTE PAUL ASTUDILLO CORTEZ con cedula de identidad número 050288778-9, aspirante a Magíster en Electricidad mención Sistemas Eléctricos de Potencia.

.....
MsC. MARIA FERNANDA AGUAIZA IZA
C.C 050345849-9



CENTRO
DE IDIOMAS

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO I	8
1. Fundamentación Teoría	8
1.1. Antecedentes	8
1.2. Fundamentación teórica.	10
1.2.1. Líneas de transmisión eléctrica.....	10
1.2.1.1. Tipos de líneas de transmisión.....	11
1.2.1.2. Elementos de las líneas de transmisión aérea.....	11
1.2.2. Mantenimiento de aisladores	13
1.2.2.1. Lavado	13
1.2.2.2. Cambio Parcial.....	13
1.2.2.3. Cambio Total	13
1.2.3. Fallas en los aisladores	13
1.2.3.1. Falla electrotérmica	14
1.2.3.2. Falla electromecánica	14
1.2.3.3. Falla electroquímica	14
1.2.3.4. Fallas por descargas eléctricas internas.....	14
1.2.3.5. Falla superficial.....	14
1.2.4. Factores que afectan a los aisladores	14
1.2.4.1. Contaminación ambiental.....	14
1.2.4.2. Descargas atmosféricas.....	16
1.2.4.3. Envejecimiento de los materiales	16

1.2.5. Visión artificial.....	17
1.2.5.1. Sistema de visión por computadora	17
1.2.5.2. Adquisición y representación de datos.....	19
1.2.5.3. Procesamiento de imágenes.....	20
1.2.6. Aprendizaje automático.....	22
1.2.6.1. Aprendizaje supervisado.....	22
1.2.6.2. Aprendizaje no supervisado.....	23
1.2.7. Python	23
1.2.7.1. Ventajas de python	24
1.2.7.2. Librerías de python	25
1.2.8. Redes neuronales artificiales	27
1.2.8.1. Local binary patterns (LBP)	29
1.2.8.2. Algoritmos de detección de imágenes.....	30
1.2.8.3. Yolo vs adaboost	34
1.2.8.4. Evaluación del algoritmo.....	35
1.2.8.5. Exactitud.....	36
1.2.8.6. Sensibilidad	36
1.2.8.7. Posibles fuentes de error	37
1.3. Fundamentación metodológica	37
1.4. Conclusiones Capítulo I	38
CAPÍTULO II	39
2. PROPUESTA	39
2.1. Título de la propuesta.	39
2.2. Objetivo del proyecto.	39

2.3. Descripción de la propuesta.....	39
2.3.1. Concepción del diseño	41
2.3.2. Adquisición de imágenes	41
2.3.2.1. Selección del dron	42
2.3.2.2. Distancia de seguridad del dron	43
2.3.3. Base de datos.	44
2.3.4. Preprocesamiento de imágenes	45
2.3.5. Entrenamiento	46
2.3.6. Clasificación.....	49
2.3.7. Interfaz gráfica	56
2.3.7.1. Detector en tiempo real.....	58
2.3.7.2. Detector en imágenes	60
2.3.7.3. Detector en video.....	62
2.3.7.4. Etapa de guardado	64
2.3.8. Dispositivo de procesamiento.....	64
2.4. Conclusiones Capítulo II	65
CAPÍTULO III	66
3. VALIDACIÓN Y/O APLICACIÓN DE LA PROPUESTA.....	66
3.1. Análisis de los resultados.	66
3.1.1. Validación del algoritmo	67
3.1.2. Detector en tiempo real.....	68
3.1.3. Detector de imágenes	71
3.1.4. Detector en video.....	73
3.2. Validación técnica.	75

3.3. Validación económica.....	76
3.3.1. Análisis económico.....	76
3.3.2. Análisis costo-beneficio.....	76
3.4. Conclusiones del Capítulo III	78
CONCLUSIONES	79
RECOMENDACIONES.....	81
REFERENCIAS BIBLIOGRÁFICAS	82
ANEXOS.....	86

ÍNDICE DE FIGURAS

Figura 1. Relación del sistema eléctrico nacional. [2]	1
Figura 2. producción de electricidad por tipo de generación. [2]	2
Figura 3. Esquema sistema eléctrico de potencia. [19].....	10
Figura 4. Sistema de transmisión de 500 kV. [20].....	11
Figura 5. Aislador rígido.....	12
Figura 6. Aislador de suspensión.....	12
Figura 7. Formación de una imagen. [26].....	17
Figura8. Luz frontal. [26].....	18
Figura 9. Luz lateral. [26]	18
Figura 10. Fases del procesamiento de imágenes con visión artificial. [27]	19
Figura 11. Resultados de la segmentación. [28]	20
Figura 12. Método de bordes. [28].....	21
Figura 13. Aplicación del método del área. [28].....	21
Figura 14. Diagrama aprendizaje automático. [29].....	22
Figura 15. Aprendizaje supervisado. [30]	22
Figura 16. Aprendizaje no supervisado. [30]	23
Figura 17. Tiempo de simulación versus número de paquetes.[31].....	24
Figura 18. Módulos OpenCV. [29]	25
Figura 19. Modelo de una neurona artificial.....	27
Figura 20. Estructura de la red neuronal [33].	28
Figura 21. Proceso LBP. [17]	29
Figura 22. Ejemplo del cálculo de coordenadas con Yolo.....	31
Figura 23. Arquitectura de Yolo [35].....	31
Figura 24. Cálculo de IOU. [35]	32
Figura 25. Probabilidad de Yolo. [18]	33
Figura 26. Procedimiento Adaboost [17].....	33
Figura 27. Arquitectura de la investigación.....	41
Figura 28. Dron Dji mini 2.	43
Figura 29. Posicionamiento del dron.	43
Figura 30. Preprocesamiento de las imágenes.	45

Figura 31. Proceso para encontrar puntos característicos.	46
Figura 32. Etiquetado de imágenes.	47
Figura 33. Proceso de entrenamiento.	47
Figura 34. Proceso de entrenamiento Google Colab.....	48
Figura 35. Localización de aisladores.	48
Figura 36. Arquitectura del algoritmo utilizado (Yolo).....	49
Figura 37. Análisis pixel vacío.	50
Figura 38. Análisis pixel con objeto.	51
Figura 39. Vector $S \times S \times C \times K$	53
Figura 40. Neurona de clasificación.	53
Figura 41. Etapa de clasificación.	56
Figura 42. Interfaz de inicio.	57
Figura 43. Proceso la visualización de resultados.....	57
Figura 44. Interfaz detector en tiempo real.	58
Figura 45. Rutina botón Detectar.	59
Figura 46. Rutina botón Regresar.	59
Figura 47. Interfaz detector de imágenes.	60
Figura 48. Rutina botón Elegir Imagen.....	61
Figura 49. Rutina botón Detectar.	61
Figura 50. Rutina botón Regresar.	62
Figura 51. Interfaz detector en video.	62
Figura 52. Rutina botón Elegir video.....	63
Figura 53. Rutina botón Regresar.	64
Figura 54. Aisladores de prueba.	66
Figura 55. Línea de 13.8 kV S/E Salcedo-el coliseo.	68
Figura 56. Fotogramas extraídos detección en tiempo real.	70
Figura 57. Detección aislador sucio/flameado.	72
Figura 58. Detección aislador bueno.....	72
Figura 59. Detección aislador roto.	73
Figura 60. Fotogramas del algoritmo de detección en videos.	75

ÍNDICE DE TABLAS

Tabla 1: Actividades objetivos específicos.	5
Tabla 2. Fuentes de contaminación de los aisladores.....	16
Tabla 3. Ventajas y desventajas Yolo vs Adaboost.	34
Tabla 4. Matriz de confusión. [36].....	35
Tabla 5. Matriz de confusión multiclase. [36]	36
Tabla 6. Estado de aisladores propuestos.....	40
Tabla 7. Representación de estados.	40
Tabla 8. Características del dron elegido [42].....	42
Tabla 9. Distancias mínimas de seguridad de conductores energizados [43].	44
Tabla 10. Distancias mínimas de seguridad de conductores energizados con viento[43]......	44
Tabla 11. Características del computador elegido. [48].....	65
Tabla 12. Matriz de confusión validación de la base de datos.....	67
Tabla 13. Sensibilidad validación de datos.	68
Tabla 14. Matriz de confusión detector en tiempo real.....	69
Tabla 15. Sensibilidad en tiempo real.	70
Tabla 16. Matriz de confusión detector en imágenes.....	71
Tabla 17. Sensibilidad en imágenes.	72
Tabla 18. Matriz de confusión detector en video.....	73
Tabla 19. Sensibilidad en video.	74
Tabla 20. Análisis económico.	76
Tabla 21. Costos actuales.....	77
Tabla 22. Flujo de caja.	77
Tabla 23. Referencia costo-beneficio.....	78

INTRODUCCIÓN

Según las Naciones Unidas la población mundial alcanzó los 7000 millones de habitantes en 2011 y se proyecta que aumente hasta los 8500 millones para el 2030 [1]. El crecimiento exponencial de la población ha causado una mayor demanda energética, por lo tanto, muchos países se han visto en la necesidad de aumentar su capacidad de producción eléctrica, Ecuador no es ajeno a este fenómeno pese a que la demanda eléctrica se redujo en un 0.47% en el 2020 en comparación al año 2019 debido a la pandemia causada por el COVID-19 [2].

Ecuador cuenta con un sistema eléctrico el cual según el artículo 39 de la Ley Orgánica del Servicio Estrategia de Energía Eléctrica se encuentra conformado por empresas dedicadas a las actividades de generación, autogeneración, transmisión, alumbrado público, importación y exportación de energía, distribución, comercialización y finalmente los usuarios finales o consumidores [2].

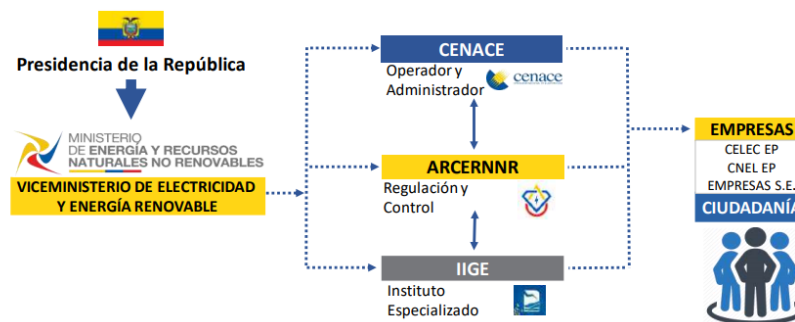


Figura 1. Relación del sistema eléctrico nacional. [2]

En la Figura 1 se muestra la conformación del Sistema Eléctrico Nacional, el cual es presidido por el Viceministerio de Electricidad y Energía Renovable representante del Presidente de la República adicionalmente del Operador Nacional de Electricidad (CENACE) como operador y administrador, la Agencia de Regulación y Control de Energía y Recursos Naturales no Renovables (ARCERNNR) como el ente regulador y de control, el Instituto de Investigación Geológico y Energético (IIGE) como instituto de investigación y finalmente las empresas distribuidoras y usuarios.

Ecuador al ser un país privilegiado en recursos naturales cuenta con diferentes empresas de generación públicas y privadas, en la actualidad la generación de dicha energía se la realiza mediante hidroeléctricas, termoeléctricas y fuentes de generación no convencionales (solar, eólica, biomasa, biogás) [2], siendo las hidroeléctricas las de mayor porcentaje de producción como se muestra en la Figura 2.

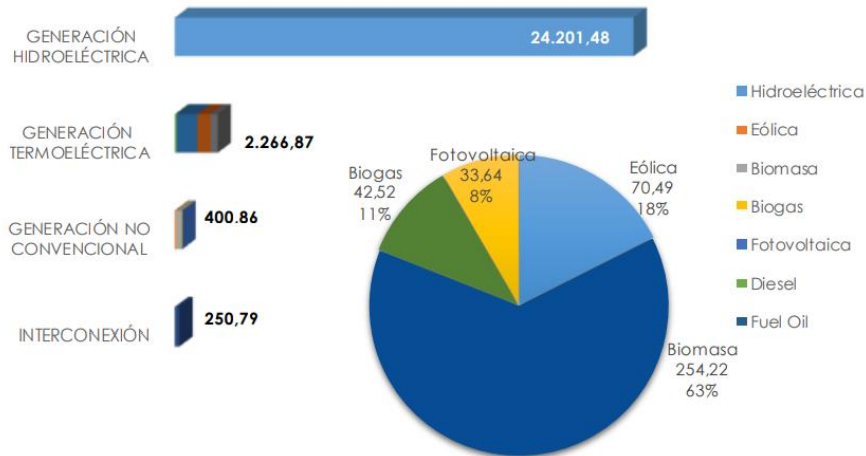


Figura 2. producción de electricidad por tipo de generación. [2]

Actualmente ante el aumento de la generación eléctrica a lo largo del país, se ha incrementado los métodos de transporte hacia el usuario final, los cuales se basan en líneas de transmisión que se encuentran conformadas por aisladores cuya función es ser la unión entre la estructura y el conductor desnudo, suelen ser de un material aislante que evite de forma segura al paso de corriente y al mismo tiempo aislarlo de tierra [3].

Dado que estos elementos son de vital importancia es necesario darles un mantenimiento preventivo y en el peor de los casos correctivo, puesto que la vida útil varía de los 6 meses hasta 3 años [4], dependiendo del lugar geográfico además de las condiciones climáticas, vida silvestre, etc. Debido a esto es necesario utilizar nuevas tecnologías para poder realizar un correcto análisis de los aisladores.

En la actualidad los avances tecnológicos nos permite mejorar imágenes y poder destacar varias características, por tal motivo, la visión artificial ha ido ganando terreno en diferentes campos como la medicina, biología e industrias [5], tomando en consideración que su objetivo es modelar matemáticamente los procesos de visualización de los seres vivos y permitir simular esta capacidad mediante en ordenadores [6].

Por lo tanto para desarrollar el planteamiento de la investigación se realizó una búsqueda exhaustiva sobre el reconocimiento de objetos en imágenes y videos, en la cual se encontró diversos proyectos y/o estudios relacionados al tema, se pudo constatar que existen diversos algoritmos que pueden ser utilizados para el reconocimiento de objetos[5], pero para el reconocimiento de aisladores eléctricos es necesarios utilizar algoritmos especiales.

Antecedentes: La presente investigación se encuentra directamente relacionada con las líneas de investigación de la Universidad Técnica de Cotopaxi específicamente con el campo de energías alternativas y renovables, eficiencia energética y protección ambiental adicional dentro de la sublínea explotación y diseño de sistemas de potencia y su temática calidad de energía puesto que el tema es “Identificación de fallas en los aisladores de una línea de transmisión mediante visión artificial”, conviene señalar que al ser un proyecto que se basa en los aisladores eléctricos permitirá prevenir fallas por parte de las empresas distribuidoras lo que significa una mejora en la calidad de energía para los usuarios finales.

En el plan maestro de Electricidad 2019-2027 en su apartado 1.5.2.1. que muestra sus metas propuestas, nos habla de la modernización y automatización de información de la distribución de energía [7], con lo que se busca aumentar el porcentaje de dispositivos para medir y/o controlar la transmisión de energía, instalando sistemas de medición avanzada (AMI por sus siglas en inglés), automatización de alimentadores, subestaciones automatizadas, transformadores de distribución monitoreado y monitoreo de líneas de distribución con la finalidad de poseer más información para mejorar la calidad de la energía.

Planteamiento del Problema: Debido al aumento de la demanda energética en el país se ha visto la necesidad de aumentar la capacidad de producción de energía eléctrica a 7.274 MW hasta el 2020, con un total 6.016 km de líneas de transmisión, dichas líneas de transmisión requieren de un mantenimiento programado y en el peor de los casos un mantenimiento correctivo en el año 2020 el porcentaje de fallas en dichas estructuras fue de 46.5% [2].

Una línea de transmisión es un elemento que forma parte del sistema nacional interconectado (SNI) que permite transportar energía desde un punto de generación hacia un punto de consumo, se basa en un enlace físico entre dos subestaciones y está conformado por varios elementos: estructuras metálicas, conductores, aisladores y accesorios [8]. La función de los aisladores eléctricos es mantener aisladas o alejadas las líneas energizadas de la estructura y los elementos que la conforman.

Las impurezas en los aisladores eléctricos que se encuentran expuestos constituye un gran inconveniente para la transmisión de energía eléctrica, dado que se depositan materiales orgánicos e inorgánicos, por tanto, se crea una capa de humedad que puede ser conductora, produciendo diversos fenómenos que pueden afectar a las propiedades aislantes y finalmente producirse roturas [9].

Al estar contaminados los aisladores son más propensos a almacenar humedad en su superficie en especial en las zonas de gran altitud o zonas tropicales/húmedas, dando una probabilidad muy alta de que se presenten descargas atmosféricas, produciendo una cantidad de energía suficiente para superar la capacidad de aislamiento y provocando una fuga de la línea hacia tierra, que a su vez provoca una disminución de la integridad mecánica provocando fisuras y en algunos casos roturas.

Para evitar dichos efectos se realizan mantenimientos periódicos, los cuales se basan en el análisis visual del personal para poder identificar si el aislador se encuentra en buen estado, deteriorado, sucio y/o fisurado. Para realizar dicha inspección es necesario contar con el carro grúa si existe una vía de acceso cerca de la línea de transmisión, si no es el caso es necesario que el técnico escale la estructura, lo cual podrían incurrir en accidentes para el personal.

En consecuencia, se pierde un tiempo valioso puesto que el personal debe hacer un análisis correcto para tomar la mejor decisión de las actividades a desarrollar ya sea el lavado del aislador, la sustitución o no hacer nada ya que el estado del aislador es óptimo, por tal motivo se propone implementar un método automático para el análisis de aisladores.

Formulación del problema

Determinación de la forma correcta de identificar las fallas de los aisladores de porcelana de la línea de 13.8 kV automáticamente.

Objetivo General

Identificar las fallas de manera autónoma en los aisladores de porcelana de la red de transmisión de 13.8 kV mediante visión artificial para su posterior clasificación según su estado.

Objetivos Específicos:

- Recopilar información referente al tema de investigación para la elaboración del estado del arte.
- Obtener imágenes de los aisladores preinstalados en las líneas de transmisión de 13.8 kV para la creación de una base de datos.
- Seleccionar el algoritmo adecuado para el uso de una red neuronal para procesamiento de imágenes de los aisladores.
- Realizar pruebas de campo para la comprobación del algoritmo de identificación de imágenes y su exactitud.

Sistemas de tareas en relación con los objetivos específicos:

Tabla 1: Actividades objetivos específicos.

Nº	Actividades (Tareas)	Resultado de la actividad	Descripción de la actividad
1	Recopilación de información referente al tema de investigación para la elaboración del estado del arte. Búsqueda de artículos científicos sobre visión artificial aplicado a la detección de aisladores.	Información verificada para el sustento, desarrollo y culminación de la investigación.	Análisis investigativo de la bibliografía encontrada con el fin de determinar las herramientas adecuadas para la investigación.
2	Creación de una base de datos de imágenes de aisladores con diferentes estados. Clasificación de las imágenes obtenidas para agruparlas por el estado de los aisladores.	Material visual de aisladores eléctricos de la línea de transmisión de 13.8 kV	Mediante el uso de un cuadricóptero con cámara incorporada, se obtiene imágenes de alta resolución de los aisladores instalados en la línea de transmisión, posteriormente se identifica el estado que se encuentra cada uno.

3	<p>Análisis de algoritmos de visión artificial aplicado a la detección de imágenes para seleccionar el que más se adapte a la investigación.</p> <p>Entrenamiento del algoritmo seleccionado para la identificación de las fallas en aisladores.</p>	<p>Algoritmo entrenado para el análisis de las imágenes y videos de aisladores.</p>	<p>Por medio de la investigación bibliográfica, se obtendrá el algoritmo adecuado para el análisis de las fallas de los aisladores. Mediante técnicas de visión artificial previamente seleccionadas se procede a al entrenamiento del método de clasificación con las imágenes de la base de datos.</p>
4	<p>Pruebas de validación del algoritmo con el porcentaje de la base de datos destinado para este fin.</p> <p>Pruebas de campo para la comprobación del algoritmo de identificación.</p>	<p>Resultados de la exactitud del algoritmo en campo.</p>	<p>Ingreso de imágenes de la base de datos distintas a las usadas en el entrenamiento para proceder con la validación del algoritmo entrenado.</p> <p>Selección de un tramo de la línea de transmisión de 13.8 kV donde no se obtuvo muestras para la base de datos y proceder al ingreso de cantidades significativas de videos e imágenes al algoritmo para verificar cuantos elementos clasifica de manera correcta.</p>

Justificación

La presente investigación tiene como objetivo realizar una inspección automática de los aisladores localizados en las líneas de transmisión, la cual ha sido limitada a la red de 13.8 kV de la Provincia de Cotopaxi en específico dentro de las ciudades de Latacunga y Salcedo, se debe tomar en consideración que el tipo de aisladores a analizar son los de suspensión que son generalmente de porcelana.

Dichos elementos tienen la función de aislar a el conductor de la estructura que generalmente son metálicas para que no exista ningún tipo de corriente de fuga hacia tierra, lo que lo convierte en parte fundamental para evitar algún tipo de falla o corte de energía que puedan verse afectados los usuarios.

Los aisladores al estar sometidos a los cambios climáticos tan repentinos en las diferentes regiones del país, son susceptibles a ser contaminados por impurezas de carácter biológico como microorganismos y del tipo inerte como el polvo o residuos de las industrias, lo que reduce significativamente su capa aislante, aumentando la probabilidad de que se presente algún tipo de falla.

Con la ayuda de un vehículo aéreo no tripulado (dron) se realizará la inspección de los aisladores de las líneas de transmisión, en consecuencia, se reducirá el tiempo de inspección de dichos elementos, adicionalmente se busca proteger al personal de mantenimiento ante accidentes que podrían estar expuesto a accidentes al realizar la inspección en líneas energizadas.

Se espera que los resultados obtenidos tengan una gran exactitud al momento de detectar el estado de los aisladores y así poder clasificarlos, con el fin de ayudar a las diferentes empresas eléctricas encargadas de las líneas de transmisión a tomar una decisión de manera inmediata sobre las actividades necesarias para cada caso encontrado.

Finalmente, con los resultados se busca que los costos disminuyan para la empresa encargada de las líneas de transmisión, dado que, el mantenimiento se realizaría solo a aisladores específicos y no será necesario realizar una inspección de todos, además se busca evitar cualquier tipo de accidente que se pueda presentar al realizar la inspección visual de dichos elementos por parte del personal técnico.

Hipótesis

La visión artificial con el uso de algoritmos de reconocimiento de imágenes servirá para detectar las fallas que se pueden presentar en los aisladores cerámicos de una línea de transmisión 13.8 kV, reduciendo costos para la empresa encargada de la línea de transmisión, de igual manera, reducir el riesgo laboral que existe al realizar dicha actividad.

CAPÍTULO I

1. Fundamentación Teoría

1.1. Antecedentes

En los últimos años ha existido un gran aumento de la producción nacional de electricidad debido a la incorporación de nuevas hidroeléctricas al sistema nacional interconectado [10], debido a que anteriores gobiernos se pusieron la meta de cambiar la matriz productiva en especial la matriz energética del país, lo que provocaría la disminución de emisiones de carbono. Este aumento de la capacidad de producción eléctrica del país ha causado que se construyan nuevas estructuras para poder distribuir esta energía a todo el territorio Ecuatoriano [11], por ende, el aumento de aisladores eléctricos.

Puesto que los aisladores eléctricos son elementos fundamentales en la transmisión de energía en la actualidad se han desarrollado varios estudios, por ejemplo en el estudio de Herbert Rojas sobre el impacto de impurezas en los aisladores por hongos [9], se la realizó en Bogotá Colombia durante 50 días consecutivos, utilizando un ambiente cerrado para controlar la luz, humedad y temperatura, dando como resultado que la contaminación de los aisladores produjo un aumento de las corrientes de fuga y flameos.

Como se observa el impacto ambiental afecta a las líneas de transmisión y a los aisladores según la región por ejemplo en el litoral manabita, se realizó un estudio de la influencia de los factores ambientales en la región debido a que en la región se procesa café, pescado y aceite de manera industrial adicional a la sal marina [12], lo que provoca la creación de una lámina de suciedad en los elementos de las líneas de distribución que se encuentra compuesta por varios elementos por ejemplo calcio, magnesio, cloruro de sodio y potasio lo que contribuye a la reducción de la vida útil de los mismos.

En la actualidad la tecnología ha avanzado en todos los campos de la ciencia, la visión artificial ha permitido mejorar procesos, en la investigación de Estarita Jorge sobre un sistema de reconocimiento de objetos en tiempo real [13], se pudo aplicar algoritmos para el reconocimiento de objetos mediante una cámara web para identificar patrones únicos de objetos para su posterior comparación, con lo que se pudo concluir que es necesario tener imágenes claras para que el tiempo de cómputo sea lo mínimo posible.

Debido a este inconveniente en la investigación de Ruiz Rail [14], se presentan diversas directrices para evitar para llevar a cabo un modelo de reconocimiento exitoso principalmente la validación cruzada que permite realizar un análisis estadístico y dividir en dos grupos cada uno destinado a entrenar y evaluación de la base de datos respectivamente.

Para el reconocimiento de imágenes comúnmente se utilizan redes neuronales como en la investigación de Quintero Carlos [15], que utiliza esta herramienta para reconocer macroinvertebrados, sabiendo que dichos insectos poseen muchas características únicas se puede concluir que la visión artificial se adapta a cualquier tipo de aplicación y/o campo de la ciencia.

Mediante la investigación realizada por Shen Young Wong [16], se demuestra que existen varios métodos utilizando inteligencia artificial para la detección de fallas en varios elementos de las líneas de transmisión, con los cuales se puede identificar varios fallas como la descarga de energía, control de voltaje, etc.

Por tal motivo la visión artificial que es parte del campo de la inteligencia artificial es la mejor herramienta para identificar el estado de los aisladores como lo demostró la investigación de Patricio Valencia [17], en la cual se analiza imágenes de diversos aisladores fijos de una línea de transmisión perteneciente a un campo petrolero obteniendo buenos resultados, pero fue necesario realizarlo de forma local es decir tomar las fotografías en campo para su posterior análisis.

Adicionalmente en la investigación de Sergio Beleño [18], se demostró que es necesario realizar un preprocesamiento de las imágenes de los aisladores para resaltar los atributos y sea mucho más sencillo el posterior análisis mediante el algoritmo dado que el dron debe posicionarse a una cierta distancia debido a los campos electromagnéticos producidos por la línea de transmisión.

1.2. Fundamentación teórica.

1.2.1. Líneas de transmisión eléctrica.

Tiene como objetivo la transmisión de energía eléctrica a los distribuidores y grandes consumidores, como el intercambio internacional de energía, no forman parte de las líneas los transformadores de los grupos de generación y las instalaciones de conexión de dichos grupos [19].

Para realizar la evacuación de la energía producida por las centrales de generación se requiere la instalación de un transformador elevador, para que esta energía se encuentre en los niveles requeridos, que tiene como objetivo evitar caídas de tensión y disminuir la corriente en las líneas de transmisión para finalmente llegar hacia los usuarios finales, como se muestra en la Figura 3 en un diagrama unifilar.

