

**UNIVERSIDAD POLITÉCNICA SALESIANA**

**SEDE CUENCA**

**CARRERA: INGENIERÍA ELECTRÓNICA**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE:  
INGENIERO ELECTRÓNICO**

PROCESAMIENTO DE IMÁGENES MEDIANTE EL USO DE UN VEHÍCULO AÉREO  
NO TRIPULADO (UAV) PARA EL MONITOREO Y DETECCIÓN DE INCENDIOS  
FORESTALES

**Autores:**

Jean Carlo Macancela Poveda

Pablo Moisés Lucero Otorongo

**Tutor:**

Ing. Diego Cabrera. M.Sc.

**CUENCA, JUNIO DE 2016**

Datos de catalogación bibliográfica

MACANCELA POVEDA JEAN CARLO. - LUCERO OTORONGO PABLO MOISES.

**Procesamiento de Imágenes mediante el uso de un vehículo aéreo no tripulado (UAV) para el monitoreo y detección de incendios forestales**

Universidad Politécnica Salesiana, Cuenca – Ecuador, 2016

INGENIERÍA ELECTRÓNICA

Formato 170 x 240 mm

Páginas: 55

*Breve reseña de los autores e información de contacto:*



**Jean Carlo Macancela Poveda.**

Candidato a Ingeniero en Ingeniería Electrónica por la Universidad Politécnica Salesiana – Sede Cuenca.

[jmacancelap@est.ups.edu.ec](mailto:jmacancelap@est.ups.edu.ec)



**Pablo Moisés Lucero Otorongo.**

Candidato a Ingeniero en Ingeniería Electrónica por la Universidad Politécnica Salesiana – Sede Cuenca.

[pluceroo@est.ups.edu.ec](mailto:pluceroo@est.ups.edu.ec)

*Tutor:*



**Ing. Diego Román Cabrera Mendieta, M. Sc.**

Ingeniero en Electrónico por la Universidad Politécnica Salesiana – Sede Cuenca. Master en Lógica, Computación e Inteligencia Artificial por la Universidad de Sevilla – España. Candidato a Doctorado en Ingeniería informática línea de investigación en Ciencias de la Computación e Inteligencia Artificial por la Universidad de Sevilla España. Investigador en el Grupo de Investigación y Desarrollo de Tecnologías Industriales. Docente de la Carrera de Ingeniería Electrónica

[dcabrera@ups.edu.ec](mailto:dcabrera@ups.edu.ec)

Todos los derechos reservados.

Queda prohibida, salvo excepción prevista en la Ley, cualquier forma de reproducción, distribución, comunicación pública y transformación de esta obra para fines comerciales, sin contar con autorización de los titulares de propiedad intelectual. La infracción de los derechos mencionados puede ser constitutiva de delito contra la propiedad intelectual. Se permite la libre difusión de este texto con fines académicos investigativos por cualquier medio, con la debida notificación a los autores.

DERECHOS RESERVADOS

©2016 Universidad Politécnica Salesiana.

CUENCA – ECUADOR – SUDAMERICA

MACANCELA POVEDA JEAN CARLO. – LUCERO OTORONGO PABLO MOISES.

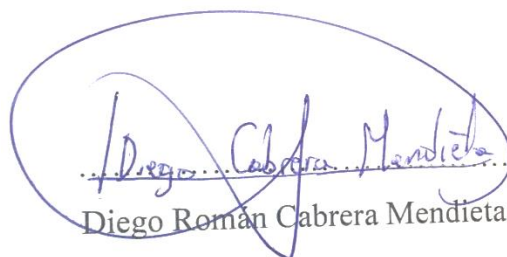
Procesamiento de Imágenes mediante el uso de un vehículo aéreo no tripulado (UAV) para el monitoreo y detección de incendios forestales.

IMPRESO EN ECUADOR – PRINTED IN ECUADOR

## DECLARATORIA DE COAUTORÍA DEL DOCENTE TUTOR

Yo, Diego Román Cabrera Mendieta declaro que bajo mi dirección y asesoría fue desarrollado el trabajo de titulación “*PROCESAMIENTO DE IMÁGENES MEDIANTE EL USO DE UN VEHÍCULO AEREO NO TRIPULADO (UAV) PARA EL MONITOREO Y DETECCIÓN DE INCENDIOS FORESTALES*” realizado por Pablo Moisés Lucero Otorongo y Jean Carlo Macancela Poveda, obteniendo un producto que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana para ser considerados como trabajo final de titulación.

Cuenca, 8 de junio del 2016.



Diego Román Cabrera Mendieta

CI: 010402830-3.

## DECLARATORIA DE RESPONSABILIDAD

Los autores son los únicos responsables por los contenidos, conceptos, ideas, análisis, resultados investigativos y manifestados en el presente trabajo de titulación “Procesamiento de Imágenes mediante el uso de un vehículo aéreo no tripulado (UAV) para el monitoreo y detección de incendios forestales”.

Cuenca, 8 de junio del 2016.



JEAN CARLO MACANCELA  
CI: 010508363-8



PABLO LUCERO  
CI: 010496385-5

## © DERECHOS DE AUTOR


Nosotros Pablo Moisés Lucero Otorongo y Jean Carlo Macancela Poveda, por medio del presente documento certificamos que hemos leído la Política de Propiedad Intelectual de la Universidad Politécnica Salesiana y estamos de acuerdo con su contenido, por lo que los derechos de propiedad intelectual del presente trabajo quedan sujetos a lo dispuesto en la Política.

Asimismo, autorizamos a la Universidad Politécnica Salesiana para que realice la digitalización y publicación de este trabajo en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Cuenca, 8 de Junio del 2016



JEAN CARLO MACANCELA  
CI: 010508363-8



PABLO LUCERO  
CI: 010496385-5

## **AGRADECIMIENTO**

Queremos expresar nuestro más sincero agradecimiento a nuestro tutor de proyecto técnico Ing. Diego Cabrera M.Sc. , al director del Grupo de Investigación y Desarrollo En Tecnologías Industriales (GIDTEC) Ing. Vinicio Sanchez y a todos los miembros del grupo, por todo el apoyo y la ayuda brindada para realizar el presente trabajo de investigación, quienes con su conocimiento y de una manera desinteresada, han sabido enseñarnos a la vez que guiarnos para ejecutar adecuadamente los procesos del presente proyecto.

**Pablo Moises - Jean Carlo**

# **DEDICATORIA**

A Dios, a mis abuelos Alfonso y Luz (+) que siempre soñaron con verme como un profesional, a mis padres por darme la oportunidad de ser alguien en la vida, a mi tíos y a mis primos por sus guías, apoyo y buenos consejos en todo momento, a mis amigos y gente allegada que de una u otra manera siempre estuvieron ahí para brindarme una mano amiga cuando mas lo necesitaba, gracias a todos por ayudarme a superar y seguir adelante.

**Jean Carlo Macancela Poveda**

Dedico este trabajo principalmente a Dios, por haberme dado la vida y permitirm el haber llegado hasta este momento ta importante de mi formacion profesional. A mi madre, por ser el pilar mas importante y por demostrarme siempre su cariño y apoyo incondicional sin importar nuestras diferencias de opiniones. A mi hermano por apoyarme en esos momentos de necesidad. A mi compañero y amigo, Jean Carlo por que sin el equipo que formanos, no hieramos logrado esta meta.

**Pablo Moises Lucero Otorongo**

# RESUMEN

Muchos de los sistemas actuales de detección, monitoreo y supresión de incendios están basados en la observación humana y son de carácter estático, es decir, se ubican puntos de vigilancia en las zonas de riesgo y una persona es la encargada de monitorear el sector en búsqueda de posibles incendios a través de binoculares o cámaras (digitales, térmicas, etc.), en caso de detectar la presencia de fuego el operador es quien informa a las diferentes entidades de socorro para los procedimientos de supresión de fuego.

Debido a que los sistemas tradicionales no disponen de flexibilidad y dadas las limitaciones humanas y de acceso al terreno, es necesario el desarrollo de sistemas integrales para la detección y monitoreo de incendios forestales, por lo que se ha optado por el uso de una UAV (drone) al cual se le ha conectado con un sistema encargado de las tareas de detección y reconocimiento de incendios.

El presente trabajo cuenta con tres fases importantes; (i) Generación de la base de datos, mediante la recopilación de fotos utilizando el ARDrone 2.0 y el navegador Google Imágenes. (ii) Diseño de la arquitectura para el sistema, tomando en cuenta criterios como; el tamaño de imágenes de 64x64 píxeles, el tipo de red neuronal que este caso fue una Red Neuronal Convolutiva con una etapa de clasificación a la salida del tipo Perceptron Multipaca, y las respectivas métricas de evaluación. (iii) implementación del sistema; se diseñó una interfaz simple donde se observa la imagen capturada por el UAV y en la parte inferior una alarma de alerta de presencia de incendio forestal.



# ÍNDICE GENERAL

AGRADECIMIENTO .....	VI
DEDICATORIA .....	VII
RESUMEN .....	VIII
ÍNDICE GENERAL .....	IX
ÍNDICE DE FIGURAS .....	XI
ÍNDICE DE TABLAS .....	XIII
ANTECEDENTES .....	XIV
INTRODUCCIÓN .....	XVI
OBJETIVOS .....	XVII
General.....	XVII
Específicos.....	XVII
CAPÍTULO 1. ESTADO DEL ARTE.....	1
1.1 INCENDIOS FORESTALES .....	1
1.2 DETECCION DE INCENDIOS FORESTALES .....	2
1.2.1 METODOS TRADICIONALES PARA EL CONTROL, DETECCION Y SUPRESION DE FUEGO .....	3
1.2.2 METODOS ACTUALES PARA LA DETECCION DE FUEGO .....	3
1.3 COMPARACION ENTRE LOS DIFERENTES METODOS DE DETECCION DE INCENDIOS FORESTALES .....	6
CAPÍTULO 2. EXTRACCION DE CARACTERISTICAS EN IMAGENES .....	8
2.1 TECNICAS DE EXTRACCION TRADICIONALES DE CARACTERISTICAS .....	10
2.1.1 CARACTERISTICAS DE COLOR .....	10
2.1.2 CARACTERISTICAS DE TEXTURA .....	14
2.1.3 CARACTERISTICAS DE FORMA.....	17
2.2 MODELOS DE EXTRACCIÓN DE ATRIBUTOS AUTOMÁTICOS .....	18
2.2.1 AUTOCODIFICADORES .....	18
2.2.2 AUTOCODIFICADOR DISPERSO .....	19
2.2.3 REDES CONVOLUCIONALES.....	20
2.2.4 MAQUINA RESTRINGIDA DE BOLTZMANN .....	21
CAPÍTULO 3. METODOLOGIA PROPUESTA .....	22
3.1 ARQUITECTURA RED CONVOLUCIONAL.....	23
3.2 FASE DE ENTRENAMIENTO .....	26
3.3 FASE DE EVALUACION Y PRUEBA .....	28

3.3.1	METRICAS USADAS .....	28
3.4	FASE DE EVALUACION EN TIEMPO REAL DEL SISTEMA.....	30
CAPÍTULO 4. CONFIGURACIÓN EXPERIMENTAL .....		31
4.1	UNIDAD DE VUELO NO TRIPULADA (UAV) .....	31
4.2	HARDWARE UTILIZADO.....	32
4.3	SOFTWARE UTILIZADO .....	32
4.4	OBTENCION DE IMAGENES .....	33
4.5	ETAPA DE PRE-PROCESAMIENTO Y GENERACION DE LA BASE DE DATOS.....	35
CAPÍTULO 5. ANALISIS DE RESULTADOS .....		42
5.1	RESULTADOS DEL ENTRENAMIENTO.....	42
CONCLUSIONES .....		46
RECOMENDACIONES.....		47
TRABAJOS FUTUROS.....		47
REFERENCIAS .....		48

## ÍNDICE DE FIGURAS

Figura 1: Incendios forestales y precipitación acumulada de 1 mes de julio (2010-2013) [5]. ....	XIV
Figura 2: Triangulo del Fuego .....	1
Figura 3: Métodos para el control y detección de fuego potencial en la zona de riesgo .....	3
Figura 4: Ciclo de un Sistema de Detección de Incendios Basado en Satélite .....	4
Figura 5: Sistemas de Detección Basados en sensores inalámbricos.....	5
Figura 6: Sistemas de Detección basados en cámaras infrarrojas y ópticas .....	6
Figura 7: Proceso típico en aplicaciones basados en clasificadores de imágenes .....	8
Figura 8: Secuencia típica para la clasificación de una imagen.....	9
Figura 9: Espacios de Color: a) Espacio de Color RGB. b) Espacio de Color HSV .....	10
Figura 10: Ejemplo de representación de un histograma.....	14
Figura 11: Texturas diferentes con la misma distribución de intensidad (color).....	14
Figura 12: Estructura de un autocodificador.....	19
Figura 13: Autocodificador Disperso .....	20
Figura 14: Arquitectura de una red neuronal convolucional (CNN). .....	20
Figura 15: Arquitectura de Maquina restringida de Boltzmann. ....	21
Figura 16: Metodología Clásica de un sistema de clasificación mediante imágenes .....	22
Figura 17: Metodología Con Redes Neuronales Convolucionales (CNN).....	23
Figura 18: Componentes Típicos de la capa de una red neuronal convolucional.....	24
Figura 19: Técnica de Max-Pooling (sub-muestreo) .....	25
Figura 20: Secuencia de procesos en una Red Neuronal Convolucional.....	26
Figura 21: Mosaico de la base de Datos usadas para el entrenamiento: (Izq) Imágenes que contienen No Fuego. (Der) Imágenes que contienen Fuego.....	26
Figura 22: Preparación e inicio del Entrenamiento.....	27
Figura 23: Secuencia completa del sistema de detección de Incendios Forestales.....	30

Figura 24: Proceso General del Sistema de Detección de Incendios.....	30
Figura 25: Ubicación geográfica del cantón Santa Isabel.....	34
Figura 26: Ubicación geográfica de la comunidad de Rodeo en la parroquia Baños .....	34
Figura 27: Algoritmo para la transformación de recorte .....	36
Figura 28: Algoritmo para la transformación de rotación .....	37
Figura 29: Algoritmo para la transformación de escalado.....	38
Figura 30: Algoritmo para la generación del dataset.pkl.....	40
Figura 31: Algoritmo para de entrenamiento.....	41
Figura 32: Diagrama de desarrollo del sistema.....	41
Figura 33: Curvas de error para los datos de prueba y validación para la prueba 7 .....	45

## ÍNDICE DE TABLAS

Tabla 1: Tabla comparativa entre diferentes métodos de detección de incendios .....	7
Tabla 2: Información de los Descriptores de Color .....	12
Tabla 3: Momentos de Color e Indexación.....	13
Tabla 4: Resumen 5 características estándar de la matriz de co-ocurrencia.....	15
Tabla 5: Momentos de los filtros de Gabor .....	16
Tabla 6: Descriptores de textura .....	17
Tabla 7: Matriz de Confusión de 2x2 .....	29
Tabla 8: Características del AR Drone 2.0 .....	31
Tabla 9: Especificaciones de la GPU.....	32
Tabla 10: Características técnicas del equipo .....	32
Tabla 11: Imágenes Iniciales sin transformación.....	35
Tabla 12: Transformaciones Geométricas .....	38
Tabla 13: Imágenes escaladas (Base de datos Final).....	39
Tabla 14: Parámetros para cada Prueba.....	43
Tabla 15: Métricas de Rendimiento análisis por clase.....	43
Tabla 16: Métricas de rendimiento por prueba.....	44
Tabla 17: Tiempos de Entrenamiento.....	44
Tabla 18: Matriz de Confusión de la Prueba 8. ....	44

## ANTECEDENTES

Según la Organización de las Naciones Unidas para la Agricultura y la Alimentación (FAO) [1], los bosques cubren el 30,6% de la zona terrestre del planeta, 3.999 millones de hectáreas, de las cuales el 13%, 519,87 millones de hectáreas, está destinado a la conservación de la biodiversidad. Los bosques, por otra parte, satisfacen la creciente demanda de productos y servicios; desde 1990 más de 200 millones de  $m^3$  de madera han sido extraídos para solventar las necesidades de la población mundial.

Según datos del ministerio, de la Secretaria de Gestión de Riesgos (SGR) de Ecuador, en julio, durante el periodo 2010 - 2013, el registro de incendios forestales tiende a incrementarse en los últimos años, debido a una tendencia descendente de las precipitaciones, como lo muestra la Figura 1. Las variables climatológicas en conjunto a eventos desfavorables, aumenta la probabilidad de un incendio forestal [5].

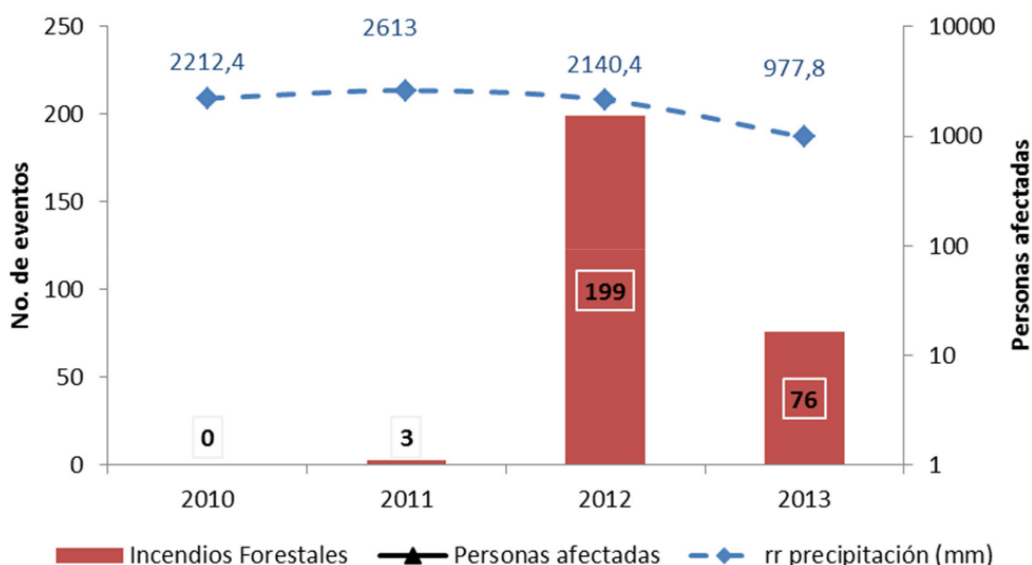


Figura 1: Incendios forestales y precipitación acumulada de 1 mes de julio (2010-2013) [5].

Otros datos de la SGR, presentan que la población, a nivel nacional, potencialmente amenazada (escenario con probabilidades climatológicas bajo la normal; 1.6 a 146mm para la Sierra y 67 a 94mm para la Costa), para el mes de Julio de 2014, fue de 169.065 habitantes en zonas con probabilidades de lluvia bajo la normal (60-100%), el mayor porcentaje pertenece a la provincia de Manabí, cantones; Montecristi (26%) y Manta (28%). De forma general la SRG presenta:

- Alrededor del 92,89% de la población potencialmente amenazada, del Ecuador, se encuentran en zonas de alta amenaza de incendios y 7,11% restante de muy alta amenaza.
- En cuanto a las zonas vegetales con un alto potencial de amenaza, es debido a una baja probabilidad de lluvias tienden a ser zonas de amenaza de incendios forestales alta y muy

alta, en mayor porcentaje se encuentran áreas Agropecuarias (38%) y Bosque nativo (25%) potencialmente amenazada.

- Dentro de los servicios básicos potencialmente amenazados se encuentra la infraestructura vial con un total de 515Km de vías, de las cual el 98% se encuentra en zona de alta amenaza y el 2% en muy alta amenaza; en segundo tenemos la infraestructura eléctrica: cinco subestaciones, estructuras FODUR, 67 estructuras de transmisión y tres centrales hidroeléctricas, de ellos se ubican en alta amenaza 67% y el 33% en muy alta amenaza; en tercero esta la infraestructura de salud con un total de 39 centros, de los cuales el 100% se ubican en zonas de alta amenaza y en cuarto lugar tenemos la infraestructura de educación, en la cual alrededor de 275 centros se localizan en zonas de amenaza alta 98% y en zonas de muy alta amenaza 2%.

Para el del 2015 alrededor de 3.102 hectáreas, entre pastizales, pajonales, asbestos y boscosas, fueron quemadas solo en Quito [6]. Por otra parte el Parque Nacional “El Cajas”, ubicado en la provincia del Azuay, registró alrededor de 40 hectáreas consumidas por el fuego en este mismo año [7].

