Trabajo Fin de Máster

Detección de imágenes fuera de distribución en redes neuronales

Daniel Jove Díaz

Máster en Técnicas Estadísticas

Curso 2023-2024

Propuesta de Trabajo Fin de Máster

Título en galego: Detección de imaxes fora da distribución en redes neuronales
Título en español: Detección de imágenes fuera de distribución en redes neuronales
English title: Out-of-Distribution image detection in neural networks
Modalidad: Modalidad B
Autor/a: Daniel Jove Díaz, Universidad de A Coruña
Director/a: Marta Sestelo Pérez, Universidade de Vigo
Tutor/a: Alfonso Lorenzo Rodríguez, Gradiant
Breve resumen del trabajo: En este trabajo se realiza una análisis teórico y práctico de los algoritmos estado del arte en detección de imágenes fuera de distribución en redes neuronales. Este servirá para identificar cual de ellos resulta más adecuado para su aplicación en un sistema de detección y evasión para UAVs
Recomendaciones:
Otras observaciones:

Doña Marta Sestelo Pérez, Profesora Titular de la Universidade de Vigo, y Don Alfonso Lorenzo Rodríguez, Gestor de proyectos aeronáuticos y Asuntos regulatorios de Gradiant, informan que el Trabajo Fin de Máster titulado

Detección de imágenes fuera de distribución en redes neuronales

fue realizado bajo su dirección por don/doña Daniel Jove Díaz para el Máster en Técnicas Estadísticas. Estimando que el trabajo está terminado, dan su conformidad para su presentación y defensa ante un tribunal.

En A Coruña, a 22 de Julio de 2024.

El/la director/a:
Don/doña Marta Sestelo Pérez

SESTELO PEREZ MARTA - 76998300G
Firmado digitalmente por SESTELO PEREZ MARTA - 76998300G
Fecha: 2024.07.22 22:08:32 +02'00'

El/la tutor/a:
Don/doña Alfonso Lorenzo Rodríguez

LORENZO RODRIGUEZ , ALFONSO (FIRMA)
Firmado digitalmente por LORENZO RODRIGUEZ, ALFONSO (FIRMA)
Fecha: 2024.07.22 17:09:28 +02'00'

El/la autor/a:
Don/doña Daniel Jove Díaz

JOVE DIAZ DANIEL - 76584486Y
Firmado digitalmente por JOVE DIAZ DANIEL - 76584486Y
Fecha: 2024.07.22 17:18:44 +02'00'

Declaración responsable. Para dar cumplimiento a la Ley 3/2022, de 24 de febrero, de convivencia universitaria, referente al plagio en el Trabajo Fin de Máster (Artículo 11, Disposición 2978 del BOE núm. 48 de 2022), **el/la autor/a declara** que el Trabajo Fin de Máster presentado es un documento original en el que se han tenido en cuenta las siguientes consideraciones relativas al uso de material de apoyo desarrollado por otros/as autores/as:

- Todas las fuentes usadas para la elaboración de este trabajo han sido citadas convenientemente (libros, artículos, apuntes de profesorado, páginas web, programas,...)
- Cualquier contenido copiado o traducido textualmente se ha puesto entre comillas, citando su procedencia.
- Se ha hecho constar explícitamente cuando un capítulo, sección, demostración,...sea una adaptación casi literal de alguna fuente existente.

Y, acepta que, si se demostrara lo contrario, se le apliquen las medidas disciplinarias que correspondan.

Agradecimientos

Quisiera expresar mi más sincero agradecimiento a Gradiant por brindarme la oportunidad de realizar este Trabajo de Fin de Máster. En especial, agradecer a los miembros de la línea Video Analytics por hacerme sentir parte del grupo desde el primer día y por ayudarme en el desarrollo de este trabajo.

A mi tutor en Gradiant, Alfonso por compartir sus amplios conocimientos y guiarme en el diseño del proyecto.

Finalmente, a mi tutora Marta, por asesorarme en la estructura y redacción de la memoria de este trabajo gracias a su punto de vista académico.

Índice general

Resumen xi

1. Introducción 1
1.1. Contexto . . . 1
1.2. Introducción del concepto Out-Of-Distribution . . . 2
1.3. Gradiant . . . 3
1.4. Motivación y objetivos . . . 4
1.5. Estructura del documento . . . 5

2. Redes Neuronales para la clasificación de imágenes 7
2.1. Introducción . . . 7
2.2. ResNet . . . 8
2.3. WideResnet . . . 9
2.4. Gradiente . . . 9
2.5. Data augmentation . . . 10
2.6. Métricas de evaluación . . . 10

3. Estado del Arte 13
3.1. Detección Out-of-Distribution . . . 13
3.2. Algoritmos utilizados . . . 16
3.2.1. ODIN . . . 16
3.2.2. Mahalanobis . . . 17
3.2.3. Generalized ODIN . . . 18
3.2.4. *Energy-based OOD detection* . . . 20

4. Experimentos y resultados 23
4.1. Conjuntos de datos públicos . . . 23
4.1.1. Condiciones de luz . . . 25
4.2. Conjuntos de datos propiedad de Gradiant . . . 27
4.2.1. Entorno operacional . . . 27
4.2.2. Reconocimiento de objetos voladores . . . 27

5. Conclusiones y trabajo futuro 31

Bibliografía 33

Resumen

Resumen en español

En este trabajo se realiza un análisis teórico y experimental de los algoritmos de detección de imágenes fuera de distribución (*Out-of-Distribution*, OOD) en redes neuronales, con el objetivo de identificar el más adecuado para su aplicación en sistemas de detección y evasión en vehículos aéreos no tripulados (*Unmanned Aerial Vehicles*, UAVs).

La capacidad de detectar imágenes OOD es crucial para garantizar la seguridad y eficiencia de los UAVs, especialmente en operaciones más allá de la línea de vista (*Beyond Visual Line of Sight*, BVLOS). Identificar objetos voladores y situaciones del entorno operacional no previstos durante el entrenamiento ayuda a evitar que los sistemas de detección y evasión tomen decisiones incorrectas basadas en predicciones erróneas de las redes neuronales.

English abstract

In this paper we perform a theoretical and experimental analysis of out-of-distribution (OOD) detection algorithms in neural networks, with the aim of identifying the most suitable algorithm for application in unmanned aerial vehicle (UAV) detection and avoidance systems.

The ability to detect OOD images is crucial to ensure the safety and efficiency of UAVs, especially in beyond visual line of sight (BVLOS) operations of the operator. Identifying unanticipated flying objects and situations in the operational environment during training helps prevent detection and avoidance systems from making incorrect decisions based on erroneous predictions from neural networks.

Capítulo 1

Introducción

En este primer capítulo se procederá a realizar una introducción al concepto de imágenes *Out-of-Distribution*, destacando la importancia de incorporar métodos para su detección en tareas de visión por computador y el interés de Gradiant en aplicarlos en el desarrollo de un sistema de detección y evasión (DAA, por su nombre en inglés *Detect and Avoid*) para vehículos aéreos no tripulados (UAV, por sus siglas en inglés *Unmanned Aerial Vehicles*).

1.1. Contexto

En el contexto de los vehículos aéreos no tripulados (UAV) existe el reto de poder realizar operaciones a una distancia que supere el alcance visual del operador. A este tipo de operaciones se las denominan más allá de la línea de vista (BVLOS, *Beyond Visual Line of Sight*) [1] y representan un avance significativo en la tecnología y operatividad de drones. La capacidad de volar a distancias alejadas del piloto es crucial porque amplía considerablemente el rango y las aplicaciones de los UAV, permitiendo misiones más largas y complejas que serían imposibles dentro de las limitaciones visuales directas del operador. BVLOS habilita una amplia gama de usos en industrias como la agricultura, la logística, la inspección de infraestructuras y la seguridad pública, optimizando procesos, reduciendo costos y mejorando la eficiencia operativa. Además, posibilita intervenciones en áreas de difícil acceso o peligrosas para los seres humanos, incrementando la seguridad y abriendo nuevas oportunidades para la innovación y el desarrollo tecnológico.

Para garantizar la seguridad en este tipo de operaciones, la Agencia Europea de Seguridad Aérea (EASA) establece una serie de normativas que motivan el desarrollo de tecnologías de detección y evasión (DAA) [2]. Estos sistemas permiten a los drones detectar y esquivar otros objetos automáticamente, mejorando la seguridad y facilitando su integración en el espacio aéreo.

Desarrollar un sistema de detección y evasión para drones enfrenta desafíos complejos, especialmente cuando se trata de objetos voladores que operan a alturas bajas y no están equipados con ADS-B (*Automatic Dependent Surveillance-Broadcast*), un sistema que permite a las aeronaves comunicar su ubicación precisa a otras aeronaves y controladores de tráfico aéreo. Debido a la normativa actual, drones y otros objetos voladores en cotas bajas no están obligados a incorporar esta tecnología, dificultando conocer su ubicación precisa en todo momento. En este escenario, la inteligencia artificial, y en concreto, la visión artificial emerge como una herramienta indispensable, ya que permite a los drones detectar y comprender su entorno de manera autónoma mediante el análisis de imágenes en tiempo real capturadas con las cámaras que incorporan.

Los sistemas DAA basados en inteligencia artificial (IA) deben adherirse a las directrices de la EASA sobre IA [3], que incluyen estándares de seguridad y ética. Dentro de estos estándares se recoge que los sistemas de DAA tienen que ser capaces de manejar situaciones no anticipadas, asegurando una operación segura y eficiente de los drones en diversos contextos. A lo largo de este trabajo se demostrará

que los métodos de discriminación *Out-of-Distribution* (OOD) son capaces de detectar las situaciones no previstas, por lo que su aplicación resulta de notable interés para el correcto funcionamiento de estos sistemas.

1.2. Introducción del concepto Out-Of-Distribution

Los modelos de aprendizaje automático que conforman el estado del arte actual, en concreto las redes neuronales profundas, se han convertido en una herramienta con un enorme potencial, siendo de gran utilidad en muchos ámbitos, debido a la capacidad de realizar diversas tareas, por ejemplo, reconocimiento de voz, detección de objetos, clasificación de imágenes, etc.

Sin embargo, existen campos de aplicación como la medicina o la conducción autónoma, donde las predicciones que realizan las redes neuronales pueden suponer tomar decisiones erróneas que provoquen daños personales o en el entorno de aplicación. La incertidumbre en las redes neuronales se refiere a la falta de certeza en las predicciones realizadas por estos modelos. En los últimos años muchas investigaciones tratan esta problemática, buscando entender y cuantificar la incertidumbre en las predicciones, observando los motivos que la producen y proponiendo soluciones para detectarla o reducirla al máximo.

En el proceso de construcción de una red neuronal existen muchos aspectos que pueden aumentar la incertidumbre de las predicciones obtenidas, desde aspectos controlables como puede ser la preparación del conjunto de datos de entrenamiento o la elección de la red neuronal, a otros imposibles de tener en cuenta que dependen de los datos obtenidos durante la aplicación. Estas fuentes de incertidumbre se pueden agrupar en la incetidumbre epistémica [4] y la incertidumbre aleatoria en los datos [5].

La primera surge como resultado de errores durante la construcción del modelo, que pueden deberse a la selección incorrecta del modelo, tener un conjunto de datos de entrenamiento incompleto, o una estructura del modelo inapropiada. Para mitigar esta incertidumbre, es posible realizar nuevos entrenamientos o mejorar los conjuntos de datos utilizados.

Por otro lado, la incertidumbre en los datos depende únicamente de la calidad y naturaleza de los datos que el sistema desplegado pueda obtener. Esta incertidumbre no puede ser controlada directamente, ya que es provocada por la variabilidad en los escenarios de aplicación y por errores en los sistemas de medición o en la recopilación de datos.

En el contexto de este trabajo, los datos que se utilizan como covariables son imágenes, por lo que es importante entender que el dominio de los datos de entrenamiento se refiere a la distribución de las imágenes seleccionadas para entrenar de los modelos de visión artificial. Esta distribución abarca la organización y representación de las imágenes dentro del conjunto de datos, incluyendo la frecuencia y variedad de las diferentes clases de imágenes, así como la diversidad de características visuales presentes. La forma en que se estructuran y presentan estos datos puede influir significativamente en la capacidad del modelo para generalizar a nuevos datos.