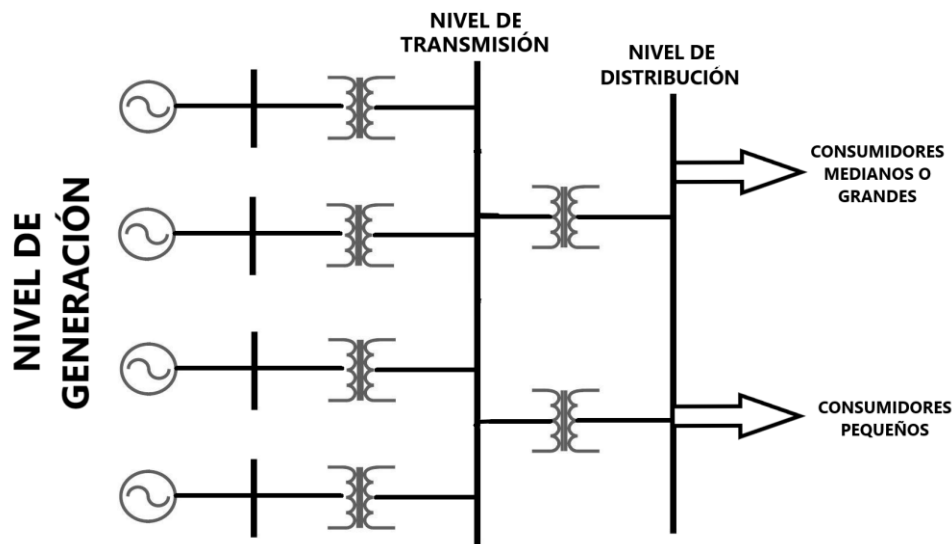


Figura 3. Esquema sistema eléctrico de potencia. [19]

En Ecuador el sistema de evacuación más reciente es la línea de 500 kV de extra alto voltaje [11] (Figura 4), el cual se encuentra conformado por 4 subsistemas y 21 obras, se basa en el tendido de 602 kilómetros de líneas de transmisión que atravesará por 12 provincias [20], adicionalmente se conoce que Cotopaxi posee aproximadamente 4000 Km de tendido eléctrico [8], (ver Anexo H).

Con el avance de un 99.99% de este proyecto se aumentó la confiabilidad del sistema eléctrico ecuatoriano ante diferentes fallas que se pueden presentar, con el tendido de estas nuevas líneas de transmisión se construyeron diferentes estructuras y se utilizó diferentes elementos entre ellos los aisladores eléctricos.



Figura 4. Sistema de transmisión de 500 kV. [20]

1.2.1.1. Tipos de líneas de transmisión.

El transporte y distribución de energía eléctrica de alta tensión se lo realiza de dos maneras:

- **Líneas Aéreas.**

Se denomina línea aérea a la estructura formada por apoyos, aisladores y cables desnudos destinados al transporte de la energía eléctrica trifásica, con una tensión nominal de línea superior a 1000 voltios [19].

- **Líneas Subterráneas.**

Se denomina línea subterránea a la estructura formada por una canalización que no sea aérea (zanja, galerías en el interior de edificios, fondos acuáticos, etc.) con cable aislado [19].

En el país existen diferentes niveles de transmisión: 500 kV, 230 kV, 138 kV, 69 kV y 46 kV, 13.8 kV mientras que a niveles de distribución secundaria se cuenta con voltajes 127/220 V.

1.2.1.2. Elementos de las líneas de transmisión aérea.

Las líneas de transmisión se encuentran constituidos por 3 elementos principales:

- **Conductores**

Los conductores para las líneas de alta tensión por lo general siempre se encuentran desnudos y son de aluminio, debido a que este material es más económico que el cobre, pero tiene las mismas propiedades de conductividad [21].

- **Soporte o Estructuras**

Las estructuras son las encargadas de mantener alejada a los conductores a una distancia segura del suelo y a una distancia segura uno del otro para evitar que se pueda generar arcos eléctricos, generalmente son estructuras de metal [21].

- **Aisladores**

Sirven para mantener alejada la línea energizada del soporte de metal y así evitar fugas, principalmente son construidos en porcelana y de material sintético [21].

Los aisladores se pueden clasificar según su diseño:

Aisladores Rígidos: Se encuentran unidos mediante un herraje fijo y no pueden cambiar de posición después de su colocación, se lo utiliza normalmente en sistemas de mediana y baja tensión [22] (ver Figura 5).



Figura 5. Aislador rígido.

Aisladores de Suspensión: Se encuentran constituidos por elementos llamados campanas, son cadenas móviles en el punto de unión al soporte, este tipo de aislador es el más utilizado en alta y media tensión. Se encuentran construidos por materiales como la porcelana y vidrio templado, la porcelana es la más utilizada debido a que es un material económico y de fácil acceso, estos dos elementos son sometidos a un proceso térmico para aumentar su resistencia mecánica [17], (ver Figura 6).



Figura 6. Aislador de suspensión.

1.2.2. Mantenimiento de aisladores

El mantenimiento para este tipo de elementos es importante dado que, al estar expuestos a las condiciones climáticas [23], suelen presentar grietas o contaminantes que reducen propiedad aisladora, generalmente se realiza tres tipos de mantenimiento:

1.2.2.1. Lavado

Se refiere al lavado de los aisladores con ayuda de agua a presión generalmente se realiza en líneas desenergizadas [23], para líneas energizadas se suele utilizar agua desmineralizada dado que es necesaria que el agua cuente con una baja resistividad para no ocasionar descargas a través del chorro.

1.2.2.2. Cambio Parcial

Cuando un aislador ha cumplido con su vida útil o se encuentra con alguna rotura significativa se realiza el cambio del mismo, al trabajar con cadenas de aisladores no es necesario realizar el cambio de todos los aisladores simplemente la unidad que se encuentre afectada.

1.2.2.3. Cambio Total

Se basa en el cambio completo de la cadena de aisladores para volver a una condición inicial de los elementos.

1.2.3. Fallas en los aisladores

Los aisladores que se encuentran a la intemperie pueden presentar diversas fallas, dado que son susceptibles a contaminación ambiental, descargas eléctricas, etc. Dichas fallas pueden clasificarse por el tipo o fenómeno que se presenta:

Tipos de Fallas:

- Electrotérmico
- Electromecánico
- Electroquímico
- Descargas eléctricas internas
- Superficial

1.2.3.1. Falla electrotérmica

Producida generalmente por la circulación de una corriente de fuga la cual por el efecto Joule [24], genera calor lo que implica un descenso de la resistencia del elemento, tomando en consideración que los elementos dieléctricos cuentan con un coeficiente negativo de variación de la resistencia frente a la temperatura, lo que desencadena en un aumento de la corriente causando el rompimiento del aislador.

1.2.3.2. Falla electromecánica

Este tipo de fallas comúnmente se presenta en los aisladores de suspensión debido al peso del conductor, el viento, fuerzas electrostáticas, corrosión, envejecimiento, etc. Produciendo la disminución de la resistencia mecánica y/o roturas, generalmente producidas en lugares de altitud considerable como paramos.

1.2.3.3. Falla electroquímica

Debido al clima tan diverso a los que son sometidos los aisladores tienden a cambiar sus propiedades químicas, provocando la disminución del tiempo de vida, dichas fallas se presentan en los aisladores que se encuentran formados por sustancias orgánicas generalmente por la presencia de ácido nítrico [25].

1.2.3.4. Fallas por descargas eléctricas internas

Se presenta este tipo de falla en los aisladores por defectos de fabricación, por la presencia de burbujas de aire que con el paso del tiempo se hacen más notorias lo que provoca un paso de baja resistencia ante las corrientes de fuga.

1.2.3.5. Falla superficial

Producida en la superficie de los aisladores debido a corrientes de fuga o descargas eléctricas, en el caso que se presente en un aislador sucio o perforado la tensión requerida para que se presente un arco eléctrico es mucho menor respecto a un aislador limpio, produciendo deterioro de la superficie aislante.

1.2.4. Factores que afectan a los aisladores

Existen varios factores que afectan directamente a los aisladores.

1.2.4.1. Contaminación ambiental

Dado que estos elementos se encuentran a la intemperie y a merced de diferentes condiciones ambientales que en nuestro país son muy diversas, los hace propensos a diferentes contaminantes (ambiental, industrial y biológica) [9].

Dichos elementos son:

- **Temperatura**

La temperatura en una línea de transmisión puede variar dependiendo el lugar y temporada del año, este efecto no influye de manera directa en el rendimiento del aislador, pero sí puede disminuir paulatinamente su vida útil [22].

- **Radiación ultravioleta**

Acelera el envejecimiento de los aisladores que se encuentran compuestos de materiales no cerámicos, pero no tiene ningún efecto sobre los aislantes de porcelana y vidrio [22].

- **Lluvia**

La lluvia moja la superficie de los aisladores y crea una pequeña capa conductora en gran parte de las ocasiones, lo que ocasiona una reducción de descarga disruptiva de los aisladores [22].

- **Contaminación**

El viento lleva diferentes partículas contaminantes que son depositadas en los aisladores, debido a que estos elementos producen pequeñas turbulencias en el flujo normal del aire [22]. Este fenómeno no causa mayor problema si se realiza un mantenimiento de los aisladores de manera periódica, si no fuera el caso se puede formar una pequeña capa de suciedad en la parte superior del elemento y sumamente gruesa en la parte inferior. Esta contaminación reduce el voltaje de descarga disruptiva en un 20 a 25%.

- **Altitud**

Este fenómeno afecta a los aisladores debido a que la altitud causa una disminución de la descarga disruptiva, que dentro del país en especial en la zona de la Sierra cuenta con una altitud considerable, una forma para solucionar este problema es un aumento del 3% de la holgura de las cadenas de aisladores por cada 300 metros de altitud [22].

En la Tabla 2 se puede observar las fuentes más comunes de contaminación que se suelen afectar a los aisladores, como se puede observar las zonas costeras son las más afectadas, así como también las zonas industrializadas que en el país son muy comunes debido a que existen diferentes fábricas muy cercanas a las líneas de transmisión existentes.

Tabla 2. Fuentes de contaminación de los aisladores.

Tipo de Contaminación	Fuente de Contaminación	Características	Área de Afectación
Áreas Rurales	Polvo del suelo	Capa de alta resistividad, lavado efectivo con lluvia.	Áreas extensas
Áreas Costeras	Sal de mar	Baja resistencia, lavado fácil con lluvia.	10-20 km del mar
Zonas Industriales	Plantas de cemento, plantas químicas, canteras, etc.	Alta resistencia, difícil de remover, insoluble.	Cerca de la fuente de contaminación.
Zonas Mixtas	Industrial, carreteras	Resistencia de rango medio, muy adhesivo.	Cerca de la fuente de contaminación.

1.2.4.2. Descargas atmosféricas

Suelen presentarse en días lluviosos, al producirse una descarga atmosférica cerca del elemento se crean ondas transitorias de corriente y voltaje que viajan en hacia las dos direcciones de la línea de transmisión, al llegar a la torre de transmisión se crea una diferencia de potencial con los aisladores y su propia estructura, lo que provoca que se supere el Nivel Básico De Aislamiento Por Impulso Tipo Rayo (NBAI) [24], provocando una descarga ente la fase y la estructura de la línea de transmisión dicho fenómeno se denomina “flameo”.

Este fenómeno puede producirse de dos maneras:

Flameo Directo: Se produce la descarga desde la fase hacia tierra a través de la estructura de la torre de transmisión.

Flameo Indirecto: Se produce cuando la descarga parte desde los conductores de tierra hacia la fase a lo largo de dos torres de transmisión.

1.2.4.3. Envejecimiento de los materiales

La continua descarga de sobretensiones muy intensas y de corta duración ocasionadas por descargas atmosféricas, adicionalmente por la contaminación producida por el ambiente hace que los materiales aislantes que constituyen al aislador entren en un proceso de disminución de sus propiedades dieléctricas además de su estructura físico-química.

1.2.5. Visión artificial

Es una rama de la inteligencia artificial también conocida como visión por computador, es la ciencia que permite procesar imágenes o videos adquiridos de manera física y lo transforma a una forma digital para que un computador pueda entenderlos, se basa en el sistema de visión humana donde el cerebro procesa las imágenes que son obtenidas mediante los ojos [26].

1.2.5.1. Sistema de visión por computadora

Con el paso de los años los sistemas caseros de computadores son sumamente económicos, muchos de los cuales vienen incorporados con un sistema de visión incorporado al incluir una cámara y su propia interfaz dentro de la misma [26].

En la Figura 7 se observa los elementos necesarios para tomar una imagen y posteriormente realizar el procesamiento de imágenes con ayuda de una hardware y software.

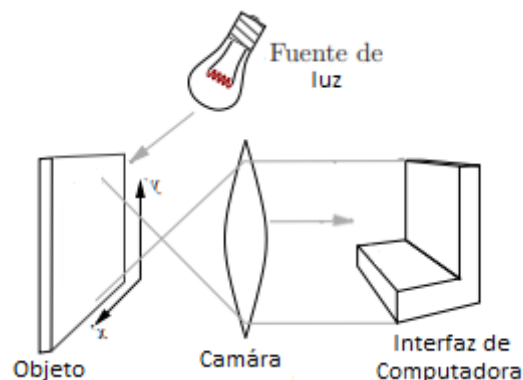


Figura 7. Formación de una imagen. [26]

- **Dispositivo de captura de imágenes**

Es un elemento del tipo sensor, se basa en la propiedad de la luz para producir pares de electrones huecos en un material conductor, al aplicar un potencial dicha carga se la detecta como corriente.

Según la ley de ohm, el voltaje a través de una resistencia es proporcional a la corriente que atraviesa, por lo que la corriente se puede convertir en voltaje, pasando a través de una resistencia, el número de electrones es proporcional a la cantidad de luz, por ende, una mayor carga es causada por una mente en el brillo, de esta manera, las cámaras pueden proporcionar como salida un voltaje que proporciona el brillo de los puntos tratados por la cámara [26].

- **Interfaz de computadora**

Es el sistema encargado de procesar la imagen comúnmente es un computador o un sistema embebido, el sistema puede hacer un procesamiento de imágenes, así como también técnicas de visión artificial y la velocidad con la que haga esas tareas dependerá directamente de las características del sistema y el lenguaje utilizado para su programación.

- **Iluminación**

La iluminación es un factor importante para el tratamiento de imágenes debido a que si es deficiente no se mostrarán todos los atributos [26], la función de la cámara al momento de capturar un objeto también se captura las sombras que se generan por la luz reflejada en los elementos.

Luz Frontal: Esta técnica se basa en que la cámara se coloca en frente al objeto, lo que permite destacar los atributos ya que la dirección de la luz se encuentra en la misma dirección del objeto [26].

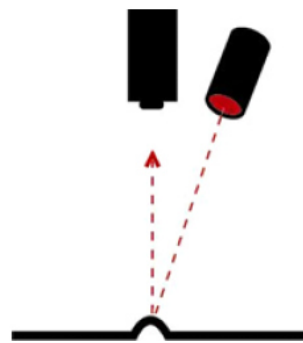


Figura 8. Luz frontal. [26]

Luz Lateral: Esta técnica se basa en que la cámara se coloca en frente al objeto mientras que la iluminación como su nombre lo indica viene de los laterales, es utilizada cuando se desea resaltar atributos en particular [26].

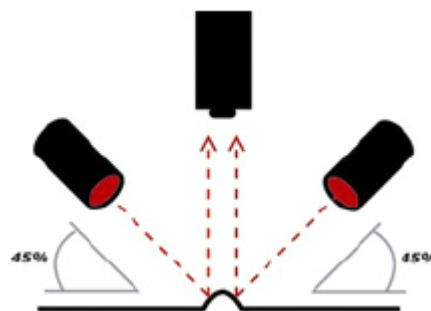


Figura 9. Luz lateral. [26]

1.2.5.2. Adquisición y representación de datos

Para que una imagen pueda ser procesada por un computador (hardware/software) se contemplan una serie de pasos que se muestran en la Figura 10, los pasos del procesamiento de imágenes mediante las diferentes herramientas que posee la visión artificial. Estas fases no dependen una de la otra ni tampoco son obligatorias, se toma en consideración según el enfoque del tratamiento escogido para la imagen [27].

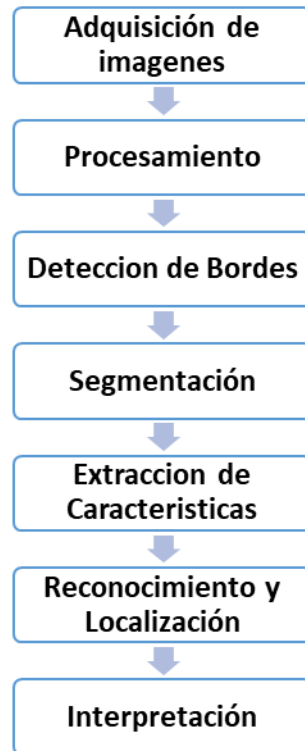


Figura 10. Fases del procesamiento de imágenes con visión artificial. [27]

La adquisición de imágenes se basa en la captura y digitalización de la imagen de lo físico a lo digital, al digitalizar la imagen entra en un preprocesamiento, el cual facilita las posteriores etapas, dentro de la detección de bordes en el cual se discrimina el fondo y separando los objetos de interés.

Posteriormente se realiza la segmentación donde se puede variar los valores RGB (Red-Green-Blue), en los cuales se puede filtrar las imagen según la conveniencia, para finalmente entrar en el reconocimiento y localización donde se clasifican los objetos con características comunes mediante técnicas de triangulación para seleccionar la imagen deseada [27].

Se puede pensar que el tratamiento de imágenes es sumamente sencillo debido a que nosotros percibimos las imágenes y nuestro cerebro las procesa de manera automática, pero realizar esta tarea para un computador es un poco complicado debido a que es necesario extraer información de las imágenes, para lo cual es necesario aplicar diferentes filtros para poder obtener dicho resultado.

El uso de un dispositivo sensible a la luz permite obtener imágenes de forma digital, la dificultad del procesamiento de imágenes puede variar según el nivel de la iluminación mientras que la complejidad de la imagen u otros procesos que puedan interferir en el tratamiento de la imagen.

1.2.5.3. Procesamiento de imágenes

Para poder resaltar los atributos que pueda contener una imagen digital se aplicarán diferentes mecanismos principalmente filtros, dichos mecanismos nos permiten descomponer la imagen para que un computador sea capaz de diferenciar y distinguir objetos presentes. Se sabe que una imagen posee diferentes tipos de atributos, los cuales mediante un software especializado es posible resaltarlos, modificarlos y/o eliminarlos para tener una imagen más clara [26].

- **Segmentación**

Se basa en dividir la imagen en varios segmentos o regiones, es una de las primeras tareas de los sistemas de visión por computador. Es una etapa crítica en los sistemas de análisis, debido a que usualmente se siguen arquitecturas de procesamiento secuenciales que tienen a la segmentación como una de las etapas iniciales y sus resultados afectan a las etapas posteriores [28].

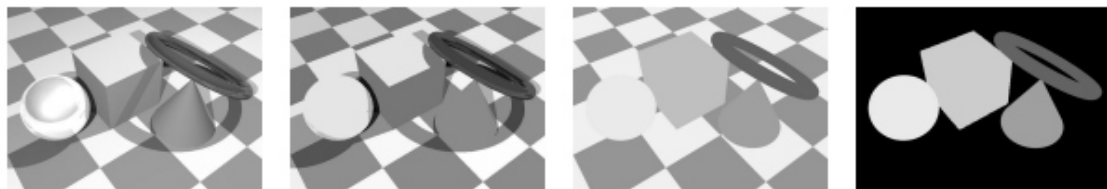


Figura 11. Resultados de la segmentación. [28]

En la Figura 11 se observa el resultado de la segmentación en varios niveles, es posible diferenciar las figuras del fondo, como también sus atributos como sombra y nivel de contraste.

- **Método de bordes**

El método de bordes se basa en discontinuidades de cada objeto puesto que son atributos únicos que se presentan los píxeles adyacentes basada en bordes. Por lo general se debe realizar una correcta segmentación debido a que el método suele dejar espacios entre cada uno de los límites de dichos objetos se requieren algunos mecanismos para presionarlos adecuadamente.

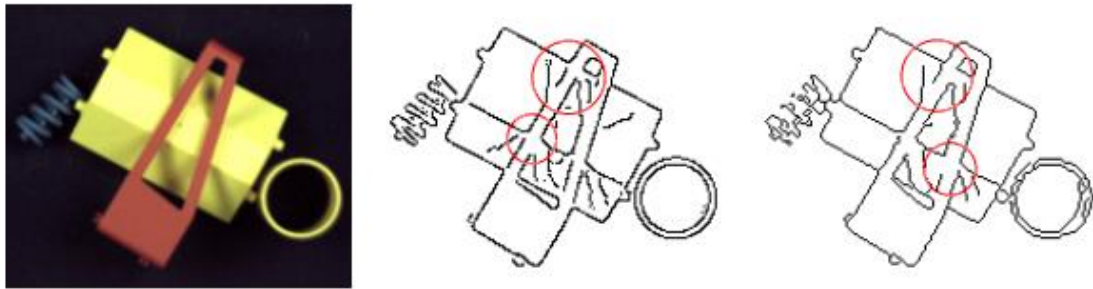


Figura 12. Método de bordes. [28]

Como se observa en la Figura 12, el método de bordes es efectivo cuando se necesita saber el contorno de un objeto dentro de una imagen, muchas de las veces se presentan errores los cuales deben ser corregidos mediante un algoritmo o manualmente.

- **Método de área**

Es una técnica tradicional que utiliza dos principios, crecimiento regional o división y fusión es método de crecimiento regional asume la existencia de algunos puntos de principales a los que se les agrega píxeles adyacentes y cumple un criterio de homogeneidad la principal ventaja de este método es la creación de regiones compactas y conectadas espacialmente que contrastan con la participación de imágenes generalmente ruidosas. Se aplica generalmente para separar un objeto de su fondo.

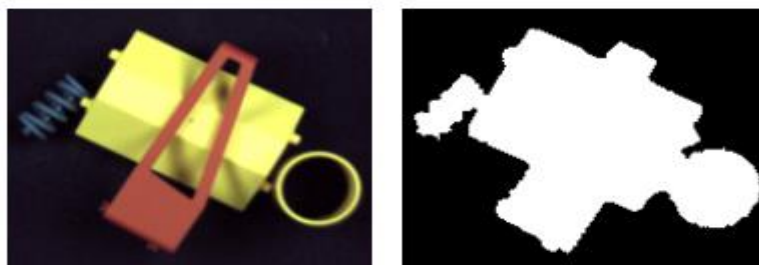


Figura 13. Aplicación del método del área. [28]

1.2.6. Aprendizaje automático

También conocido como machine learning es uno de los principales pilares de la inteligencia artificial, se lo puede definir como la forma que los ordenadores o máquinas aprendan que actividades realizar sin que fueran programados para ello [26]. Por lo general se utilizan algoritmos para aprender ciertos datos y según sea el caso tomar las decisiones pertinentes.

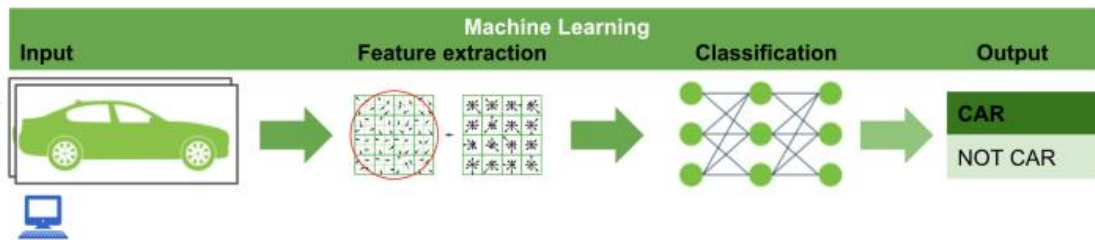


Figura 14. Diagrama aprendizaje automático. [29]

Como se observa en la Figura 14, nos muestra el proceso del aprendizaje automático en el cual se basa en el ingreso de imágenes, después la extracción de atributos, una vez encontrado los atributos se procede a clasificarlos y compararlo con la base de datos para finalmente dar un resultado afirmativo o negativo.

1.2.6.1. Aprendizaje supervisado

Es necesario de la intervención humana la cual debe dar una retroalimentación de las actividades a realizar, es decir es necesario enseñar al algoritmo mediante una base de datos a reconocer los atributos deseados. Como se observa en la Figura 15 el aprendizaje supervisado utiliza la regresión lineal para obtener atributos similares de una base de datos para su posterior clasificación.

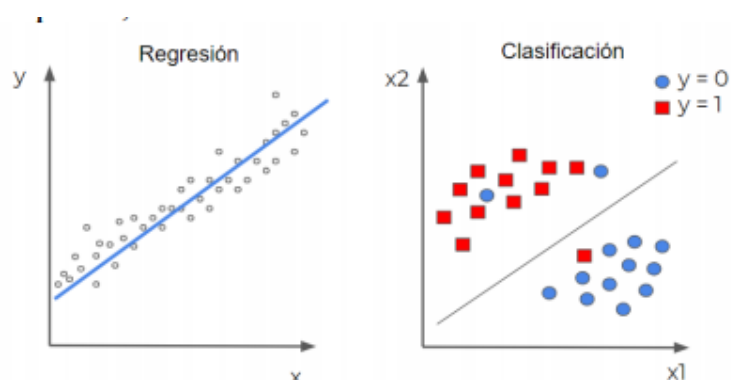


Figura 15. Aprendizaje supervisado. [30]

1.2.6.2. Aprendizaje no supervisado

Se lleva a cabo solamente con entradas sin retroalimentación de la salida, es decir el algoritmo solo aprende de las entradas para predecir los atributos de futuros atributos, son utilizados cuando se utiliza datos con características similares y no hay iteración humana.

Como se observa en la Figura 16 el aprendizaje no supervisado clasifica los datos según atributos similares de manera automática, pero toma más tiempo y recursos computacionales.

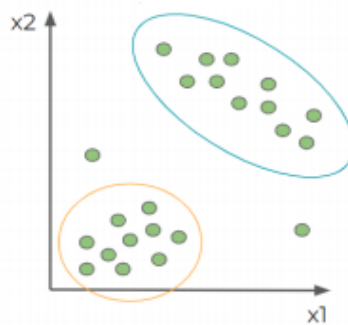


Figura 16. Aprendizaje no supervisado. [30]

1.2.7. Python

Es un lenguaje de programación de alto nivel y secuencial de comandos, sencillo y fácil de utilizar, es un software libre y posee bibliotecas propias lo cual lo hace el favorito en infinidad de plataformas [26], posee varios puntos a favor comparado con otros lenguajes de programación.

- **Expresivo:** Debido a que está basado en lenguaje C y es compacto, es decir su programación es mucho más corta.
- **Legible:** Su escritura es sencilla y de igual manera su lectura es sencilla comparada a otros lenguajes.
- **Interactivo:** Ofrece un entorno amigable que permite realizar pruebas respecto a las características del lenguaje.
- **Ejecución:** Detecta la mayoría de los errores que se les escapan a los compiladores tradicionales y proporciona la información necesaria para corregirlos.
- **Orientado a objetos:** Al ser un lenguaje de alto nivel se lo puede utilizar en programación orientada a objetos, que pueden ser manipulados de manera sencilla.

