



TRANSFER LEARNING EN LA CLASIFICACIÓN BINARIA DE IMÁGENES TÉRMICAS

TRANSFER LEARNING FOR BINARY CLASSIFICATION OF THERMAL IMAGES

Daniel Pérez-Aguilar^{1,*} , Redy Risco-Ramos^{1,*} , Luis Casaverde-Pacherrez¹ 

Recibido: 13-11-2020, Recibido tras revisión: 17-12-2020, Aceptado: 22-02-2021, Publicado: 01-07-2021

Resumen

La clasificación de imágenes térmicas es un aspecto clave en el sector industrial, debido a que suele ser el punto de partida en la detección de fallos en equipos eléctricos. En algunos casos, esta tarea se automatiza mediante el uso de técnicas tradicionales de inteligencia artificial, mientras que en otros, es realizada de manera manual, lo cual puede traer consigo altas tasas de error humano. Este artículo presenta un análisis comparativo entre once arquitecturas de *transfer learning* (AlexNet, VGG16, VGG19, ResNet, DenseNet, MobileNet v2, GoogLeNet, ResNeXt, Wide ResNet, MNASNet y ShuffleNet) mediante el uso de *fine-tuning*, con la finalidad de realizar una clasificación binaria de imágenes térmicas en una red de distribución eléctrica. Para ello, se dispone de una base de datos con 815 imágenes, divididas mediante la técnica tipo hold-out 60-20-20 y validación cruzada con 5-folds, para finalmente analizar su rendimiento mediante el test de Friedman. Luego de los experimentos, se obtuvieron resultados satisfactorios con exactitudes superiores a 85 % en diez de las arquitecturas previamente entrenadas. Sin embargo, la arquitectura que no se entrenó previamente presentó una exactitud baja; concluyéndose que la aplicación de *transfer learning* mediante el uso de arquitecturas previamente entrenadas es un mecanismo adecuado en la clasificación de este tipo de imágenes, y representa una alternativa confiable frente a técnicas tradicionales de inteligencia artificial.

Palabras clave: imágenes térmicas, *fine-tuning*, preentrenamiento, test de Friedman, *transfer learning*

Abstract

The classification of thermal images is a key aspect in the industrial sector, since it is usually the starting point for the detection of faults in electrical equipment. In some cases, this task is automated through the use of traditional artificial intelligence techniques, while in others, it is performed manually, which can lead to high rates of human error. This paper presents a comparative analysis between eleven transfer learning architectures (AlexNet, VGG16, VGG19, ResNet, DenseNet, MobileNet v2, GoogLeNet, ResNeXt, Wide ResNet, MNASNet and ShuffleNet) through the use of fine-tuning, in order to perform a binary classification of thermal images in an electrical distribution network. For this, a database with 815 images is available, divided using the 60-20-20 hold-out technique and cross-validation with 5-Folds, to finally analyze their performance using Friedman test. After the experiments, satisfactory results were obtained with accuracies above 85 % in 10 of the previously trained architectures. However, the architecture that was not previously trained had low accuracy; with this, it is concluded that the application of transfer learning through the use of previously trained architectures is a proper mechanism in the classification of this type of images, and represents a reliable alternative to traditional artificial intelligence techniques.

Keywords: fine-tuning, Friedman test, pre-training, thermal images, transfer learning

^{1,*}Laboratorio de Sistemas Automáticos de Control, Universidad de Piura, Perú.

Autor para correspondencia: ✉: daniel.perez.a@alum.udel.edu.pe

Forma sugerida de citación: Pérez-Aguilar, D.; Risco-Ramos, R. y Casaverde-Pacherrez, L. (2021). «Transfer learning en la clasificación binaria de imágenes térmicas». INGENIUS. N.º 26, (julio-diciembre). pp. 71-86. DOI: <https://doi.org/10.17163/ings.n26.2021.07>.

1. Introducción

Actualmente la inteligencia artificial (IA) es una disciplina en auge que ha redefinido muchos de los procesos que se realizan en la industria, presentando aplicaciones muy diversas, las cuales abarcan el reconocimiento de objetos mediante visión artificial, reconocimiento y síntesis de voz, comprensión lectora, sistemas de traducción, comprensión del lenguaje, etc. [1]. La IA no es un término nuevo, pues existe desde hace muchos años; sin embargo, lo que ha cambiado en las últimas épocas es la potencia computacional, con la cual se pueden computar modelos mucho más complejos en un menor tiempo [2]. Se la define como un conjunto de algoritmos cuyo propósito es crear máquinas que emulen las capacidades de los seres humanos. Visto de otra manera, es un software que puede ser entrenado para reconocer patrones y realizar predicciones, en algunos casos de manera más precisa que los seres humanos [3]. Dentro de la IA, encontramos términos como el *machine learning* (ML) y el *deep learning* (DL) [4].

machine learning (ML) o aprendizaje automático, es una rama de la IA que busca generalizar comportamientos de un conjunto de datos de entrada, es decir, que su objetivo es predecir comportamientos futuros basados en encontrar patrones dentro de grandes conjuntos de datos [5]. A su vez, el DL o aprendizaje profundo, forma parte del *machine learning* [6], cuyo objetivo es que los sistemas imiten el comportamiento y el razonamiento de las personas automáticamente; en otras palabras, que el ser humano esté involucrado lo menos posible en el proceso. Este objetivo se basa en el uso de redes neuronales artificiales (ANN), las cuales simulan la sinapsis del cerebro humano [7]. En la Figura 1 se puede apreciar la estructura tradicional de una red neuronal artificial, en la que se incluyen la capa de entrada, capas ocultas y la capa de salida.

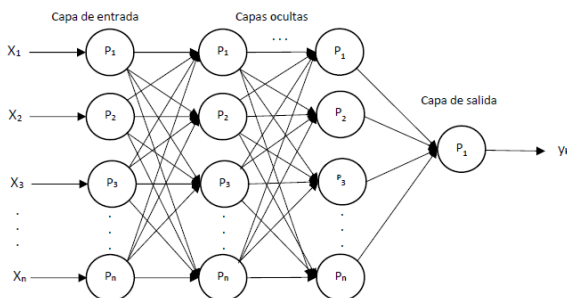


Figura 1. Estructura de una red neuronal artificial [8]

Además, se debe señalar que dentro de las ANN encontramos a las redes neuronales convolucionales (CNN), que son un tipo avanzado y de alto potencial del modelo clásico de red neuronal artificial, diseñadas para abordar problemas de mayor complejidad, y usadas generalmente en la clasificación de imágenes [9].

En relación con la separación del set de datos, la técnica tipo *hold-out* [10] es una de las más utilizadas, y consiste en realizar una división de los datos en tres subconjuntos: el 60 % destinado a entrenamiento, el 20 % a validación y el 20 % restante a las pruebas del modelo, tal como se observa en la Figura 2. Sin embargo, este tipo de técnicas no puede considerarse suficiente para evaluar el rendimiento de los modelos, por lo que la bibliografía sugiere aplicar una validación cruzada con *k-folds* [11] a través de la división del set de datos aleatoriamente en *k* subconjuntos, de los cuales *k* - 1 se utilizan para entrenar el modelo, y uno para validarlo. Este mecanismo se debe repetir *k* veces en cada iteración, empleando distintos subconjuntos de validación, tal como se aprecia en la Figura 3. Finalmente, se recomienda realizar una comparación estadística de los resultados de cada modelo [12], en donde entran a tallar técnicas paramétricas como el análisis ANOVA, o no paramétricas como el test de Friedman [13].

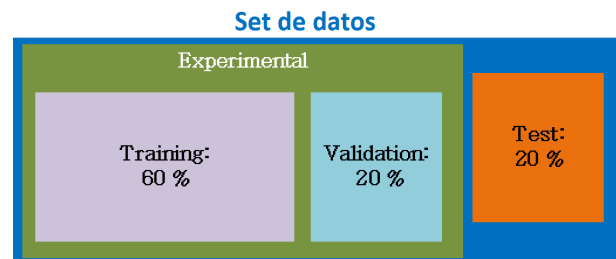


Figura 2. Separación tipo hold-out

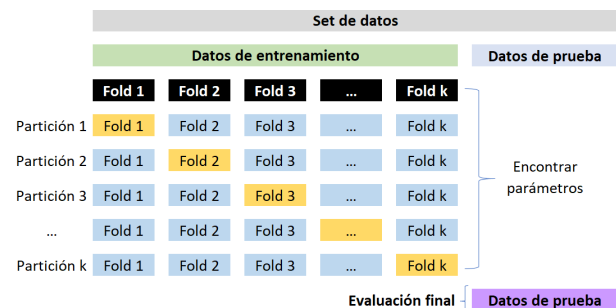


Figura 3. Validación cruzada con k-folds

Algunas investigaciones muestran que la aplicación de inteligencia artificial es útil en el sector eléctrico a través de la utilización de imágenes térmicas para automatizar la clasificación de las mismas [14]. Este tipo de imágenes son capturadas mediante el uso de cámaras infrarrojas, por lo que entra a tallar otro término relevante: la termografía, una técnica que permite captar la radiación infrarroja del espectro electromagnético, cuya ventaja principal es la no necesidad de tener contacto físico con el objeto o pieza a estudiar, con lo que, el operar de la maquinaria no se detiene [15].

La Figura 4 muestra un ejemplo de una imagen térmica de un equipo de alta tensión capturada desde tierra.