Teniendo esto en cuenta, la incertidumbre en los datos se puede categorizar en tres tipos de acuerdo con la publicación de Jakob Gawlikowski et al. [6], según la similitud entre los datos que se utilizarán para la predicción y los datos de entrenamiento:

- Incertidumbre en el dominio: Esta incertidumbre surge cuando se asume que las covariables utilizadas en las predicciones pertenecen a la misma distribución que las covariables de entrenamiento. Este tipo de incertidumbre se basa en la expectativa de que los datos de predicción sean representativos de los datos sobre los cuales se entrenó el modelo.

- Incertidumbre por desviación del dominio: Se presenta cuando los valores de las covariables que se utilizan para dar una predicción provienen de una distribución diferente a la de entrenamiento, pero están relacionadas. Esta desviación puede deberse a una falta de cobertura en el conjunto de datos de entrenamiento o a cambios que ocurren en el escenario real durante la aplicación del modelo. Esta incertidumbre refleja la capacidad limitada del modelo para generalizar a datos ligeramente distintos de aquellos en los que fue entrenado.

- Incertidumbre por datos fuera del dominio: Esta situación ocurre cuando los datos de entrada en la fase de predicción pertenecen a una distribución completamente diferente a la de los datos de entrenamiento, lo que implica que la red neuronal no puede extraer conclusiones precisas de estas nuevas entradas. Estos datos se consideran como datos *Out-of-Distribution* (OOD), y representan un desafío significativo para el rendimiento del modelo, ya que operan fuera del conocimiento adquirido durante el entrenamiento.

Considerando lo anterior, este trabajo se centra en mitigar la incertidumbre generada por datos fuera del dominio, enfocándose en la detección de datos *Out-of-Distribution* (OOD) en redes neuronales utilizadas para la clasificación de imágenes, con el objetivo de mejorar su robustez. Definiremos la robustez como la capacidad del modelo para mantener un rendimiento consistente y preciso, incluso cuando se enfrenta a datos que difieren de los utilizados durante el entrenamiento. Este desafío es especialmente relevante en el contexto de la clasificación de imágenes, donde los modelos pueden encontrarse con una amplia variedad de escenarios y condiciones que no estaban presentes en el conjunto de datos de entrenamiento.

Una distribución adecuada y representativa de los datos de entrenamiento es fundamental para que el modelo pueda generalizar correctamente a nuevas imágenes. Sin embargo, incluso con un conjunto de datos de entrenamiento bien estructurado, las redes neuronales pueden encontrarse con datos que no se ajustan a la distribución esperada. Estos datos, considerados *Out-of-Distribution* (OOD), presentan diferentes características visuales, estilos o clases que no fueron contemplados durante el entrenamiento.

La presencia de datos OOD puede causar que el modelo haga predicciones incorrectas, comprometiendo su robustez y fiabilidad. Por tanto, identificar y gestionar estos datos OOD es crucial para mejorar la eficacia y la seguridad de los sistemas de clasificación de imágenes basados en redes neuronales.

Para abordar este problema, es esencial implementar algoritmos capaces de detectar imágenes OOD que puedan identificar cuándo una imagen de entrada no pertenece a la distribución de los datos de entrenamiento. Al detectar estas anomalías, el sistema puede tomar decisiones más informadas, como rechazar la predicción y solicitar una intervención humana, o activar mecanismos de control alternativos. Esto no solo mejora la robustez del sistema, sino que también aumenta la confianza en su capacidad para manejar situaciones imprevistas de manera segura y eficaz.

1.3. Gradiant

Gradiant, Centro Tecnológico de Telecomunicaciones de Galicia (TIC), tiene como objetivo fundacional mejorar la competitividad de las empresas mediante la transferencia de conocimiento y tecnologías en los ámbitos de la conectividad, inteligencia y seguridad. Con más de 170 profesionales y 14 patentes solicitadas, Gradiant ha desarrollado más de 800 proyectos diferentes de I+D+i, convirtiéndose en uno de los principales motores de la innovación en Galicia. El Centro fue creado en 2008 y se conforma a partir de un patronato que agrupa a representantes del sector público y privado. Está formado por las Universidades de A Coruña, Santiago de Compostela y Vigo; las empresas Abanca, Altia, Arteixo Telecom, Egatel, Indra, Plexus, R, Telefónica, Televés, y la Asociación empresarial INEO.

El compromiso del centro con la calidad es una constante desde sus inicios. El Centro cuenta con los siguientes certificados: Sistema de Gestión de Calidad UNE-EN ISO 9001:2015, Sistema de Gestión de Proyectos de I+D+i UNE 166002:2014, Sistemas de Gestión de la Seguridad de la Información UNE-EN ISO/IEC 7001:2013. Además, forma parte del registro estatal de Centros de Innovación Tecnológica (Sello CIT). Tras 14 años de actividad, Gradiant se sitúa como socio tecnológico de la industria orientado a sus necesidades en el ámbito de las TIC, aportando su experiencia nacional e internacional en tecnologías para la seguridad y la privacidad; el procesado de señales multimedia; internet de las cosas; la biometría y analítica de datos y los sistemas de comunicaciones avanzadas.

Este trabajo se desarrolla dentro del área multimodal en la que se realizan proyectos relacionados con el procesado de señales multimedia, en concreto dentro de la línea de Video Analytics, dónde se aplican técnicas de análisis de video obtenido desde cámaras multiespectrales que permite, por ejemplo,

la detección y seguimiento de objetos para, aportar soluciones para la vigilancia y monitorización en entornos aéreos, marítimos o en tierra que ayudan a tomar decisiones más rápidas y eficaces.

1.4. Motivación y objetivos

Desde Gradiant se está desarrollando un sistema de detección y evasión que debe cumplir con los requisitos técnicos y estándares establecidos por diversas autoridades y organismos reguladores europeos y nacionales, como la *European Organisation for Civil Aviation Equipment* (EUROCAE), una organización que desarrolla estándares para la aviación civil; el *Joint Authorities for Rulemaking on Unmanned Systems* (JARUS), un grupo de autoridades de aviación de todo el mundo que trabaja en la creación de regulaciones armonizadas para sistemas no tripulados; la *European Union Aviation Safety Agency* (EASA), la agencia de seguridad aérea de la Unión Europea; y la Agencia Estatal de Seguridad Aérea (AESA), el organismo regulador de la aviación civil en España. Su cumplimiento es esencial para habilitar vuelos de UAV más allá del alcance visual del piloto.

Entre los requisitos propuestos por las autoridades se incluye la implementación de un sistema de detección y evasión (DAA) capaz de detectar y tomar decisiones ante situaciones en las que el dron sea susceptible a tener una colisión con otros elementos en el espacio aéreo, utilizando exclusivamente la información extraída de las cámaras equipadas en el UAV. Para lograrlo, se necesita una estimación en tiempo real del tamaño y las trayectorias de los objetos voladores que se pueda encontrar a partir de las imágenes capturadas por las cámaras.

Esta predicción implica detectar los objetos voladores y clasificarlos según el tipo de aeronave (dron, avion, helicóptero, etc.) de una forma detallada y confiable, en este caso, utilizando redes neuronales de detección y clasificación. Sin embargo, el entrenamiento de estas redes para cubrir todos los posibles escenarios es complejo, por lo que resulta fundamental identificar la presencia en el aire de otros elementos no contemplados durante el entrenamiento, como pueden ser un ala delta, aves, etc., para alertar adecuadamente al piloto y garantizar la seguridad del vuelo.

Además de la detección y clasificación de objetos voladores, el diseño y entrenamiento del sistema debe considerar las condiciones específicas del entorno operacional. Dado que no es factible generalizar todas las condiciones del entorno operacional durante el entrenamiento, es esencial identificar situaciones anómalas que puedan surgir durante la operación del sistema, tales como falta o exceso de luz, presencia de niebla, lluvia u otros eventos inesperados.

Estos requisitos han impulsado el interés en los métodos de detección de imágenes *Out-of-Distribution*, motivando el desarrollo de este trabajo, en el que se propone hacer una revisión y comparación de algunos de los algoritmos del estado del arte más utilizados para este fin.

Con el objetivo de conseguir un método que permita solucionar las problemáticas presentadas, se realizarán pruebas con distintos conjuntos de entrenamiento, que permitan evaluar la eficacia de cada uno de los algoritmos para así poder decidir cuál de ellos se adecúa mejor a los casos de uso presentados. Además, se valorará la demanda de carga computacional y de memoria, dado que el sistema de DAA debe ejecutarse en un procesador embarcado en un dron, en el cual todos los componentes que se añadan deben seguir el principio de tener un bajo SWaP (*Size, Weight and Power*), un concepto ampliamente utilizado en el ámbito de la industria aeroespacial y de seguridad [7], que busca conseguir sistemas pequeños con bajo peso y que consuman poca potencia.

La integración de los métodos seleccionados permitirá mejorar la seguridad durante los vuelos en zonas remotas respecto al piloto, cumpliendo además con uno de los requisitos establecidos por la EASA para la aplicación de técnicas de IA [8]. En particular, se atenderá el requisito DM-07 (*"The applicant should ensure verification of the data, as appropriate, throughout the data management process so that the data management requirements (including the DQRs) are addressed"*), que exige la verificación de que los datos utilizados para la toma de decisiones mediante IA sean adecuados y representativos. En caso de que los datos no cumplan con estas características, el sistema debe ceder el control al piloto.

1.5. Estructura del documento

Habiendo introducido el marco conceptual y el contexto empresarial en el que se sitúa la realización de este trabajo, se procede a detallar la estructura del documento seguida en este estudio.

En el Capítulo 2, se presenta una introducción a las redes neuronales para la clasificación de imágenes, y a los conceptos relacionados con esta temática que serán necesarios para el desarrollo de los algoritmos de detección de imágenes *Out-of-Distribution* (OOD).

A continuación, en el Capítulo 3, se lleva a cabo un análisis detallado del estado del arte actual en la detección de imágenes OOD, con especial atención a los algoritmos más adecuados para cumplir con los requisitos previamente mencionados.

Finalmente se exponen, en el Capítulo 4, los experimentos realizados durante la realización del trabajo, recogiendo las principales ventajas y desventajas de los distintos algoritmos con el fin de exponer, en el Capítulo 5, las conclusiones sobre cuáles de ellos son los adecuados para implementar en el sistema de detección y evasión.

Capítulo 2

Redes Neuronales para la clasificación de imágenes

En este capítulo se realiza una breve introducción a las redes neuronales utilizadas para la clasificación de imágenes, llevando a cabo una explicación general de las redes neuronales convolucionales y presentando dos de las arquitecturas utilizadas durante los experimentos llevados a cabo en este Trabajo de Fin de Máster. Por otro lado se presentan una serie de conceptos relacionados con las redes neuronales convolucionales necesarios para la comprensión de los métodos de detección de imágenes *Out-of-Distribution*.