1.2.7.1. Ventajas de python

- Su uso y proceso de programación es sumamente sencillo comparado con otros lenguajes de programación, especialmente para gente que no especializada en el campo de la programación.
- Al ser desarrollado bajo código abierto su uso y distribución es de forma gratuita tanto para el entorno aprendizaje como para el comercial.
- Es adaptable a cualquier tipo de proceso, dado que, se puede utilizar tanto una programación orientada a objetos como programación estructurada y funcional.
- Posee un gran número de librerías que son adaptables a cualquier proceso desarrolladas de forma oficial, así como también por usuarios.
- Al poseer la característica multiplataforma es posible utilizarlo en varios sistemas operativos.
- Al contar con gran acogida en el mundo existen diversas plataformas de ayuda para la solución de problemas de código.

Por tales motivos la elección más acertada es utilizar Python para el desarrollo y puesta en marcha de la red neuronal, dado que es adaptable adicionalmente los costos para la empresa o persona que decidiera utilizar el algoritmo de reconocimiento de fallas podría utilizarlo sin ningún tipo de licencia comercial y al ser un lenguaje fácil de comprender se puede modificarlo para mejorar su rendimiento a futuro.

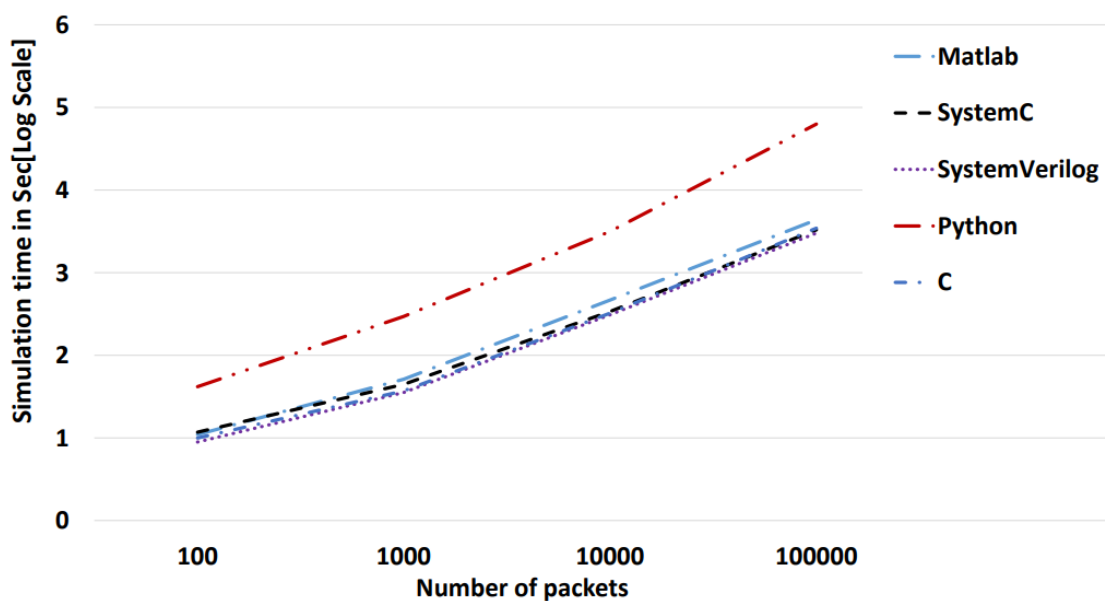


Figura 17. Tiempo de simulación versus número de paquetes.[31]

Se observa en la Figura 17, la comparación de velocidad versus el número de paquetes, indicando que Python es muy superior a otros lenguajes, lo que nos indica que es la opción más adecuada para realizar la presente investigación.

1.2.7.2. Librerías de python

A continuación, se detalla los principales paquetes o librerías para el desarrollo del algoritmo de identificación.

- **OpenCV**

Proviene de visión por computadora de código abierto por sus siglas en inglés, comprende bibliotecas, herramientas y módulos especialmente diseñados para el tratamiento de imágenes por computadora. Es compatible con diferentes programas como C/C++, Python y Java.

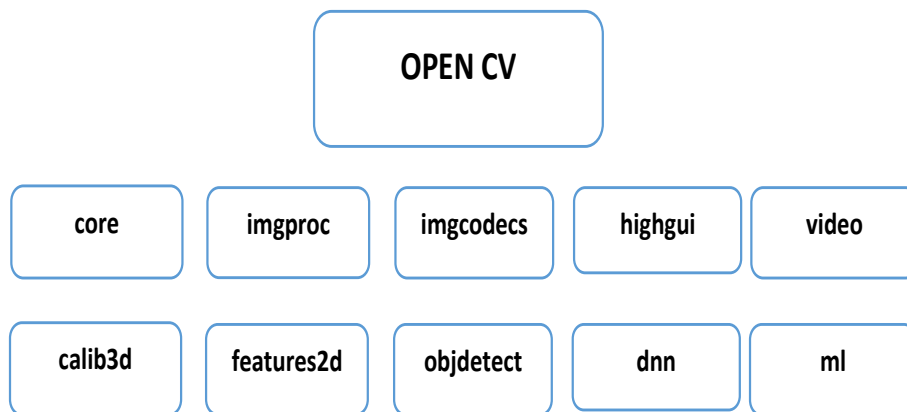


Figura 18. Módulos OpenCV. [29]

- **Core:** Es la funcionalidad principal que se asocia con las demás funciones básicas.
- **Imgproc:** El módulo incluye filtrado, transformación de imágenes y metamorfosis de histogramas.
- **Imgcodecs:** Lectura y escritura de imágenes
- **Highgui:** Gui de alto nivel, permite crear con facilidad ventanas e imágenes a mostrar.
- **Videoio:** Captura de video.
- **Calib3d:** Calibración de la cámara y reconstrucción 3D.
- **Features2d:** Detector de características y comparadores.
- **Objdetect:** Detección de objetos e instancias de clases predefinidas.
- **Dnn:** Módulo de red neuronal profunda.
- **Ml:** Módulo de aprendizaje automático.

- **Tkinter**

Librería especializada en desarrollo de herramientas Interfaz Gráfica de Usuario (GUI) [26], se basa en la instalación de secuencia de comandos y gráficos desarrollado por John Ousterhout. Su desarrollo se encuentra basado en lenguaje C con la adición de módulos para admitir scripts interpretativos o aplicaciones. Su éxito radica que su programación orientada a objetos permite que sea una plataforma ideal para desarrollar marcos de aplicaciones.

- **Imutils**

Librería especializada en para el procesamiento de imágenes y videos [26], es una forma más sencilla de usar la librería OpenCV cuyas funciones principales son:

- Rotación
- Translación
- Cambio de Tamaño

- **Pillow**

Librería especializada en la edición de imágenes, soportando varios formatos como JPG, PNG, etc [26]. Su código base fue desarrollado en el lenguaje de programación C, adicionalmente se puede utilizar para aplicar diferentes filtros para poder resaltar características.

- **Numpy**

Librería especializada que proporciona un objeto de una matriz multidimensional, varios objetos heredados como matrices [26], además de una variedad de arreglos matemáticos, manipulación de formas, clasificación, selección, transformadas de Fourier, etc.

- **Sift**

Librería especializada en la extracción de puntos clave y calcula su orientación dentro de la imagen y mostrarlos en una imagen de salida [26], es una herramienta muy utilizada en el tratamiento de imágenes, dado que permite eliminar puntos característicos de bajo contraste dejando los más representativos, la precisión puede verse afectada por el ruido, para lo cual es necesario realizar un correcto filtrado.

- **Matplotlib**

Biblioteca que permite realizar trazado en 2D, producido en una gran variedad de formatos impresos y entornos interactivos [26], su uso se puede utilizar tanto mediante comando como por medio de aplicaciones web. Permite generar diagramas, histogramas, espectros, gráficos de barra, etc.

- **LabelImg**

Es una herramienta de anotación grafica de imágenes, desarrollado en Python, al realizar la etiqueta genera un archivo XML en formato Pascal VOC [26], dichos datos permiten entrenar una red neuronal, para predecir e identificar datos idénticos que no se encuentran dentro de dicho conjunto de datos.

Al ser librerías no es posible obtener un accuracy o exactitud de las mismas, dado que, la exactitud se la puede medir en conjunto mediante las aplicaciones desarrolladas con el uso de las librerías [32].

1.2.8. Redes neuronales artificiales

Se conoce como redes neuronales a un conjunto interconectado de elementos, unidades o nodos de procesamiento simples, que se asemeja a una neurona humana [33]. Son algoritmos por computadora que son basados en modelos matemáticos, cuyo objetivo es aprender y tomar decisiones de manera autónoma.

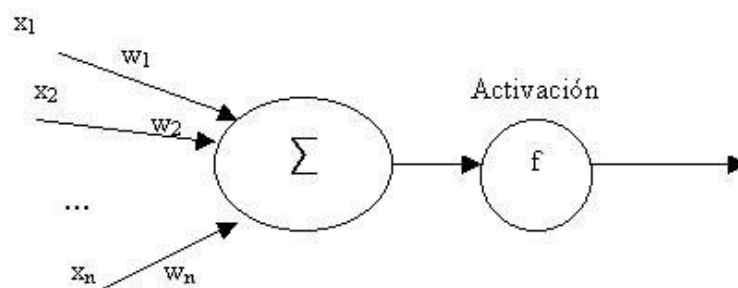


Figura 19. Modelo de una neurona artificial.

En la Figura 19, se muestra el funcionamiento en la cual se observa que ingresan a un nodo de comparación los pesos pre-entrenados de la red y los datos de entrada, en el cual si la entrada supera un límite conocido como umbral se produce un valor 1 de lo contrario será un 0, es decir nos dará un resultado afirmativo o negativo.

Dicha salida se obtiene a partir de la regla de propagación que viene dada por:

$$Y_i(t) = f_i \left(\sum_{j=0}^n w_{ij} * X_j \right)$$

Donde:

$X(i)$ = Entradas de la Neurona

$W(ij)$ = Pesos Asociados a las entradas

$f(i)$ = Funcion de Activación

$y(i)$ = Salida de la Red Neuronal

Como se observa las variables de entrada vienen dadas del resultado del análisis de cada pixel de una imagen a predecir, posteriormente se realiza una operación con los pesos pre-entrenados, seguido de una función de activación, dado por una función escalón,

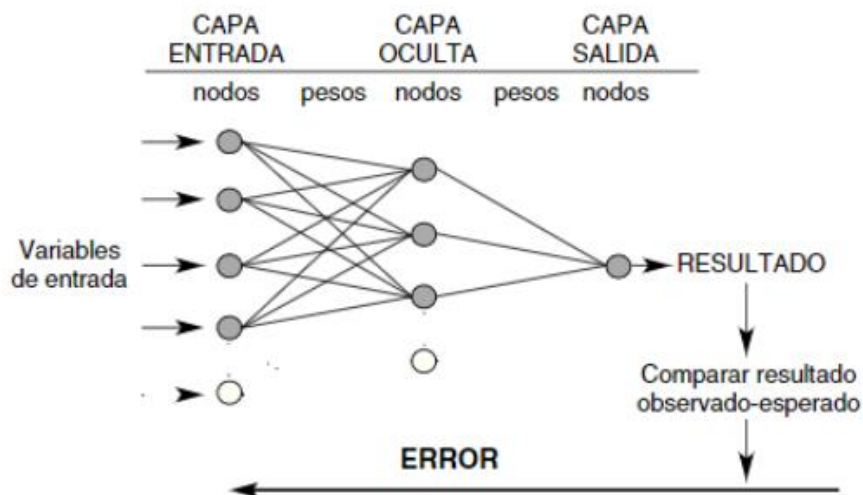


Figura 20. Estructura de la red neuronal [33].

Como se observa en la Figura 20, la estructura de la red neuronal de varias entradas conformado por varias unidades, su estructura parte de las variables de entrada es decir los datos de las imágenes a predecir.

Cada capa se encuentra unido mediante nodos que son los parámetros internos (Pesos/Weights) que son obtenidos como resultado del entrenamiento con las imágenes de muestra, dichos parámetros van variando dependiendo de la conformación de la red neuronal. Existen dos tipos de redes neuronales:

Redes Neuronales recurrente: Realiza conexiones de retroalimentación entre neuronas de la misma capa e incluso con otras capas [34]. Lo que permite que las neuronas tienen memoria lo que permite una mayor tasa de exactitud.

Redes Neuronales no recurrente: La diferencia con las redes neuronales recurrentes vienen dado por las conexiones son del tipo unidireccionales[34], es decir no cuentan con retroalimentación, lo que las hace contar con menos exactitud.

Para elegir el número de capas no hay una respuesta afirmativa, dado que si se utilizan demasiadas capas será muy compleja y se puede producir overheating, mientras que si es muy simple la exactitud puede ser baja [33].

1.2.8.1. Local binary patterns (LBP)

Es una de las técnicas más utilizada para la clasificación de objetos por visión artificial, su principio de funcionamiento se basa en la filtración de cada uno de los pixeles de una imagen con sus pixeles vecinos [17], obteniendo un valor numérico con el objetivo que pueda ser analizado por un computador.

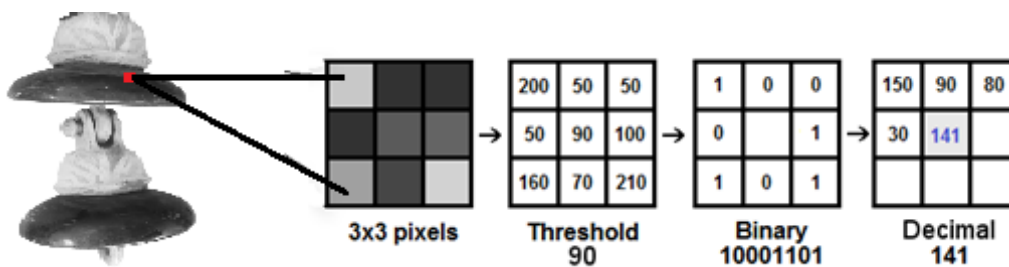


Figura 21. Proceso LBP. [17]

Como se observa en la Figura 21, se toma un píxel de imagen seleccionada para analizarlo con los píxeles vecinos, en este caso se toman 8 píxeles vecinos para obtener una matriz 3x3 al ser analizados se procede a aplicar una función escalón denominado valor de Threshold o umbral, al aplicar dicha función escalón nos entrega un valores binarios (0 o 1), que puede ser interpretado por un computador y finalmente calcular el valor de histograma en decimales, finalmente con ayuda de estadística se conecta para encontrar un vector característicos LBP de la imagen completa y posteriormente utilizar un algoritmo de clasificación.

Todos estos datos se obtienen a partir de la siguiente fórmula:

$$LBP(X_c, Y_c) = \sum_{p=0}^{p-1} 2^p s(i_p - i_c)$$

Donde:

$(X_c - Y_c)$ = Ubicación del pixel central

I_c = Valor de la Intensidad de grises

I_p = Valor de la intensidad de grises del pixel adyacente

p = Numero de pixeles en la imagen

S = Función de activación

$$S(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ 0 & \text{si } x < 0 \end{cases}$$

Se observa que la operación se realiza desde el pixel 0 hasta el n-1, es decir se realiza un barrido completo de número total de pixeles en la imagen a analizar.

1.2.8.2. Algoritmos de detección de imágenes

- **Algoritmo you only look once (YOLO)**

El algoritmo Yolo utiliza una red neuronal convolucional se diferencia de otros algoritmos dado que posee la ventaja que puede predecir varias clases en una sola imagen mediante cuadros delimitadores analizando todas las características de la imagen. Yolo divide la imagen de entrada en una cuadrícula SxS según el número de pixeles de la imagen de entrada.

Cada celda se encarga de realizar la probabilidad de que exista un objeto dentro de él, la predicción realizada se encuentra conformado por 5 elementos:

$$(x, y, w, h, k)$$

Donde:

(x, y) = Coordenadas del centro de la celda

(w, h) = Tamaño del cuadro de la predicción

k = Valor de la probabilidad de la predicción

El algoritmo mediante la herramienta Intersección sobre la Unión (IOU) por sus siglas en inglés, permite calcular un valor de intersección entre la predicción y la anotación a realizar [35] como se observa en la Figura 24.

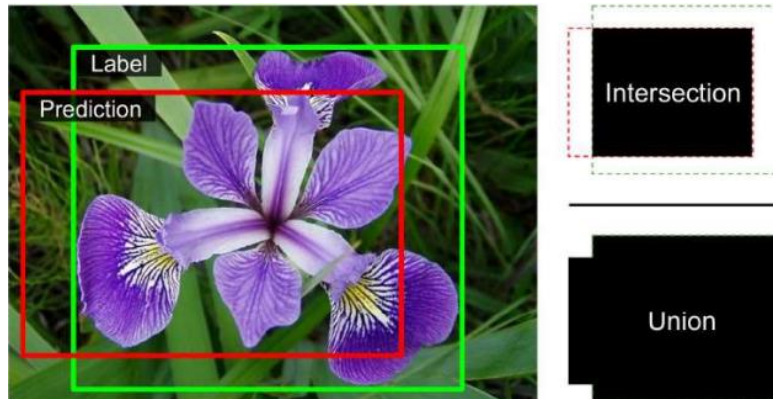


Figura 24. Cálculo de IOU. [35]

Al ser calculado el IOU, se procede a realizar la operación para obtener el valor de confianza del objeto, el cual va a depender de varias probabilidades, dado que cada celda predice C probabilidades de clases encontradas. Adicionalmente se toma en consideración el número de cuadros delimitadores B . Al realizar la detección, se multiplica la probabilidad de clase por la probabilidad de celda, dando como resultado las puntuaciones de confianza.

$$Pc = Pr(Class) * Pr(Obj) * IOU$$

Donde:

$Pr(Class) = Probabilidad\ de\ la\ Clase$

$Pr(Obj) = Probabilidad\ del\ Objeto$

$IOU = Valor\ IOU$

Se observa en la Figura 25 el proceso de identificación de Yolo, donde se parte de una imagen dividida en $S \times S$ píxeles, posteriormente se crea varias cuadrículas para identificar los objetos, se analiza cada cuadrícula para comparar si están prediciendo un mismo objeto que está dentro de las clases pre-entrenadas, finalmente arroja como resultado la misma imagen de entrada con las predicciones encerradas en un cuadrado.

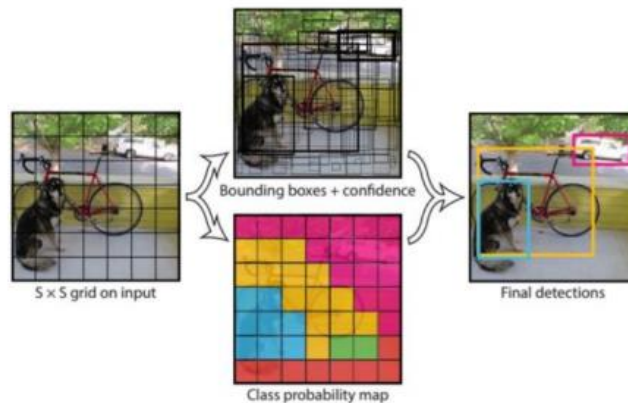


Figura 25. Probabilidad de Yolo. [18]

- **Algoritmo adaboost**

Algoritmo basado en clasificadores simples para producir la predicción del objeto, por ende de clasificadores débiles, dando como resultado un clasificador robusto mediante la combinación lineal de dichas probabilidades lo conforman [17]. Su entrenamiento se basa en el enteramiento de clasificadores débiles de manera iterativa para minimizar el error.

Al dar un peso mayor en cada iteración se busca que las clases mal clasificadas se les da preferencia al momento del aprendizaje, mientras que las que fueron bien clasificadas se les da un peso menor dado que su clasificación se encuentra aceptable, en la Figura 26 se muestra el procedimiento de clasificación del algoritmo.

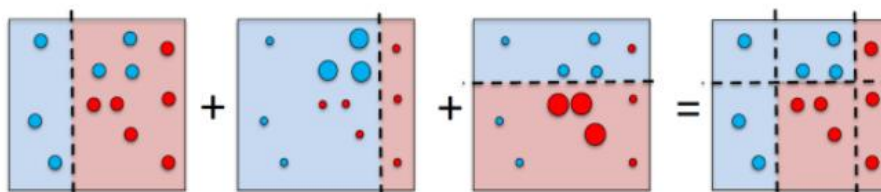


Figura 26. Procedimiento Adaboost [17].

El algoritmo se basa en definir las entradas, posteriormente se determina el número de iteraciones necesarias para la clasificación y se procede con el análisis junto con los pesos obtenidos con el entrenamiento, dicho procedimiento viene dado por:

Input: Dataset $D = \{d_1, d_2, \dots, d_n\}$, $d_i = (X_i, Y_i)$ donde $X_i \in X, Y_i \in \{-1, +1\}$

Output: A clasificado $H; X \rightarrow \{-1, +1\}$

Los pasos a considerar son:

1. Inicialización de los pesos $w_i = 1/N, i \in \{1 \dots N\}$
Desde $m=1$ hasta M se realiza:

2. Se estima la probabilidad estimada mediante:

$$P_m(x) = P_w(y = 1/x), \text{ usando } w_i$$

3. Se calcula el peso de la iteración actual:

$$H_m = \frac{1}{2} \log((1 - P_m(x))P_m(x)) \in R$$

4. Se actualiza el peso:

$$w_i \leftarrow w_i \exp(-y_i H_m(x_i))$$

5. Reconstruir los pesos:

6. Obteniendo el resultado de la clasificación mediante:

$$H(x) = \text{sign} \left(\sum_{j=1}^M \alpha_j H_j(x) \right)$$

Al seguir dichos pasos se obtiene una formula final en la cual se procede a sumar todas las probabilidades débiles obtenidas al ingresar los atributos de las imágenes y finalmente encontrar la hipótesis fuerte, por ende, el objeto dentro de la imagen.

1.2.8.3. Yolo vs adaboost

Al analizar el funcionamiento de cada uno de los logaritmos se definen las siguientes ventajas y desventajas de cada uno:

Tabla 3. Ventajas y desventajas Yolo vs Adaboost.

	Yolo	Adaboost
Ventajas	<ul style="list-style-type: none"> -La velocidad media de detección en video es de 45 FPS. -Alta tasa de exactitud -Bajo tiempo de procesamiento de datos. 	<ul style="list-style-type: none"> -Sencillo de programar -Versátil y rápido. -No requiere conocimiento previo.
Desventajas	<ul style="list-style-type: none"> -Al detectar un objeto mediante celdas, es posible que se pierda algún tipo de elemento. -Es necesario poseer una GPU para alcanzar una tasa aceptable de FPS. 	<ul style="list-style-type: none"> -Es vulnerable al ruido. -Bajo rendimiento en video -Los clasificadores débiles lo hacen propenso a overfitting.

Al analizar Tabla 3 se observa que Yolo cuenta con varias ventajas sobre el algoritmo Adaboost al analizar objetos en videos y poseer una alta exactitud, por ende, es la opción más acertada para el desarrollo de identificación de aisladores en tiempo real.

1.2.8.4. Evaluación del algoritmo

Para evaluar el algoritmo creado, se utilizará una herramienta matemática denominada como la matriz de confusión [36], los resultados obtenidos pueden ser de 4 tipos:

- **Verdadero positivo (TP):** El valor real es positivo, mientras que el algoritmo lo predijo positivo.
- **Falso positivo (FP):** El valor real es negativo, mientras que el algoritmo lo predijo positivo.
- **Verdadero negativo (TN):** El valor real es negativo, mientras que el algoritmo lo predijo negativo.
- **Falso negativos (FN):** El valor real es positivo, mientras que el algoritmo lo predijo negativo.

Según se muestra en la **¡Error! La autoreferencia al marcador no es válida.**, el número total de resultados positivos es la suma de falsos negativos y verdaderos positivos, mientras que los resultados negativos es la suma de falsos positivos más verdaderos negativos.

Tabla 4. Matriz de confusión. [36]

Valor Real	Predicción	
	μ_1	μ_2
μ_1	TP	FN
μ_2	FP	TN

Cuando se cuenta con una red neuronal multiclase, es decir más de dos clases, es necesario utilizar una variación de la matriz de confusión [36], aquí se pueden obtener varios resultados:

- **TP(A,B,C):** Verdaderos positivos de cada muestra.
- **E_{AB}:** Muestras de la clase A que fueron clasificados de manera errónea como clase B.
- **E_{AC}:** Muestras de la clase A que fueron clasificados de manera errónea como clase C.
- **E_{BA}:** Muestras de la clase B que fueron clasificados de manera errónea como clase A.

- E_{BC} : Muestras de la clase B que fueron clasificados de manera errónea como clase C.
- E_{CA} : Muestras de la clase C que fueron clasificados de manera errónea como clase A.
- E_{CB} : Muestras de la clase C que fueron clasificados de manera errónea como clase B.

Como se puede apreciar en la Tabla 5, según sea el número de clases aumentara la dimensión de la matriz, los valores verdaderos positivos se muestran de color verde.

Tabla 5. Matriz de confusión multiclasa. [36]

Valor Real	Predicción		
	μA	μB	μC
μA	T_{pA}	E_{BA}	E_{CA}
μB	E_{AB}	T_{pB}	E_{CB}
μC	T_{AC}	E_{BC}	T_{pC}

1.2.8.5. Exactitud

La exactitud o accuracy representa el porcentaje de predicciones correctas frente al total de la muestra, puede verse afectada por el nivel de ruido y el nivel de correlaciones entre variables de entrada y salida [36]. Para encontrar el porcentaje de exactitud del clasificador se utilizará la siguiente formula:

$$Exactitud = \frac{Tp}{T} * 100\%$$

Donde:

Tp = Sumatoria de verdaderos positivos de cada muestra.

T = Total de Muestras.

1.2.8.6. Sensibilidad

La sensibilidad representa la porción de casos positivos de cada clase que fueron encontradas de forma correcta por la red neuronal, es decir la tasa de aciertos que tuvo, en caso de redes neuronales multiclasa, el cálculo se lo realiza clase por clase y no en conjunto como la exactitud [37], se utiliza la siguiente formula:

$$Sensibilidad(i) = \frac{Tp(i)}{Tp(i) + FN(i)}$$

Donde:

$Tp(i)$ = Verdaderos Positivos de la Clase i

$FN(i)$ = Falsos Negativos de la Clase i

1.2.8.7. Posibles fuentes de error

El error en la exactitud que se pueda presentar en una red neuronal poder deberse a varios factores [38], se calcula mediante la siguiente formula:

$$Error = (1 - Exactitud) * 100 \%$$

- Errores en la entrada de datos al ingresar a la red neuronal.
- Variación en la clasificación y delimitación de los datos de referencia
- Errores en la clasificación de los datos de la base de datos.
- Modelo mal acondicionado debido a que la red neuronal puede estar en saturación.
- Overfitting y Underfitting

1.3. Fundamentación metodológica

Para obtener el conocimiento necesario para realizar y culminar el proyecto de investigación, además de comprobar la hipótesis formulada, incluido los objetivos trazados el planteamiento metodológico será del tipo cuantitativo.

Debido a que dicho enfoque es secuencial y probatorio, es decir, cada etapa viene precedida de la anterior no hay atajos ni se puede eludir pasos, utiliza frecuentemente mediciones numéricas y análisis estadísticos para establecer comportamientos y probar teorías [39].

El alcance que se ha elegido es del tipo correlacional debido, que se va a comprobar la fiabilidad de un algoritmo para reconocimientos de imágenes y elegir el mejor software para aplicarlo al reconocimiento de aisladores eléctricos.