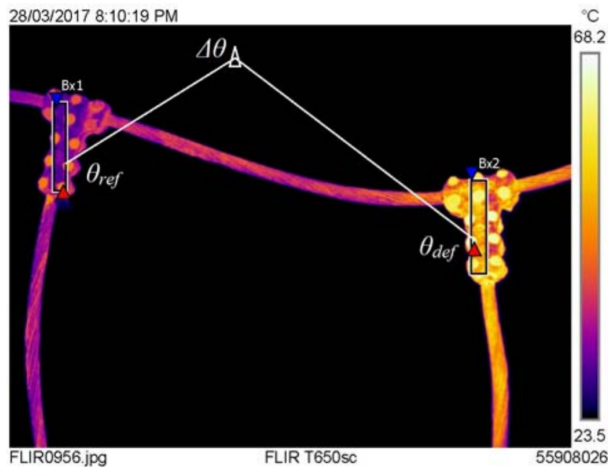


Figura 4. Imagen térmica de un equipo de alta tensión [15]

A continuación, se exponen las investigaciones más relevantes del uso de técnicas tradicionales de inteligencia artificial en la clasificación de imágenes térmicas. Un claro ejemplo es la propuesta de un sistema de reconocimiento automático para la clasificación de imágenes termográficas de una red de distribución de energía eléctrica [16], en donde se implementó una CNN y el algoritmo JSEG o segmentación J, el cual consiste en una reducción del número de colores y la fusión de los mismos basado en la similitud de las regiones de las imágenes [17]. Al igual que una investigación realizada en el Departamento de Tecnología de Chongqing, China [18], donde se aborda la visión por computadora mediante el uso de imágenes térmicas infrarrojas capturadas sin perturbar el funcionamiento de las subestaciones eléctricas. Para ello entrenaron un perceptrón multicapa o *multilayer perceptron* (MLP), el cual es un tipo de red neuronal artificial formado por diversas capas de neuronas intermedias u ocultas, utilizado para solucionar problemas que no se pueden separar linealmente [19].

En [20] se propone un enfoque semiautomático para evaluar la condición térmica de las instalaciones eléctricas de un edificio mediante el análisis de imágenes infrarrojas, utilizando un perceptrón multicapa (MLP) y el análisis de componentes principales (PCA); este último es una técnica estadística que tiene como finalidad simplificar la complejidad de la muestra mediante la selección o extracción de las características más representativas de los datos de entrada [21]; mientras que en [22], se expone un método de diagnóstico inteligente para clasificar diferentes condiciones de equipos eléctricos utilizando datos obtenidos de imágenes infrarrojas mediante el uso del algoritmo K-means, el cual se encarga de agrupar las imágenes de los equipos eléctricos

a través del encuentro y clasificación de clústeres o grupos con características similares [23].

Las investigaciones mencionadas se centran en entrenar un modelo desde cero o de manera tradicional para un escenario específico; sin embargo, actualmente, existen técnicas que facilitan este proceso como el *transfer learning* (TF), el cual forma parte del *deep learning* y consiste en utilizar una red preentrenada, es decir, reusar la arquitectura y los pesos de un modelo entrenado con grandes volúmenes de datos de entrada y aplicarlos a escenarios distintos con otros set de datos, buscando realizar clasificaciones más rápidas y utilizando menor carga computacional [24]. Un ejemplo de las bases de datos utilizadas para entrenar estos modelos es el caso de ImageNet, la cual contiene más de catorce millones de imágenes [25].

Uno de los paradigmas del *transfer learning* es el *fine-tuning* del modelo, el cual busca adaptarlo a un nuevo dominio de aplicación [26]; para ello se toma el modelo previamente entrenado y se varían algunos parámetros como la tasa de aprendizaje, teniendo como objetivo lograr mejoras significativas en las predicciones [27].

La revisión bibliográfica revela que existen diversas aplicaciones de *transfer learning* en sectores como (i) salud, mediante la clasificación de patologías en imágenes neurológicas [28], detección de objetos como pistolas o armas blancas en imágenes de rayos X [29] o cervicales [30], entre otros. (ii) En la agroindustria, como se observa en [31], donde se presenta una comparación de las arquitecturas de *transfer learning* ResNet, GoogLeNet, VGG16, AlexNet y DenseNet, con la finalidad de clasificar un conjunto de datos que contiene imágenes de flores, demostrando que la arquitectura VGG16 previamente entrenada obtiene niveles de exactitud superiores a las demás. (iii) O en el sector alimenticio, como se expone en [32], donde se indica que las CNN son las técnicas de clasificación de imágenes más utilizadas, por lo que centran su investigación en la clasificación de alimentos con la finalidad de obtener un estilo de vida más saludable; para ello utilizan una base de datos de 500 imágenes, además de las arquitecturas previamente entrenadas VGG16, VGG19, ResNet e InceptionV3, siendo esta última la que obtuvo mejores resultados.

Sin embargo, al estudiar la aplicación de imágenes térmicas y técnicas de *transfer learning* en el sector eléctrico, nos encontramos con una escasez de las mismas. Siendo algunas de las más representativas el caso de [33], donde se propone un mecanismo de clasificación de imágenes térmicas de los sistemas de rodamientos de rotores; para ello modifican una red neuronal convolucional con el uso de *transfer learning*, no obstante, no especifica la arquitectura de TF utilizada. O como se detalla en [34], en donde se busca automatizar la supervisión del estado de la maquinaria industrial mediante el uso de imágenes térmicas y una CNN, indicando

que una desventaja de esta última es la necesidad de contar con grandes volúmenes de datos para su entrenamiento, por lo que proponen el uso de la arquitectura VGG16 como método de reutilización de capas de la red neuronal.

Lo señalado en párrafos anteriores evidencia que las ventajas del uso de *transfer learning* no se están explotando en su totalidad en el sector eléctrico, ya que no existe un estudio que aplique diversas arquitecturas con el mismo set de datos, por lo que el objetivo del presente estudio es proponer una alternativa al uso tradicional de técnicas de inteligencia artificial mediante el análisis de once arquitecturas de *transfer learning* y el paradigma de auto-tuning, aplicadas a la clasificación binaria de imágenes térmicas en una red de distribución eléctrica.

2. Materiales y métodos

La Figura 5 representa la metodología seguida en la realización de la investigación, la cual parte de la recolección de datos de campo mediante la captura de imágenes térmicas; posteriormente se realiza el diseño de una arquitectura base en la que se incluyen distintas arquitecturas de *transfer learning* entrenadas previamente, cada una de ellas es entrenada y probada para finalmente comparar los resultados obtenidos. Todo este proceso se realizó con la ayuda del servicio en línea Google Colaboratory, a través del uso de Jupyter Notebooks [35].

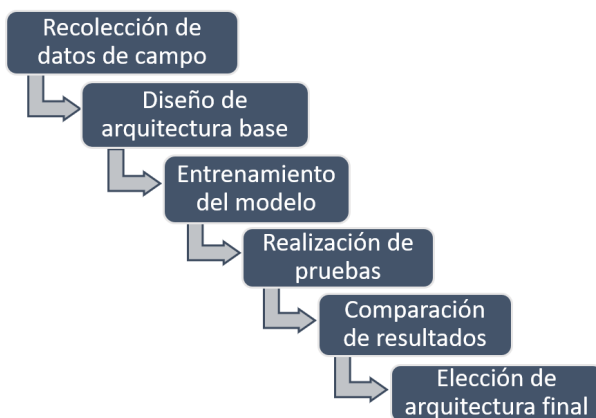


Figura 5. Metodología

2.1. Base de datos

La base de datos utilizada en este trabajo corresponde a 815 imágenes térmicas, pertenecientes a una empresa de distribución eléctrica peruana, clasificadas en dos categorías, subestaciones eléctricas y líneas de transmisión. Este conjunto de datos es superior en tamaño al utilizado en [18], [20], [22], [36] y [37]. Las imágenes

fueron capturadas mediante la cámara infrarroja TP8S, cuyas especificaciones se aprecian en la Tabla 1.

Tabla 1. Especificaciones técnicas de la cámara TP8S [38]

Característica	Descripción
Tipo de detector	FPA (384 × 288 píxeles, 35 m)
Rango espectral	8-14 m
Sensibilidad térmica	0.08 °C a 130 °C
Campo de visión	22° × 16° / 35 mm
Enfoque electrónico	Automático o motorizado
Zoom	Zoom continuo de ×1 a ×10

El procedimiento que la empresa de distribución eléctrica ha venido ejecutando incluye cinco fases que se detallan a continuación y se grafican en la Figura 6. (i) En primer lugar, se contrata a una empresa externa para la captura de imágenes térmicas, específicamente líneas de transmisión y subestaciones eléctricas, esto se realiza de forma semanal o mensual. (ii) La empresa de servicios entrega todas las imágenes a un especialista certificado en análisis de imágenes térmicas. (iii) El especialista debe clasificar manualmente las imágenes y dividir las entre líneas de transmisión o equipos de subestaciones eléctricas, ya que requieren diferentes tipos de análisis. (iv) Luego el especialista procede al análisis de cada imagen y determina, de acuerdo con su conocimiento y experiencia, si en cada una hay evidencia de fallos presentada por la detección de algún punto caliente. (v) Finalmente, si el especialista detecta un punto caliente, prepara el informe respectivo y luego toma las medidas correctivas apropiadas.

El presente estudio se centró en automatizar la fase 3 del proceso mediante el uso de técnicas de *transfer learning*.

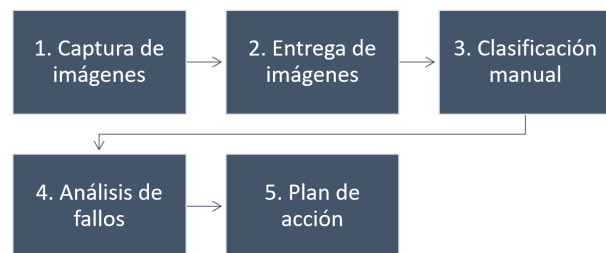


Figura 6. Proceso de clasificación de imágenes térmicas

Las imágenes presentan una resolución de 384 × 288 píxeles. En la Figura 7 se aprecian algunos ejemplos de las imágenes típicas de equipos de líneas de transmisión, y en Figura 8 de subestaciones eléctricas las cuales representan las dos clases del modelo.