2.1. Introducción

Las redes neuronales de clasificación de imágenes han emergido como una herramienta poderosa en el análisis de datos visuales, ofreciendo una capacidad sin precedentes para interpretar y categorizar información visual de manera automatizada. Estas redes representan una aplicación sofisticada de técnicas de aprendizaje automático que se fundamentan en modelos probabilísticos y métodos de optimización. En concreto para la tarea de clasificar imágenes se usan las redes neuronales convolucionales (CNN) [9], que siguen la siguiente estructura (ver Figura 2.1):

- Capas convolucionales: Las CNN están compuestas principalmente por capas convolucionales. Cada capa convolucional aplica un conjunto de filtros a la entrada para extraer características específicas de la imagen, como bordes, texturas o formas. Estos filtros se aplican sobre la imagen realizando operaciones de convolución.
- Funciones de activación: Después de cada operación de convolución, se aplica una función de activación, típicamente ReLU (Rectified Linear Unit), para introducir no linealidades en la red y permitir el modelado de relaciones más complejas entre las variables de entrada y salida.
- Capas de agrupación: Las capas de agrupación (*pooling*) se emplean para disminuir la dimensionalidad de las variables generadas por las capas convolucionales, lo que facilita la gestión de la representación de los datos y disminuye el número de parámetros en la red neuronal. Entre las operaciones de agrupación más comunes se encuentra el *max-pooling*. En el *max-pooling*, se divide la entrada en regiones pequeñas y, de cada una de estas regiones, se selecciona el valor máximo. Esto permite conservar las características más destacadas y relevantes de cada región. Otra técnica común es el *average-pooling*, en la cual se calcula el valor promedio de todos los valores dentro de cada región, suavizando la representación pero conservando la información general.
- Capas directamente conectadas: Después de varias capas convolucionales y de agrupación, la red suele incluir una o más capas directamente conectadas. Estas capas tienen conexiones entre

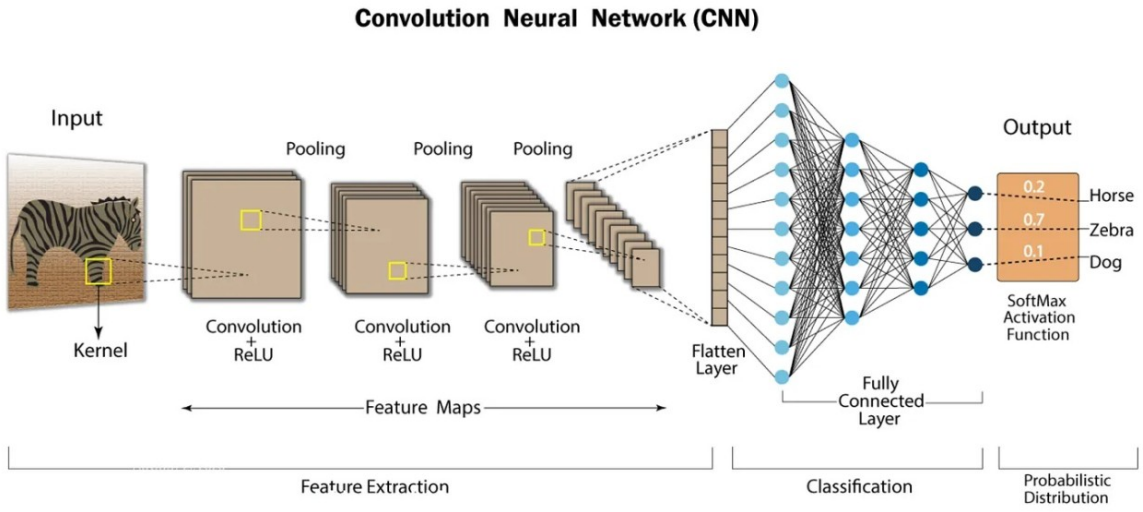


Figura 2.1: Esquema explicativo de la arquitectura de las redes neuronales convolucionales utilizadas en la clasificación de imágenes. Fuente: [10]

todas las neuronas de la capa anterior y la capa siguiente, lo que permite que la red combine las características extraídas para realizar la clasificación final.

- Capa de salida: La capa de salida de una CNN usualmente consiste en una capa directamente conectada seguida de una función de activación adecuada para el tipo de problema. En el caso de la clasificación multiclase, se utiliza la función de activación *softmax*, que produce un vector de probabilidades. Este vector se interpreta como una estimación de la probabilidad condicional $P(y|x)$, donde y representa la clase de la imagen y x representa la imagen de entrada.

De manera resumida, el entrenamiento de redes neuronales consite en el ajuste de los parámetros de la red con el objetivo de minimizar una función de pérdida. Esta función de pérdida cuantifica qué tan lejos está la salida predicha de la salida real (la etiqueta verdadera) para cada elemento del conjunto de entrenamiento. Una métrica de pérdida empleada habitualmente en problemas de clasificación es la entropía cruzada. Además, se usa un algoritmo de optimización llamado descenso de gradiente estocástico (SGD) para ajustar los pesos de las conexiones entre las neuronas de la red de manera que la pérdida se minimice [11].

2.2. ResNet

ResNet (*Residual Network*) es una arquitectura de red neuronal convolucional introducida por He et al. [12], que aborda el problema del desvanecimiento del gradiente en redes profundas mediante el uso de bloques residuales. El problema del desvanecimiento del gradiente ocurre cuando los gradientes que se propagan hacia atrás a través de la red se vuelven muy pequeños, dificultando el ajuste de los parámetros y, por ende, el aprendizaje efectivo en capas profundas. Los bloques residuales de ResNet permiten que las capas aprendan funciones residuales respecto a la entrada, facilitando el entrenamiento de redes profundas sin degradar el rendimiento.

Una función residual se refiere a la idea de que, en lugar de aprender directamente el mapeo deseado $H(x)$, la red aprende la función residual $F(x) = H(x) - x$. Al reescribir el mapeo original como $H(x) = F(x) + x$, se facilita el aprendizaje porque es más sencillo ajustar los parámetros cuando el modelo solo necesita aprender las diferencias (residuales) respecto a la identidad.

ResNet-18, una versión más ligera de la familia de redes residuales, consta de 18 capas organizadas en bloques residuales. Su estructura se puede observar en la Figura 2.2, esta incluye una capa de entrada, cuatro conjuntos de bloques residuales con capas convolucionales de 64, 128, 256 y 512 filtros respectivamente, y una capa final de promedio global seguida de una capa directamente conectada para la clasificación. ResNet-18 se destaca por su facilidad de entrenamiento, eficiencia computacional y buen rendimiento, siendo adecuada para aplicaciones con recursos limitados.

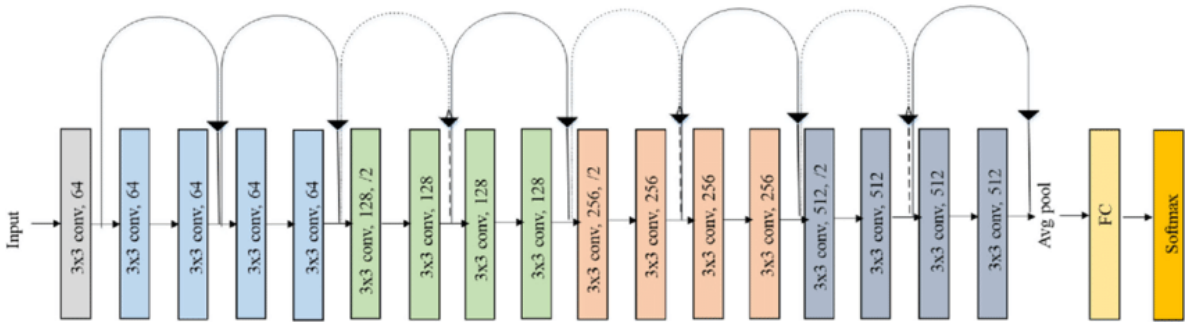


Figura 2.2: Representa la estructura de la red neuronal convolucional de clasificación de imágenes ResNet-18. Fuente: Farheen Ramzan et al. (2019) [13]

2.3. WideResnet

WideResNet (Wide Residual Networks) es una variante de las redes residuales (ResNet) que amplía significativamente el ancho de las capas residuales, mejorando el rendimiento sin aumentar proporcionalmente la profundidad de la red. Introducida por Sergey Zagoruyko y Nikos Komodakis [14], WideResNet aborda algunas limitaciones de las redes profundas tradicionales al incrementar el número de filtros en cada capa convolucional, lo que permite capturar más características detalladas con menos capas.

Al igual que ResNet, WideResNet utiliza bloques residuales, pero estos bloques son más anchos en términos de número de filtros. Este enfoque se basa en la observación de que aumentar el ancho de las capas puede ser más beneficioso que simplemente incrementar la profundidad, ya que redes excesivamente profundas pueden ser difíciles de entrenar y más propensas a problemas como el desvanecimiento del gradiente.

En una arquitectura WideResNet típica, se siguen los principios básicos de ResNet pero con una mayor amplitud en cada bloque residual. La estructura incluye una capa de entrada, seguida de varios bloques residuales, donde cada bloque consiste en capas convolucionales más anchas que en ResNet. Además, cada bloque está compuesto por dos o tres capas convolucionales, seguidas de normalización por lotes y una función de activación ReLU, con conexiones residuales que permiten la suma directa de la entrada y la salida del bloque.

WideResNet se destaca por su capacidad para ofrecer un equilibrio entre rendimiento y eficiencia computacional. Aunque utiliza más parámetros que una ResNet de la misma profundidad, su estructura más ancha permite que el entrenamiento sea más rápido y efectivo, ya que reduce la profundidad necesaria para alcanzar un alto rendimiento. Esto la hace particularmente adecuada para aplicaciones donde la precisión es crítica y se dispone de recursos computacionales adecuados.

2.4. Gradiente

El gradiente en las redes neuronales se refiere a la derivada parcial de la función de pérdida con respecto al espacio de características, que es el conjunto de parámetros generados por el modelo para

describir las entradas. En el contexto de la clasificación de imágenes, la función de pérdida generalmente se define para medir la discrepancia entre las etiquetas reales de las imágenes y las predicciones hechas por la red neuronal.

El algoritmo más utilizado para el calculo del gradiente es el algortimo de retropropagación ('Back-propagation') [15]. Funciona en dos fases clave: propagación hacia adelante y retropropagación del error. Durante la propagación hacia adelante, los datos de entrada se introducen en la red, y las activaciones se calculan capa por capa hasta obtener la salida. Esta salida se compara con los valores reales para calcular la pérdida. En la fase de retropropagación, el error calculado se propaga de vuelta a través de la red, capa por capa, utilizando la regla de la cadena para determinar cómo cada peso contribuye al error total. Los pesos se ajustan en función de estos gradientes para minimizar la pérdida en iteraciones sucesivas, afinando así el rendimiento de la red neuronal. Adicionalmente, en algunos de los algoritmos utilizados en este trabajo se necesitará obtener el signo del gradiente en la imagen, que indica en qué dirección debería modificarse el valor de cada pixel de la imagen para acercarse a la etiqueta especificada.

2.5. Data augmentation

El aumento de datos (*data augmentation*, en inglés) es una técnica utilizada en el campo del aprendizaje automático, especialmente en problemas de visión por computadora como la clasificación de imágenes. Consiste en aplicar transformaciones aleatorias y controladas a las imágenes de entrenamiento existentes para crear nuevas instancias de datos. Estas instancias generadas tienen características similares a las originales pero presentan variaciones en términos de rotación, traslación, escala, brillo, entre otros aspectos [16].

2.6. Métricas de evaluación

Con el fin de evaluar la eficacia de los algoritmos de detección de imágenes OOD, se utilizan métricas clásicas ampliamente usadas en estadística o aprendizaje automático en la evaluación de modelos de clasificación binaria. En primer lugar, en este Trabajo Fin de Máster se consideran como casos positivos las imágenes *In-Distribution* y como casos negativos las imágenes *Out-of-Distribution*. Siguiendo esta clasificación las medidas utilizadas serán:

- Verdaderos positivos (TP): Son las imágenes *In-Distribution* (casos positivos) que el algoritmo clasifica correctamente como *In-Distribution*.
- Falsos positivos (FP): Son las imágenes *Out-of-Distribution* (casos negativos) que se clasifican incorrectamente como *In-Distribution*.
- Verdaderos negativos (TN): Son las imágenes *Out-of-Distribution* (casos negativos) que se clasifican correctamente como *Out-of-Distribution*.
- Tasa de verdaderos positivos (TPR): También conocida como sensibilidad o *recall*, mide la proporción de TP (*In-Distribution*) que son correctamente identificados por el algoritmo.

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

- Tasa de falsos positivos (FPR): Mide la proporción de casos negativos reales (*Out-of-Distribution*) que son incorrectamente identificados como positivos (FP)(*In-Distribution*)

$$FPR = \frac{FP}{FP + TN}$$

Además, se ha considerado como métrica el área bajo la curva ROC (AUCROC) [17]. El AUCROC representa la probabilidad de que un clasificador ordene una instancia positiva elegida al azar más alto que una instancia negativa elegida al azar. En otras palabras, mide la capacidad del modelo para distinguir entre clases positivas y negativas. Por ejemplo, un AUCROC de 0.5 indica que el modelo no tiene capacidad discriminativa, similar a una clasificación aleatoria, mientras que un AUCROC de 1.0 indica una capacidad perfecta de discriminación. Esta métrica es especialmente útil en contextos donde las clases están desbalanceadas, proporcionando una evaluación robusta de la habilidad del modelo para distinguir correctamente entre las clases. Además proporciona una visión completa del rendimiento del clasificador, ya que no depende del valor que define la separación entre clases.