# INTRODUCCIÓN

Según datos de la Secretaria de Gestión de Riesgos de Ecuador, en los últimos años se ha dado un incremento en la generación de incendios forestales, especialmente en épocas de verano [5].

Para los bosques, los incendios forestales son parte del balance ecológico, ya sean naturales o provocados, destruyendo viejas poblaciones de plantas y árboles, permitiendo el nacimiento de nueva vegetación más productiva. Estos incendios son favorables siempre y cuando se mantengan dentro de un rango natural estable [2].

Según E. Rochford [3], anualmente se producen más de 100 mil incendios forestales en Estados Unidos, distribuidos en 50 de sus estados, quemando alrededor de 1.6 y 2 millones de hectáreas.

Un artículo presentando en la “International Wildland Fire Conference 2015” [4], da a conocer que en Sur América, entre los años de 2005 y 2014 se ha registrado alrededor de 2.9 millones de incendios, de los cuales 60% detectados en Brasil, 10% en Argentina, 10% en Bolivia, y 20% en el resto de países.

Estos incidentes han afectado no solo la vegetación y el hábitat de diferentes especies, también han desplazado a moradores de las zonas aledañas a las catástrofes con el fin de salvaguardar su vida. Las infraestructuras críticas potencialmente amenazadas por los incendios forestales han sido: vial, eléctrica, salud y educación [5].

La detección temprana de estos acontecimientos permite mitigar los daños a posteriori y también utilizar eficientemente los recursos disponibles. Por esta razón es de suma importancia el desarrollo de herramientas que permitan brindar un soporte para la detección temprana de incendios y gracias al avance tecnológico es posible llevar a cabo esta actividad mediante el uso de sistemas remotos.

Por este motivo, el presente trabajo busca desarrollar un sistema inteligente para la detección y monitoreo de incendios forestales mediante el uso de un vehículo aéreo no tripulado (Unmanned Aerial Vehicle, UAV). El cual se encuentra estructurado de la siguiente manera: Capítulo 1 Estado del Arte, Capítulo 2 una revisión de las técnicas de extracción de características de imágenes, Capítulo 3 Metodología propuesta, Capítulo 4 Configuración experimental, Capítulo 5 Análisis de Resultados y finalmente se presentan las conclusiones, recomendaciones y trabajos futuros.



# **OBJETIVOS**

## **General**

Desarrollar un sistema para el monitoreo y detección de incendios forestales a través de procesamiento de imágenes mediante el uso de un vehículo no tripulado (drone).

## **Específicos**

- Investigar la bibliografía correspondiente a incendios forestales y unidades de vuelo no tripuladas.
- Crear una base de datos con imágenes de Incendios y de No Incendios.
- Diseñar el algoritmo entrenamiento para la identificación de fuego y humo.
- Diseñar la interfaz gráfica para la integración de los algoritmos correspondientes.



# CAPÍTULO 1. ESTADO DEL ARTE

## 1.1 INCENDIOS FORESTALES

La detección temprana de incendios forestales ha sido desde siempre una necesidad vital a la hora de tomar acciones para el control y/o extinción del fuego. Este trabajo es cada vez más eficiente gracias a los avances tecnológicos con los que se cuentan actualmente dado que la información que se recolecta es más rápida y precisa.

Como se ha explicado a lo largo de los antecedentes, los incendios forestales son un fenómeno recurrente a nivel mundial, cuya presencia viene ocasionando daños significativos cada año en el medio ambiente, elevados costes económicos, inseguridad ciudadana e incluso pérdidas de vidas humanas.

### *TRIANGULO DEL FUEGO*

Para que el fuego pueda producirse, deben estar presentes tres elementos en una adecuada proporción (véase Figura 2), estos elementos son:

- Calor (Temperatura)
- Oxígeno
- Combustible

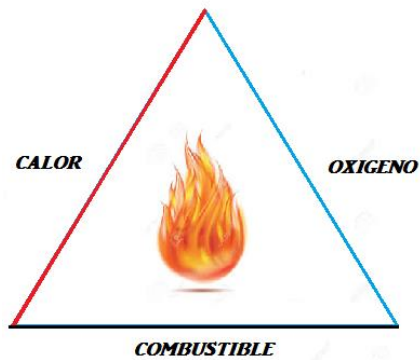


Figura 2: Triangulo del Fuego

A continuación se expondrán breves conceptos relacionados con los elementos del triángulo del fuego:

### **Calor**

El calor según el diccionario de Oxford es la “Energía que se manifiesta por un aumento de temperatura y procede de la transformación de otras energías; es originada por los movimientos vibratorios de los átomos y las moléculas que forman los cuerpos.”. En términos generales, el calor (temperatura) de un combustible tiene que elevarse para que este alcance su temperatura de ignición y empiece a arder.

## **Oxígeno**

El oxígeno es un elemento químico de número atómico 8, masa atómica 15,99 y símbolo O; es un gas incoloro e inodoro que se encuentra en el aire, en el agua, en los seres vivos y en la mayor parte de los compuestos orgánicos e inorgánicos en el planeta; es esencial en la respiración y en la combustión [20].

## **Combustible**

Con el nombre de combustible se le conoce a cualquier Sustancia o materia que al combinarse con oxígeno es capaz de reaccionar desprendiendo calor, especialmente las que se aprovechan para producir calor, que arde o es capaz de arder [21].

El fuego aparece cuando estos 3 elementos están presentes en la proporción correcta. Este concepto es la base para cualquier actividad de extinción.

## **TIPOS DE HUMOS**

El humo que se divisa durante un incendio tiene diferentes características y estas vienen dadas por la fuente (combustible) que tienen las llamas para propagarse. Algunas de las características de los humos se explican a continuación:

### **Por su Color:**

- **Blanco:** Quema de combustibles ligeros (pastizales, etc.)
- **Gris claro:** Quema de combustibles medios como matorrales.
- **Gris oscuro:** Quema de combustibles pesados como árboles.
- **Amarillento:** Tonalidad por quema de resinosas.

### **Por su Textura:**

- **Ligera:** Poca densidad, liviano. Indica que existe poca cantidad de combustible y que se encuentra disperso
- **Densa:** Humo espeso. Nos indica presencia de gran cantidad de combustible y/o donde la combustión es muy intensa.

Cabe mencionar que existen diferentes orígenes del humo pudiéndolos clasificar como falsos, legítimos periódicos, eventuales e ilegítimos. Dentro de todos estos, los humos que se busca detectar a tiempo son los ilegítimos puesto a que son de procedencia desconocida pudiendo ser un incendio [12].

## **1.2 DETECCIÓN DE INCENDIOS FORESTALES**

El fuego y el humo son dos elementos con características distintas: mientras que el fuego tiene colores vivos, se mantiene siempre en un mismo sitio pero es muy variante; el humo tiene unos colores muy apagados y tiende a moverse expandiéndose por el aire; así la detección debe ser independiente para ambos elementos.

## 1.2.1 METODOS TRADICIONALES PARA EL CONTROL, DETECCION Y SUPRESION DE FUEGO

Para poder realizar una detección y seguimiento del fuego es necesario tener un control sobre la zona de fuego potencial. Las técnicas de control, detección y de extinción de incendios más utilizados a lo largo de los últimos años se resumen en la Figura 3:

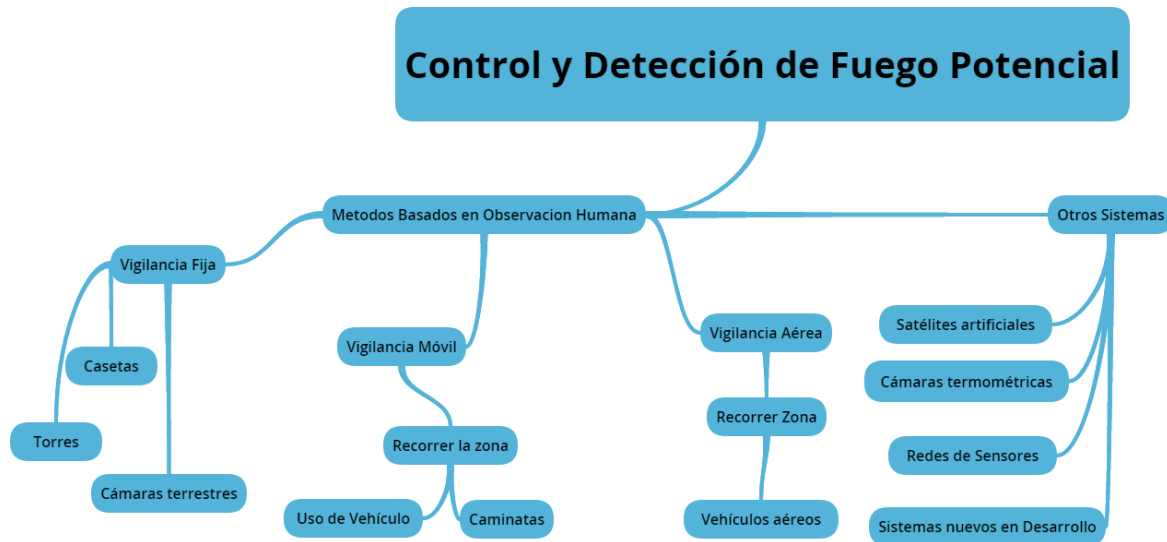


Figura 3: Métodos para el control y detección de fuego potencial en la zona de riesgo

Dentro de los sistemas tradicionales de control de zonas y detección de incendios forestales encontramos todos aquellos que son basados en la observación humana [8] [12]. Existen otros sistemas para la detección de incendios forestales basados en el uso de sensores y sistemas computacionales que se encuentran en auge y otros que se encuentran en desarrollo, algunos de ellos se los analizará en la siguiente sección.

Cabe resaltar que cualquier medio para el control de la zona debe contar con equipamiento adecuado para la comunicación con la central, mapas de la zona, sistemas de localización y registro de incidencias.

Los vehículos aéreos se han usado tradicionalmente para el control y extinción del fuego, pero el uso de vehículos aéreos para la detección no es tan común.

## 1.2.2 METODOS ACTUALES PARA LA DETECCION DE FUEGO

### *SISTEMAS BASADOS EN SATELITES*

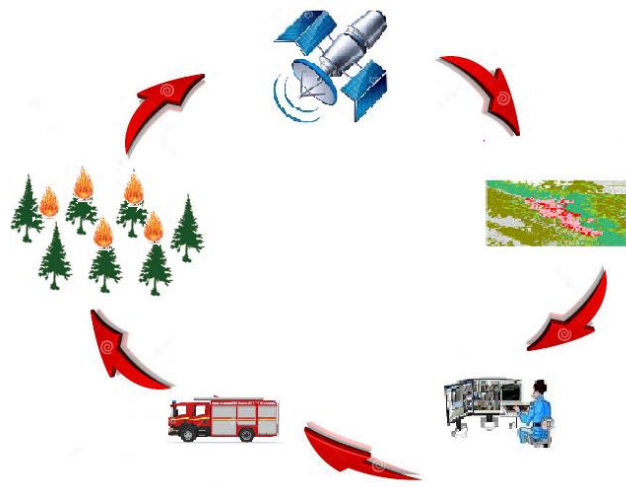
Un método para el control de incendios forestales es mediante el uso de satélites artificiales para la observación de extensas áreas de terreno. Una característica de este sistema es su elevado coste [8]. Los satélites detectan la temperatura en la superficie terrestre y cuando esta supera un valor aproximado de 38'85°C, el satélite considera que hay un fuego [14].

Satélites en órbita terrestre e incluso dispositivos de aire flotantes se han empleado para la observación y detección de incendios forestales. Las imágenes por satélite recogidas por dos satélites principales lanzados con fines de detección de incendios forestales, la muy alta resolución del radiómetro (AVHRR) [18], lanzado en 1998, y el espectro radiómetro de imágenes de resolución

moderada (MODIS), puesto en marcha en 1999 avanzaron, se han utilizado [19] [13]. Por desgracia, estos satélites pueden proporcionar imágenes de las regiones de la tierra cada dos días y que es mucho tiempo para escanear el fuego; además de la calidad de las imágenes de satélite se puede ver afectada por las condiciones climáticas [13].

Usualmente un sistema de detección de incendio basado en satélite comprende de los siguientes pasos:

1. El satélite realiza tomas (fotografías) y/o mapas de áreas quemadas o zonas del incendio en progreso [9], y realiza el envío de toda la información a una central.
2. En la central, los operadores realizan el procesamiento correspondiente de la información proporcionada por el satélite y se analiza el hecho de activar o no la alarma.
3. Como paso final, la central es la encargada de dar órdenes para que se efectúen los trabajos de control y/o extinción del fuego.



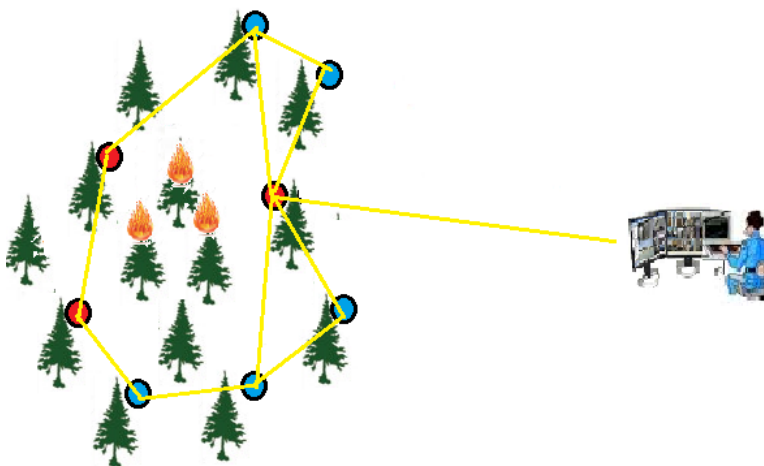
*Figura 4: Ciclo de un Sistema de Detección de Incendios Basado en Satélite*

Un resumen sobre la modalidad de trabajo de un sistema de detección por medio de satélites se puede apreciar en la Figura 4. Si bien es cierto estos sistemas tienen grandes prestaciones y con una amplia información acerca del incendio en cuestión, sin embargo, la gran desventaja de este método es su elevado costo y la dificultad de detección cuando el incendio se encuentra en etapas tempranas, sin mencionar que un satélite está diseñado para numerosas funciones (telecomunicaciones, etc.) por lo que podría no estar completamente preparado para detección de incendios forestales.

### ***USO DE REDES DE SENSORES INALÁMBRICOS***

Una nueva tecnología llamada red de sensores inalámbricos (WSN) está recibiendo más atención hoy en día y ha empezado a aplicarse en la detección de incendios forestales. Los nodos inalámbricos se integran en la misma placa de circuito impreso, los sensores, el procesamiento de datos, y el transceptor inalámbrico y todos ellos consumen energía de las mismas baterías de origen. A diferencia de los teléfonos celulares, WSN no tienen la capacidad de recarga periódicas. Estos sensores son dispositivos capaces de detectar su entorno y el cómputo (cálculo) de los datos. Los sensores detectan parámetros físicos tales como la temperatura, presión y humedad, así como los

parámetros químicos tales como monóxido de carbono, dióxido de carbono, y dióxido de nitrógeno. Los sensores funcionan de una auto-curación y entorno de red inalámbrica de auto-organización. Un tipo de tecnología inalámbrica es ZigBee que es un nuevo estándar industrial basado en IEEE 802.15.4. Esta tecnología hace hincapié en la batería de bajo coste de aplicación solar y paneles solares pequeños y es adecuado para bajas velocidades de datos y comunicaciones de gama baja. [13] [15] [16] [17]



*Figura 5: Sistemas de Detección Basados en sensores inalámbricos*

En la Figura 5 se puede apreciar un ejemplo de sistema basados en sensores inalámbricos. Este tipo de sistema consiste en montar una red de pequeños sensores inalámbricos arraigados en el entorno de preferencia interconectados en una topología estrella. Este tipo de sistemas con redes es ideal para ambientes que pasan de ser estáticos a dinámicos en poco tiempo. Esto es puesto a que en el ambiente como un bosque, la red de sensores puede permanecer inactiva por mucho tiempo, en caso de un incendio, la red deberá comenzar de manera casi instantánea numerosas actividades para poder obtener el estado neto del siniestro en cuestión [22]. De esta manera los sensores más cercanos al incendio reaccionaran y brindaran los datos correspondientes a la magnitud y trayectoria que toma el incendio.

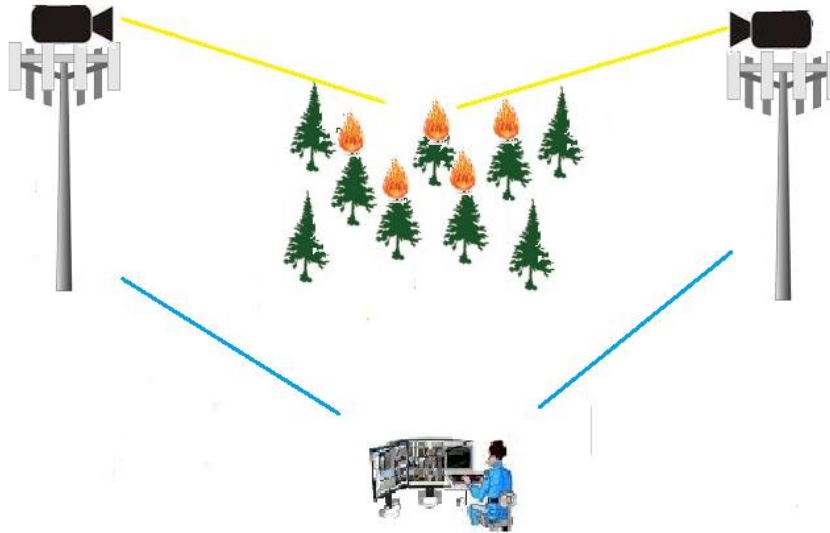
De una manera similar al anterior sistema, todos los datos adquiridos por la red de sensores son enviados hasta una pequeña central en la cual toda la información es documentada y eventualmente se toman las medidas necesarias para la contención del incendio.

### ***DETECCIÓN A PARTIR DE INFRARROJOS Y CÁMARAS ÓPTICAS***

Tiene como objetivo la vigilancia del territorio forestal y la detección automática de incendios mediante la utilización de cámaras de infrarrojos y ópticas, así como la localización exacta del incendio [14].

Una de las mayores prestaciones que tienen estos sistemas es que tienen total control sobre el terreno que posible fuego, esto se lo realiza ubicando torres de control tratando de abarcar todos los puntos de riesgo de la zona. Las cámaras (montadas sobre las torres) mandan las imágenes grabadas a la unidad de procesamiento de imágenes y es esta unidad la que detectara un posible incendio a través

de las imágenes que le han enviado. Cuando aparece una alarma de incendio automáticamente la cámara que ha detectado el incendio se detendrá en la posición de él y aparecerá una alarma sobre el monitor de supervisión, avisando a las personas encargadas de la central. Cuando la persona encargada de la central de alarmas comprueba que se trata de un verdadero incendio forestal da la alarma general [14]. Un esquema general de este sistema se puede observar en la Figura 6



*Figura 6: Sistemas de Detección basados en cámaras infrarrojas y ópticas*

Como se ha podido observar, la utilidad de este tipo de sistemas de detección mediante cámaras es alta, sin embargo, tiene grandes desventajas como ser que requieren de un montaje bastante significativo de infraestructura a lo largo de la extensión de terreno y es un sistema estático, por lo que tiene un elevado coste económico.

Adicionalmente la central de procesamiento de datos tiene que estar constantemente monitoreada puesto a que este tipo de sistemas tiene alto riesgo de alarmas falsas debidos a los ángulos a los que vigilan las cámaras.