Este tipo de estudios tiene como objetivo conocer la relación que existe entre dos o más conceptos, categorías o variables en un contexto en particular, evalúan el grado de relación entre dos o más variables dichas correlaciones se sustentan en hipótesis sometidas a pruebas [39].

Para la implementación de la interfaz de reconocimiento de aisladores eléctricos, se va a trabajar con diferentes aisladores de porcelana del tipo campana, es importante tener conocimientos de visión artificial, programación e informática, por lo cual se seguirá los lineamientos del método analítico-sintético.

El método analítico-sintético se basa en el estudio de los hechos, puesto que se parte de la descomposición del objeto de estudio en cada parte para analizarlas en forma individual y luego integrar las partes para estudiarlas de manera holísticas e integral [40].

Adicionalmente una de las técnicas a utilizar será del tipo observación directa debido a que en la actualidad es una de las más utilizadas, debido a que permite obtener información directa y confiable, siempre y cuando se realice mediante un proceso sistematizado [40], debido a que todo el procesamiento de información se basa en visión artificial, en conjunto a esta técnica se empleará internet siendo la mayor fuente para recabar información.

1.4. Conclusiones Capítulo I

- Toda la información del Capítulo 1, la cual fue recabada de fuentes confiables y verificadas, se obtuvo un amplio conocimiento respecto a las líneas de transmisión en especial de los aisladores eléctricos y sus principales fallas.
- Al seleccionar Yolo como algoritmo para la identificación del estado de los aisladores en una línea de transmisión se busca que el entrenamiento sea sencillo y la tasa de error sea muy baja.
- La hora más adecuada para realizar las pruebas de campo, es antes del mediodía dado que la luz solar no ha alcanzado el punto máximo y no podría afectar a la toma de video y fotografías con el dron.

CAPÍTULO II

2. PROPUESTA

En el presente capítulo se detalla cada una de las etapas del desarrollo de la investigación, adicionalmente de los materiales requeridos para su finalización.

2.1. Título de la propuesta.

Identificación de fallas en los aisladores de una línea de transmisión mediante visión artificial.

2.2. Objetivo del proyecto.

Identificar las fallas de manera autónoma en los aisladores de porcelana de la red de transmisión 13.8 kV mediante visión artificial para su posterior clasificación según su estado.

2.3. Descripción de la propuesta.

La presente investigación se encuentra dirigida a las empresas encargadas de las líneas de transmisión y/o redes de distribución en las cuales se cuente con aisladores eléctricos en sus diferentes tipos, se contará con varias etapas partiendo con una investigación bibliográfica, adquirir y crear una base de datos con imágenes de aisladores, diseño del algoritmo de clasificación con la finalidad de realizar pruebas de campo para poder identificar las posibles fallas, el estudio se llevara a cabo en las ciudades de Latacunga y Salcedo, la investigación esta limitada solo al análisis de la superficie de los elementos.

Para la detección sea sencilla para el algoritmo se proponen 3 estados:

- Bueno.
- Roto.
- Sucio/Flameado.

En la Tabla 6, se muestran los estados propuestos para la investigación, se toma en consideración los flameos como parte de los aisladores sucios, dado que, dicha falla genera una pequeña capa de hollín en la superficie del aislador y para el algoritmo es imposible diferenciarlos, en aisladores rotos se incluyen todas las roturas y/o fisuras que se puedan presentar y los aisladores buenos todos aquellos no necesitan ningún mantenimiento.

Tabla 6. Estado de aisladores propuestos.

Estado del Aislador	Características	Mantenimiento Requerido
Bueno	Aislador en óptimas condiciones, no presenta ninguna suciedad o flameos y/o rotura.	No
Sucio/ Flameado	Aislador con presencia de suciedad orgánica e inorgánica en su superficie. Aislador con presencia de hollín en su superficie generado por flameos previos.	Preventivo / Correctivo
Roto	Aislador con presencia de roturas y/o fisuras en su superficie.	Correctivo

Para la muestra de dichos aisladores dentro del algoritmo se propone una manera sencilla de visualizar dichos estados. En la Tabla 7, cada estado de los 3 posibles, cuentan con una etiqueta y color único que permitirá realizar el análisis de los elementos de una manera más sencilla para el operador.

Tabla 7. Representación de estados.

Estado del Aislador	Representación dentro del Algoritmo	Etiqueta
Bueno	Cuadrado Verde	Bueno
Sucio/Flameado	Cuadrado Morado	Sucio/ Flameado
Roto	Cuadrado Celeste	Roto

Dichas fallas serán analizadas solo en la superficie del elemento, dado que, la distancia de seguridad no permite acerca el dron para analizar las fallas internas que se puedan presentar, adicionalmente las fallas internar como suciedad o rotura con el paso del tiempo se harán presentes en la superficie [23], lo que haría posible realizar el análisis.

2.3.1. Concepción del diseño

Para poder llevar a cabo el diseño se toma en consideración varios pasos como se muestra en la Figura 27, la adquisición de imágenes se lo realizara mediante un cuadricóptero no tripulado (dron) para proceder a crear una base de datos de aisladores que presenten diferentes problemas.

Se procederá a realizar un preprocesamiento de las imágenes aplicando diferentes técnicas de visión artificial y encontrar puntos característicos, estos atributos sirven para realizar la clasificación para finalmente mostrar los resultados es decir la identificación de la falla.

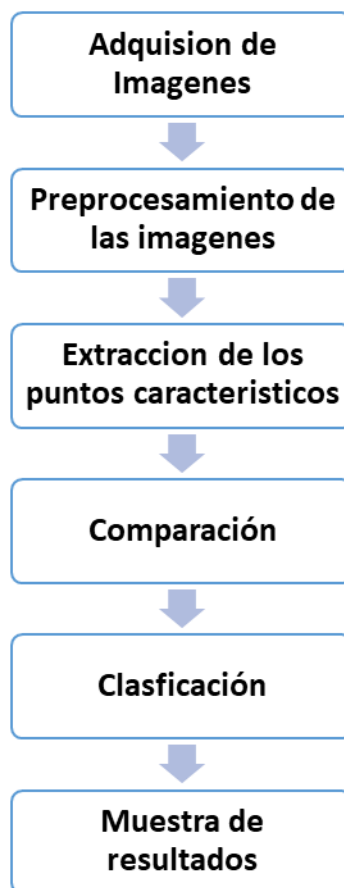


Figura 27. Arquitectura de la investigación.

2.3.2. Adquisición de imágenes

La adquisición de imágenes para la base de datos se lo realizara mediante un cuadricóptero aéreo no tripulado (dron) para obtenerlas de los postes, el procedimiento a seguir, se verifica el equipo y todos sus componentes, se lo enciende y se lo posiciona a una distancia segura para que no exista interferencia con la línea de transmisión.

2.3.2.1. Selección del dron

Para llevar a cabo la investigación es necesario analizar que dron es el apropiado para la captura de imágenes, tomando en consideración los siguientes requerimientos:

- Poseer cámara incorporada
- Buena calidad de imagen
- Comunicación Wifi
- Distancia de Vuelo
- GPS
- Resistencia a la velocidad del viento

En el mercado existen gran variedad de marcas y modelos de drones, pero los que más se ajustan a las características deseadas son los drones de la marca Dji, los cuales cuentan con varios modelos, pero para el desarrollo de la investigación se optó por utilizar el modelo Dji Mini 2, dado que cumple con todos los requerimientos deseados.

Adicionalmente se seleccionó el Dji Mini 2, dado que supera la velocidad promedio del viento de la provincia de Cotopaxi de 7m/s [41].

Tabla 8. Características del dron elegido [42].

Características	
Calidad de la Cámara	4K: 3840x2160/ 24-30 fps 2.7K: 2720x1530 / 24-60 fps FHD: 2720x1530 / 24-60 fps
Comunicación	Wifi / Bluetooth
Distancia de Vuelo	10 Km
GPS	Precisión: $\pm 0.5 m$
Resistencia Máxima al Viento	10 m/s

Como se muestra en la Tabla 8, las características del dron elegido son aceptables, tomando en consideración que el equipo cuenta con un peso reducido, tiene cámara incorporada con una excelente calidad de imagen llegando a una calidad de 4K, además cuenta con GPS incorporado y la comunicación es WIFI lo que permitirá la comunicación con el computador y la red Neuronal. El Datasheet completo del dron Dji Mini 2 se puede observar de manera más detallada en el Anexo A.



Figura 28. Dron Dji mini 2.

Una vez seleccionado el dron se procede a seleccionar varios tramos de la línea de transmisión de 13.8 kV de la ciudad de Latacunga donde se encuentren aisladores cerámicos. Al seleccionar el tramo deseado se procede a posicionar el dron, se buscará el mejor ángulo posible para poder obtener fotografías de los elementos deseados, tomando en consideración la investigación bibliográfica, la cámara no debe encontrarse a contraluz, ni debe existir sombras que puedan incurrir en la pérdida de las características al momento de obtener las imágenes.

2.3.2.2. Distancia de seguridad del dron

En la Figura 29 se observa un ejemplo del posicionamiento del dron respecto a los aisladores, se trata de colocar el aislador en el centro de la fotografía para poder obtener la mayoría de las características.

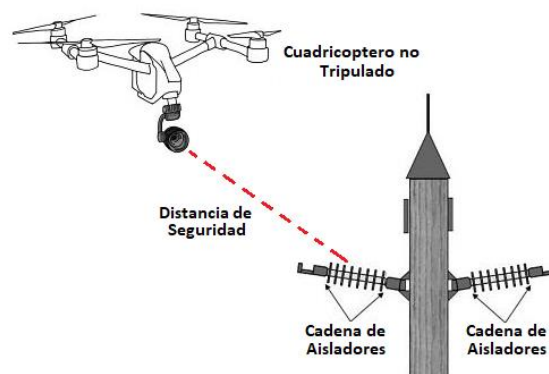


Figura 29. Posicionamiento del dron.

La distancia de seguridad para el posicionamiento del dron fue obtenida de la Regulación Arconel 001/18 [43] de las páginas 8 y 9, en la cual se presenta las distancias de seguridad ante cables energizados dependiendo del voltaje de los mismos, para la presentación se tomó en consideración el apartado de distancias mínimas de seguridad horizontales para antenas de radio y televisión, chimeneas, carteles, etc.

En la Tabla 9 y Tabla 10 se muestra las distancias de seguridad horizontal para antenas de televisión y radio, por ende, se los tomara en consideración para el vuelo del dron, dado que la adquisición de imágenes y pruebas de campo se las realizara en la línea de 13.8 kV, se optó por utilizar una distancia de seguridad aproximada de 2 metros para que no exista ningún tipo de contratiempo ante la presencia o ausencia de viento, dado que la distancia mínima de seguridad de un conductor es de 1.7 metros, esta distancia es uno de los factores por la cual no se puede analizar el interior de los aisladores, dado que no se puede obtener un ángulo adecuado para obtener dichas características.

Tabla 9. Distancias mínimas de seguridad de conductores energizados [43].

Voltaje de la Línea	Distancia de Seguridad horizontal (HR), de conductores en reposo(m)
0 a 750 V	1.7
750 V a 22 Kv	2.3
Mayores a 22 kV	$H_R = 2.3 + 0.01 * (V - 22)$ Donde V se encuentra en kV

Tabla 10. Distancias mínimas de seguridad de conductores energizados con viento[43].

Voltaje de la Línea	Distancia de Seguridad horizontal (HR), de conductores en desplazados por el viento(m)
0 a 750 V	1.1
750 V a 22 kV	1.4
Mayores a 22 kV	$H_R = 1.4 + 0.01 * (V - 22)$ Donde V se encuentra en kV

2.3.3. Base de datos.

Para empezar con el desarrollo de la investigación se procedió a crear la base de datos con las imágenes obtenidas con ayuda del dron y posteriormente clasificarlos de manera manual según el tipo de falla (roto, sucio, bueno), para finalmente ubicarlos en carpetas individuales.

Dado que no existe un número específico de muestras para el entrenamiento de redes neuronales [44], en diferentes investigaciones se realizar con un numero de acuerdo a sus propias características, tomando en consideración la dificultad para conseguir dichas imágenes, experiencia del investigador, etc.

Para la presente investigación se tomará en consideración la regla de 1000 muestras por clase [44] [18], tomando en consideración que en las investigaciones que se tomó como punto de partida esta regla se obtuvieron buenos resultados.

La base de datos constara de 3000 imágenes de aisladores de cerámica utilizados en la línea de transmisión de 13.8 kV distribuidas en los estados de los aisladores: Roto, Sucio y Bueno, las cuales fueron obtenidas en los postes pertenecientes a la Empresa Eléctrica Provincial Cotopaxi (ELEPCO S.A) y adicionalmente de sus bodegas, las cuales serán sometidas a un procesamiento digital para eliminar objetos y/o imperfecciones ajenas al aislador como el fondo.

2.3.4. Preprocesamiento de imágenes

Esta es la etapa primordial para que la investigación tenga éxito, considerando que se realiza el mejoramiento de las imágenes de los aisladores a detectar tomadas por el dron.

Gracias a la investigación bibliográfica que se realizó en el Capítulo 1, se encontró que es necesario realizar un preprocesamiento de las imágenes aplicando diversos filtros, para eso se eligió utilizar Python debido a que cuenta con librerías especializadas para el tratamiento de imágenes, con lo cual se podrá detectar los bordes, formas y colores de los aisladores.

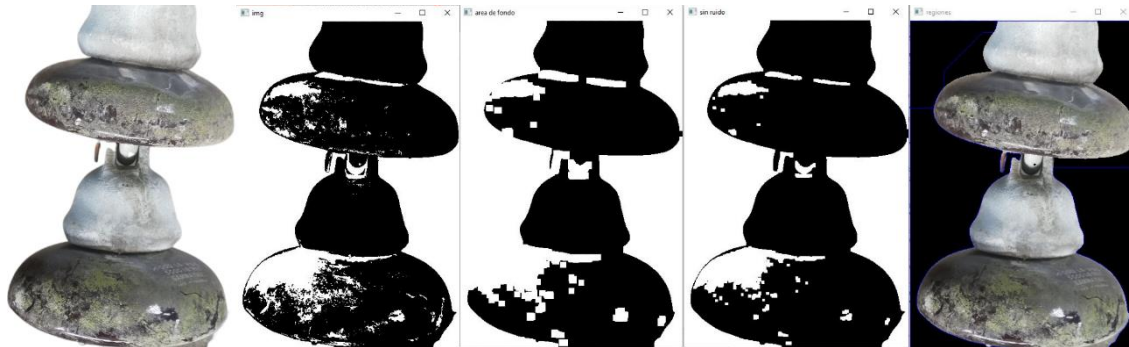


Figura 30. Preprocesamiento de las imágenes.

En la Figura 30, se observa como la imagen fue tratada mediante filtros para la eliminación de ruido y todos los elementos ajenos al aislador, así como también resaltar las características del elemento, se utilizaron técnicas de segmentación, binarización, bordes, eliminación de fondo.

Este paso es uno de los más importantes puesto que para posteriores etapas no sea contaminado con objetos ajenos al aislador.

Para la extracción de las características del aislador se utilizó la herramienta transformación de características de escala invariable SIFT por sus siglas en inglés que es parte de la librería OpenCV, dicho algoritmo nos permite detectar puntos clave sin importar la escala de la imagen, el proceso se muestra en la Figura 31.

Para realizar el proceso de puntos características, se declara las librerías a utilizar y las variables necesarias para el proceso, posteriormente se leen las imágenes deseadas mediante la cámara, posteriormente se realiza un redimensionamiento para que sea más fácil visualizar los resultados, se aplica la segmentación para resaltar la mayor cantidad de atributos para finalmente encontrar los puntos importantes de la imagen.

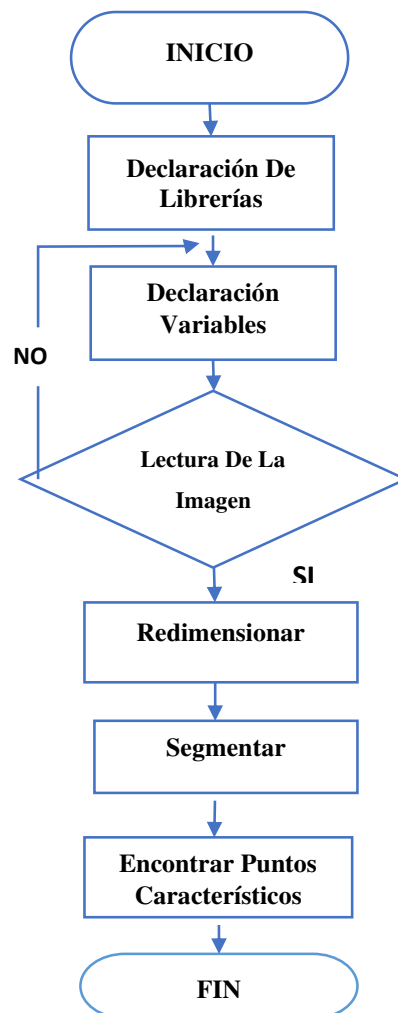


Figura 31. Proceso para encontrar puntos característicos.

2.3.5. Entrenamiento

Para la etapa de entrenamiento del algoritmo se realiza como primer paso un etiquetado de las imágenes preprocesadas, es decir, enseñarle al algoritmo que es un aislador y cuál es su estado, para lo cual se utilizara LabelImg herramienta disponible en Python.

LabelImg es una interfaz gráfica con la cual se puede realizar anotaciones de imagen y obtenerlos en archivo XML o TXT, dentro de los cuales se guardará la etiqueta de la imagen, así como sus coordenadas y tamaño, en la Figura 32 se muestra el etiquetado de las imágenes.

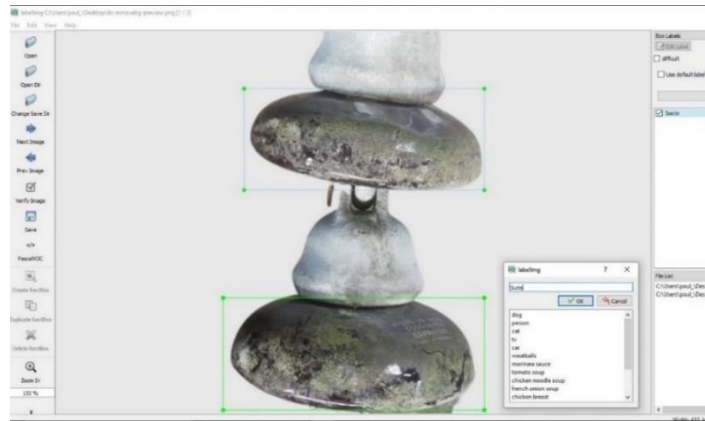


Figura 32. Etiquetado de imágenes.

Una vez realizado el etiquetado de las imágenes de cada estado (roto, sucio, bueno), se procede a entrenar a la red neuronal, es decir, que sepa diferenciar el estado que posee cada aislador, las imágenes etiquetadas son guardadas en la base de datos, posteriormente con la ayuda del procesador gráfico se procede al proceso de entrenamiento, esta etapa puede durar algunas horas dependiendo de la capacidad del procesador así como también el número de imágenes, finalmente se obtiene un archivo weights o pesos que está compuesto por 24 capas de información, el proceso se muestra en la Figura 33.

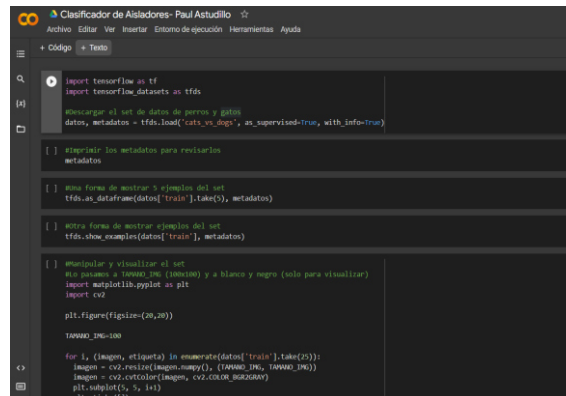


Figura 33. Proceso de entrenamiento.

Para continuar con el proceso de entrenamiento se realizó un análisis para identificar el valor del porcentaje adecuado de la muestra que debe ser usado para el entrenamiento y validación de la red neuronal, varios autores en sus investigaciones sugieren varios porcentajes 75-25 [17],70-30 [18],80-20 [45], al realizar una comparación de dichos porcentajes para el entrenamiento se pudo conocer que el mejor rendimiento de la red neuronal fue al ser entrenado con un porcentaje de 70-30 [46], por tanto será el elegido para la red neuronal de identificación de fallas en los aisladores.

Para finalizar el proceso de entrenamiento se utilizó Google Colab, herramienta disponible de manera gratuita en la cual se puede utilizar una tarjeta de video virtual, con el objetivo de disminuir el tiempo que llevara el proceso, se prevé que sea necesario 1000 épocas completar el entrenamiento

El proceso se lleva a cabo subiendo las imágenes junto con sus etiquetas a la nube de Google y se procede al entrenamiento, el código se puede observar en el Anexo B.



```
import tensorflow as tf
import tensorflow_datasets as tfds

#descargar el set de datos de perros y gatos
datos, metadatos = tfds.load('cats_vs_dogs', as_supervised=True, with_info=True)

#inspeccionar los metadatos para revisarlos
metadatos

#una forma de mostrar 5 ejemplos del set
tfds.as_dataframe(datos['train'].take(5), metadatos)

#otra forma de mostrar 5 ejemplos del set
tfds.show_examples(datos['train'], metadatos)

#manipular y visualizar el set
#si quieres a tener un 0 (blanco) y a blanco y negro (solo para visualizar)
import matplotlib.pyplot as plt
import cv2

plt.figure(figsize=(20,20))

TAMANO_IMG=100

for i, (imagen, etiqueta) in enumerate(datos['train'].take(25)):
    imagen = cv2.resize(imagen.numpy(), (TAMANO_IMG, TAMANO_IMG))
    imagen = cv2.cvtColor(imagen, cv2.COLOR_BGR2GRAY)
    plt.subplot(5, 5, i+1)
    plt.imshow(imagen)
```

Figura 34. Proceso de entrenamiento Google Colab.

Al utilizar Google Colab como herramienta de entrenamiento es posible evitar el sobre entrenamiento o overfitting de manera sencilla, dado que durante cada iteración de entrenamiento se realiza un cálculo de perdida (comparación entre valor esperado y valor obtenido), la herramienta al encontrar un valor menor a 2%, se detiene automáticamente y da el proceso de entrenamiento finalizado [47].

Como se observa en la Figura 35, con la ayuda de la etapa de preprocesamiento y el entrenamiento de la red neuronal es posible encontrar un aislador y encerrarlo en un cuadrado en negro para poder diferenciarlo del resto de la imagen.

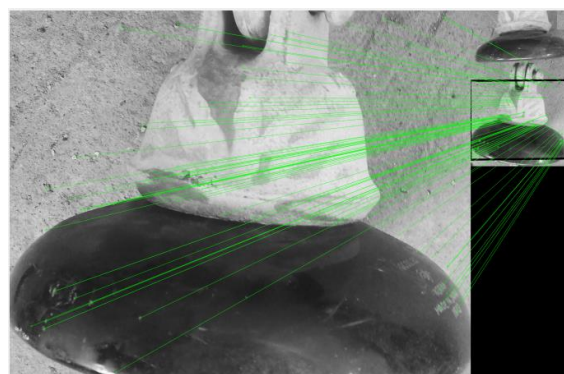


Figura 35. Localización de aisladores.

pc = Probabilidad de Objeto
 bx, by = Coordenadas x, y del pixel central.
 bh = Valor de la altura del cuadro contenedor
 bw = Valor de ancho del cuadro contenedor
 $c1$ = Clase 1
 $c2$ = Clase 2
 $c3$ = Clase 3

Para el cálculo de dichos datos se aplican las siguientes formulas, se tomando en consideración la Figura 22:

$$\begin{aligned}
 bx &= \frac{A - D}{D} \\
 by &= \frac{\left(\frac{E}{S} - C\right)}{C} \\
 bw &= \frac{B}{E} \\
 bh &= A/F
 \end{aligned}$$

Para la investigación se toma en consideración el tamaño de la imagen de entrada, de igual manera el número de cuadrículas. Al culminar con los cálculos de todos los pixeles se procede a aplicar la primera tanda de capas convolucionales.

La predicción de cada celda de un número de BB (cajas contenedoras), donde probablemente se encuentre el objeto, cada celda entrega coordenadas (x,y,w,h) y una tasa de confianza o predicción $(c1,c2,c3)$. Yolo utiliza 3×3 cuadrículas por cada celda, en las que se busca objetos dentro de ella, dado que poseemos tres clases, cada celda tendrá un vector de dimensiones, viene dado por:

$$Pc = Pr(Class) * Pr(Obj) * IOU$$

Donde:

$Pr(Class)$ = Probabilidad de la Clase

$Pr(Obj)$ = Probabilidad del Objeto

IOU = Valor IOU

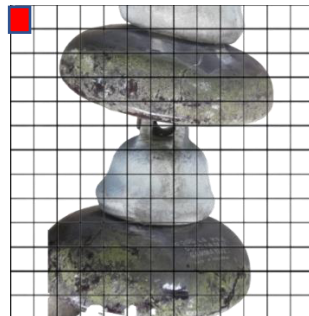


Figura 37. Análisis pixel vacío.

$$Y = SxSxK$$

Donde:

S= 3

A= 2

K=3+5

Al analizar cada una de las celdas, nos dará valores con los cuales se va llenando nuestro vector resultado, por ejemplo, al analizar la celda de la Figura 37, obtendremos el siguiente resultado.

$$Y = \begin{bmatrix} pc \\ bx \\ by \\ bh \\ bw \\ c1 \\ c2 \\ c3 \\ pc \\ bx \\ by \\ bh \\ bw \\ c1 \\ c2 \\ c3 \end{bmatrix} \rightarrow Y = \begin{bmatrix} 0 \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ 0 \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \end{bmatrix}$$

Se observa que no existe ningún objeto de interés dentro de dicha celda, por lo que se la descarta de manera inmediata.

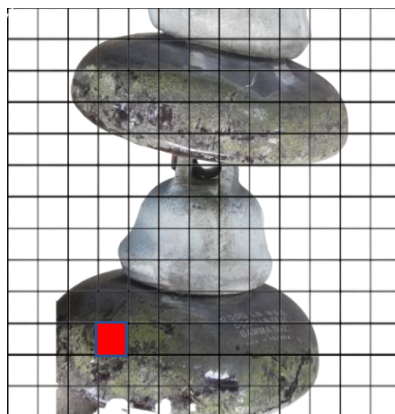


Figura 38. Análisis pixel con objeto.

Al analizar la celda de la Figura 38 se observa que se encuentra una parte del aislador, el vector resultante se muestra a continuación:

$$Y = \begin{bmatrix} pc \\ bx \\ by \\ bh \\ bw \\ c1 \\ c2 \\ c3 \\ pc \\ bx \\ by \\ bh \\ bw \\ c1 \\ c2 \\ c3 \end{bmatrix} \rightarrow Y = \begin{bmatrix} 0 \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ 1 \\ bx \\ by \\ bh \\ bw \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

Se observa que en la segunda ancla se encontró un objeto de la clase c1, de igual manera sus atributos (bx,by,bh,bw) lo que permite encerrarlo en su cuadro contenedor. Dicho proceso se lleva a cabo hasta terminar con cada una de las celdas.