Figura 7. Imagen típica de equipos de líneas de transmisión

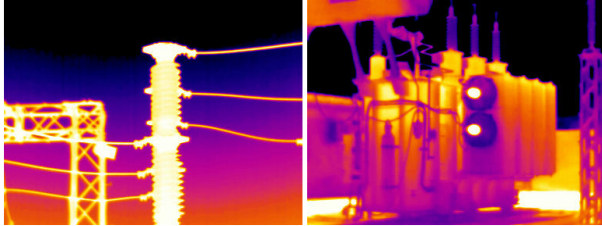


Figura 8. Imagen típica de equipos de subestaciones eléctricas

2.2. Set de datos

Para la realización de los experimentos se utilizaron 815 imágenes térmicas, divididas en tres conjuntos de datos con una división 60-20-20, conocido como una separación tipo *hold-out* (ver Figura 2). El 60 % de las imágenes se consideraron como datos de entrenamiento y el 20 % de validación, mientras que el 20 % restante, se incluye en los datos de prueba del modelo (ver Tabla 2), es decir, que este es entrenado y validado paralelamente para finalmente ser probado con imágenes nuevas que no han sido consideradas con anterioridad. En [39] se indica que los modelos que generalizan adecuadamente muestran métricas de exactitud y pérdida similares en el entrenamiento y validación, evitándose el sobreajuste.

Tabla 2. Distribución de conjuntos de datos

Clase	Entrenamiento	Validación	Prueba
Línea	206	68	68
Subestación	283	95	95
TOTAL	489	163	163

2.3. Arquitecturas

En el estudio se consideraron once arquitecturas de modelos previamente entrenados mediante el uso del paquete TorchVision, el cual forma parte de PyTorch, que es una biblioteca de aprendizaje automático y de código abierto; ya que como se indica en [40], los modelos en PyTorch son más rápidos y fáciles de implementar y entrenar. Las arquitecturas utilizadas fueron:

2.3.1. AlexNet

Red neuronal convolucional compuesta por ocho capas [41], cinco de las cuales son *max-pooling*, y las tres restantes, *fully connected*. Esta arquitectura fue entrenada con la función de activación ReLU (*Rectified Linear Units*) y la base de datos ImageNet. Como se observa en la Figura 9, la entrada de la red son las imágenes de 224×224 píxeles, las cuales se van transformando en cada una de las capas, hasta obtener como salida, la clasificación de mil categorías.

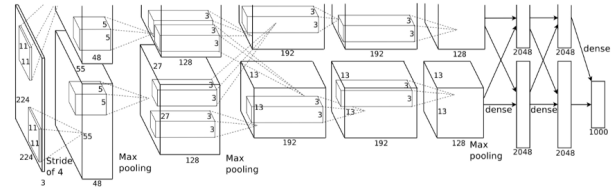


Figura 9. Arquitectura AlexNet [41]

2.3.2. VGG16

Red compuesta por 16 capas y que también fue entrenada con la base de datos ImageNet [42], suponiendo mejoras en relación con la arquitectura AlexNet puesto que reemplaza los grandes filtros de los kernels por un conjunto de filtros de tamaño 3×3 . La arquitectura de VGG16 se aprecia en la Figura 10.

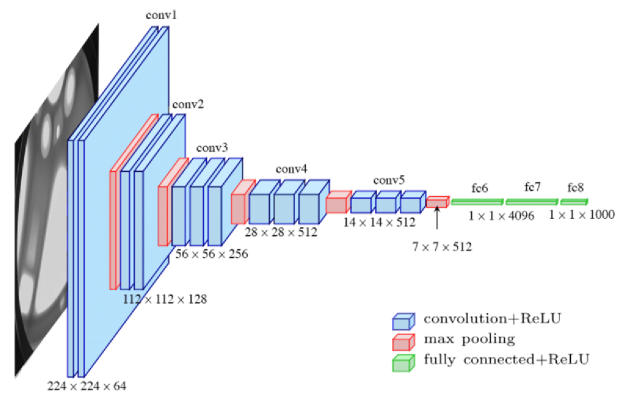


Figura 10. Arquitectura VGG16 [42]

2.3.3. VGG19

Red neuronal convolucional compuesta por 16 capas convolucionales [43], tres fully-connected, cinco Max-Pool y una SoftMax, con un aproximado de 143 millones de parámetros. La arquitectura de VGG19 se aprecia en la Figura 11.

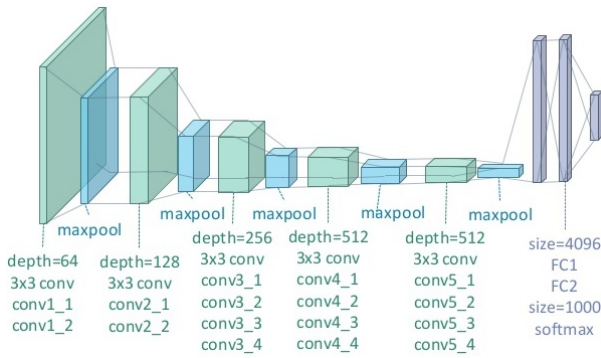


Figura 11. Arquitectura VGG19 [43]

2.3.4. ResNet

Arquitectura que busca que el incremento de capas se realice de manera distinta a la tradicional [44], por lo que agrega una conexión residual con una capa identidad, la cual pasa a la siguiente capa de manera directa mejorando considerablemente el entrenamiento del modelo. Un bloque tradicional de la arquitectura ResNet se observa en la Figura 12.

2.3.5. DenseNet

CNN en la que cada capa obtiene entradas adicionales de todas las capas anteriores y pasa sus propios mapas de características a todas las capas posteriores [45], es decir, que se concatenan cada una de las salidas de las capas anteriores con las posteriores, buscando contar con menor cantidad de parámetros y una mayor exactitud que redes como ResNet. La arquitectura de DenseNet se aprecia en la Figura 13.

2.3.6. GoogLeNet

Red neuronal desarrollada por Google con la finalidad de realizar clasificaciones de imágenes [46]. Esta CNN está basada en la arquitectura Inception [47], por lo que usa módulos que dan la posibilidad de elegir entre distintos tamaños de filtro convolucional en cada uno de los bloques. Un ejemplo del módulo inception se aprecia en la Figura 14.

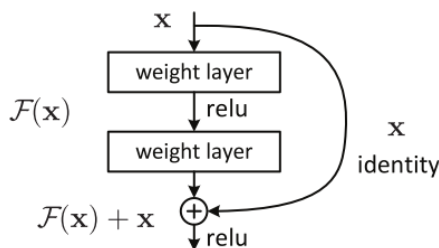


Figura 12. Bloque de la arquitectura ResNet [44]

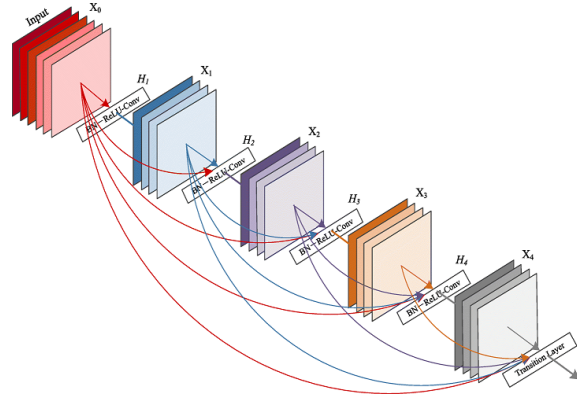


Figura 13. Bloque de la arquitectura DenseNet [45]

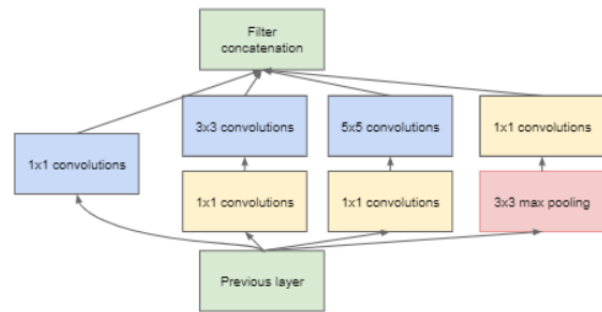


Figura 14. Módulo inception con reducciones de dimensión [48]

2.3.7. MobileNet v2

Se basa en el uso de convoluciones separables en profundidad y utiliza una estructura residual invertida [49], donde la entrada y salida del bloque residual son capas delgadas de cuello de botella opuestas a los modelos residuales tradicionales que utilizan representaciones expandidas en la entrada, tal como se detalla en la Figura 15.

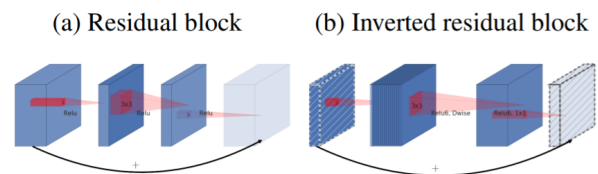


Figura 15. Bloques de la arquitectura MobileNet v2 [49]

2.3.8. ResNeXt

Es una variante de ResNet que busca aumentar el número de caminos o rutas paralelas a la conexión residual [50], es decir, que ResNeXt es una CNN con múltiples ramas, tal como se observa en la Figura 16, la cual muestra un bloque con una cardinalidad de 32.

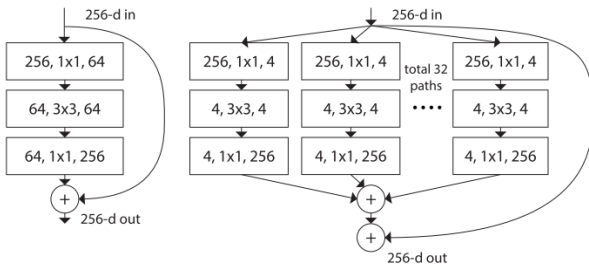


Figura 16. Bloque de la arquitectura ResNeXt [50]

2.3.9. Wide ResNet

Es una red neuronal que presenta una variación a la arquitectura ResNet tradicional [51], disminuyendo la profundidad del modelo y aumentando el ancho de las redes residuales. Los bloques característicos dentro de esta CNN son: basic, bottleneck, basic-wide y wide-dropout. La Figura 17 muestra el detalle de este último.