Capítulo 3

Estado del Arte

Una vez establecida una base de conocimiento sobre las redes neuronales, y ciertos conceptos derivados de ellas, se realiza una revisión del estado del arte de los algoritmos de detección de imágenes *Out-of-Distribution*, comentando una visión general de la actualidad de esta temática y exponiendo detalladamente cada uno de los algoritmos con los que se realizaron experimentos, debido a su adecuación al contexto en el que se van a aplicar.

3.1. Detección Out-of-Distribution

El término reconocimiento de imágenes fuera de distribución (*Out-of-Distribution*, OOD, por sus siglas en inglés) fue introducido por primera vez en 2017. Desde entonces, ha captado una atención creciente por parte de la comunidad investigadora. La falta de una definición clara y unívoca del término ha propiciado el surgimiento de diversas líneas de investigación dentro de este campo, recogidas de una forma detallada en el trabajo de Jingkang Yang et al. [18] (ver Figura 3.1). Entre estas líneas destacan la detección de anomalías, la detección de novedades, la detección de novedades multiclase, la detección de imágenes *Out-of-Distribution*, y la detección de atípicos, cada una abordando diferentes aspectos y desafíos asociados con los datos OOD.

La detección de anomalías [19] se enfoca en identificar desviaciones respecto a la distribución de un conjunto de imágenes, tales como cambios de estilo o características visuales inusuales. Por ejemplo, en un conjunto de fotos de perros, una anomalía sería detectar un dibujo hecho a mano de un perro.

La detección de novedades [20] busca reconocer imágenes cuya clase es diferente de las presentes en el conjunto de imágenes inicial. Por ejemplo, si el conjunto está compuesto por fotos de perros, una novedad sería detectar una imagen de un gato. Una vertiente de esta línea es la detección de novedades multiclase, que tiene el mismo objetivo pero aplicado a conjuntos de imágenes que contienen múltiples clases.

La detección de imágenes *Out-Of-Distribution* (OOD) comparte el objetivo de la detección de novedades multiclase. Sin embargo, los algoritmos desarrollados en esta línea se diferencian de las demás porque son capaces de predecir la clase de las imágenes.

Finalmente, la detección de imágenes atípicas [21] explora técnicas para identificar imágenes cuyas clases o características se desvían considerablemente de la mayoría de las imágenes del conjunto inicial. Esta línea se centra en detectar casos extremos dentro de la distribución general de los datos de entrenamiento.

Como ya se ha mencionado, en el sistema de detección y evasión es esencial clasificar el entorno en el que se encuentra el UAV, asegurando que no surjan situaciones anómalas y clasificando correctamente los objetos voladores para evitar dar información errónea sobre ellos. Debido a esto, se considera que la línea de investigación más adecuada entre las presentadas es la detección de imágenes *Out-of-Distribution* (OOD), por lo que se entrará más en detalle en ella, realizando un análisis más formal

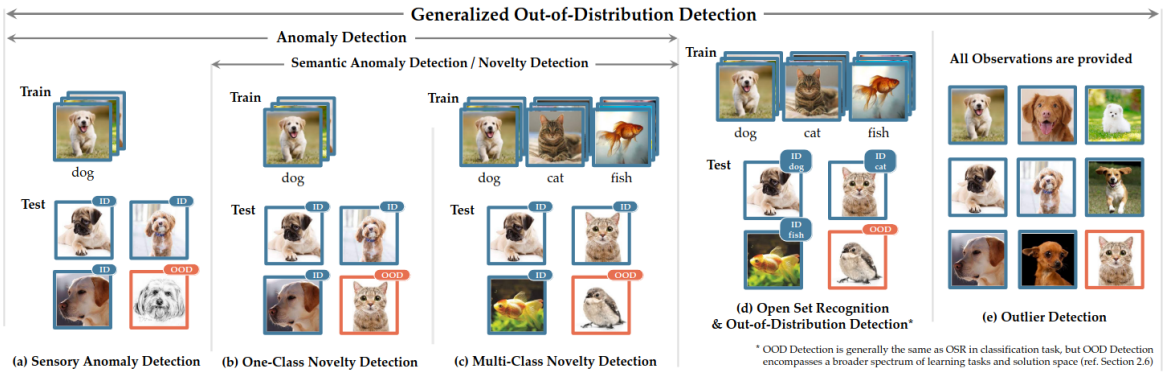


Figura 3.1: Líneas de investigación derivadas de la detección de imágenes *Out-of-Distribution*. a) Detección de anomalías b) Detección de novedades. c) Detección de novedades multiclase. d) Reconocimiento de conjunto abierto o detección OOD. e) Detección de atípicos. Fuente: [18]

del conjunto de técnicas recogidas en esta línea.

Para el estudio de la detección de imágenes *Out-of-Distribution* es necesario conocer de antemano que una distribución de datos usados en la clasificación de imágenes se puede definir como una función de probabilidad que describe cómo se distribuyen los datos de imagen en el espacio de características. Esta distribución incluye información sobre la frecuencia y la variabilidad de las diferentes clases de imágenes y sus características visuales.

Consideremos un conjunto de datos de imágenes $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, donde x_i representa una imagen y y_i su etiqueta de clase correspondiente, con N siendo el número total de ejemplos en el conjunto de datos. La distribución de estos datos se puede definir de la siguiente manera:

- Espacio de Características: Cada imagen x_i se representa como un punto en un espacio de características $\mathcal{X}$. Por lo tanto, se trata de un espacio de alta dimensionalidad, que puede reducirse gracias a un procesamiento previo, como la extracción de características [22] mediante una red neuronal convolucional.

- Etiquetas de Clase: Las etiquetas y_i pertenecen a un conjunto finito de clases $\mathcal{Y} = \{1, 2, \ldots, C\}$, donde C es el número total de clases.

- Distribución de Probabilidad Conjunta: La distribución conjunta de las imágenes y sus etiquetas se denota como $P(X, Y)$, donde X es una variable aleatoria que representa las imágenes e Y es una variable aleatoria que representa las etiquetas de clase. Esta distribución conjunta describe la probabilidad de observar una imagen particular y su etiqueta correspondiente.

- La distribución marginal $P(X)$: describe cómo se distribuyen las imágenes en el espacio de características sin considerar sus etiquetas.

- La distribución condicional $P(Y|X)$: describe la probabilidad de que una imagen X pertenezca a una clase particular Y.

- Independencia e Identidad de Distribución (i.i.d.): En la mayoría de los casos de aprendizaje supervisado, se asume que los datos de entrenamiento son muestras independientes e identicamente distribuidas (i.i.d.) según la distribución conjunta $P(X, Y)$.

Un modelo de clasificación de imágenes intenta aprender la distribución condicional $P(Y|X)$ a partir del conjunto de datos $\mathcal{D}$, de manera que pueda predecir la clase Y de una nueva imagen X no vista durante el entrenamiento.

En resumen, una distribución de datos usados en la clasificación de imágenes se define formalmente por la distribución conjunta $P(X, Y)$, que encapsula la forma en que las imágenes y sus etiquetas se relacionan en el espacio de características y cómo están distribuidas las diferentes clases de imágenes.

Conociendo esta definición, en el desarrollo de los algoritmos de detección de imágenes *Out-of-Distribution*, se puede considerar imágenes de fuera de la distribución aquellas que pertenecen a una clase diferente al conjunto finito de clases de entrenamiento $\mathcal{Y}$.

De acuerdo con los objetivos de este trabajo, se revisaron métodos que no implicasen la expansión de la arquitectura del modelo base, es decir, que no requieren la adición de capas adicionales al modelo de red neuronal ya establecido. También se evitaron métodos que implicasen la incorporación de modelos adicionales o auxiliares, como modelos secundarios que trabajen en paralelo o en conjunto con el modelo principal para la tarea de clasificación. Además, se excluyeron métodos que requieren realizar múltiples inferencias para una sola entrada, lo que puede incrementar significativamente el tiempo de computación y los recursos necesarios.

La investigación evolucionó partiendo de distintos enfoques, buscando discriminar de la forma más efectiva las imágenes de fuera de distribución. La primera aproximación que se toma como referencia para la detección de imágenes OOD es usar la probabilidad máxima del vector de probabilidades que devuelve la capa de salida *softmax* como puntuación que cuantifique la pertenencia a la distribución de entrenamiento. A esta puntuación la llamaremos puntuación OOD.

Por lo tanto, si se considera una red neuronal de clasificación de imágenes, denotada como $\hat{f} = (\hat{f}_1, ..., \hat{f}_C)$ la cual está entrenada para clasificar en C clases, para cada imagen de entrada x, la puntuación OOD se define como $S_{\hat{y}} = argmax_i\ S_i(x)$, dónde $S_i(x)$ es la función *softmax* definida por:

$$S_i(x) = \frac{exp(\hat{f}_i^{-1}(x))}{\sum_j^N exp(\hat{f}_j^{-1}(x))}.$$

En esta formulación $\hat{f}_i^{-1}(x)$ denota la salida del modelo para la clase i antes de aplicar la función *softmax*, es decir, la penúltima capa del modelo. Por otro lado, $S_i(x)$ es la probabilidad asignada a cada clase i después de aplicar la *softmax*. Finalmente, la puntuación OOD $S_{\hat{y}}$ se obtiene seleccionando la mayor probabilidad de las estimadas por la función *softmax*.

Con la puntuación OOD de cada imagen de entrenamiento se puede definir un límite δ, a partir del cual una imagen es clasificada como ID (0) o OOD (1), este clasificador binario $g(x; \delta)$ se puede definir como:

$$g(x;\delta) = \begin{cases} 1 & \text{si } max_i\ S_i(x) \leq \delta \\ 0 & \text{si } max_i\ S_i(x) > \delta. \end{cases}$$

Para elegir el parámetro δ se escoge el valor que incluye como correctamente clasificadas el 95 % de las imágenes *in-distribution* utilizadas en el entrenamiento del modelo, es decir, se escoge el δ que da lugar a una tasa de verdaderos positivos del 95 %.

Esta idea inicial no aporta información suficiente para discriminar las imágenes de entrada, porque la salida de la capa *softmax* acostumbra a devolver valores de confianza en las predicciones muy elevados, incluso para imágenes generadas aleatoriamente con ruido gaussiano [23]. Sin embargo, en el estado del arte se usa como método de referencia para demostrar y comparar la capacidad de detección de imágenes OOD.

La escasa capacidad de detectar imágenes *Out-of-Distribution* que tiene el método de referencia promueve la busca de métodos que lo mejoren. Por ello, en el estado del arte actual se proponen una serie de algoritmos que incorporan distintos procedimientos con el objetivo de conseguir que la puntuación devuelta por los algoritmos sea facilmente separable entre la de las imágenes ID y las OOD y así mejorar la clasificación. A continuación, se muestra un pequeño resumen de estos procedimientos. Para una revisión detallada, consultar Jingkang Yang et al. [18]:

- Modificaciones en la salida de la red: Sencillos cálculos que se realizan con los valores de salida de la red que permiten aumentar la separación entre las puntuaciones ID de las OOD.

- Métodos basados en gradiente: Los enfoques existentes para la detección de OOD se basan principalmente en el espacio de características para calcular puntuaciones de OOD, en cambio estos métodos incorporan la información del gradiente.

- Métodos basados en densidad: Los métodos basados en la densidad modelan explícitamente la distribución de entrenamiento con algunos modelos probabilísticos, y marcan los datos de prueba en regiones de baja densidad como OOD.

- Métodos basados en la distancia: La idea básica de los métodos basados en la distancia es que las muestras OOD deben estar relativamente lejos de los centroides o prototipos de las clases en distribución.

- Exposición a datos atípicos: Algunos métodos hacen uso de un conjunto de muestras de OOD, aplicadas durante el entrenamiento para ayudar a los modelos a aprender la discrepancia ID/OOD.

- Generación de datos OOD: Los enfoques de exposición a valores atípicos imponen una fuerte suposición en la disponibilidad de datos de entrenamiento OOD, que puede ser inviable en la práctica. Cuando no se dispone de muestras OOD se pueden sintetizar muestras OOD para permitir separabilidad ID/OOD. Los trabajos existentes aprovechan GANs para generar muestras de entrenamiento OOD.