### **1.3 COMPARACION ENTRE LOS DIFERENTES METODOS DE DETECCION DE INCENDIOS FORESTALES**

En los apartados anteriores se ha mencionado las tendencias actuales y diferentes tecnologías que se usan (o estudian) para la detección de incendios forestales, así como también un resumen de las acciones que realiza cada sistema para realizar dicha detección, sin embargo, si bien es cierto que estas tecnologías tienen grandes prestaciones, también tienen grandes limitantes y hasta el momento no existe vínculos entre ellas. **The University of SouthWales, UK** en su artículo **A Review on Forest Fire Detection Techniques** [13], realiza una comparación entre las diferentes características que debe cubrir un Sistema de detección de incendios y se muestra a continuación:



Comparación	Basado en Observación Humana	Sistema de Satélite	Cámaras Ópticas	Redes de Sensores Inalámbricos
Costo	Bajo	Muy Alto	Alto	Medio
Eficiencia y Practicidad	Bajo	Bajo	Medio	Alto
Repetición de Alarmas Erróneas	Bajo	Bajo	Medio	Medio
Precisión en localización de incendios	Bajo	Medio	Medio	Alto
Retardo en la detección	Largo	Muy Largo	Largo	Corto
Información del comportamiento del Fuego	-----	Si	-----	Si
Uso para otros propósitos	No	Si	No	Si

*Tabla 1: Tabla comparativa entre diferentes métodos de detección de incendios*

Como se puede apreciar inspeccionando la Tabla 1, ningún sistema, por más tecnología que integre, es perfecto y cubre todos los requerimientos en su totalidad para la detección de incendios forestales. Los sistemas actuales tienen un problema en común y es el tiempo de respuesta hacia la estación base. Cabe resaltar que sin importar cuán avanzado sea un sistema, este siempre será supervisado por una persona encargada en la central para validar la veracidad de las alarmas.

El sistema que se propone desarrollar es mediante cámaras ópticas y vehículos aéreos no tripulados (drones) para la detección de incendios forestales, intentando cubrir los requerimientos de costo, retardos en detección, de explorar la zona de posible fuego y precisión en la localización del fuego. Para lograr dicho objetivo se propone desarrollar un sistema inteligente que use un conjunto de algoritmos en aprendizaje automático.

## CAPÍTULO 2. EXTRACCION DE CARACTERISTICAS EN IMAGENES

Cuando los datos de entrada a un algoritmo son demasiado grandes para ser procesados y si se sospecha de una posible redundancia (muchos datos, poca información), entonces los datos de entrada deberían ser transformados a una representación reducida de un conjunto de patrones, el mismo es llamado comúnmente vector de características. El acto de transformar los datos de entrada en un conjunto de características se denomina extracción de características.

La extracción de características describe la información relevante contenida en la forma y/o secuencia de un patrón (imagen) de modo que la tarea de clasificar patrones sea más fácil gracias a un procedimiento formal. La meta principal de la extracción de características es la de obtener la información más relevante a partir de los datos originales y representar dicha información en un espacio dimensional más bajo (vector de características) que constituye una representación de la imagen. [22].

Si las características extraídas son seleccionadas cuidadosamente, se esperaría que el conjunto de características contenga la información relevante de los datos de entrada con el fin de realizar la tarea deseada usando esta representación reducida en lugar de usar el conjunto completo.



*Figura 7: Proceso típico en aplicaciones basados en clasificadores de imágenes*

Usualmente los sistemas y/o algoritmos usados para aplicaciones tales como visión por computador y clasificadores de imágenes, basan su proceso en tres grandes partes: la extracción, la selección y la clasificación, una ilustración de esto se puede observar en la Figura 7. Se ha mencionado que la extracción de características es el proceso de obtener atributos (características) para ser usadas en la selección y clasificación, por su parte, la selección reduce el número de características que se proveen para la tarea de clasificación. De las tres actividades mencionadas anteriormente, la extracción de características es la más crucial puesto que las características particulares disponibles para la

discriminación influyen directamente con la eficacia de la clasificación [25]. Dicho esto, en aplicaciones con imágenes las características se pueden clasificar en los siguientes tipos:

- **Globales:** calculadas o encontradas sobre la imagen completa.
- **Locales:** obtenidas por subdivisión de bandas o detección de bordes
- **Nivel de píxeles:** encontradas en cada pixel como la tonalidad
- **Específicas de cada dominio:** Características tales como caras humanas, huellas digitales, rasgos marcados, etc [25].

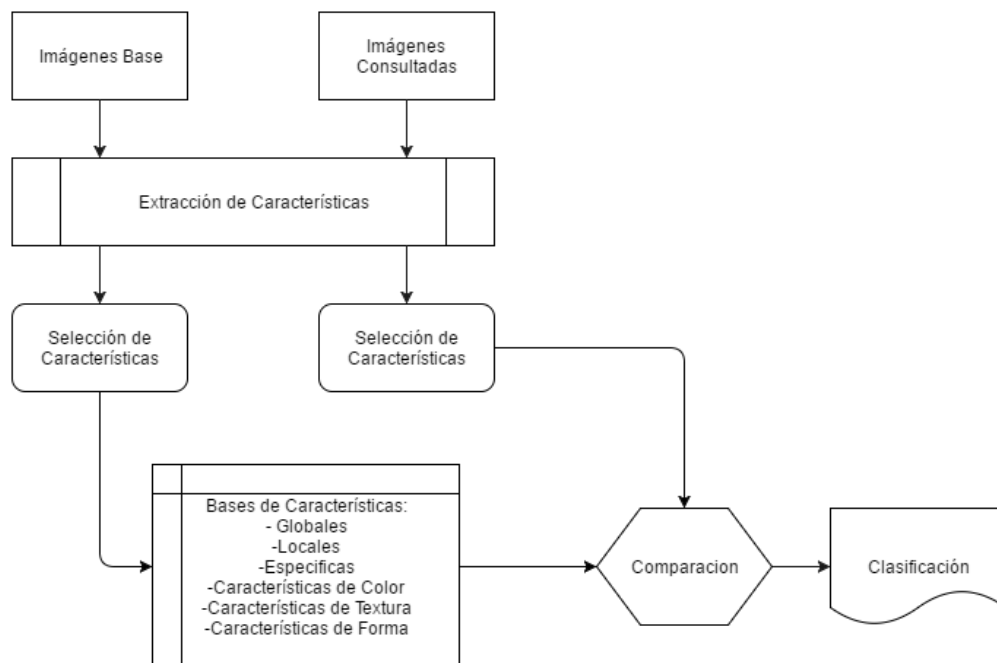


Figura 8: Secuencia típica para la clasificación de una imagen

Las características que se pueden extraer de una imagen son numerosas y dependerán siempre de la aplicación que tenga, no obstante, todas los tipos de características y muchos otros más se pueden extraer a través de características generales que son independientes de la aplicación tales como el color, la textura y la forma, estas características de una imagen usualmente toman el nombre de descriptores de información general o descriptores visuales [30] [38]. Existen varias técnicas de extracción de características basadas en la extracción independiente y/o aislamiento de cada uno de los descriptores mencionados anteriormente de una misma imagen. De la necesidad de aislar un atributo o patrón específico nacen la mayoría de las técnicas usadas actualmente para la extracción de características en imágenes.

A continuación, se expondrán detalles de la información que se puede obtener de estas características generales de las imágenes continuando con la explicación de técnicas matemáticas para la extracción de características más usadas actualmente como son los momentos de Hu y las transformadas wavelets.

## 2.1 TECNICAS DE EXTRACCION TRADICIONALES DE CARACTERISTICAS

### 2.1.1 CARACTERISTICAS DE COLOR

La característica del color es una de las características visuales más utilizadas en aplicaciones como recuperación de imágenes y clasificación [26] [30] [53] [54] por su robustez, la facilidad de cómputo e implementación, entre otros. Típicamente, el color de una imagen es representado a través de un modelo que describe información de color. Un modelo de color comúnmente es representado por un sistema coordinado en 3D y un subespacio que en conjunto con el sistema representan cada color por un punto único [25].

El color es percibido por los humanos como una combinación de 3 estímulos de intensidad conocido como Síntesis aditiva de color [55]: Rojo, Verde, Azul, las cuales forman un espacio de color. Actualmente existen diferentes tipos de modelos de color y espacios de color como son el CIE XYZ, Hunter Lab, CIELAB [56] [57] [58], entre otros, los cuales usan parámetros tales como longitudes de onda, luminosidad y gradientes para poder generar sus espacios de color, sin embargo, los modelos aditivos de color más usados en medios que reproducen y/o capturan imágenes como cámaras, computadores, etc., son el RGB/CMY y el HSL/HSV que son modelos de colores primarios y de propiedades del color.

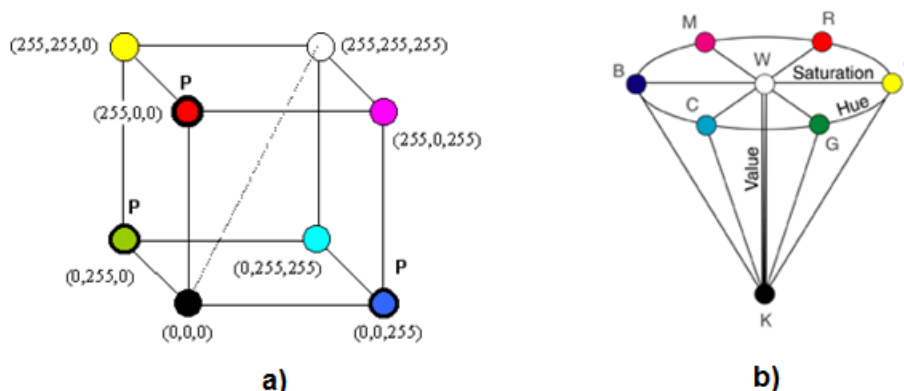


Figura 9: Espacios de Color: a) Espacio de Color RGB. b) Espacio de Color HSV

En la Figura 9.a Se muestra el espacio RGB, los colores con una P son colores primarios, la línea discontinua indica donde encuentra los grises y va desde la posición (0,0,0) hasta la (255,255,255) que es el color blanco. Por su parte en la Figura 9.b Se muestra el espacio HSV, se puede apreciar el espacio de color, la saturación varia en un rango de 0 a 1, los rangos de matiz van desde 0 hasta 360 grados empezando en el rojo hasta el magenta y volviendo al rojo. El espacio de color HSV busca representar las relaciones entre los colores y mejorar el modelo de color RGB. Los parámetros de este modelo son:

- **Matiz (H):** hace referencia a la frecuencia dominante del color en el espectro visible.
- **Saturación (S):** se refiere a cantidad de color o a la pureza de este.

- **Valor (V):** Es la intensidad de luz de un color (cantidad de blanco o negro).

Para poder utilizar un modelo de color adecuado en aplicaciones específicas, es necesaria una conversión entre modelos de color [28]. A continuación se muestran fórmulas para convertir el modelo de color RGB al HSV:

$$H = \cos^{-1} \left( \frac{\frac{1}{2}[R - G] + [R - B]}{\sqrt{(R - G)^2 + (R - B)(G - B)}} \right) \quad (1)$$

$$S = 1 - \frac{3}{R + G + B} [\min(R, G, B)] \quad (2)$$

$$V = \frac{1}{3}(R + G + B) \quad (3)$$

Indistintamente del modelo, el objetivo principal de analizar el color es encontrar información relevante dentro de las imágenes y que facilite la tarea de realizar una clasificación. Por ello se expondrán los métodos más relevantes para obtener información a partir del color.

### **DESCRIPTORES DE COLOR**

Como se ha mencionado previamente, una de las ventajas de las técnicas que usan características de color es la facilidad de cómputo. Las herramientas digitales con las que se cuentan actualmente (computadores, cámaras, reproductores de video, visualizadores de imágenes, etc.) Utilizan diferentes tipos de estándares que cuentan con descriptores de color. Computacionalmente se pueden obtener diferentes descriptores de color que se encuentran inmersos dentro de los estándares de compresión de imágenes y videos (JPEG, MPEG, MPEG-7, etc) que brindan cierta información dependiendo de que se busca. Algunos de los descriptores de color más usados [30] en conjunto con la información que brindan se muestran a continuación en la siguiente tabla:

<b>Descriptor de Color</b>	<b>Información obtenida</b>
Descriptor de color Dominante (DCD)	Da una descripción de los colores representativos de una imagen o región de imagen.
Descriptor de Color Escalable (DCE)	Codificado por transformación Haar (1D wavelet). Provee características de color medidas sobre una imagen completa.
Descriptor de la Estructura del Color (DEC)	Captura algunas características espaciales de la distribución de color en una imagen. En cada posición se determina el color de cada pixel cubierto por el elemento estructurante.
Descriptor de la Distribución del Color (DDC)	Captura la disposición espacial de los colores más representativos de una rejilla superpuesta en una región o toda la imagen.
Grupo de Marco o de fotos (Group of frame GoF)	Una extensión del descriptor de color escalable que define una estructura para representar

	características de color de una colección similar de marcos o fotogramas de video.
--	--

Tabla 2: Información de los Descriptores de Color

Para poder brindar dicha información los descriptores se valen de herramientas matemáticas como son la transformada Haar (Wavelet 1D) o la transformada del coseno en combinación de métodos de conteo de pixeles que convergen en la selección del color más representativo y la disposición espacial que este tiene dentro de una de la imagen o de una región de la misma.

### HISTOGRAMAS Y MOMENTOS DE COLOR

Se ha mencionado la robustez que presentan las características de color, dicha robustez viene dada por la representación mediante histogramas de color. Un histograma es una representación gráfica de la frecuencia de una distribución de datos numéricos, es una estimación de la distribución de probabilidad de una o más variables continuas o discretas [32]. Dicha propiedad de robustez que mantienen los histogramas de color vienen dados por los momentos invariantes y momentos centrales [32] [33].

Por lo expuesto anteriormente, los momentos de color encuentran sus bases en funciones de distribuciones de probabilidad al igual que los histogramas, por lo que estos momentos de color son invariantes a la rotación, traslación y escalamiento [32] [33] [34]. Dado a que dichos momentos codifican tanto información de color como de forma, son buenas características para usar bajo condiciones de luz cambiante pero no pueden manejar la oclusión de manera satisfactoria [34]. Además, los momentos de color pueden ser computados por cualquier modelo de color y 3 momentos son calculados por cada 3 canales de color, por lo que una imagen puede ser caracterizada por 9 momentos de color. El cálculo de los momentos de color se lo realiza de la misma manera que los momentos de una distribución de probabilidad [34].

A continuación se presenta un cuadro resumen de los momentos de color por cada canal [29] [34]:

Momento	Definición/Formula
Media	$E_i = \sum_{j=1}^N \frac{1}{N} p_{ij}$ <p>La media puede ser entendida como el valor promedio del color en la imagen. Donde N es el número de pixeles en la imagen, P<sub>ij</sub> es el valor del j-ésimo pixel en la imagen en el i-ésimo canal de color.</p>
Desviación Estándar	$\sigma_i = \sqrt{\left( \frac{1}{N} \sum_{j=1}^N (p_{ij} - E_i)^2 \right)}$ <p>La desviación estándar es la raíz cuadrada de la varianza de la distribución de color. Donde E<sub>i</sub> es el valor de la media o primer momento de color para el i-ésimo canal de color de la imagen.</p>
	$S_i = \sqrt[3]{\left( \frac{1}{N} \sum_{j=1}^N (p_{ij} - E_i)^3 \right)}$

Oblicuidad	La oblicuidad puede ser entendida como una medida del grado de asimetría en la distribución de color. Además nos brinda información acerca de la forma de la distribución de color.
Kurtosis	$k = \frac{N \sum_{i=1}^N (x_i - \mu)^4}{\left[ \sum_{i=1}^N (x_i - \mu)^2 \right]^2}$ <p>Es el cuarto momento de color y de manera similar a la oblicuidad, proporciona información acerca de la forma de la distribución de color. Específicamente, es una medida de cuan plana o alta es la distribución en comparación con la distribución normal.</p>
Índices de momentos de color	$d_{mom}(H, I) = \sum_{i=1}^r W_{I1}  E_i^1 - E_i^2  + W_{I2}  \sigma_i^1 - \sigma_i^2  + W_{I3}  S_i^1 - S_i^2 $ <p>Donde:  (H,I): son 2 distribuciones de imagen siendo comparadas  I: índice del canal actual  R: número de canales  E<sub>i</sub>: primeros momentos (media) de las 2 distribuciones de imagen  σ<sub>i</sub>: Segundos momentos (Desviación estándar)  S<sub>i</sub>: tercer momento (oblicuidad)  W<sub>i</sub>: Pesos de cada momento especificados por el usuario para cada momento de color usado.</p>

Tabla 3: Momentos de Color e Indexación

Dados los diferentes momentos de color usados para obtener las características de color de cada pixel y agruparlos por sus similares a más de las propiedades intrínsecas de los momentos, se puede apreciar la robustez que presenta esta técnica de extracción de características por color y las ventajas que representan a la hora de realizar una clasificación por color. Un histograma de color no es más que una representación gráfica del número de pixeles en una imagen [26] [27] [34]. Para ilustrar esto se puede ver la Figura 10.

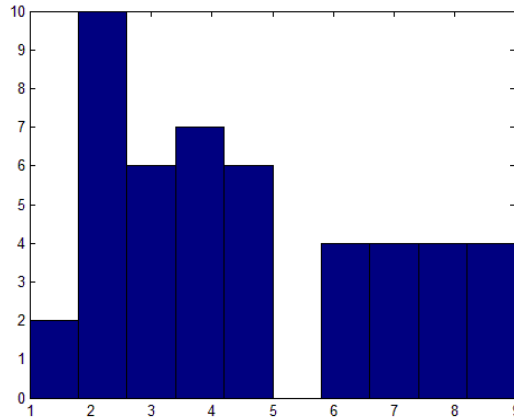


Figura 10: Ejemplo de representación de un histograma

### 2.1.2 CARACTERÍSTICAS DE TEXTURA

Otra propiedad importante de las imágenes son las características de textura. La textura dentro de las imágenes son atributos útiles por lo que son usados como de descriptores de región [38] [40] [59]; observan la homogeneidad y los histogramas de bordes de dichas regiones ayudan en la segmentación o clasificación de una imagen. En algunas imágenes, la textura puede ser la característica definitoria de las regiones y una referencia crítica en la obtención de un análisis de patrones adecuado [38] [59].

La textura nos da información sobre la disposición espacial de los colores o intensidades en una imagen [35]. El análisis de la textura empieza cuando las características de color no son suficientes para clasificar o segmentar imágenes, por ejemplo, se puede suponer que un histograma de color nos indica 50% de color rojo y 50% de color azul, muchos patrones y mosaicos en imágenes pueden brindar una distribución de intensidad de esta manera pero al mismo tiempo pueden estar contenidas en texturas diferentes y variadas como se puede apreciar en la Figura 11, las imágenes mantienen la misma distribución de intensidad (color) pero tienen diferentes patrones geométricos.

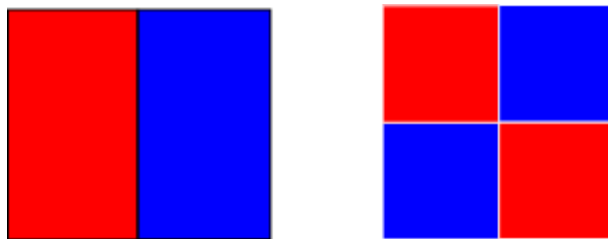


Figura 11: Texturas diferentes con la misma distribución de intensidad (color)

Para el caso de la Figura 11, pese a que mantienen diferentes texturas, resulta sencilla la segmentación de estas regiones de un solo color y reconocer patrones simples, no obstante, en imágenes que contienen texturas naturales (fotografías de paisajes, etc.), describir un arreglo espacial puede ser más dificultoso puesto a que dichos arreglos no son regulares, sin embargo, se podría argumentar que en cada imagen existe un arreglo notable.

La textura puede definirse de diferentes maneras, se presentan 2 enfoques principales y son:



- **Enfoque Estructural (espaciales):** Define a la textura como un conjunto de texeles (texture element) en alguna relación regular o repetida.
- **Enfoque Estadístico (espectrales):** Define a la textura como una medida cuantitativa de un arreglo de intensidades en una región.

La unidad fundamental de un espacio de textura es el texel o elemento de textura, utilizados en los gráficos por computador. Las texturas están representadas por matrices de texeles, al igual que las imágenes se representan por matrices de píxeles. Los texeles también pueden ser descritos por áreas de imagen que se obtienen a través de procedimientos como umbralización (thresholding) [35].

En general, el color es usualmente una propiedad de pixel mientras que la textura puede ser medida únicamente a partir de un grupo de píxeles [35] [38] [40]. Basados en el dominio del cual la característica de textura es extraída, pueden ser clasificados en general en métodos de extracción de atributos de textura espaciales y espectrales [36]. En este contexto, las características de textura son extraídas mediante el cálculo de las estadísticas de píxeles o encontrando las estructuras de píxeles locales en el dominio original de la imagen, mientras que el último transforma una imagen en el dominio de la frecuencia y luego calcula la característica de la imagen transformada [36].