2.3.10. MNASNet

Es una red neuronal convolucional, que al igual que MobileNet [52], diseñada y optimizada para dispositivos móviles y busca que el modelo obtenga un equilibrio entre latencia y precisión. En la Figura 18 se aprecia un ejemplo del diseño de una capa convolucional con un kernel de 5×5 .

2.3.11. ShuffleNet

Es una CNN que presenta como componente principal una nueva operación de reorganización de canales [53], buscando que la información fluya con mayor facilidad en los mismos. En la Figura 19 se observa una unidad ShuffleNet, que es un elemento central dentro de esta arquitectura.

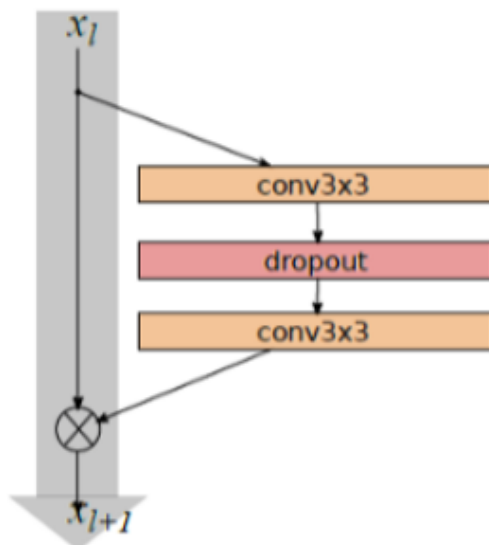


Figura 17. Bloque wide-dropout [51]

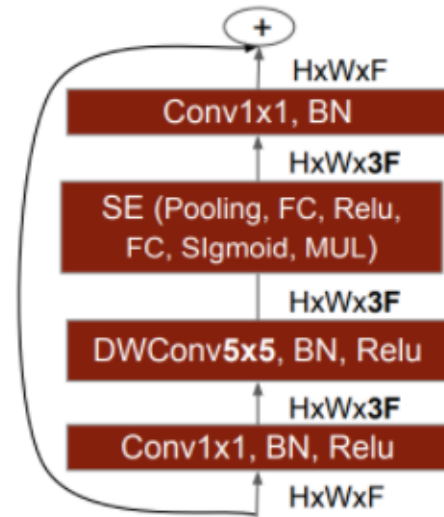


Figura 18. Capa convolucional de MNASNet [52]

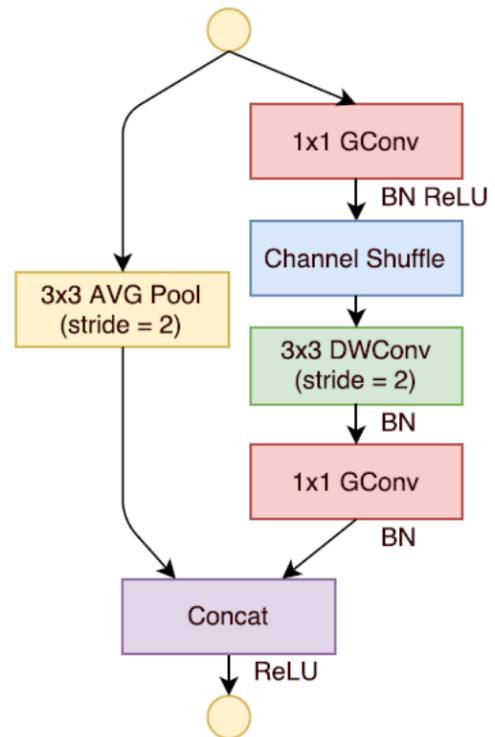


Figura 19. Unidad ShuffleNet [53]

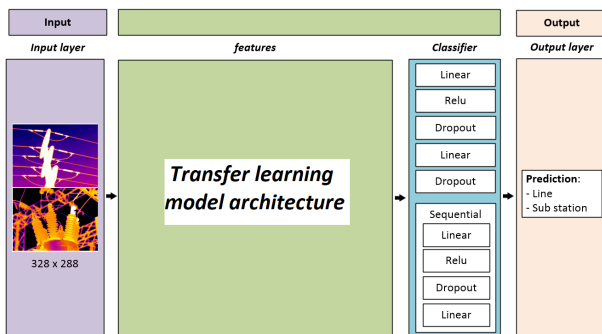
Finalmente, se elaboró la Tabla 3 mediante información proporcionada en el sitio web de torchvision [54]; en ella se resumen las características de estas arquitecturas en relación con el peso en megabytes y número de parámetros que utilizan para el entrenamiento del modelo, resaltando que ShuffleNet es la única arquitectura que actualmente no permite el uso del parámetro de configuración de preentrenamiento, es decir, que su peso es cero megabytes.

Tabla 3. Características de las arquitecturas utilizadas

Nº	Arquitectura	Peso (mb)	Parámetros (millones)
1	AlexNet	233	61.1
2	VGG16	528	138.36
3	VGG19	548	143.67
4	ResNet	230	60.19
5	DenseNet	77.4	20.01
6	GoogLeNet	49.7	13
7	MobileNet v2	13.6	3.5
8	ResNeXt	340	44.55
9	Wide ResNetx	243	126.89
10	MNASNet	16.9	4.38
11	ShuffleNet	0*	7.39

2.4. Modelo

La Figura 20 muestra el diseño de la arquitectura base a utilizar. La primera sección corresponde a la capa de entrada, en la cual se incluyeron las imágenes de 328×288 píxeles. Posteriormente en el apartado denominado «Transfer learning model architecture» se agregaron las once arquitecturas, resaltando que la capa de clasificación de cada una de ellas fue editada para realizar una clasificación binaria ya que originalmente fueron diseñadas para clasificar aproximadamente mil imágenes. Finalmente, se tiene a la capa de salida que corresponde a las predicciones del modelo.

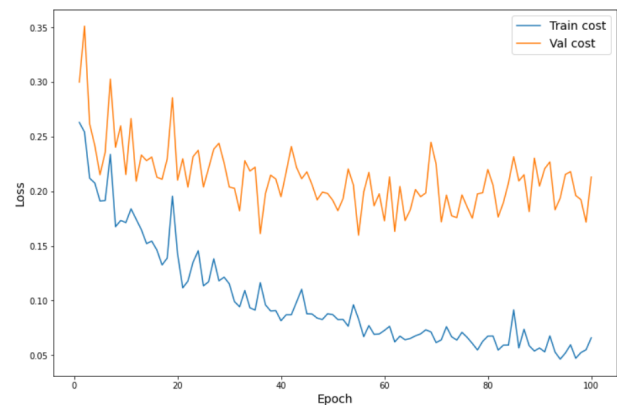
**Figura 20.** Arquitectura base

Debido a que no se encontró investigaciones que comparen este número de arquitecturas aplicadas a la termografía, y con la finalidad de realizar una comparación justa entre los modelos, se consideraron los mismos hiperparámetros en todas las pruebas realizadas. La elección de estos fue tomada sobre la base de experimentos empíricos, tal como se propone en [55–57], teniendo como base los valores de la Tabla 4. Los resultados de los experimentos iniciales arrojaron que existe un punto de quiebre aproximadamente en la época número 20, en donde las curvas de entrenamiento y validación presentan una tendencia de separación,

evidenciándose problemas de *overfitting* o sobreentrenamiento (ver Figura 21). Se concluye que los modelos obtienen mejores resultados según los hiperparámetros detallados en la Tabla 5 y es con estos datos con los que se ejecutaron las demás pruebas.

Tabla 4. Experimentos empíricos de hiperparámetros

Hiperparámetro	Valor
Tasa de aprendizaje	10^{-2} , 10^{-3} , 10^{-4} y 10^{-5}
Imágenes por Batch	16, 32, 64 y 128
Número de épocas	5, 10, 20, 30, 50 y 100
Función de costo	Cross Entropy, Multi Margin Loss y MSE
Optimizador	Adagrad, Adam, Adamax, RMSprop y SGD

**Figura 21.** Experimento con 100 épocas**Tabla 5.** Configuración final de hiperparámetros

Hiperparámetro	Valor
Tasa de aprendizaje	0.0001
Imágenes por Batch	32
Número de épocas	20
Función de costo	Multi Margin Loss
Optimizador	Adam

Los modelos mejoraron sus prestaciones al normalizar los datos de entrada calculando la desviación estándar y media de los conjuntos de datos. Además, a pesar de que el set de datos (815 imágenes) es superior a otras investigaciones como [18], [20], [22], [36] y [37] en las que la máxima cantidad de imágenes utilizada es 500. La bibliografía sugiere como buena práctica la aplicación de técnicas que contribuyan a la mejora de la calidad del entrenamiento, por lo que se utilizó Data Augmentation [58, 59] mediante las transformaciones *random horizontal flip*, *random vertical flip* y *random rotation*. La Figura 22 muestra algunos resultados de las transformaciones utilizadas.