Además de estos procedimientos, el artículo previamente mencionado [18] recoge otros enfoques, tales como el uso de modelos auxiliares diseñados para diferenciar entre imágenes ID y OOD, así como métodos bayesianos [24]. No obstante, en el presente trabajo no se consideran estos métodos, dado que todos ellos requieren un número considerable de iteraciones de inferencia, es decir, necesitan realizar numerosas predicciones, ya sea utilizando un único modelo o varios modelos. Esta exigencia incrementa significativamente la complejidad computacional y el tiempo de procesamiento.

3.2. Algoritmos utilizados

Tras una revisión inicial de los distintos enfoques para la detección de imágenes OOD disponibles en la literatura, se presentan a continuación los algoritmos seleccionados y utilizados en este trabajo.

3.2.1. ODIN

Uno de los algortimos más sencillos en el estado del arte actual es el propuesto por Shiyu Liang et al. [25]. Este detector de imágenes OOD, denominado *ODIN*, promete mejorar al método de referencia utilizando los dos componentes que se describen a continuación.

Temperature Scaling

La modificación *Temperature Scaling* consiste en aplicar a la función *softmax* ya mencionada un parámetro de escalado $T \in R^+$, así la puntuación OOD sería $S_{\hat{y}}(x;T) = argmax_i\ S_i(x;T)$, dónde $S_i(x;T)$ es

$$S_i(x,T) = \frac{exp(\hat{f}_i^{-1}(x)/T)}{\sum_j^N exp(\hat{f}_j^{-1}(x)/T)}.$$

Según trabajos previos [26], el uso de esta modificación permite calibrar la confianza de las predicciones en las tareas de clasificación y tiene como efecto, en este caso, separar de una forma más efectiva las puntuaciones entre las imágenes *In-Distribution* (ID) y *Out-of-Distribution*(OOD).

Input Preprocessing

Además del *Temperature Scaling*, este algoritmo incorpora un método basado en el gradiente, que consiste en añadir una pequeña perturbación a la imagen de entrada calculada utilizando la información que aporta el gradiente. Denotando a la entrada como x, la imagen resultado sería:

$$\tilde{x} = x - \varepsilon \; sign(-\nabla_x log S_{\hat{y}}(x;T)),$$

dónde ε es la magnitud de la perturbación y $sign(-\nabla_x log S_{\hat{y}}(x;T))$ el signo del gradiente de la imagen.

Este método se inspira en la idea de los ataques adversarios [27], dónde pequeñas perturbaciones en la imagen pueden afectar a la puntuación que devuelve el *softmax*, llegando a realizarse predicciones incorrectas. En este caso se pretende realizar el efecto contrario, incrementar esa puntuación con respecto a la de la predicción original. Esto resulta de interés en la detección de imágenes OOD porque estas pequeñas perturbaciones tienen mayor efecto en imágenes *In-Distribution* que en imágenes *Out-of-Distribution*.

Para construir el algoritmo *ODIN* se combinan los dos componentes anteriores de la siguiente forma. Primero se obtiene la imagen $\tilde{x}$ de acuerdo a la expresión presentada en el apartado *Input Preprocessing*, a continuación, con la imagen preprocesada se realiza inferencia en la red neuronal calculando el *softmax* aplicando la modificación *Temperature scaling*, así obtenemos la puntuación OOD.

Finalmente, la elección de los parámetros ε, T se realiza aplicando al algoritmo a un conjunto de imágenes consideradas como OOD y eligiendo la combinación que consiga una menor tasa de falsos positivos.

Cabe mencionar que *ODIN* es uno de los algoritmos más sencillos que mejora notablemente al de referencia en cuanto a detección de imágenes OOD. Su sencillez viene dada por la posibilidad de ser aplicado a un modelo previamente entrenado y además no requiere un gasto computacional excesivamente mayor, ya que el cálculo de la puntuación OOD solo implica realizar una predicción a mayores de la etapa de clasificación.

3.2.2. Mahalanobis

En Kimin Lee et al. [28] se propone un algoritmo basado en distancias con el objetivo de obtener una puntuación OOD más separable.

Considerando el uso de redes neuronales cuya última capa es una función *softmax*

$$S_i(x) = \frac{exp(\hat{f}_i^{-1}(x))}{\sum_j^N exp(\hat{f}_j^{-1}(x))},$$

se demuestra que las características aprendiadas por la penúltima capa de la red neuronal $\hat{f}^{-1}(x)$ provocan que la salida de esta sigan una distribución Gaussiana condicionada a la clase $\hat{y}$ al realizar predicciones con el conjunto de imágenes de entrenamiento. Dentro de esta distribución, los valores con menor probabilidad se corresponden a las imágenes con características menos frecuentes dentro de la distribución.

Partiendo de esta característica de las redes neuronales de clasificación de imágenes, se estiman los parámetros de esta distribución Gaussiana condicionada por cada clase (c) mediante el método de máxima verosimilitud y considerando las imágenes del conjunto de entrenamiento $[(x_1, y_1), ..., (x_N, y_N)]$ como variables de entrada, de este modo la media $\hat{\mu}$ y covarianza $\hat{\sigma}$ serían:

$$\hat{\mu}_c = \frac{1}{N_c} \sum_{i:y_i=c} \hat{f}^{-1}(x_i),$$

$$\hat{\sigma} = \frac{1}{N} \sum_c \sum_{i:y_i=c} (\hat{f}^{-1}(x_i) - \hat{\mu}_c)(\hat{f}^{-1}(x_i) - \hat{\mu}_c),$$

dónde N_c es el número de muestras con etiqueta de clasificación c.

Teniendo esto en cuenta, se define la puntuación OOD $M(x)$, como la distancia de *Mahalanobis* entre la muestra de test x y la distribución gaussiana condicionada a la clase más cercana:

$$M(x) = \max_c - (\hat{f}^{-1}(x) - \hat{\mu}_c)^T \hat{\sigma}^{-1} (\hat{f}^{-1}(x) - \hat{\mu}_c).$$

Además, se utiliza un preprocesado de las imágenes de entrada (*Input Preprocessing*), con el mismo objetivo, hacer que la puntuación OOD sea más separable entre las imágenes *In-Distribution* y las *Out-of-Distribution*. Este prepocesado se obtiene de la siguiente forma:

$$\hat{x} = x + \varepsilon sign(\nabla_x M(x)),$$

dónde ϵ es la magnitud del ruido al aplicar a la imagen, que debe ser el valor que produzca una menor tasa de falsos positivos al aplicar el método con un conjunto de imágenes OOD.

Este algoritmo mejora considerablemente los métodos anteriormente presentados y además tiene un proceso de aplicación similar al mencionado en el algoritmo *ODIN* (apartado 3.2.1). Por lo tanto se puede emplear en un modelo ya entrenado y computacionalmente sólo necesita una etapa de inferencia a mayores de la etapa de clasificación. Como característica adicional, este método promete detectar imágenes modificadas mediante ataques adversarios permitiendo mejorar la seguridad y robustez de los sistemas que lo incorporen.

3.2.3. Generalized ODIN

Este algoritmo presentado en Yen-Chang Hsu et al.[29] surge como evolución del propuesto en [25], con el objetivo de evitar usar imágenes OOD para ajustar los hiperparámetros T y ε.

Un motivo por el cual los clasificadores *softmax* clasifican las observaciones OOD con un exceso de confianza es porque estos asumen que el el conjunto de datos de entrenamiento es capaz de generalizar todos los datos observados durante una predicción. Sin embargo, esto no es lo habitual, entonces en este algoritmo se decide construir un clasificador que tenga en cuenta un variable binaria d_{in} que indique si la imagen para la que se está realizando la predicción tiene características similares a las utilizadas en el entrenamiento, es decir, si pertenece a la distribución de entrenamiento.

Por ello, se propone realizar la clasificación teniendo en cuenta el siguiente cociente, denominado *Decomposed Confidence*:

$$P(y|d_{in}, x) = \frac{P(y, d_{in}|x)}{P(d_{in}|x)}.$$

Para comprender las relaciones que se establecen al modelar la probabilidad de clasificación de esta forma se considera como ejemplo una imagen OOD x. Es esperable que la probabilidad $P(y, d_{in}|x)$ sea baja (ej. 0.09) para la clase con más confianza. Además, tambien se espera que la probabilidad $P(d_{in}|x)$ sea baja (ej. 0.1). Por lo tanto, calculando $P(y|d_{in}, x)$ se obtiene una probabilidad alta (ej. 0.9).

Teniendo en cuenta esta característica, el trabajo citado en esta sección propone diseñar un clasificador capaz de obtener en la predicción los valores de $P(d_{in}|x)$ o $P(y|d_{in}, x)$, que son capaces de caracterizar la pertenencia o no a la distribucion de entrenamiento.

La primera aproximación para calcular la probabilidad conjunta $p(y, d_{in}|x)$ es enseñar a un clasificador a hacerlo teniendo supervisión de la etiqueta y y del dominio d. En este caso durante el entrenamiento del modelo, se calcula la pérdida a partir de la etiqueta de la imagen y su variable indicadora d_{in} que define la imagen como ID o OOD. Sin embargo, este enfoque implica tener imágenes OOD, algo no deseable para un modelo que se va aplicar en un contexto de mundo abierto. Continuando la idea de *Decomposed Confidence*, se propone una nueva estructura dividendo/divisor para la última capa de la red neuronal de clasificación, que tendrá la siguiente forma:

$$f_i(x) = \frac{h_i(x)}{g(x)}.$$

En este caso la red se optimiza según la salida $f_i(x)$, normalizada con la función *softmax*. Por lo tanto, la función de pérdida puede ser minimizada de dos formas, incrementando $h_i(x)$ o disminuyendo $g(x)$. Teniendo en cuenta esto, se puede deducir de la siguiente manera que $h_i(x)$ y $g(x)$ tienen un comportamiento similar a $p(y, d_{in}|x)$ y $p(d_{in}|x)$, respectivamente. Por ejemplo, si una imagen pertenece a una región de baja densidad de la distribución de entrenamiento $h_i(x)$ tiende a valores pequeños, entonces $g(x)$ está obligado a ser tambien pequeño para tratar de minimizar la función de pérdida. Por lo contrario, si tenemos en cuenta una imagen de una región de alta densidad $h_i(x)$ va a conseguir valores más elevados y por consecuencia, $g(x)$ va a conseguir un valor elevado más fácilmente, este efecto se puede ver en la Figura 3.2.

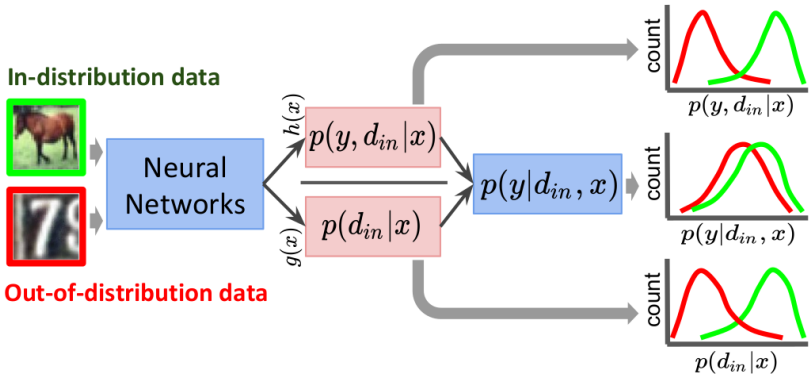


Figura 3.2: Efecto de incluir la variable d_{in} en una red neuronal de clasificación en la que se calcula la puntuación de salida utilizando el concepto de textitDecomposed Confidence. Fuente: [[29]]

Partiendo de que $\hat{f}^{-1}(x)$ es la salida de la penúltima capa de una red neuronal y $\hat{w}$ y $\hat{b}$ representan los parámetros que aprende, se sustituye la capa *softmax* por la ecuación *Decomposed Confidence*, dónde $\hat{g}(x) = \sigma(BN(\hat{w}_g\hat{f}^{-1}(x) + \hat{b}_g))$, siendo BN la aplicación de la técnica de normalización por lotes y σ, la función sigmoide. Por otro lado, para $h_i(x)$ en [29] se proponen tres medidas diferentes:

- Producto interior:
$$\hat{h}_i^I(x) = \hat{w}_i^T\hat{f}^{-1}(x) + \hat{b}_i.$$
- Distancia euclidea:
$$\hat{h}_i^E(x) = -||\hat{f}^{-1}(x) - \hat{w}_i||^2.$$
- Similitud coseno:
$$\hat{h}_i^C(x) = \frac{\hat{w}_i^T\hat{f}^{-1}(x)}{||\hat{w}_i||||\hat{f}(x)||}.$$

Teniendo en cuenta esto se entrena la red modificada utilizando como predicciones la salida que aporta $\hat{f}_i^{-1}(x)$ y en las etapas de test la predicción se puede hacer tanto usando $argmax_i\hat{f}_i^{-1}(x)$ como $argmax_i\hat{h}_i(x)$ ya que ambas devolveran la misma predicción. Y para la detección de imágenes *Out-of-Distribution* se tomará como puntuación OOD $S(x) = max_i\hat{h}_i(x)$ o $\hat{g}(x)$

Además en este algoritmo tambien se propone realizar el preprocesado de las imágenes de la misma forma que los algoritmos anteriores:

$$\hat{x} = x - \varepsilon sign(-\nabla_x S(x)).$$

Sin embargo, la selección del parámetro que define la magnitud de la perturbación de las imágenes se escoge de entre una rejilla de valores el que maximiza la suma de las puntuaciones OOD de las imágenes del conjunto de validación D_{val}:

$$\varepsilon^* = \underset{e}{argmax} \sum_{x \in D_{val}} .$$

A pesar de ser un algoritmo que requiere entrenar el modelo con las modificaciones mencionadas, es una propuesta muy interesante por no necesitar datos OOD para calibrar los hiperparámetros. Además, de la misma forma que los algoritmos ya presentados, únicamente necesita realizar dos predicciones para obtener la etiqueta de la imagen y la puntuación OOD que permita detectar las imágenes *Out-of-Distribution*.