Para la mayor parte de los casos, segmentar los texeles es más difícil en imágenes reales que en imágenes producidas o con patrones generados manualmente, para esto, una medida de textura cuantitativa o estadística puede ser calculada a partir de imágenes en escala de grises o a color, este enfoque puede ser implementado computacionalmente y funciona tanto para clasificar como segmentar texturas [35] [36]. A continuación se expondrán algunas técnicas/métodos que realizan medidas cuantitativas (estadísticas) de texturas.

#### **MATRICES Y CARACTERISTICAS DE CO-OCURRENCIA**

Una matriz de co-ocurrencia es un arreglo bidimensional en el cual tanto filas como columnas representan un conjunto de posibles valores de imagen [35]. Las matrices de co-ocurrencia capturan las propiedades de la textura, sin embargo, no son utilizadas directamente para análisis posteriores, en su lugar, características numéricas calculadas a partir de la matriz de co-ocurrencia pueden ser usadas para representar a la textura de una manera más compacta. Un resumen de las características derivadas de la matriz de co-ocurrencia se presenta a continuación [25]:

<b>Características Estándar</b>	<b>Formula</b>
Energía	$\sum_i \sum_j C(i, j)^2$
Inercia o Contraste	$\sum_i \sum_j C(i, j)^2 C(i, j)$
Correlación	$\frac{\sum_i \sum_j (ij) C(i, j) - \mu_i \mu_j}{\sigma_i \sigma_j}$
Momento Diferencia/Homogeneidad	$\sum_i \sum_j \frac{1}{1 + (i + j)^2} C(i, j)$
Entropía	$\sum_i \sum_j C(i, j) \log C(i, j)$

Tabla 4: Resumen 5 características estándar de la matriz de co-ocurrencia

Donde  $\mu_i \mu_j$  son las medias y  $\sigma_i \sigma_j$  son la desviación estándar de la fila y columna.

### FILTROS DE GABOR

Un filtro Gabor es obtenido por la modulación de una senoide con una señal Gaussiana (kernel) [36] [37]. Los filtros Gabor están diseñados para muestrear por completo el dominio de frecuencia de una imagen mediante la caracterización de la frecuencia centro y la orientación de parámetros. La imagen es filtrada con un banco de filtros Gabor u ondeletas Gabor (Gabor Wavelets) de frecuencias y orientaciones espaciales diferentes preferidas [37]. Cada wavelet captura la energía que existe en cada frecuencia y dirección específica lo cual proporciona una frecuencia localizada representada como un vector de características [36]. Por consiguiente, los atributos de textura pueden ser extraídos de este grupo de distribuciones de energía [36] [40].

Las Magnitudes de las respuestas de los filtros de Gabor son representados por 3 momentos [25]:

Momento	Formula
Media	$\mu(W, \theta, \sigma_x, \sigma_y) = \frac{1}{XY} \sum_{x=1}^X \sum_{y=1}^Y FGab(x, y, W, \theta, \sigma_x, \sigma_y)$
Std	$std(W, \theta, \sigma_x, \sigma_y) = \sqrt{\sum_{x=1}^X \sum_{y=1}^Y  FGab(x, y, W, \theta, \sigma_x, \sigma_y) - \mu(W, \theta, \sigma_x, \sigma_y) ^2}$
Skew	$Skew = \frac{1}{XY} * \sum_{x=1}^X \sum_{y=1}^Y \left( \frac{FGab(x, y, W, \theta, \sigma_x, \sigma_y) - \mu(W, \theta, \sigma_x, \sigma_y)}{std(W, \theta, \sigma_x, \sigma_y)} \right)^3$

Tabla 5: Momentos de los filtros de Gabor

El vector de características es construido usando  $\mu(W, \theta, \sigma_x, \sigma_y)$ ,  $std(W, \theta, \sigma_x, \sigma_y)$  y Oblicuidad (Skew) como los componentes de atributos.

Los filtros de Gabor difieren de técnicas como Sobel y Canny por el tipo de kernel que usan que es específico en cada caso y con el cual se consiguen diferentes resultados (características) [40] [41] [42]. Ambos son técnicas de detección de bordes.

### DESCRIPTORES DE TEXTURA

Existen diferentes técnicas para realizar la extracción de características tanto espaciales como espectrales, dependiendo de las características que se desean extraer se usara uno u otro dominio. Gracias a su facilidad de cómputo [59], los atributos de textura pueden ser calculados/representados de manera sencilla por herramientas digitales, muchas de estas al igual que los descriptores de color tienen sus propios descriptores de textura que comúnmente se usan en reproductores, visualizadores o en el protocolo de compresión de imágenes (MPEG-7, etc) [59]. Los descriptores de textura más usados son [38] [59]:

Descriptor	Descripción
Texture Browsing Descriptor (TBD)	Descriptor compacto que requiere únicamente 12 bits para caracterizar regularidades en textura, direccionalidad y Tosquedad.
Homogeneous Texture Descriptor (HTD)	Provee una caracterización cuantitativa de textura similar a la comparación de imagen a imagen. Este descriptor es calculado primero filtrando la imagen con un banco de filtros

	sensitivos a la orientación y escala, calculando la media y la desviación de las salidas filtradas en el dominio de frecuencia.
Edge histogram Descriptor (EHD)	Captura la distribución espacial de los bordes. Una imagen dada es primero sub dividida en sub imágenes de 4x4 y es calculado un histograma de borde locales para cada sub imagen. Los bordes se agrupan en categorías y la imagen continua subdividiéndose hasta que el descriptor resultante alcanza un tamaño de 240 bits.

Tabla 6: Descriptores de textura

### 2.1.3 CARACTERISTICAS DE FORMA

Como la última de las características principales de las imágenes tenemos los atributos de forma. Un objeto puede ser definido por su textura o forma. La forma de un objeto es una de las características visuales más importantes de una imagen, la cual ayuda en una detección y clasificación automática de objetos de toda clase del mundo real como lo hacemos las personas [23]. La característica de forma posee una valiosa información semántica, sin embargo, dicha información solo puede ser extraída mediante una segmentación similar a la que realiza el sistema visual humano. Actualmente existen únicamente aproximaciones que describen regiones y contornos dentro de las imágenes de una manera muy eficiente.

El propósito general de cualquier aproximación de forma consiste en codificar formas geométricas simples como líneas rectas en diferentes direcciones o círculos esparcidos a lo largo de un plano, un ejemplo de esto es la transformada Hough [43]. En general, las técnicas de extracción de características de forma pueden ser clasificadas en 2 grandes grupos, métodos basados en contornos y regiones [36] [41]. Los primeros métodos calculan los atributos de forma solo a partir de los límites de la forma, mientras que los últimos métodos extraen características de una región completa [36].

En la actualidad existen numerosos métodos para extraer características de forma dentro de las imágenes, algunos de estos métodos son basados en los ya conocidos momentos invariantes que encuentran su origen en la probabilidad y estadística y los momentos centrales. Otros métodos se encuentran basados en aproximaciones poligonales o en funciones unidimensionales para la representación de formas. Así también existen métodos espacio-escala, atributos de interrelaciones espaciales y transformadas de forma en dominios. Por esta razón, se revisaran únicamente los métodos más significativos y los más usados en cada caso. De una manera alternativa, existen revisiones profundas sobre la extracción de características de forma, **M. Yang**, [42] en su artículo “**A survey of shape feature extraction techniques**”, **Pattern Recognition**, presenta una recopilación sobre las aproximaciones existentes de extracción de características basadas en forma.

#### **MOMENTOS INVARIANTES DE HU**

Los momentos invariantes fueron introducidos por primera vez en el año de 1962 por Hu [39] quien derivó 6 invariantes ortogonales absolutos y una posición oblicua invariante ortogonal basado en invariantes algebraicas los cuales no solo son independientes de la posición, tamaño y orientación

sino que también son independientes a la proyección paralela [39]. Los momentos invariantes han demostrado ser medidas adecuadas para rastrear los patrones de imágenes en relación con la traslación, escalamiento y rotación de imágenes bajo la asunción de que dichas imágenes mantienen funciones continuas y son libres de ruido [25] [26]

Basados en momentos centrales normalizados mostrados en [39], Hu presento 7 momentos invariantes:

$$\begin{aligned}\emptyset_1 &= n_{20} + n_{02} \\ \emptyset_2 &= (n_{20} - n_{02})^2 + 4n_{11}^2 \\ \emptyset_3 &= (n_{30} - 3n_{12})^2 + (3n_{21} - \mu_{03})^2 \\ \emptyset_4 &= (n_{30} + n_{12})^2 + (n_{21} + \mu_{03})^2 \\ \emptyset_5 &= (n_{30} - 3n_{12})(n_{30} + n_{12})[(n_{30} + n_{12})^2 - 3(n_{21} + n_{03})^2] \\ &\quad + (3n_{21} - n_{03})(n_{21} - n_{03})[3(n_{30} + n_{12})^2 - (n_{21} - n_{03})^2] \\ \emptyset_6 &= (n_{20} - n_{02})[(n_{30} + n_{12})^2 - (n_{21} - n_{03})^2] + 4n_{11}(n_{30} + n_{12})(n_{21} - n_{03}) \\ \emptyset_7 &= (3n_{21} - n_{03})(n_{30} + n_{12})[(n_{30} + n_{12})^2 - 3(n_{21} + n_{03})^2] \\ &\quad - (n_{30} - 3n_{12})(n_{21} + n_{03})[3(n_{30} + n_{12})^2 - (n_{21} + n_{03})^2]\end{aligned}$$

Los 7 momentos de Hu son propiedades útiles siendo invariantes bajo escalamiento, traslación y rotación de imágenes. Además de los momentos de Hu, existen momentos muy conocidos como son momentos geométricos, momentos de zernike, momentos rotacionales y momentos complejos [25] [26], cada uno de ellos nos brindan atributos de forma con rasgos diferentes, los cuales pueden ser útiles de acuerdo a la aplicación [25] [26] [44] [60]. Todos estos son métodos para extracción de características de forma basados en los conocidos momentos invariantes centrales.

### **TRANSFORMADA WAVELET DISCRETA (DWT)**

La transformada wavelet es una técnica matemática que descompone una señal en una serie de pequeñas funciones base denominadas ondeletas o wavelets [24] [61] [62]. Esta herramienta permite el análisis multirresolución de una imagen y su análisis se encuentra situado tanto en el dominio del tiempo como de la frecuencia [61] [62]. Para llevar a cabo una representación en tiempo-frecuencia con la DWT se necesitan usar técnicas de filtrado digital [63], como resultado de la transformada wavelet, la imagen se descompone en componentes de frecuencia altas y bajas. El análisis multirresolución se puede realizar haciendo pasar la imagen original en un primer instante a través de los filtros de descomposición paso alto y paso bajo para generar cuatro componentes de menor resolución [24]. La información contenida dentro de estas sub imágenes que corresponde a las direcciones horizontal, vertical y diagonal, implica una característica única de una imagen [61] [62] [63].

## **2.2 MODELOS DE EXTRACCIÓN DE ATRIBUTOS AUTOMÁTICOS**

### **2.2.1 AUTOCODIFICADORES**

Un autocodificador es una red neuronal de tres capas (Figura 12), la cual es entrenada para codificar las entradas  $x$  en una representación  $c$ , usando un algoritmo no supervisado con retropropagación de error. Este tipo de red reconstruye la entrada  $x$ , a su salida  $x'$  usando la función  $c$ , con el fin de obtener datos similares a los de la entrada [45]. Donde;

$$c = f(Wx + b) \quad (4)$$

$$x' = f(W'c + b') \quad (5)$$

La ecuación (4) representa la etapa de codificación de la red y la ecuación (5) la etapa de decodificación, siendo  $f$  es una función de activación no lineal.

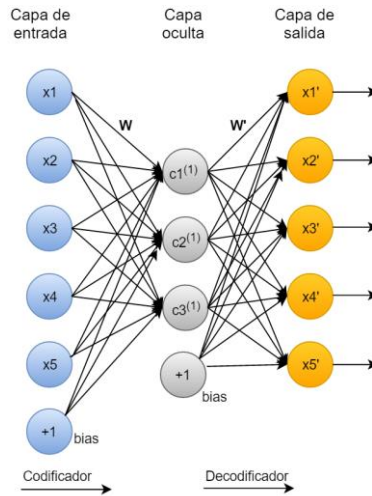


Figura 12: Estructura de un autocodificador

Esta red permite reducir la dimensión de los datos de entrada, consiguiendo así una extracción de atributos relevantes, teniendo en cuenta que el número de neuronas para la capa oculta, es menor a la capa de entrada. Existe una variación de esta red conocida como autocodificador disperso.

### 2.2.2 AUTOCODIFICADOR DISPERSO

Como se vio anteriormente un autocodificador es un tipo de red neuronal no supervisada, cuyo número de neuronas en la capa oculta es menor a la capa de entrada. En un autocodificador disperso el número de neuronas en la capa oculta es superior a las neuronas de entrada; la activación de las neuronas ocultas se realiza de forma dispersa como se ve en la Figura 13, también existen métodos como la regularización o el ingreso de ruido en la entrada, evitando que la red se limite aprender la función identidad, de esta forma la red puede aprender [45].

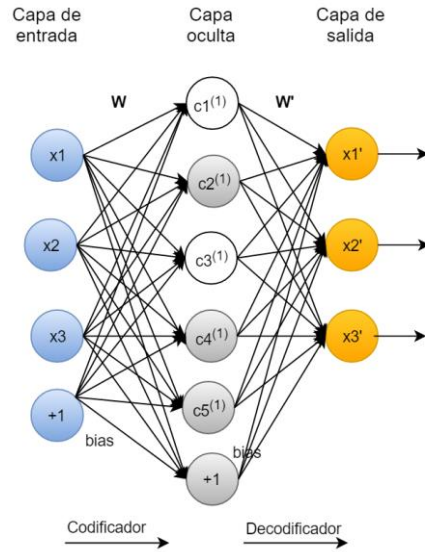


Figura 13: Autocodificador Disperso

Existen diferentes métodos como los presentados en [45] y [46] que permiten integrar un factor de dispersión en el entrenamiento, el cual se encarga de limitar la activación de las neuronas ocultas.

### 2.2.3 REDES CONVOLUCIONALES

Las redes convolucionales (CNN, por sus siglas en inglés) fueron inspiradas por el estudio de procesos biológicos, siendo estas, una variación del perceptron multicapa (MLP, por sus siglas en inglés), cuyo enfoque se centra en tareas para el reconocimiento de patrones en imágenes; reconocimiento de rostros, números manuscritos, etc [47].

La primera parte de una CNN, utiliza un filtro convolucional en la imagen de entrada para obtener un mapa de características, si se aplican múltiples filtros de convolución se generará múltiples mapas de características las cuales se van apilando uno tras otro (véase Figura 14).

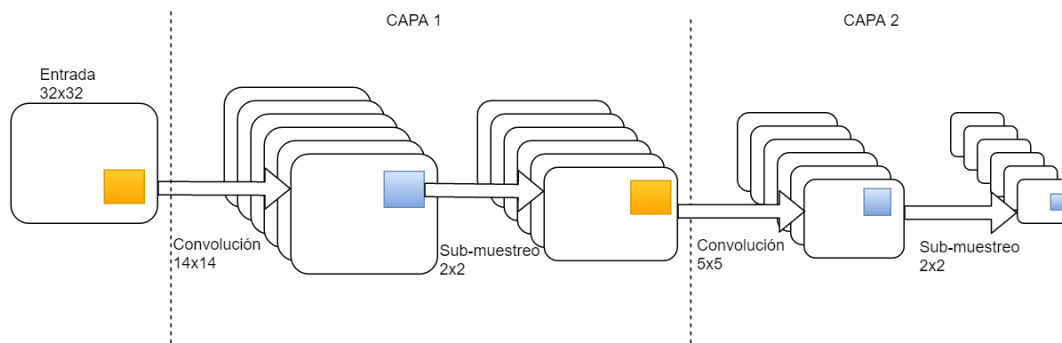


Figura 14: Arquitectura de una red neuronal convolucional (CNN).

La segunda etapa, es la implementación de una capa de sub-muestreo (subsampling, en inglés), con el fin de lograr la invarianza espacial y la reducción de la dimensión del mapa de características. Las operaciones de sub-muestreo puede ser mediante max-pooling [48]. El proceso de convolución y sub-muestreo se lo puede repetir para una segunda capa aumentando así la profundidad de la red.

## 2.2.4 MAQUINA RESTRINGIDA DE BOLTZMANN

Una Máquina Restringida de Boltzmann (RBM, por sus siglas en inglés) es un tipo de red neuronal estocástica, basadas en los modelos de energía [45], que han permitido aplicaciones como, la reducción de la dimensionalidad [49], o la clasificación de los patrones de entrada [50].

La distribución de probabilidad para esta función de energía es presentada en [51], siendo esta:

$$P(v, h) = \frac{1}{Z} e^{-E(v, h)} \quad (6)$$

Y la función de energía es conseguida mediante:

$$E(v, h) = -b^T - v - c^T h - v^T W h \quad (7)$$

Siendo Z la constante de normalización conocida también como función de partida:

$$Z = \sum_v \sum_h e^{-E(v, h)} \quad (8)$$

Esta red se compone de m unidades visibles  $v = \{v_1, v_2, v_3 \dots v_m\}$  para representar los datos observables y unidades ocultas  $h = \{h_1, h_2, h_3 \dots h_m\}$  para capturar las dependencias entre las variables observadas. Las neuronas en esta red son estocásticas binarias, y la conexión se realiza entre la capa de entrada y la oculta, pero nunca entre neuronas de la misma capa (véase Figura 15) permitiendo el uso de algoritmos eficientes como el “Contrastive Divergence”[52][45].

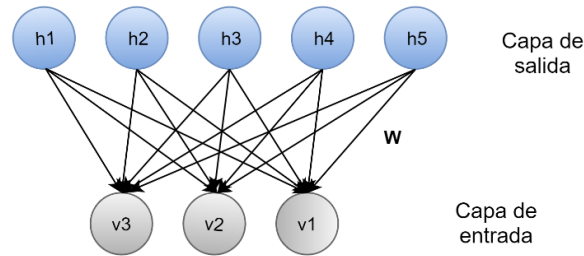


Figura 15: Arquitectura de Máquina restringida de Boltzmann.

## CAPÍTULO 3. METODOLOGIA PROPUESTA

Con base en los diferentes medios y/o métodos para la extracción de atributos en imágenes se optó por utilizar un método de extracción automática de características; el uso de redes convolucionales profundas; puesto a que en la literatura tienen un alto rendimiento en la clasificación de imágenes [64] [66] [68] [71] [72] [73].

En la metodología tradicional, si se desea entrenar una red neuronal para la clasificación de imágenes; es necesario realizar primero una extracción y selección de todos los parámetros y/o atributos relevantes en las imágenes para posteriormente realizar la clasificación deseada. Un ejemplo de la metodología clásica de un sistema de clasificación mediante imágenes se puede apreciar en la siguiente figura:

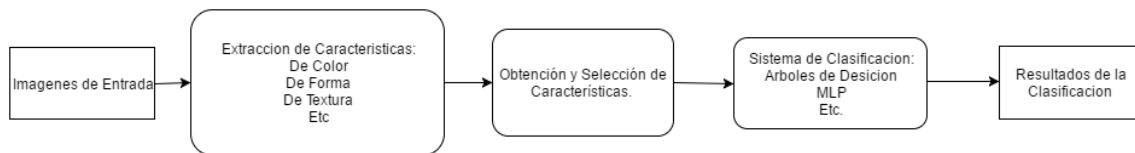


Figura 16: Metodología Clásica de un sistema de clasificación mediante imágenes

Esta metodología, para nuestro caso de aplicación específico, tiene dos claras desventajas: La primera data del hecho del preprocesamiento que necesitan los datos de entrada al sistema, en este contexto, cuando hablamos del preprocesamiento de los datos, nos referimos a todo el trabajo relacionado con la adecuación de las imágenes para la base de datos, la extracción de características y la selección de las mismas previo al acto de entrenar una red y clasificar imágenes puesto a que repercute tanto en requerimientos computacionales como en tiempos de trabajo, entrenamiento, ejecución y pruebas.