Al aplicar las capas convolucionales de igual manera se reduce la imagen a (112x112) pixeles, con el objetivo de ir aumentando el cuadrículas, por ende, el número de atributos, el proceso se realiza mediante LBP.

$$LBP(bx, by) = \sum_{N=0}^{N-1} 2^p s(i_p - i_c)$$

Donde:

(bx, by) = Ubicación del pixel central

I_c = Valor de la Intensidad de grises

I_p = Valor de la intensidad de grises del pixel adyacente

N = Numero de pixeles en la imagen

S = Función de activacion

Mediante la fórmula de LBP se obtiene los atributos de cada pixel, para posteriormente ir llenando cada espacio del vector de la imagen a detectar. De igual manera Yolo predice los valores de (pc, bx, by, bh, bw, c1, c2, c3). Este proceso se lo realiza hasta cuando se culmine las 24 capas convolucionales, de igual manera se reduce el tamaño de la imagen.

La función de activación se denomina Relu (Rectified Linear Unit), viene dado por.

$$f(x) = x^+ = \max(0, x)$$

Al llegar a la etapa de las capas totalmente conectadas, se busca mapear las características distribuidas durante el proceso de convolución, para colocar los atributos encontrados en una matriz de (7x7x30).

El proceso de predicción finalmente ocurre mediante el análisis del vector final, ver Figura 39.

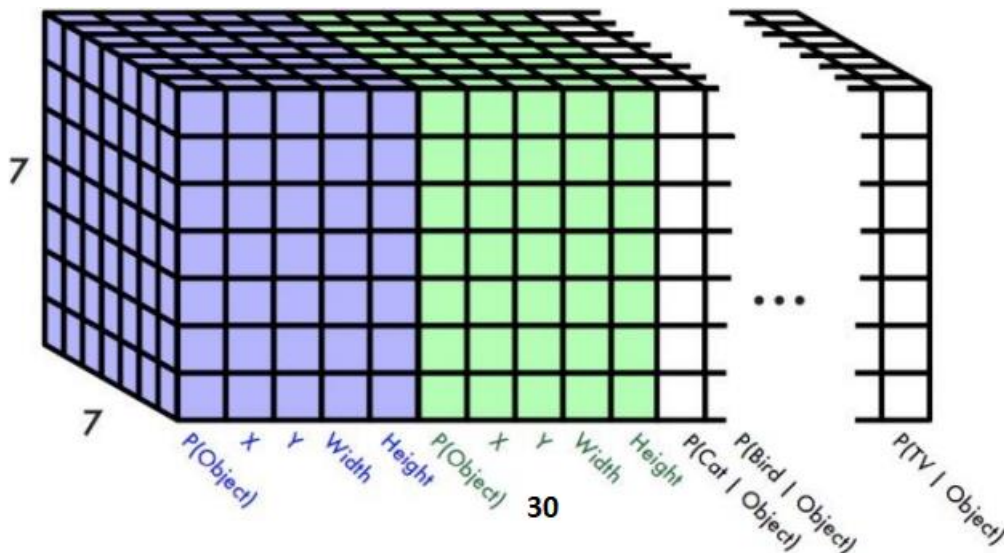


Figura 39. Vector SxSxCxK.

Mediante el análisis de las características obtenidas, de los pesos pre-entrenados, función de transferencia y de activación, nos dará una predicción, que en nuestro caso será un aislador sucio.

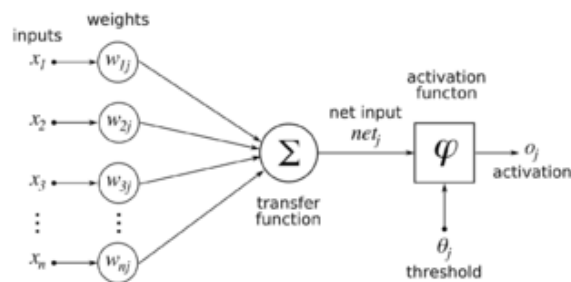


Figura 40. Neurona de clasificación.

$$Y_i(t) = f_i \left(\sum_{j=0}^n w_{ij} * X_j \right)$$

Donde:

X(i) = Entradas de la Neurona

W(ij) = Pesos Asociados a las entradas

f(i) = Funcion de Activación

o(i) = Salida de la Red Neuronal

La tasa de perdida que Yolo utiliza viene definida por:

$$\begin{aligned}
& \lambda_{coord} \sum_{I=0}^S \sum_{J=0}^B l_{ij}^{obj} [(x_i - x'_i)^2 + (y_i - y'_i)^2] + \\
& \lambda_{coord} \sum_{I=0}^S \sum_{J=0}^B l_{ij}^{obj} \left[(\sqrt{w_i} - \sqrt{w'_i})^2 + (\sqrt{h_i} - \sqrt{h'_i})^2 \right] + \\
& \sum_{I=0}^S \sum_{J=0}^B l_{ij}^{obj} (c_i - c'_i)^2 + \\
& \lambda_{noobj} \sum_{I=0}^S \sum_{J=0}^B l_{ij}^{obj} (c_i - c'_i)^2 + \\
& \sum_{J=0}^B l_i^{obj} \sum (p_i(c) - p'_i(c))^2
\end{aligned}$$

Donde:

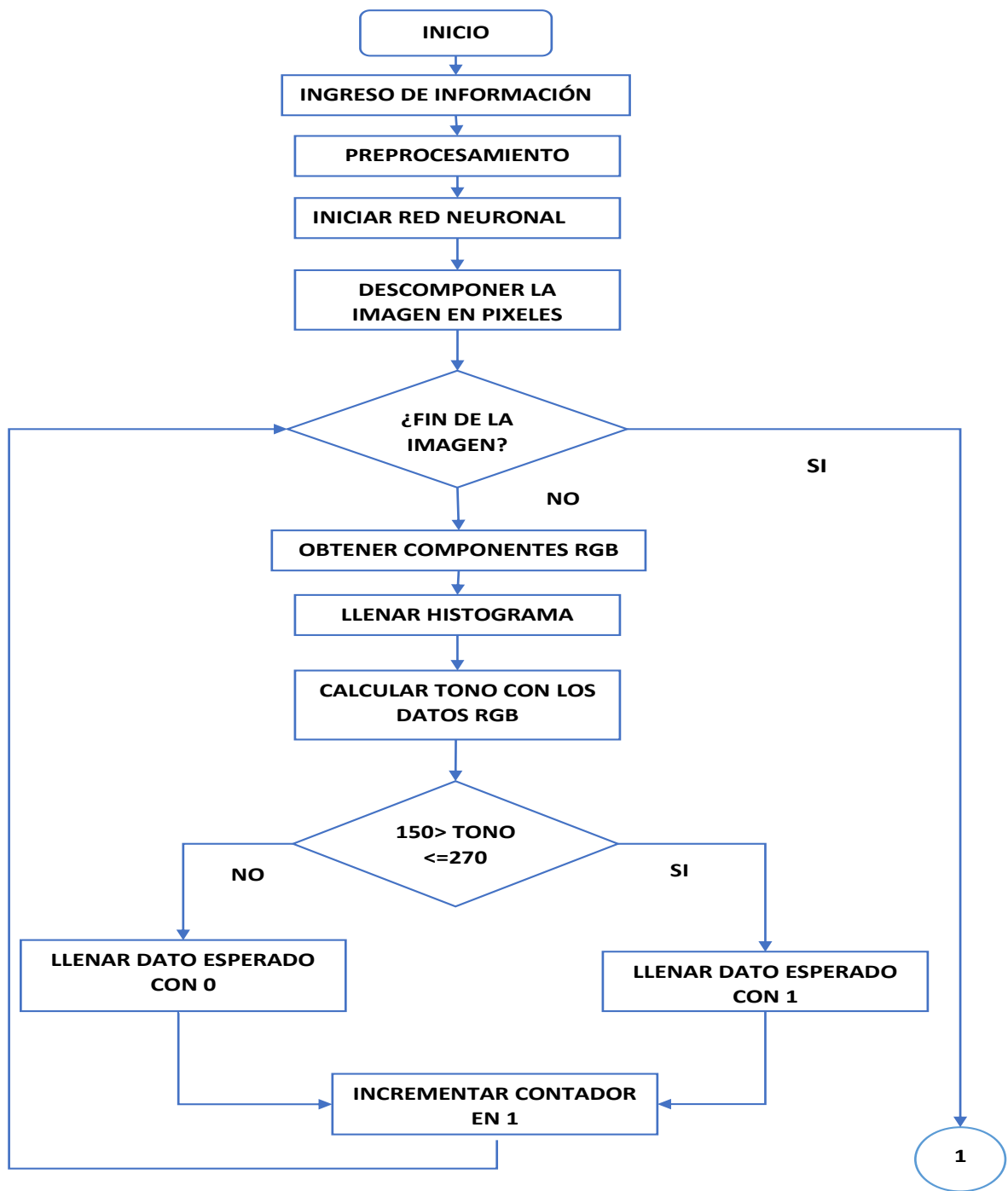
l_i^{obj} = Objeto presente en la celda i

l_{ij}^{obj} = Cuadro delimitador de la celda i

$\lambda_{coord}, \lambda_{noobj}$ = Equilibrar las perdidas, en la investigacion se opto por (5)

Al analizar las dos primeras partes de la ecuación de perdida, representa el error cuadrático de las coordenadas del objeto, así como también el error de tamaño del cuadro delimitador, la tercera parte de la ecuación representa el error entre la predicción deseada con la obtenida, el cuarto apartado calcula el límite de las celdas que no poseen ningún objeto y finalmente el último apartado calcula el error de las clases que fueron obtenidas en la celda actual

El proceso de clasificación se muestra en la Figura 41, con la ayuda del algoritmo realizado en el programa Python, se tomará el video en tiempo real y se lo descompondrá en fotogramas, se procede al preprocesamiento de las imágenes previo al ingreso de los fotogramas a la red neuronal.



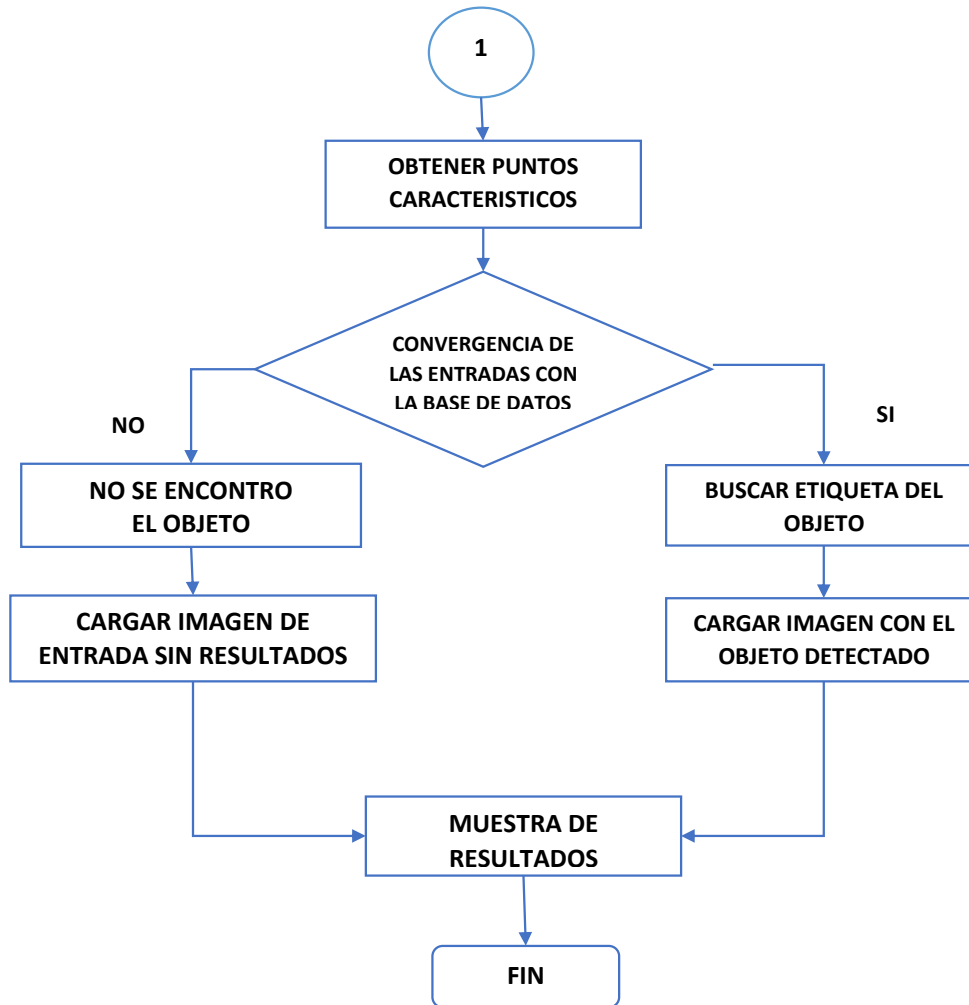


Figura 41. Etapa de clasificación.

Se descompone la imagen de entrada en píxeles, los cuales serán analizados de manera individual encontrando las características RGB, las cuales nos permitirá encontrar los puntos característicos de la imagen, con dichos puntos se procede a buscar alguna convergencia con la base de datos de entrenamiento.

Si se encuentra alguna coincidencia se procede a buscar la etiqueta de dicho objeto, finalmente se procede mostrar la imagen de entrada con el objeto encontrado en un recuadro de un determinado color según su estado como se muestra en la Tabla 7.

2.3.7. Interfaz gráfica

Para la muestra de resultados de la clasificación se optó por mostrarlos mediante una interfaz que sea amigable con el usuario desarrollada en Python Figura 42, con el objetivo que pueda ser accesible mediante cualquier computador que se tenga comunicación con el dron, se busca que su uso sea sumamente sencillo para los operarios encargados del mantenimiento de las líneas de transmisión.

Se adaptó la red neuronal para que pueda detectar aisladores no solo en tiempo real, sino también en videos pregrabados e imágenes de manera local.



Figura 42. Interfaz de inicio.

En la Figura 42 se observa la interfaz de inicio hacia el usuario, en la cual como primer paso se podrá elegir la forma de detección de los aisladores ya sea en tiempo real, imágenes o videos pregrabados, el código se muestra a mayor detalle en el Anexo C. El proceso para la visualización de resultado se muestra en la Figura 43, al presionar cualquiera de los 3 botones aparecerá una nueva interfaz.

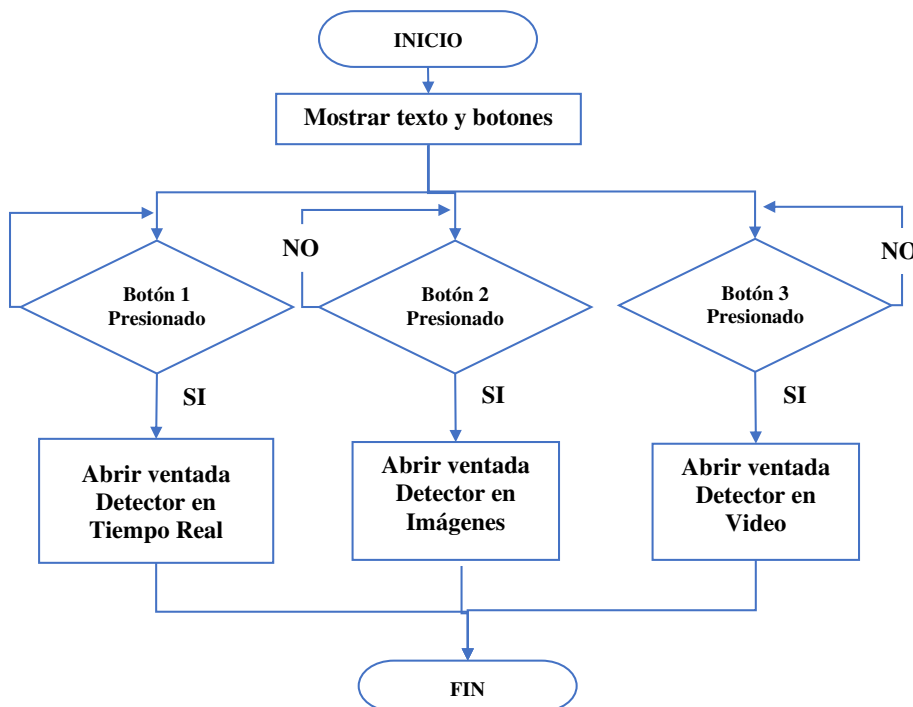


Figura 43. Proceso la visualización de resultados.

2.3.7.1. Detector en tiempo real

En la Figura 44 se muestra la interfaz utilizada para el ingreso y muestra de resultados del detector en tiempo real, cuenta con dos botones el primero “Detectar”, el cual nos permite iniciar el proceso de identificación y muestra de resultados, mientras que el botón “regresar” nos permite volver al menú principal para tener la opción de elegir una forma de detección diferente.

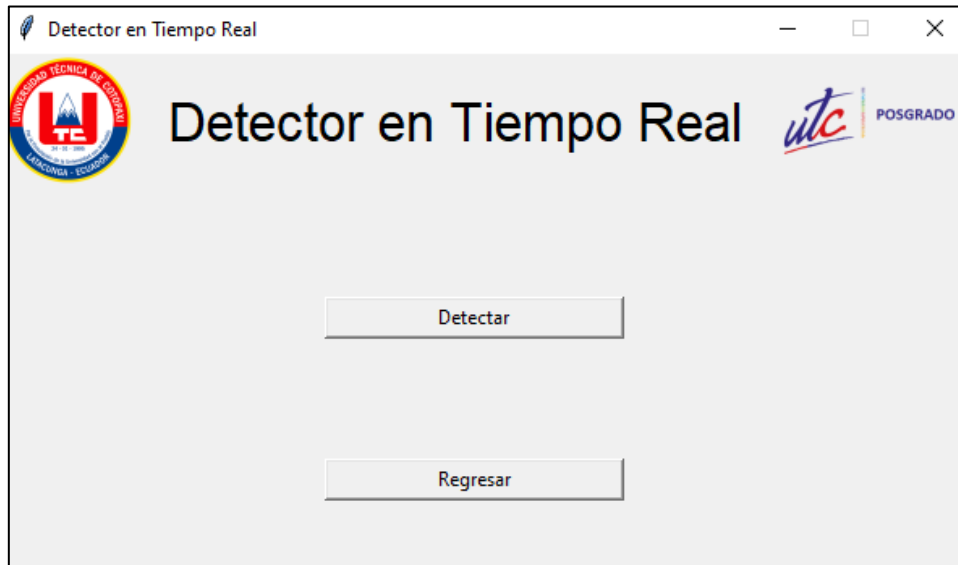


Figura 44. Interfaz detector en tiempo real.

El diagrama de flujo utilizado para el botón detectar se muestra en la Figura 45, el cual nos permite ingresar los datos obtenidos en tiempo real por el dron, ingresarlos a la red neuronal, analizarlos, clasificarlos, guardarlos y finalmente mostrar el resultado. La etapa de clasificación se puede observar de una forma más detallada en la Figura 41, el código se muestra a mayor detalle en el Anexo D.

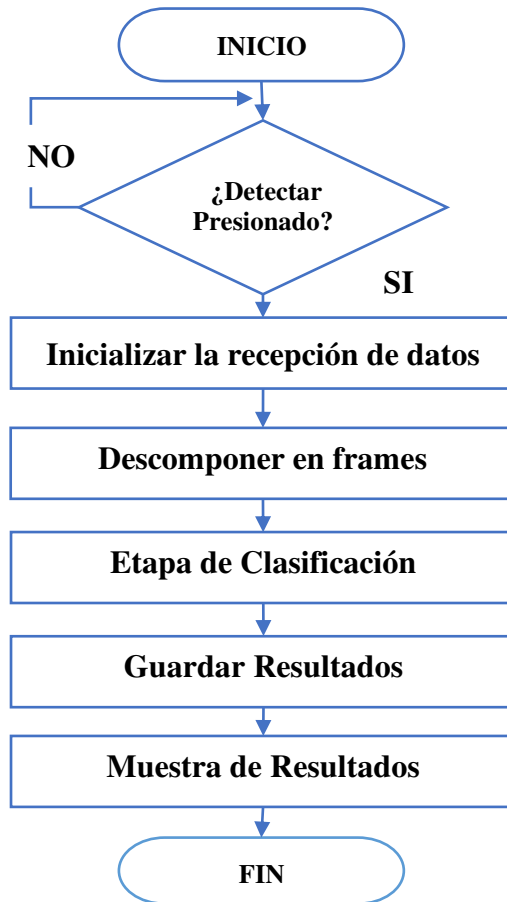


Figura 45. Rutina botón Detectar.

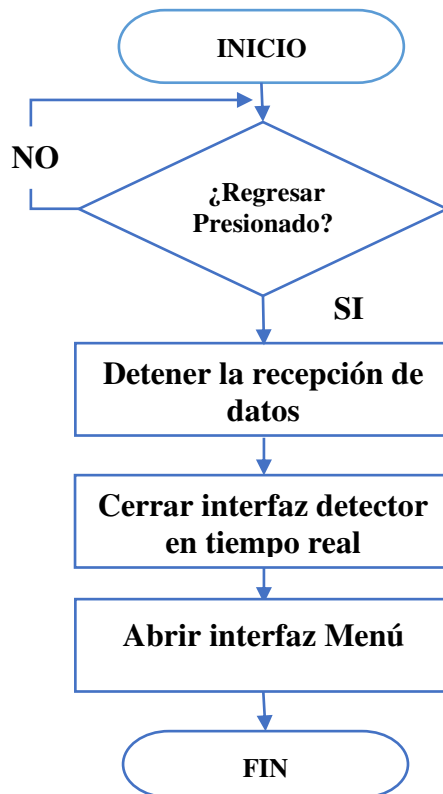


Figura 46. Rutina botón Regresar.

En la Figura 46, se muestra la rutina del botón Regresar el cual detiene la recepción de datos del dron, cierra la interfaz Detector en tiempo real y abre la interfaz Menú.

2.3.7.2. Detector en imágenes

En la Figura 47, se muestra la interfaz utilizada para el ingreso y muestra de resultados del detector de imágenes que fueron adquiridas con anterioridad en campo, a diferencia que el detector en tiempo real nos mostrara la imagen antes y después de analizarla con la red neuronal.

La interfaz cuenta con 3 botones el primero “Elegir imagen”, el cual nos permite buscar la imagen deseada, mientras que el botón “Detectar” analizar la imagen previamente seleccionada, finalmente el botón “Regresar” nos permite volver al menú principal para tener la opción de elegir una forma de detección diferente.



Figura 47. Interfaz detector de imágenes.

El diagrama de flujo utilizado para el botón “Seleccionar imagen” se muestra en la Figura 48, el cual nos permite elegir la imagen deseada y mostrarla en la interfaz, el código se muestra a mayor detalle en el Anexo E.

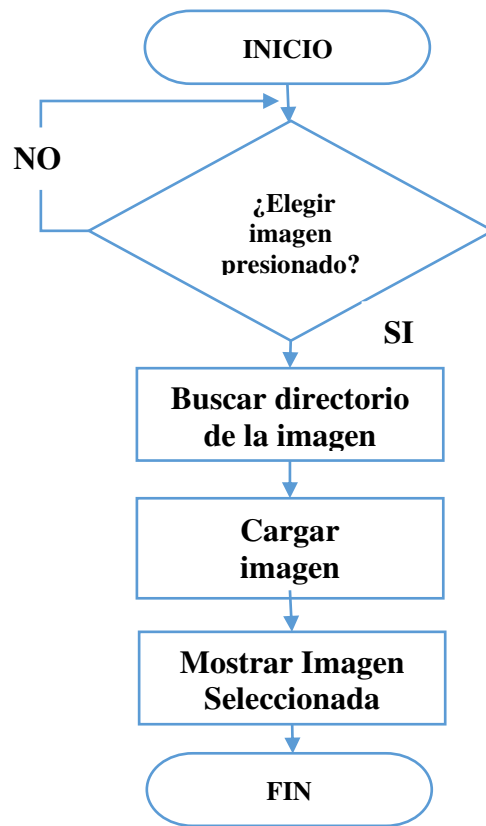


Figura 48. Rutina botón Elegir Imagen.

El diagrama de flujo utilizado para el botón detectar se muestra en la Figura 49, nos permite tomar la imagen seleccionada previamente, ingresarla a la red neuronal, analizarla, clasificarla y finalmente mostrar el resultado en la parte derecha de la interfaz. La etapa de clasificación se puede observar de una forma más detallada en la Figura 41.

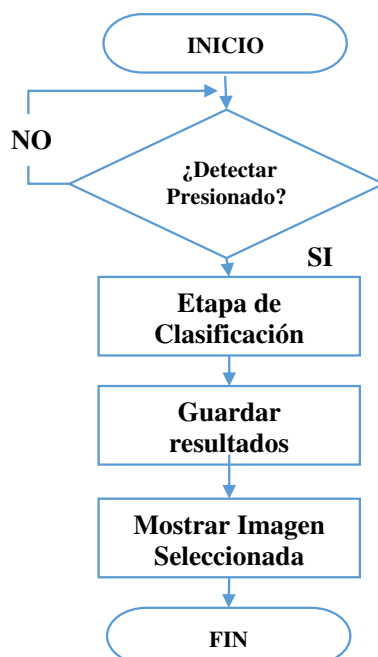


Figura 49. Rutina botón Detectar.

En la Figura 50, se muestra la rutina del botón Regresar el cual detiene el proceso de la red neuronal, cierra la interfaz Detector en imágenes y abre la interfaz Menú.

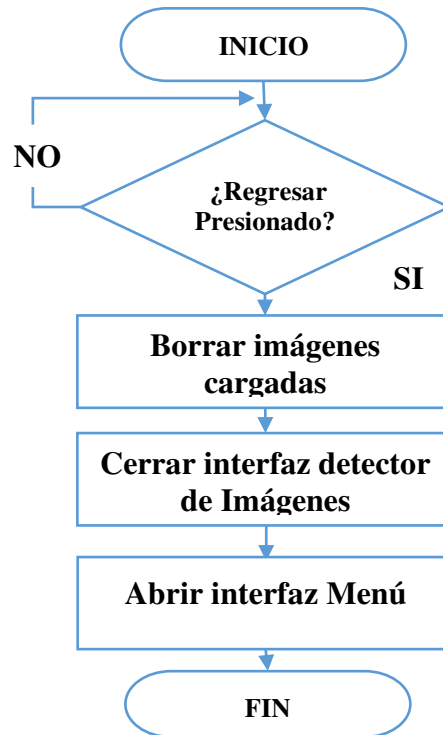


Figura 50. Rutina botón Regresar.

2.3.7.3. Detector en video

En la Figura 51, se muestra la interfaz utilizada para la selección y muestra de resultados del detector de videos pregrabados. La interfaz cuenta con 2 botones el primero “Elegir video”, el cual nos permite buscar el video deseado y analizarlo, mientras el botón “Regresar” nos permite volver al menú principal para tener la opción de elegir una forma de detección diferente.



Figura 51. Interfaz detector en video.

El diagrama de flujo utilizado para el botón “Elegir video” se muestra en la Figura 52, el cual nos permite seleccionar un video pregrabado e ingresarlo a la red neuronal, analizarlo, clasificarlo y finalmente mostrar el resultado. La etapa de clasificación se puede observar de una forma más detallada en la Figura 41, el código se muestra a mayor detalle en el Anexo F.