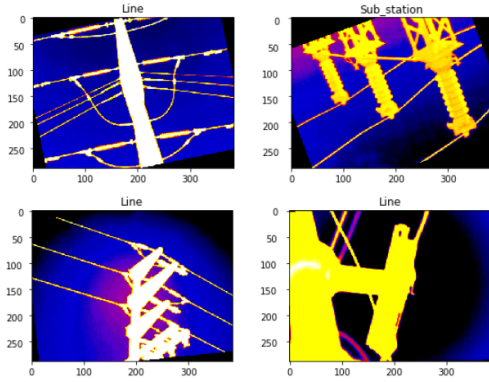


Figura 22. Imágenes con data augmentation

3. Resultados y discusión

Se realizó el entrenamiento de las once arquitecturas tomando los valores de la Tabla 5, obteniendo resultados de exactitud de cada uno de ellos, los cuales se muestran en la Tabla 6. Por otro lado, en la Tabla 7 se observan las tasas de pérdida de cada una de las arquitecturas. En función de estos resultados se aprecia que DenseNet obtiene un mayor grado de exactitud, mientras que VGG16 presenta una tasa de pérdida más baja. Un dato adicional a considerar es que ShuffleNet alcanza los peores resultados debido a que fue la única arquitectura sin entrenamiento previo, evidenciando que las arquitecturas previamente entrenadas, obtienen mejores resultados.

Tabla 6. Exactitud de las arquitecturas

Arquitectura	Entrenamiento	Validación	Prueba
DenseNet	96.52	92.02	98.15
VGG19	93.66	90.18	96.31
Wide ResNetx	94.68	90.18	96.31
MobileNet v2	94.68	88.95	95.70
VGG16	95.91	91.41	95.09
ResNeXt	94.06	92.02	95.09
ResNet	93.66	84.66	94.47
AlexNet	95.50	91.41	93.86
GoogLeNet	95.09	88.95	93.86
MNASNet	71.41	69.93	79.14
ShuffleNet	62.78	68.09	76.68

Tabla 7. Tasa de pérdida de las arquitecturas

Arquitectura	Entrenamiento	Validación	Prueba
VGG16	0.130571	0.195825	0.106733
VGG19	0.175259	0.257767	0.124051
MobileNet v2	0.179914	0.283927	0.124497
DenseNet	0.134446	0.198757	0.126488
AlexNet	0.127838	0.226890	0.134585
GoogLeNet	0.205683	0.242438	0.164706
ResNet	0.220600	0.295843	0.175979
ResNeXt	0.215153	0.241967	0.179627
Wide ResNetx	0.246464	0.271462	0.19046
MNASNet	0.507049	0.529690	0.395779
ShuffleNet	0.647315	0.579723	0.524782

Investigaciones como [60], señalan que las métricas de exactitud y tasa de pérdida presentan un alto grado de subjetividad, por lo que se propone el uso de técnicas estadísticas para evaluar los resultados de las arquitecturas, específicamente F1-score, cuyo mecanismo de cálculo se observa en (1). Precision y Recall se obtienen de (2) y (3) respectivamente, donde TP representa a los verdaderos positivos, FP a los falsos positivos y FN a los falsos negativos. Los resultados se detallan en la Tabla 8, demostrando que la arquitectura VGG16 se ubica en primer lugar con el valor más alto de F1-score, el cual corresponde a 95.11 %.

$$F1_{score} = 2 \times \frac{precision \times recall}{precision + recall} \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Precision = \frac{TP}{TP + FN} \quad (3)$$

Tabla 8. Comparación entre arquitecturas: F1-score

Arquitectura	Precision	Recall	F1-score
VGG16	96.12	94.12	95.11
ResNeXt	92.86	93.79	93.32
MobileNet v2	93.62	92.54	93.08
ResNet	92.36	93.48	92.92
VGG19	92.25	92.85	92.55
DenseNet	93.48	91.59	92.53
Wide ResNetx	93.13	91.8	92.46
AlexNet	93.03	89.91	91.44
GoogLeNet	92.11	89.39	90.73
MNASNet	85.71	86.11	85.91
ShuffleNet	29.14	50.00	36.82

El mecanismo de separación tipo *hold-out* puede no ser suficiente al comparar distintos modelos, es por ello por lo que con la finalidad de eliminar esta fuente de variabilidad se realizaron otros experimentos mediante validación cruzada con 5-folds (ver Figura 23) tal como se sugiere en [61], seguida de una comparación estadística de los resultados de cada modelo, de la misma forma como se realiza en [12]. Los resultados de la validación cruzada se pueden apreciar en la Tabla 9.

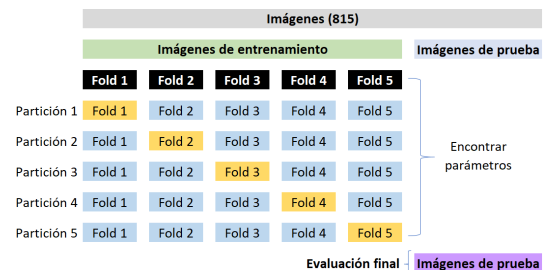


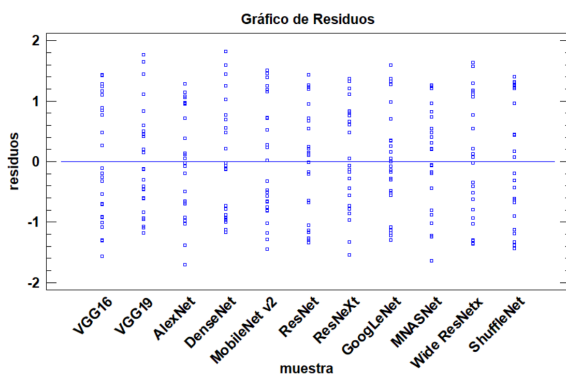
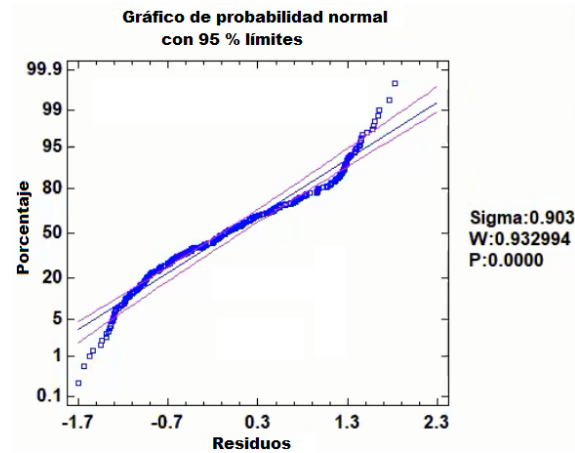
Figura 23. Validación cruzada con 5-folds

Tabla 9. Resultados – Validación cruzada con 5-folds

Arquitectura	Fold 1	...	Fold 5	Promedio
VGG16	93.87	...	98.16	96.81
VGG19	93.87	...	98.77	95.83
AlexNet	90.18	...	99.38	95.34
ResNeXt	94.48	...	98.16	94.6
DenseNet	88.34	...	98.16	94.36
MobileNet v2	90.18	...	96.32	93.99
ResNet	93.87	...	95.09	93.74
GoogLeNet	88.96	...	93.86	93.5
MNASNet	86.51	...	98.15	92.52
Wide ResNetx	84.66	...	94.48	89.82
ShuffleNet	68.09	...	72.39	70.06

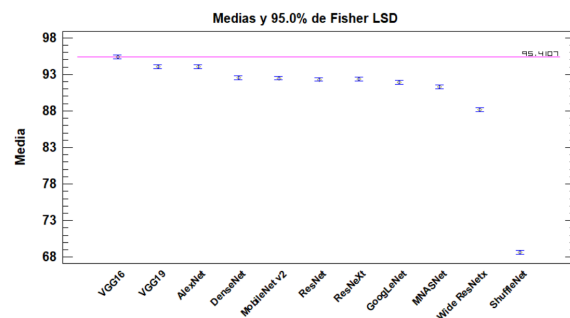
En relación con las pruebas estadísticas para evaluar el rendimiento de las arquitecturas, en primer lugar, se ejecutó 30 veces cada modelo, tal como se llevó a cabo en [12]. Posteriormente se realizó el análisis de residuos y de probabilidad normal evidenciándose que, según el primero, se podría aplicar una prueba paramétrica ya que los residuos presentan una dispersión similar (ver Figura 24). Sin embargo, al analizar el gráfico de probabilidad normal para verificar que los residuos se aproximen a una distribución normal, se aprecia que existen datos que se encuentran fuera del intervalo de confianza, con el coeficiente de Shapiro–Wilk igual a 0.932994 y un *p-value* igual a 0 (ver Figura 25). De igual manera, se realizó la transformación de los datos mediante la raíz cuadrada, sin embargo, estos siguen encontrándose fuera del intervalo de confianza.

Debido a que las poblaciones no se aproximan a una distribución normal, no se puede aplicar una prueba paramétrica; es por ello por lo que se hace necesario utilizar una prueba no paramétrica, específicamente el Test de Friedman en el que no se requiere cumplir la condición de normalidad o de homocedasticidad (igualdad de varianzas). Mediante este análisis plasmado en la Tabla 10, se obtiene que el *p-value* es igual a cero, es decir, que existe diferencia entre las poblaciones, mostrando que VGG16 es superior a las demás arquitecturas.

**Figura 24.** Gráfico de residuos**Figura 25.** Gráfico de probabilidad normal**Tabla 10.** Resultados estadísticos – Test de Friedman

Ranking	Arquitectura	Rango promedio
1	VGG16	10.68
2	AlexNet	9.28
3	VGG19	9.28
4	DenseNet	6.48
5	MobileNet v2	6.32
6	ResNeXt	6.06
7	ResNet	5.78
8	GoogLeNet	5.16
9	MNASNet	3.96
10	Wide ResNetx	2.00
11	ShuffleNet	1.00

Este mismo resultado se evidencia en el análisis de medias, en el que se aprecia que el intervalo de confianza que se ha construido tanto con el método Fisher LSD (Figura 26) como Tukey HSD (Figura 27), es superior en el caso de VGG16 ya que no se interseca con el de las otras arquitecturas. En el caso del análisis de medianas se observa que la arquitectura VGG16 es superior a la gran mayoría, pero presenta una pequeña intersección con las arquitecturas AlexNet y VGG19 (ver Figura 28).