La segunda desventaja va directamente relacionada con el fenómeno físico que se desea clasificar y/o detectar mediante imágenes; Incendios Forestales. Como se ha mencionado en el capítulo 1, los incendios forestales tienen diferentes formas de presentarse en el medio ambiente y en sus primeras etapas no existe una visión clara de factores como el humo y el fuego. Dicho esto, en qué estado se encuentra el fuego y el humo dentro de una imagen depende directamente del momento y del estado del incendio al momento en que la imagen fue tomada y si analizamos diferentes imágenes tomadas en diferentes etapas de un incendio se puede notar que tienen diferentes formas, patrones, tendencias, etc. Sin mencionar factores como el escenario u hora del día. Por este motivo, la extracción y aún más la selección de características ideales para realizar un entrenamiento a una red neuronal se vuelve una tarea extenuante y poco generalizada en diferentes ambientes para la detección de incendios.

Por esta razón, se optó por utilizar una red convolucional, debido a la facilidad del trabajo con imágenes y la extracción automática de atributos en las mismas. A continuación, se muestra el modelo de la red convolucional utilizado en este caso de aplicación:



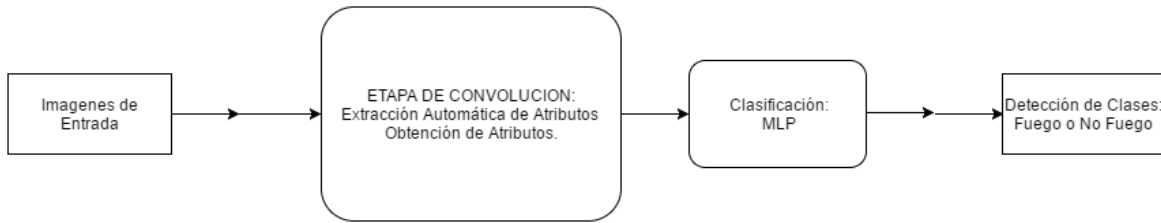


Figura 17: Metodología Con Redes Neuronales Convolucionales (CNN)

Como se puede apreciar en la Figura 17, las imágenes de entrada ingresan a un sistema completo de aprendizaje en el cual se realiza la extracción automática de atributos, los mismos que son aprendidos por una red neuronal y se realiza la clasificación para la detección de incendios (Presencia o Ausencia de Fuego). La principal ventaja sobre la metodología clásica es que se ven reducidos los procesos de extracción y selección de características lo cual disminuye en gran medida el trabajo necesario previo al entrenamiento de las redes neuronales.

A continuación, se explicarán diferentes detalles del proceso del entrenamiento y pruebas del sistema de detección de incendios al igual que se indicarán detalles de la base de datos y de la red convolucional.

### 3.1 ARQUITECTURA RED CONVOLUCIONAL

Las redes neuronales convolucionales son redes neuronales que usan convolución en lugar de multiplicaciones generales de matrices sus capas. La operación de convolución se define como sigue:

$$s(t) = \int x(a)w(t - a)da \quad (9)$$

Y se denota con un asterisco como se muestra a continuación:

$$s(t) = (x * w)(t) \quad (10)$$

Dentro de la terminología de las redes neuronales convolucionales, el primer argumento de las ecuaciones anteriores ( $x$ ) para la convolución es referido usualmente como la “entrada” mientras que el segundo argumento de las ecuaciones (9) y (10) se lo conoce como el “kernel” ( $w$ ) (o núcleo en español). La salida a la convolución ( $s$ ) se denomina muchas veces como mapa de características (feature map) [64].

De una manera más práctica para el uso de la convolución en señales discretas, se puede definir una convolución discreta como sigue:

$$s(t) = (x * w)(t) = \sum_{a=-\infty}^{\infty} x(a)w(t - a) \quad (11)$$

Dentro de los diferentes tipos de aplicaciones que usan el proceso de convolución, la entrada puede ser un arreglo unidimensional o multidimensional de datos y el kernel, de igual manera, puede ser un arreglo unidimensional o multidimensional de parámetros. Para el caso de aplicaciones de convolución usando imágenes; comúnmente se utilizan convoluciones sobre más de un eje al mismo

tiempo. Dicho esto, en una operación de convolución usando imágenes, tanto la entrada como el kernel serían bidimensionales [64], obteniendo la siguiente expresión:

$$s(i, j) = (I * K)(i, j) = \sum_m \sum_n I(i - m, j - n)K(m, n) \quad (12)$$

Donde  $I$  es la imagen de entrada y  $K$  es el Kernel (núcleo).

En las redes neuronales convencionales; cada capa oculta está compuesta por un arreglo de neuronas, en donde cada neurona se encuentra conectada con todas las neuronas de la capa anterior y dichas conexiones tienen asociadas un valor o número al cual se denomina peso. Este tipo de redes neuronales aceptan un solo vector en la capa de entrada [73]. Dicho esto, las redes neuronales clásicas encuentran grandes limitaciones a la hora de trabajar con imágenes debido a la gran cantidad de parámetros que tienen, lo que converge en riesgo de sobreentrenamiento (overfitting) y la necesidad de una inicialización cuidadosa de los pesos [64] [68] [70] [73].

Las Redes Convolucionales constan de diferentes partes; cada parte cumple una función específica dentro de la arquitectura lo que hace este tipo de redes tan útiles y robustas. En la literatura se pueden encontrar diferentes ideas, etapas, partes o características de una red convolucional, los conceptos más estandarizados se los puede encontrar en [64] [66] [68] [70] [73] y de los cuales se pueden rescatar 3 ideas o características de una red convolucional que se mantienen en común:

- La idea de conectividad local en campos receptivos.
- La idea de parámetros compartidos mediante el uso de pesos y bias comunes.
- La idea de una invariancia a la transición mediante técnica de Pooling.

Así, dentro de la Red Neuronal Convolucional se ejecutan capas o etapas con cierto nivel de complejidad [64]. Para ilustrar esto, se puede inspeccionar la siguiente figura:

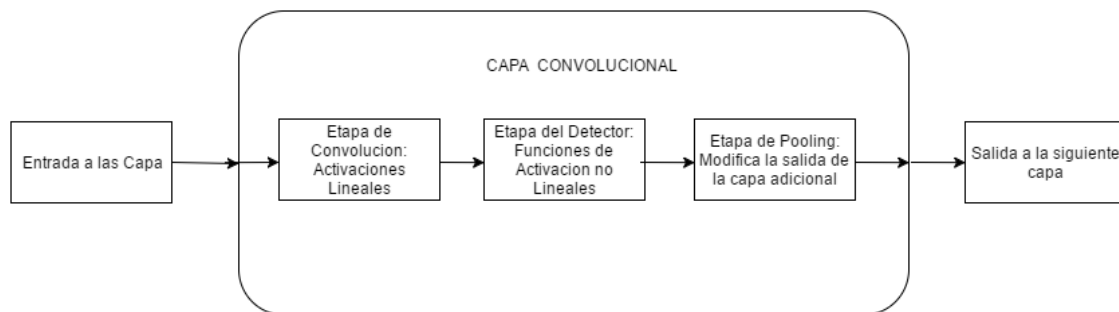


Figura 18: Componentes Típicos de la capa de una red neuronal convolucional

Centrándonos en la idea del proceso de convolución con imágenes, internamente la red convolucional ejecuta este proceso con un kernel (o filtro) [70] [76] realizando un barrido de pixel a pixel abarcando toda la imagen, de esta manera se generan los mapas de características (o feature maps en inglés) [64] [73]. El tamaño del kernel define el tamaño de los campos receptivos [73] y el barrido que se realiza a través de toda la imagen nos brinda una búsqueda pixel a pixel de una característica determinada, esto se logra gracias a que los pesos y bias asociados en cada pixel y en todo el campo receptivo son los mismos [64] [73]. Este proceso de convolución se lo realiza con un número establecido de kernels lo cual nos brinda “features maps” con diferentes características [64]. Para

este caso de aplicación, se estableció el número de kernels usados de manera heurística y se menciona con mayor detalle en el capítulo 5 de Análisis de Resultados.

De esta manera se puede entender las ideas de conectividad local y parámetros compartidos. Por otra parte, posterior a la obtención de los mapas de características es necesario en muchas ocasiones disminuir el número de parámetros que pasan de capa a capa, por este motivo es necesario una etapa de sub-muestreo (Pooling o downsampling en inglés) [66] [68].

La etapa de sub-muestreo tiene 2 funciones principales: La primera tiene la intención de brindar una robustez a la información que se está transmitiendo de capa a capa haciéndola invariante a la traslación local, mientras que la segunda función es la reducción de parámetros que se transmiten a la siguiente capa [64] [73]. Dichas funciones se logran mediante un mismo proceso; mediante el resumen o el promedio de valores en pequeñas cantidades, explicado de otra manera, la técnica de sub-muestreo (Pooling) toma pequeños grupos de valores provenientes de los mapas de características (features maps) y realiza un promediado de dichos valores para lograr reducir los parámetros y una invariancia a la traslación [68] [70]. De una manera análoga al proceso de convolución, el Pooling se lo realiza con un barrido a lo largo de todo y en todos los mapas de características. La técnica más usada para sub-muestreo (Pooling) es la de Max-Pooling [72] [73] la cual se realiza sin solapamiento y en grupos de 2x2 en imágenes, como se puede ver en la siguiente figura:

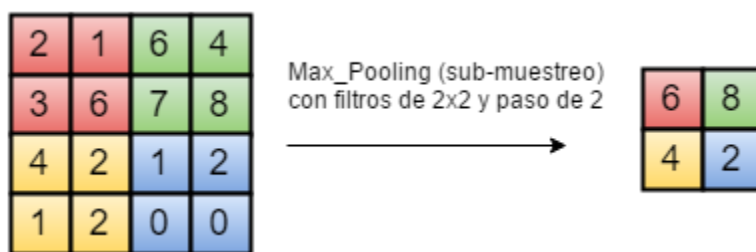


Figura 19: Técnica de Max-Pooling (sub-muestreo)

Todo el proceso descrito anteriormente se repite un número de veces según el número de capas que tenga la red, en cada ocasión y a medida que avanza en profundidad los atributos que se detectan se vuelven más abstractos [74] [75] [76]. De esta manera es como se logra una detección jerárquica de atributos [76].

Al final de la red de convolución se utiliza una capa de neuronas MLP (Multilayer Perceptron) en la cual se optimiza el entrenamiento y se realiza la clasificación.

Agrupando todas las capas y las etapas que existen dentro de una Red Convolutiva obtenemos una secuencia de procesos como se puede observar en la Figura 20 los cuales conforman una arquitectura típica de una red neuronal convolutiva.

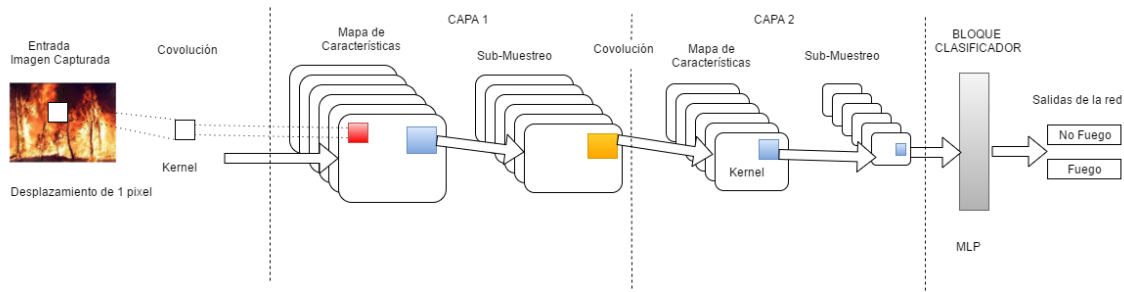


Figura 20: Secuencia de procesos en una Red Neuronal Convolutiva

### 3.2 FASE DE ENTRENAMIENTO

Para la fase de entrenamiento de la Red Neuronal Convolutiva se vio necesaria la creación de una base de datos de 60 000 imágenes que muestren una condición de incendio o no incendio, en otras palabras, imágenes que contengan fuego o no fuego. Los Detalles acerca de la creación de la base de datos se los revisa en el capítulo 4 de configuración experimental.

Esta base de datos se la creó debido a que no existe ningún repositorio digital con imágenes de este tipo como hay en el caso de bases con imágenes de objetos diferentes como es el caso del CIFAR y el MNIST. En la siguiente figura se muestran un mosaico con imágenes que contienen Fuego y No Fuego que se usaron para el entrenamiento de la red.



Figura 21: Mosaico de la base de Datos usadas para el entrenamiento: (Izq) Imágenes que contienen No Fuego. (Der) Imágenes que contienen Fuego.

Una vez obtenida la base de datos, se procedió a dividir la misma en 3 partes y con proporciones diferentes estableciendo así las bases de datos o conjuntos de Entrenamiento, Prueba y Validación (Train, test, val). Las Redes convolucionales mejoran su rendimiento de acuerdo al volumen de datos de entrada [71] [72] [73] [77] [78], por esta razón y con base en [71] [72] [78] se eligieron los porcentajes del 70% para el conjunto de Entrenamiento, 15% para el de Prueba y el 15% restante para Validación. Se utiliza el conjunto de “Entrenamiento” para realizar el entrenamiento del sistema como tal, el conjunto de “Validación” como un criterio de paro en caso de que no exista mejoría en la reducción del error y un conjunto de “Prueba” para probar y asegurar que el sistema entrenado

pueda generalizar y clasificar datos que no haya visto antes. Estos dos últimos se explicaran mejor en la siguiente sección.

Previo a la división de los porcentajes, las imágenes pasan por un proceso de “aplanamiento”, en el cual se unen los 3 canales de color uno a continuación de otro en un solo vector. Esto se repite para cada imagen, colocando cada imagen “aplanada” en una siguiente fila, generando así una matriz de imágenes “aplanadas”. De esta manera es como los datos (imágenes) ingresan a la red convolucional, en donde el primer gran conjunto de Entrenamiento, Prueba y validación contienen la base dividida con imágenes de Fuego y el segundo gran grupo contiene imágenes de No Fuego.

Debido al gran volumen de imágenes que ingresan a la red, las mismas ingresan en lotes proporcionales mediante grupos de 400 imágenes al mismo tiempo, este número de lote (batch) fue escogido de una manera heurística para esta aplicación en cuestión.

En la Figura 22 se puede observar de manera esquematizada el proceso que se sigue para el inicio del entrenamiento de la red neuronal convolucional.

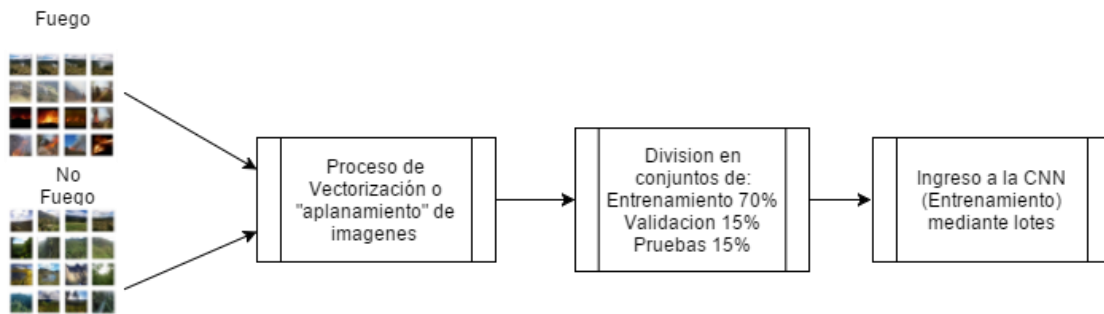


Figura 22: Preparación e inicio del Entrenamiento

Al ser una red neuronal convolucional, se requiere comprobar si efectivamente la red se encuentra aprendiendo, por lo que se requiere un método de optimización para la reducción del error y la actualización de los pesos; el método de optimización utilizado es el de Gradiente Descendiente Estocástico (SGD) que viene definido como sigue:

$$\theta = \theta - \alpha \nabla_{\theta} J(\theta; x^{(i)}, y^{(i)}) \quad (13)$$

Donde  $\alpha$  es la tasa de aprendizaje y  $J(\theta)$  es la función de costo en pares de  $x^{(i)}, y^{(i)}$  provenientes del conjunto de entrenamiento. El uso del SGD lleva a una convergencia más estable hacia el mínimo local y el paso o saltos que este da en cada iteración viene dado por la tasa de aprendizaje [67] [68].

Se realiza el ajuste de los pesos de la red neuronal para minimizar el error medio cuadrático (MSE) en el conjunto de Entrenamiento como se observa en las ecuaciones (14) (15) donde  $W$  representa los pesos y  $y^i$  representa la salida del sistema:

$$J(W) = \frac{1}{2} \sum_{i=1}^N (\bar{y}^i - y^i)^2 \quad (14)$$

$$J_i(W) = \frac{1}{2}(\bar{y}^i - y^i)^2 \quad (15)$$

### 3.3 FASE DE EVALUACION Y PRUEBA

Pese a que la extracción de atributos en las imágenes se lo realiza de manera automática en la red convolucional, este sigue siendo un aprendizaje supervisado. En las redes neuronales convencionales usualmente se maneja una base de datos para el entrenamiento y un porcentaje para realizar pruebas, teniendo únicamente conjuntos de Entrenamiento y Prueba. En el caso de esta aplicación debido al uso de una red convolucional, al gran volumen de datos que se tiene y a la necesidad de obtener los mejores rendimientos posibles en entrenamiento y clasificación, se optó por utilizar un conjunto de validación y otro conjunto de pruebas por separado.

El termino Overfitting (En inglés) puede ser entendido en el español como un sobre ajuste o sobre aprendizaje [67] y hace referencia a que la red está especializándose en un solo tipo de característica o en una región particular lo que conduce a un error de generalización en la clasificación aun cuando se obtiene un error de entrenamiento bajo como explica Erick Daseaev en [65]. Por esta razón, se ve necesario el uso de un conjunto de datos para validación.

Mediante el uso del SGD se optimiza el entrenamiento (error) mientras que con el uso de la validación nos brinda una regularización para evitar el sobreentrenamiento y “penalizar” cada vez que la red se equivoque de manera que se convierta en un criterio de paro para el algoritmo y que este no vaya más lejos en el entrenamiento. Posteriormente, después de obtener el modelo final, el conjunto de prueba sirve para comprobar si el sistema podrá clasificar datos que no ha visto nunca. Detalles de los algoritmos seguidos en esta etapa se encuentran en el capítulo 4.

Para llevar a cabo un seguimiento total del progreso del entrenamiento y verificar el estado del error, se realizó la supervisión del aprendizaje mediante un monitoreo del costo en función del número de épocas, medición de la exactitud del entrenamiento y validación [68] y el uso de diferentes métricas. Esto se puede apreciar de una manera concreta en el capítulo 5 de Análisis de resultados.

Dentro de la fase de evaluación, se ve necesaria la cuantificación de la calidad de las predicciones que tiene nuestro clasificador [79], para llevar a cabo esta acción es posible valerse del uso de métricas. Una métrica puede ser entendida como una función de distancia que mide similaridad [81]. El clasificador de esta aplicación al tener únicamente 2 opciones (clases) recae en la categoría de binario. En la literatura existen diferentes métricas de rendimiento en clasificadores booleanos [82], de los cuales se han seleccionado los siguientes:

- Exactitud (Accuracy).
- Recall
- F1-Score
- Matriz de Confusión.