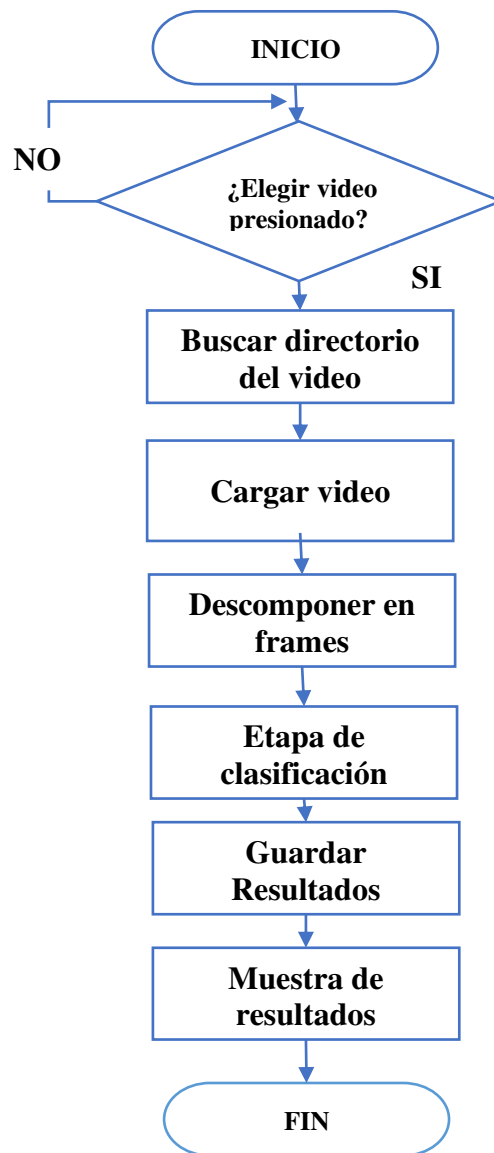


Figura 52. Rutina botón Elegir video.

En la Figura 53, se muestra la rutina del botón Regresar el cual detiene el procesamiento del video, cierra la interfaz Detector en video y abre la interfaz Menú.

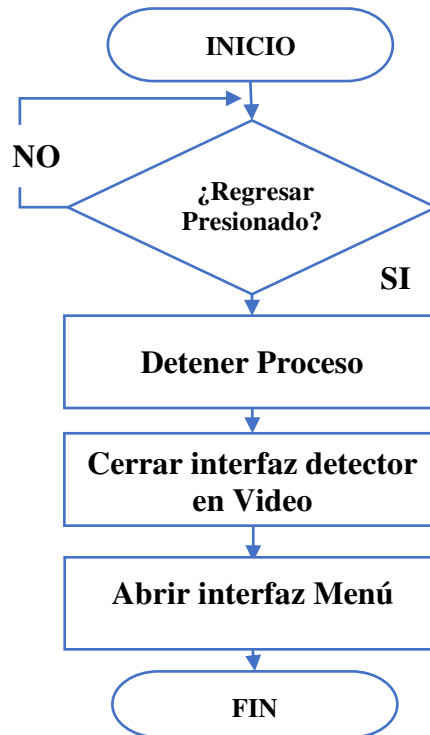


Figura 53. Rutina botón Regresar.

2.3.7.4. Etapa de guardado

Al procesar los datos de entrada en la red neuronal, también se realiza el guardado de los resultados obtenidos en carpetas específicas para videos e imágenes, cada archivo guardado es etiquetado con la fecha y la hora exacta en la que fueron procesados para que no exista ningún tipo de confusión al momento de su posterior revisión.

2.3.8. Dispositivo de procesamiento

El dispositivo de procesamiento elegido debe cumplir con varias características para que la red neuronal funcione de manera correcta y pueda ser utilizada por el personal encargado del mantenimiento de las líneas de transmisión. Se debe contar con una tarjeta gráfica de Nvidia, puesto que, este componente nos ayudara a procesar de mejor manera los datos de entrada y obtener una tasa aceptable de FPS (Frames por Segundo). Por tal motivo se eligió una laptop de la marca Asus modelo F15 FX506LI.

Tabla 11. Características del computador elegido. [48]

Características	
Procesador	Procesador Intel® Core™ i7-10870H; 2,2 GHz (caché de 16M; hasta 2,2 GHz, 8 núcleos)
Gráficos	NVIDIA® GeForce® GTX 1650 Ti, GDDR6 de 4 GB
Memoria	16 GB DDR4 (2933MHz)
Comunicación	Wi-Fi 6(802.11ax) +Bluetooth 5.2 (Dual band)

2.4. Conclusiones Capítulo II

- Es necesario poseer un dron que posea una resistencia al viento superior a 7 m/s, dado que las pruebas se realizarán en la provincia de Cotopaxi sector sierra centro del Ecuador.
- Es fundamental realizar un preprocesamiento a las imágenes antes de ingresarlos al algoritmo para no contaminar la base de datos con objetos ajenos a los aisladores.
- Se realizó una interfaz gráfica amigable con el usuario para que sea utilizada por un solo operario con o sin experiencia en programación.

CAPÍTULO III

3. VALIDACIÓN Y/O APLICACIÓN DE LA PROPUESTA

En el presente capítulo se presentarán los resultados de las pruebas realizadas en campo.

3.1. Análisis de los resultados.

Para la evaluación de la interfaz diseñada para el funcionamiento del algoritmo, se procedió a utilizar un conjunto de imágenes, de igual manera videos tomados en campo y finalmente videos en tiempo real, la forma de utilizar de manera correcta, por ende, se obtenga los mejores resultados se puede observar en el manual de usuario. (ver Anexo G)

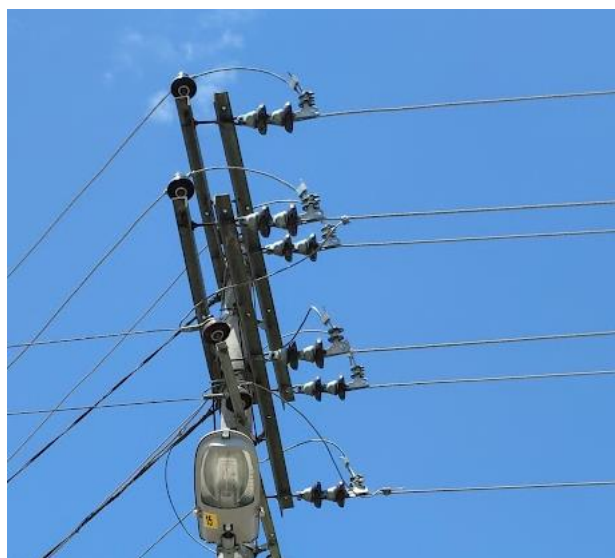


Figura 54. Aisladores de prueba.

Como se observa en la Figura 54, los aisladores utilizados para las pruebas son los que se encuentran instalados en los diferentes postes de la línea de 13.8 kV pertenecientes a ELEPCO S.A.

3.1.1. Validación del algoritmo

Antes de realizar las pruebas de campo se procedió a realizar la validación de resultados con el porcentaje de imágenes de la base de datos destinado a este fin, el total de imágenes que fueron validadas son 1000 que corresponde al 30% de la base de datos.

Dichos resultados se muestran en la Tabla 12, posteriormente se aplica la matriz de confusión, la cual es una herramienta utilizada en el aprendizaje automático para determinar qué tan fiable es el algoritmo, es decir, se obtiene todos los valores posibles y obtener la exactitud del algoritmo para reconocer aisladores de porcelana.

Tabla 12. Matriz de confusión validación de la base de datos.

Estado	Predicción			Total
	Bueno	Roto/ Flameado	Roto	
Bueno	325	15	4	344
Roto/Flameado	11	315	4	330
Roto	4	4	318	326
Total	340	334	326	1000

Aplicando la fórmula de la exactitud se obtiene:

$$Accuracy = \frac{Vp1 + Vp2 + Vp3}{TotaldeMuestras}$$

Donde:

$Vp1, Vp2, Vp3 = Verdaderospositivosdecadamuestra.$

$$Accuracy = \frac{325 + 315 + 318}{1000} * 100\%$$

$$Accuracy = 0.958 = 95.8 \%$$

$$Error = (1 - 0.958) = 0.042 * 100 \% = 4.2 \%$$

De igual manera se calcula la sensibilidad para cada una de las clases:

$$Sensibilidad = \frac{Vp}{Vp + Fn} * 100 \%$$

Donde:

$F_n = Falsos Negativos$ $V_p = Verdaderos Positivos /$

Tabla 13. Sensibilidad validación de datos.

Buenos	Rotos	Sucios/Flameados
$Sen. = \frac{325}{325 + 15} * 100 \%$	$Sen. = \frac{315}{315 + 19} * 100 \%$	$Sen. = \frac{318}{318 + 8} * 100 \%$
Sen. = 0.956 = 95.6 %	Sen. = 0.943 = 94.3 %	Sen. = 0.975 = 97.5 %

Al realizar la validación de la red neuronal no arroja una exactitud 95.8 %, lo que indica que el entrenamiento se realizó con éxito dado que el error de 4.2 % es mínimo, dicho error puede darse por una mala clasificación en la etapa de creación de la base de datos.

De igual manera se observa que la sensibilidad en la Tabla 13, con lo que se puede concluir que la red neuronal en la etapa de validación de resultados es exacta y sensible para cada una de las clases.

3.1.2. Detector en tiempo real

Para la prueba del algoritmo de reconocimiento de aisladores en tiempo real se procedió a realizar la prueba en campo, mediante una conexión wifi el dron se envía el video en tiempo real hacia la interfaz para clasificarlos.

Para la evaluación de los resultados se descompuso el video recibido por el dron y que fue analizado por el algoritmo en frames (imágenes individuales), para analizarlos y obtener el estado de los aisladores.

Para la verificación del correcto funcionamiento de Yolo como algoritmo se procedió a realizar las pruebas de campo en varios postes de la línea de 13.8 kV, comprendida desde la Subestación Salcedo (69/13.8 kV) hasta el centro de la ciudad sector el coliseo (ver Figura 55), en la que se encontraron un total de 120 aisladores de diferentes estados (sucio/flameado, roto, bueno).

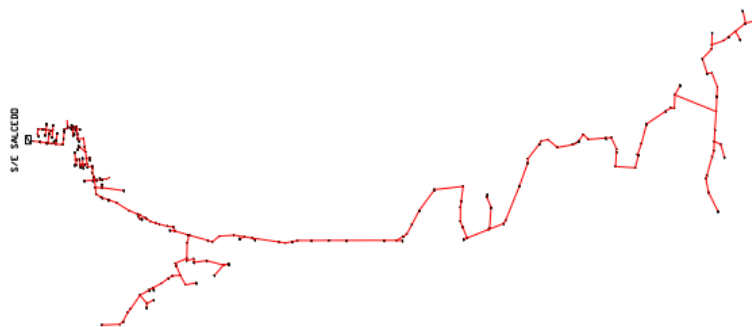


Figura 55. Línea de 13.8 kV S/E Salcedo-el coliseo.

Dichos resultados se muestran en la Tabla 14, para posteriormente aplicar la matriz de confusión.

Tabla 14. Matriz de confusión detector en tiempo real.

Estado	Predicción			Total
	Bueno	Roto/ Flameado	Roto	
Bueno	53	2	2	57
Roto/Flameado	5	42	2	49
Roto	2	2	10	14
Total	60	46	14	120

Aplicando la fórmula de la exactitud se obtiene:

$$Accuracy = \frac{Vp1 + Vp2 + Vp3}{TotaldeMuestras}$$

Donde:

$Vp1, Vp2, Vp3 = Verdaderospositivosdecadamuestra.$

$$Accuracy = \frac{53 + 42 + 10}{120} * 100\%$$

$$Accuracy = 0.875 = 87.5\%$$

$$Error = 1 - 0.875 = 0.125 = 12.5 \%$$

De igual manera se calcula la sensibilidad para cada una de las clases:

$$Sensibilidad = \frac{Vp}{Vp + Fn}$$

Donde:

$Fn = Falsos Negativos$

$Vp = verdaderos Positivos$

Tabla 15. Sensibilidad en tiempo real.

Buenos	Rotos	Sucios/Flameados
$Sen. = \frac{53}{53 + 7} * 100 \%$	$Sen. = \frac{42}{42 + 4} * 100 \%$	$Sen. = \frac{10}{10 + 4} * 100 \%$
Sen. = 0.883 = 88.33 %	Sen. = 0.913 = 91.3 %	Sen. = 0.714 = 71.4 %

La exactitud del algoritmo detector de aisladores es de 87.5%, lo que lo hace sumamente exacta para la detección de aisladores en tiempo real, el restante 12.5 % de error puede deberse a que al momento de posicionar el dron se produjeron algún tipo de sombras o destello que el algoritmo los tomo como falsos positivos y/o falsos negativos.

De igual manera se observa en la Tabla 15 que la sensibilidad es aceptable, excepto al detectar aisladores rotos, dado que, detectar un aislador roto es más complicado para el algoritmo cuando dichas roturas no son tan visibles, con lo que se puede concluir que el algoritmo es exacto y sensible para la detección en tiempo real a excepción de aisladores rotos.

En la Figura 56 se muestra a los fotogramas tomados del video en tiempo real, en el cual se muestran los resultados aplicando el algoritmo de detección de estado de los aisladores.



Figura 56. Fotogramas extraídos detección en tiempo real.

3.1.3. Detector de imágenes

Para la prueba del algoritmo de reconocimiento de aisladores en imágenes se procedió a realizar la prueba con imágenes tomadas en campo con la ayuda del dron.

Para la verificación del correcto funcionamiento del algoritmo se procedió a obtener 100 imágenes de aisladores de varios postes de la línea de 13.8 kV en los que se encontraron un total de 150 aisladores de diferentes estados (sucio, roto, bueno).

Tabla 16. Matriz de confusión detector en imágenes.

Estado	Predicción			Total
	Bueno	Roto/ Flameado	Roto	
Buenos	61	2	0	63
Sucio	2	45	1	48
Roto/Flameado	2	2	35	39
Total	64	48	36	150

Aplicando la fórmula de la matriz se obtiene:

$$Accuracy = \frac{Vp1 + Vp2 + Vp3}{TotaldeMuestras}$$

Donde:

$Vp1, Vp2, Vp3 = Verdaderospositivosdecadamuestra.$

$$Accuracy = \frac{61 + 46 + 35}{150} * 100\%$$

$$Accuracy = 0.9467 = 94.67\%$$

$$Error = 1 - 0.9467 = 0.053 = 5.33 \%$$

De igual manera se calcula la sensibilidad para cada una de las clases:

$$Sensibilidad = \frac{Vp}{Vp + Fn} * 100 \%$$

Donde:

$Fn = Falsos Negativos$

$Vp = verdaderos Positivos$

Tabla 17. Sensibilidad en imágenes.

Buenos	Rotos	Sucios/Flameados
$Sen. = \frac{61}{61 + 4} * 100 \%$	$Sen. = \frac{45}{45 + 4} * 100 \%$	$Sen. = \frac{35}{35 + 1} * 100 \%$
Sen. = 0.938 = 93.8 %	Sen. = 0.918 = 91.8 %	Sen. = 0.972 = 97.2 %

La exactitud del algoritmo detector de aisladores es de 94.67%, lo que lo hace más que aceptable para la detección de aisladores en imágenes, el restante 5.33 % de error puede deberse a que al momento de obtener las imágenes se encontraron algún tipo de sombras o características similares con lo que se obtuvieron falsos positivos y/o falsos negativos.

De igual manera se observa que la sensibilidad en la Tabla 17 es aceptable ya que supera en los tres casos el 90%, con lo que se puede concluir que el algoritmo es exacto y sensible para la detección en imágenes.

En la Figura 57, Figura 58 y Figura 59 se muestran los resultados al aplicar el algoritmo de detección a diferentes aisladores obteniendo los resultados deseados.



Figura 57. Detección aislador sucio/flameado.



Figura 58. Detección aislador bueno.



Figura 59. Detección aislador roto.

Como se observa en la Figura 59, el algoritmo es capaz de detectar dos estados de los aisladores en este caso bueno y roto, lo que indicaría que solo sería necesario cambiar un solo aislador y no el par completo.

3.1.4. Detector en video

Para la prueba del algoritmo de reconocimiento de aisladores en videos se procedió a realizar la prueba con videos previamente grabados con ayuda del dron.

Para la verificación del correcto funcionamiento del algoritmo se procedió a obtener varios 40 videos de varios postes de la línea de 13.8 kV en los que se encontraron un total de 160 aisladores de diferentes estados (sucio, roto, bueno).

Tabla 18. Matriz de confusión detector en video.

Estado	Predicción			Total
	Bueno	Roto/ Flameado	Roto	
Bueno	83	5	2	90
Roto/ Flameado	2	50	3	55
Roto	2	3	10	15
Total	87	58	15	160

Aplicando la fórmula de la matriz se obtiene:

$$Accuracy = \frac{Vp1 + Vp2 + Vp3}{TotaldeMuestras} * 100 \%$$

Donde:

$Vp1, Vp2, Vp3 = Verdaderospositivosdecadamuestra.$

$$Accuracy = \frac{83 + 50 + 10}{160} * 100\%$$

$$Accuracy = 0.8938 = 89.38\%$$

$$Error = 1 - 0.8938 = 0.106 = 10.62\%$$

De igual manera se calcula la sensibilidad para cada una de las clases:

$$Sensibilidad = \frac{Vp}{Vp + Fn} * 100\%$$

Donde:

$Fn =$ Falsos Negativos

$Vp =$ verdaderos Positivos

Tabla 19. Sensibilidad en video.

Buenos	Rotos	Sucios/Flameados
$Sen. = \frac{83}{83 + 4} * 100\%$	$Sen. = \frac{50}{50 + 8} * 100\%$	$Sen. = \frac{10}{10 + 5} * 100\%$
$Sen. = 0.954 = 95.4\%$	$Sen. = 0.862 = 86.2\%$	$Sen. = 0.666 = 66.6\%$

La exactitud del algoritmo detector de aisladores es de 89.38 %, lo que lo hace aceptable para la detección de aisladores en videos, el restante 10.62 % de error puede deberse a que al momento de obtener los videos se encontraron algún tipo de sombras o destellos de luz con lo que se obtuvieron falsos positivos y/o falsos negativos.

De igual manera se observa que la sensibilidad en la Tabla 19, excepto al detectar aisladores rotos, dado que, detectar un aislador roto es más complicado para el algoritmo cuando dichas roturas no son tan visibles, con lo que se puede concluir que el algoritmo es exacto y sensible para la detección en videos a excepción de aisladores rotos.

En las Figura 60 se muestran algunos fotogramas de los videos de los resultados al aplicar el algoritmo de detección a diferentes videos de aisladores obteniendo los resultados deseados.

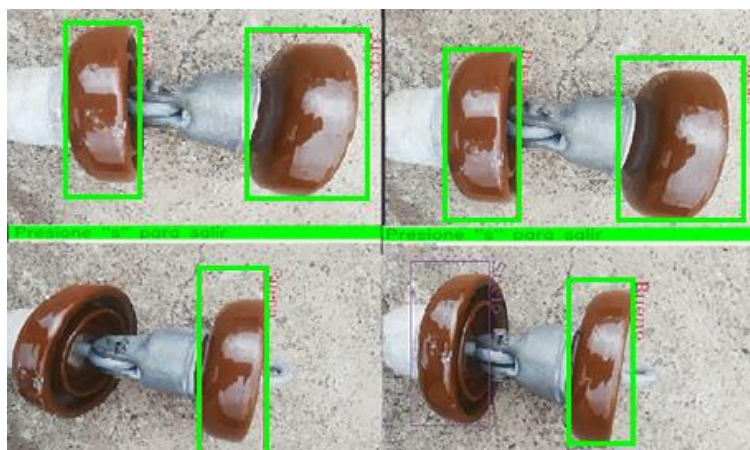


Figura 60. Fotogramas del algoritmo de detección en videos.

3.2. Validación técnica.

Al realizar un análisis de varios estudios sobre redes neuronales, se pudo constatar que no existe un valor específico para que la exactitud sea aceptable, en dichos estudios se pudo constatar que el valor mínimo obtenido fue de 80% [49] [17] [18], mientras que el mayor valor obtenido fue de 93% [50] [51] [52], por ende, se concluye que un valor de exactitud aceptable debe estar dentro del rango del 80. al 93%.

Al finalizar las pruebas de campo del algoritmo de identificación de fallas en los aisladores, los resultados obtenidos han sido más que aceptables debido a que la exactitud se encuentra sobre el 87.5 % y un error de 12.5 % en tiempo real.

Este porcentaje de error podría deberse al hardware utilizado además del manejo del dron, puesto que, al momento de realizar las pruebas de campo el dron se vio afectado por el viento, lo que provoco dificultad la toma de video e imágenes, de igual manera el ángulo de toma de videos no fue el correcto debido a la luz solar sobre los aisladores.

Otro aspecto a resaltar al momento de analizar el video enviado por el dron hacia el computador, el costo computacional es sumamente alto y podría presentarse algún tipo de perdida de datos al analizarlos. Además dicho error viene dado por el tamaño de la muestra de 3000 imágenes de diferentes fallas de aisladores, puesto que, en el caso de aisladores sucios/flameados la suciedad puede presentarse de varias maneras, además de los aisladores rotos las roturas siempre se van a presentar de diferente forma [33].

Se observa que es factible su utilización dentro de los programas de mantenimiento de las líneas de transmisión, con lo cual se puede reducir los tiempos de ejecución. Tomando en consideración, que para su uso solo sería necesario poseer un dron y un computador cargado con el algoritmo, además su uso es sumamente sencillo con lo que cualquier operador podría utilizarlo.

Seria optimo poder ampliar al algoritmo para poder identificar otros problemas que se presentan en las líneas de transmisión, como en análisis de los conductores en lugares donde el acceso de un vehículo grúa dificulta la inspección y es necesario que los operarios escalen las estructuras para este propósito, debido a que la visión artificial es una rama que se puede adaptar a diferentes usos y escenarios.

3.3. Validación económica.

3.3.1. Análisis económico

En la Tabla 20 se muestra el análisis económico, detallando cada elemento utilizado, la cantidad y sus respectivos precios de implementación.

Tabla 20. Análisis económico.

	Detalle	Cantidad	V. Unitario	V. Total
1	Dron Dji Mini 2	1	450.00	450.00
2	Computador	1	1200.00	1200.00
3	Mano de obra	100 horas	4.55	455.00
4	Imprevistos		100.00	100.00
		Total		2205.00

Se observa que el costo total fue de 2205.00 dólares americanos, lo cual se dividen en 450 dólares en un cuadricóptero que cumpla con todas las características para la ejecución del proyecto, 1200.00 en un computador personal que cumpla con unas características específicas con la finalidad que sea capaz de correr el algoritmo en tiempo real, mientras que 455 distribuido en 100 horas de mano de obra con un costo por unitario de 4.55 según el ministerio de trabajo [53] y 100 dólares de imprevistos.

3.3.2. Análisis costo-beneficio

Al realizar el análisis de los aisladores de una red de 13.8 Kv, la empresa eléctrica provincial Cotopaxi S.A. incurre en varios gastos:

Tabla 21. Costos actuales.

Material	Costo x Hora	Costo Total
Grúa Canasta Aislada	25.00	25.00
Mano de Obra	4.16	4.16
	Total	29.16

Al analizar la Tabla 21, se observa que la empresa incurre en gastos por hora tanto en el equipo como la grúa canasta aislada, además del personal de mantenimiento, dichos gastos se toman en consideración por hora.

Si se toma en consideración que se realiza la inspección de una línea se realiza cada 6 meses, y para llevar a cabo la inspección de un tramo de 12 Km, toma aproximadamente 8 horas. Se tiene un costo total de:

$$\text{Costo Anual} = \text{Costo Hora} * \text{Numero de Horas} * \text{Dias al Año}$$

$$\text{Costo Anual} = 29.16 * 8 * 2$$

$$\text{Costo Anual} = 466.56$$

Al realizar el cálculo del costo total de la inversión necesaria para realizar el análisis de aisladores de un tramo de 12 km de una línea de transmisión, es necesario incurrir en un gasto de 466.56 anuales.

Tabla 22. Flujo de caja.

	Inversión	\$2.205,00		
	Tasa de descuento	10%		
FLUJO DE CAJA				
Periodo	Inversión	Ingresos	Egresos	Flujo de Efectivo Neto
0	\$2.205,00	0	0	-\$2.205,00
1		\$466,56	\$20,00	\$446,56
2		\$466,56	\$20,00	\$446,56
3		\$466,56	\$20,00	\$446,56
4		\$466,56	\$20,00	\$446,56
5		\$466,56	\$80,00	\$386,56
6		\$466,56	\$20,00	\$446,56
7		\$466,56	\$20,00	\$446,56
8		\$466,56	\$20,00	\$446,56
9		\$466,56	\$20,00	\$446,56

Tabla 23. Referencia costo-beneficio.

Costo Beneficio	
VNA Ingresos	\$2.686,93
VNA Egresos	\$138,58
VNA Egresos +Inver- sión	\$2.343,58
Costo Beneficio	1,146507689
VAN	\$329,49
TIR	14%

Como se observa la

Tabla 23, el análisis costo beneficio es superior a 1 lo que indica que el proyecto es viable para su implementación, se debe tomar en consideración que para el análisis de la Tabla 22, se tomó en consideración solo un tramo de 12 Km de líneas de transmisión, si se toma en consideración que el proyecto puede ser implementado en varios sectores de la provincia de Cotopaxi, dado que cuenta con cerca de 4000 Km de líneas de transmisión [8] (Ver Anexo H), y el costo beneficio subiría exponencialmente.

Además, se puede observar el valor del VAN de 329.49, lo que representa un ahorro para la empresa, finalmente al analizar el valor del TIR se observa que se tiene una tasa de retorno de la inversión de un 14%.

3.4. Conclusiones del Capítulo III

- Diversas investigaciones sitúan a la exactitud de una red neuronal en un rango de 80% al 95 %, considerando que las pruebas de campo se obtuvieron un 87 % el algoritmo es aceptable.
- Es necesario poseer un computador con una tarjeta gráfica externa de 4 GB para obtener una tasa superior a 25 FPS de manera que los operarios puedan observar los resultados de manera fluida.
- Al momento de utilizar el dron es necesario ubicarlo en un ángulo el cual no se refleje la luz solar para no perder rasgos importantes de los aisladores y aumentar la exactitud de la red neuronal.

CONCLUSIONES

- Se implementó Yolo como algoritmo para la detección de tres estados de los aisladores de una línea de transmisión mediante visión artificial dichos estados son bueno, roto y sucio/flameado, con lo cual, se reduce el tiempo de toma de decisiones de los operarios de mantenimiento, mismo que está compuesto por las etapas: adquisición, procesamiento y recepción mediante comunicación WIFI.
- El algoritmo implementado fue capaz de identificar el estado de aisladores de una línea de transmisión de 13.8 kV se tomó como muestra 120 aisladores del tramo subestación Salcedo hasta sector el coliseo, obteniendo como resultados: aisladores buenos en un 44.2 %, aisladores rotos 35% y aisladores rotos 15 %, lo que nos indica que si no se da un mantenimiento de los aisladores rotos y sucios y/o flameados existe una gran probabilidad que se presente una falla.
- Al comparar la exactitud de Yolo como algoritmo para la detección de fallas en los aisladores aplicado a imágenes, videos y tiempo real, se logró determinar que la exactitud es aceptable debido a que se obtuvo 87% ,89% y 94% respectivamente, tomando en consideración que dichos resultados concuerdan con resultados obtenidos por investigaciones de similares características.
- Al recolectar las imágenes para la base de datos del algoritmo de identificación, se pudo observar que existen más aisladores sucios/flameados que rotos o fisurados, lo que dificultó el entrenamiento del algoritmo Yolo, ya que fue necesario adquirir dichas imágenes en varias líneas de transmisión de las ciudades de Latacunga y Salcedo.
- Al realizar las pruebas de campo se pudo observar, que la suciedad y hollín en los aisladores es mucho más sencilla de procesar para el algoritmo debido a que cubren un gran porcentaje del área del aislador, mientras que las roturas y/o fisuras son más complejas debido a que algunas son pequeñas comparadas al área total del elemento.