**Figura 26.** Gráfico de medias – Fisher LSD

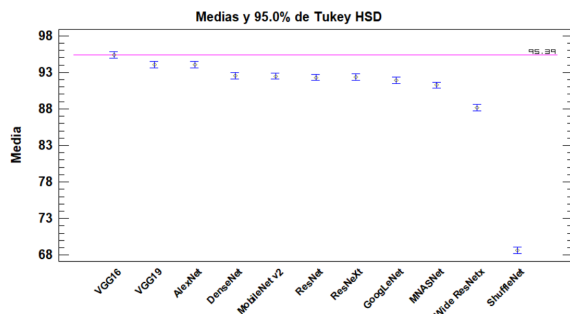


Figura 27. Gráfico de medias – Tukey HSD

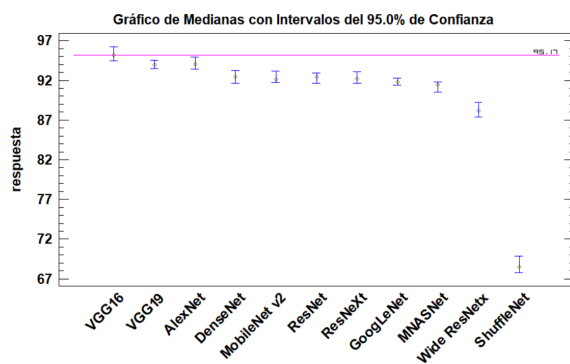


Figura 28. Gráfico de medianas

Adicionalmente, se realizó el análisis de comparaciones múltiples mediante la prueba con múltiples rangos (ver Tabla 11), en donde si las «X» de los grupos homogéneos se encuentran en la misma columna las arquitecturas se comportan de manera similar, apreciándose que VGG16 es superior y diferente a las demás arquitecturas.

Tabla 11. Prueba de múltiples rangos

Arquitectura	Media	Grupos homogéneos
ShuffleNet	686.560	X
Wide ResNetx	881.820	X
MNASNet	912.540	X
GoogLeNet	919.068	XX
ResNet	923.080	X
ResNeXt	923.704	X
MobileNet v2	924.816	X
DenseNet	925.336	X
AlexNet	940.500	X
VGG19	940.652	X
VGG16	953.756	X

El presente artículo no pretende obtener un «mejor» modelo, sino plantear un mecanismo alternativo frente a las técnicas tradicionales de inteligencia artificial. Sin embargo, con fines académicos, se presentan los resultados del modelo que obtuvo prestaciones más altas, es decir, VGG16, cuya arquitectura original se aprecia en la Figura 10, por lo que la arquitectura final queda definida como se detalla en la Figura 29.

En esta arquitectura se aprecia que la entrada está representada por las imágenes de 328×288 píxeles, las cuales ingresan a la red neuronal convolucional previamente entrenada VGG16, compuesta por trece capas convoluciones seguidas por tres capas totalmente conectadas, las dos primeras presentan 4096 canales y la última 1000, por lo que esta fue editada para poder realizar una clasificación binaria (2 canales). Las capas ocultas utilizan la función de activación ReLU además de diversos kernels 3×3 . Finalmente, se obtiene como salida la clasificación entre ambas clases, es decir, líneas y subestaciones.

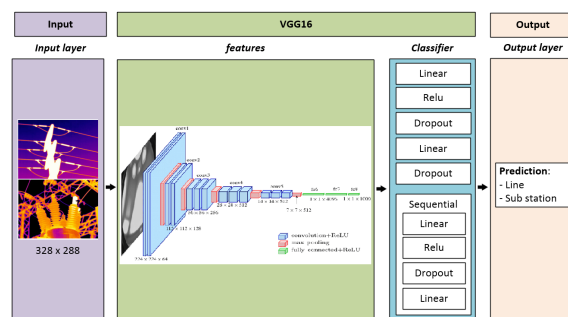


Figura 29. Propuesta de arquitectura de transfer learning para la clasificación de imágenes térmicas

Con este modelo se obtuvieron precisiones de 95.91 % y 91.41 % en el entrenamiento y validación (Figura 30). Esta arquitectura fue probada con nuevas imágenes pertenecientes al conjunto de datos de prueba, el resultado obtenido fue 94.43 % de exactitud para la categoría Líneas, y 92.81 % para las subestaciones. Esta se puede apreciar en la matriz de confusión plasmada en la Figura 31.

Se representó gráficamente la exactitud del modelo mediante la curva característica de operación del receptor (ROC), cuya área bajo la curva (AUC) muestra un valor del 94 %, lo cual es indicio de un alto rendimiento de la arquitectura propuesta en la clasificación de las imágenes térmicas (ver Figura 32).

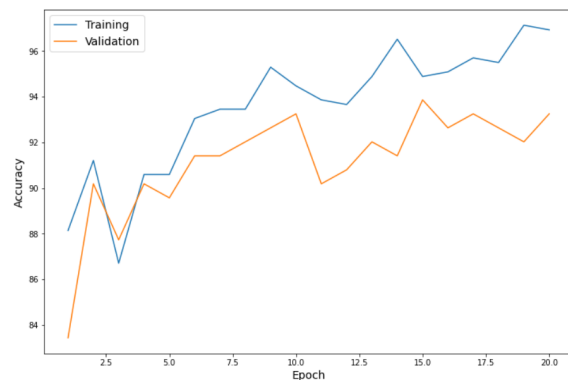


Figura 30. Accuracy: entrenamiento y validación del modelo

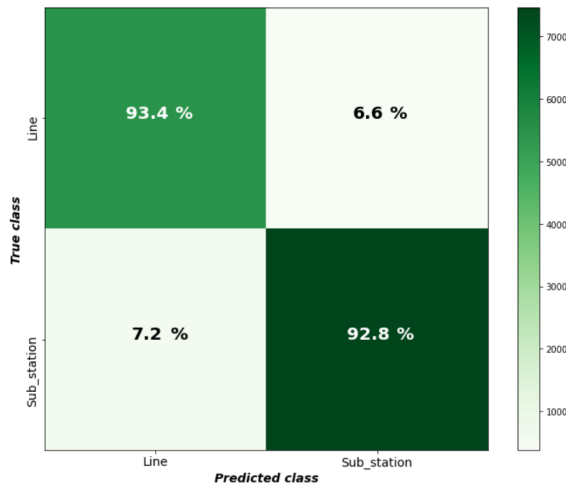


Figura 31. Matriz de confusión

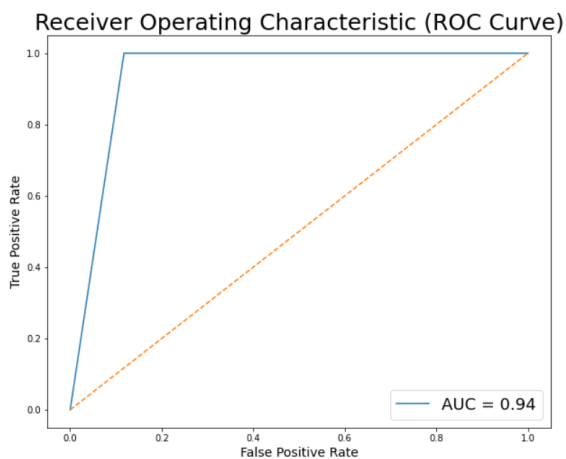


Figura 32. Curva ROC

Finalmente, en la Figura 33 se muestran algunos ejemplos de las predicciones del modelo. En el lado izquierdo del título de cada imagen se representa la clasificación real, y en el lado de la derecha, la obtenida por el modelo.

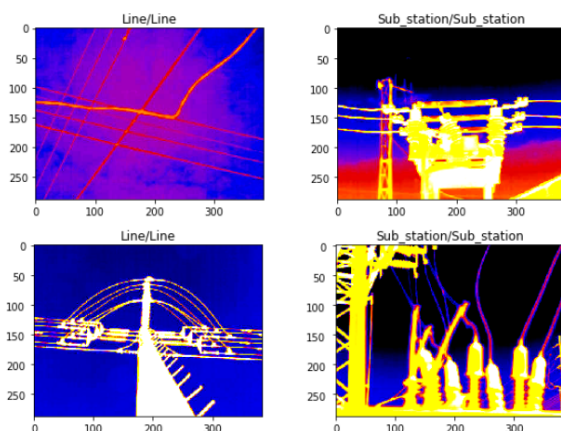


Figura 33. Predicciones del modelo

4. Conclusiones

En este trabajo se han analizado once redes neuronales previamente entrenadas, mediante el uso del paradigma de *transfer learning* basado en el *fine-tuning* del modelo aplicado a la clasificación binaria de imágenes térmicas. El objetivo final no es encontrar un «mejor» modelo, sino plantear alternativas frente al uso de técnicas tradicionales de inteligencia artificial, buscando un ahorro de tiempo y carga computacional.

Los resultados de los modelos arrojan precisiones entre 79.14 % y 98.15 %, valores de F1-score entre 85.91 % y 95.11 % en las arquitecturas previamente entrenadas; estos resultados son indicio que el uso de técnicas de *transfer learning* supone una alternativa confiable como mecanismo de clasificación de imágenes térmicas en el sector eléctrico; sin embargo, se recomienda realizar un análisis específico en cada caso de aplicación particular.

El uso de data augmentation, transformaciones y normalización de las imágenes, son aspectos importantes para mejorar el rendimiento del modelo; mientras que la división del conjunto de datos en subconjuntos de entrenamiento, validación y pruebas mediante la técnica de tipo *hold-out* ayudó a evitar el sobreajuste, generalizar el modelo y, por lo tanto, realizar predicciones más precisas. Sin embargo, con la finalidad de realizar una comparación más justa, el estudio también incluyó la aplicación de una validación cruzada con 5-folds, además de la realización de un análisis estadístico mediante el test de Friedman.