#### 3.3.1 METRICAS USADAS

##### *EXACTITUD (ACCURACY)*

La exactitud se comprende como el grado de cercanía que tiene una cantidad medida con respecto al valor real de dicha cantidad [83]. Este es el concepto más aceptado y generalizado sobre exactitud [80] [83]. La exactitud en términos de repetitividad es el grado en que mediciones repetidas bajo

condiciones inalteradas muestran los mismos resultados [83]. De modo computacional, si  $\hat{y}$  es el valor predicho de la  $i$ -ésima muestra y  $y_i$  es el valor verdadero correspondiente, entonces la fracción de predicciones correctas sobre las muestras está definido por [79]:

$$Acc(y, \hat{y}) = \frac{1}{n \text{ muestras}} \sum_{i=0}^{n-1} 1(y_i = \hat{y}_i) \quad (16)$$

### **MATRIZ DE CONFUSION**

También conocida como la matriz de error [84]. Es un arreglo o tabla que permite una visualización sobre el rendimiento de un algoritmo. Esta tabla contiene 2 columnas que representan la clase predicha y 2 filas que representan la clase actual [85]. Esta métrica permite un análisis más detallado sobre el rendimiento real del clasificador. La terminología típica usada por la matriz de confusión es:

- TP: Verdadero Positivo
- TN: verdadero Negativo
- FP: falso positivo
- FN: falso negativo

Total de Muestras/Poblacion		Condición Predicha	
		C. P. Positiva	C. P. Negativa
Condición Verdadera	Cond. Positiva	Verdadero Positivo	Falso Negativo
	Cond. Negativa	Falso Positivo	Verdadero Negativo

Tabla 7: Matriz de Confusión de 2x2

### **PRECISION, RECALL y F1-SCORE**

La métrica de RECALL y PRECISION vienen definidas como la siguiente proporciones:

$$RECALL = \frac{tp}{tp + fn}; PRECISION = \frac{tp}{(tp + fp)} \quad (17)$$

En donde  $tp$  es el número de verdaderos positivos,  $fn$  el número de falsos negativos y  $fp$  el número de falsos positivos. RECALL es la habilidad intuitiva del clasificador de encontrar todas las muestras positivas [79] mientras que la precisión es la capacidad del clasificador de no etiquetar como positivo algo que es negativo [79]. El mejor valor es 1 y el peor es 0.

Por su parte, la métrica de la puntuación F1 (F1-SCORE) puede ser interpretada como una media ponderada de la precisión y recall [79]. La fórmula de esta métrica es la siguiente:

$$F1 = 2 * (precision * recall) / (precision + recall) \quad (18)$$

De igual manera esta métrica alcanza su mejor valor en 1 y el peor en 0.

La implementación de las métricas mencionadas se las llevó a cabo mediante la herramienta scikit-learn que contiene funciones propias para el análisis de datos compatibles con Python. El uso de estas métricas para este caso de aplicación específico se puede apreciar en el capítulo 5 de Análisis de Resultados.

### 3.4 FASE DE EVALUACION EN TIEMPO REAL DEL SISTEMA

En la Etapa final de esta aplicación, se desarrolló un sistema integral de detección de incendios forestales y/o la presencia de Fuego. Un esquema general de la secuencia que sigue el sistema desarrollado se muestra a continuación:

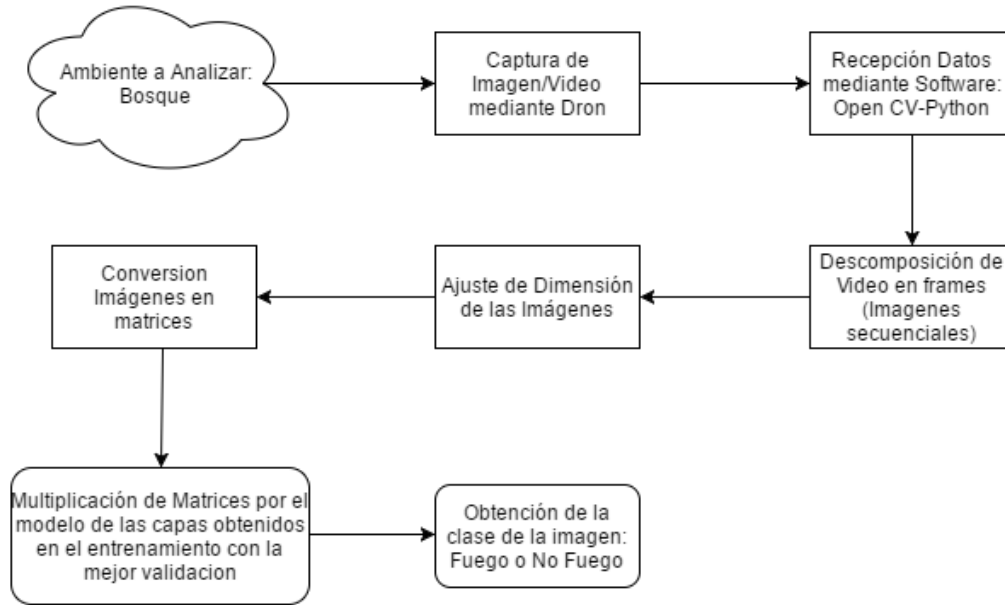


Figura 23: Secuencia completa del sistema de detección de Incendios Forestales

La orden de realizar la captura de imágenes se realiza desde un ordenador, de esta manera se habilita la transmisión de video por parte del dron hasta el computador con el software requerido. Como se puede apreciar en la figura anterior, la imagen que se captura se procesa internamente a través del modelo obtenido previamente mediante el entrenamiento y se obtiene como resultado una clase: Presencia o Ausencia de Fuego. A continuación se presenta un esquema general del sistema de detección de Incendios Forestales.

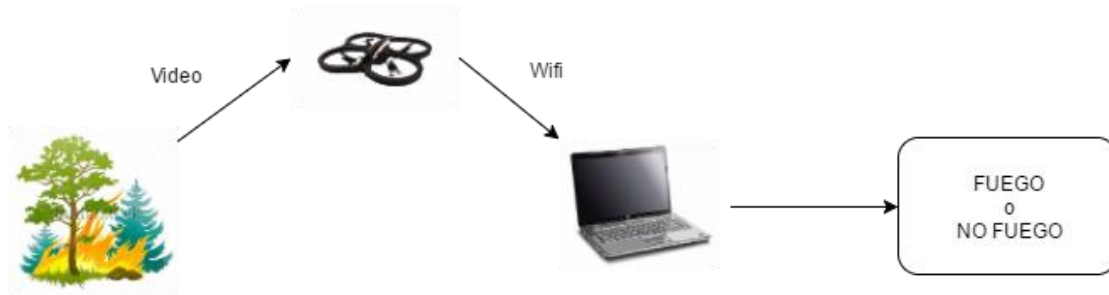


Figura 24: Proceso General del Sistema de Detección de Incendios

El Dron realiza la captura del video y este es transmitido mediante wifi al ordenador en donde se procesa y se obtiene el resultado; Si es Fuego o No Fuego.



# CAPÍTULO 4. CONFIGURACIÓN EXPERIMENTAL

## 4.1 UNIDAD DE VUELO NO TRIPULADA (UAV)

Se utilizó la UAV de la casa Parrot SA, este cuadricoptero es conocido como AR Parrot 2.0 puede ser controlado mediante un Smartphone o desde un computador via wifi, dispone de una autonomía de vuelo de 18 minutos por carga completa con un rango de alcance de 50m, en la Tabla 8 se observan todas sus características [86].

Grabación de video	Asistencia Electrónica	Estructura	Motores
Cámara HD 720p a 30fps.	Procesador AR M Cortex A8 de 32 bits a 1 GHz con DSP de video TMS320DMC64x a 800 MHz Linux 2.6.32 RAM	ubos de fibra de carbono: peso total de 380 g	4 motores de rotor interno ("inrunner") sin escobillas. 14,5 W.
Objetivo gran angular: 92°.	USB 2.0 de alta velocidad para extensiones y grabación Wi-Fi b/g/n	Piezas de plástico nylon cargado con 30% de fibra de vidrio de alta calidad.	Rodamiento de bolas en miniatura.
Almacenamiento de vídeo sobre la marcha con Wi-Fi.	DDR2 de 1 GB a 200 MHz	Casco de protección de polipropileno.	Engranajes de Nylatron de bajo.
Resolución de la imagen: 1280x720 píxeles	Giroscopio de 3 ejes 2.0000/s	Nanorevestimiento repelente a los líquidos en los sensores de ultrasonidos.	Eje de las hélices de acero templado.
	Acelerómetro de 3 ejes +/-50 mg		Microcontrolador AVR de 8 MIPS por controlador de motor.
	Magnetómetro de 3 ejes, precisión 6°		Batería recargable Li-Po de 3 elementos, 1.000 mAh.

Tabla 8: Características del AR Drone 2.0

## 4.2 HARDWARE UTILIZADO

Las unidades de procesamiento gráfico (GPUs por sus siglas en inglés) son dispositivos formados múltiples procesadores conectados en paralelo, comúnmente fueron utilizados para la implementación de filtros convolucionales como lo muestra en [87] con el fin de mejorar el rendimiento de sus algoritmos. En la actualidad son usados para computación acelerada en conjunto con la CPU. Según [88] el uso de estos dispositivos permite obtener ganancias de 7000x en tarjetas NVIDIA con tecnología CUDA (vista en el siguiente apartado) siempre y cuando el problema pueda ser paralelizado. Se debe tener presente que entre más prestaciones brinde la GPU de igual manera deberá crecer la robustez de la CPU.

A continuación, se presentan las características tanto del equipo de trabajo (CPU) con su respectiva GPU:

### *GPU NVIDIA Titan X*

Las especificaciones detalladas de la tarjeta gráfica obtenidas de [89] se observan en la Tabla 9.

Nucleos	3072
Frecuencia de reloj	1000 MHz
Frecuencia acelerada	1075 MHz
Tasa de relleno de texturas	192 GigaTexels/s
Frecuencia de memoria (Gbps)	7,0
Cantidad de memoria	12GB
Interfaz de memoria	348-bit GDDR5
Ancho de banda máximo	336,5 GB/s

*Tabla 9: Especificaciones de la GPU*

### *Equipo de trabajo*

El equipo utilizado para el entrenamiento de la CNN, cuenta con las características técnicas expuestas en la Tabla 10.

Procesador	Intel Core I7
Memoria RAM	64 GB DDR4
Tarjeta Grafica	12GB Nvidia Titan X
Disco SSD	250 GB
Disco HDD	1.5 TB
Sistemas Operativo	Linux

*Tabla 10: Características técnicas del equipo*

## 4.3 SOFTWARE UTILIZADO

Python: Es un lenguaje multiplataforma de propósito general que permite una programación en alto nivel. Permitiendo la programación orientada a objetos, programación imperativa y programación funcional. Una de sus debilidades es la creación de objetos para la definir escalares, esto aumenta la sobrecarga de cálculo numérico, por lo cual es necesario usar alternativas como NumPy o Theano. [90]

## **CUDA**

Compute Unified Device Architecture (en inglés) es una arquitectura para cálculo paralelo desarrollada por NVIDIA, permite aprovechar al máximo la GPU mediante una programación en alto nivel; en lenguajes como C++, Fortran o OpenACC. Soporta las series G8X NVIDIA, en adelante, incluyendo GeForce, Quadro, ION y la línea Tesla.

Esta arquitectura permite la ejecución de múltiples hilos de tarea, en forma simultánea, aprovechando así el paralelismo de las tarjetas gráficas y su alto ancho de banda en memoria, volviendo los procesos de cálculo mucho más rápidos [91]. Es utilizada en aplicaciones como:

- Cálculos convolucionales.
- Bioinformática.
- Cálculo financiero.
- Dinámica de fluidos computacionales.
- Ciencia de los datos, analítica y bases de datos.
- Defensa e inteligencia.
- Procesamiento de imágenes y visión computarizada.
- Aprendizaje automático.
- Dinámica molecular.

## **Theano**

Es un compilador para expresiones matemáticas en lenguaje Python, mediante una programación de alto nivel similar a la sintaxis de Numpy, fue desarrollado por el grupo de machine learning de la Université de Montréal. Theano optimiza la ejecución de expresiones matemáticas convirtiendo el código a lenguaje C++ o (CUDA para la GPU) para luego ser compilado mediante sus módulos. Los algoritmos de machine learning son los más utilizados cuya implementación es 7.5x (compilado en CPU) y 44x (compilado en GPU) más rápidos que sus similares; Matlab, NumPy y SciPy. [92]

**Eclipse:** Es un entorno de programación compuesto por diferentes herramientas para el desarrollo de código abierto [93].

**PyDev:** Es un plug-in de terceros, para Eclipse, que puede ser usado para el desarrollo de código en lenguaje Python [94].

## **4.4 OBTENCION DE IMAGENES**

Dada la falta de existencia de una base de datos de imágenes de incendios forestales disponible en la web, se optó por crear una base de datos. La obtención de imágenes fue realizada en dos etapas; para la primera se utilizó una unidad de vuelo no tripulada, ArDrone 2.0 cuyas características están expuestas en la Tabla 8, para la segunda etapa se utilizó google imágenes con el fin de ampliar la gama de imágenes.

### **PRIMERA ETAPA**

Se realizó en el periodo del 20 Noviembre al 20 Diciembre del 2015, pertenecientes a la estación de verano en Ecuador. Se optó por realizar la toma en dos localidades diferentes; la primera está ubicado

en el cantón de Santa Isabel localizado al sur de la provincia del Azuay, a unos 55 Km de la ciudad de Cuenca (véase Figura 25: Ubicación geográfica del cantón Santa Isabel); la segunda se encuentra ubicado en la comunidad de Rodeo perteneciente a la parroquia Baños, ubicada al sur del cantón Cuenca, a unos 14Km de centro de la capital azuaya (véase Figura 26).

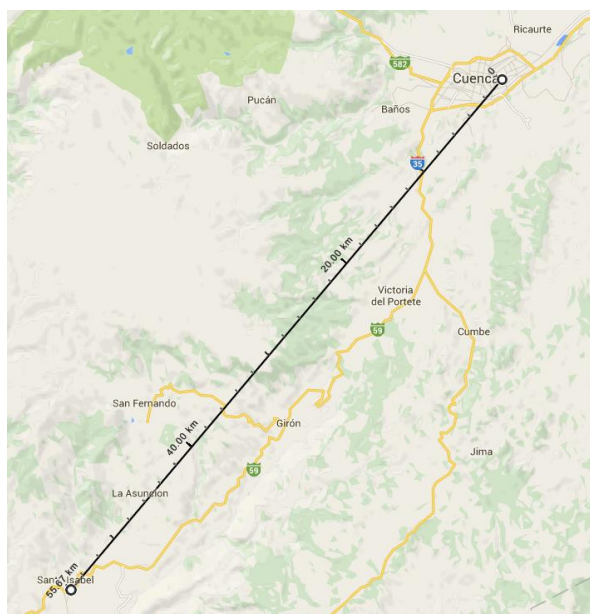


Figura 25: Ubicación geográfica del cantón Santa Isabel

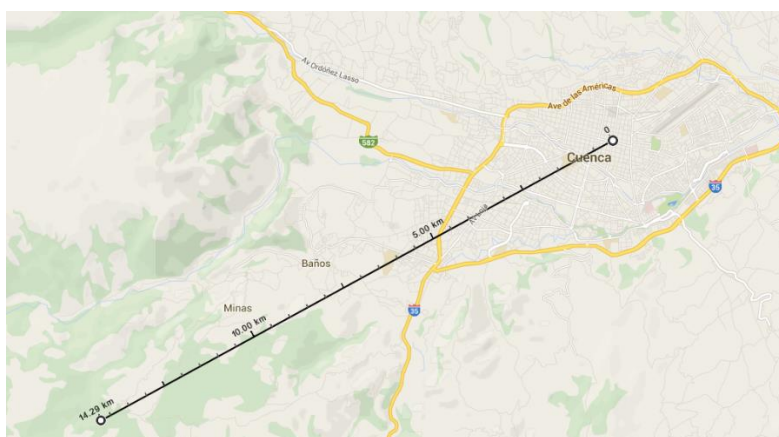


Figura 26: Ubicación geográfica de la comunidad de Rodeo en la parroquia Baños

Por cada zona se realizó una simulación de un incendio forestal dentro de un espacio controlado, centrándose en la generación de humo, dejando a un lado la obtención de fuego; esto debido a la dificultad de controlar llamas de más de 2 metros, lo cual es compensado en la segunda etapa.

Por zona se obtuvo 300 imágenes dando un total de 600 imágenes, donde, el 50% representan un incendio forestal y el otro 50% representan una zona libre de incendio.

Para la toma de imágenes se utilizó la cámara del AR Drone 2.0 especificada en la Tabla 8.

**SEGUNDA ETAPA**

Se utilizó el buscador Google para compensar la falta de incendios forestales con presencia de fuego, obteniendo un total de 400 imágenes, que al igual que en la primera etapa, el 50% representan un incendio forestal y el otro 50% representan una zona libre de incendio.

De esta forma se obtuvo 1000 imágenes (720x360 pixels) de diferentes lugares como se observa en la Tabla 11, las cuales posteriormente pasaran a una etapa de pre-procesamiento con el fin de generar la base de datos final.






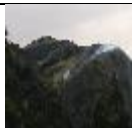
























Incendio					
					
					
No incendio					
					
					

Tabla 11: Imágenes Iniciales sin transformación

**4.5 ETAPA DE PRE-PROCESAMIENTO Y GENERACION DE LA BASE DE DATOS**

Para la obtención de la base de datos se implementó tres transformaciones geométricas a cada imagen; recorte, rotación y escalado, esta última se realizó a fin de disminuir el costo computacional debido al gran tamaño de las imágenes. Posteriormente cada una de las imágenes fue transformada

en un vector de una sola dimensión. De forma aleatoria se crearon tres matrices: entrenamiento (train), prueba (test) y validación (val), con el 70%, 15% y 15% de las imágenes totales respectivamente. Todos estos procesos fueron creados usando la librería Opencv y Numpy dentro del lenguaje Python.

**PROCEDIMIENTO:**

En la transformación de recorte se realizó una incisión del 30% de cada una de las 1000 imágenes iniciales, obteniendo así 1000 nuevas imágenes, dando un total de 2000, el algoritmo e presenta en la Figura 27.

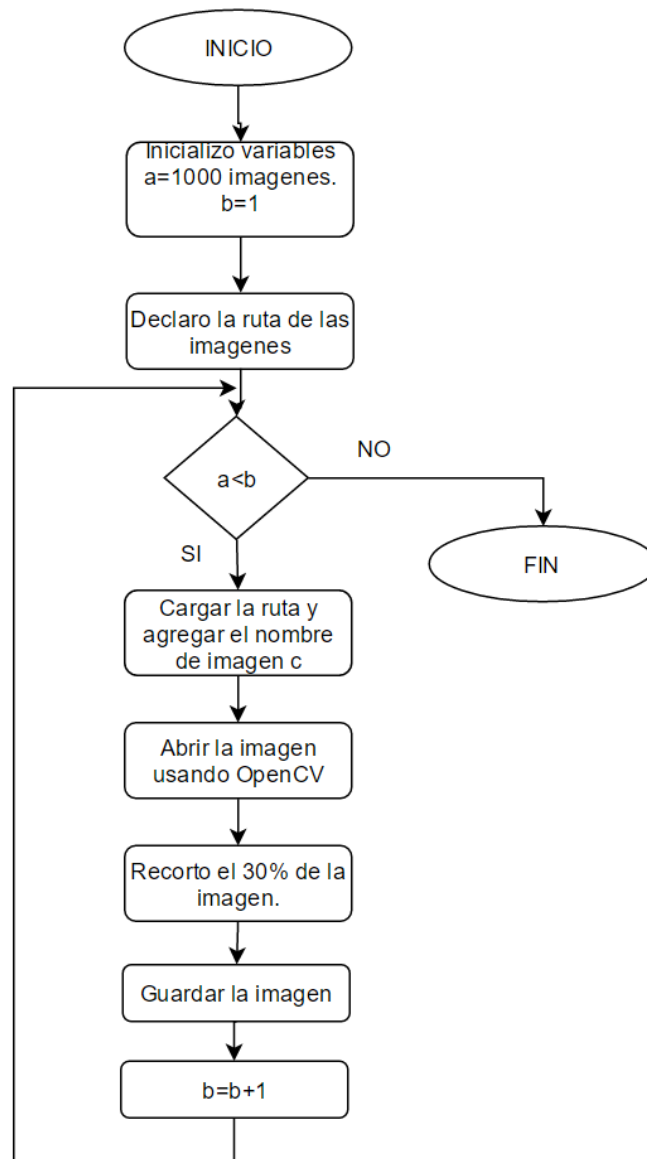


Figura 27: Algoritmo para la transformación de recorte

Para la rotación se utilizaron las 2000 imágenes del literal anterior, en donde a cada una de ellas se realizó una rotación con paso de 12 grados hasta completar los 360 grados, de esta forma se obtuvo 30 imágenes por vuelta, obteniendo así un total de 60000 imágenes (Total=1000\*2\*30=60000) (véase Tabla 12) el algoritmo correspondiente se presenta en la Figura 28.