3.2.4. *Energy-based OOD detection*

La idea de los métodos basados en densidad, es intentar alinear las puntuaciones OOD con la probabilidad de que una imagen sea ID, $P(d_{in}|x)$ y considerar como OOD los datos que ocurren con poca frecuencia. Para llevar a cabo este objetivo, en Weitang Liu et al. [30] se busca replicar el funcionamiento de los modelos basados en energía (EBM) [31].

Un modelo basado en energía consiste en construir una función $E(x) : R^D \rightarrow R$ que relaciona cada punto x de la distribución de entrada a un valor escalar no probabilístico llamado energía.

Según el artículo citado en esta sección, la función de energía libre de Helmholtz ($E(x; \hat{f}^{-1}) = -Tlog \sum_i^k e^{\hat{f}_i^{-1}(x)/T}$.), permite obtener un valor escalar único para cada imagen de entrada x, que se tomará como valor de energía. En la notación anterior hay que tener en cuenta que $\hat{f}^{-1}(x)$ es la salida de la capa previa a la función *softmax* de la red neuronal de clasificación, y T, la constante ya mencionada *Temperature scaling*. Por lo tanto, se considera como puntuación OOD el valor negativo obtenido de la función de energía libre de Helmholtz $-E(x; \hat{f}^{-1})$ de modo que los valores obtenidos de las imágenes *In-Distribution* tienen valores mayores a los *Out-of-Distribution*, como se muestra en la Figura 3.3.

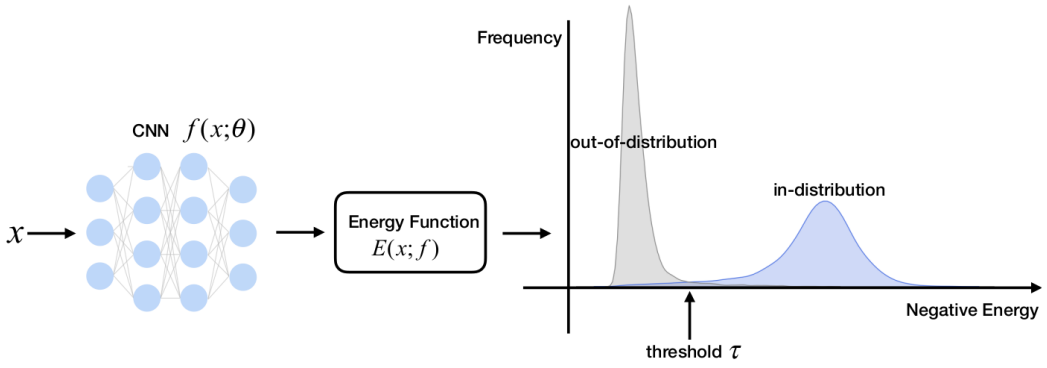


Figura 3.3: Método de detección de imágenes *Out-of-Distribution* basado en energía.

Este método resulta muy interesante por no necesitar datos *Out-of-Distribution*, además solamente se necesita una etapa de predicción, lo que lo sitúa como el método más prometedor para usar aplicaciones en tiempo real.

Aunque este método supone una mejora para conseguir una correcta clasificación de las imágenes OOD/ID, no consigue una diferenciación óptima, por lo sus autores también proponen realizar un nuevo entrenamiento de la red previamente entrenada, en el caso de que se puedan conseguir imágenes OOD reales o generadas con *data augmentation*. A este algoritmo lo denominan *Energy fine-tuned*.

Esta forma de reentrenar la red consiste en especificar durante el aprendizaje unos valores límite, m_{in} y m_{out} con el objetivo de penalizar las muestras *In-Distribution* que producen valores de energía por encima del valor m_{in} y también las muestras *Out-of-Distribution* que producen energías menores que el límite m_{out}. Como veremos en el apartado de experimentos y resultados, este reentrenamiento provoca una mayor separación de los scores OOD calculados de igual forma que con el método sin reentrenamiento.

Capítulo 4

Experimentos y resultados

En el anterior capítulo, se ha llevado a cabo una exhaustiva revisión teórica de los algoritmos más prometedores, evidenciando tanto su implementación como su aptitud para discernir entre imágenes *In-Distribution* y *Out-of-Distribution*, valorando su idoneidad para su integración en el sistema *Detect and Avoid*.

En el presente capítulo se realizan una serie de experimentos con múltiples objetivos. En primer lugar, se pretende verificar la reproducibilidad de las métricas presentadas en los estudios en los que se exponen los algoritmos, así como realizar una comparativa de las mismas con conjuntos de datos de acceso público. En segundo término, se busca identificar cuál de los algoritmos es capaz de detectar con mayor precisión cambios en el entorno operativo. Por último, se aspira a determinar cuál de ellos puede identificar de manera efectiva objetos detectados en el espacio aéreo que no se corresponden con ninguna de las categorías de aeronaves contempladas durante el proceso de entrenamiento.

Con el propósito de tomar decisiones informadas acerca de cuáles de los algoritmos presentados son los más prometedores para los casos de uso específicos abordados, se lleva a cabo una comparación de su capacidad de clasificación entre muestras *In-Distribution* y *Out-of-Distribution*. A tal fin, se han empleado como métricas de evaluación la tasa de falsos positivos cuando la tasa de positivos verdaderos es del 95 % (FPR—TPR95) y el área bajo la curva ROC (AUC).

4.1. Conjuntos de datos públicos

En toda la literatura citada en el capítulo anterior se realizan demostraciones con conjuntos de imágenes públicos, que incluyen:

- CIFAR-10 [32]: Contiene 6000 imágenes a color con tamaño 32x32 píxeles, de cada una de las 10 categorías que conforman este conjunto de datos (avion, coche, pájaro, gato, reno, perro, rana, caballo, barco y camión).
- Imagenet [33]: Este conjunto de datos consta de más de 14 millones de imágenes de más de 20000 categorías diferentes. En este trabajo se reduce el tamaño de las imágenes a 32 x 32 píxeles, para igualar la resolución con los demás conjuntos de datos.
- SVHN [34] : Las imágenes de este conjunto de datos son de tamaño 32x32 pixeles y se compone de imágenes a color de dígitos (del 0 al 9) obtenidas a partir de fotografías de números de casas tomadas por Google Street View.

El primer experimento llevado a cabo fue diseñado de acuerdo a lo expuesto en los artículos del estado del arte, replicando algunas de las pruebas comunes a todos ellos, con el objetivo de comprobar que las métricas presentadas eran reproducibles, y al mismo tiempo compararlos usando conjuntos de imágenes públicos.

En este estudio, se entrena una red WideResNet utilizando el subconjunto de entrenamiento de CIFAR-10. Para la evaluación de las métricas de rendimiento, se consideran 2000 imágenes del subconjunto de validación de CIFAR-10 como datos *In-Distribution* (ID). Además, se utilizan 2000 imágenes de los conjunto de datos SVHN e ImageNet como datos *Out-of-Distribution* (OOD). En la Tabla 4.1, se presentan las métricas de clasificación obtenidas. Estas métricas se comparan con las reportadas en publicaciones previas en las que se introducen los algoritmos comparados, quedando demostrada así la coherencia y la validez de los resultados obtenidos en este trabajo.

	ID	**CIFAR-10**	
	OOD	**Imagenet**	**SVHN**
Baseline	**AUROC**	0.87	0.92
	FPR/TPR95	61.05 %	49.15 %
ODIN	**AUROC**	0.91	0.89
	FPR/TPR95	39.55 %	42 %
Mahalanobis	**AUROC**	0.94	0.97
	FPR/TPR95	34.45 %	17.25 %
Energy	**AUROC**	0.89	0.91
	FPR/TPR95	47.03 %	35.75 %
Gen ODIN	**AUROC**	0.99	0.98
	FPR/TPR95	6.13 %	9.51 %
Energy Fine Tuned	**AUROC**	0.98	0.99
	FPR/TPR95	7.45 %	1.55 %

Cuadro 4.1: Tabla comparativa de las métricas de clasificación de los distintos algoritmos OOD presentados. En el caso del algoritmo *Energy fine-tuned*, el reentrenamiento se realiza utilizando como muestra OOD un subconjunto de las imágenes del conjunto de datos Imagenet.

A pesar de usar conjunto de datos genéricos, no adaptados a un ámbito de aplicación, se pueden extraer las primeras conclusiones sobre el rendimiento de los algoritmos que conforman el estado del arte haciendo una comparación con el método de referencia. Los algoritmos *ODIN*, *Mahalanobis* y *Energy*, que son los algoritmos que utilizan los parámetros de la red neuronal previamente entrenada, consiguen mejorar notablemente la capacidad de detectar las imágenes consideradas OOD, pero no tiene la misma capacidad de detección que los algoritmos *Generalized ODIN* y *Energy fine-tuned*, que

consiguen una tasa de falsos positivos del 6,13 % y 7,45 %, respectivamente, cuando la tasa de falsos positivos es del 95 %.

Adicionalmente, durante este experimento se han extraído métricas de la capacidad computacional, en concreto del incremento del gasto de GPU al usar de estos algoritmos con respecto a las redes neuronales de clasificación. Para ello, se ha calculado el porcentaje de uso de GPU durante el cálculo del *score* OOD de 2000 imágenes de cada uno de los conjunto de datos OOD. Los resultados, presentados en la Tabla 4.2, indican que los métodos basados en energía no incrementan el uso de GPU con respecto al método de referencia, sin embargo, los demás tienen un incremento cercano al doble. Este resultado es el esperado, ya que en estos métodos es necesario relizar dos predicciones para obtener los scores OOD.

	OOD	**Imagenet**	**SVHN**
Baseline	**GPU %**	30.40 %	39.90 %
ODIN	**GPU %**	55.70 %	64.40 %
Mahalanobis	**GPU %**	66.50 %	73.70 %
Energy	**GPU %**	35.30 %	33.80 %
Gen ODIN	**GPU %**	56.9 %	63.7 %
Energy Fine Tuned	**GPU %**	34.9 %	35.6 %

Cuadro 4.2: Porcentajes de uso de GPU con cada uno de los algoritmos seleccionados durante el procesado de 2000 imágenes de cada uno de los conjunto de datos considerados como OOD.

4.1.1. Condiciones de luz

Uno de los objetivos principales al realizar este trabajo fue la detección de situaciones anómalas en el entorno operacional. Durante el vuelo de UAVs, las condiciones de iluminación pueden variar significativamente, por lo que es crucial identificar contextos en los que las imágenes capturadas por la cámara a bordo tengan una luminosidad diferente a las imágenes utilizadas durante el entrenamiento. Con este objetivo en mente, se propone evaluar la capacidad de los algoritmos OOD para detectar estos cambios de brillo en las imágenes, clasificándolas como OOD.