Agradecimientos

Daniel Pérez-Aguilar reconoce el apoyo financiero del Proyecto CONCYTEC-Banco Mundial, a través de su unidad ejecutora, el Fondo Nacional de Desarrollo Científico, Tecnológico y de Innovación Tecnológica (FONDECYT), en el marco de la convocatoria E033-2018-01-BM del Contrato N.º 06-2018-FONDECYT/BM, para su trabajo de investigación denominado Transfer learning en la clasificación binaria de imágenes térmicas, ejecutado como parte del programa de doctorado en Ingeniería con mención en Automatización, Control y Optimización de Procesos, desarrollado en el Laboratorio de Sistemas Automáticos de Control de la Universidad de Piura, Perú.

Referencias

- [1] M. Haenlein and A. Kaplan, “A brief history of artificial intelligence: On the past, present, and future of artificial intelligence,” *California Management Review*, vol. 61, no. 4, pp. 5–14, 2019. [Online]. Available: <https://doi.org/10.1177/0008125619864925>

- [2] M. Flasiński, *Introduction to artificial intelligence*. Springer International Publishing, 2016. [Online]. Available: <http://doi.org/10.1007/978-3-319-40022-8>
- [3] M.-H. Huang and R. T. Rust, “Artificial intelligence in service,” *Journal of Service Research*, vol. 21, no. 2, pp. 155–172, 2018. [Online]. Available: <https://doi.org/10.1177/1094670517752459>
- [4] Z. Aung, I. S. Mikhaylov, and Y. T. Aung, “Artificial intelligence methods application in oil industry,” in *2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EICoN Rus)*, 2020, pp. 563–567. [Online]. Available: <https://doi.org/10.1109/EICoN Rus49466.2020.9039330>
- [5] T. P. Carvalho, F. A. A. M. N. Soares, R. Vita, R. da P. Francisco, J. ao P. Basto, and S. G. S. Alcalá, “A systematic literature review of machine learning methods applied to predictive maintenance,” *Computers & Industrial Engineering*, vol. 137, p. 106024, 2019. [Online]. Available: <https://doi.org/10.1016/j.cie.2019.106024>
- [6] S. Wan, L. Qi, X. Xu, C. Tong, and Z. Gu, “Deep learning models for real-time human activity recognition with smartphones,” *Mobile Networks and Applications*, vol. 25, no. 2, pp. 743–755, Apr. 2020. [Online]. Available: <https://doi.org/10.1007/s11036-019-01445-x>
- [7] V. Golodov, A. Zavei-Boroda, S. Ivanov, and K. Nikolskaya, “Development of a deep learning neural network for human movements analysis,” in *2017 Second Russia and Pacific Conference on Computer Technology and Applications (RPC)*, 2017, pp. 72–74. [Online]. Available: <https://doi.org/10.1109/RPC.2017.8168071>
- [8] E. A. Galindo, J. A. Perdomo, and J. C. Figueroa-García, “Estudio comparativo entre máquinas de soporte vectorial multiclase, redes neuronales artificiales y sistema de inferencia neuro-difuso autoorganizado para problemas de clasificación,” *Información tecnológica*, vol. 31, pp. 273–286, 02 2020. [Online]. Available: <http://dx.doi.org/10.4067/S0718-07642020000100273>
- [9] A. Brunetti, D. Buongiorno, G. F. Trotta, and V. Bevilacqua, “Computer vision and deep learning techniques for pedestrian detection and tracking: A survey,” *Neurocomputing*, vol. 300, pp. 17–33, 2018. [Online]. Available: <https://doi.org/10.1016/j.neucom.2018.01.092>
- [10] I. Yildiz, P. Tian, J. Dy, D. Erdoğmuş, J. Brown, J. Kalpathy-Cramer, S. Ostmo, J. Peter Campbell, M. F. Chiang, and S. Ioannidis, “Classification and comparison via neural networks,” *Neural Networks*, vol. 118, pp. 65–80, 2019. [Online]. Available: <https://doi.org/10.1016/j.neunet.2019.06.004>
- [11] Y. Jung, “Multiple predicting k-fold cross-validation for model selection,” *Journal of Nonparametric Statistics*, vol. 30, no. 1, pp. 197–215, 2018. [Online]. Available: <https://doi.org/10.1080/10485252.2017.1404598>
- [12] F. Pacheco, J. Valente de Oliveira, R.-V. Sánchez, M. Cerrada, D. Cabrera, C. Li, G. Zurita, and M. Artés, “A statistical comparison of neuroclassifiers and feature selection methods for gearbox fault diagnosis under realistic conditions,” *Neurocomputing*, vol. 194, pp. 192–206, 2016. [Online]. Available: <https://doi.org/10.1016/j.neucom.2016.02.028>
- [13] D. W. Zimmerman and B. D. Zumbo, “Relative power of the Wilcoxon test, the Friedman test, and repeated-measures ANOVA on ranks,” *The Journal of Experimental Education*, vol. 62, no. 1, pp. 75–86, 1993. [Online]. Available: <https://doi.org/10.1080/00220973.1993.9943832>
- [14] C. Lile and L. Yiqun, “Anomaly detection in thermal images using deep neural networks,” in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 2299–2303. [Online]. Available: <https://doi.org/10.1109/ICIP.2017.8296692>
- [15] A. Dragomir, M. Adam, M. Andruçcă, A. Munteanu, and E. Boghiu, “Considerations regarding infrared thermal stresses monitoring of electrical equipment,” in *2017 International Conference on Electromechanical and Power Systems (SIELMEN)*, 2017, pp. 100–103. [Online]. Available: <https://doi.org/10.1109/SIELMEN.2017.8123307>
- [16] F. Fambrini, Y. Iano, D. G. Caetano, A. A. D. Rodríguez, C. Moya, E. Carrara, R. Arthur, F. C. Cabello, J. V. Zubem, L. M. Del Val Cura, J. a. B. Destro Filho, J. R. Campos, and J. H. Saito, “Gpu cuda jseg segmentation algorithm associated with deep learning classifier for electrical network images identification,” *Procedia Computer Science*, vol. 126, pp. 557–565, 2018, knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 22nd International Conference, KES-2018, Belgrade, Serbia. [Online]. Available: <https://doi.org/10.1016/j.procs.2018.07.290>
- [17] X. W. X. L. Z. J. Wenzhen Yang, Jiali Luo and Z. Pan, “Image tactile perception with an improved jseg algorithm,” *International Journal of Performability Engineering*, vol. 14,

- no. 1, p. 77, 2018. [Online]. Available: <https://doi.org/10.23940/ijpe.18.01.p9.7788>
- [18] I. Ullah, F. Yang, R. Khan, L. Liu, H. Yang, B. Gao, and K. Sun, “Predictive maintenance of power substation equipment by infrared thermography using a machine-learning approach,” *Energies*, vol. 10, no. 12, 2017. [Online]. Available: <https://doi.org/10.3390/en10121987>
- [19] H. Ramchoun, Y. Ghanou, M. Ettaouil, and M. A. J. Idrissi, “Multilayer perceptron: Architecture optimization and training,” *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 4, no. 1, 2016. [Online]. Available: <http://doi.org/10.9781/ijimai.2016.415>
- [20] A. S. Nazmul Huda, S. Taib, M. S. Jadin, and D. Ishak, “A semi-automatic approach for thermographic inspection of electrical installations within buildings,” *Energy and Buildings*, vol. 55, pp. 585–591, 2012. [Online]. Available: <https://doi.org/10.1016/j.enbuild.2012.09.014>
- [21] C. Yuan, X. Sun, and R. Lv, “Fingerprint liveness detection based on multi-scale OPQ and PCA,” *China Communications*, vol. 13, no. 7, pp. 60–65, 2016. [Online]. Available: <https://doi.org/10.1109/CC.2016.7559076>
- [22] H. Zou and F. Huang, “A novel intelligent fault diagnosis method for electrical equipment using infrared thermography,” *Infrared Physics & Technology*, vol. 73, pp. 29–35, 2015. [Online]. Available: <https://doi.org/10.1016/j.infrared.2015.08.019>
- [23] S.-S. Yu, S.-W. Chu, C.-M. Wang, Y.-K. Chan, and T.-C. Chang, “Two improved k-means algorithms,” *Applied Soft Computing*, vol. 68, pp. 747–755, 2018. [Online]. Available: <https://doi.org/10.1016/j.asoc.2017.08.032>
- [24] T. V. Phan, S. Sultana, T. G. Nguyen, and T. Bauschert, “Q - transfer: A novel framework for efficient deep transfer learning in networking,” in *2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC)*, 2020, pp. 146–151. [Online]. Available: <https://doi.org/10.1109/ICAIIIC48513.2020.9065240>
- [25] M. Ebrahim, M. Al-Ayyoub, and M. A. Alsmirat, “Will transfer learning enhance imagenet classification accuracy using imagenet-pretrained models?” in *2019 10th International Conference on Information and Communication Systems (ICICS)*, 2019, pp. 211–216. [Online]. Available: <https://doi.org/10.1109/IACS.2019.8809114>
- [26] T. Alshalali and D. Josyula, “Fine-tuning of pre-trained deep learning models with extreme learning machine,” in *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, 2018, pp. 469–473. [Online]. Available: <https://doi.org/10.1109/CSCI46756.2018.00096>
- [27] G. Vrbančić and V. Podgorelec, “Transfer learning with adaptive fine-tuning,” *IEEE Access*, vol. 8, pp. 196 197–196 211, 2020. [Online]. Available: <https://doi.org/10.1109/ACCESS.2020.3034343>
- [28] T. Kaur and T. K. Gandhi, “Automated brain image classification based on vgg-16 and transfer learning,” in *2019 International Conference on Information Technology (ICIT)*, 2019, pp. 94–98. [Online]. Available: <https://doi.org/10.1109/ICIT48102.2019.00023>
- [29] R. L. Gálvez, E. P. Dadios, A. A. Bandalá, and R. R. P. Vicerra, “Threat object classification in X-ray images using transfer learning,” in *2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*, 2018, pp. 1–5. [Online]. Available: <https://doi.org/10.1109/HNICEM.2018.8666344>
- [30] D. Xue, X. Zhou, C. Li, Y. Yao, M. M. Rahaman, J. Zhang, H. Chen, J. Zhang, S. Qi, and H. Sun, “An application of transfer learning and ensemble learning techniques for cervical histopathology image classification,” *IEEE Access*, vol. 8, pp. 104 603–104 618, 2020. [Online]. Available: <https://doi.org/10.1109/ACCESS.2020.2999816>
- [31] E. Cengil and A. Çinar, “Multiple classification of flower images using transfer learning,” in *2019 International Artificial Intelligence and Data Processing Symposium (IDAP)*, 2019, pp. 1–6. [Online]. Available: <https://doi.org/10.1109/IDAP.2019.8875953>
- [32] J. R. Rajayogi, G. Manjunath, and G. Shobha, “Indian food image classification with transfer learning,” in *2019 4th International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)*, vol. 4, 2019, pp. 1–4. [Online]. Available: <https://doi.org/10.1109/CSITSS47250.2019.9031051>
- [33] H. Shao, M. Xia, G. Han, Y. Zhang, and J. Wan, “Intelligent fault diagnosis of rotor-bearing system under varying working conditions with modified transfer convolutional neural network and thermal images,” *IEEE Transactions on Industrial Informatics*, vol. 17,