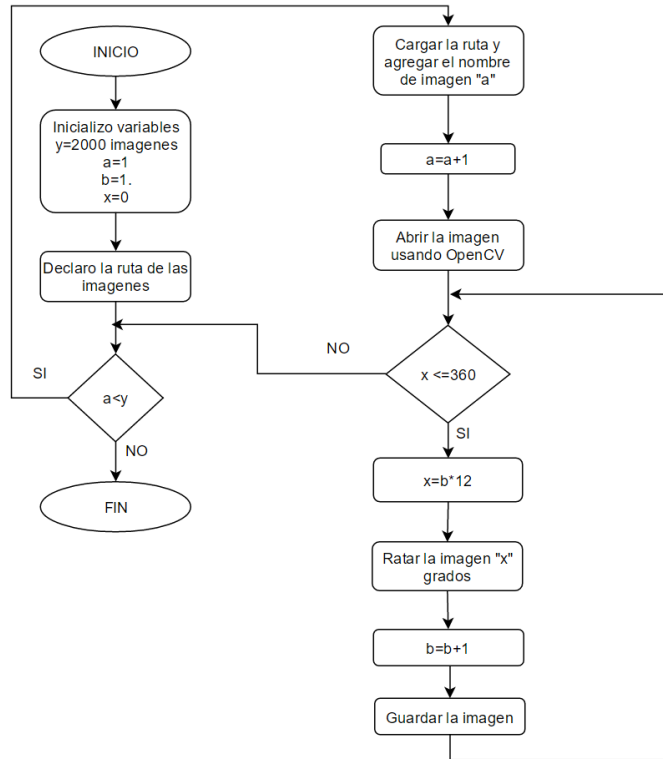






Figura 28: Algoritmo para la transformación de rotación

		Transformación de rotación			
		12 Grados	24 grados	...	360 grados
Inicial					

Transformación de recorte escalado					
------------------------------------	---	---	--	--	---

Tabla 12: Transformaciones Geométricas

Para la etapa de escalado se optó por una dimensión de 64 x 64, esto con el fin de evitar una sobrecarga en el tamaño del tensor que soporta Python (1GB máximo) el algoritmo utilizado se presenta en la Figura 29. En la tabla 13 se puede observar 30 imágenes aleatorias de la base de datos final con sus respectivas transformaciones.

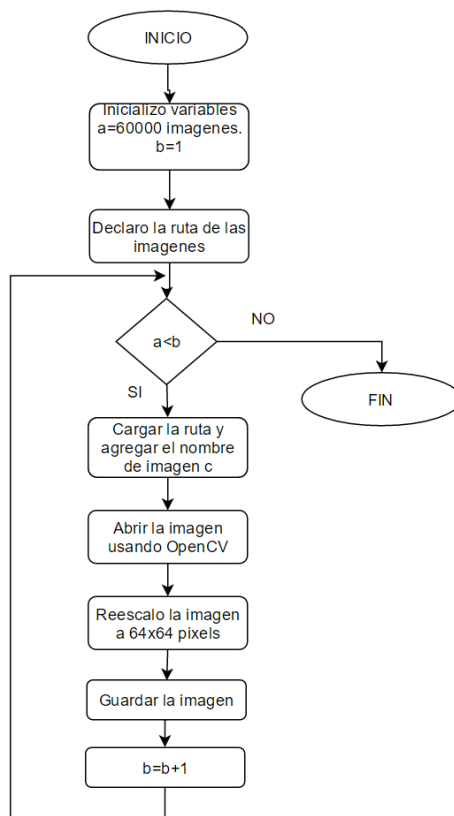


Figura 29: Algoritmo para la transformación de escalado








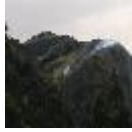

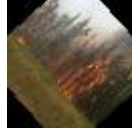



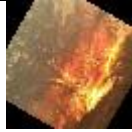


















Incendio					
					
					
No incendio					
					
					

Tabla 13: Imágenes escaladas (Base de datos Final)

Cada una de las 60000 imágenes fueron convertidas en vectores según lo mencionado en el capítulo 3, posteriormente se procedió a formar tuplas (lista) con cada uno de estos vectores; 70% de las imágenes para entrenamiento, 15% para validación y 15% para prueba. Estas tres matrices se guardaron en una matriz multidimensional para luego ser transformada en un archivo comprimido de extensión pkl, el algoritmo utilizado para este apartado se observa en la Figura 30.

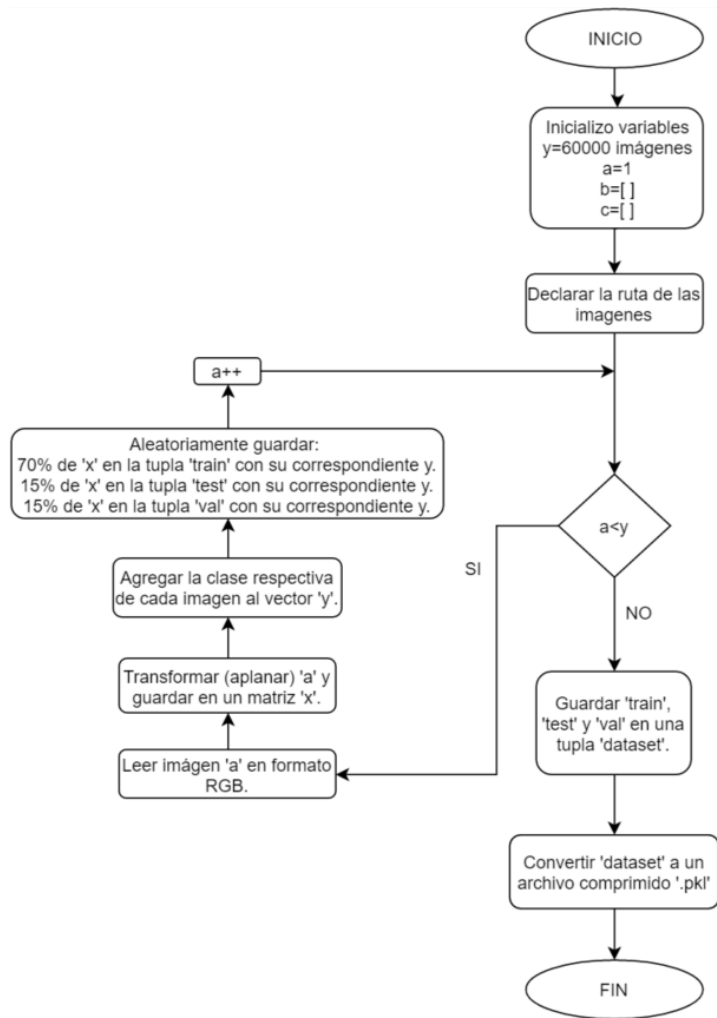


Figura 30: Algoritmo para la generación del dataset.pkl

En la figura 31 se observa el algoritmo de entrenamiento para la red CNN.

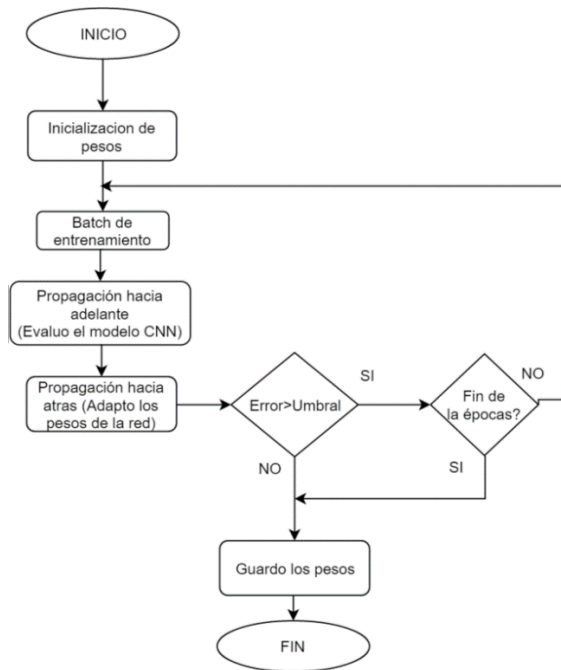


Figura 31: Algoritmo para de entrenamiento.

Este punto se presenta el algoritmo de supervisión de aprendizaje (Figura 32) que se utilizará para el desarrollo de diferentes pruebas que se verán en el capítulo siguiente.



Figura 32: Diagrama de desarrollo del sistema

# CAPÍTULO 5. ANALISIS DE RESULTADOS

## 5.1 RESULTADOS DEL ENTRENAMIENTO

Ocho conjuntos de pruebas han sido analizados, presentado en la Tabla 14, para evaluar el rendimiento del proceso de entrenamiento de la red, utilizando los siguientes parámetros de configuración:

- **Tamaño de la imagen (tam\_i):** Se implementó utilizando dos tamaños; 28x28 y 64x64.
- **Tipo de imagen (tip\_i):** Representa el tipo, si es a escala de gris o a RGB.
- **Tasa de aprendizaje (tas\_a):** Permite definir la velocidad de convergencia del algoritmo, puede tomar valores entre 0 – 1.
- **Número de kernels (num\_k):** Representa el número de activaciones no líneas por cada capa convolucional.
- **Tamaño del kernel (tam\_k):** Define el tamaño de los campos receptivos.
- **Numero de neuronas (num\_n):** Cantidad de neuronas implementadas en la capa de clasificación.
- **Tamaño del batch (tam\_b):** Cantidad de imágenes en el que se divide el conjunto de entrenamiento.
- **Numero de épocas (num\_e):** Número de veces que se entrenara la red.
- **Capas convolucionaes (C):** esta definido por las variables C seguido del número de capa convolucional. Ejm: C1,C2...Cn.

Para las diferentes pruebas se ha mantenido constante; la tasa de entrenamiento, el tamaño del batch y el número de épocas.

Parámetros Pruebas	tam_i (pixel)	tip_i	tas_a	num_k	tam_k	num_n	tam_b	num_e
P1	28x28	Gris	0.01	C1=40 C2=80	C1=5x5 C2=5x5	1000	400	400
P2	28x28	Gris	0.01	C1=80 C2=160	C1=5x5 C2=5x5	1000	400	400
P3	28x28	RGB	0.01	C1=20 C2=40	C1=5x5 C2=5x5	1000	400	400
P4	28x28	RGB	0.01	C1=40 C2=80	C1=5x5 C2=5x5	1000	400	400
P5	28x28	RGB	0.01	C1=80 C2=160	C1=5x5 C2=5x5	1000	400	400

P6	64x64	RGB	0.01	C1=20 C2=40	C1=17x17 C2=17x17	2000	400	400
P7	64x64	RGB	0.01	C1=40 C2=80	C1=17x17 C2=17x17	2000	400	400
P8	64x64	RGB	0.01	C1=80 C2=160	C1=17x17 C2=17x17	2000	400	400

Tabla 14: Parámetros para cada Prueba

Los resultados por clase son expuestos en la Tabla 15 donde se obtiene el rendimiento de las métricas de precisión, recall y f1-Score. Los mejores resultados se dan en la prueba 7 (P7) y la prueba 8 (P8) siendo esta última la mejor, cuya precisión para la clase 0 es de 0.9 y para la clase 1 de 0.89, teniendo como configuración de parámetros es: tamaño de la imagen igual a 64x64 píxeles; una tasa de aprendizaje de 0.01, dos capas convolucionales con 80 kernels para la primera y 160 kernels para la segunda; con un tamaño de kernel de 17x17.

Métricas		Precisión	Recall	F1-Score
Pruebas por clases				
P1	No incendio (0)	0.63	0.81	0.71
	Incendio (1)	0.74	0.53	0.62
P2	No incendio (0)	0.68	0.80	0.72
	Incendio (1)	0.74	0.59	0.69
P3	No incendio (0)	0.80	0.79	0.80
	Incendio (1)	0.82	0.83	0.82
P4	No incendio (0)	0.80	0.80	0.80
	Incendio (1)	0.83	0.83	0.83
P5	No incendio (0)	0.83	0.82	0.83
	Incendio (1)	0.84	0.85	0.85
P6	No incendio (0)	0.87	0.87	0.87
	Incendio (1)	0.87	0.87	0.87
P7	No incendio (0)	0.90	0.88	0.89
	Incendio (1)	0.88	0.90	0.89
P8	No incendio (0)	0.90	0.89	0.89
	Incendio (1)	0.89	0.90	0.89

Tabla 15: Métricas de Rendimiento análisis por clase

Los resultados por entrenamiento se exponen en la Tabla 16 estos valores son los promedios, entre clases por cada prueba, de las métricas de Precisión y f1-Score, además, como medida adicional se agrega la exactitud. La diferencia de rendimientos de las métricas (Exactitud, Precisión y F1-Score)

entre las pruebas 7 y 8, es mínima, teniendo como variación un valor absoluto de 0.0021 para la Exactitud y manteniendo el resto de métricas iguales a 0.89.

Métricas \ Pruebas	Exactitud	Precision	F1-Score
P1	0.6694	0.68	0.66
P2	0.6945	0.7	0.69
P3	0.8101	0.81	0.81
P4	0.8164	0.82	0.82
P5	0.8369	0.84	0.84
P6	0.8735	0.87	0.87
P7	0.8917	0.89	0.89
P8	0.8938	0.89	0.89

Tabla 16: Métricas de rendimiento por prueba

Si bien el rendimiento entre P7 y P8 son similares, no ocurre lo mismo con los tiempos de entrenamiento (Tabla 17), donde, para P7 tarda 146.18mins y para P8 284.18mins existiendo una diferencia de 128 mins. Esto nos lleva a reconocer que entre más robusta sea la configuración de nuestra red más tiempo le tomara terminar su entrenamiento.

Pruebas	Tiempo de entrenamiento (mins)
P1	23.01
P2	32.95
P3	17.57
P4	28.32
P5	59.15
P6	100.41
P7	146.18
P8	284.18

Tabla 17: Tiempos de Entrenamiento

La matriz de confusión obtenida de la mejor prueba se observa en la Tabla 18, esta matriz fue conseguida a partir del conjunto de prueba (test).

		Clases reales	
		Incendio	No incendio
Clases estimadas	Incendio	5510	690
	No incendio	627	5573

Tabla 18: Matriz de Confusión de la Prueba 8.

Las curvas de entrenamiento y validación, Figura 33, son descendentes; la primera asegura que la red está aprendiendo nueva información cuyo mse final es alrededor de 0.11 y para la segunda, la disminución del mse hacia 0.1 asegura que la red está generalizando el conocimiento.

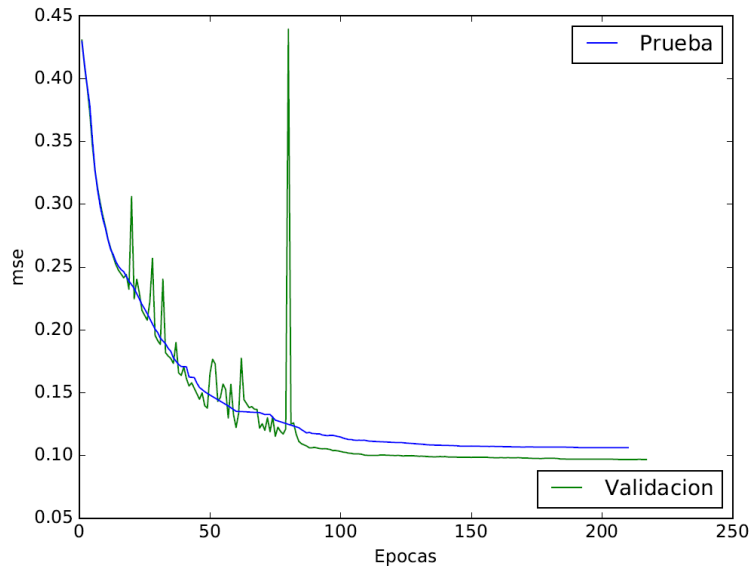


Figura 33: Curvas de error para los datos de prueba y validación para la prueba 7

# CONCLUSIONES

En el presente trabajo se trató dos apartados importantes, por una parte, la generación de la base de datos sobre incendios forestales, debido a que no existe alguna disponible en la web, y por otra parte la generación de 8 arquitecturas de red neuronal convolucional (CNN). Este enfoque pretende descubrir cuál es el conjunto de parámetros más óptimos para la creación del sistema propuesto en este trabajo y, por otra parte, dar a conocer las ventajas del uso de redes neuronales convolucionales en el procesamiento de imágenes. La ventaja más clara que tienen las CNNs son:

- Eliminación de la etapa de extracción de características; En sistemas tradicionales es necesario una extracción, pero aún más importante para el entrenamiento de una red neuronal es la selección de las características que se desea que nuestro sistema detecte y aprenda. Esto en diferentes aplicaciones representa un gran problema puesto que es necesario en muchas ocasiones un agente (persona) experto para que realice una adecuada extracción de atributos y la red tenga un nivel de clasificación aceptable. Además, dicha extracción debe repetirse para cada caso de aplicación siempre con atributos diferentes. Lo que se ve obviado con el uso de CNNs.

Inspeccionando los resultados obtenidos de las pruebas se puede decir lo siguiente:

- a) EL uso de imágenes con un tamaño de 28x28 pixeles a escala de gris, entregan una exactitud de 0.6694 para redes con 40 kernels y 80 kernels perteneciente a P1, este valor aumenta si varía el número de kernels; 80 para la primera capa convolucional y 160 para la segunda, llegando a una exactitud de 0.6945 en P2. Se utilizó un tamaño de kernel de 5x5 y 1000 neuronas para la capa de clasificación como se observa en la tabla 14.
- b) El uso de imágenes a color en el formato RGB y el aumento del número de kernels para cada capa convolucional, ayuda a mejorar la precisión en las diferentes pruebas a pesar de mantener el mismo tamaño de 28x28 pixeles, obteniendo los siguientes resultados; 0.8101, 0.8164, 0.8369 de exactitud para P3, P4 y P5 respectivamente. EL tamaño del kernel y el número de neuronas se mantuvieron como en (a).
- c) La variación del tamaño de la imagen de 28x28 pixeles a 64x64 pixeles, ha permitido mejorar, aún más, el rendimiento de la exactitud obteniendo valores de 0.8735 para P6, 0.8917 para P7 y 0.8938 para P8. Como se observa en la tabla 15, estos valores se han conseguido gracias a variar el número de kernels por cada capa convolucional y manteniendo el resto de parámetros constantes. En este caso el tamaño del kernel para las tres pruebas fue de 17x17 y el número de neuronas utilizadas en la capa de clasificación fue de 2000.

Como se ha visto en los literales anteriores, variar tanto el número y tamaño de los kernels, como el tamaño de la imagen, ha permitido llevar un valor de exactitud de 0.7 (P1, literal a) a 0.8938 (P8, literal c), mejorando de forma sustancial el rendimiento del clasificador CNN.

Con base en los resultados obtenidos en la Tabla 15 y Tabla 16 y teniendo en cuenta el área de aplicación; el usar imágenes de una dimensión de 64x64 pixeles ha permitido obtener un rendimiento de clasificación muy aceptable. Por lo que se puede decir, que la información presente en las imágenes originales (256x256 pixels) se encuentra resumida en las imágenes redimensionadas y que no es absolutamente necesario el uso de imágenes con mayor tamaño para realizar el entrenamiento de las CNN.



## **RECOMENDACIONES**

A continuación se mencionan algunas recomendaciones de gran importancia para el desarrollo de trabajos similares:

Se recomienda el uso de un UAV (drone) que tenga adecuadas prestaciones en cuanto a resolución de la cámara, y tiempo de vuelo basado en la batería puesto a que estos factores son los más importantes a la hora de realizar tomas aéreas y recorrer un sector establecido en busca de incendios. Además, disponibilidad de framework para el desarrollo de futuros proyectos o aplicaciones.

Por otra parte, es recomendable diversificar los ambientes (imágenes) con los cuales se entrena el sistema para generalizar el conocimiento aprendido, permitiendo así una detección de incendios más eficiente en diferentes ambientes.

También se recomienda probar diferentes frameworks actuales para el entrenamiento de las redes convolucionales a fin de disminuir los tiempos de entrenamiento debido a que estos vienen optimizados para el uso GPU como es el caso de DIGITS de NVIDIA. Por lo cual es necesario utilizar computadores que dispongan de tarjetas gráficas.

## **TRABAJOS FUTUROS**

El proyecto va enfocado hacia la industria de la vigilancia y sistemas de seguridad, centrándose en el monitoreo forestal, permitiendo detectar cuando el fuego comienza a esparcirse a través de la visualización de humo y fuego, pero el sistema puede ser migrado a una vigilancia general y/o específico de anomalías abarcando desde áreas abiertas hasta lugares cerrados, siempre y cuando se realice los cambios adecuado en su software y hardware.