Para este propósito, se ha entrenado una red neuronal de arquitectura WideResNet utilizando el conjunto de datos de entrenamiento de CIFAR-10. Además, se han empleado dos conjuntos de 1000 imágenes cada uno considerados como OOD, generados mediante técnicas avanzadas de aumento de datos. Estas imágenes derivan directamente de CIFAR-10, pero incorporan variaciones significativas en la luminosidad y otros aspectos, diferenciándolas del conjunto de entrenamiento. La eficacia de los algoritmos OOD en la detección de estos cambios permitirá mejorar la capacidad de los UAVs para adaptarse a condiciones de vuelo dinámicas y variables, garantizando así una operación más segura y eficiente.

- CIFAR-10 bright: Imágenes del conjunto de datos CIFAR-10 modificadas aumentado su brillo en distintos porcentajes.
- CIFAR-10 dark: Imágenes del conjunto de datos CIFAR-10 modificadas disminuyendo su brillo en distintos porcentajes.

- CIFAR-10 mixed: Conjunto formado por una mezcla equilibrada de imágenes con las modificaciones de brillo mencionadas en los demás conjunto de datos anteriores.

Con esta configuración del experimento, se han obtenido los resultados presentados en la Tabla 4.3. En ellos se observa que la mayoría de los algoritmos no tienen la capacidad de detectar correctamente las imágenes consideradas OOD, no mejorando prácticamente las métricas obtenidas usando el método establecido como referencia. Sin embargo, el método *Energy fine-tuned* reentrenado con el conjunto de datos CIFAR-10 mixed funciona correctamente, obteniendo una tasa de falsos positivos cuando la tasa de verdaderos positivos es del 95 % muy cercana al 0 %.

	ID	**CIFAR-10**	
	OOD	**CIFAR-10 bright**	**CIFAR-10 dark**
Baseline	**AUROC**	0.68	0.84
	FPR/TPR95	86,6 %	70,15 %
ODIN	**AUROC**	0.68	0.85
	FPR/TPR95	85,35 %	70 %
Mahalanobis	**AUROC**	0.67	0.79
	FPR/TPR95	86,15 %	77,55 %
Energy	**AUROC**	0.69	0.85
	FPR/TPR95	78,95 %	55,9 %
Generalized ODIN	**AUROC**	0.75	0.80
	FPR/TPR95	74,4 %	69,55 %
Energy Fine Tuned (CIFAR-10 mixed)	**AUROC**	0.99	1
	FPR/TPR95	0.95 %	0 %

Cuadro 4.3: Tabla comparativa de las métricas de clasificación de los distintos algoritmos OOD presentados. En el caso del algoritmo *Energy fine-tuned*, el reentrenamiento se realiza utilizando como muestra OOD el conjunto de datos generado con data agumentation CIFAR-10 mixed.

4.2. Conjuntos de datos propiedad de Gradiant

Las imágenes contenidas en los conjuntos de datos públicos son variadas y abarcan múltiples categorías generales, pero no incluyen la especificidad necesaria para entrenar redes neuronales en la detección y clasificación de objetos aéreos como drones, aviones, helicópteros, aves, o alas delta ni de entornos operacionales. Por esta razón, se utilizan conjuntos de imágenes propios de la empresa, que están específicamente diseñados para reflejar los escenarios y objetos relevantes en el espacio aéreo. Esto garantiza que los algoritmos se entrenen y validen con datos representativos y específicos, incrementando así la eficacia y precisión del sistema de *Detect and Avoid*.

4.2.1. Entorno operacional

Tras realizar el experimento anterior en 4.1.1, se ha observado que es posible detectar imágenes con cambios en las condiciones de luz en imágenes de los conjuntos de datos públicos utilizando el algoritmo *Energy Fine-Tuned*, pero también es interesante conocer su rendimiento en un entorno aéreo real y con otro tipo de situaciones, como puede ser la lluvia.

Para ello, se ha tomado un conjunto de imágenes propiedad de Gradiant que consideraremos como ID. Este, contiene 6300 imágenes de espacios naturales y urbanos. Las clases consideradas son cielo con nubes, cielo claro, bosque y urbano (ver Figura 4.1). Como conjunto de datos OOD se ha considerado un subconjunto de 3000 imágenes del anterior pero transformadas usando técnicas de aumento de datos, con el objetivo de simular oscuridad y lluvia (ver Figura 4.2).

Figura 4.1: Ejemplos de imágenes consideradas ID en el experimento de entorno operacional.

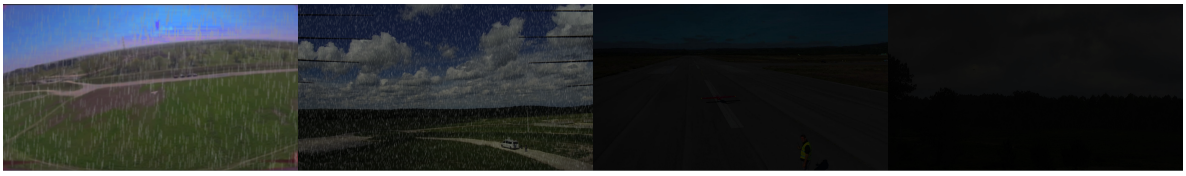


Figura 4.2: Ejemplos de imágenes consideradas OOD en el experimento de entorno operacional.

Utilizando un subconjunto de entrenamiento del conjunto de datos ID, se ha entrenado una red neuronal de clasificación de imágenes denominada ResNet-18. Las métricas se calcularon aplicando exclusivamente este algoritmo, ya que, como se demostró en el apartado anterior, la capacidad de clasificación de los demás algoritmos es nula. Finalmente, se obtuvieron las métricas representadas en la Tabla 4.4, diferenciadas según la situación del contexto operacional analizada. Se observa que el algoritmo seleccionado es capaz de diferenciar las imágenes con un rendimiento elevado, consiguiendo en ambas situaciones una tasa de falsos positivos cercana al 0 % cuando la tasa de verdaderos positivos es del 95 %. Estos resultados destacan la eficacia del algoritmo *Energy fine-tuned* en la clasificación precisa y confiable de imágenes, incluso en contextos operacionales variables.

4.2.2. Reconocimiento de objetos voladores

El último experimento realizado durante este trabajo de investigación tiene como objetivo demostrar si alguno de los algortimos presentados tiene la capacidad de diferenciar, de entre un conjunto de

	OOD	Oscuridad	Lluvia
Energy Fine Tuned	**FPR/TPR95**	0 %	0.4 %
	AUROC	0.999	0.998

Cuadro 4.4: Tabla de métricas obtenidas en el experimento de detección de cambios en el entorno operacional en un conjunto de datos de clasificación de fondos que se pueden ver en un entorno aéreo.

imágenes de objetos voladores, cuáles corresponden a las categorías usadas para el entrenamiento de una red neuronal y cuáles se desvían de estas categorías.

Para diseñar un experimento en el que se demuestre lo anterior se necesitan imágenes en entornos aéreos, que son difíciles de encontrar en conjuntos de datos públicos, motivo por el cual se decide utilizar un conjunto de imágenes propiedad de la empresa Gradiant, que contiene fotografías de distintos modelos de drones.

En este experimento se reparte el conjunto de datos mencionado de la siguiente manera:

- Flying Objects ID: 5500 imágenes de cada uno de los modelos de drones dji matrice 210, dji matrice 600, dji mavic pro, dji phantom y parrot disco. Se pueden ver ejemplos de estos modelos en la Figura 4.3.
- Flying Objects OOD: 6900 imágenes de otros objetos voladores no anotados, incluyendo imágenes de drones de modelos no mencionados en el punto anterior (ver ejemplos en Figura 4.4).

Figura 4.3: Ejemplos de los distintos modelos de drones incluidos en el conjunto de datos *In-Distribution*.

Figura 4.4: Ejemplos de los distintos modelos de drones incluidos en el conjunto de datos *Out-of-Distribution*.

Usando el conjunto de datos Flying Objects ID se entrena una red neuronal de clasificación de imágenes WideResNet, y con ella se aplican los algoritmos OOD, utilizando como OOD una muestra aleatoria de 5000 imágenes del conjunto de datos Flying Objects OOD, obteniendo así los resultados presentados en la Tabla 4.5

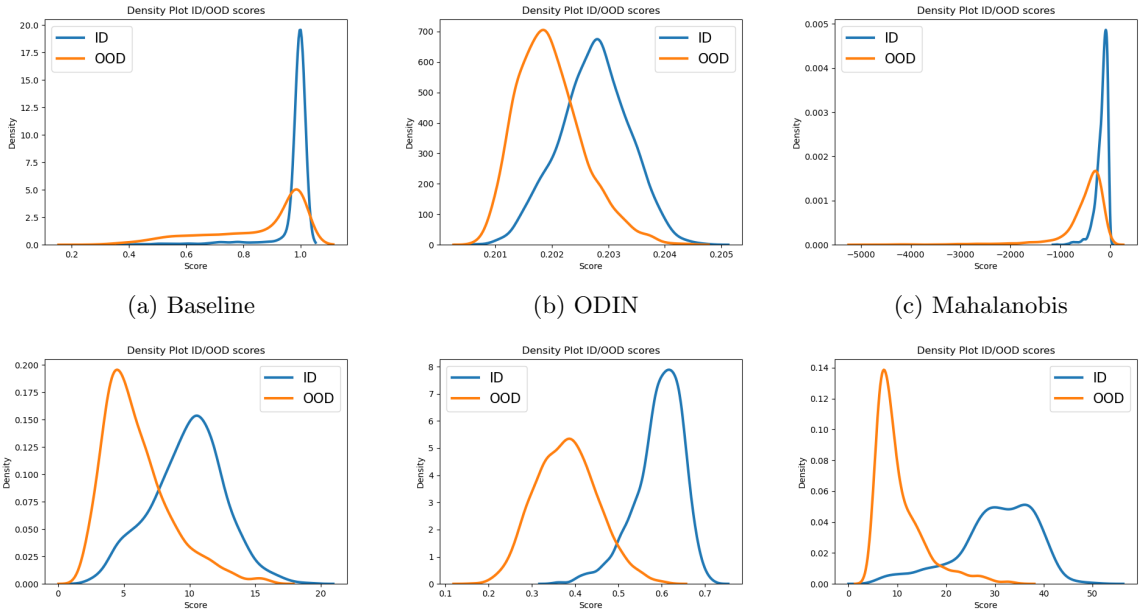
A pesar de ser un contexto donde aparentemente es más compleja la diferenciación entre imágenes ID y OOD, los algoritmos *Generalized ODIN* y *Energy fine-tuned* consiguen mejorar considerablemente las métricas con respecto al método de referencia. En cambio, los demás métodos no consiguen distinguir correctamente los dos tipos de imágenes.

Es pertinente resaltar el resultado obtenido con el algoritmo *Generalized ODIN*. Como se mencionó en el apartado 3.2.3, este método no requiere muestras OOD para su reentrenamiento ni para la calibración de los hiperparámetros. Esta característica lo convierte en la opción más prometedora para la identificación de objetos voladores que no pueden clasificarse en ninguna de las categorías incluidas en el conjunto de datos de entrenamiento. La dificultad de preparar un conjunto de imágenes que abarque todo el espectro de posibles objetos que un dron puede encontrar durante su operación hace que la adaptabilidad de *Generalized ODIN* sea particularmente valiosa.

	ID	**Flying Objects Train**
	OOD	**Flying Objects OOD**
Baseline	**AUROC**	0.84
	FPR/TPR95	67,9 %
ODIN	**AUROC**	0.83
	FPR/TPR95	61,75 %
Mahalanobis	**AUROC**	0.90
	FPR/TPR95	50,1 %
Energy	**AUROC**	0.84
	FPR/TPR95	62,75 %
Generalized ODIN	**AUROC**	0.96
	FPR/TPR95	14,07 %
Energy Fine-Tuned (Flying Objects OOD)	**AUROC**	0.96
	FPR/TPR95	20,7 %

Cuadro 4.5: Tabla comparativa de las métricas de clasificación de los distintos algoritmos OOD presentados. En el caso del algoritmo *Energy fine-tuned*, el reentrenamiento se realiza utilizando como muestra OOD una submuestra del conjunto de datos Flying objects OOD.