- Es necesario realizar un preprocesamiento de las imágenes antes de analizarlas con el algoritmo de identificación de fallas en los aisladores, puesto que la localización de los aisladores en campo presenta un escenario complejo, es decir se necesitaría más tiempo y recursos computacionales para predecir las fallas y en otros casos se podría dar un resultado errado sin este paso.
- Al utilizar software libre para el entrenamiento y ejecución del algoritmo de identificación de fallas en los aisladores utilizando visión artificial, se eliminan los costos de uso para la empresa encargada del mantenimiento de la línea de transmisión, puesto que no es necesario ningún tipo de licencia comercial a diferencia de otros softwares en el mercado.
- Para que el algoritmo trabaje de forma óptima en tiempo real en la identificación de las fallas de los aisladores, es necesario correrlo en un computador que posea una alta capacidad de procesamiento para obtener el resultado a una tasa aceptable de FPS (Fotogramas por Segundo).

RECOMENDACIONES

- Al recolectar las imágenes en campo es necesario tomar en consideración la luz natural que se presenta, debido a que si existe demasiada podría ocasionar la pérdida de atributos de los aisladores, lo que provocaría un resultado errado al momento de correr el algoritmo.
- Al trabajar con un dron cerca de las líneas de transmisión, es recomendable tener una distancia de seguridad, debido a que los campos electromagnéticos podrían ocasionar interferencia, perder el control y producir algún tipo de siniestro del equipo.
- Para seguir obteniendo a futuro una tasa de exactitud aceptable, se recomienda reentrenar el algoritmo con nuevas imágenes, ya que al utilizar software libre es factible seguir aumentando la base de datos de forma rápida y sencilla.
- Se recomienda realizar un entrenamiento previo sobre drones para que no exista ningún tipo de inconveniente al momento de utilizarlo, adicional conocer la normativa sobre su uso que actualmente rige en el país.
- Se recomienda tomar esta investigación como base para el análisis de la parte interna de los aisladores, tomando en consideración que hay una alta tasa de fallos que se producen en esta parte específica de dichos elementos.

REFERENCIAS BIBLIOGRÁFICAS

- [1] U. Nations, «Día Mundial de la Población | Naciones Unidas», *United Nations*, 2019. <https://www.un.org/es/observances/world-population-day> (accedido 29 de agosto de 2020).
- [2] Operador Nacional de Electricidad, CENACE, «Informe Anual Cenace 2020», Quito, 2020, 2020. Accedido: 29 de mayo de 2021. [En línea]. Disponible en: <http://www.cenace.gob.ec/informe-anual-2020/>
- [3] F. J. E. González, *Montaje de redes eléctricas aéreas de alta tensión. ELEE0209*. IC Editorial, 2013.
- [4] D. Ghosh y D. Khastgir, «Degradation and stability of polymeric high-voltage insulators and prediction of their service life through environmental and accelerated aging processes», *ACS omega*, vol. 3, n.º 9, pp. 11317-11330, 2018.
- [5] J. Estarita, A. Jim, J. Brochero, H. Escobar, y S. Moreno, «Sistema de Reconocimiento de objetos en tiempo real», *Investigación y desarrollo en TIC*, vol. 8, n.º 2, Art. n.º 2, 2017.
- [6] L. M. Vargas, L. F. Muñoz Sanabria, y F. J. Álvarez, «ALEDO Algoritmo para la detección de objetos», *Tecnología Educativa Revista CONAIC*, vol. 6, n.º 1, pp. 68-73, may 2019, doi: 10.32671/terc.v6i1.82.
- [7] Ministerio de Electricidad y Energía Renovable, «Plan Maestro de Electricidad 2018-2027», Quito, 2018. Accedido: 28 de marzo de 2021. [En línea]. Disponible en: <https://www.cnelep.gob.ec/wp-content/uploads/2020/01/Plan-Maestro-de-Electricidad-2018-2027.pdf>
- [8] Agencia de Regulación y Control de Electricidad - ARCONEL, «Atlas Del Sector Eléctrico 2019», Quito, 2020. [En línea]. Disponible en: <https://www.regulacionelectrica.gob.ec/estadistica-del-sector-electrico/>
- [9] H. E. Rojas y H. D. Rojas, «Evaluación del Desempeño de Aisladores Eléctricos de Distribución Cerámicos y Poliméricos bajo Biocontaminación por Hongos», *Información tecnológica*, vol. 28, n.º 2, pp. 03-10, 2017, doi: 10.4067/S0718-07642017000200002.
- [10] E. O. Mancheno Moreira y Q. Gaibor, «Análisis: hidroeléctricas y calidad de vida en Ecuador», *Observatorio de la Economía Latinoamericana*, n.º noviembre, 2019, [En línea]. Disponible en: <https://www.eumed.net/rev/oel/2019/11/hidroelectricas-calidad-vida.html>
- [11] J. Ramírez *et al.*, «Cálculo de la distribución de voltajes en las cadenas de aisladores de una línea de transmisión de 500 kV utilizando el método de los elementos finitos», *Enfoque UTE*, vol. 11, n.º 3, pp. 1-14, sep. 2020, doi: 10.29019/enfoque.v11n3.619.
- [12] W. C. C. Jurado, M. R. Gámez, O. S. Suárez, y A. V. Pérez, «Impacto ambiental en las redes eléctricas próximas al litoral manabita», *Revista Cubana de Ingeniería*, vol. 9, n.º 3, pp. 69-77, 2019.
- [13] J. Estarita, A. Jim, J. Brochero, H. Escobar, y S. Moreno, «Sistema de Reconocimiento de objetos en tiempo real», *Investigación y desarrollo en TIC*, vol. 8, n.º 2, Art. n.º 2, 2017.
- [14] J. R. Ruiz Sarmiento, F. Á. Moreno Dueñas, J. González Monroy, y A. J. Gonzalez-Jimenez, «Tutorial para el reconocimiento de objetos basado en características empleando herramientas Phytion», sep. 2018, Accedido: 23 de agosto de 2020. [En línea]. Disponible en: <https://riuma.uma.es/xmlui/handle/10630/16501>

- [15] C. Quintero, F. Merchán, A. Cornejo, y J. S. Galán, «Uso de Redes Neuronales Convolucionales para el Reconocimiento Automático de Imágenes de Macroinvertebrados para el Biomonitorio Participativo», *1*, pp. 585-596, feb. 2018, doi: 10.18502/keg.v3i1.1462.
- [16] S. Y. Wong, C. W. C. Choe, H. H. Goh, Y. W. Low, D. Y. S. Cheah, y C. Pang, «Power transmission line fault detection and diagnosis based on artificial intelligence approach and its development in uav: A review», *Arabian Journal for Science and Engineering*, vol. 46, n.º 10, pp. 9305-9331, 2021.
- [17] P. G. Valencia, «Detección de fallas en los aisladores de la línea de transmisión eléctrica bloque MDC-PBHI aplicando visión artificial.», Escuela Politécnica Nacional, Quito, 2019. Accedido: 23 de agosto de 2020. [En línea]. Disponible en: <http://bibdigital.epn.edu.ec/handle/15000/20154>
- [18] S. L. Beleño Díaz, C. V. Martínez Luna, I. F. Mondragón Bernal, y C. A. Parra Rodríguez, «Inspección De Aisladores En Líneas De Transmisión Eléctrica Usando Inteligencia Artificial», *Revista Colombiana De Tecnologías De Avanzada (RCTA)*, vol. 2, n.º 36, Art. n.º 36, jul. 2020, doi: 10.24054/16927257.v36.n36.2020.4018.
- [19] M. Alcázar Ortega, C. S. Cañas Peñuelas, G. Escrivá Escrivá, V. L. Fuster Roig, y J. Roger Folch, *Generación, transporte y distribución de energía eléctrica*. Editorial Universitat Politècnica de València, 2019. Accedido: 28 de marzo de 2021. [En línea]. Disponible en: <https://riunet.upv.es/handle/10251/118519>
- [20] Ministerio de Energía y Recursos Naturales no Renovables, «Sistema de Transmisión 500 kV», 2020. <https://www.recursoyenergia.gob.ec/sistema-de-transmision-500-kv/> (accedido 10 de junio de 2021).
- [21] W. Theodore, *Electrical machines, drives, and power systems*. 2007.
- [22] L. L. Grigsby, *Electric Power Generation, Transmission, and Distribution*. CRC Press, 2018.
- [23] M. Plieva, E. Gurieva, y E. Lysokon, «Analysis of different modes of cleaning insulators of air transmission lines in mountain conditions», en *2020 International Ural Conference on Electrical Power Engineering (UralCon)*, 2020, pp. 318-323.
- [24] D. E. M. Dorta, «Análisis de flameo en aisladores. Parte 2.».
- [25] P. Morcelle del Valle, M. F. P. Deorsola, y M. D. del Pozo, «Evaluación del efecto de la contaminación sobre aisladores poliméricos retirados de servicio», 2009.
- [26] M. Nixon y A. Aguado, *Feature Extraction and Image Processing for Computer Vision*. Academic Press, 2019.
- [27] V. Alvear, P. Rosero, D. Peluffo, y J. Pijal, «Internet de las Cosas y Visión Artificial, Funcionamiento y Aplicaciones: Revisión de Literatura», *Enfoque UTE*, vol. 8, n.º 1: Edición Especial INCISCOS, 2016.
- [28] J. P. A. Moya, *Procesamiento y Análisis de imágenes digitales*. Costa Rica, 2012.
- [29] A. F. Villán, *Mastering OpenCV 4 with Python: A practical guide covering topics from image processing, augmented reality to deep learning with OpenCV 4 and Python 3.7*. Packt Publishing Ltd, 2019.
- [30] A. F. Villán, *Mastering OpenCV 4 with Python: A practical guide covering topics from image processing, augmented reality to deep learning with OpenCV 4 and Python 3.7*. Packt Publishing Ltd, 2019.

- [31] A. Moursi, R. Samhoud, Y. Kamal, M. Magdy, S. El-Ashry, y A. Shalaby, «Different reference models for uvm environment to speed up the verification time», en *2018 19th International Workshop on Microprocessor and SOC Test and Verification (MTV)*, 2018, pp. 67-72.
- [32] M. N. Gevorkyan, A. V. Demidova, T. S. Demidova, y A. A. Sobolev, «Review and comparative analysis of machine learning libraries for machine learning», *Discrete and Continuous Models and Applied Computational Science*, vol. 27, n.º 4, pp. 305-315, 2019.
- [33] K. Gurney, *An introduction to neural networks*. CRC press, 2018.
- [34] R. F. López y J. M. F. Fernández, *Las redes neuronales artificiales*. Netbiblo, 2008.
- [35] S. Rozada Raneros, «Estudio de la arquitectura YOLO para la detección de objetos mediante deep learning», 2021.
- [36] D. Krstinić, M. Braović, L. Šerić, y D. Božić-Štulić, «Multi-Label Classifier Performance Evaluation With Confusion Matrix», *Computer Science & Information Technology*, p. 1.
- [37] A. Tharwat, «Classification assessment methods», *Applied Computing and Informatics*, 2020.
- [38] G. Rotskoff y E. Vanden-Eijnden, «Parameters as interacting particles: long time convergence and asymptotic error scaling of neural networks», *Advances in neural information processing systems*, vol. 31, 2018.
- [39] R. Hernandez Sampieri, C. Fernandez Collado, y M. del P. Baptista Lucio, *Metodología De La Investigacion*, Quinta. Mexico: McGraw-Hill Interamerica na México^ eD. F DF, 2014.
- [40] C. A. Bernal Torres, *Metodología de la investigación: para administración, economía, humanidades y ciencias sociales*, Tercera Edicion. Bogota: Worldcolor., 2010.
- [41] Á. G. P. Palacios, S. A. I. Rodríguez, E. D. V. Fuentes, V. M. C. Quinto, N. L. M. Párraga, y F. E. Z. Gavilanes, «Producción de energía eólica en Ecuador», *Ciencia Digital*, vol. 3, n.º 3, pp. 22-32, 2019.
- [42] Dji, «DJI Mini 2 User Manual v1.0». 2021. Accedido: 10 de enero de 2022. [En línea]. Disponible en: https://dl.djicdn.com/downloads/DJI_Mini_2/20210630/DJI_Mini_2_User_Manual-EN.pdf
- [43] Agencia de Regulación y Control de la Electricidad (ARCONEL), *Regulación Nro. ARCONEL 001/18*. 2018, p. 13. Accedido: 10 de mayo de 2022. [En línea]. Disponible en: <https://www.regulacionelectrica.gob.ec/wp-content/uploads/downloads/2018/07/018-18-Proyecto-de-Regulacion-Franjas-de-Servidumbre-en-lineas-del-servicio-de-energia-electrica-y-distancias-de-seguridad-entre-las-redes-electricas-y-edificaciones.pdf>
- [44] I. Balki *et al.*, «Sample-size determination methodologies for machine learning in medical imaging research: a systematic review», *Canadian Association of Radiologists Journal*, vol. 70, n.º 4, pp. 344-353, 2019.
- [45] J. Dehesa, A. Vidler, J. Padget, y C. Lutteroth, «Grid-Functioned Neural Networks», en *International Conference on Machine Learning*, 2021, pp. 2559-2567.
- [46] F. L. Saca, A. F. Ramirez, C. A. Cruz, J. V. Cortez, A. Z. López, y E. R. Martinez, «Preprocesamiento de bases de datos de imágenes para mejorar el rendimiento de redes neuronales convolucionales», *Research in Computing Science*, vol. 147, n.º 7, pp. 35-45, 2018.

- [47] P. Jiang, D. Ergu, F. Liu, Y. Cai, y B. Ma, «A Review of Yolo Algorithm Developments», *Procedia Computer Science*, vol. 199, pp. 1066-1073, 2022.
- [48] Asus, «Manual Asus FX506LI». 2021. Accedido: 10 de mayo de 2022. [En línea]. Disponible en: https://dlcdnets.asus.com/pub/ASUS/GamingNB/FX506LH/E18392_FX506_FX706_EM_V3.pdf
- [49] V. M. G. Corzo, D. G. Galindo, M. Á. R. Pinto, F. R. L. Estrada, y J. E. D. Zenteno, «Red Neuronal Artificial Backpropagation aplicada al reconocimiento de dígitos hexadecimales».
- [50] B. Zhuang, L. Liu, M. Tan, C. Shen, y I. Reid, «Training quantized neural networks with a full-precision auxiliary module», en *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1488-1497.
- [51] A. Taherkhani, G. Cosma, y T. M. McGinnity, «AdaBoost-CNN: An adaptive boosting algorithm for convolutional neural networks to classify multi-class imbalanced datasets using transfer learning», *Neurocomputing*, vol. 404, pp. 351-366, 2020.
- [52] E. H. Galvis-Serrano, I. Sánchez-Galvis, N. Flórez, y S. Zabala-Vargas, «Clasificación de Gestos de la Lengua de Señas Colombiana a partir del Análisis de Señales Electromiográficas utilizando Redes Neuronales Artificiales», *Información tecnológica*, vol. 30, n.º 2, pp. 171-180, 2019.
- [53] Ministerio del Trabajo, «Estructuras Ocupacionales – Sueldos Y Salarios Mínimos Sectoriales Y Tarifas Salarios Mínimos Sectoriales 2022». 2022. Accedido: 11 de mayo de 2022. [En línea]. Disponible en: https://drive.google.com/file/d/1mTjrMbd393IUUJksZG_KZnDIAAAc9Q0e/view

ANEXOS

Anexo A: Datasheet Dron Dji Mini 2.

Appendix

Specifications

Aircraft	
Takeoff Weight	< 249 g (International version) 199 g (JP version)
Dimensions	International version Folded: 138×81×58 mm Unfolded: 159×203×56 mm Unfolded (with propellers): 245×289×56 mm (JP version) Folded: 138×81×57 mm Unfolded: 159×202×55 mm Unfolded (with propellers): 245×289×55 mm
Diagonal Distance	213 mm
Max Ascent Speed	5 m/s (Sport Mode) 3 m/s (Normal Mode) 2 m/s (Cine Mode)
Max Descent Speed	3.5 m/s (Sport Mode) 3 m/s (Normal Mode) 1.5 m/s (Cine Mode)
Max Speed (near sea level, no wind)	16 m/s (Sport Mode) 10 m/s (Normal Mode) 6 m/s (Cine Mode)
Max Service Ceiling Above Sea Level	4,000 m (International version) 3,000 m (JP version)
Max Flight Time	31 mins (International version (measured while flying at 17 kph in windless conditions)) 18 mins (JP version (measured while flying at 17 kph in windless conditions))
Max Wind Speed Resistance	10 m/s (Scale 5)
Max Tilt Angle	40° (Sport Mode) 25° (Normal Mode) 25° (Cine Mode)
Max Angular Velocity	250°/s (Sport Mode) 250°/s (Normal Mode) 250°/s (Cine Mode)
Operating Temperature	0° to 40° C (32° to 104° F)
GNSS	GPS+GLONASS+Galileo
Operating Frequency	2.400-2.4835 GHz, 5.725-5.850 GHz
Transmission Power (EIRP)	2.4 GHz: ≤26 dBm (FCC), ≤20 dBm (CE/SRRC/MIC) 5.8 GHz: ≤26 dBm (FCC/SRRC), ≤14 dBm (CE)

Hovering Accuracy Range	Vertical: ± 0.1 m (with Vision Positioning), ± 0.5 m (with GPS Positioning) Horizontal: ± 0.3 m (with Vision Positioning), ± 1.5 m (with GPS Positioning)
-------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Gimbal

Mechanical Range	Tilt: -110° to $+35^\circ$ Roll: -35° to $+35^\circ$ Pan: -20° to $+20^\circ$
Controllable Range	Tilt: -90° to 0° (default), -90° to $+20^\circ$ (extended)
Stabilization	3-axis (tilt, roll, pan)
Max Control Speed (tilt)	100°/s
Angular Vibration Range	$\pm 0.01^\circ$

Sensing System

Downward	Hovering Range: 0.5-10 m
Operating Environment	Non-reflective, discernible surfaces with diffuse reflectivity of $>20\%$; Adequate lighting of lux >15

Camera

Sensor	1/2.3" CMOS, Effective Pixels: 12 M
Lens	FOV: 83° 35 mm format equivalent: 24 mm Aperture: f/2.8 Focus range: 1 m to ∞
ISO Range	Video 100-3200 Photo 100-3200
Electronic Shutter Speed	4-1/8000 s
Max Image Size	4:3: 4000×3000 16:9: 4000×2250
Still Photography Modes	Single shot Interval: 2/3/5/7/10/15/20/30/60 s (JPEG), 5/7/10/15/20/30/60 s (JPEG+RAW) Automatic Exposure Bracketing (AEB): 3 Frames at 2/3 EV Step
Video Resolution	4K: 3840×2160 24/25/30p 2.7K: 2720×1530 24/25/30/48/50/60p FHD: 1920×1080 24/25/30/48/50/60p
Max Video Bitrate	100 Mbps
Supported File Formats	FAT32 (≤ 32 GB) exFAT (> 32 GB)
Photo Format	JPEG/DNG (RAW)
Video Format	MP4 (H.264/MPEG-4 AVC)

Remote Controller

Operating Frequency	2.400-2.4835 GHz, 5.725-5.850 GHz
Max Transmission Distance (unobstructed, free of interference)	10 km (FCC), 6 km (CE/SRRC/MIC)
Transmission Distance (in common scenarios)	Strong interference (e.g., city center): approx. 3 km Moderate interference (e.g., outer suburbs, small towns): approx. 6 km No interference (e.g., rural areas, beaches): approx. 10 km
Operating Temperature	-10° to 40° C (14° to 104° F)

Transmitter Power (EIRP)	2.4 GHz: ≤26 dBm (FCC), ≤20 dBm (CE/SRRC/MIC) 5.8 GHz: ≤26 dBm (FCC/SRRC), ≤14 dBm (CE)
Battery Capacity	5200 mAh
Operating Current/Voltage	1200 mA@3.6 V (with Android device) 700 mA@3.6 V (with iOS device)
Supported Mobile Device Size	180×86×10 mm (Height×Width×Thickness)
Supported USB Port Types	Lightning, Micro USB (Type-B), USB-C
Video Transmission System	OcuSync 2.0
Live View Quality	720p@30fps
Max Bitrate	8 Mbps
Latency (depending on environment and mobile device)	200 ms
Charger	
Input	100-240 V, 50/60 Hz, 0.5 A
Output	12V 1.5A / 9V 2A / 5V 3A
Rated Power	18 W
Intelligent Flight Battery (International Version)	
Battery Capacity	2250 mAh
Voltage	7.7 V
Charging Voltage Limit	8.8 V
Battery Type	LiPo 2S
Energy	17.32 Wh
Weight	82.5 g
Charging Environment Temperature	5° to 40° C (41° to 104° F)
Max Charging Power	29 W
Intelligent Flight Battery (JP version)	
Battery Capacity	1065 mAh
Voltage	7.6 V
Charging Voltage Limit	8.7 V
Battery Type	LiPo 2S
Energy	8.09 Wh
Weight	48.9 g
Charging Environment Temperature	5° to 40° C (41° to 104° F)
Max Charging Power	18 W
App	
App	DJI Fly
Required Operating System	iOS v11.0 or later; Android v6.0 or later
SD Cards	
Supported SD Cards	UHS-I Speed Grade 3 rating or above microSD card

Anexo B. Código de Entrenamiento.

▼ Detector de Aisladores

- Vicente Paul Astudillo Cortez

```
!nvidia-smi
```

▼ Conexion a Google Drive

Necesitamos vincular nuestro Google Colab con google Drive para realizar backups del entrenamiento.

```
from google.colab import drive
drive.flush_and_unmount()

Drive not mounted, so nothing to flush and unmount.

drive.mount('/content/drive')
!ln -s /content/drive/My\ Drive/ /MyGDrive
!ls /MyGDrive
```

▼ Acceso a Dataset desde Google Drive

Almacenamos nuestro dataset en Google Drive y lo transferimos a Google Colab.

```
!ls /content/drive/My\ Drive/Yolov3

!cp /content/drive/My\ Drive/Yolov3/images.zip /content/
```

▼ 1) Clonando la Darknet

```
!git clone https://github.com/AlexeyAB/darknet
```

▼ 2) Compilando la Darknet usando Nvidia GPU

```
%cd darknet/

!sed -i 's/OPENCV=0/OPENCV=1/' Makefile
```

```
!sed -i 's/GPU=0/GPU=1/' Makefile
!sed -i 's/CUDNN=0/CUDNN=1/' Makefile
!make
```

▼ **3) Configurando la red para entrenamiento *

```
%cd /content/darknet/

/content/darknet

!cp /MyGDrive/Yolov3/yolov3_ygb.cfg cfg/yolov3_ygb.cfg
!cp /MyGDrive/Yolov3/yolov3_custom.cfg cfg/yolov3_custom.cfg

!cp /content/drive/My\ Drive/Yolov3/obj.names /content/darknet/data/
!cp /content/drive/My\ Drive/Yolov3/obj.data /content/darknet/data/

!unzip ../images.zip -d /content/darknet/data/colab/
```

▼ 4) Generando el archivo train.txt

```
!cp /content/drive/My\ Drive/Yolov3/generate_train.py ../

%cd /content/
!python generate_train.py
```

▼ 5) Entrenamiento de nuestro detector de objetos!

```
# Definimos una función de ayuda para mostrar imágenes
def imshow(path):
    import cv2
    import matplotlib.pyplot as plt
    %matplotlib inline

    image = cv2.imread(path)
    height, width = image.shape[:2]
    resized_image = cv2.resize(image, (3*width, 3*height), interpolation = cv2.INTER_CUBIC)

    fig = plt.gcf()
    fig.set_size_inches(18, 10)
    plt.axis("off")
    plt.imshow(cv2.cvtColor(resized_image, cv2.COLOR_BGR2RGB))
    plt.show()
```

```
%cd /content/darknet
!./darknet detector train data/obj.data cfg/yolov3_ygb.cfg /content/darknet53.conv.74 -don

#Cada 100 paso se guarda una version de avance y posterior cada 1000 esto deberia guardase
#imshow('chart.png')

%cd /content/darknet
!./darknet detector train data/obj.data cfg/yolov3_ygb.cfg /MyGDrive/Yolov3/backup/yolov3_
```

▼ 6) Ejecutando nuestro Detector de Objetos!

```
# Necesitamos configurar nuestro archivo custom.cfg para el modo de detección
%cd /content/darknet/cfg
!sed -i 's/batch=64/batch=1/' yolov3_custom.cfg
!sed -i 's/subdivisions=16/subdivisions=1/' yolov3_custom.cfg
%cd ..

    /content/darknet/cfg
    /content/darknet

# Ejecutamos nuestro detector personalizado con este comando (cargamos la(s) image(s) a nu
!./darknet detector test data/obj.data cfg/yolov3_custom.cfg /MyGDrive/Yolov3/backup/yolov
imshow('predictions.jpg')
```