- no. 5, pp. 3488–3496, 2021. [Online]. Available: <https://doi.org/10.1109/TII.2020.3005965>
- [34] O. Janssens, R. Van de Walle, M. Locufier, and S. Van Hoecke, “Deep learning for infrared thermal image based machine health monitoring,” *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 1, pp. 151–159, 2018. [Online]. Available: <https://doi.org/10.1109/TMECH.2017.2722479>
- [35] T. Carneiro, R. V. Medeiros Da Nóbrega, T. Nepomuceno, G.-B. Bian, V. H. C. De Albuquerque, and P. P. R. Filho, “Performance analysis of google colab as a tool for accelerating deep learning applications,” *IEEE Access*, vol. 6, pp. 61 677–61 685, 2018. [Online]. Available: <https://doi.org/10.1109/ACCESS.2018.2874767>
- [36] A. S. N. Huda and S. Taib, “Suitable features selection for monitoring thermal condition of electrical equipment using infrared thermography,” *Infrared Physics & Technology*, vol. 61, pp. 184–191, 2013. [Online]. Available: <https://doi.org/10.1016/j.infrared.2013.04.012>
- [37] M. S. Jadin, S. Taib, and K. H. Ghazali, “Feature extraction and classification for detecting the thermal faults in electrical installations,” *Measurement*, vol. 57, pp. 15–24, 2014. [Online]. Available: <https://doi.org/10.1016/j.measurement.2014.07.010>
- [38] W. I. Technology, *ThermoPro TP8S™ IR Thermal Camera. User Manual*. Wuhan Guide Infrared Technology Co., Ltd., 2007. [Online]. Available: <https://bit.ly/3bVGd0u>
- [39] L. Sandjakoska and F. Stojanovska, “How initialization is related to deep neural networks generalization capability: Experimental study,” in *2020 55th International Scientific Conference on Information, Communication and Energy Systems and Technologies (ICEST)*, 2020, pp. 163–166. [Online]. Available: <https://doi.org/10.1109/ICEST49890.2020.9232882>
- [40] C. Heghedus, A. Chakravorty, and C. Rong, “Neural network frameworks. comparison on public transportation prediction,” in *2019 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, 2019, pp. 842–849. [Online]. Available: <https://doi.org/10.1109/IPDPSW.2019.00138>
- [41] A. A. Almisreb, N. Jamil, and N. M. Din, “Utilizing alexnet deep transfer learning for ear recognition,” in *2018 Fourth International Conference on Information Retrieval and Knowledge Management (CAMP)*, 2018, pp. 1–5. [Online]. Available: <https://doi.org/10.1109/INFRKM.2018.8464769>
- [42] S. Liu and W. Deng, “Very deep convolutional neural network based image classification using small training sample size,” in *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, 2015, pp. 730–734. [Online]. Available: <https://doi.org/10.1109/ACPR.2015.7486599>
- [43] J. Xiao, J. Wang, S. Cao, and B. Li, “Application of a novel and improved VGG-19 network in the detection of workers wearing masks,” *Journal of Physics: Conference Series*, vol. 1518, p. 012041, apr 2020. [Online]. Available: <https://doi.org/10.1088/1742-6596/1518/1/012041>
- [44] A. Budhiman, S. Suyanto, and A. Arifianto, “Melanoma cancer classification using resnet with data augmentation,” in *2019 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, 2019, pp. 17–20. [Online]. Available: <https://doi.org/10.1109/ISRITI48646.2019.9034624>
- [45] K. Zhang, Y. Guo, X. Wang, J. Yuan, and Q. Ding, “Multiple feature reweight densenet for image classification,” *IEEE Access*, vol. 7, pp. 9872–9880, 2019. [Online]. Available: <https://doi.org/10.1109/ACCESS.2018.2890127>
- [46] T. Fang, “A novel computer-aided lung cancer detection method based on transfer learning from googlenet and median intensity projections,” in *2018 IEEE International Conference on Computer and Communication Engineering Technology (CCET)*, 2018, pp. 286–290. [Online]. Available: <https://doi.org/10.1109/CCET.2018.8542189>
- [47] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9. [Online]. Available: <https://doi.org/10.1109/CVPR.2015.7298594>
- [48] GeeksforGeeks. (2020) Understanding googlenet model - CNN architecture. [Online]. Available: <https://bit.ly/2RLmiuc>
- [49] K. Fu, L. Sun, X. Kang, and F. Ren, “Text detection for natural scene based on mobilenet V2 and U-net,” in *2019 IEEE International Conference on Mechatronics and Automation (ICMA)*, 2019, pp. 1560–1564. [Online]. Available: <https://doi.org/10.1109/ICMA.2019.8816384>

- [50] C. Qiu, M. Schmitt, H. Taubenböck, and X. X. Zhu, “Mapping human settlements with multi-seasonal sentinel-2 imagery and attention-based resnext,” in *2019 Joint Urban Remote Sensing Event (JURSE)*, 2019, pp. 1–4. [Online]. Available: <https://doi.org/10.1109/JURSE.2019.8809009>
- [51] S. Zagoruyko and N. Komodakis, “Wide residual networks,” in *Proceedings of the British Machine Vision Conference (BMVC)*, E. R. H. Richard C. Wilson and W. A. P. Smith, Eds. BMVA Press, September 2016, pp. 87.1–87.12. [Online]. Available: <https://dx.doi.org/10.5244/C.30.87>
- [52] M. Tan, B. Chen, R. Pang, V. Vasudevan, M. Sandler, A. Howard, and Q. V. Le, “Mnasnet: Platform-aware neural architecture search for mobile,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 2815–2823. [Online]. Available: <https://doi.org/10.1109/CVPR.2019.00293>
- [53] Y. Li and C. Lv, “Ss-yolo: An object detection algorithm based on YOLOv3 and shufflenet,” in *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, vol. 1, 2020, pp. 769–772. [Online]. Available: <https://doi.org/10.1109/ITNEC48623.2020.9085091>
- [54] PyTorch. (2019) TORCHVISION.MODELS. [Online]. Available: <https://bit.ly/2QSc1Ge>
- [55] X. Song, Y. Du, and J. Jackson, “An empirical study on hyperparameters and their interdependence for RL generalization,” *arXiv preprint arXiv*, vol. abs/1906.00431, 2019. [Online]. Available: <https://bit.ly/3ulY3zZ>
- [56] J. N. van Rijn and F. Hutter, “Hyperparameter importance across datasets,” in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD ’18. New York, NY, USA: Association for Computing Machinery, 2018, pp. 2367–2376. [Online]. Available: <https://doi.org/10.1145/3219819.3220058>
- [57] A. Aravkin, J. V. Burke, A. Chiuso, and G. Pillonetto, “On the estimation of hyperparameters for empirical bayes estimators: Maximum marginal likelihood vs minimum MSE,” *IFAC Proceedings Volumes*, vol. 45, no. 16, pp. 125–130, 2012, 16th IFAC Symposium on System Identification. [Online]. Available: <https://doi.org/10.3182/20120711-3-BE-2027.00353>
- [58] A. Mikolajczyk and M. Grochowski, “Data augmentation for improving deep learning in image classification problem,” in *2018 International Interdisciplinary PhD Workshop (IIPhDW)*, 2018, pp. 117–122. [Online]. Available: <https://doi.org/10.1109/IIPHDW.2018.8388338>
- [59] C. Shorten and T. M. Khoshgoftaar, “A survey on image data augmentation for deep learning,” *Journal of Big Data*, vol. 6, no. 1, p. 60, Jul. 2019. [Online]. Available: <https://doi.org/10.1186/s40537-019-0197-0>
- [60] D. Avola, L. Cinque, G. L. Foresti, F. Lamacchia, M. R. Marini, L. Perini, K. Qorraj, and G. Telesca, “A shape comparison reinforcement method based on feature extractors and f1-score,” in *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, 2019, pp. 2155–2159. [Online]. Available: <https://doi.org/10.1109/SMC.2019.8914601>
- [61] J. Amat Rodrigo. (2020) Validación de modelos predictivos: Cross-validation, one-leaveout, bootstrapping. [Online]. Available: <https://bit.ly/3bYgPHk>