Además de las métricas observadas en cada uno de los experimentos, tambien es interesante observar como se separan las densidades de las puntuaciones OOD de los conjuntos de datos OOD de las



(a) Baseline (b) ODIN (c) Mahalanobis

(d) Energy (e) Generalized ODIN (f) Energy fine-tuned

Figura 4.5: Representación de la densidad de las puntuaciones OOD separadas por tipo de imagen (ID, OOD), para cada uno de los métodos presentados en este trabajo. Se utiliza como imágenes ID una muestra del conjunto de datos Flying Objects ID, y como imágenes OOD una muestra del conjunto de datos Flying Objets OOD.

puntuaciones ODD del conjunto ID utilizando los distintos métodos presentados en el trabajo. En la Figura 4.5 se puede observar a simple vista que los algoritmos que se proporcionan mejores resultados en cuanto a métricas consiguen una mayor separación de las puntuaciones OOD entre poblaciones.

Capítulo 5

Conclusiones y trabajo futuro

En este capítulo se presenta una revisión exhaustiva de las conclusiones derivadas de los experimentos realizados, conectándolas con los objetivos establecidos al inicio del trabajo.

Por un lado, se ha conseguido abordar la problemática de la clasificación incorrecta de objetos detectados en el aire. Los resultados del experimento 4.2.2 han demostrado que el algoritmo *Generalized ODIN* es altamente eficaz en la identificación de objetos que no pertenecen a ninguna de las clases presentes en el conjunto de entrenamiento. Este hallazgo es fundamental para alcanzar el primer objetivo mencionado en el apartado 1.4. Se ha verificado que es posible entrenar una red neuronal con imágenes de diferentes aeronaves (drones, aviones, helicópteros, etc.) y aplicar este algoritmo para detectar objetos aéreos no contemplados durante el entrenamiento, como aves o alas delta. Esto incrementa significativamente la seguridad del vuelo al alertar sobre posibles colisiones con elementos no previstos.

Por otro lado, el algoritmo *Energy fine-tuned* ha demostrado ser superior a otros métodos, logrando resultados excelentes en la identificación de imágenes con lluvia o con intensidades de luz inapropiadas para el vuelo. Este resultado valida el objetivo de identificar situaciones operacionales que no cumplen con las especificaciones necesarias para el vuelo seguro de un UAV. La capacidad del algoritmo para alertar sobre condiciones anómalas como la falta o exceso de luz es crucial para mantener la seguridad del dron en diversos escenarios operacionales.

Como trabajo futuro, se proponen varias líneas de investigación y desarrollo para mejorar y expandir el sistema de detección y evasión. En primer lugar, se sugiere utilizar el algoritmo *Energy fine-tuned* para detectar una gama más amplia de situaciones anómalas de manera generalizada. Esto se puede lograr mediante el reentrenamiento del algoritmo con imágenes obtenidas a través de técnicas de aumento de datos que representen condiciones como niebla, lluvia y desenfoque. Desarrollar un sistema capaz de identificar cualquier tipo de situación adversa que pueda surgir durante el vuelo es esencial para garantizar la seguridad operativa en todo momento.

Asimismo, es importante realizar pruebas con otras redes neuronales de clasificación para mejorar la robustez y precisión del sistema. Evaluar diferentes arquitecturas y métodos de entrenamiento permitirá identificar la combinación óptima que maximice la precisión y minimice los errores en la detección de imágenes *Out-of-Distribution*. Además, resultaría de interés conseguir nuevos conjuntos de datos para realizar experimentos que permitan validar que los algoritmos de detección de imágenes OOD elegidos como prometedores son robustos ante nuevas situraciones no consideradas en los experimentos realizados.

Finalmente, es crucial integrar estos algoritmos en el sistema *Detect and Avoid* y desplegarlo en entornos operacionales reales. Esto no solo ayudará a verificar la eficacia de los algoritmos en condiciones prácticas, sino que también permitirá identificar y corregir posibles limitaciones, mejorando continuamente el sistema.

El uso de algoritmos de detección de imágenes fuera de distribución (OOD) ha demostrado mejorar significativamente la seguridad y la robustez en las tareas de clasificación de imágenes mediante redes

neuronales. Aunque en este trabajo se ha aplicado al contexto aeronáutico, las lecciones aprendidas pueden extenderse a otros ámbitos donde una decisión incorrecta podría dar lugar a situaciones peligrosas. Esto destaca la importancia de estos avances en inteligencia artificial para la seguridad y eficiencia operativa en una amplia variedad de aplicaciones.

Bibliografía

[1] Concepto normativo p.13, https://www.seguridadaerea.gob.es/sites/default/files/Curso.Formacion.A1.A3.Completo.v10.pdf

[2] Muñoz, César & Narkawicz, Anthony & Hagen, George & Upchurch, Jason & Dutle, Aaron & Consiglio, María & Chamberlain, James(2015) DAIDALUS: Detect and Avoid Alerting Logic for Unmanned Systems. IEEE/AIAA 34th Digital Avionics Systems Conference (DASC), Prague, Czech Republic, 2015, pp. 5A1-1-5A1-12, doi: 10.1109/DASC.2015.7311421.

[3] EASA Artificial Intelligence Roadmap 2.0. https://www.easa.europa.eu/en/document-library/general-publications/easa-artificial-intelligence-roadmap-20

[4] Huang, Ziyi & Lam, Henry & Zhang, Haofeng. (2023). Quantifying Epistemic Uncertainty in Deep Learning. *arXiv preprint arXiv:2110.12122*

[5] Jörg Martin & Clemens Elster (2022). Aleatoric Uncertainty for Errors-in-Variables Models in Deep Regression. Neural Processing Letters. 55. 1-20. 10.1007/s11063-022-11066-3.

[6] Gawlikowski, Jakob & Tassi, Cedrique & Ali, Mohsin & Lee, Jongseok & Humt, Matthias & Feng, Jianxiang & Kruspe, Anna & Triebel, Rudolph & Jung, Peter & Roscher, Ribana & Shahzad, Muhammad & Yang, Wen & Bamler, Richard & Zhu, Xiao. (2023). A survey of uncertainty in deep neural networks. Artificial Intelligence Review. 56. 1-77. 10.1007/s10462-023-10562-9.

[7] Gross, Jason & Jones, Kennie. (2014). Reducing Size, Weight, and Power (SWaP) of Perception Systems in Small Autonomous Aerial Systems. AIAA AVIATION 2014 -14th AIAA Aviation Technology, Integration, and Operations Conference. 10.2514/6.2014-2705

[8] EASA Artificial Intelligence Concept Paper Issue 2. https://www.easa.europa.eu/en/document-library/general-publications/easa-artificial-intelligence-concept-paper-issue-2

[9] Li, Zewen & Liu, Fan & Yang, Wenjie & Peng, Shouheng & Zhou, Jun. (2021). A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects. IEEE Transactions on Neural Networks and Learning Systems. PP. 1-21. 10.1109/TNNLS.2021.3084827.

[10] Kh. Nafizul Haque (2023). What is Convolutional Neural Network — CNN (Deep Learning) https://www.linkedin.com/pulse/what-convolutional-neural-network-cnn-deep-learning-nafiz-shahriar

[11] Chirstopher M. Bishop (2006). Pattern recognition and machine learning. Editorial Springer

[12] K. He & X. Zhang & S. Ren & J. Sun & (2016). Deep Residual Learning for Image Recognition. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

[13] Ramzan, Farheen & Khan, Muhammad Usman & Rehmat, Asim & Iqbal, Sajid & Saba, Tanzila & Rehman, Amjad & Mehmood, Zahid. (2019). A Deep Learning Approach for Automated Diagnosis and Multi-Class Classification of Alzheimer's Disease Stages Using Resting-State fMRI and Residual Neural Networks. Journal of Medical Systems. 44. 10.1007/s10916-019-1475-2.

[14] Sergey Zagoruyko & Nikos Komodakis (2017). Wide Residual Networks. *arXiv preprint 1605.07146*

[15] Understanding Backpropagation Algorithm. https://towardsdatascience.com/understanding-backpropagation-algorithm-7bb3aa2f95fd

[16] Shorten, C. & Khoshgoftaar & T.M. (2019). A survey on Image Data Augmentation for Deep Learning. J Big Data 6, 60 (2019).

[17] Zou KH & O'Malley AJ & Mauri L. (2007). Receiver-operating characteristic analysis for evaluating diagnostic tests and predictive models. Circulation. 2007 Feb 6;115(5):654-7. doi: 10.1161/CIRCULATIONAHA.105.594929. PMID: 17283280.

[18] Jingkang Yang & Kaiyang Zhou & Yixuan Li & Ziwei Liu (2024). Generalized Out-of-Distribution Detection: A Survey. *arXiv preprint arXiv:2110.11334*

[19] Manpreet Singh Minhas & John Zelek (2019). Anomaly Detection in Images. *arXiv preprint arXiv:2110.11334*

[20] Marco A.F. Pimentel & David A. Clifton & Lei Clifton & Lionel Tarassenko (2014). A review of novelty detection. *Signal Processing* (Vol. 99)

[21] Yakovlev, K. & Bekkouch, I.E.I. & Khan, A.M. & Khattak, A.M. (2021). Abstraction-Based Outlier Detection for Image Data. In: Arai, K. & Kapoor, S. & Bhatia, R. (eds) Intelligent Systems and Applications. IntelliSys 2020. Advances in Intelligent Systems and Computing, vol 1250. Springer, Cham.

[22] G. Kumar & P. K. Bhatia (2014). A Detailed Review of Feature Extraction in Image Processing Systems. Fourth International Conference on Advanced Computing & Communication Technologies, Rohtak, India, 2014, pp. 5-12

[23] Szegedy, Christian & Zaremba, Wojciech & Sutskever, Ilya & Bruna, Joan & Erhan, Dumitru & Goodfellow, Ian & Fergus, Rob. (2013). Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*

[24] D'Angelo, Francesco & Henning, Christian. (2021). On out-of-distribution detection with Bayesian neural networks. *arXiv preprint arXiv:2110.06020*

[25] Shiyu Liang & Yixuan Li & R. Srikant (2020) Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks. Published as a conference paper at ICLR 2018. *arXiv preprint arXiv:1706.02690*

[26] Geoffrey Hinton & Oriol Vinyals & Jeff Dean (2015). Distilling the Knowledge in a Neural Network. *arXiv preprint arXiv:1706.02690*

[27] Chakraborty, A.& Alam, M.& Dey, V.& Chattopadhyay, A. & Mukhopadhyay, D. (2021), A survey on adversarial attacks and defences. CAAI Trans. Intell. Technol, 6: 25-45.

[28] Kimin Lee & Kibok Lee & Honglak Lee & Jinwoo Shin (2018). A Simple Unified Framework for Detecting Out-of-Distribution Samples and Adversarial Attacks. *arXiv preprint arXiv:1807.03888*

[29] Yen-Chang Hsu & Yilin Shen & Hongxia Jin & Zsolt Kira (2020) Generalized ODIN: Detecting Out-of-distribution Image without Learning from Out-of-distribution Data. *arXiv preprint arXiv:2002.11297*

[30] Weitang Liu & Xiaoyun Wang & John D. Owens & Yixuan Li (2021). Energy-based Out-of-distribution Detection. *arXiv preprint arXiv:2010.03759*

[31] Song, Yang & Kingma, Diederik. (2021). How to Train Your Energy-Based Models. *arXiv preprint arXiv:2101.03288*

[32] Krizhevsky, A. & Nair, V. & Hinton, G. (2014). The CIFAR-10 dataset. online: http://www. cs. toronto. edu/kriz/cifar. html, 55(5), 2.

[33] Deng, Jia & Dong, Wei & Socher, Richard & Li, Li-Jia & Li, Kai & Li, Fei-Fei. (2009). ImageNet: a Large-Scale Hierarchical Image Database. IEEE Conference on Computer Vision and Pattern Recognition. 248-255. 10.1109/CVPR.2009.5206848.

[34] Netzer, Yuval & Wang, Tao & Coates, Adam & Bissacco, Alessandro & Wu, Bo & Ng, Andrew. (2011). Reading Digits in Natural Images with Unsupervised Feature Learning. NIPS.