Por otra parte, se puede integrar al sistema funcionalidades como autonomía de vuelo para evitar posibles colisiones y el trazado de rutas para el monitoreo automático de zonas.

Finalmente se puede buscar la integración de todas las funcionalidades previamente mencionadas en un sistema integral dentro las capacidades de procesamiento propias del dron para obtener un sistema portátil de monitoreo y detección de incendios forestales

## REFERENCIAS

- [1] FAO, “Evaluación de los recursos forestales mundiales”, Septiembre. 2015, [En línea], Disponible: <http://www.fao.org/3/a-i4886s.pdf> [Consulta: 9 febrero 2016].
- [2] R. Nasi, R. Dennis, E. Meijaard, G. Applegate P. Moore, “Los incendios forestales y la diversidad biológica”, Febrero 2002, [En línea], Disponible: <http://www.fao.org/docrep/004/y3582s/y3582s08.htm> [Consulta: 9 febrero 2016].
- [3] E. Rochford, Module 1 en “CAPE Communication Studies: Practical Exercises for Paper 02 Essays”, Ed. iUniverse, Bloomington, 2015, [En línea], Disponible: [https://books.google.com.ec/books?id=TD\\_wCgAAQBAJ&pg](https://books.google.com.ec/books?id=TD_wCgAAQBAJ&pg) [Consultado: 8 febrero 2016]
- [4] International Wildland Fire Conference, “Regional Statement South America Region – An Input Papers to the Conference Statement”, Octubre 2015, [En línea], Disponible: <http://www.fire.uni-freiburg.de/korea-2015/Files-Korea-2015/Statements/IWFC-6-Regional-Statement-South-America.pdf> [Consultado: 9 febrero 2016]
- [5] Secretaria de Gestión de Riesgos (Ecuador), “Amenazad Incendios Forestales”, Julio 2014, [En línea], Disponible: [http://www.gestionderiesgos.gob.ec/wp-content/uploads/downloads/2014/07/Escenarios\\_JUNIO\\_lf.pdf](http://www.gestionderiesgos.gob.ec/wp-content/uploads/downloads/2014/07/Escenarios_JUNIO_lf.pdf) [Consultado: 10 febrero 2016]
- [6] Diario El Tiempo. “Incendios forestales quemaron 3.102 hectáreas en Quito”, Septiembre 2015, [En línea], Disponible: <http://www.eltiempo.com.ec/noticias-cuenca/169087-incendios-quemaron-3-102-has-en-quito/> [Consultado: 3 febrero 2016]
- [7] Diario El Comercio, “40 hectáreas afectadas por un incendio forestal en Azuay”, Septiembre 2015, [En línea], Disponible: <http://www.elcomercio.com/actualidad/hectareas-afectadas-incendio-forestal-cuenca.html> [Consultado: 3 febrero 2016]
- [8] David Martín-Borregón Domènech. Sistema de detección de incendios forestales utilizando técnicas de procesamiento de imagen. Universidad Politecnica de Cataluña.. 2011-2012
- [9] Jesús San-Miguel-Ayanz. Comprehensive Monitoring of Wildfires in Europe: The European Forest Fire Information System (EFFIS) European Commission, Joint Research Centre Italy.
- [10](Steele and Stier, 1998). The Economics of Forest Disturbances: Wildfires, Storms, and Invasive Species

- [11] JING JING LIANG. Spatial factors influencing large wildland fire expenditures forestry sciences lab rocky mountain research station u.s. department of agriculture.
- [12] Manual de formación de incendios forestales para cuadrillas. Gobierno de Aragón. Departamento de Medio Ambiente. Disponible en: <http://www.aragon.es/>
- [13] Ahmad A. A. Alkhatib. A Review on Forest Fire Detection Techniques. The University of SouthWales, UK.
- [14] Víctor Hugo Medina García. Software para la detección y extinción de incendios forestales. Universidad Distrital Francisco José de Caldas, Bogotá, Colombia
- [15] Yunus Emre Aslan. A FRAMEWORK FOR THE USE OF WIRELESS SENSOR NETWORKS IN FOREST FIRE DETECTION AND MONITORING. A thesis submitted to the department of computer engineering and the institute of engineering and science of bilkent university.
- [16] Margaret Richardson Ansah, Yan Shu, Cao Qing Hua. High Temperature Wireless Sensor Network Monitoring System for Coalmine Fire. School of Computer Science and Telecommunication Engineering, Jiangsu University, Zhenjiang, 212013.
- [17] Byungrak Son, Yong-sork Her, and Jung-Gyu Kim. A Design and Implementation of Forest-Fires Surveillance System based on Wireless Sensor Networks for South Korea Mountains. † School of Computer and Communication Engineering, Daegu University, Daegu, 712-714 Korea.
- [18] NOAA SATELLITE INFORMATION SYSTEM. Advanced Very High Resolution Radiometer – AVHRR available: <http://noaasis.noaa.gov/NOAASIS/ml/avhrr.html>
- [19] NASA, “MOIDS, National Aeronautics and space administration,” 1999, Disponible: <http://modis.gsfc.nasa.gov/>.
- [20] Ariadne Gallardo Figueroa. EL OXÍGENO Y LOS PROCESOS DE OXIDACIÓN DE MATERIALES. ENTREVISTA A LA DOCTORA LUCIEN VELEVA. Disponible en: <http://casanchi.com/ref/oxigeno01.pdf>
- [21] Oxford Dictionaries. Language matters. Disponible en: [/www.oxforddictionaries.com/](http://www.oxforddictionaries.com/)
- [22] Chien-Liang Fok, Gruia-Catalin Roman, and Chenyang Lu. Mobile agent middleware for sensor networks: an application case study. Washington University in Saint Louis Saint Louis, Missouri 63130.
- [23] Gaurav Kumar & Pradeep Kumar Bhatia. A Detailed Review of Feature Extraction in Image Processing Systems. CONFERENCE PAPER · FEBRUARY 2014. IEEE Xplore.

- [24] Drashti Jasani et al, / (IJCSIT). Review of Shape and Texture Feature Extraction Techniques for Fruits. International Journal of Computer Science and Information Technologies, Vol. 6 (6), 2015.
- [25] Renjini, Jyothi R. Wavelet Based Image Analysis: A Comprehensive Survey. International Journal of Computer Trends and Technology (IJCTT) – Volume 21 Number 3 – Mar 2015.
- [26] Ryszard S. Choras. Image Feature Extraction Techniques and Their Applications for CBIR and Biometrics Systems. INTERNATIONAL JOURNAL OF BIOLOGY AND BIOMEDICAL ENGINEERING.
- [27] Seyyid Ahmed Medjahed. A Comparative Study of Feature Extraction Methods in Images Classification. I.J. Image, Graphics and Signal Processing, 2015, 3, 16-23 Published Online February 2015 in MECS.
- [28] E. Saber, A.M. Tekalp, "Integration of color, edge and texture features for automatic region-based image annotation and retrieval," Electronic Imaging, 7, pp. 684–700, 1998.
- [29] Khoulood Meskaldji, Samia Boucherkha. Color Quantization and its Impact on Color Histogram Based Image Retrieval. 2009 First International Conference on Networked Digital Technologies. IEEE Xplore Digital Library.
- [30] Noah Keen. Color Moments February 10, 2005. Instructor Dr. Bob Fisher. The University of Edinburgh School of Informatics. Disponible en: [http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL\\_COPIES/AV0405/KEEN/av\\_as2\\_nkeen.pdf](http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/AV0405/KEEN/av_as2_nkeen.pdf)
- [31] Roshini Johri. Color Descriptors from Compressed Images. The University of Edinburgh School of Informatics. Disponible en: [http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL\\_COPIES/AV0910/johri.pdf](http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/AV0910/johri.pdf)
- [32] Pearson, K. (1895). "Contributions to the Mathematical Theory of Evolution. II. Skew Variation in Homogeneous Material". Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 186: 343–414. Bibcode:1895RSPTA.186..343P. doi:10.1098/rsta.1895.0010.
- [33] Grimmett, Geoffrey and Stirzaker, David (2009). Probability and Random Processes. Oxford, England: Oxford University Press. ISBN 978 0 19 857222 0.
- [34] Probability and Statistics. Moments. Central Moment. Wolfram Math World. Disponible en: <http://mathworld.wolfram.com/CentralMoment.html> Tomado el 03/03/2016.
- [35] Xiaoyin Duanmu. "Image Retrieval Using Color Moment Invariant". Seventh International Conference on Information Technology, pp. 200-203, 2010.
- [36] Shapiro and Stockman. CSEP 576: Computer Vision. Computer Science and Engineering. University of Washington Courses and Course Webs.

- [37] Dong ping Tian. A Review on Image Feature Extraction and Representation Techniques. International Journal of Multimedia and Ubiquitous Engineering Vol. 8, No. 4, July, 2013. Institute of Computer Software and Institute of Computational Information Science, Baoji University of Arts and Sciences, Baoji, Shaanxi, 721007, China.
- [38] V. Shiv Naga Prasad, Justin Domke. Gabor Filter Visualization. University of Maryland.
- [39] B. S. Manjunath, Jens-Rainer Ohm. Color and Texture Descriptors. IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, VOL. 11, NO. 6, JUNE 2001.
- [40] Zhihu Huang, Jinsong Leng. Analysis of Hu's Moment Invariants on Image Scaling and Rotation. Edith Cowan University Research Online ECU Publications Pre. 2011.
- [41] S. E. Grigorescu, N. Petkov and P. Kruizinga, "Comparison of texture features based on Gabor filters", IEEE TIP, vol. 11, no. 10, (2002), pp. 1160-1167.
- [42] D. Zhang and G. Lu, "Review of shape representation and description techniques", Pattern Recognition, vol. 37, no. 1, (2004), pp. 1-19.
- [43] M. Yang, K. Kpalma and J. Ronsin. "A survey of shape feature extraction techniques", Pattern Recognition, (2008), pp. 43-90.
- [44] Anne Solberg. INF 4300 – Hough transform. Universidad de Oslo
- [45] Y. Bengio, "Learning deep architectures for AI", Foundations and Trends in Machine Learning, Vol. 2, No. 1, 2009, pp. 30-47, [En línea], Disponible: [http://www.iro.umontreal.ca/~bengioy/papers/ftml\\_book.pdf](http://www.iro.umontreal.ca/~bengioy/papers/ftml_book.pdf).
- [46] M. Shu, A. Fyshe, "Sparse Autoencoders for Word Decoding from Magnetoencephalography", Proceedings of the third NIPS Workshop on Machine Learning and Interpretation in NeuroImaging, (MLINI), 2013, [En línea], Disponible: <http://www.cs.cmu.edu/~afyshe/papers/SparseAE.pdf>
- [47] A. Makhzani, F. Brendan, "k-Sparse Autoencoders", arXiv preprint arXiv:1312.5663, Marzo 2014, [En línea], Disponible: <http://arxiv.org/pdf/1312.5663.pdf>
- [48] M. Matusugu, K. Mori, Y. Mitari, Y. Kaneda, "Subject independent facial expression recognition with robust face detection using a convolutional neural network", Elsevier, 2003, Neural Networks 16. 555–559. [Enlínea], Disponible: [http://www.iro.umontreal.ca/~pift6080/H09/documents/papers/sparse/matsugo\\_etal\\_face\\_expression\\_conv\\_nnet.pdf](http://www.iro.umontreal.ca/~pift6080/H09/documents/papers/sparse/matsugo_etal_face_expression_conv_nnet.pdf).
- [49] D. Scherer, A. Muller, S. Behnke, "Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition", ICANN, Septiembre 2010. [En línea], Disponible: [http://ais.uni-bonn.de/papers/icann2010\\_maxpool.pdf](http://ais.uni-bonn.de/papers/icann2010_maxpool.pdf).

- [50] G. Hinton, R. Salakhutdinov, "Reducing the Dimensionality of Data with Neural Networks", Science, Julio 2006, Vol. 313, pp. 504-507. [En línea], Disponible: <https://www.cs.toronto.edu/~hinton/science.pdf>
- [51] H. Larochelle, Y. Bengio, "Classification using discriminative restricted Boltzmann machines", JLMR, Vol. 13, Marzo 2012, pp: 643-669, [En línea ], Disponible: <http://www.jmlr.org/papers/v13/>
- [52] Miguel. Carreira, G. Hinton, " On Contrastive Divergence Learning", AI & Statistics, Enero 2005, [En línea], Disponible: [http://www.cs.toronto.edu/~hinton/absps/ijcnn\\_constraints.pdf](http://www.cs.toronto.edu/~hinton/absps/ijcnn_constraints.pdf)
- [53] K. Seetharaman & Bachala Shyam Kumar. Texture and Color Features based Color Image Retrieval using Canonical Correlation.
- [54] Jing Huang, S Ravi Kumar, Mandar Mitraz, Wei-Jing, Zhux Ramin Zabih. Image Indexing Using Color Correlograms. Cornell University Ithaca, NY 14853
- [55] Anibal de los santos Y. Fundamentos Visuales II Teoría del Color. Disponible: [www.anibaldesigns.com](http://www.anibaldesigns.com)
- [56] Color iQC and Color iMatch Color Calculations Guide. Version 8.0. 30 July 2012 Revision 1.0. Disponible en: [http://www.xrite.com/documents/literature/en/09\\_color\\_calculations\\_en.pdf](http://www.xrite.com/documents/literature/en/09_color_calculations_en.pdf)
- [57] Hunter Lab. Applications Note. Hunter L, a, b Versus CIE 1976 L\*a\*b. February.2001, Vol 13. Disponible en: <http://elynxsdk.free.fr/ext-docs/ColorSpace/HunterLab.pdf>.
- [58] Hunter Lab. Applications Note. Hunter Lab Color Scale. August 1-15, 1996, Vol. 8. Disponible en: [https://support.hunterlab.com/hc/en-us/article.../an08\\_96a2.pdf](https://support.hunterlab.com/hc/en-us/article.../an08_96a2.pdf)
- [59] Yong Man Ro, Munchurl Kim, Ho Kyung Kang, B.S. Manjunath, MPEG-7 Homogeneous Texture Descriptor. Disponible en: <https://vision.ece.ucsb.edu/sites/vision.ece.ucsb.edu/files/publications/01ETRImpeg7.pdf>
- [60] Karina Viridiana Sánchez López. Descriptores de imágenes digitales con momentos de Zernike. UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO.
- [61] Eduardo Cano, Rubén Salcedo & Gustavo Soto. Análisis de principios y aplicaciones de la transformada wavelet. Universidad Nacional de Catamarca.
- [62] Minrong Zhao , Qiao Chai , and Shanwen Zhang. A Method of Image Feature Extraction Using Wavelet Transforms.
- [63] Dipalee Gupta , Siddhartha Choubey. Discrete Wavelet Transform for Image Processing. International Journal of Emerging Technology and Advanced Engineering.

- [64] Yoshua Bengio, Aaron Courville, Ian Goodfellow. . Deep Learning. Book in preparation for MIT press Chapter 9: Convolutional Networks. Disponible: [www.deeplearningbook.org](http://www.deeplearningbook.org).
- [65] Erick Dasaev de la Rosa Montero. El aprendizaje profundo para la identificación de sistemas no lineales. Unidad Zacatenco Departamento de Control Automático. México D.F. Agosto 2014.
- [66] Deep Learning Tutorial. LISA lab, University of Montreal. Chapter 6. Convolutional Neural Networks (LENET). Release 0.1 September 01, 2015. Disponible: [www.deeplearning.net/tutorial](http://www.deeplearning.net/tutorial).
- [67] Ricardo Gutierrez-Osuna. CSCE 666 Pattern Analysis course. L13: Cross-validation. Texas A&M University.
- [68] Andrej Karpathy, Justin Johnson. CS231n: Convolutional Neural Networks for Visual Recognition. Module 2: Convolutional Neural Networks. Stanford University.
- [69] Christopher M. Bishop. "Pattern Recognition and Machine Learning" Book. Chapter 4. Linear Models for Classification. Springer.
- [70] Dr. Richard E. Turner. Lecture 14: Convolutional neural networks for computer vision. University of Cambridge November 20, 2014. Disponible: <http://learning.eng.cam.ac.uk/Public/Turner/Teaching>.
- [71] Xiaodong Zhou. Supervised Deep Learning for Multi-Class Image Classification. Sección 2 y 3.
- [72] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. University of Toronto.
- [73] Michael A. Nielsen, "Neural Networks and Deep Learning", Determination Press, 2015. Chapter 6: Deep Learning. Enero/2016. Disponible: <http://neuralnetworksanddeeplearning.com/chap6.html>.
- [74] Matthew D. Zeiler, Rob Fergus. "Visualizing and Understanding Convolutional Networks". Dept. of Computer Science, Courant Institute, New York University.
- [75] Jason Yosinski, Jeff Clune, Yoshua Bengio, Hod Lipson. How transferable are features in deep neural networks? In Advances in Neural Information Processing Systems 27 (NIPS '14), NIPS Foundation, 2014.
- [76] Julien Mairal, Piotr Koniusz, Zaid Harchaoui, Cordelia Schmid. Convolutional Kernel Networks. Cornell University Library.
- [77] Yoshua Bengio. Practical Recommendations for Gradient-Based Training of Deep Architectures. Neural Network Hyper-Parameters. Version 2, Sept 16, 2012.

- [78] Andrew Ng. Part VII. Regularization and Model selection. CS229 Lecture Notes. Stanford University.
- [79] Scikit-learn. User guide. 3.3 Model evaluation: quantifying the quality of predictions. Disponible: [http://scikit-learn.org/stable/modules/model\\_evaluation.html#precision-recall-f-measure-metrics](http://scikit-learn.org/stable/modules/model_evaluation.html#precision-recall-f-measure-metrics).
- [80] John R. Taylor. An Introduction to Error Analysis. The study of uncertainties in physical measurements. Second Edition. University of Colorado.
- [81] Tomasz Malisiewicz. Supervised Distance Metric Learning. Presented at CMU's Computer Vision. Misc-Read Reading Group. May 9, 2007.
- [82] Rich Caruana, Alexandru Niculescu-Mizil. Data Mining in Metric Space: An Empirical Analysis of Supervised Learning Performance Criteria. Computer Science. Cornell University.
- [83] JCGM 200:2008. International vocabulary of metrology — Basic and general concepts and associated terms (VIM).
- [84] Stephen V. Stehman. Selecting and Interpreting Measures of Thematic Classification Accuracy. Remote Sensing of Environment. ELSEVIER.
- [85] David M W Powers. Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation. School of Informatics and Engineering Flinders University of South Australia.
- [86] ARDrone2, Especificaciones Técnicas, En línea, Disponible: <http://ardrone-2.es/especificaciones-ar-drone-2>.
- [87] V. Victor Podlozhnyuk, "Image Convolution with CUDA", NVIDIA, Junio 2007, En línea, Disponible: <http://igm.univ-mlv.fr/~biri/Enseignement/MII2/Donnees/convolutionSeparable.pdf>.
- [88] F. Mesa, M. Arias, "Convolución paralela con GPU", Journal of Educational Experiences on Computer Engineering, Numero 5, Mayo 2015, pp. 139-155, En línea, Disponible: [http://digibug.ugr.es/bitstream/10481/36575/1/T11\\_N5\\_Revista\\_EAIC\\_2015.pdf](http://digibug.ugr.es/bitstream/10481/36575/1/T11_N5_Revista_EAIC_2015.pdf).
- [89] Nvidia, GeForce GTX TITAN, En línea, Disponible: <http://www.nvidia.es/object/geforce-gtx-titan-es.html#pdpContent=2>.
- [90] Python, "Documentación Python 2.7", En línea, Disponible: <https://docs.python.org/2/whatsnew/2.7.html>.
- [91] Nvidia, Procesamiento Paralelo CUDA, En línea, Disponible: <http://www.nvidia.es/object/cuda-parallel-computing-es.html>.



- [92] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde, Y. Bengio, "Theano: A CPU and GPU Math Compiler in Python", SCIPY 2010, En línea, Disponible: [http://www.iro.umontreal.ca/~lisa/pointeurs/theano\\_scipy2010.pdf](http://www.iro.umontreal.ca/~lisa/pointeurs/theano_scipy2010.pdf).
- [93] Eclipse, Guía del usuario, En línea, Disponible: <http://help.eclipse.org/mars/index.jsp?nav=%2F0>.
- [94] PyDev, Manual, En línea, Disponible: <http://www.pydev.org/manual.html>.