Anexo C. Código Detector en Tiempo Real.

```
from tkinter import *
import tkinter
from tkinter import ttk
from tkinter import filedialog
from PIL import Image
from PIL import ImageTk
import cv2
import imutils
import numpy as np
from scipy import signal
from scipy import misc
from skimage.metrics import structural_similarity
import time
from datetime import datetime
from imutils.video import FPS
from imutils.video import VideoStream
from pydarknet import Detector
from pydarknet import Image as Image1
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
import pafy # pip3 install youtube-dl
LABELS_FILE='data/coco.names'
CONFIG_FILE='cfg/yolov3_custom.cfg'
WEIGHTS_FILE='yolov3.weights'
DATA_FILE='cfg/coco.data'
CONFIDENCE_THRESHOLD=0.3
def close():
    ventana.destroy()
#-----#MENU-----#
ventana=Tk()
ventana.title("Menu")
ventana.geometry('800x600')
ventana.resizable (0,0)
#titulos
titulo1=Label(ventana, text="UNIVERSIDAD TECNICA DE COTOPAXI",fg="black",
font=("Arial bold",25))
titulo1.place(x=90,y=0)
titulo2=Label(ventana, text="MAESTRIA EN ELECTRICIDAD MENCIÓN ",fg="black",
font=("Arial",17))
titulo2.place(x=180,y=35)
titulo6=Label(ventana, text="SISTEMAS ELECTRICOS DE POTENCIA",fg="black",
font=("Arial",17))
titulo6.place(x=180,y=60)
titulo3=Label(ventana, text="DETECCIÓN DE FALLAS EN LOS AISLADORES DE LA LINEA
",fg="black", font=("Arial",15))
titulo3.place(x=130,y=85)
titulo4=Label(ventana, text="DE TRANSMISIÓN ELÉCTRICA MEDIANTE VISIÓN
ARTIFICIAL",fg="black", font=("Arial bold",15))
titulo4.place(x=125,y=110)
#logo
logo1 = Label(ventana)
logo1.place(x=10, y=30)
```

```

image = cv2.imread("imagenes/logol.png")
logos1= imutils.resize(image, width=80)
logos1 = cv2.cvtColor(logos1, cv2.COLOR_BGR2RGB)
im1 = Image.fromarray(logos1)
img1 = ImageTk.PhotoImage(image=im1)
logol.configure(image=img1)
logol.image = img1
#logo2
logo2 = Label(ventana)
logo2.place(x=680, y=40)
image1 = cv2.imread("imagenes/logoposgrado.png")
logos2= imutils.resize(image1, width=110)
logos2 = cv2.cvtColor(logos2, cv2.COLOR_BGR2RGB)
im2 = Image.fromarray(logos2)
img2 = ImageTk.PhotoImage(image=im2)
logo2.configure(image=img2)
logo2.image = img2
#portada
portada = Label(ventana)
portada.place(x=0, y=140)
image = cv2.imread("imagenes/portada.jpg")
portada2= imutils.resize(image, width=800)
portada2 = cv2.cvtColor(portada2, cv2.COLOR_BGR2RGB)
im3 = Image.fromarray(portada2)
img3 = ImageTk.PhotoImage(image=im3)
portada.configure(image=img3)
portada.image = img3
titulo5=Label(ventana, text="AUTOR: ASTUDILLO CORTEZ VICENTE PAUL ",fg="black",
font=("Bold Arial ",15))
titulo5.place(x=180,y=570)
#Boton
btn = Button(ventana, text="Detector en Tiempo Real", width=25,command=openwind)
btn.place(x=275,y=250)
btn4 = Button(ventana, text="Detector en Imagenes", width=25,command=openwind2)
btn4.place(x=275,y=300)
btn9 = Button(ventana, text="Detector en Video", width=25,command=openwind3)
btn9.place(x=275,y=350)
btn9 = Button(ventana, text="SALIR", width=10,command=close)
btn9.place(x=675,y=550)
ventana.mainloop()

```

Anexo D. Código Detector Tiempo Real.

```
#-----ventana 1-----
def openwind():
    ventana.withdraw()
    v1=tkinter.Toplevel(ventana)
    v1.focus_set()
    v1.title("Detector en Tiempo Real")
    v1.geometry('600x400')
    v1.resizable(0,0)
    #logo
    titulo1=Label(v1, text="Detector en Tiempo Real",fg="black", font=("Arial bold",22))
    titulo1.place(x=100,y=20)
    logo1 = Label(v1)
    logo1.place(x=0, y=0)
    image = cv2.imread("imagenes/logo1.png")
    logos1= imutils.resize(image, width=80)
    logos1 = cv2.cvtColor(logos1, cv2.COLOR_BGR2RGB)
    im1 = Image.fromarray(logos1)
    img1 = ImageTk.PhotoImage(image=im1)
    logo1.configure(image=img1)
    logo1.image = img1
    #logo2
    logo2 = Label(v1)
    logo2.place(x=480, y=15)
    image1 = cv2.imread("imagenes/logosgrado.png")
    logos2= imutils.resize(image1, width=110)
    logos2 = cv2.cvtColor(logos2, cv2.COLOR_BGR2RGB)
    im2 = Image.fromarray(logos2)
    img2 = ImageTk.PhotoImage(image=im2)
    logo2.configure(image=img2)
    logo2.image = img2
    def detectar1():
        fps = FPS().start()
        fourcc = cv2.VideoWriter_fourcc(*"XVID")
        salida = cv2.VideoWriter("Aisladores/video/VideoAislador"+str(datetime.now())+".mp4",
fourcc, 10,
        (800, 600), True)
        LABELS = open(LABELS_FILE).read().strip().split("\n")
        np.random.seed(2)
        COLORS = np.random.randint(0, 255, size=(len(LABELS), 3),
        dtype="uint8")
        COLOR_LABEL={}
        for i in range(0, len(LABELS)):
            COLOR_LABEL[LABELS[i]]=COLORS[i]
        net = Detector(bytes(CONFIG_FILE, encoding="utf-8"), bytes(WEIGHTS_FILE,
encoding="utf-8"), 0, bytes(DATA_FILE,encoding="utf-8"))
        #webcam
        vs = cv2.VideoCapture(0)
        #steaming
        #url = "https://www.youtube.com/watch?v=VqTFZLceulA"
        #video = pafy.new(url)
        #best = video.getbest(preftype="mp4")
        #vs = cv2.VideoCapture()
```

```

#vs.open(best.url)
#ipcam
#vs = cv2.VideoCapture(https://192.168.1.1/videofeed)
cnt=0
while True and cnt < 1000:
    cnt+=1
    print ("Frame number", cnt)
    try:
        (grabbed, image1) = vs.read()
    except:
        break
    img_darknet = Image1(image1)
    results = net.detect(img_darknet)
    for cat, score, bounds in results:
        x, y, w, h = bounds
        color = [int(c) for c in COLOR_LABEL[str(cat)]]
        text = "{: .4f}".format(str(cat), score)
        cv2.rectangle(image1, (int(x - w / 2), int(y - h / 2)), (int(x + w / 2), int(y + h / 2)), color,
thickness=4)
        cv2.putText(image1,text,(int(x - w/2),int(y -h/2 -
5)),cv2.FONT_HERSHEY_COMPLEX,1,color)
    salida.write(cv2.resize(image1,(800,600)))
    image2=image1
    cv2.rectangle(image2, (0,450), (1280,500), (0,255,0), -1)
    cv2.putText(image2, 'Presione "s" para salir',(10,475),cv2.FONT_ITALIC ,1,10,1)
    cv2.imshow("Deteccion",cv2.resize(image2,(1280, 720)))
    fps.update()
    key = cv2.waitKey(1) & 0xFF
    if key == ord("s"):
        break
    fps.stop()
    print("[INFO] elapsed time: {:.2f}".format(fps.elapsed()))
    print("[INFO] approx. FPS: {:.2f}".format(fps.fps()))
    cv2.destroyAllWindows()
    print("[INFO] cleaning up...")
    salida.release()
    vs.release()
def star():
    ventana.deiconify()
    v1.withdraw()
#botones
lblVideo=Label(v1)
lblVideo.place(x=225,y=150)
btn1= Button(v1, text="Detectar", width=25,command=detectar1)
btn1.place(x=180,y=150)
btn7 = Button(v1, text="Regresar", width=25, command=star)
btn7.place(x=180,y=250)

```

Anexo E. Código Detector Imágenes.

```
#-----ventana 2-----
def openwind2():
    ventana.withdraw()
    v2=tkinter.Toplevel()
    v2.title("Detector en Imagenes")
    v2.geometry('1600x800')
    #logo
    titulo1=Label(v2, text="DETECTOR DE IMAGENES",fg="black", font=("Arial bold",35))
    titulo1.place(x=460,y=40)
    logo1 = Label(v2)
    logo1.place(x=50, y=10)
    image = cv2.imread("imagenes/logo1.png")
    logos1= imutils.resize(image, width=100)
    logos1 = cv2.cvtColor(logos1, cv2.COLOR_BGR2RGB)
    im1 = Image.fromarray(logos1)
    img1 = ImageTk.PhotoImage(image=im1)
    logo1.configure(image=img1)
    logo1.image = img1
    #logo2
    logo2 = Label(v2)
    logo2.place(x=1300, y=25)
    image1 = cv2.imread("imagenes/logoposgrado.png")
    logos2= imutils.resize(image1, width=200)
    logos2 = cv2.cvtColor(logos2, cv2.COLOR_BGR2RGB)
    im2 = Image.fromarray(logos2)
    img2 = ImageTk.PhotoImage(image=im2)
    logo2.configure(image=img2)
    logo2.image = img2
    def star2():
        ventana.deiconify()
        v2.withdraw()
    def detectar():
        global image
        global txt
        global lblInfo3
        count=0
        image= imutils.resize(image, width=350)
        LABELS = open(LABELS_FILE).read().strip().split("\n")
        np.random.seed(4)
        COLORS = np.random.randint(0, 255, size=(len(LABELS), 3),
            dtype="uint8")
        net = cv2.dnn.readNetFromDarknet(CONFIG_FILE, WEIGHTS_FILE)
        (H, W) = image.shape[:2]
        # determine only the *output* layer names that we need from YOLO
        ln = net.getLayerNames()
        ln = [ln[i - 1] for i in net.getUnconnectedOutLayers()]
        blob = cv2.dnn.blobFromImage(image, 1 / 255.0, (416, 416),
            swapRB=True, crop=False)
        net.setInput(blob)
        start = time.time()
        layerOutputs = net.forward(ln)
        end = time.time()
```

```

boxes = []
confidences = []
classIDs = []
# loop over each of the layer outputs
for output in layerOutputs:
    for detection in output:
        scores = detection[5:]
        classID = np.argmax(scores)
        confidence = scores[classID]
        if confidence > CONFIDENCE_THRESHOLD:
            box = detection[0:4] * np.array([W, H, W, H])
            (centerX, centerY, width, height) = box.astype("int")
            x = int(centerX - (width / 2))
            y = int(centerY - (height / 2))
            boxes.append([x, y, int(width), int(height)])
            confidences.append(float(confidence))
            classIDs.append(classID)
idxs = cv2.dnn.NMSBoxes(boxes, confidences, CONFIDENCE_THRESHOLD,
CONFIDENCE_THRESHOLD)
txt=[]

if len(idxs) > 0:
    for i in idxs.flatten():
        (x, y) = (boxes[i][0], boxes[i][1])
        (w, h) = (boxes[i][2], boxes[i][3])
        color = [int(c) for c in COLORS[classIDs[i]]]
        cv2.rectangle(image, (x, y), (x + w, y + h), color, 2)
        text = "{}".format(LABELS[classIDs[i]], confidences[i])
        cv2.putText(image, text, (x, y - 5), cv2.FONT_HERSHEY_SIMPLEX,
0.5, color, 2)
        print(i)
        j=i
        if j==1:
            txt.append(text)
        if j==0:
            txt.append(text)
    print(txt)
    lblInfo3 = Label(v2, text=txt, fg="red", font=("Arial",20))
    lblInfo3.place(x=1200,y=135)
image1=cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
im=Image.fromarray(image1)
img=ImageTk.PhotoImage(image=im)
lblOutputImage.configure(image=img)
lblOutputImage.image = img
lblInfo2 = ""
lblInfo2 = Label(v2, text="Estado de los Aisladores:", fg="red", font=("Arial bold",20))
lblInfo2.place(x=1080,y=108)
cv2.imwrite("Aisladores/Img/Aislador"+str(datetime.now())+".jpg", image)
#seleccionar imagen
def elegir_imagen():
    global txt
    txt=[]

```

```

path_image = filedialog.askopenfilename(filetypes = [
    ("image", ".jpeg"),
    ("image", ".png"),
    ("image", ".jpg")])
if len(path_image) > 0:
    global image
    image = cv2.imread(path_image)
    imageToShow= imutils.resize(image, width=350)
    imageToShow = cv2.cvtColor(imageToShow, cv2.COLOR_BGR2RGB)
    im = Image.fromarray(imageToShow )
    img = ImageTk.PhotoImage(image=im)
    lblInputImage.configure(image=img)
    lblInputImage.image = img
    lblInfo1 = Label(v2, text="IMAGEN DE ENTRADA", fg="red", font=("Arial bold",20))
    lblInfo1.place(x=130,y=110)
    lblInfo3.destroy()
img=None
#img de entrada
lblInputImage = Label(v2)
lblInputImage.place(x=110, y=150)
lblOutputImage = Label(v2)
lblOutputImage.place(x=1080, y=170)
#botones
btn5= Button(v2, text="Elegir imagen", width=25, command=elegir_imagen)
btn5.place(x=600,y=350)
btn6 = Button(v2, text="Detectar", width=25,command=detectar)
btn6.place(x=600,y=400)
btn8 = Button(v2, text="Regresar", width=25, command=star2)
btn8.place(x=600,y=500)

```

Anexo F. Código Detector en Video

```
#-----ventana 3-----
def openwind3():
    ventana.withdraw()
    v3=tkinter.Toplevel()
    v3.title("Detector en Video")
    v3.geometry('600x400')
    v3.resizable (0,0)
    #logo
    titulo1=Label(v3, text="Detector en Video",fg="black", font=("Arial bold",25))
    titulo1.place(x=135,y=20)
    logo1 = Label(v3)
    logo1.place(x=0, y=0)
    image = cv2.imread("imagenes/logo1.png")
    logos1= imutils.resize(image, width=80)
    logos1 = cv2.cvtColor(logos1, cv2.COLOR_BGR2RGB)
    im1 = Image.fromarray(logos1)
    img1 = ImageTk.PhotoImage(image=im1)
    logo1.configure(image=img1)
    logo1.image = img1
    #logo2
    logo2 = Label(v3)
    logo2.place(x=480, y=15)
    image1 = cv2.imread("imagenes/logoposgrado.png")
    logos2= imutils.resize(image1, width=110)
    logos2 = cv2.cvtColor(logos2, cv2.COLOR_BGR2RGB)
    im2 = Image.fromarray(logos2)
    img2 = ImageTk.PhotoImage(image=im2)
    logo2.configure(image=img2)
    logo2.image = img2
    def star3():
        ventana.deiconify()
        v3.withdraw()
        cap=None
#Vizualizar
    def vizualizar():
        video_path = filedialog.askopenfilename(filetypes = [
            ("all video format", ".mp4"),
            ("all video format", ".avi")])
        if len(video_path) > 0:
            lblInfoVideoPath.configure(text=video_path)
            fps = FPS().start()
            fourcc = cv2.VideoWriter_fourcc(*"XVID")
            salida = cv2.VideoWriter("Aisladores/video/VideoAislador"+str(datetime.now())+".mp4",
fourcc, 10,
            (800, 600), True)
            LABELS = open(LABELS_FILE).read().strip().split("\n")
            np.random.seed(2)
            COLORS = np.random.randint(0, 255, size=(len(LABELS), 3),
                dtype="uint8")
            COLOR_LABEL={}
            for i in range(0, len(LABELS)):
                COLOR_LABEL[LABELS[i]]=COLORS[i]
```

```

net = Detector(bytes(CONFIG_FILE, encoding="utf-8"), bytes(WEIGHTS_FILE,
encoding="utf-8"), 0, bytes(DATA_FILE, encoding="utf-8"))
vs = cv2.VideoCapture(video_path)
cnt=0

while True and cnt < 1000:
    cnt+=1
    print ("Frame number", cnt)
    try:
        (grabbed, image1) = vs.read()
    except:
        break
    img_darknet = Image1(image1)
    results = net.detect(img_darknet)
    for cat, score, bounds in results:
        x, y, w, h = bounds
        color = [int(c) for c in COLOR_LABEL[str(cat)]]
        text = "{}: {:.4f}".format(str(cat), score)
        cv2.rectangle(image1, (int(x - w / 2), int(y - h / 2)), (int(x + w / 2), int(y + h / 2)), color,
thickness=2)
        cv2.putText(image1, text, (int(x - w/2), int(y - h/2 -
5)), cv2.FONT_HERSHEY_COMPLEX, 1, color)

        salida.write(cv2.resize(image1, (800, 600)))
        image2 = image1
        cv2.rectangle(image2, (0, 590), (1280, 640), (0, 255, 0), -1)
        cv2.putText(image2, 'Presione "s" para salir', (10, 625), cv2.FONT_ITALIC, 1, 10, 1)
        cv2.imshow("Deteccion", cv2.resize(image2, (1280, 720)))
        imgplot = plt.imshow(image2)
        fps.update()
        key = cv2.waitKey(1) & 0xFF
        if key == ord("s"):
            break
        fps.stop()
        print("[INFO] elapsed time: {:.2f}".format(fps.elapsed()))
        print("[INFO] approx. FPS: {:.2f}".format(fps.fps()))
        cv2.destroyAllWindows()
        print("[INFO] cleaning up...")
        salida.release()
        vs.release()
    else:
        lblInfoVideoPath.configure(text="No se ha seleccionado un video")
        lblInfoVideoPath1 = Label(v3, text="Video de Entrada:", font=("Arial", 12))
        lblInfoVideoPath1.place(x=0, y=100)
        lblInfoVideoPath = Label(v3, text="No se ha seleccionado un video", font=("Arial", 12))
        lblInfoVideoPath.place(x=150, y=100)
        btn11 = Button(v3, text="Elegir Video", width=25, command=vizualizar)
        btn11.place(x=180, y=150)
        btn13 = Button(v3, text="Regresar", width=25, command=star3)
        btn13.place(x=180, y=300)

```

Anexo G. Manual de Usuario.



**Identificación De Fallas En Los Aisladores De Una Línea De
Transmisión Mediante Visión Artificial**

Manual de Usuario

Versión: 1.0

2022

ADVERTENCIA



Para el uso correcto del Algoritmo de Identificación de falla en los Aisladores es necesario leer las siguientes instrucciones:

Seguir todas las indicaciones antes de utilizar la aplicación.

Verificar los requisitos mínimos del sistema.

Conocer las restricciones de uso de drones en Ecuador.

<https://www.aviacioncivil.gob.ec/wp-content/uploads/downloads/2020/11/5-DGAC-DGAC-2020-0110-R-Reglamento-de-RPAs.pdf>



**Identificación De Fallas En Los Aisladores De Una
Línea De Transmisión Mediante Visión Artificial.
Manual de Usuario**



1	DESCRIPCIÓN DEL SISTEMA	4
1.1	Alcance	4
1.2	Funcionalidad	4
1.3	Requisitos Mínimos del Sistema	4
1.4	Programas Requeridos	5
1.5	Librerías de Python Requeridas	5
2	DESCRIPCIÓN DEL SISTEMA	6
2.1	Detector en Tiempo Real	7
2.2	Detector en Imágenes	8
2.3	Detector en Video	11
3	PREGUNTAS FRECUENTES	13



1 DESCRIPCIÓN DEL SISTEMA

El siguiente programa fue desarrollado para el análisis del estado de los aisladores cerámicos de las líneas de transmisión de 13.8 kV, con el objetivo de disminuir el tiempo de análisis y toma de decisión al momento de inspeccionar estos elementos, de igual manera se busca disminuir el costo hacia la empresa encargada, ya que no es necesario contar con el carro grupo para acceder a los aisladores. Y finalmente evitar cualquier tipo de accidente que podría verse afectado el personal de mantenimiento.

1.1 Alcance

El algoritmo ha sido delimitado para los aisladores de porcelana del tipo campana de la línea de transmisión de 13.8 kV de la provincia de Cotopaxi

1.2 Funcionalidad

Permite detectar de forma automática el estado de los aisladores de porcelana, mediante el uso de un dron y un computador cargado con la red neuronal, ha sido diseñado para que su uso se limite solo a un operador.

1.3 Requisitos Mínimos del Sistema

Procesador	Intel I5-6600 / AMD FX-6300
RAM	8GB
Tarjeta de Video	Nvidia Geforce GTX 760
Almacenamiento Disponible	1 GB
Sistema Operativo	Ubuntu 20



1.4 Programas Requeridos

- Python 3.0 o superior
- OpenCV 4.2 o superior
- Cuda Toolkit 10 o superior
- Cudnn 7 o superior

1.5 Librerías de Python Requeridas

- Tkinter
- Pillow
- Imutils
- Numpy
- Scipy
- Matplotlib





2 DESCRIPCIÓN DEL SISTEMA

Se ha desarrollado una interfaz gráfica amigable con el usuario para que pueda ser utilizada por cualquier persona con conocimientos o no en programación.

Al iniciar la interfaz gráfica desarrollada en Python aparecerá la siguiente ventana:




Figura 1. Menú Principal.

Se puede visualizar 4 botones principales, en los cuales se pueden elegir 3 maneras de identificar fallas en los aisladores.

- ❖ **Botón Detector en Tiempo Real**
- ❖ **Botón Detector en Imágenes**
- ❖ **Botón Detector en Video**
- ❖ **Botón Salir**

	<p align="center">Identificación De Fallas En Los Aisladores De Una Línea De Transmisión Mediante Visión Artificial. Manual de Usuario</p>	
-----------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------

2.1 *Detector en Tiempo Real*

	<p align="center"><i>Antes de ingresar al algoritmo de Identificación de Aisladores, es necesario tener en cuenta la dirección ip del dispositivo de captura de imágenes, para que no exista ningún tipo de error, la dirección puede ser cambiada en la línea 76 del código principal.</i></p>
-----------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Al presionar el botón de tiempo real, nos aparecerá una nueva ventana que consta de dos botones:



Figura 2. Menú Detector en Tiempo Real.

- ❖ **Botón Detectar**
- ❖ **Botón Regresar**

Colocar el dron en un Angulo adecuado para la toma de video, tomando en consideración la luz solar.

Al presionar “Detectar” nos permite ingresar al algoritmo para detectar fallas en los aisladores en tiempo real, antes de iniciar es recomendable posicionar al dron a una distancia de seguridad de las líneas de transmisión y en un ángulo correcto.



**Identificación De Fallas En Los Aisladores De Una Línea De Transmisión Mediante Visión Artificial.
Manual de Usuario**



Es importante mantener al dron a una distancia de seguridad para que no se vea afectado por los campos electromagnéticos producidos por los conductores, se recomienda 2 metros para un voltaje de 13.8 kV



Figura 3. Detector en Tiempo Real.

Aparecerá una nueva ventana con el video en tiempo real, siendo detectado fotograma a fotograma, si se desea salir de la ventana será necesario presionar la tecla “s” para salir, lo que nos enviará nuevamente a la ventana de detección en tiempo real.

Si se presiona el botón Regresar, cerrara la ventana actual y volveremos al menú principal.

2.2 Detector en Imágenes

Al presionar el botón de Detector de Imágenes nos aparecerá una nueva ventana que consta de dos botones:



Figura 4. Menú Detector en Imágenes.

- ❖ **Botón Elegir Imagen**
- ❖ **Botón Detector en Imágenes**
- ❖ **Regresar**

Para empezar a analizar imágenes, se debe presionar el botón elegir imagen.

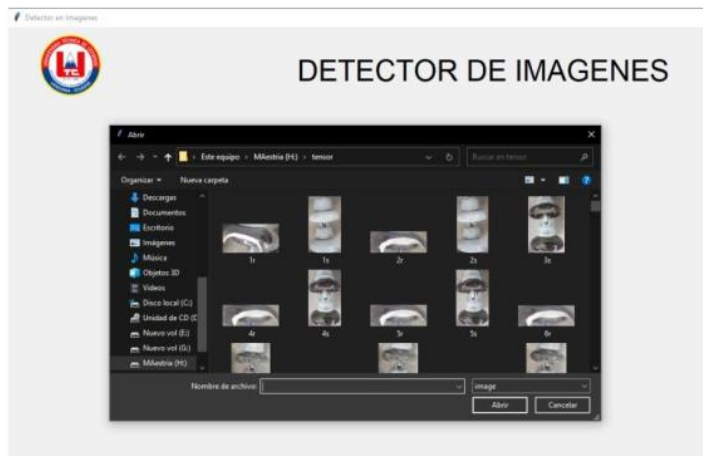


Figura 5. Ventana para seleccionar imágenes.

Nos aparecerá una nueva ventana en la cual debemos elegir la imagen del aislador a analizar, una vez elegida la imagen nos aparecerá al lado izquierdo con el título imagen de entrada.



Figura 5. Detector de Imágenes.

Al presionar el botón Detectar, se procede a analizar la imagen mediante la red neuronal, mostrando el resultado al lado derecho de la pantalla con el título imagen de salida, así como también el estado del aislador.



Figura 6. Imagen Detectada.

Si se presiona el botón Regresar, cerrara la ventana actual y volveremos al menú principal.

	<p align="center">Identificación De Fallas En Los Aisladores De Una Línea De Transmisión Mediante Visión Artificial. Manual de Usuario</p>	
-----------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------

2.3 *Detector en Video*

Al presionar el botón Detector en Video, nos aparecerá una nueva ventana que consta de dos botones:



Figura 2. Menú Detector en Video.

- ❖ **Botón Elegir Video**
- ❖ **Botón Regresar**

Al presionar “Elegir Video” nos aparecerá una nueva ventana en la que nos pedirá elegir el video a detectar.

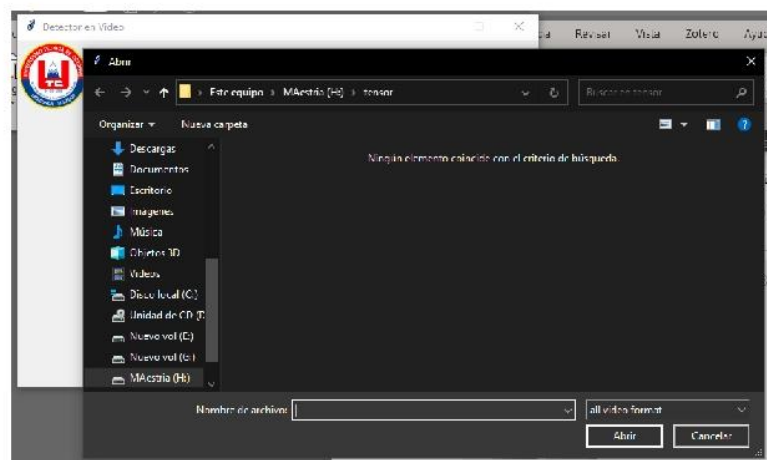
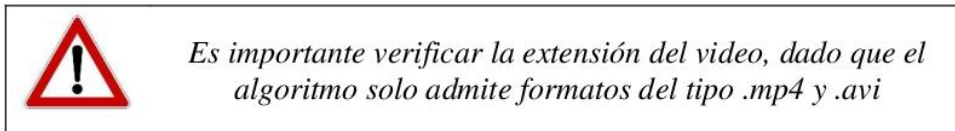


Figura 8. Ventana para seleccionar videos.



Al elegir el video y presionar el botón Abrir, automáticamente el video será analizado por la red neuronal.



Figura 9. Fotograma del video detectado.

Si se desea salir de la ventana de detección es necesario presionar la tecla “s”, y volveremos a la ventana principal de detección de videos.

Si se presiona el botón Regresar, cerrara la ventana actual y volveremos al menú principal.

Aparecerá una nueva ventana con el video en tiempo real, siendo detectado fotograma a fotograma, si se desea salir de la ventana será necesario presionar la tecla “s” para salir.

	<p>Identificación De Fallas En Los Aisladores De Una Línea De Transmisión Mediante Visión Artificial. Manual de Usuario</p>	
-----------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------

3 PREGUNTAS FRECUENTES



¿Al presionar el Botón Detectar en la ventana de Tiempo Real no pasa nada?

Esta situación se debe a que el computador elegido no cuenta con las características mínimas para el correcto funcionamiento.

¿Al momento de mostrar el video en tiempo real esta lento?

Esta situación se debe a que el computador elegido no cuenta con las características mínimas para tener una tasa aceptable de frames y verlo de una manera fluida.

¿Al presionar el Botón Detectar en la ventana de Tiempo Real aparece un error inesperado?

Se debe a que al momento de iniciar el programa no se tomó en consideración la dirección ip del drone, revisar la línea 76 del código principal y modificarla.

¿Al buscar el video en nuestro computador no aparece?

El video en cuestión se encuentra en un formato no admitido por el algoritmo, se recomienda convertirlo a formato MP4 o AVI.

Anexo H. Líneas de Transmisión de Cotopaxi [8].

Mapa Nro. 41: Infraestructura eléctrica de E.E. Cotopaxi

